KEYPUNCH KEYTAPE & KEYDISC

A Complete & Concise Manual for Data-Processing Students

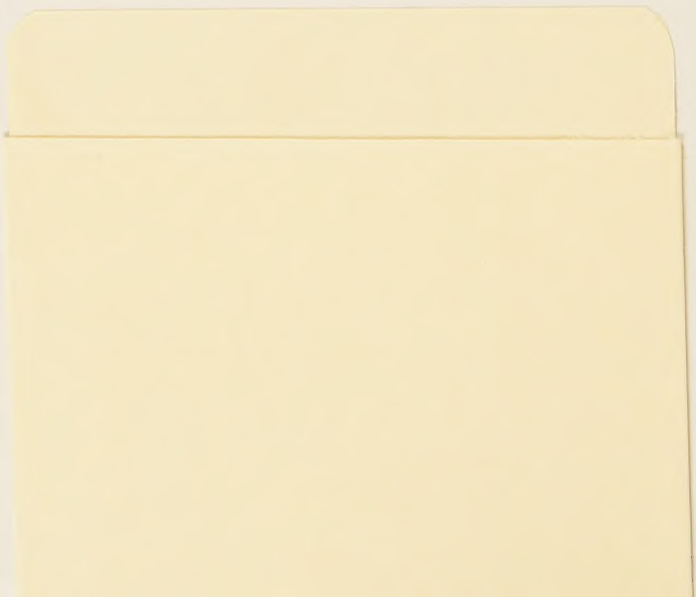
J. Blum

Senior Research Associate
Lever Bros. Co. Research Center
Edgewater N.J.

Esther L. Blum

Verifier, Port of New York & New Jersey Authority, New York, N.Y.

GORDON AND BREACH



KEYPUNCH, KEYTAPE AND KEYDISC

Keypunch, Keytape and Keydisc

J. BLUM

Senior Research Associate, Lever Bros. Co. Research Centre, Edgewater, N.J.

ESTHER L. BLUM

Verifier, Port of New York and New Jersey Authority, New York, N.Y.

with two chapters co-authored by

LESTER FOX

Senior Systems Analyst, Lever Bros. Co. New York, N.Y.

GORDON AND BREACH

New York London Paris

Editorial office for the United Kingdom
Gordon and Breach Science Publishers, Ltd.
41/42 William IV Street
London W.C.2.

Editorial office for France
Gordon & Breach
7-9 rue Emile Dubois
Paris 75014

Library of Congress catalog card number 72-78322.

ISBN 0 677 03950 6.

Printed in Great Britain.

To Claude, Andrea and Deborah

Contents

*By Lester Fox and J. Blum.
†By Jaime Blum and Lester Fox.

Erratum

The figure on page x should have the following caption:

Card Puncher 024
Reprinted by permission from IBM

Introduction

This basic instruction manual for card punches and key tape has been specifically prepared for those students who are about to study electronic data processing (EDP). One of the most important facets of EDP is the transmission to the computer of information coded in a "language" that the computer "understands".

Since 1945 the use of computers has experienced a tremendous surge. This increase in the processing of information is due to the incredible speed with which these machines carry out a complete series of operations that previously were effected manually. For this reason, human productivity has increased spectacularly. In order, however, to take advantage of the numerous possibilities of these electronic marvels, one must learn the system or language by which the computer understands the orders it receives in the processing of millions of data.

We "talk to the computer" by means of punched cards or magnetic tapes. The information carried in the form of rectangular holes (punches), or magnetic pulses, is transcribed to tape in a special converter. The speed with which these converters "read" the information is ridiculously slow compared with the speed of the electronic equipment. For example. The punched card has 80 columns and there are 12 positions which can be punched in each card. This means that 960 digits can be punched per card. Assuming there is only one digit per column, reading at the rate of one hundred cards per minute, the converter accepts 8,000 bits of data per minute or 130 per second. "Reading" cards at the rate of 240 per minute, the average is 320 digits per second. These two rates, which were mentioned only as an example, do not tax the capability of the converter since its reading speed is much higher. Speeds of 2,000 to 15,000 digits per second are commonplace, but much higher speeds can be found in newer equipment.

The job that is carried out when the data has been transferred from the punched card to tape, naturally depends on the job itself, on how it has been organized, the nature of the mathematics involved, what calculations are to be performed, what decisions are to be taken, etc. All these tasks which only a few years ago required much time, and a lot of written and oral communication, are now carried out automatically, without human involvement, without error and with amazing speed.

The growing importance of electronic data processing derives from the place the computer has taken in our daily lives. In consequence, the stature of the cardpunch operator has also grown since her services are essential to "talk to the computer".

The intention of the authors of this manual is to guide the student to know the meaning of the job being done. Sufficient material has been included for learning to handle the cardpunch and the keytape, and for developing the necessary ability to create programs with the adequate field distribution into which the programs will be divided, and the correct number of columns to be assigned to each of these fields. This text is, therefore, a basic manual which will have to be used primarily as a reference textbook in combination with "hands-on" instruction directly on the machines. Its simple format will allow the student to follow the reference manuals published by the hardware manufacturers without difficulty.

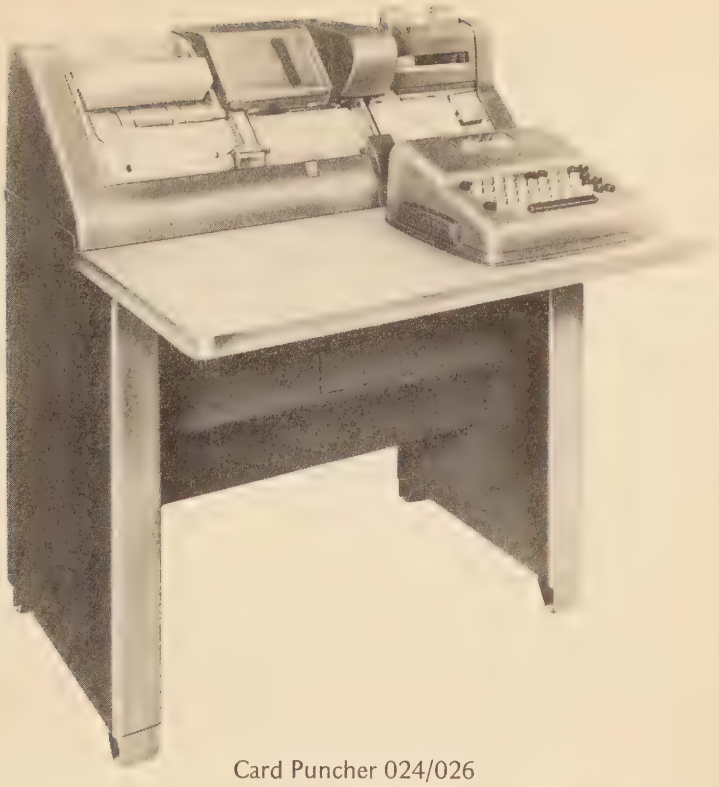
We have received excellent co-operation from hardware manufacturers in the presentation of particular chapters. In particular, we would like to express our appreciation for encouragement received and permission to reproduce material from their manuals, to the following companies:

IBM, Chapters One through Eighteen
Mohawk Data Sciences, Chapters Twenty One and Twenty Two
Inforex, Inc., Chapter Twenty Three

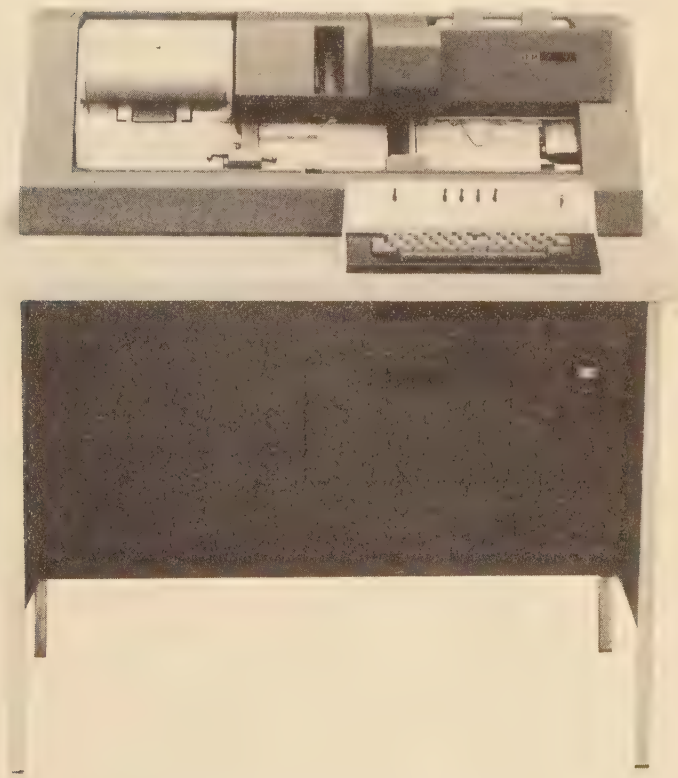
Their willingness to supply educational materials for their respective topics has been extremely reassuring.

We also wish to express our thanks to Mrs. Harriette Beyea and Mr. Fred Rubin for their editorial assistance; and to Mr. Louis Garcia, of Operation Analysis (Lever Research) for his help in preparing the computer programs.

Leonia, New Jersey



Card Puncher 024/026
Reprinted by permission from IBM



Card Puncher 029
Reprinted by permission from IBM

129 Card Data Recorder

IBM 5496 Data Recorder

1 The card

The cards used in all the jobs measure 7 3/8 X 3 1/4". These cards have slanted upper corner cuts on the right or left corners which enable the operator to observe if all the cards are properly placed, that is facing the operator.

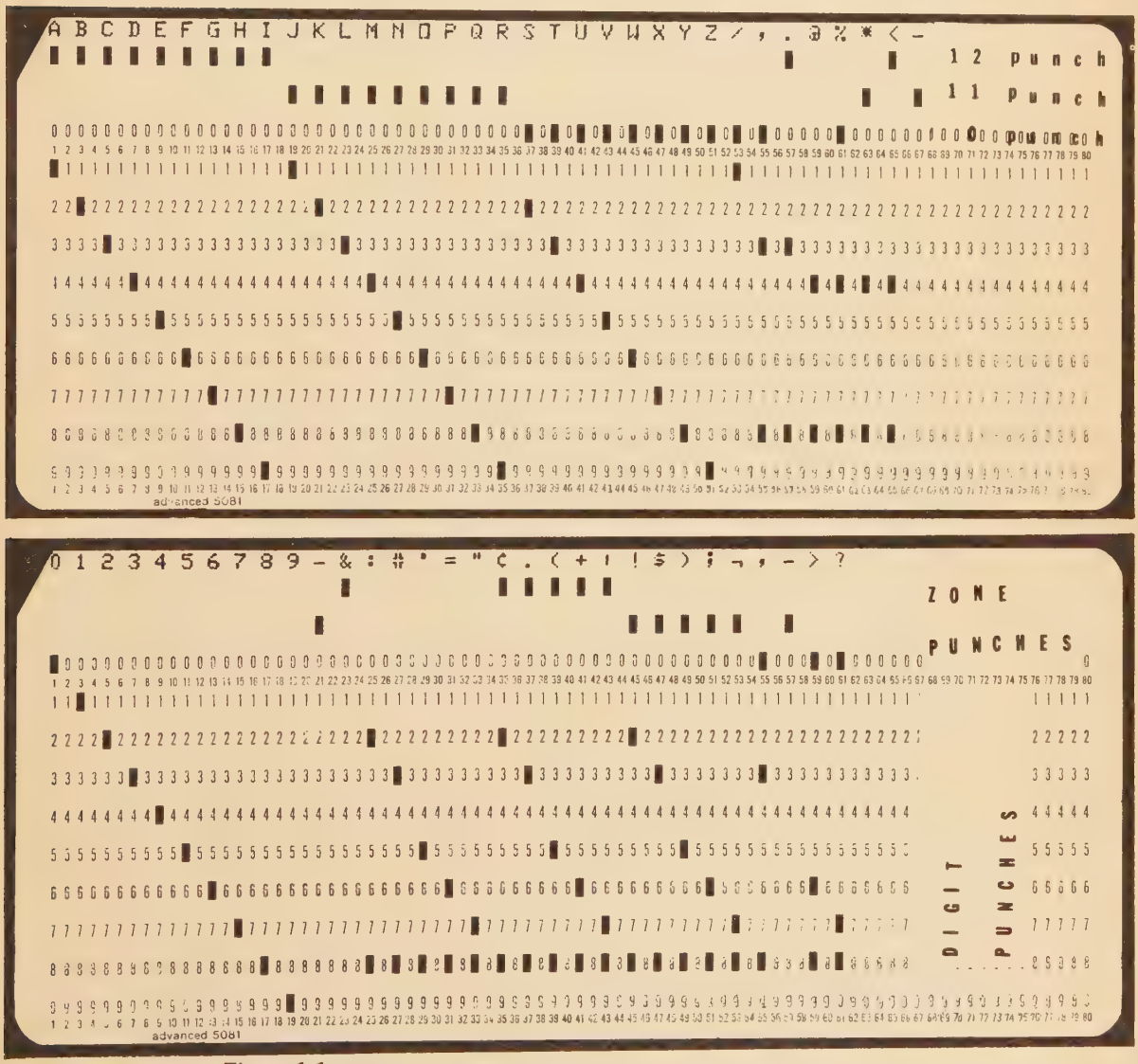


Figure 1.1

All the cards have eighty vertical columns which are identified by their respective column number. These column numbers are printed in a series of horizontal rows numbered from one to eighty. In the cards shown in the previous figure, these rows are (a) between the rows of digits, 0 and 1, and (b) immediately below the row of digit 9. These column numbers are extremely useful, especially when preparing the program card.

Each column has twelve punching positions: one for each of the digits from one to nine (nine positions in all), and three extra positions above the digit one, in the positions that are called "zone punches", which are zones 0, 11 and 12. Punches of digits one through nine are called "digit punches". The eleven-zone punch above the zero-number digit, is sometimes referred to as the "X-punch".

Letters require two punches per column, one zone punch and one digit punch. For instance, the letter A is a combination of two punches, one in the twelve zone and the other in digit one; the letter J is a combination of a punch in the eleven zone and digit one; the letter S is a combination of a punch in the zero zone and digit two. Special characters require two and three punches per column: the character / (slash) is a combination of the zero zone punch and of digit one. The character $ (dollars) is a combination of a punch in the eleven zone and two digits, eight and three. Digits from one through nine require just one punch per column.

This combination of punches to represent numbers, letters, and other characters was invented by Herman Hollerith ("An electric tabulating system", *School of Mines, Quarterly,* **10**, #3, April, 1889, pp. 238-255), and was used for the first time in the Population Census of New York City in 1890.

Table 1.1 below, shows all the combinations that can be made with the card punches when these machines are operated in the alphabetic or numeric shifts. The first and second columns of Table 1.1 list the holes which are punched on the card and the graphic representation of such punches when the cardpunch is in the alphabetic shift. The third and fourth columns show the same representation but with the cardpunch in numeric shift.

Table 1.1

Alphabetic shift		Numeric shift	
Punches at	Graphic representation	Punches at	Graphic representation
12-1	A	0	0
12-2	B	1	1
12-3	C	2	2
12-4	D	3	3
12-5	E	4	4
12-6	F	5	5
12-7	G	6	6
12-8	H	7	7
12-9	I	8	8
11-1	J	9	9
11-2	K	11	– (minus or dash)

(cont.)

11-3	L	12	& (ampersand)
11-4	M	8-2*	: (colon)
11-5	N	0-8-5*	_ (underscore)
11-6	O	0-8-6*	> (more than)
11-7	P	0-8-7*	? (question mark)
11-8	Q	0-8-3	, (comma)
11-9	R	8-3	# (number sign)
0-2	S	11-8-3	$ (dollar sign)
0-3	T	12-8-3	. (period)
0-4	U	12-8-6*	+ (plus sign)
0-5	V	11-8-5*	) (right parenthesis)
0-6	W	12-8-2*	¢ (cent sign)
0-7	X	12-8-7*	| (logical)
0-8	Y	11-8-6*	; (semi-colon)
0-9	Z	11-8-7*	¬ (logical NOT)
0-1	/ (slash)	8-5*	' (apostrophe)
0-8-3	, (comma)	8-7*	" (quote mark)
12-8-3	. (period)	8-6*	= (equal)
8-4	@ (at sign)	11-8-2*	! (exclamation mark)
0-8-4	% (per cent sign)	12-8-5*	((left parenthesis)
11-8-4	* (asterisk)	0-8-2†	
12-8-4	< (less than sign)		
11	– (minus or dash)		

*Graphic not available in the 024/026.

†This code does not yet have a graphic representation.

All these punches are produced automatically when the operator depresses the proper key on the keyboard and, as mentioned before, the holes which will be made will depend on whether the cardpunch is in alphabetical shift or in numeric shift. The concept of shift is a direct application of the shift key on the typewriter keyboard. The cardpunch operates in the same fashion: while normally it is in the alphabetic shift, depressing the NUM key will place the punchcard in the numeric shift.

When a group of cards have been punched, the accuracy of such punches has to be checked. Another machine, called the verifier, is used for this job. If the card has been punched without errors, it automatically becomes a permanent entity to be used in a large variety of mathematical processes generally entailing the use of the computer.

In order to transcribe hand-written information into punches in the card, the eighty columns in the card are bunched in fields. The number of columns assigned to each field will naturally depend on each particular job. The operator should always keep in mind that, when certain information is punched in certain columns in a certain field, that information will be exclusively punched in those columns of that field and in no other for all the cards in that particular job. In this fashion, all the information present in one card which is part of a job will be easily identified and therefore easily duplicated from card to card in the same job.

Figure 1.2 shows a bill of sale and underneath, one card with information taken from that invoice. A great variety of data is shown on the card, all of that information derived from the invoice: the customer's name between

GENERAL MANUFACTURING COMPANY

ENDICOTT, N. Y.

CUSTOMER'S ORDER NO. 311 INVOICE DATE 12-31 INVOICE NO. 12349

SOLD TO New Mexico Company
216 Wysor Building
Houston, Texas

CUSTOMER NO. 59751

SHIP TO Above

Make all checks payable to

GENERAL MANUFACTURING COMPANY
Endicott, N. Y

SALESMAN Macy-67

SHIPPED VIA Truck Prepaid

TERMS 2% 10 Days Net 30

QUANTITY	COMMODITY No.	DESCRIPTION	PRICE	AMOUNT
		Casters		
40	11202	Sq. Shank Swivel	.83	33.20
75	13102	Flat Top Rigid	.84	63.00
5	17203	Ext. Shank with Brk.	1.62	8.10
2	32105	Bolt and Nut Shank	2.64	5.28
4	44104	Rnd. Spr. Ring Stem	3.51	14.04
40	62110	Bolt and Nut Shank	7.25	290.00
		Freight		.78
				414.40

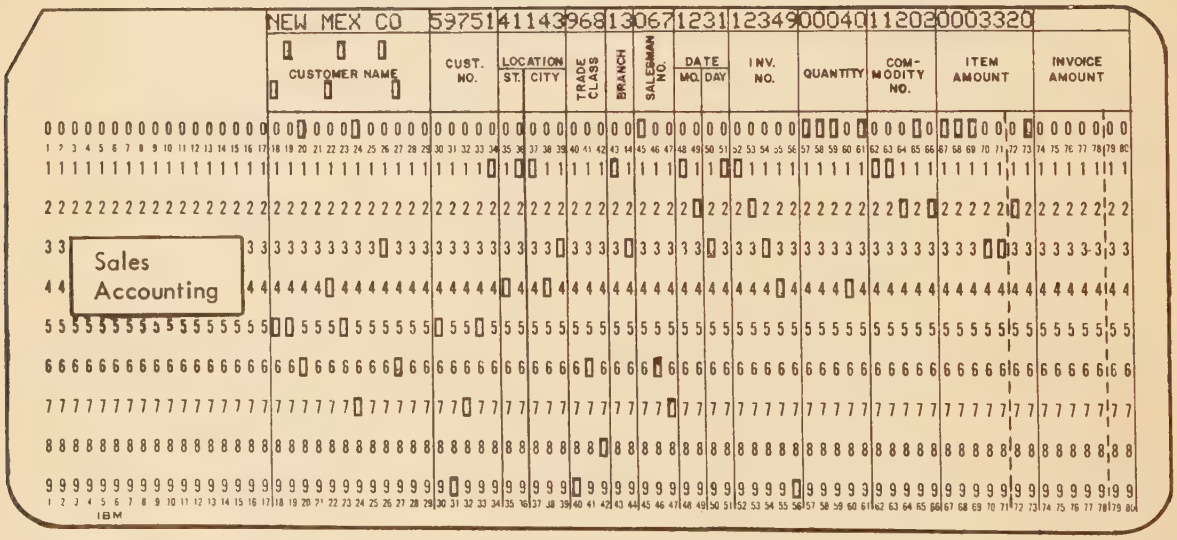


Figure 1.2 Reprinted by permission from IBM

columns 18 and 29; customer's number in coded form (30-34); its location by coded state and city (35-39); trade class (40-42); branch (43-44); salesman number (45-47); date, month and day (48-51); invoice number (52-56); quantity (57-61); commodity number (62-66); item amount (67-73) and invoice amount (74-80). The first seventeen columns of that card are left blank in this job for special accounting purposes. The groups of columns, 18-29; 30-34; 35-39; etc., are fields of information and in this accounting job the same information will always be punched in the same fields and *not in any others*. Consequently, the second card will carry information on the second commodity of the invoice; the information punched through column 56 will also be shown since it is common to all the cards in this job. This information, identical in all cards, will be automatically duplicated from card to card without any intervention on the part of the operator because the card punch has already received instructions in a coded manner. These instructions are punched on another card, the program card, which is placed on the program drum. The preparation of the program card is essential and we will study this subject in depth later. In the meanwhile, the operator should be aware of the fact that field distribution and the number of columns in each field will always be the same in *each job*. When, however, the operator punches another job such as computation of salesmen's commissions or inventories, the whole set-up will change, and the program card, the master card, the field distribution and the number of columns assigned to each field will also change.

In this particular punching job, the invoice, the operator should note that the fields following column 56 carry information which vary from commodity to commodity and consequently one card per commodity will be needed. The information to be punched will vary from card to card, reflecting the type of commodity involved.

QUESTIONS ABOUT CHAPTER 1

1 What is the purpose of the upper right (or left) hand corner-cuts in the card?

2 How many columns are there in a card?

3 How many punching positions are there in each column?

4 Name the three upper punching positions.

5 What are the other positions called?

6 The "11" punched above the row of zeros is also known as — punch?

7 How many punches per column do the digits require?

8 The alphabetic characters need — punches per column?

9 How many punches per column do the special characters require?

10 Using the Hollerith code, write the letter D.

11 Using the Hollerith code, write the letter O and the letter S.

12 Using the Hollerith code, write your full name and address.

13 What does the verifier do?

14 Define a field.

KEYBOARD EXERCISES

Do not space groups of four figures when keying. Space has been provided for easy reading only. If punching has been accurate, each two lines of figures will fill one card. Place a blank card on the drum and lower on it the sensing mechanism. Turn the FEED switch ON.

8137	8137	4295	4295	8319	8319	9037	9037	0428	0428
3179	3179	5263	5263	7529	7529	7642	7642	3796	3796
1780	1780	2706	2706	9997	9997	8493	8493	8264	8264
9731	9731	8927	8927	6350	6350	4900	4900	9531	9531
2444	2444	6829	6829	3777	3777	4786	4786	5891	5891
4321	4321	3800	3800	1929	1929	6018	6018	7403	7403
1984	1984	1278	1278	2737	2737	1497	1497	5238	5238
4719	4719	2560	2560	4010	4010	1040	1040	1140	1140
1000	1000	0100	0100	0010	0010	0001	0001	1001	1001
0020	0020	0200	0200	0220	0220	0221	0221	0122	0122
4000	4000	0800	0800	0500	0500	0400	0400	7000	7000
0060	0060	9000	9000	0090	0090	5751	5751	5723	5723
3906	3906	8522	8522	4722	4722	0963	0963	7242	7242
6903	6903	2285	2285	0692	0692	1274	1274	3725	3725
8014	8014	9863	9863	2614	2614	7803	7803	1385	1385
0460	0460	9125	9125	2174	2174	1682	1682	7794	7794
9648	9648	8659	8659	8549	8549	5894	5894	4895	4895
5689	5689	9865	9865	4589	4589	9854	9854	9745	9745
7197	7197	3407	3407	5382	5382	8570	8570	5716	5716
0590	0590	7346	7346	8137	8137	3934	3934	7731	7731
8591	8591	4053	4053	7071	7071	8428	8428	8041	8041
9180	9180	5963	5963	9396	9396	3004	3004	5428	5428
8382	8382	5336	5336	1910	1910	4773	4773	2387	2387
0128	0128	0737	0737	0849	0849	5058	5058	2390	2390
7364	7364	0188	0188	2613	2613	3874	3874	2540	2540
9743	9743	2847	2847	3980	3980	1790	1790	9017	9017
1397	1397	9317	9317	7456	7456	5486	5486	3719	3719
5846	5846	7485	7485	1739	1739	4756	4756	7675	7675
5463	5463	2625	2625	7091	7091	5921	5921	1641	1641
1892	1892	5564	5564	1397	1397	1295	1295	6445	6445
0609	0609	1741	1741	3624	3624	0963	0963	2582	2582
0603	0603	7147	7147	1729	1729	2152	2152	5838	5838
5228	5228	7654	7654	8765	8765	9876	9876	0987	0987
1323	1323	6390	6390	1234	1234	4321	4321	3030	3030
3212	3212	3213	3213	3214	3214	3215	3215	3616	3616
2852	2852	9096	9096	3074	3074	1256	1256	5270	5270
8563	8563	3215	3215	2852	2852	6429	6429	3727	3727
9572	4171	8374	2759	2528	2507	0272	6093	8729	1174
5729	3215	1174	3727	8563	2258	2795	8374	6093	7529

5327	1741	7529	6325	5228	1305	6093	5270	2037	1017
4171	5282	1917	3822	7302	7395	3960	3690	3822	2561
2059	2073	1147	2528	1255	0370	2059	2073	8522	4711
5228	6093	0727	3649	9735	6292	1741	2852	4528	7353
6492	2946	3960	3690	7257	5384	2076	9462	5822	3096
2757	4481	1592	6429	1417	7114	1430	7869	4692	2670
2964	0267	2852	1174	4712	0750	6207	1705	7411	8522
0623	3782	7315	5137	6309	4711	0649	7212	9534	1375
5228	3960	1055	4080	7364	7351	0369	4117	2345	2397
3715	9453	2528	1771	1429	9131	3194	8249	7411	8522
0421	7295	0538	8350	6309	3690	2152	1576	1913	3508
5282	7511	7512	0369	6138	0583	7411	9630	9065	4835
5038	1391	1147	2913	0918	6305	9131	3194	8249	8156
1147	2258	3420	2529	2961	6192	9036	5822	2934	1081
6852	1926	3096	4171	2804	9514	2905	2961	0963	1741
3690	1612	6717	6129	8625	2685	3027	4117	5228	2969
1928	7171	3728	8325	9306	3928	6482	7245	7245	3276
9371	0090	3276	7091	0971	0070	0004	1724	6418	0395
1471	2582	8291	8263	9060	4286	4268	1739	6491	9747
0900	1709	2016	0876	0404	7174	8285	2981	2783	0303
2781	2983	0303	9173	6428	3791	5683	9778	8070	0791
4429	0569	0082	0008	0037	1426	1771	2882	1892	3872
1872	3892	0369	6071	8246	1973	1770	2735	0708	1970
2735	3510	1890	0071	0500	2583	2884	1171	2282	1892
3872	1872	3892	3396	4196	8246	1973	4186	7508	0708
1970	2489	1652	0017	0050	0217	2213	1141	5981	5783
2552	5781	5983	3660	3074	4862	7391	0981	0599	6050
7091	0981	0599	1913	5708	0069	0007	1417	1895	3876
2528	1875	3895	3609	5274	2684	1937	1670	5383	0506
1907	0058	0020	3529	1437	2010	6836	1295	3275	7171
8262	1295	3275	7171	8382	9393	0314	1275	3295	2864
1397	0304	4619	2395	6393	4879	0063	0900	1480	0426
7415	5951	5753	8528	5751	5953	9603	2547	6842	9731
1946	1228	2010	9701	6071	2079	0052	0060	4592	4491
8048	0714	9269	2507	4104	0752	9829	3215	6093	1174
8564	8374	2258	6093	5327	6325	1741	5228	6093	8160
5270	1017	1917	4171	5282	3822	2561	3960	3690	1255
0370	1147	2528	4266	8522	2354	4711	2258	0702	3649
4528	7353	1741	2852	7257	5384	0639	5822	2757	4481
3096	1417	1430	7869	7114	3852	1174	4712	0750	6093
0332	2639	7411	8522	1232	2191	6309	4711	9504	1006
5228	3960	0822	1161	0369	4117	2528	2823	6197	2258
0421	7295	4117	5228	2152	1576	3096	9630	7512	0369
7114	2852	9036	9065	4835	2852	3096	2913	0918	4080
2969	1424	7114	8225	8846	1212	6903	1741	0416	7654
8225	5209	7469	6476	6309	1471	8225	3960	5931	0748
1174	2285	1448	1243	2041	1729	0963	2853	1527	7458
6903	7211	3526	3328	2705	4711	8522	1174	9281	3231
1514	4711	6822	3690	1090	9524	0369	7114	8522	9524
4862	3460	1969	8522	1147	9630	4117	7536	3023	3134
2095	3728	1728	4646	1641	2643	3536	6482	9371	0090
8090	8273	8271	6565	4161	4262	8293	4268	3536	1623

2 Operating features

Figures 2.1 and 2.2 show the operating features of the cardpunches 024/026 and 029, respectively.

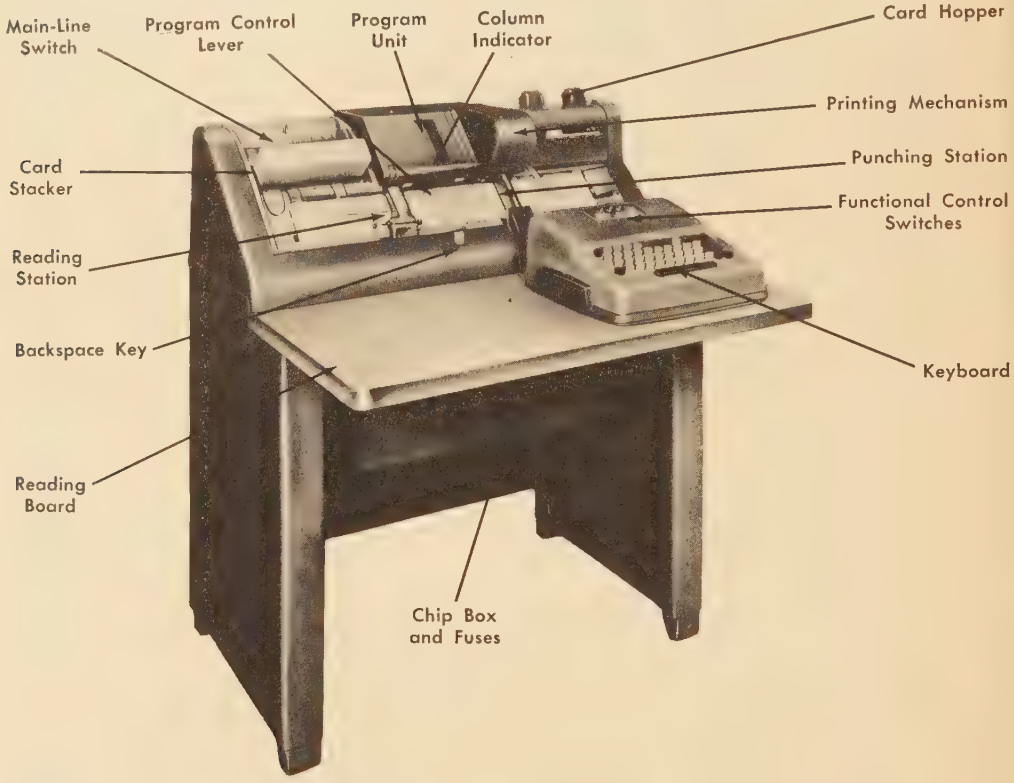


Figure 2.1 IBM 26 operating features

The cards to be punched are placed in the hopper. In the punching process, each card will pass in sequence from the hopper to the card bed, the punching station, the reading station and will reach the stacker passing, without stopping, through the card bed at the left.

The punched program card, on the program drum, commands the card-punch to perform a number of functions. The program control lever will activate the program unit when it is in the "ON" position, that is, to the left. The functional switches are above the keyboard on a panel and the backspace key, located immediately below the program unit, will backspace

Figure 2.2 IBM 29 operating features

the card continuously, one column at a time, until column one is reached.

The hopper has a storage capacity of about 500 cards. These are placed so they "face" the operator with a "9" row down.

When the key FEED is actuated, one card is fed down from the card bed. At the beginning of a new job, two cards have to be fed manually by depressing the FEED key twice. When the second card is fed, it will be placed under the first one and the first is automatically registered, ready to be punched. Feeding cards automatically is under the direct command of a functional control switch. It is for this reason that the operator, at the beginning of a new job and after keying FEED twice, moves the control switch AUTO FEED (on the panel) to the ON position.

While the first card is being punched, it will slowly move to the left, as the operator is keying. The second card, pre-registered below, does not move on the card bed. When the operator finishes the first card (or it is released by

keying REL), the first card moves to the card bed in the center (reading station) while the second one is automatically registered at the punching position. A new card, the third one, will be fed down, either by depressing FEED or automatically under control of the functional switch AUTO FEED. The third card will remain pre-registered at the punching station. This process will be repeated constantly during the punching job.

The reading station is to the left of the punching station. Every card that reaches the reading position does it in synchronization with the new card being punched at the punching station, and also in synchronization with the program card. That is, if the operator is keying, say, the tenth column of the second card, the first card is being "read" at the reading station on the tenth column. The reason is a very simple one: when the card punch is under control of the program card, information already punched at the tenth column of the first card may have to be duplicated on the second one. This is carried out under program control without the cumbersome task of re-keying the same information over and over again.

The program drum also moves in synchronization. However, the column indicator, at the base of the drum, points to the *next* column to be punched. That is, if the operator is punching the second card at the tenth column, the first card is at the reading station at the tenth column but the column indicator points to column 11.

After punching the second card, the first card will automatically pass to the stacker without stopping at the card bed at the left; the second card passes to the reading station; the third card, previously pre-registered, becomes registered and ready to be punched at the punching station, and a new card is fed from the hopper to become pre-registered at the punching station. The cards remain stored in the stacker in the same sequence in which they were punched, with their "back" to the operator, showing their un-imprinted side, their "12's" down. The stacker can also hold about five hundred cards. When there are more than 500 cards in the stacker it automatically shuts the machine off, indicating that the cards must be removed in order to continue the punching job. When the excess cards are removed, the automatic feed feature again becomes operational.

The backspace key is located immediately below the center card bed. Each time this key is actuated, the cards at the reading and punching station and the program drum will back one space. If this key is held down, the cards and the program drum will backspace continuously until column one is reached. It is, however, advisable not to:

a back-space if the card being punched is beyond column 78 unless the previously pre-registered card is removed;
b back-space more than 20 columns at one time, since the column register may be inaccurate.

In order to punch one single card, it is not necessary to load the hopper. Individual cards may be placed manually, one by one, at the punching station. In order to register a single card, key REG each time a single card is placed at the punching station.

Cards may also be removed one by one from either the reading or the punching stations by keying the REL (release) key. Another way is by holding down the pressure roll release lever while pulling the card out. Nevertheless, care should be exercised in removing a card. If, in the process, a card is torn and its pieces are caught at either station, those pieces can be pushed

out by using another card or by using a smooth edged metal blade while the pressure roll release lever is held down. Never use saw-edged metal blades for this purpose.

QUESTIONS ABOUT CHAPTER 2

1 In a brief sentence, describe the functions of the punching and reading stations.

2 In a brief sentence, describe how the cards are positioned in the hopper and in the stacker.

3 Where is the column indicator? To which column does it point?

4 How many cards can be placed in the hopper at one time? And in the stacker?

5 In a brief sentence, explain how the cards can be removed from the reading or punching stations.

6 At the beginning of a job, how many cards must be fed down from the hopper? How is this carried out?

7 In a punching job, indicate the sequence of positions through which a card passes.

8 What is the maximum number of columns which it is advisable to backspace?

3 The program drum

The program drum is a removable part of the cardpunch (see Figure 3.1). This drum is housed in the program unit at the upper center of the card punch, underneath a cover which hinges toward the operator. The only function of the program drum is to support the program card. Underneath the program drum and on the machine metal body, there is a butterfly lever, the program control lever, which controls the program drum sensing mechanism. In order to introduce and/or remove the drum, this lever must be in the OFF position, to the right. The introduction or removal of the drum must never be attempted when the lever is to the left, that is, in the ON position.

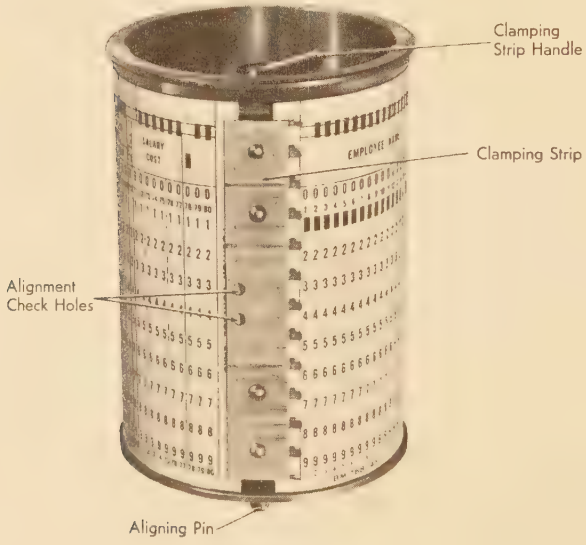


Figure 3.1 Program drum

The physical action produced when the lever is moved from right to left (OFF to the ON position) is simply the lowering of the sensing mechanisms which consist of 12 star wheels. These wheels are spaced in the same fashion as the 12 punching positions on a card. The drum revolves in step with the movement of the card and when one star finds a hole in the program card it closes the electric circuit by contacting the drum. This, in turn, produces an

electric impulse, the nature and effects of which will depend directly on the position of the punch on the program card.

If the wheels do not find any perforations (blank card, without punches) the sensing mechanism does not make contact with the drum. In this case, the only resulting effect is that a combined keyboard (alphanumeric) will function exclusively as a numeric keyboard: the depression of combined keys (that is keys which simultaneously carry digits and letters, or special characters) will punch numeric shift characters.

The operator should be aware that substantial damage will be inflicted upon the card punch if the sensing mechanism is lowered upon a drum which does not carry a card. Therefore, the sensing mechanism should only be lowered when the drum is in place and a card (punched or blank) is on it.

When the machine is performing under control of the program card, the cardpunch automatically carries out numerous functions in a much faster and more accurate fashion than if the same functions were to be performed manually.

Mounting the Card on the Drum

In order to mount a card on the drum, the following procedure should be followed:

1 Keep the drum in palm of your left hand with the clamping strip handle at top. Turn the handle inside the drum counterclockwise as fully as it will go. This will loosen the smooth edge of the clamping strip (the other edge is toothed).

2 Inset the card with the column-80 edge underneath the smooth strip so the "9" row is parallel to, and above, the lower rim of the drum. Two alignment holes make it possible to observe if the card is flush with the metal edge under the strip.

3 Turn the handle on the drum to the center. This tightens the smooth clamp on the card and loosens the toothed one.

4 Wrap the card tightly around the drum holding it in place with the left thumb. Insert the column-1 edge underneath the toothed edge of the pressure strip.

5 Turn the handle as far as it will go to the right, clockwise. This will tighten the toothed edge over the card.

6 The drum is now ready to be inserted in the program unit.

Card Removal

1 With the drum in the palm of your left hand and the handle on top, turn the handle halfway to the left. Remove the column-1 edge of the card from under the toothed pressure strip.

2 Now turn the handle fully counterclockwise and remove the card from the drum.

Drum Insertion

1 With the sensing mechanism raised, slide the drum onto the mounting shaft, positioned in such a fashion that the aligning pin will fall in the aligning hole in the column indicator dial.

2 Turn the butterfly lever to the left, to the ON position, so as to lower the sensing mechanism to the program card. Press the REL key to engage the reading mechanism.

Drum Removal

1 Raise the sensing mechanism by turning the butterfly lever to the right, OFF position.

2 Pull the drum out in a direction parallel to the sensing mechanism.

Immediately below the drum, on its base, there is a circular indicator numbered from 1 to 80. This indicator points to the column *following* the one being punched. For instance if the operator is keying column 10, the indicator points to 11. This indicator is particularly useful for column spacing or as a reference when backspacing.

4 Keyboards

There are three *basic* IBM cardpunches known as the IBM 024, the IBM 026 and the IBM 029. Within each of these categories, there are several models set up for specific applications.

Cardpunches 024 and 026 are practically identical. The only difference is that the 026 has a printing mechanism which simultaneously prints the graphic representation of the key being depressed. This printing is effected on the top edge of the card. The 024 does not have the printing mechanism. The 029 has other characteristics such as the unlimited use of two completely different programs, which can be punched on the same program card. The insertion of zeros to the left of significant digits (in numeric fields) and of special characters for computer programming, are other very important features of the 029. All these card punches have keyboards which can be totally numeric or a combination of numeric, alphabetic and special characters. This last type of keyboard is called alphanumeric.

In all keyboards the punching keys are round or square, of a grey color with letters, numbers or special characters imprinted in blue. The functional keys, which will be discussed later, are blue or black with white imprinting. The punching keys are:

1 exclusively alphabetic;
2 exclusively numeric;
3 combined.

It is from this classification that the combined keyboard is labelled alphanumeric.

The numeric keyboard in the 024/026 has 11 punching keys, 9 functional keys and a space bar. The numeric keyboard of either the 024/026 or the 029 is actuated with the fingers of the right hand, freeing the left hand to handle documents or papers on the desk. The keys for the 4, 5 and 6 digits are slightly more concave than the other keys in order to facilitate a better and faster "touch-system" keying. The three middle, right-hand fingers, properly placed over the keys, allow the operator a fast up and down movement of these fingers. In the alphanumeric keyboard of the 024/026 there are 33 punching keys (some of them combined), 11 functional keys and the space bar.

The numeric keyboard of the 029 has 12 punching keys, 12 functional keys and the space bar. The alphanumeric keyboard of the 029 has 34 punching keys, 14 functional keys and the space bar. The position of the fingers on these combined keyboards is similar to that mentioned before.

Figure 4.1 Numeric keyboard of the 024/026 card punch

The keys of Row A through key 6/L, slightly more concave than the other keys, are the "home keys". The touch system for the ten numeric keys is: the index finger for digits 1, 4 and 7, the middle finger for digits 2, 5 and 8 and the ring finger for digits 0, 3, 6 and 9. Figure 4.5 shows the position of the hands on a combined alphanumeric keyboard.

The combined keys will punch either a letter or a digit (or a special character) depending on whether the cardpunch is in alphabetic or numeric shift. For instance, pressing the 5/K key will produce a digit 5 punch if the keyboard is in the numeric shift but it will produce the holes corresponding to the letter K if it is in alphabetic shift. As mentioned before, the concept of shift is a direct application of the shift key on the typewriter keyboard which produces a lower or upper case letter. The difference is that a card punch can "shift" either automatically under program control or manually by key.

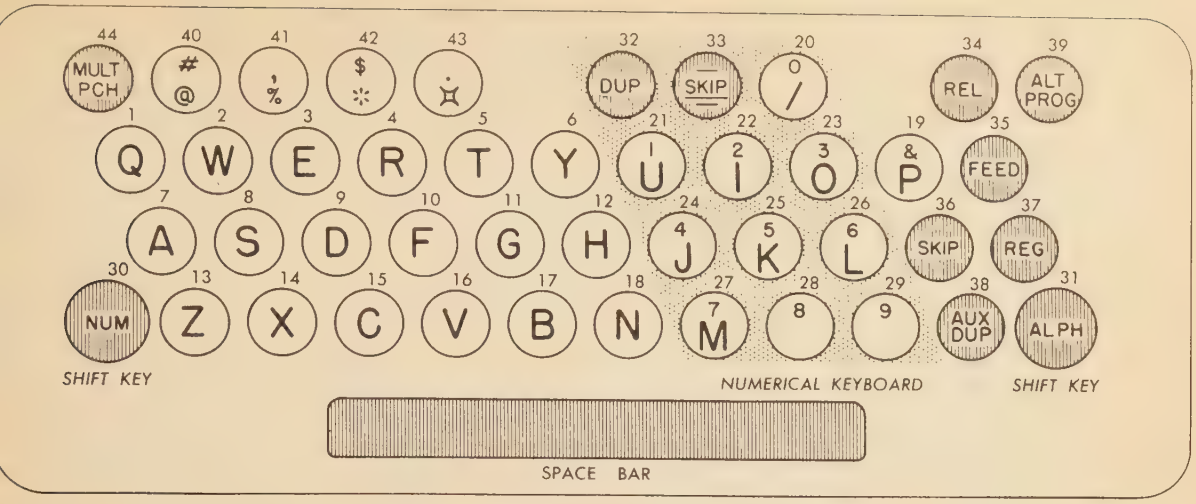


Figure 4.2 Alphanumeric keyboard of the 026 card punch

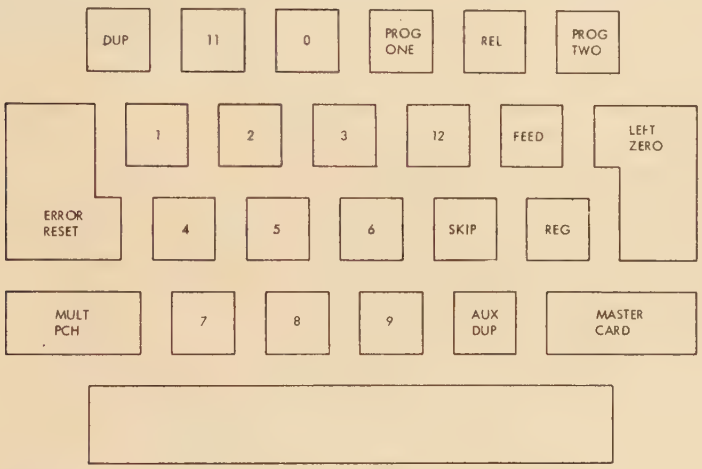


Figure 4.3 Numeric keyboard of the 029 card punch

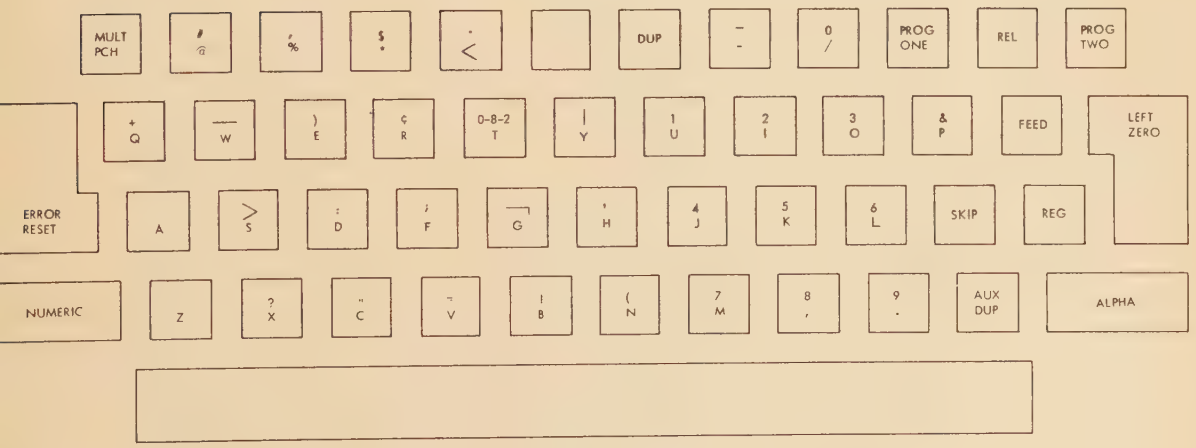


Figure 4.4 Alphanumeric keyboard of the 029 card punch

Figure 4.5 Reprinted by permission from IBM

The combined alphanumeric keyboard of the 029 cardpunch has a maximum capacity of 62 graphic characters while the 024/026 has a maximum capacity of 47 graphic characters. The 15 special characters available in the 029 (but not in the 024/026) are listed in Table 4.1 and are functional (operable) when the 029 is in numeric shift.

Table 4.1

Alphabetic shift	Numeric shift
Q	+
W	—
E	)
R	¢
T	(*)
Y	|
S	>
D	:
F	;
G	¬
H	'
X	?
C	"
V	=
B	!
N	(

(*) Punch 0-8-2 which does not yet have a graphic representation.

In the 024/026, in alphabetic shift, the depression of these keys produces locking of the keyboard. In the 024/026 as well as in the 029, keys A and Z do not yet have a graphic representation when the cardpunch is in numeric shift. Consequently the depression of these keys will lock the keyboard.

The special character , (comma) can be punched either in the alphabetic or numeric shifts in the 029. In both cases the punches will be the same (0-8-3) but the key is different: in numeric shift the comma is above the special character %, and in the alphabetic shift it is below digit 8. In addition, the 0-8-2 punch does not yet have a graphic representation.

In the combined alphanumeric keyboard of the 024/026 the 12-8-4 punch in the alphabetic shift corresponds to the graphic character ¤ (lozenge) while the same punch in the 029 in the alphabetic shift corresponds to the graphic character <.

In most cases, the majority of the characters are those of daily usage. Some of the special characters listed in Table 4.1 are only used in preparing programs for computers where they are essential. The operator should keep in mind that only the 029 is prepared for computer programs. The 024/026 does not have the appropriate characters. The 024/026, therefore, can not be used to prepare programs, and any attempt to prepare these programs in the 024/026 will inflict serious damage on the card punch.

FUNCTIONAL KEYS

MULT PCH (*MULT*iple *P*un*CH*)
This key permits punching more than one digit per column. On depressing this key, regardless of shift, the keyboard will automatically shift to numeric. This key is extremely useful when program cards are to be prepared. There are operators who, in the preparation of these cards, will simultaneously keep depressed this functional key and NUM (*NUM*eric). As mentioned above, this is an unnecessary and tiring practice. In the 024/026 keeping the space bar depressed has the same effect as MULT PCH.

DUP (*DUP*licate)
This key permits the duplication of information from card to card. When the cardpunch is under program control, pressing this key just once is sufficient to start duplication. Automatic duplication, under program control, is carried out at the rate of 18 or 20 columns per second depending on whether the cardpunch has (or does not have) printing mechanism. Duplication under program control, continues until the end of the field on the program card.

If the cardpunch is not under program control, duplication will be continuous as long as the operator keeps the DUP key depressed. This manual duplication is carried out at the rate of 9 to 10 columns per second. Duplication stops when the operator releases the DUP key: this allows a closer control of the number of columns to be duplicated.

If, in the process of duplicating a numeric field, the cardpunch finds a blank column, the keyboard locks (this locking will also occur when the machine encounters a forbidden character). The functional key ALPH (ALPHA) unlocks the keyboard skipping and duplicating, at the same time, that blank column. In the 029, the functional key ERROR RESET also unlocks the keyboard. To unlock the keyboard in the 024/026 the backspace key is used.

SKIP (–)
In the 024/026 there are two keys labelled SKIP. The one in the upper row is dual. In numeric shift it punches the 11-X punch and skips the rest of the field. In alphabetic shift it punches an 11-X punch but does not skip the rest of the field. The second SKIP key in the 024/026 keyboard and the one in the 029 keyboard permits automatic skipping of one field without any punches made on the card. If the machine is under program control, skipping takes place at the rate of 80 columns per second and for all the columns of a programmed field. If the card punch is *not* under program control, pressing this key will skip *one* column at a time. This key is used mainly in alphabetic fields to skip that part of the field which is not going to be used, to the right of the last alphabetic character punched (left justification).

*REL (REL*ease)
Depression of this key causes the movement of the cards from the reading and punching stations to the next station. A card being punched will move to the reading station on pressing this key and the card at the reading station will move to the stacker. The card in the pre-registered position becomes registered at the punching station. If, at the same time, the functional switch AUTO FEED is ON, a new card will be fed down from the hopper to the card bed where it will remain-pre-registered.

If the cardpunch is under program control, depressing REL in an automatic duplicating field will cause the automatic duplication of information from the card at the reading station to the next card at the punching station. Therefore, information which has been correctly keyed is not lost by error.

In the 024/026 to remove all cards in transit, the functional control switch AUTO FEED should be OFF and REL should be keyed as many times as it is necessary. In the 029 this can also be effected with the functional control switch CLEAR, but in this particular case the cards in transit will not duplicate information automatically and the information to be duplicated is lost. This does not happen when REL is keyed. The application and consequences of these two alternatives should be weighed by the operator before keying.

PROG ONE, PROG TWO, ALT PROG
The program cards can carry two coded programs at two different levels: a normal and an alternate level, or, as they are called, program one and program two (or alternate) program. With the cardpunch under program control, the operator can switch the program level as many times as it is necessary. In the 029 there are two keys, PROG ONE and PROG TWO, which function in combination with a functional control switch ONE/TWO PROG SEL. It is this switch that controls, at the beginning of a job, which program level will be "read" by the cardpunch. This level, however, can be changed in the middle of a job by depressing the PROG ONE or PROG TWO keys.

Suppose the operator starts a job under program control with the ONE/TWO PROG SEL switch in position ONE, that is, under control of the normal program, program 1. At any stage of the punching job the operator can depress the PROG TWO key and the remainder of the job will be punched as if the switch were in the TWO position, that is at the alternate level. The physical position of the switch is not altered. It will still be

pointing to ONE. It has not moved. At the end of that card being punched, the program level in force will once again automatically be the normal or ONE (the alternate reasoning is also correct, that is with the cardpunch under program control, with the functional switch in position TWO and depressing the ONE key). This switching of program levels can be effected the number of times the operator deems necessary while punching a card. If the functional key AUTO FEED is OFF, keying PROG TWO will produce a new feed cycle. The new card will be under the command of the alternate program (TWO) regardless of the setting of the program selector switch.

If, on the other hand, a program card has only one program punched, say at the normal level (ONE) keying TWO will suspend the control of that program for the remainder of that card. The normal program will become operative, again, with the next card. This action has the advantage that if the operator wants to suspend the control of the program while punching a card, this can be easily effected by depressing the key that corresponds to the program level not punched in the program card.

In the 024/026 (which does not have such a switch to change program levels) the ALT PROG key has the same effect. In all cases the ALT PROG key must be depressed for every alternate card to be punched in the 024/026.

ERROR RESET

This key is found at the left in the 029 keyboard but not in the 024/026. It is used when there is a keyboard lock. In the 024/026 one of the most usual keyboard unlocking techniques is by keying ALPH. On the other hand, ERROR RESET is extremely useful when keying LEFT-ZERO (L-Z) fields in the 029. These L-Z fields place zeros before the first significant digit in a numeric field under program control. In L-Z fields the operator keys the digits, but punching does not actually take place until the operator depresses LEFT-ZERO key. However, if during the keying of the L-Z fields the operator notices keying errors, such errors can be removed from storage by depressing ERROR RESET and rekeying again. This application will be discussed in depth in the programming chapter.

FEED

This key is used to feed cards manually from the hopper. At the beginning of a job this key has to be depressed twice (or must be kept depressed until two cards are released by the hopper). In a regular job, automatic card feeding is under control of the functional switch AUTO FEED.

LEFT-ZERO (L-Z)

This key is present in the 029 keyboard but not in the 024/026. In fields programmed for LEFT-ZERO insertion, its use is extremely advantageous. Suppose that in a numeric field of 8 columns the operator has to punch a number of digits less than the 8 columns which have been assigned to this field. The procedure to follow in these fields under program control is to key the digits followed by a depression of the LEFT-ZERO key. In these cases, while the operator is keying the digits, no punching takes place but when the operator depresses this key, the digits kept in temporary storage are discharged, the field is punched with the keyed digits plus whatever number of zeros are necessary to fill the field. These zeros will be punched to the left of the first significant digit.

REG (*REG*ister)
This key locates a card for proper reading or punching at the respective stations. Every time a card is fed manually, this key must be operated.

NUM (*NUMERIC*, Numeric Shift)
Numeric shift will be operative when this key is operated or kept depressed. Normally the cardpunch is operated in alphabetic shift; therefore to punch digits in an alphabetic field this key must be depressed. In alphabetic fields under program control the NUM key is operated to include numbers in an otherwise alphabetic description; for instance, on keying complete addresses which include street number and zip codes in an otherwise completely alphabetic field.

AUX DUP
This key is operative in special cardpunches.

ALPH (*ALPHA*, Alphabetic Shift)
This key is used to include letters in fields programmed for numeric shift. For instance, in a car parts job, keying a number like 189 65223 G the operator will depress the ALPHA key to punch the letter G. This key is also useful when duplicating numeric fields to skip blank columns. In the 024/026 this key is also used to unlock keyboards.

QUESTIONS ABOUT CHAPTER 4

1 What is the difference between the 024 and the 026 cardpunches?

2 What is the difference between the 026 and the 029 cardpunches?

3 How many keyboard types are there and how do they differ?

4 What is the difference between functional and punching keys?

5 What does the MULT PCH key do?

6 In a short sentence describe how to mount the program card on the program drum.

7 What does the DUP key do under program control?

8 In a 024/026 cardpunch, without MULT PCH, how can multiple punches per column be obtained?

9 Explain the differences between manual and automatic duplication.

10 What does the SKIP key do?

11 What keys must be operated to unlock the keyboard?

12 What does the REL key do?

13 How is the program control lever used?

14 What does the ERROR RESET key do?

15 What happens in the 024/026 when the SKIP key is operated in the alphabetic and in the numeric shifts?

16 What does the FEED key do?

17 When you operate the program control lever, what mechanism is moved?

18 What does the REG key do?

5 Functional control switches

These small switches are located above the keyboard and some of them have functions directly linked to the automatic commands that the cardpunch receives from the program card. The next chapter will carry a complete study of programs and codes. However, for illustrative purposes and in order to understand more fully the functions of these control switches, we will now discuss briefly the program card.

In the first chapter we mentioned one of the more frequent jobs carried out with a cardpunch: invoices. In the same chapter (Figure 1.2) we have the copy of an invoice and the first card (the master card) prepared from this invoice. When all the cards derived from this invoice are punched, the accounting job that the computer will have to do is greatly simplified.

We also mentioned in that chapter that the 80 columns in the card had been grouped in fields, each with a different number of columns. In each of these fields the same (and in no others) similar information is to be punched. We also mentioned that the information of columns 18 through 56 is common to all the cards in this job and would have to be duplicated from the master card to the following ones. When the cardpunch is under program control, all this duplicating task is carried out automatically, without the operator's intervention, with no errors and at a much faster speed than that of any operator.

From the master card in Figure 1.2, it can be observed that there are 13 fields which have a different number of columns, depending on the type of information to be entered in each field. There are 9 fields which carry information to be duplicated from card to card. These fields are between columns 1-17; 18-29; 30-34; 35-39; 40-42; 43-44; 45-47; 48-51 and 52-56 which have the following number of columns per field, respectively: 17; 12; 5; 5; 3; 2; 3; 4 and 5. Field 1-17 remains blank for other accounting purposes and consequently these 17 columns will be skipped. The program card is coded so that the cardpunch, on each new card, will skip the first 17 columns leaving them blank. The program card is coded for this field with an 11-punch at column 1; the remaining 16 columns (2 through 17) carry field definition punches &, 12-punches or ampersand. In all program cards, the first column of each field is the high priority punch or code, the remaining columns simply limit the length of the field with field definition punches. The second field of the master card in Figure 1.2, columns 18 through 29, with alphabetic information, will carry the customer's name. This will consequently be an alphabetic field. For this field the program card will be prepared to enter alphabetic information which will be duplicated from card to card. Therefore the first

column of the program card for this field (#18), will be punched with a slash (/) which punches a 0 and a 1 simultaneously in the same column. This multiple punch in the program card indicates that the following field is alphabetic and its information will have to be duplicated from card to card. Columns 19 through 29 in the program card will be punched with A's. This is a multiple punch in 12 and 1 and for alphabetic fields, these are the field definition punches.

All the following fields carry numeric information which is duplicated from card to card. Consequently, for those fields, 30-34; 35-39; 40-42; 43-44; 45-47; 48-51 and 52-56 the respective first columns of each field (30, 35, 40, 43, 45, 48 and 52) will carry a 0-punch. Field definition punches (&s, 12-punches) will follow, as many as are necessary to complete each field. Therefore, there will be 12-punches in columns 31, 32, 33, 34, 36, 37, 38, 39, 41, 42, 44, 46, 47, 49, 50, 51, 53, 54, 55 and 56.

The master card of this invoice, from columns 57 through 80, has 4 numeric fields. However, this numeric information will vary from card to card and it is not to be duplicated since each card corresponds to one commodity. (Consequently quantity, commodity number, and item and invoice amount will change.) We therefore have 4 numeric, non-duplicating

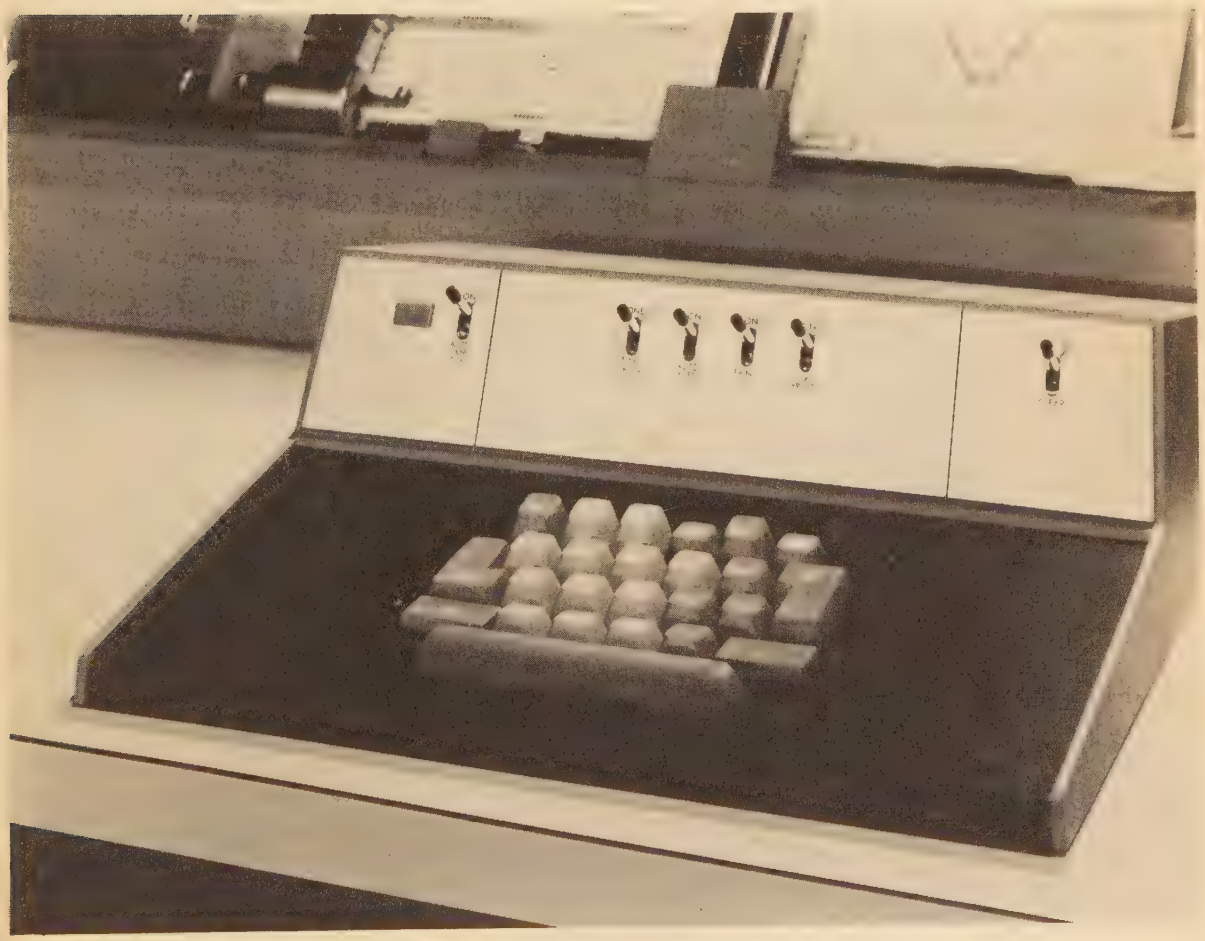


Figure 5.1 Reprinted by permission from IBM

fields and in the program card provisions will be made to this effect. The respective first columns of each of these 4 fields (columns 57, 62, 67 and 74) will be left blank (which is the high-priority code for numeric, non-duplicating fields). The remaining positions in the program card, will be filled with enough &s to complete them: columns 58, 59, 60, 61, 63, 64, 65, 66, 68, 69, 70, 71, 72, 73, 75, 76, 77, 78, 79, and 80 will carry field definition punches.

As previously mentioned, automatic duplication from card to card, is subject to the commands that the cardpunch receives from the program card. However, in some cases, the execution of those commands is conditioned by the functional control switches which are located above the keyboards (see Figure 5.1).

The number of functional control switches which each cardpunch has, varies with the model. But all of them have the same effect regardless of category; for instance the function of the AUTO SKIP/DUP switch is the same in the card punch 024/026 and in the 029.

FUNCTIONAL CONTROL SWITCHES

All these switches (except CLEAR) are of the type seen at home (light switches) with an ON and an OFF position. When any switch is moved from one position to the other, the switch remains in that position. CLEAR, a functional switch found in the 029, is different since, being spring loaded, it restores to its original position after being released.

AUTO SKIP/DUP

When we discussed the preparation of the program card, it was indicated that there were fields that (1) were completely skipped, or (2) carried information that was duplicated from card to card. In the case of our invoice, these fields were exclusively alphabetic or exclusively numeric. These functions, the automatic skip and the automatic duplication, will be carried out only if the AUTO SKIP/DUP functional switch is operational, that is, if it is ON. If it is OFF, the program card command is inoperative. What this means is that in this case as well as in all cases where the execution of a command from a program card is subordinated to a functional switch, the switch has a higher priority than the program card. Therefore, for the automatic skipping and duplication, the program card should be properly coded and this switch should be ON. In any other case, manual skipping or manual duplication can be carried out by depressing the respective keys on the keyboard.

ONE TWO PROG SEL

The program card can carry two programs which may be used in an alternate fashion. In the invoice which we are using as an example the program card suggested was prepared using zone-punches 12, 11 and 0 and with digit-1 punches. Nevertheless the same invoice can be the source of other informations and the operator can punch two or more cards per commodity. If this is the case, the operator can prepare a double program card and, during punching, select the program level more adequate to her needs. Such selection is easily made simply by changing the position of this switch from

ONE to TWO or vice-versa. It is for this reason that this switch is labelled as ONE TWO PROG(ram)SEL(ector).

AUTO FEED

The function of this switch, which is not linked to any command of the program card, is the automatic card feed from the hopper to the card bed and the punching station. When the last column of a card being punched passes through the punching station, and if this functional switch is ON, a new card will automatically be fed from the hopper. The punched card passes to the reading station and the one previously here goes to the stacker *via* the card bed at left. This sequence is automatically repeated without any intervention on the part of the operator when the AUTO FEED switch is ON, every time that a punched card is released from the punching station. When this switch is OFF, card feeding can be effected manually by simply operating one of the following *keys*: FEED, PROG ONE or PROG TWO (in the 024/026 depressing ALT PROG).

In cardpunches equipped with printing mechanism, two switches, labelled PRINT and L-Z PRINT allow or suppress printing of all the allowed characters. When the PRINT switch is ON *all* the characters are printed. When it is OFF, *all* printing is suppressed. The function of the other switch, L-Z PRINT, requires a discussion for which the master card in Figure 1.2 must again be referred to. In this card there are three fields with more columns than necessary. Those fields are: salesman # (columns 45-47); quantity (columns 57-61) and item amount (columns 67-73). As shown in Figure 1.2, these fields carry zeros to the left of the first significant digit (067; 00040; 0003320). These zeros are known as Left-Zeros or L-Z. If the operator wishes to print these zeros on the card, the L-Z PRINT switch must be ON. If this switch is OFF, the printed numbers on top of the card might have been 67, 40 and 3320. The operator should always keep in mind that the PRINT switch ranks higher than L-Z PRINT and if PRINT is OFF, it overrides L-Z PRINT although it may have been ON.

CLEAR

This switch is the only spring-loaded one; therefore it will return to the starting position whenever it is released by the operator. The function of this switch is the automatic removal of all the cards on the card bed. No new feed-cycle will occur on operating this switch and no cards will be fed down from the hopper. Just one depression is necessary to start all the clearing cycles. When this switch is activated, it overrides AUTO FEED, even if this one is ON, because no feed cycle occurs.

This switch is particularly handy at the close of day. The REL key also removes all the cards in transit, with duplication, but the REL key does not override the AUTO FEED switch and there will always be a feed cycle whenever REL is depressed.

QUESTIONS ABOUT CHAPTER 5

1 What is the function of the functional switch AUTO FEED?

2 In the program card, which punches start the automatic skip? Which punches continue the skip?

3 What is the function of the functional switch AUTO SKIP/DUP?

4 In the program card, which punch starts the automatic duplication of a numeric field? Which other punches continue this duplication?

5 What is the function of the PRINT switch?

6 In a program card, which punch starts the automatic duplication of an alphabetic field? Which other punches continue this duplication?

7 What is the function of the functional switch L-Z PRINT?

8 What is the first card of any job called?

9 What is a field?

10 How many types of fields are there?

11 What happens when SKIP is depressed in an alphabetic field? And in a numeric field?

6 Program codes

In previous chapters we have mentioned that in any job there is information common to all cards, repetitive data. Such information is grouped in fields which are always located, for that job, in the same columns. It is therefore very advantageous to duplicate this information automatically. Duplication is then faster (since automatic duplication is much faster than manual keying) and more accurate (because automatic duplication "does not commit errors"). We also mentioned in the previous chapter that all these duplications are carried out automatically only when the cardpunch is under program control and the program card is punched with the right codes.

There are six basic codes. For each field in the program code, the first column has the highest priority. It is the column that indicates to the cardpunch its next function in this field and will be effective for the entire field. This first punch is labelled "the high priority punch". The length of the field is defined by "field definition punches". For automatic skipping and numeric fields, the field definition punches are &'s (12-punches). For alphabetic fields, field definition punches are A's punches. This is for programs at the normal level or level 1. For the alternate level, these field definition punches are 4's and 7/4's respectively.

The six basic codes are characterized by the different functions the cardpunch will automatically perform under their command. The high priority punches (first column of each field of the program card) will allow:

1 numeric shift punching;
2 alphabetic shift punching;
3 automatic skipping;
4 duplication of numeric data;
5 duplication of alphabetic data;
6 insertion of left-zeros.

A complete table of codes for programs at the normal level is shown below in Table 6.1.

High priority codes that require more than one punch per column (codes 5 and 6) are punched by keeping depressed MULT PCH while keying the proper digits. After punching the second digit, release MULT PCH to continue keying.

Based on Table 6.1 we are returning to study further the invoice in Figure 1.2. As indicated in previous chapters, the first nine fields are repetitive data to be duplicated from card to card. These fields cover columns 1-17; 18-29;

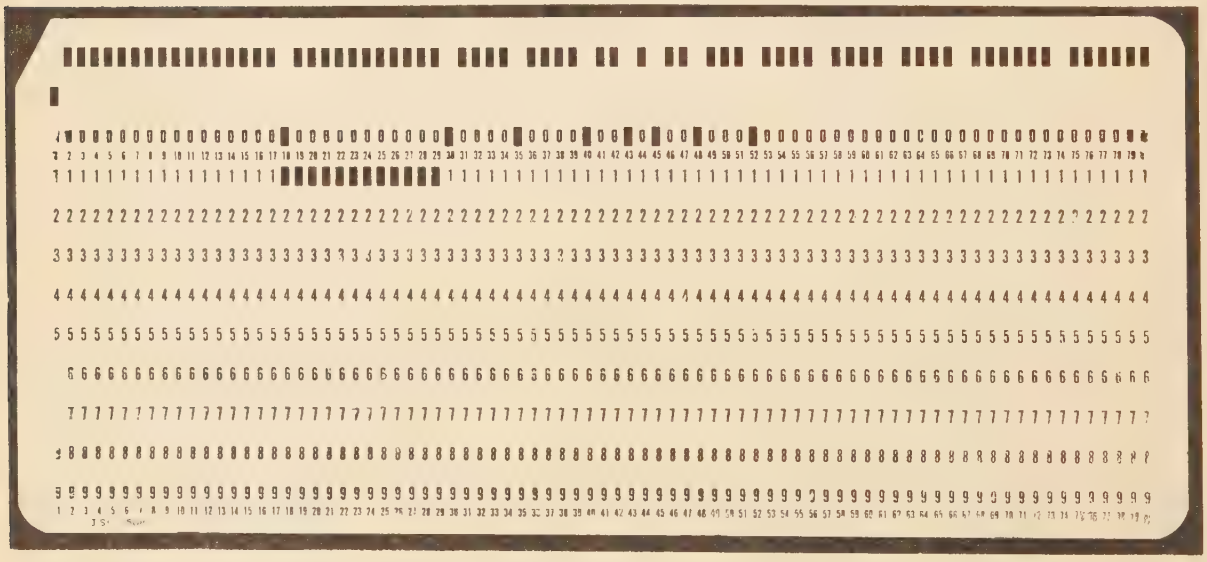
Table 6.1 Codes for normal programs

Codes		Function
High priority punch	Rest of the field	
1) Space (blank column)	&	numeric punching
2) 1	A	alphabetic punching
3) 11	&	automatic skipping
4) 0	&	numeric duplication
5) 0-1(/)	A	alphabetic duplication
6) 2*	&	L-Z, 8 column field
3*	&	L-Z, 7 column field
2-3	&	L-Z, 6 column field
1-2	&	L-Z, 5 column field
1-3	&	L-Z, 4 column field
1-2-3	&	L-Z, 3 column field

*These L-Z codes can only be used in the 029. See Table 6.3 footnote for application of these punches to the 024/026.

30-34; 35-39; 40-42; 43-44; 45-47; 48-51 and 52-56. The four numeric fields that follow (57-61, 62-66, 67-73 and 74-80) change with each commodity of the invoice and will not therefore be duplicated.

Of the first nine fields, one is automatic skip (columns 1-17), the following is alphabetic duplication (18-29) and the remaining ones, through column 56, are numeric duplication. The program card for an invoice of this type is shown below:



One-column fields are programmed with the appropriate high-priority punch. For instance for a numeric duplicated field of one-column length the program card is punched with an 0. The same reasoning applies to all other functions listed in Table 6.1.

The decimal point is never punched on cards although it may be present in the number to be keyed. If, for instance, 33.20 has to be punched the operator keys 3320 and in such fashion that the last digit falls on the last column assigned to that field. If the field has 4 columns, the operator keys 3320. But if the field is longer, the previous positions (columns) to the left of the first significant digit are filled with zeros or the L-Z function is used. In our invoice, the "Item Amount" field has 7 columns and the number keyed is 0003320 for the first item in the invoice. Under the "Item Amount" heading for the following commodities, the numbers to be keyed would be 0006300, 0000810, . . . 0000078, respectively.

A good deal of time and effort might have been saved in this job if the program card for the invoice had been prepared using the L-Z functions. For this 7-position field, at the normal level, punching a 3 in the program card would save the operator some keying effort; with a program card so prepared, for the first commodity of the invoice, the operator would only have to key 3320 and depress the LEFT ZERO key. The operator will notice that in these L-Z fields, punching is inoperative until the LEFT ZERO key is depressed. The card will not move and the keyed digits will not be punched while the numbers are kept in temporary storage. Upon depressing the LEFT ZERO key, the machine will discharge the numbers in storage, which will then be punched preceded by all the zeros at the left which are necessary to fill the field.

The insertion of zeros to the left of the first significant figure (with and without the L-Z function) is used to punch dates. In these cases, month, day and year are punched in that sequence. And for the month, its calendar number and not its name is applied. Since generally this numeric data is duplicated from card to card in invoices and other jobs, the program card usually carries a high priority punch and field definition punches for a 6-position field. For instance: January 11, 1923 is keyed as 011123; March 3, 1924: 030324; August 23, 1949: 082349; February 8, 1952: 020852 and November 25, 1958: 112558. In our invoice only month and day are used for which a 4-position field is needed. The invoice is dated December 31 and therefore the number to be duplicated will be 1231.

ALTERNATE PROGRAMS

A job which requires cards in a different sequence or two cards per commodity (in an invoice), is prepared in a different fashion. It simply entails the preparation of a double program card, that is, one card which is coded at two different programming levels.

We are returning to the invoice in Figure 1.2 as an example. We have already described how to prepare the program for the normal level. But in another accounting job, a group of cards may be necessary which, while punched with information taken from the same invoice, is keyed in a different fashion. Let us assume that we will need a second card carrying the complete customer address, with street number, city and state in columns 57 through 80. Here we can use a double program. The first program has already been described and will carry duplicated information through column 56. We will use the same information in a second card and consequently the program card will be coded, at the second level, in such a fashion as the first through column 56. The codes will, however, be different for the

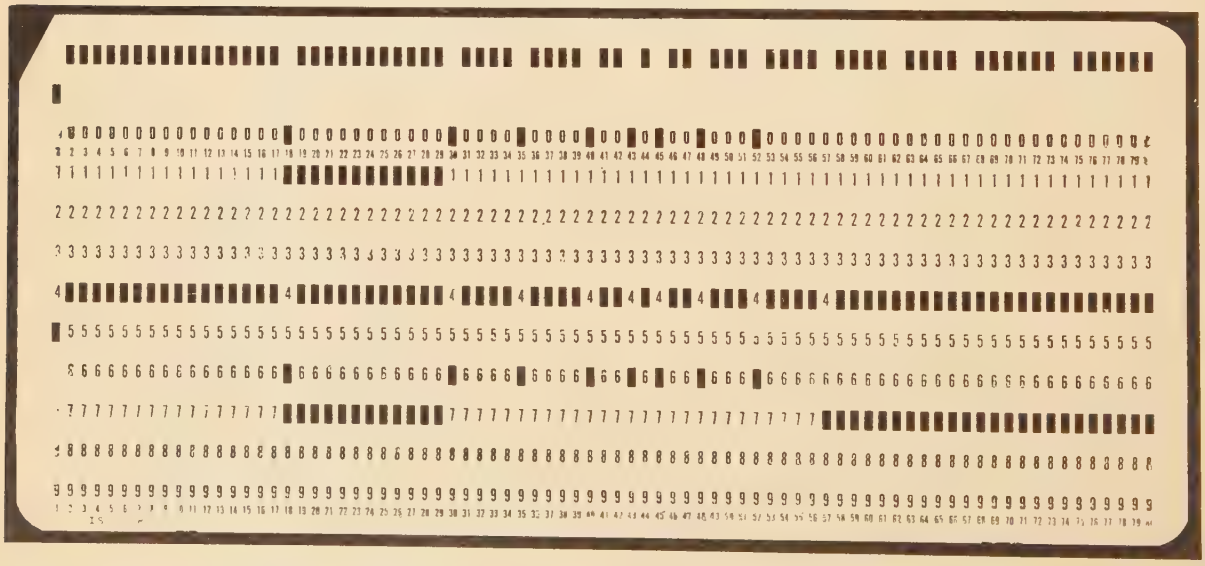
next field, columns 57 through 80. This will be an alphabetic duplicated field. It will be recalled that, for a normal level, an alphabetic duplicated field required an 0-1 punch in the high priority column followed by field definition punches, which in that case, were A-punches. It is very easy to formulate the codes for the alternate level program card. The operator will imagine that there is a horizontal line between the digit-3 and digit-4 rows. The field definition punches (&, 12-punches) at the normal level, will be 4-punches at the alternate. All the other codes are punched in a similar fashion, as shown in the next table, Table 6.2.

Table 6.2 Codes for alternate programs

	Codes		Function
	High priority punch	Rest of the field	
1)	Space (blank column)	4	numeric punching
2)	7	7-4	alphabetic punching
3)	5	4	automatic skipping
4)	6	4	numeric duplication
5)	6-7	7-4	alphabetic duplication
6)	8	4	L-Z, 8 column field
	9	4	L-Z, 7 column field
	8-9	4	L-Z, 6 column field
	7-8	4	L-Z, 5 column field
	7-9	4	L-Z, 4 column field
	7-8-9	4	L-Z, 3 column field

Note: See note following Table 6.3, for codes which require multiple punches.

With this table, double programs can be prepared in one card. Such double programs, prepared for instance for the invoice in Figure 1.2, with the addition of the customer's complete address, is shown in the next figure.



There are two very simple ways to prepare a double program in a single card. The first is to punch the normal program in a card. Upon release, this card is placed manually at the punching station and the alternate-level program is then punched. The other alternative is to punch the two programs, the normal and the alternate, in two different cards. Then these two cards are placed manually, one (either) at the punching station and the other at the reading station. Depressing DUP, information at the reading station is duplicated in the card at the punching station.

The cardpunch will automatically shift from the normal to the alternate program when the PROG TWO key is depressed in the 029. This shift can be performed even when the functional switch ONE TWO PROG SEL setting is pointing ONE. The change to the alternate program takes place immediately. Since there is no such functional switch in the 024/026 use the ALT PROG key to secure the same effect.

Switching the setting of the functional switch, say from ONE to TWO in the middle of the card, does not change the level at which the card is being punched. Only when the next card is registered, *and not before*, does this change to the alternate level become operative. The moment the next card is registered, it will be punched under control of the alternate level. The normal program will become operational only when the switch is set back to ONE. The inverse reasoning is also correct; that is to say, if the operator is keying under control of the alternate program and switches to ONE in the middle of a card, this change will become effective when the next card is registered.

As we mentioned, this change in program level can also be effected by keying PROG ONE, PROG TWO or ALT PROG. Depressing any of these keys, however, changes *immediately* the control of the program level. For instance, if an operator is keying with the ONE TWO PROG SEL functional switch set to ONE, keying PROG TWO will be sufficient to put the keying of the remaining of the card under control of the alternate program or until the operator reverses to the original program by keying ONE. Unless the functional switch setting is changed, the cardpunch will return to its level when a new card is registered, regardless of the number of changes (and their respective directions) performed in the middle of the card. Using this procedure, program levels can be switched at will.

The operator should also keep in mind that duplicating fields under a double program requires that the identical fields to be duplicated be punched in the same columns in the two programs and in the two cards. For instance, if the information "Customer Number" is keyed between positions 30 and 34 (5 columns) this field will necessarily have to be located between the same columns for the duration of this job. A complete table with all the codes is shown below.

ADDITIONAL PROGRAMMING FEATURES

The field definition punch, a 12-punch (&) or a 4-punch, for the normal or the alternate program, is very helpful to combine several consecutive fields. Assuming there are several consecutive fields which have to be automatically duplicated or automatically skipped, these fields can be programmed as a single field. The operator should not program 1-column fields with field definition punches.

Table 6.3 Normal and alternate program codes

Function	Codes			
	Normal level		Alternate level	
	High priority punch	Rest of the field	High priority punch	Rest of the field
Numeric punching	Space	&	Space	4
Alphabetic duplication	1	A	7	7-4
Automatic skipping	11	&	5	4
Numeric duplication	0	&	6	4
Alphabetic duplication	0-1(/)	A	6-7	7-4
L-Z, 8 column field	2*	&	8	4
L-Z, 7 column field	3*	&	9	4
L-Z, 6 column field	2-3	&	8-9	4
L-Z, 5 column field	1-2	&	7-8	4
L-Z, 4 column field	1-3	&	7-9	4
L-Z, 3 column field	1-2-3	&	7-8-9	4

*In the 024/026 the 2- and 3-high priority punch have an altogether different function than in the 029. In the previous chapter we mentioned the effect of the functional switches PRINT and L-Z PRINT. In the 026, printing of the left zeros can be controlled by program only, since there are no functional switches of this type. The 3-punch suppresses all print in the 026; the 2-punch suppresses printing of the left zeros. The interrelationship of these two codes is the same one we mentioned in the previous chapter, when the functional switches PRINT and L-Z PRINT were discussed.

The automatic skip, which can be programmed with high-priority 11- or 5-punches, is dependent upon the setting of the AUTO SKIP/DUP functional switch. If this switch is OFF, the program codes are not recognized and although the program card may be coded for automatic duplication or automatic skipping, the cardpunch does not carry these functions out.

When there is a program card on the drum and the sensing mechanism is engaged (down), the cardpunch is in numeric shift. In order to have the machine in alphabetic shift, under program control, the card must be coded accordingly. While the high-priority punch for an alphabetic field is a 1-punch (or a 7-punch) the field definition punch & used for numeric fields is not sufficient to extend the applicability of the alphabetic shift. Consequently, the field definition punches for alphabetic fields require two punches, the usual 12-zone punch (&) and the 1-punch (digits 7 and 4 at the alternate level). The combined 12-punch and 1-punch is the multiple punch required for letter A. Therefore, at the normal level, and only at this level, field definition punches for alphabetic fields are A-punches.

LEFT-ZERO (L-Z) FIELDS

Model B of the 029 cardpunch has been specially designed to insert zeros before the first significant digit in numeric fields of more than 2 columns.

In a L-Z field the depression of the digit keys is *not* followed by the immediate punching of the keyed digit. These digits go into temporary storage. The actual punching takes place when the LEFT ZERO key is depressed. At this time, the machine will punch all the keyed digits and all the zeros (to the left of the first significant digit) necessary to fill the field. For an 8-column L-Z field, there is a half second time lapse between the time the operator depresses the key and the actual punching. In fields of less than 8 positions, the time lapse is even shorter.

In a programmed L-Z field, if there are no digits keyed in storage, depressing the LEFT ZERO key will punch the complete field with zeros.

The L-Z fields have other important characteristics. If the operator wishes to skip a L-Z field, the SKIP key should be depressed. In order to avoid errors, however, for L-Z fields to be skipped, no digits should be in storage. Any storaged digit should be "erased" by keying ERROR RESET before skipping it. This can also be applied for the manual duplication of L-Z fields. A card at the reading station in a L-Z field can have its information duplicated into the registered card at the punching station by keying DUP. But the operator should also be aware that for this type of L-Z manual duplication, *all* the columns of the L-Z field must be duplicated. The manual duplication of part of a L-Z field and the manual keying of the rest of it, is specifically forbidden.

The maximum length of L-Z fields is limited to 8 columns. Keying more digits than the capacity of the programmed L-Z field, may produce cards punched with errors. If such an excess if recognized by the operator *before* keying LEFT ZERO, all the possible errors can be avoided by simply depressing ERROR RESET. This will automatically "erase" all the keyed digits from storage and the operator can immediately re-key the right digits or the right number of digits within the capacity of the L-Z field. There are no automatic provisions in the 029 to avoid this error which, in other cases, can lock the keyboard. But in these L-Z fields there is no such locking and the operator should avoid them.

There are, within the possibilities of the 029, a number of techniques which will permit the extension of the capacity of L-Z fields to more than 8 columns. One example will be very illustrative. Let us assume there is a numeric field of 10 columns between columns 25 and 34 which is to be programmed as a L-Z field. One alternative is to duplicate the first 2 columns of this field, columns 25 and 26, with zeros. The remaining 8 columns can be prepared as a L-Z field.

The other alternative is to divide the 10 columns into two L-Z fields of 5 columns each; or 4 and 6; or 3 and 7 and code the program card accordingly. This last technique, however, requires that the operator depress the LEFT ZERO key twice, once per each L-Z field. It is therefore more convenient to use the other technique for L-Z fields longer than 8 columns; this, in turn, results in higher production, particularly when there is a substantial number of fields of 8 or less digits.

The LEFT ZERO key must be actuated if the numbers in temporary storage are to be punched. The fact that an operator, by sheer coincidence, may key a L-Z field with the same number of digits for which it is programmed, does not mean that the digits will be discharged and punched automatically. If, for instance, there is a L-Z field of 8 columns and the operator keys 8 digits, these will not be punched unless the operator depresses the LEFT ZERO key.

Care should be exercised in these L-Z fields to avoid pressing the alphabetic keys located to the side of the digits on the keyboard. In view of the sensitivity of the cardpunch to pressure, should the operator accidentally key any of these alphabetic keys, the net effect is the equivalent of keying the LEFT ZERO key and consequently all the digits in storage would be discharged producing an incorrectly punched card which would be impossible to correct with the ERROR RESET key.

The program level should *not* be switched in the middle of a L-Z field unless all the stored digits are released from storage by keying ERROR RESET. If this depression is not effected, the following L-Z fields may be punched with errors. Nevertheless, in any L-Z field, *before* any key is depressed, say, at the first position of that field, the program level may be switched. The keyboard is locked during the short interval between the time the LEFT ZERO is keyed and the actual punches are effected. Any attempt to punch during that interval has no effect. The LEFT ZERO key in the A and the C models of the 029 cardpunch is inoperative. It is operational in the B models only.

Finally, it must be emphasized that an empty drum should never be inserted in the program drum. In addition, the sensing mechanism should never be lowered when the program drum is not in place.

The 024/026 cardpunches are not prepared for L-Z fields. The digit-2 and digit-3 codes in the program card used in the 024/026 have an altogether different function than in the 029.

QUESTIONS ABOUT CHAPTER 6

1 How many program codes are there?

2 What are the functions of the different codes?

3 Which is the high priority punch for the numeric duplication at the normal level?

4 Which is the field definition punch for skipping in alternate programs?

5 In alternate programs, which is the high priority punch for an alphabetic field?

6 Which is the field definition punch for an alphabetic duplicated field at the normal level?

7 In an alternate program, which is the high priority punch for an alphabetic field?

8 At the normal level, which is the high priority punch for an alphabetic field?

9 Which is the field definition punch for the automatic skipping in the normal programs?

10 At the normal level, which are the field definition punches for an alphabetic field?

11 Indicate the codes for a program card punched at the normal level for the following functions:

a) automatic skipping, columns 1-18;
b) numeric duplication, columns 19-30;
c) alphabetic punching, columns 31-50;
d) alphabetic duplication, columns 51-79;
e) numeric duplication, column 80.

7 How to begin a job

In previous chapters, we have described almost all the details the operator should know before starting a job. In this chapter we describe the few remaining details which will enable the operator to start a job.

Most jobs require automatic duplication, generally of numeric fields. Under a command from the program card, information "read" from the card at the reading station, is duplicated in the card at the punching station. Therefore, there must be one card where all this information is punched for the first time. This is the first card of each and every job, the "master card".

In Figure 1.2, the master card derived from the invoice is shown and as we mentioned before, it carries the information pertaining to the first commodity in the bill. It is obvious that the preparation of the master card is quite simple. With this know-how, we are ready to start a new job.

All the steps which we will now mention are linked one to another and are merely the application of things learned in previous chapters. For this reason, we will not go into detail. The information to be punched in the master card, which will be duplicated into the following cards, must necessarily be keyed *manually*. Consequently, in view of the importance of an accurate keying of the master card, automatic duplication must be suspended while the accuracy of the punches in the master card is checked: the functional switch AUTO SKIP/DUP must be OFF.

How to Begin a New Job

1 Place about five hundred blank cards in the hopper.

2 Turn the main switch ON. This switch in the 029, is underneath and to the right of the desk. Keying on the 029 can be started immediately. On the 024/026, the main switch is on the stacker. It takes about 30 seconds to be able to punch on the 024/026 after the main switch is turned ON.

3 Prepare the program card. All the functional switches must be OFF, except PRINT. When a simple program is to be prepared, say with all numeric data, it is very easy to place the card punch in numeric shift by mounting a blank card on the program drum and lowering on it the sensing mechanism. However, if the preparation of the program card involves digits and letters, there is still another approach. The punch card normally is in alphabetic shift. In any case, keying MULT PUNCH or ALPHA puts the machine in either shift. Consequently, and depending upon the type of program to be prepared, these two approaches can be used.

4 Place the program card on the drum on the program unit. Lower the sensing mechanism. Depress REL to engage it.

5 Turn the functional switch AUTO FEED ON while keeping the AUTO SKIP/DUP OFF. Depress FEED twice, so as to register the first card.

6 Prepare the master card.

7 Switch the functional switch AUTO SKIP/DUP ON. The other switches may or may not now be activated according as to what the need may be and as per the command in the program card.

How to Finish a Job

1 When there remains only two cards to be punched, regardless of the number of blank cards remaining in the hopper, switch AUTO FEED OFF. The cards which have to be punched to finish the job will be fed down by keying REG and REL, in that order.

2 The card next to the last has been punched and had been released to the reading station. Key REG. This will register the last card at the punching station.

3 Punch the last card.

4 Depress, in succession, keys REG, REL and REG. (In the 029, use the functional switch CLEAR.) This will release all cards from all stations.

5 Remove the program card from the drum and replace it with a blank card.

When a job must be momentarily stopped and/or the main switch must be turned OFF for whatever reason, before completing a job (say at lunch or on a break), the card at the punching station must be finished and released. The AUTO FEED functional switch should however be turned OFF before releasing this card. This will facilitate the return to the punching job. In cases where a group of cards in a job is to be followed by another group which uses the same master card, it is advisable to finish the first group by placing a blank card as the last one in that group. This blank card, when passing through the punching station will receive the information to be duplicated. If, at this point, the job is temporarily stopped, on returning to the second group of cards, the task of duplicating information becomes very easy.

GENERAL INFORMATION

Manual Card Insertion

A card which has been torn, punched with errors, or returned by the verifier must be re-keyed. There are other cases where pre-codified or special cards are supplied with the documents. In all these cases, the blank or special cards must be handled individually. This is carried out by placing them at either the punching or reading station, not at the hopper. The operator should not push these cards all the way in but to about one inch beyond the pressure roll. Keep in mind that if this manual insertion is not carried out propertly, duplicating errors may occur.

Manual Card Removal
Manually move the card from the punching station into the reading station and press the REL key. If for any reason a card must be removed manually from the punching or reading station, hold down the pressure roll release lever while pulling out the card. If the card is not yet released depress all the numeric keys; then while holding down the lever, pull the card out.

When a card has been released from the reading station but has not reached the stacker, move it to the extreme left by pressing down on the arm that extends from the reading station. Machines which have been used for a long time exert light pressure to the point where it is not necessary to use the lever to remove the card.

Keyboard Locking
The keyboard locks in any of the following circumstances:

1 In a combined alpha-numeric keyboard, in numeric shift, when the A and Z keys are depressed. To unlock the keyboard press the ERROR RESET or ALPHA keys.

2 The keyboard will lock when, in a numeric duplicating field, there is a blank column. Sometimes, this technique is used to check the cardpunch working condition. To unlock the keyboard, press ERROR RESET, ALPHA, or the back space key; then advance one column with a space bar.

3 A card has been manually placed at the punching station but has not been registered yet. If the operator actuates on the keyboard or the space bar in these circumstances, the keyboard will lock. To release, key REG, FEED, or REL.

4 A card is already registered at the punching station. If, in these circumstances, the operator depresses the REG or FEED keys, the keyboard locks. To release, press ERROR RESET or ALPHA.

5 The keyboard will lock in a 024/026 with an alpha-numeric keyboard when an alphabetic key is depressed in numeric shift. To release the keyboard, key REL, ALPHA or the backspace key, and press the space bar to skip the column.

6 The keyboard will lock if the main switch is turned OFF. Release with FEED key.

When a field must be punched with a combination of numeric and alphabetic characters, the operator will have to decide which is the most convenient program card or which characters are more frequent. Assuming there are addresses to be punched which carry many more alphabetic than numeric characters, it is then preferable to prepare the program card to represent an alphabetic field. In that alphabetic field whenever numbers have to be entered, by depressing the NUM (NUMERIC) key, the keyboard will be in numeric shift. Upon releasing the NUM key the keyboard will again be alphabetic and under the program control. If the information to be entered carries more numbers than letters, the program should be coded accordingly.

On the other hand, if the number of positions which has been assigned to that field is greater than the number of characters to be punched, the remaining columns of the field are conveniently skipped by depressing SKIP.

The cardpunch will skip at the rate of eighty columns per second and is much faster than the space bar. In numeric fields, when there are more available columns than digits to be punched, it is advisable to precede them with zeros, so that the last digit falls on the right-most column (right-justification). In alphabetic fields the operator starts keying from the first column at the left (left-justification) and the unused columns are skipped.

Duplication of identical information, common to a number of cards, can be carried out by depressing the DUP key. The duplication rate (number of columns per second) depends on whether or not the cardpunch is under program control. If it is not, keeping DUP depressed will duplicate at the rate of nine to ten columns per second. This allows the operator a close control on the number of columns to duplicate. When the cardpunch is under program control, duplication takes place at the rate of eighteen columns per second (when there is printing) or twenty columns per second (without printing).

Pressing the PROG ONE, PROG TWO or ALT PROG keys when a card is completely punched, will produce a feed cycle when the functional switch FEED is OFF. The next card to be registered at the punching station will be under program control at the level which corresponds to the key depressed. In a punching job, if every alternate card (say the second, fourth, etc.) is under control of the alternate level, this technique is quite advantageous. The procedure is to turn the functional switch AUTO FEED OFF before reaching column 80 of the very first card of that job. This will suspend automatic feeding and allow the operator, when the second card has to be punched, by simply depressing any of the above mentioned keys, to change the program level and simultaneously start a feed cycle. The second card will then be punched under control of that particular program level. The next card will be fed by keying FEED. This third card will now be punched at the control level indicated by the functional switch.

The operator should be aware that when the sensing mechanism is lowered it does not necessarily become engaged. For that effect, the REL key should be depressed.

On beginning a new job, by keeping depressed FEED, two feeding cycles occur. When the second card is fed from the hopper, the first one is automatically registered, while the second one stays underneath, pre-registered.

One should not attempt to duplicate, in the 024/026, cards which were punched in the 029 cardpunch. There are certain characters which can be keyed in the 029 (but not in the 024/026), which can overcome the printing capacity of the 024/026, producing damage. This damage can take place even if the functional switch PRINT is OFF.

The DUP key can be used, when the cardpunch is not under the program control, to skip columns which were not punched in the previous cards. This is a much faster operation than using the space bar.

ERRORS

When the operator commits a keying error and realizes it immediately, it is very easy to duplicate that card and, at the same time, avoid making the same mistake. The moment the operator perceives the error, REL should be keyed. This moves the incorrect card to the reading station while duplicating from the card previously at that station, all those fields programmed for

automatic duplication. Now the incorrect card is at the reading station and a blank is at the punching station. By keying DUP all the information punched correctly can be duplicated in the new card until the operator reaches the column or field in error. At this point the operator continues keying manually in the new card all the information which had been punched in error, and duplicates automatically all those so programmed. The incorrect card is removed from the stacker and discarded.

The difference between keying REL or actuating the functional switch CLEAR, is that by keying REL, duplication takes place. With CLEAR, all the cards are removed from the card beds and stations but there is no duplication. A new master card will therefore have to be prepared to recommence the job.

In the same fashion when there is information to be duplicated in a few following cards, only the first one should be punched manually. The other ones are duplicated with a single depression of DUP. When the card punch is not under program control, duplication occurs at the rate of one column each time the DUP key is actuated or continuously, by keeping it depressed.

On occasion, an apparently serious keying mistake can be made which, however, may be easily corrected. It is the very common transposition of letters or digits. If the keying error is made on punching the master card, the operator may unexpectedly wind up with a large number of cards with an inaccurate field. For example, assuming there are four hundred cards to be punched with a six-position numeric field between columns 40 and 45, and that, in the master card the operator committed a transposition error by keying 654321 when it actually was 564321. When these four hundred cards are verified, the verifier immediately locates the error. This does not necessarily mean that the entire job is lost. There are a number of simple solutions.

The first solution involves preparing two program cards. The first program card is coded in such fashion that the complete card (except those columns punched in error 40-45) are automatically duplicated. To that effect, the program is prepared for simultaneous alphabetic and numeric duplication. The digits zero and one are punched in columns 1 and 46 and the letter A in columns 2-39 and 47-80. The second program card has the 1-39 and 46-80 fields coded for automatic skipping and the 40-45 field for numeric duplication. Place about four hundred cards in the hopper, the first program card on the drum, the program control lever ON and the first inaccurate card at the reading station. Depress FEED twice, thus feeding down two cards. The first one will be registered. The correct information on the first erred card is automatically duplicated in the first blank card; the incorrect field is however automatically skipped. The first inaccurate card and the first correct one arrive in this order to the stacker. Depress the FEED key once and then REG. Place the second erred card at the reading station and proceed as before. In a short time, the operator has four hundred cards punched correctly in the 1-39 and 46-80 fields. The 40-45 field remains blank.

Sort the correct from the incorrect cards and discard the incorrect ones. The four hundred partially punched, correct cards are now placed in the hopper. The second program card is now mounted on the drum which, by turning the program control lever ON becomes effective immediately. The AUTO SKIP/DUP and AUTO FEED functional switches are OFF. Manually place a card at the punching station and register it by depressing the REG

key. Depress FEED whereby a pre-punched card is fed from the hopper and placed pre-registered underneath the blank card. Punch the blank card manually at positions 40-45, being very careful not to commit any errors. This card is now the master card. Turn the functional switches AUTO SKIP/DUP and AUTO FEED ON. Depress SKIP. From that moment on, the 400 pre-punched cards are automatically punched without any intervention from the operator. It is an efficient and accurate operation which takes much less time than punching 400 new cards with all the information.

The second solution is to prepare just one program card similar to the first one of the two mentioned above, with automatic duplication between columns 1-39 and 46-80. The 40-45 field is coded for a numeric field (non-duplicating). Proceed as follows: place the coded program card on the drum and turn the program control lever ON. Place 500 blank cards in the hopper, the AUTO SKIP/DUP switch ON but keep the AUTO FEED switch OFF. Depressing FEED, bring down two blank cards from the hopper, the first of which becomes registered. Place manually at the reading station the first of the incorrect cards. Under program control the correct information between columns 1-39 is duplicated in the new card. The cardpunch stops at position 40 and the operator keys the right digits. The rest of the card is automatically duplicated under program control. The first incorrect card and the first correct one arrive in that order to the stacker. Place a second incorrect card at the reading station. Depress FEED and REG to produce a feed and registration cycle. Proceed as above with the remaining cards. From the stacker sort the correct from the incorrect, discarding the latter.

One of the most frequent keying errors is to punch in L-Z fields a number of digits greater than the capacity of the field. For instance, for a 6 column L-Z field it is impossible to key a number with 7 or more digits. This is an error which should be avoided and, upon recognition, the stored numbers should be cancelled by keying ERROR RESET. Nevertheless if the number of keyed digits is the same as the number of columns (say 6 keyed digits for a 6 column L-Z field) it is still mandatory to key LEFT ZERO. The cardpunch does not automatically punch unless this key is depressed.

Whenever a L-Z field must be left blank, the operator should key SKIP *before* entering any digit in storage. But should this be the case, and there are some digits in storage, these should be discharged with ERROR RESET and only then SKIP should be keyed. Another way to skip L-Z fields is by depressing the L-Z key before entering any digit: the field will be filled with zeros. Complete L-Z fields can be duplicated manually keying DUP.

Another very frequent error takes place with pre- or partially punched cards. In these cases the correction job is somewhat different from that mentioned above, simply because these are pre-punched cards. The pre-perforations can be names, or codes, or numbers in a sequence not to be altered. Whenever the operator notices a keying error with these pre-punched cards, this procedure is advisable. Turn AUTO FEED OFF. This will stop any coded cards from being fed from the hopper. On keying REL the card at the reading station goes to the stacker, followed in sequence first by the card punched in error (which has duplicated information from the previous card) and then by the pre-registered card. Remove this last card and keep it aside. Then remove the incorrect card from the stacker and place it manually at the reading station. Place a blank card at the punching station and depress REG to register both.

Using the DUP key, duplicate the correct information into the blank card.

Punch carefully in the blank card the information keyed in error originally, and again, using the DUP key, duplicate all the correct information still available in the incorrect card. Turn the functional switch AUTO SKIP/DUP ON. Replace at the punching station the very first card which had been set aside, depress REG to register this card and, at the same time, feed a new pre-coded card to the pre-registered position at the punching station. Once again the cards have the same sequence as before and are properly punched. The incorrect card is removed from the stacker and discarded.

CREDIT "X"

In many accounting operations there are occasions when amounts have to be deducted from outstanding balances. In other words, there are credits to be debited against the balance. In order to identify a credit figure in a punched card, it is customary to punch the 11-zone. This punch is known as the X-punch or credit punch and is keyed over the digit in the last column of the field. This will identify the figure as one to be subtracted.

In order to X-punch a numeric field, the operator keeps the MULT PCH key depressed and actuates the SKIP (or dash) key and the digit, before releasing the MULT PCH key. In the 024/026 cardpunch, the SKIP key to be depressed is located in the upper row of the keyboard. In the 029 the dash (—) key in the upper keyboard row should be used.

In fields programmed for L-Z insertion, keying an X-punch eliminates the need of depressing the L-Z key at the end of the L-Z field. On printing models, the minus (-) sign overprints the digit in the units position.

Punching an 11-X in a numeric field causes skipping but skipping does not occur in an alphabetic field. If the operator must use an 11-X punch in a numeric field, and wants to avoid skipping, proceed as indicated above, by using the MULT PCH key. Nevertheless provisions can be made in the program card to avoid this skipping. If an X is to be punched (keyed or duplicated) alone in a column, this punch can be programmed, as a one-column alphabetic field. To this effect, in alpha-numeric machines, that single column is programmed with a 1. In numeric card punches, leave blank the *next* column following the X-punching in the program card.

The keyboard locations and functions of the SKIP keys have already been discussed in chapter 4.

QUESTIONS ABOUT CHAPTER 7

1 How can alphabetic information be entered in a numeric field?

2 How can numeric information be entered in an alphabetic field?

3 Which are the high-priority and field definition punches for automatic skipping in the alternate program?

4 When the sensing mechanism is lowered on a blank card, in what shift is the cardpunch?

5 In what shift is the cardpunch when the sensing mechanism is up?

6 What key should be actuated when alphabetic information is to be entered in a numeric field?

7 After turning the main switch ON, how long do you have to wait before you can start punching in the 029?

8 In a short sentence describe the correction procedure to correct a group of incorrect cards.

8 Billing

Some of the more frequent punching jobs will be described in this chapter as well as in the following ones. Two accounting activities, billing and payroll, are extremely simplified by the use of computers. In the IBM exercise book, which is extensively used by commercial schools, there is a series of billing

REPRESENTATIVE COMPANY

CUSTOMER'S ORDER NO. AND DATE

SIGNED BY

REQUISITION NO.

REFER TO INVOICE NO. 42401

INVOICE DATE 1/8

SALES BR.	SALES SM'N.	CITY	ST.	CUST. NO.
23	98	126	44	87631

SOLD TO

New York Stores, Inc.
1026 Madison Avenue
New York, N. Y.

SHIPPED TO AND DESTINATION

DATE SHIPPED

F. O. B ENDICOTT, N. Y. TERMS 30 DAYS NET

QUANTITY	DESCRIPTION		UNIT PRICE	AMOUNT	COST
70	SWEET POTATOES	23912	2.10	147.00	95.56
27	FLY PAPER	65393	2.20	59.40	38.60
80	MACARONI	12513	1.25	100.00	85.00
28	AMERICAN CHEESE	14008	.51	14.28	9.46
90	PRUNES	23735	2.80	252.00	216.10
200	COFFEE	45263	2.88	576.00	349.75
29	CHOW CHOW	23207	.34	9.86	8.44
1	ZINC BUCKET	65996	9.60	9.60	8.38
16	BROOMS	65135	3.10	49.60	44.90
40	CIDER	19216	1.40	56.00	36.35
21	KETCHUP	34464	1.88	39.48	37.07
12	NOODLES	12552	1.25	15.00	13.50
14	DOG BISCUITS	73335	4.50	63.00	56.95
50	LYE	63504	.64	32.00	28.80
150	CONDENSED MILK	76272	2.60	390.00	380.75
176	COCOA	46257	.80	140.80	133.00
130	PAPRIKA	43632	.60	78.00	74.68
20	CRACKERS	48312	3.10	62.00	55.70
30	TAPIOCA	50927	3.65	109.50	102.60
15	BEANS	58080	1.65	24.75	21.25
80	PEPPER	43672	.55	44.00	39.65
97	NUTMEG	43560	.92	89.24	81.19
140	SALT BUTTER	14785	7.50	1,050.00	68.35
50	SWISS CHEESE	14920	.33	16.50	10.72
70	PEAS	58664	2.15	150.50	128.00
1636		41314	58.30	3,578.51	2,124.75

and payroll exercises. Copies of pages 1 and 2 of that book are shown in this and in the next chapter by courtesy of the IBM Corporation. We will study billing and payroll on the basis of those figures. The student will derive helpful knowledge in tackling similar applications.

The study plan which we will follow for billing and payroll are as follows:

1 preparation of the program card
2 preparation of the master card
3 machine set-up.

Thus the student will be prepared to continue from this point on as if it were a real job at the office. We will also introduce the "Instruction Sheet" which is generally attached to the job order. Copies of a bill and the instruction sheet are included in this chapter.

It is suggested that the student refer constantly to the original invoice, in this chapter. We are assuming that Accounting has divided this job into 13 fields. These fields and the number of columns assigned to each field are as follows:

Table 8.1 Reprinted by permission from IBM

Columns	Data
1-20	Description
21-25	Invoice no.
26-29	Invoice date
30-31	Branch (Br.)
32-33	Salesman (S'M'N)
34-36	City
37-38	State (St.)
39-43	Customer number (Cust. No.)
44-47	Quantity
48-52	Item number
53-58	Unit price
59-64	Amount
65-80	Skip

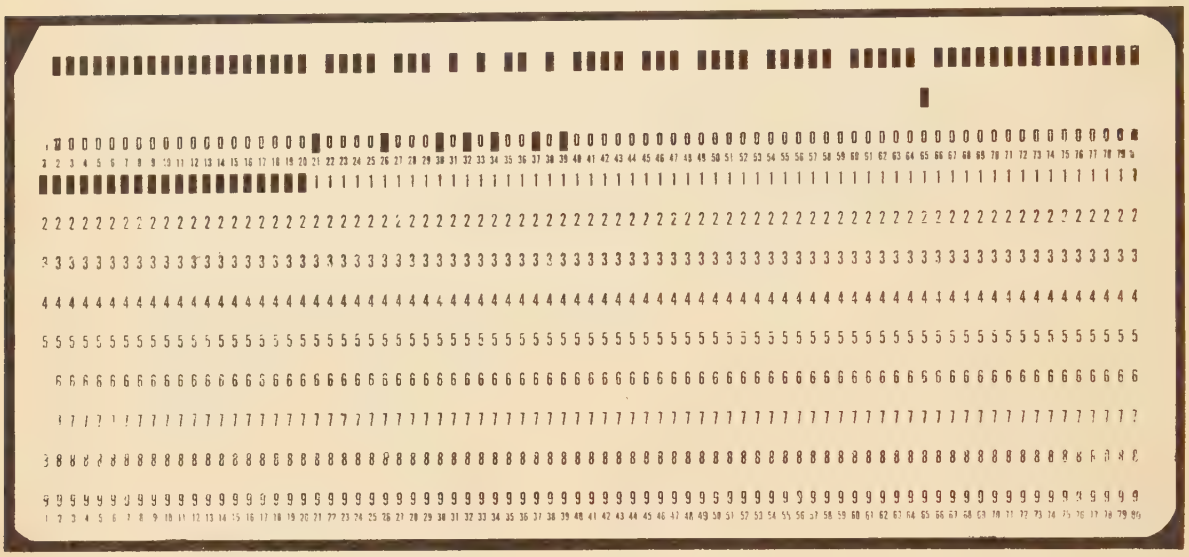
A program card which will fit a job like this, will carry the following fields:

Columns	Function	Codes	
		High priority punch	Rest of the field
1-20	alphabetic punching	1	A
21-25	numeric duplication	0	&
26-29	numeric duplication	0	&
30-31	numeric duplication	0	&
32-33	numeric duplication	0	&
34-36	numeric duplication	0	&

(cont.)

37-38	numeric duplication	0	&
39-43	numeric duplication	0	&
44-47	numeric punching	Space	&
48-52	numeric punching	Space	&
53-58	numeric punching	Space	&
59-64	numeric punching	Space	&
65-80	automatic skipping	X	&

Such a program card is shown below.



Dates are punched, as indicated in a previous chapter, in digitized numeric form. For instance, this invoice of January 8 is indicated 0108, 01 for January and 08 for the day. Another interesting feature is that branches and salesmen numbers are coded to simplify the accounting job. In this particular case, 2 positions per field. Cities are numerically coded and the codification will be decided by each accounting department according to its own needs. In general, states are listed in alphabetical order and coded in numerical sequence following the alphabetical order. In this bill (122 and 44, respectively), city and state could have been Burlington and Vermont. It is much easier to punch two digits than the complete alphabetized name. Likewise, each customer is coded as a number, 87631 in this case.

It can be observed that fields 21-25; 26-29; 30-31; 32-33; 34-36; 37-38 and 39-43 are all numeric duplicated. The preparation of the program card could be simplified making columns 21 through 43 as a single numeric duplicated field, with an 0 in the first column and with &'s for the rest of the field.

The first card of any job (the master card) must be punched manually, directly from the document, which in this case is the invoice. Each time the information to be duplicated changes, a new master card must be prepared. We will now assume that the program card has been punched accordingly to the instructions mentioned above and that this program card is already mounted on the drum, the drum is in the program unit and the program

control lever is ON where it will remain until this job (or any other using the same program card) is finished. The procedure to follow is: (1) place about 500 cards in the hopper; (2) key FEED twice; (3) flip the AUTO FEED functional switch to ON; (4) keep the functional switch AUTO SKIP/DUP OFF momentarily; (5) start punching the master card as indicated below:

Columns	Data to enter
1-20	Sweet potatoes
21-25	42401
26-29	0108
30-31	23
32-33	98
34-36	126
37-38	44
39-43	87631
44-47	0070
48-52	23912
53-58	000210
59-64	014700

At this stage, a number of observations can be made. The first field requires a smaller number of columns than those assigned in the program card. In all these cases, after keying the description of the commodity, if there are some remaining columns, these are simply skipped by depressing SKIP (left justification). This applies to alphabetic fields. The opposite applies to numeric fields where the number of significant digits to punch is less than the number of columns assigned to that field (fields 44-47; 53-58 and 59-64). In this, and in all similar cases, keying is carried out in a fashion such that the last significant digit falls on the last column assigned to that field (right justification). In these fields, the first significant digit is preceded with zeros, so as to fill them. The L-Z functional switch in conjunction with proper program codes can be used to that effect.

The operator must check the accuracy of the information punched on the master card. If it is correct, flip the AUTO SKIP/DUP functional switch ON. From this moment on, the cardpunch will duplicate onto all the following cards, the data punched in these columns (21-43). This duplication will be automatically repeated under program control, without any further action from the operator. It is therefore obvious that the information punched in the master card be correctly keyed. Otherwise all the erred information will be automatically duplicated and the job will require corrections. Finally,

65-80 Automatic Skipping

The card, under program control, will automatically skip to column 80 and a new card will be registered at the punching station.

The punching job is now extremely simplified since all the data to be duplicated will be automatically duplicated under program control. The data to be keyed in the first field of the second card is FLY PAPER, followed by a depression of SKIP. Columns 21-43 will be automatically duplicated. The operator follows up by keying 0027 and so on.

The operator is frequently handed the information sheet along with the source document. This information sheet generally has the format shown in the next figure. It is very handy, easy to read and carries all the information the operator needs to prepare the program card and start the job. Under "Remarks" there is information about the type of field to key. Letter A stands for an alphabetic field. N for numeric. A Dup for alphabetic duplicated. N Dup for numeric duplicated. A/N for alpha-numeric, P/Z for precede with zeros and L-Z, for field with left zeros.

CARD PUNCHING AND VERIFYING INSTRUCTIONS

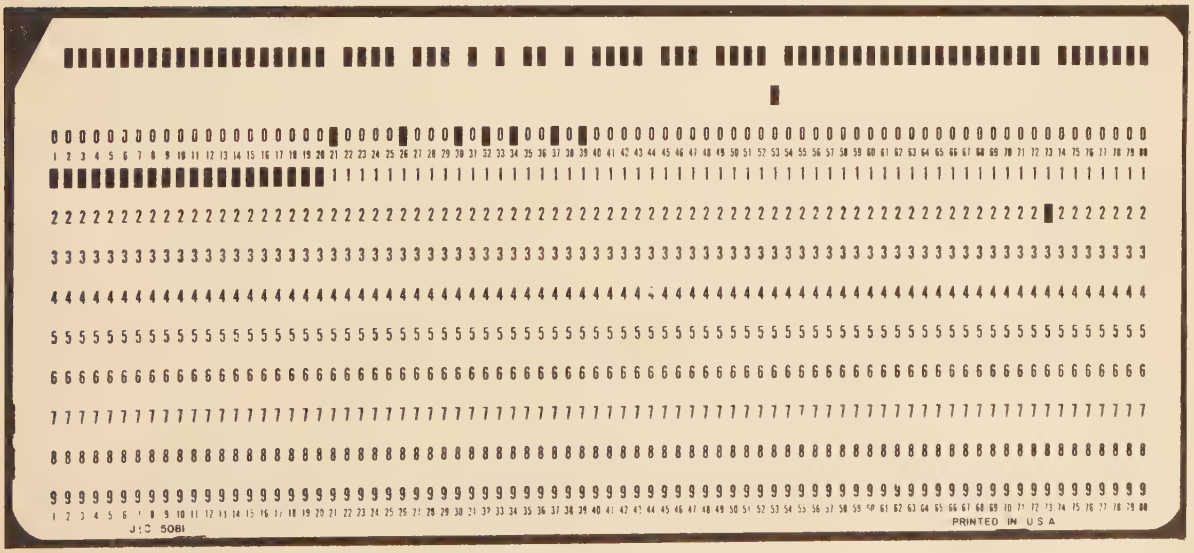
JOB NAME: BILLING			JOB NO.: 347
DATE: 3-4-70	NUMBER OF COLUMNS PUNCHED PER CARD	NUMBER OF COLUMNS VERIFIED PER CARD	PAGE 1 OF 1
SOURCE RECORDS RECEIVED FROM: Accounting		FREQUENCY	
SOURCE DOCUMENTS USED		DAY	
CARD FORM USED: Brown Edge		TIME	

FIELD NAME	COLUMNS FROM	COLUMNS TO	NO. OF COLUMNS	REMARKS
Commodity	1	20	20	A
Invoice #	21	25	5	N Dup
Invoice Date	26	29	4	N Dup
Branch (Br.)	30	31	2	N Dup
Salesman (S'm'n)	32	33	2	N Dup
City	34	36	3	N Dup
State	37	38	2	N Dup
Customer #	39	43	5	N Dup
Quantity	44	47	5	N P/Z
Item	48	52	5	N
Unit Price	53	58	6	N
Amount	59	64	6	N P/Z
Skip	65	80	16	
				Punch and Verify.

9 Costs

The job of calculating costs is directly linked to sales. In general the accounting department prepares simultaneously a study on costs and inventories as well as billing. Furthermore, another series of jobs can also be based on the same bill such as sales by salesman (to compute commissions), sales by cities and states, etc. All these jobs, before the computer era, were performed manually. Now, on the basis of one single invoice (like the one shown in the previous chapter and in Chapter 1), all this information can be readily handled with punched cards and tapes.

It is very simple to calculate costs using such a bill, using the following approach. The program card is prepared in a fashion similar to the one described in the previous chapter. The information sheet reflects the few modifications applied and the program card, prepared with that instruction sheet, is shown below.



CARD PUNCHING AND VERIFYING INSTRUCTIONS

JOB NAME			JOB NO.
COSTS			348
DATE: 8-3-70	NUMBER OF COLUMNS PUNCHED PER CARD	NUMBER OF COLUMNS VERIFIED PER CARD	PAGE 1 OF 1
SOURCE RECORDS RECEIVED FROM: Accounting		FREQUENCY	
SOURCE DOCUMENTS USED		DAY	
CARD FORM USED: Yellow Edge		TIME	

FIELD NAME	COLUMNS FROM	COLUMNS TO	NO. OF COLUMNS	REMARKS
Commodity	1	20	20	A
Invoice #	21	25	5	N Dup
Invoice Date	26	29	4	N Dup
Branch (Br.)	30	31	2	N Dup
Salesman (S'm'n)	32	33	2	N Dup
City	34	36	3	N Dup
State	37	38	2	N Dup
Customer #	39	43	4	N Dup
Quantity	44	47	4	N P/Z
Item #	48	52	5	N
Skip	53	72	20	
Net Cost	73	80	8	L-Z
				Punch Only 1

CARD PUNCHING AND VERIFYING INSTRUCTIONS

JOB NAME			JOB NO.
BILLING AND COSTS			347 A
DATE: 3-4-70	NUMBER OF COLUMNS PUNCHED PER CARD	NUMBER OF COLUMNS VERIFIED PER CARD	PAGE 1 OF 2
SOURCE RECORDS RECEIVED FROM: Accounting		FREQUENCY	
SOURCE DOCUMENTS USED		DAY	
CARD FORM USED: Yellow Edge		TIME	

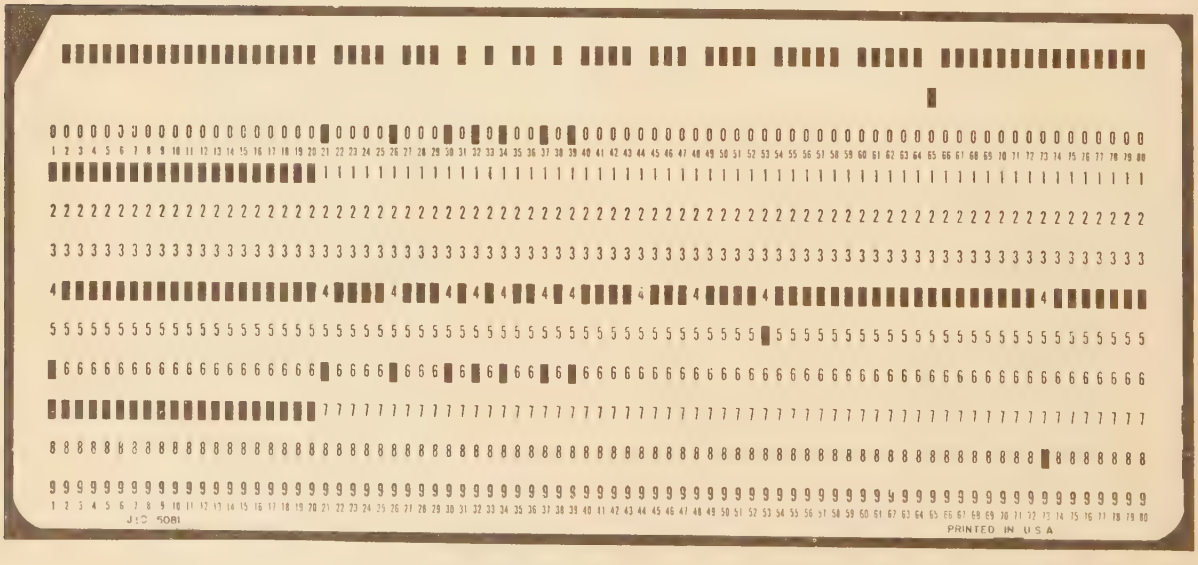
FIELD NAME	COLUMNS FROM	COLUMNS TO	NO. OF COLUMNS	REMARKS
Commodity	1	20	20	A
Invoice #	21	25	5	N Dup
Invoice Date	26	29	4	N Dup
Branch (Br.)	30	31	2	N Dup
Salesman (S'm'n)	32	33	2	N Dup
City	34	36	3	N Dup
State	37	38	2	N Dup
Customer #	39	43	5	N Dup
Quantity	44	47	4	N P/Z
Item #	48	52	5	N
Unit Price	53	58	6	N
Amount	59	64	6	N P/Z
Skip	65	80	16	
ALTERNATE PROGRAM.				
Commodity	1	20	20	A
Invoice #	21	25	5	N Dup
Invoice Date	26	29	4	N Dup
Branch (Br.)	30	31	2	N Dup
Salesman (S'm'n)	32	33	2	N Dup
City	34	36	3	N Dup
State	37	38	2	N Dup
Customer #	39	43	4	N Dup

CARD PUNCHING AND VERIFYING INSTRUCTIONS

JOB NAME			JOB NO.
Billing and Costs (Cont.)			347 A
DATE: 3-4-70	NUMBER OF COLUMNS PUNCHED PER CARD:	NUMBER OF COLUMNS VERIFIED PER CARD:	PAGE 2 OF 2
SOURCE RECORDS RECEIVED FROM: Accounting		FREQUENCY:	
SOURCE DOCUMENTS USED:		DAY:	
CARD FORM USED: Yellow Edge		TIME:	

FIELD NAME	COLUMNS FROM	COLUMNS TO	NO. OF COLUMNS	REMARKS
Quantity	44	47	4	N P/Z
Item #	48	52	5	N
Skip	53	72	20	
Net Cost	73	60	8	L-Z
				Punch and Verify

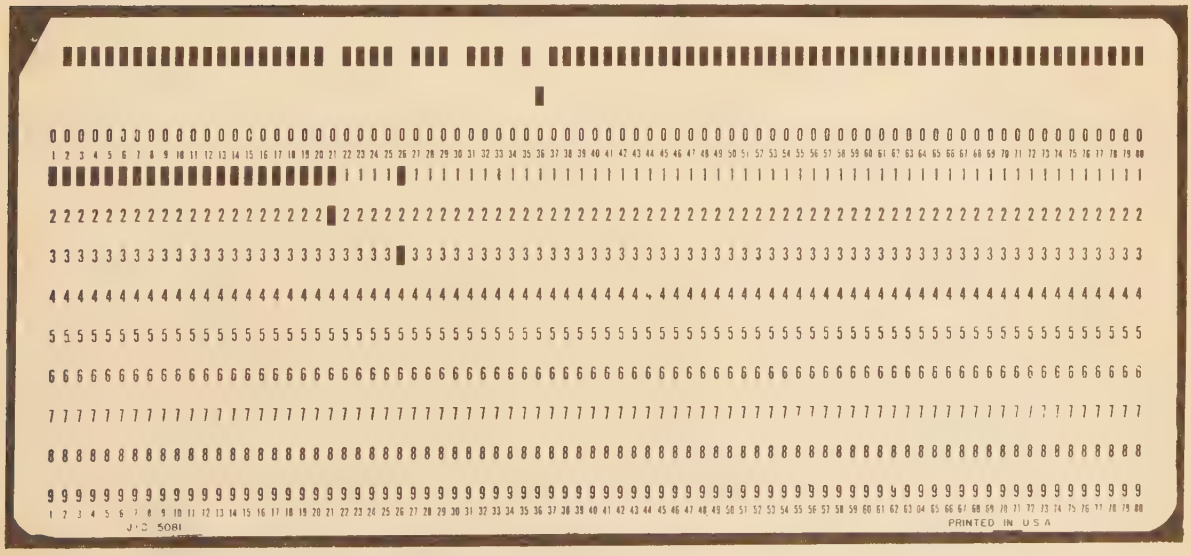
A brief study of billing and costs shows that both programs can be combined in one, since most of the data fields coincide. The attached instruction sheet shows how to prepare this double program card which is shown below.



10 Inventories

Other jobs and programs can be prepared on the basis of the same invoice. These other jobs are inventories. And, as we have previously done, we will use the same invoice shown for billing and costs.

The task of preparing and keeping control inventories for each commodity is easy when the quantities of each item sold are punched in the cards. These invoice cards, in combination with those derived from purchases of the same commodity, help facilitate the annual inventory. An information sheet to prepare the program card for inventories and the program card are included in this chapter.



CARD PUNCHING AND VERIFYING INSTRUCTIONS

JOB NAME			JOB NO.
INVENTORIES			531
DATE: 5-5-69	NUMBER OF COLUMNS PUNCHED PER CARD	NUMBER OF COLUMNS VERIFIED PER CARD	PAGE 1 OF 1
SOURCE RECORDS RECEIVED FROM: Stock Room		FREQUENCY	
SOURCE DOCUMENTS USED		DAY	
CARD FORM USED: Yellow Edge		TIME	

FIELD NAME	COLUMNS FROM	COLUMNS TO	NO. OF COLUMNS	REMARKS
Commodity	1	20	20	A
Item #	21	25	5	L-Z, 5 columns
Quantity	26	29	4	L-Z, 4 columns
Date	30	33	4	N
Branch	34	35	2	N
Skip	36	80	45	
				Punch Only 1

11 Payroll

This case is a study of payrolls. For each payroll employee a card has to be prepared. Each individual card carries a great variety of information which may be used for numerous payroll activities. For study purposes, it is advisable to keep handy and consult the payroll summary included in this chapter. This job has been divided into 15 fields.

Columns	Description	
1-16	Employee	
17	Sex	
18-26	Social security number	
27-28	Dept.	
29-31	Clock	
32-35	Rate	
36-41	Date	
42-44	Reg	Hours
45-47	O.T. (overtime)	Hours
48-51	Total	Hours
52-56	Regular	Earnings
57-60	Overtime	Earnings
61-65	Total	Earnings
66-78	Skip*	
79-80	Tax	

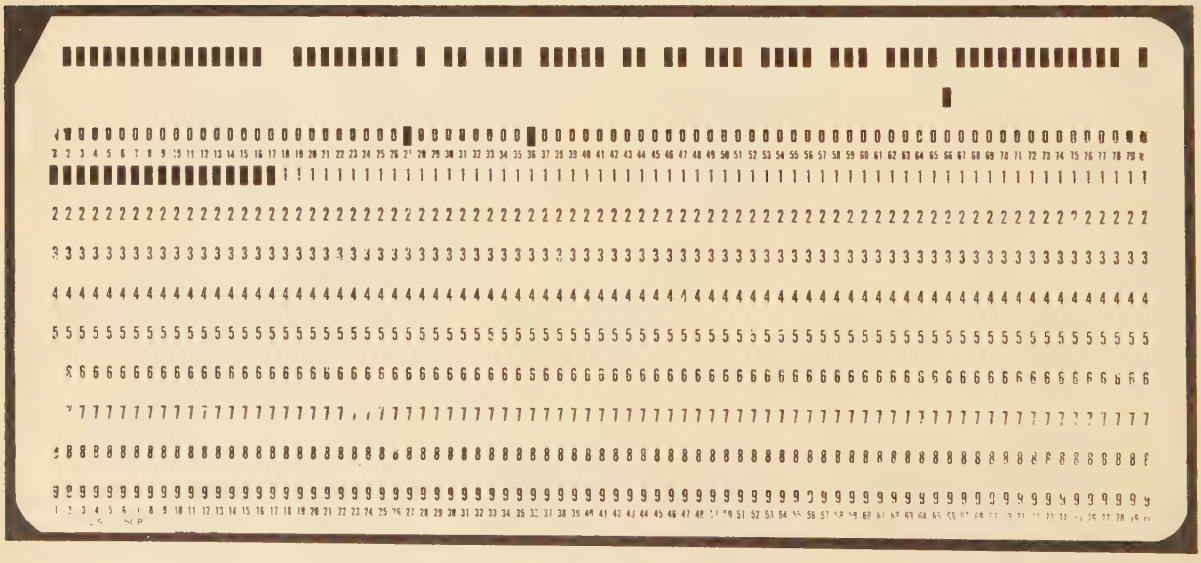
*This column has been purposely left blank for future accounting purposes.

A study of the requirements for a program card, suitable for this job, shows the following distribution:

Columns	Function	Codes	
		High priority punch	Rest of the field
1-16	alphabetic punching	1	A
17	alphabetic punching	1*	
18-26	numeric punching	Space	&
27-28	numeric duplication	0	&
29-31	numeric punching	Space	&
32-35	numeric punching	Space	&
36-41	numeric duplication	0	&
42-44	numeric punching	Space	&
45-47	numeric punching	Space	&
48-51	numeric punching	Space	&
52-56	numeric punching	Space	&
57-60	numeric punching	Space	&
61-65	numeric punching	Space	&
66-78	automatic skipping	X	&
79-80	numeric punching	Space	&

*Single column fields are always coded with the high priority punch. For alphabetic duplicating fields it is a double punch, digits 0 and 1 (which is the / symbol). A numeric field is a blank space while a numeric duplicating field is a 0 punch. An automatic skip is an X-punch.

The program card prepared as per these requirements is shown below.



Before going into the preparation of the master card, we will discuss the payroll summary included in this chapter. For the "Rate" field, columns 32-35, a 4-column field is available, of which the first column is the rate per hour unit, expressed in dollar unit. The other three columns are dollar fractions up to a thousandth. Therefore, a rate of 85 cents per hour is listed as 0850; a rate of $1.125 is expressed as 1125; and $1.65 per hour is 1650. As previously mentioned, the decimal point is not punched and the total keyed number is punched in such fashion that the last digit falls in the last column of this field, even if it is a zero.

On keying the number of hours, total number of hours and their decimal fraction (and not the number of minutes) is keyed. For this purpose, the clock dial is divided into 10 equal decimal fractions of 6 minutes each according to this table:

Minutes	Fraction
1-6	0.1
7-12	0.2
13-18	0.3
19-24	0.4
25-30	0.5
31-36	0.6
37-42	0.7
43-48	0.8
49-54	0.9
55-60	1.0

Consequently, for columns 42-44, 45-47 and 48-51, which carry hours, overtime and total time in hours and fraction, in all 3 cases the last column carries the fraction expressed as indicated in the table above. For instance, if a worker has accumulated 5 hrs and 10 minutes as overtime, his O.T. will show 5.2. Since the decimal point is not punched, the digits actually keyed are 52, and care should be exercised to make sure that the last digit falls in the last column of this 3-position field. Another number which could be punched is 052. The total number of hours can never reach 100. Consequently since 4 positions have been assigned to the 48-51 field, it would be advisable to precede the number with zeros. The use of programmed L-Z is optional and may vary with the accounting procedure used in each institution. The master card should be punched as follows:

Columns	Data
1-16	Mary Oakdale (skip the rest of the field)
17	F
18-26	439039875
27-28	06*
29-31	336
32-35	0850
36-41	10160

(cont.)

42-44	400
45-47	052*
48-51	0452*
52-56	03400*
57-60	0663*
61-65	04063*
66-78	Skip
79-80	03*

*In all these cases where the number of assigned positions is greater than the number of digits to be punched, it is advisable to precede the first significant digit with left zeros. Fields 27-28 and 36-41 are numeric duplicating. The 36-41 field includes date, expressed as month (10), day (16) and year (0).

CARD PUNCHING AND VERIFYING INSTRUCTIONS

JOB NAME: PAYROLL			JOB NO.: 089
DATE: 6-6-66	NUMBER OF COLUMNS PUNCHED PER CARD	NUMBER OF COLUMNS VERIFIED PER CARD	PAGE 1 OF 1
SOURCE RECORDS RECEIVED FROM: Personnel		FREQUENCY	
SOURCE DOCUMENTS USED		DAY	
CARD FORM USED: Brown Edge		TIME	

FIELD NAME	COLUMNS FROM	COLUMNS TO	NO. OF COLUMNS	REMARKS
Employee	1	16	16	A
Sex		17	1	A
Social Security Number	18	26	9	N
Dpt.	27	28	2	N, P/Z
Clock	29	31	3	N
Rate	32	35	4	N
Date	36	41	6	N Dup
Hours, Regular	42	44	3	N
Overtime (O.T.)	45	47	3	N, P/Z
Total Hours	48	51	4	N, P/Z
Earnings, Regular	52	56	4	N, P/Z
Overtime Earnings	57	60	4	N, P/Z
Earnings, Total	61	65	5	N, P/Z
Skip	66	78	13	
Tax Status	79	80	2	N, P/Z
				Punch and Verify.

Instructions on the sequence for this job (how to prepare the program card, the master card and the following cards) are the same as in billing. The instruction sheet and a copy of a payroll summary are included in this chapter.

PAYROLL SUMMARY

DATE 10/16

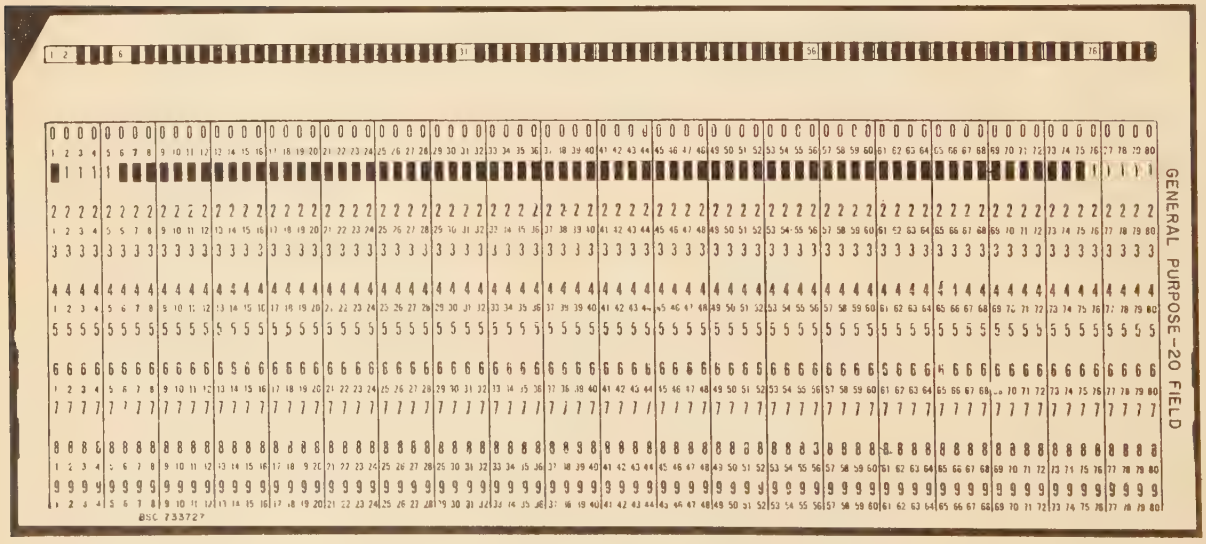
ENTRY 3

EMPLOYEE	SEX	SOCIAL SECURITY NUMBER	EMPLOYEE NO.		RATE	HOURS			EARNINGS			TAX STATUS
			DEPT.	CLOCK		REG	O. T	TOTAL	REGULAR	OVERTIME	TOTAL	
MARY OAKDALE	F	439039875	06	336	850	400	52	452	3400	663	4063	3
RAY S VIVIAN	M	671258360	06	365	850	400	100	500	3400	1275	4675	4
ANNE HOSPODOR	F	247329362	06	454	1125	305		305	3431		3431	1
FRANK E ERLENMEYER	M	100343834	06	458	1145	400	81	481	4580	1392	5972	4
HORTENSE SCULLINGER	F	169022151	06	605	890	400		400	3560		3560	1
JANE HILLOCK	F	282366417	06	652	1160	400	100	500	4640	1740	6380	2
JAMES T LEED	M	626673988	06	655	1030	400	53	453	4120	819	4939	4
PAUL K JACKSON	M	941345381	06	656	850	400	100	500	3400	1275	4675	4
ELSIE R GERSTEN	F	117132198	06	751	1025	400	100	500	4100	1538	5638	1
BARBARA WOLF	F	960792339	06	760	1025	400	99	499	4100	1523	5623	2
NORMA ADAMOWSKI	F	773603342	06	849	1140	400		400	4560		4560	1
POLLY PETERSON	F	221898831	06	859	850	400	63	463	3400	803	4203	1
PATRICIA P NEVENTI	F	884480050	06	861	1160	400	110	510	4640	1914	6554	1
JACK MOSHER	M	847497543	06	901	1025	400		400	4100		4100	4
JIM FITZGERALD	M	515048948	06	964	1140	400	112	512	4560	1915	6475	4
CHECK SUM		202167380	06	8734	734	094	29	124	40008	5142	25151	62

12 Lists of customers

Sometimes the advertising department has to reach all customers with notes affecting their daily activities, such as the addition of new items to a catalog, price changes, notes of a general nature, mailing monthly invoices and balances.

In these jobs it is very common to use the cardpunch combined with the computer. Customer lists are punched on cards which are then processed by the computer. A typical case is a customer list enclosed in this chapter. From this list the operator can prepare an instruction sheet and the program card which are included below.



CARD PUNCHING AND VERIFYING INSTRUCTIONS

JOB NAME: CUSTOMER LIST			JOB NO. 374
DATE: 5-3-69	NUMBER OF COLUMNS PUNCHED PER CARD	NUMBER OF COLUMNS VERIFIED PER CARD	PAGE 1 OF 1
SOURCE RECORDS RECEIVED FROM: Distribution		FREQUENCY	
SOURCE DOCUMENTS USED		DAY	
CARD FORM USED: Blue Cards		TIME	

FIELD NAME	COLUMNS FROM	COLUMNS TO	NO. OF COLUMNS	REMARKS
Customer Letter		1	1	A
Customer Number	2	5	4	N, P/Z
Name	6	30	25	A
Address	31	55	25	A/N
City and State	56	75	20	A
Zip	76	80	5	

CAR POOLS, Inc.

(201) 488-7401
P.O.B. 165
Cliffside Park, New Jersey
07010

CUSTOMER LIST

Customer No.	Name	Address	City and State	Zip
C 963	Cardin, John	99 E. Frams Ave.	Absecon, N.J.	08201
C 964	Carvalho, Ann	123 Stedman St.	Dayton, Ohio	45401
C 965	Carver, David	4228 Prospect Ave.	Portland, Oregon	97201
C 966	Cary, Howard	163 Gretna Expressway	Altoona, Pa.	16603
C 967	Casaletto, Dora	705 Rosegund St.	Minneapolis, Minn.	55441
C 968	Casanova, Jaime	21 Woodland Place	Leonia, N.J.	07605
C 969	Casavant, Carl	3116 Dumont Drive	Indianapolis, Ind.	46206
C 970	Case, James	6635 Neosho Drive	Providence, N.Y.	10034
C 971	Case, Laura	4 Elliott Drive	Abie, Nebraska	06801
C 972	Casebier, Ronald	821 Emerald Terrace	Houston, Texas	77001
C 973	Casedy, Gary A.	7740 N. 15th. Ave.	Norfolk, Va.	23504
C 974	Casella, Peter	2724 S. Peck Rd.	Louisville, Ky.	40222
C 975	Caselli, Albert	1535 Lombard St.	Ackley, Iowa	50601
C 976	Caserio, Frank	POB 386	Whittier, Cal.	90601
C 977	Casey, Andrew	4960 N. Marine Dr.	Livingston, Ky.	40401
C 978	Cassidy, Butch	15 N. South Place	Yonkers, N.Y.	10701
C 979	Cassone, Andrea	250 W. 125th. St.	Lowell, Mass	01851
C 980	Castaldi, Claude	766 Edgewood La.	Arlington, Ill.	60006
C 981	Castin, Deborah	201 N. Oakdene Place	Pryor, Col.	81065

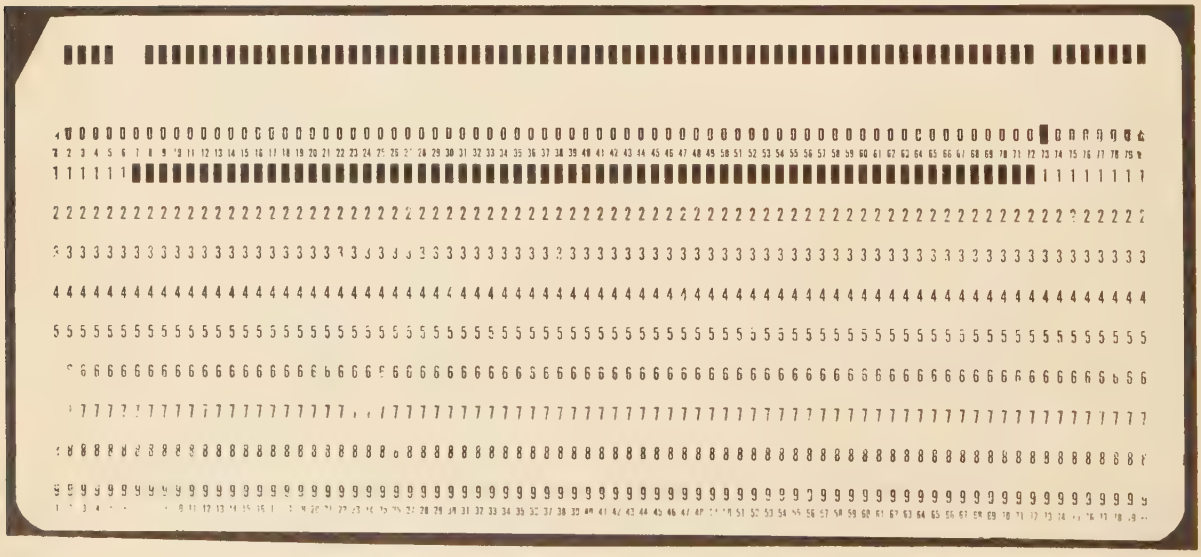
13 Fortran

One of the most frequent applications of cardpunches is the preparation of programs for computers. Communication between the programmer and the computer is carried out through "languages" for which the computer is electronically prepared. Two such languages are Fortran and Cobol.

The programmer prepares the program for the computer in the appropriate "language" and supplies the cardpunch operator with coded forms where the program statements are listed. Simultaneously, the operator is also supplied with a numeric data sheet. This is the numeric information which the computer will eventually process with the instructions given in the program statements. Consequently, the operator will have to prepare:

1 cards with the program for the computer, with coded information carrying information, and
2 cards with numeric information which the computer will use on carrying out the instructions received.

An instruction sheet, first page of a Fortran coding form and the program card for the computer program are included below.



CARD PUNCHING AND VERIFYING INSTRUCTIONS

JOB NAME: FORTRAN			JOB NO.: 100
DATE: 6-7-66	NUMBER OF COLUMNS PUNCHED PER CARD	NUMBER OF COLUMNS VERIFIED PER CARD	PAGE 1 OF 1
SOURCE RECORDS RECEIVED FROM: Operations Analysis			FREQUENCY
SOURCE DOCUMENTS USED			DAY
CARD FORM USED			TIME

FIELD NAME	COLUMNS FROM	COLUMNS TO	NO. OF COLUMNS	REMARKS
Statement #	1	5	5	N
Continuation		6	1	N
Statement	7	72	66	A/N
Identification	73	80	8	N Dup.
				Punch and Verify

FORTRAN CODING FORM

Program | Punching Instructions: Graphic | Card Form # 5081 | Page 1 of 4

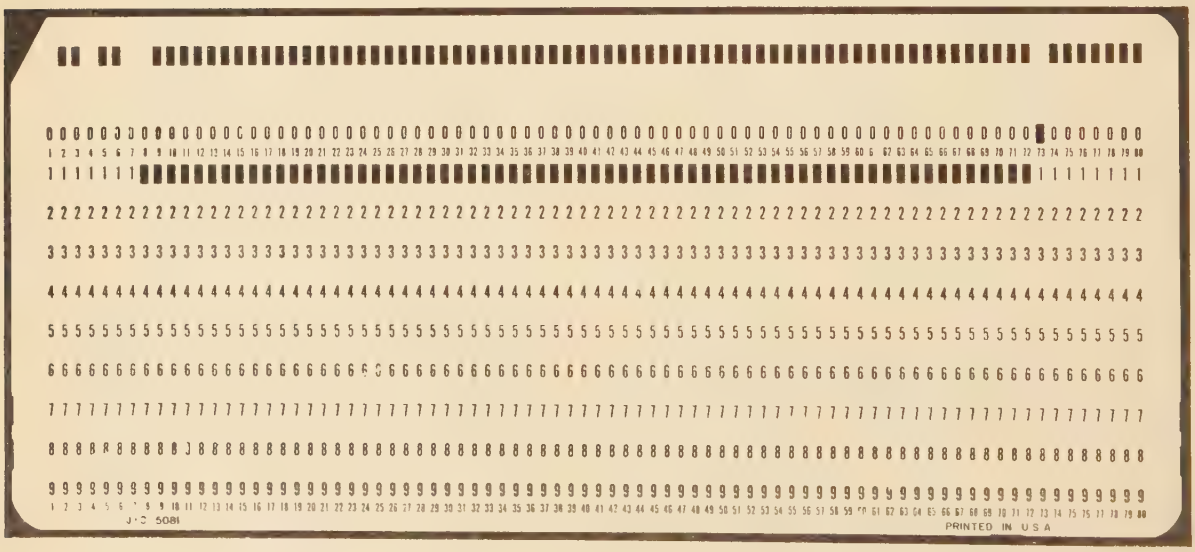
Programmer: GARCIA | Date: 12/8 | Punch | 360 | Identification (73–80)

C FOR COMMENT

STATEMENT NUMBER	CONT.	FORTRAN STATEMENT
		DIMENSION MAT(8,8)
		PCNT=0.
		MCNT=0
		PTOT=0.
		SPTOT=0.
		MAX=0
		MIN=100
		DØ 90 K1=1,8
		DØ 90 K2=1,8
90		MAT(K1,K2)=0
80		READ(5,102)I,J,M1
102		FØRMAT(2I1,I3)
		IF(I.EQ.0)STØP
40		X=RANF(0.0)
		K=IFIX(X*10.)
		IF(K.EQ.0.ØR.K.GT.8)GØ TØ 40
		DØ 70 M=1,M1
30		DØ 10 IP=1,8
		GØ TØ (1,2,3,4,5,6,7,8),K
	1	I1=I+1
		J1=J+2

14 Cobol

Cobol is a relatively new language which has become very popular in business and industry. Cobol is an acronym for *CO*mmon *B*usiness *O*riented *L*ogic. Its application is, in general, somewhat different from Fortran, since this last one is mostly applied to technology. Nevertheless the preparation of the program card for Cobol is very similar to Fortran, as shown by the attached instruction sheet, the program card and a copy of a Cobol coding form enclosed in this chapter.



CARD PUNCHING AND VERIFYING INSTRUCTIONS

JOB NAME: COBOL			JOB NO.: 101
DATE: 6-7-66	NUMBER OF COLUMNS PUNCHED PER CARD:	NUMBER OF COLUMNS VERIFIED PER CARD:	PAGE 1 OF 1
SOURCE RECORDS RECEIVED FROM: Operations Analysis		FREQUENCY:	
SOURCE DOCUMENTS USED:		DAY:	
CARD FORM USED:		TIME:	

FIELD NAME	COLUMNS FROM	COLUMNS TO	NO. OF COLUMNS	REMARKS
Sequence, Page	1	3	3	N
Sequence, Serial	4	6	3	N
Continuation		7	1	N
Statement	8	72	64	A/N
Identification	73	80	8	N Dup
				Punch and Verify.

IBM COBOL Coding Form

SYSTEM: COBOL
PROGRAM: R-29-0245-0
PROGRAMMER: IBM DATE:
PUNCHING INSTRUCTIONS — GRAPHIC / PUNCH / CARD FORM # / *
PAGE OF
IDENTIFICATION 73–80

SEQUENCE (PAGE)	SEQUENCE (SERIAL)	CONT	A	B	COBOL STATEMENT
01	01				
	02				PROCEDURE DIVISION
	03				
	04				START-RUN
	05				OPEN INPUT MASTER-FILE; OUTPUT CREDIT-LETTERS.
	06				
	07				READ-MASTER.
	08				MOVE SPACES TO OUTPUT-LINE.
	09				READ MASTER-FILE;
	10				AT END, CLOSE MASTER-FILE, CREDIT-LETTERS; STOP RUN.
	11				MOVE CUSTOMER-NAME TO DATA-AREA.
	12				WRITE OUTPUT-LINE AFTER ADVANCING 0.
	13				MOVE STREET-ADDRESS TO DATA-AREA.
	14				WRITE OUTPUT-LINE AFTER ADVANCING 1.
	15				MOVE CITY TO DATA-AREA.
	16				WRITE OUTPUT-LINE AFTER ADVANCING 1.
	17				GO TO READ-MASTER.
	18				
	19				
	20				

*A standard card form IBM Electro C81897, is available for punching source statements from this form
Instructions for using this form are given in any IBM COBOL reference manual.
Address comments concerning this form to IBM Corporation, Programming Publications, 1271 Avenue of the Americas, New York, New York 10020

Form No. X28-1464-4 U M 025
Printed in U.S.A.

15 Special card punches

The specific applications described below are limited exclusively to certain models of the 029 cardpunch. These special models are prepared in the plant for some special requirements.

Auxiliary Duplication
This feature allows the direct duplication from a second master card instead of duplicating from the card at the reading station. The information to be duplicated is punched in a master card which is then mounted on an auxiliary drum, of the same type as the program drum. The main difference is that the auxiliary drum is housed in the rear of the cardpunch, below the program unit. This auxiliary system has its own sensing mechanism and keyboard control. In other words, there are two program drums. The AUX DUP key manually controls this duplicating system.

This feature is quite advantageous when there is information to be selectively duplicated in certain cards but not in others; or when certain cards have to be duplicated because they carry some information and/or when pre-punched master cards have to be intersped.

Self-Checking
This feature permits the simultaneous verification of pre-coded numeric information at the same time it is keyed. This voids the need for a second verification.

Cards carrying information which varies from card to card (quantities, prices, etc.) only, must be subject to a second verification. This self-checking feature can be installed in model A of the 029 cardpunch. There are two self-checking types: "Modulus 10" and "Modulus 11". Modulus 10 is prepared to verify the incorrect keying of one digit or of a single digit transposition. Modulus 11 can also verify a double digit transposition. Appendices 1 and 2 show how to calculate the self-checking digit for both Modulus.

These self-checking verifiers are equipped with special switches on the keyboard which produce special notches on cards when the cards have been verified without errors. They also have a special light on the keyboard to indicate to the operator the presence or absence of keying errors.

Card Insertion
There are special cardpunch models which have funtional switches for the insertion of cards with specific functions, such as the insertion of a master card in front of a group of blank cards or the insertion of a blank card at the end of a group of punched cards.

Character Inhibit
There are special models of the 029 cardpunch which have a special switch, to the left of the program drum. This switch permits the operator the selection of a keyboard with either 48 or 64 characters. When a 48-character keyboard is selected, keying of any of the remaining 16, non-authorized characters will lock the keyboard. In those cases by depressing ERROR RESET the keyboard is unlocked.

High-Speed Skip
In all models of the 029 cardpunch (A, B and C) a special attachment can be installed which, when activated, can produce an automatic high-speed skip. This function is carried out under program control and in combination with the regular skip (also under program control), at normal speed. This high-speed skip is very advantageous when 55 (or more) columns must be skipped at one time. The program card for this special function requires special codifications. Let us assume we have a field (columns 5 through 75) to be skipped at high speed. The program card is basically prepared as for a regular skip, that is, with an 11-punch at column 5 and with &'s the rest of the field. If however, the digit 1 is punched in the program card from column 5 onwards, the machine will skip at high speed. Generally the 1-punches are stopped about 5 columns before the end of the field (and in this assumed case would be at column 70), so that the cardpunch will gradually slow down. This special program can be prepared at the normal level, as we have just indicated, as well as in the alternate level with a multiple punch in columns 5 and 7 for the high-priority punch followed by 4- and 7-punches through column 70. The remaining columns would be punches with 4's.

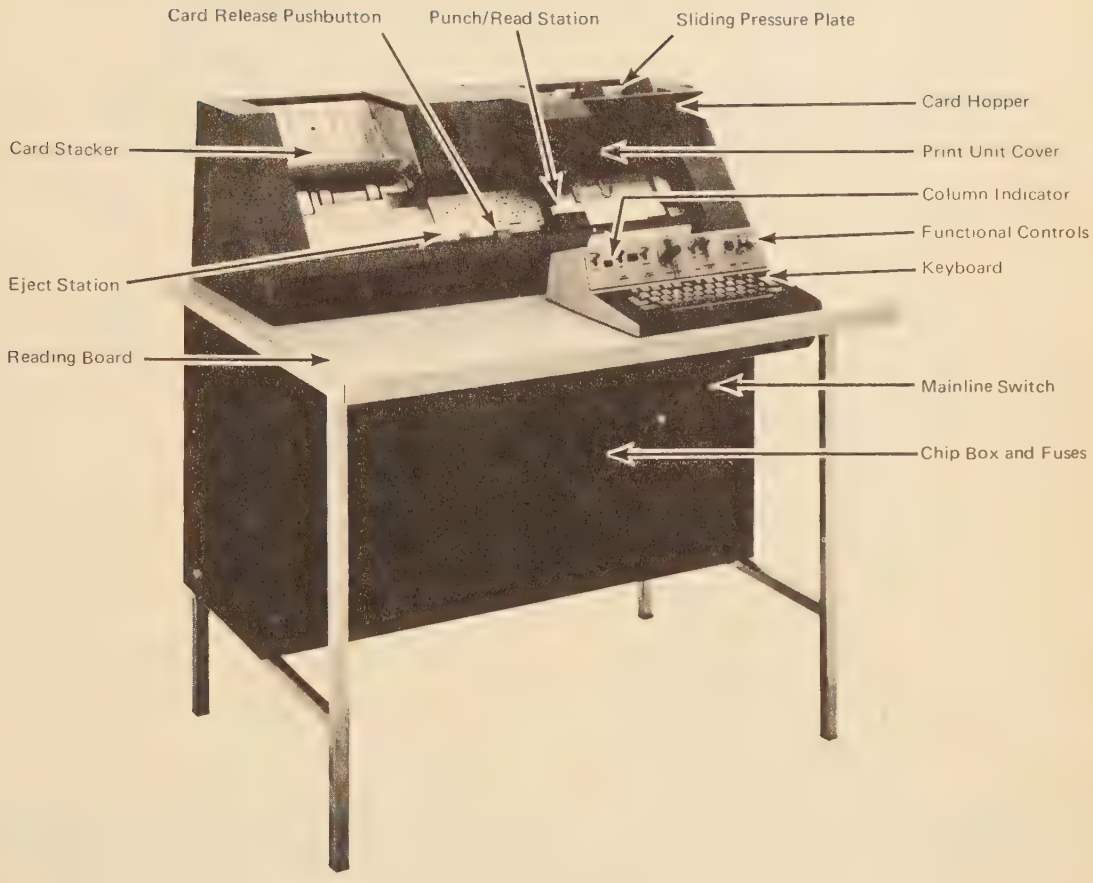
Interspersed Punching
This special feature permits the intercalation of cards with upper corner cuts opposite to that of the master card. Assume that cards with upper-right corner cuts are used for the master card and those with upper-left cuts for the regular ones (the inverse approach can also be used). In these special machines there is a lever which admits certain cuts (either right or left) but not the other. In these cases, under program control, the machine will duplicate from the preceding master card (with a specified cut) to the following cards (with a different cut) until it finds another card with the same cut. At this time, the cardpunch suspends duplication. Only when a new master card arrives at the reading station (keying FEED), will automatic duplication resume.

Variable-Length Cards
There are attachments for all models of the 029 cardpunch which permits the use of cards of 51, 60, 66 and 80 column lengths.

16 Verifiers

There are occasional instances when the operator may inadvertently commit keying errors. These erred cards may not be detected by the operator and the erred cards will be released as such. Since a punched card is an important record, it is essential that it be accurately keyed before it is further processed by the computer (or any other machine). Consequently, all the punched cards, in any job, are subjected to another operation called verification and performed in special machines called verifiers.

The verifier is a machine which looks like a cardpunch and can also function under program control. When, on keying or on automatic duplication, the verifier finds an erred punch, the machine locks and a red light turns on. An erred column can be verified three times in sequence (with three sequential lockings). After the third verification, the verifier is automatically released. In all three instances, the red light will turn itself on. On the other hand, when an accurately keyed card is verified and found without errors, a small notch is indented on the card's right side indicating that the card is ready for the next accounting step.



The IBM 129 Card Data Recorder Components

17 The IBM 129: Card Data Recorder

One of IBM's latest entries in the field of Electronic Data Processing is the 129, the Card Data Recorder (CDR or 129). This new machine combines a card puncher and a verifier in one, thus eliminating the need of two units for a job that can be done by one.

The CDR has a number of new features:

1 it can store six programs in electronic memory;
2 it has a device which puts the machine in alphabetic shift; (very handy for punching programs for computers);
3 it has the ability to correct immediately cards keyed in error, before they are actually punched;
4 or, on verification, it can make a new card immediately, with the correct data punched in;
5 it is much quieter than previous cardpunches.

Operators who are experienced with the 029 cardpunch and the 059

Figure 17.1 The IBM 129 Keyboard and Console Components

verifier will find it easy to use the CDR. Although there are a number of physical features of the CDR which make it very similar to the 029 and 059, its characteristics and versatility make this machine easier to handle, with a much wider range of applications.

A general description of the CDR's physical features follows and the reader is urged to read them carefully in order to understand the CDR's capabilities as a cardpunch and as a verifier. The subchapters on punch and verify modes, which follow, describe the greater operational advantages of the CDR.

PHYSICAL FEATURES

There is a striking physical resemblance between the metal body of the CDR and that of the 029 and the 059. There is the hopper, a card bed and a stacker. The console and keyboard are of the same size and shape as the 029 and it rests on the desk, where it can be moved about. But there is no program unit nor a program drum because the CDR's program controls operate on the basis of an electronic memory (instead of the physical circuitry provided by the star wheels). The 6-level electronic memory does not need a program drum. The column indicator is now located at the console. The backspace button, located at the base of the program unit in the 029, is a card release push-button in the CDR which permits the manual removal of a card from the punch/read station.

Backspacing either a column, a field or a complete card is another of the distinctive features that the CDR offers. The CDR's punch/read station is located where the punching station of the 029 used to be. The chip box and mainline switch are located below the desk, as in the 029. The mainline switch, however, discharges all the programs and data from memory when it is turned OFF. Then, when it is turned ON again, the program and data storage memories are blank and the machine is set at program level O (which is used only for alphabetic punching).

Regular cards, with upper left and right corner cuts are accepted by the CDR. Round corner cards can also be used but cards with lower corner cuts can not be fed. These 80-column cards are familiar to the operator. The CDR however, for verifying purposes, uses the column adjacent to column 80 which will be labelled column 81. When a card is verified with no errors, a 2-3 hole is punched automatically in column 81 to signify that the card has been verified without errors.

The keyboard in the CDR is also very similar to that of the 029, but since new functions have been added, some functional keys have been changed accordingly. The punching keys remain the same and, as before, the keyboard can be used as a 64- or a 48-character unit.

Major changes are noted at the console where new switches, a column indicator, a program selector disc and pushbuttons are found. Data are recorded at two storage memories, the input and output data storages. The data storages have a capacity of up to 80 columns or characters. Programs can be entered into an electronic memory with a capacity of up to six programs. The input/output storage and the program storage are two different memories.

The column indicator in the 029 has been replaced in the CDR by a

digitalized column indicator (at the console) which reads out the next column number to be keyed. Data keyed in, enters into input storage as an electronic signal but the card is not punched out. After column 80 has been keyed, the column indicator has moved to 00 but the card has not been punched yet. This allows time for the operator to "erase" from electronic storage any character (or field) keyed in error. Punchout takes place only when the next card is fed and the indicator changes from 00 to 01. At this time the data is transferred from input to output storage while, simultaneously, the card is actually punched out. The input storage is now blank, ready to accept new data keyed in or duplicated. Data to be duplicated is "copied" (under program control or manually) as an electronic impulse from output storage and becomes part of the new input record.

With the column indicator at 00, a character (or field), found in error, can be corrected. The machine can be backspaced (or backfielded) and the correct character (or characters) be keyed again at the same position.

In verifying operations, the punched cards to be verified are "read" into input storage and the data being keyed by the operator, in output storage, is "compared" with the data in the input storage. If no errors are found, the input data is "erased" when a new card is fed to be verified and the new data is accepted as new input. If, however, a card is found to be in error, a correct card can be punched out immediately following the verification of the complete card. Actually, any correct data keyed in at a particular column, for verification, "erases" the data existing in input storage at the same column location. Consequently, the corrected and verified data in input storage can be duplicated into a new, blank card if necessary. The new, corrected card, becomes a part of the job and the incorrect card can be discarded. This incorrect card will *not* be punched at column 81 with the 2-3 OK Verify punch mentioned before. Column 81 will be left blank, for easy identification. The new, corrected card, will carry a 2-punch (instead of the 2-3) to signify that a correction was made during the verify operation. Besides, the CDR has a pre-stacked position in the stacker which, in fact, makes the last card processed "pause" before being stacked. This facilitates the removal of cards which have been found in error and for which a corrected card has been prepared. These and other operational and functional characteristics of the CDR will become clear when the punch and verify modes are discussed.

PROGRAM CODES

The program cards prepared for the CDR use a simplified form of the same codes used by the 029 at the program one level. These codes are:

Function	First column	Rest of the field
AUTO SKIP	11	&
AUTO DUP	0	&
Alpha shift	1	A's
Numeric shift	blank	&

As before, &s are field definition punches. Any alphabetic or numeric field can be used either as a regular or LZ (left zero or left-blank) filled field (see LZ, below). The same program codes are used for punching and verifying (except for LZ verification). The procedures for punching program cards and for entering these programs in memory are described under PUNCH MODE.

The 11-punch is effected in alphabetic shift (the same key, in numeric shift, is the -LZ function). The 0-punch in the first column of a programmed field is the code for an automatic duplication field as well as the first column of an automatic verify field. The programmed AUTO SKIP or AUTO DUP functions are not operational unless the console AUTO SKIP/DUP switch is set at ON.

LEFT ZERO (OR LEFT-BLANK) FIELDS

Any field, from 2 to 79 columns long, regardless of whether the CDR is in alphabetic or numeric shift, can be used as a LZ field. No codes are necessary. The decision to use any field as a left zero (or left blank) rests with the operator. Assume we have a programmed alphabetic field of 8 columns and the name Joan is to be entered. The operator, after keying this 4-letter word, can proceed in two ways: (1) if SKIP is depressed, after keying JOAN, the field is filled with 4 blank columns to the right (JOAN _ _ _ _) (left justification). (2) If the L-Z tab is depressed, instead, the data is shifted to the right and the first 4 columns are left blank (_ _ _ _ JOAN) (right justification).

For numeric fields, the same reasoning applies. Assume that in a programmed 8-column numeric field, the number 1234 is to be entered. The operator, after keying 1234, can (1) depress SKIP and the field will show 1234 followed by 4 blank columns (1234 _ _ _ _) (left justification), or (2) depress the L-Z tab instead, and the data will be shifted to the right and left zeros will be punched in the first four columns (00001234) (right justification).

As indicated, no special codes are necessary for the left-zero (or left-blank) fields. Any alphabetic or numeric field can be used as a regular or a LZ field, at the convenience of the operator. But the LZ fields have the additional advantage that they can be used in conjunction with credit data. These "minus left zero" fields carry a minus sign over the unit digit (say 00001234̄). This can be effected by depressing the -LZ key instead of the LEFT ZERO CTRL tab.

When the LZ key is depressed, the machine first establishes under what shift data is being keyed. If the field is numeric, the positions to the left are filled with zeros. If it is an alphabetic field, the fill is blanks.

There are a number of possible uses and combinations of the new LZ in conjunction with numeric data. First, as mentioned, the shift is noted and then:

1 In numeric shift, manual (levels 1-6) or programmed fields, depressing the LZ tab (or the -LZ key) results in a left zero (or a minus left zero) operation. Data is shifted to the right and zeros are inserted to the left (00001234 or 00001234̄). Left zeros are punched but not printed. -LZ is inoperative when there is a full field.

2 Numeric data can also be entered in a programmed numeric field with blanks to the left. To this effect, enter all the digits; then, at the end of the field, depress ALPHA and then LZ (_ _ _ _ 1 2 3 4).

3 In cases that credit (a minus over the last digit) is to be entered with blanks (instead of zeros) to the left, in a programmed numeric field, enter all the digits except the last one. For the last digit use MULT PCH to enter the minus sign and the digit. Then depress ALPHA and LZ (_ _ _ _ 1 2 3 $\bar{4}$). The combinations negative 1 through negative 9 represent alphabetic characters which can be keyed in alphabetic shift, for the last digit of a credit. (See Table 18.4). Since the CDR can not distinguish between alphabetic characters and a negative digit, such combinations will be printed as the alphabetic characters J through R. The combination negative 0 (a double punch at the 11 zone and 0), is not a character recognized by the 129 and in these cases printing is suppressed.

4 Numeric data can be entered in an alphabetic field with blanks (instead of zeros) to the left. To that effect, keeping NUMERIC depressed, enter the digit data and then, after releasing the NUMERIC key, depress the LZ tab. The data is shifted to the right and blanks are at the left, before the first digit. (_ _ _ _ 1 2 3 4).

5 Numeric data can be entered in an alphabetic field with blanks to the left and a minus sign over the last digit (a minus left blank). To this effect, keeping NUMERIC depressed enter all the digits but the last one. At this point, release NUMERIC; depress MULT PCH and key the last digit and -LZ (depression of the -LZ key, when MULT PCH is depressed causes an 11-punch) (_ _ _ _ 1 2 3 $\bar{4}$).

The use of MULT PCH to enter a credit (or minus sign) is not restricted to the last column of the field; it can be used in any column. For an all-zero field, key one zero and then depress the LZ tab.

There are no restrictions to skip or duplicate manually LZ fields. Programmed verification of LZ fields require keying a 3-punch in the program card, in the first and last column of that field. All other columns of these fields are punched with field definition punches (&s).

KEYBOARD

The CDR's keyboard is very similar to that of the 029. The total number of keys, punching plus functional, are the same. The punching keys are identical, produce the same holes when depressed and are located at the same location as before.

There are however new functional keys which have replaced some of the 029 which, in the process, have become obsolete since the CDR has greater flexibility. These new functional keys and those of the 029 which they replace, are listed in this Table:

Table 17.1

	CDR	029	
	VER CORR	Blank (next to DUP)	
	VER DUP	DUP	
	-LZ —	— —	
	VER RES	PROG ONE	
	PROG SEL	PROG TWO	
key tabs	FIELD WORD BKSP	ERROR RESET	key tabs
	BLANK COLUMNS LEFT ZERO CTRL	LEFT ZERO	
	CHAR BKSP	AUX DUP	

There are keys which have functions directly related with switches on the console. These keys will be described under "Console". Other keys have already been discussed, as, for instance, the LZ key.

VER DUP

This functional key has the same function as the DUP in the 029. That is, in the punch mode it will duplicate common information from card to card. The duplication rate is 10 *columns* per second when this key is held depressed (manual operation) at the 0 program level. Under program control (1-6) duplication takes place at the rate of 10 *fields* per second.

Since the CDR functions also as a verifier, in the Verify Mode this key is functional only in the high order position of a L-Z programmed field (a 3-punch in the first column).

-LZ

This key has a double function. In alphabetic shift it produces an 11-punch but in numeric shift it produces a minus left zero. This is a new concept which was not available in the 029. The 029 had the left-zero application alone. That is, in a programmed L-Z field a number entered could be preceded by zeros before the first significant digit. Assume that the operator keyed 1234 in an 8-column L-Z field. Depression of the LEFT ZERO tab key in the 029 punched 00001234. Entering "Credit", that is, a negative number, was also possible in L-Z fields in the 029.

In the CDR, entering credit in LZ fields (of any length) is an extremely easy operation. Assuming we have to enter -1234 in an 8-column LZ field, the operator, after keying the digits will depress the -LZ key. The number punched will be -1234, printed as 0000123$\overline{4}$.

SKIP
This functional key has the same functions as in the 029. If there is no field definition or if punching is carried out at program level 0, depressing this key spaces one single column.

REG
This functional key registers (aligns) cards, as in the 029. Individual cards can be inserted in the card bed and registered by depressing this key once.

MULT PCH
This key, when depressed, puts the keyboard in numeric shift, like in the 029. In this case, the column indicator does not advance, which permits entering more than one punch per column. MULT PCH is also used for Credit Punch.

REL
This functional key has the same functions as in the 029. If there is data to be duplicated in fields following the column where REL is depressed (with AUTO SKIP/DUP ON), the card being released will be punched with the data taken from output storage. This makes the difference between the REL key and the CLEAR switch on the console. If AUTO SKIP/DUP is OFF, on keying REL, any data in storage is erased and replaced with blanks, beyond the point of release. The position of the switch REC ADV/CARD FEED will determine if the column indicator will stop at 01 and a new card will be fed (switch set at AUTO), or the column indicator will move to 00 only (switch set at MANUAL).

During verify operations REL will release the card without the 2-3 OK verify punch in column 81. The data in storage will be saved depending on the setting of the AUTO SKIP/DUP as described above.

The Card Release Pushbutton, located underneath the center position of the card bed, permits the removal of any card at any station. The CLEAR switch on the Console removes all cards from the card bed with no cards being fed from the hopper.

FEED
This functional key has similar functions as in the 029. But since the CDR can be operated as a puncher or a verifier, the FEED function is slightly different in each mode. In the PUNCH mode, keeping FEED depressed will feed two cards from the hopper. The first one will be registered at the punch/read station and the second card will remain pre-registered. In the VERIFY mode, keying FEED will supply one single card. No card will therefore be present in the pre-registered position. This allows for the insertion of a blank for an immediate correction of erred cards.

The *ALPHA* and *NUMERIC* functional keys have the same functions as in the 029. NUMERIC has a higher hierarchy than ALPHA and overrides it. If both (ALPHA and NUMERIC) keys are held depressed, the keyboard is still in numeric shift. This NUMERIC key has to be kept depressed when entering numeric data under the 0 program level.

CONSOLE

There are 6 switches on the console, 5 of which are of the ON/OFF type. There is also a rectangular, two-place, digitalized column indicator window, a lighted "Verify" window, a dial for program selection and two pushbuttons.

The *PUNCH/VERIFY* switch has two positions, according to the mode in which the machine will be used, either as a cardpunch or as a verifier. The selection setting should be carried out only when the card bed is clear and the column indicator shows 01. Switching should not be effected in the middle of a card.

REC ADV/CARD FEED
This switch has two positions: AUTO and MANUAL. In AUTO, after a card is keyed at column 80, a new one is fed, which will be under the program level indicated in the PROGRAM MODE dial. The column indicator will then move to show 01.

If, however, keying column 80 is done with this switch set at MANUAL, no card will be fed and the column indicator will move to 00, instead. This setting is particularly advantageous if a change in program level is desired. In this case, after column 80 has been keyed with the switch set at MANUAL, the operator should key PROG SEL followed by the key reflecting the program desired (1 to 6), to advance the column indicator from 00 to 01.

At this time, switching from MANUAL to AUTO will produce the punchout at the read/punch station, a FEED cycle and the column indicator will move to 01. Data keyed in error can also be corrected in storage when the column indicator shows 00.

The *Column Indicator* has the regular column numbers from 01 to 80 but, in addition, it has a 00, and 88 and a CC indicator. Regularly the column indicator will show the next storage position into which data will be entered. This column indicator shows CC when a card is found to be in error upon verification, and a new, corrected card (CC) should be punched out. (Although the machine is at the verify mode it still can punchout and print in these verify-correct cases.) Number 88 appears at the column indicator when a field or card must be reverified in verify mode. Simultaneously, there is keyboard locking.

AUTO SKIP/DUP, ON/OFF
This switch has the same functions as in the 029. In fields programmed for automatic skipping or duplication, these functions will be performed at electronic speed if this switch is ON. If it is OFF, these functions are inoperative.

PROGRAM MODE (Circular knob on console)
READ (pushbutton on console)
PROG SEL (key on keyboard)

The descriptions of these functions have been grouped together since they are interrelated.

The PROGRAM MODE dial on the console has 7 numbers (0-6) and 3

labelled positions (Data Read, Prog Pch and Stat Pch). The positions numbered 1 through 6 relate to the program levels which can be loaded into the different memory levels or used in keying jobs. Level 0 can not be loaded or programmed and it is only for an 80 single-column, alphabetic field. The machine, which normally is in numeric shift, can be placed in alphabetic shift by dialing PROGRAM MODE to 0.

The procedure to load programs into the different memory levels involves the use of the PROGRAM MODE dial and the READ pushbutton. For the procedure to be described below, we will suppose that the program cards have already been punched and are available for storage. To load these programs into memory these steps are followed:

1 remove all cards from the card bed with CLEAR;

2 set: a) AUTO SKIP/DUP switch to OFF,
b) PROGRAM MODE to the program level into which each of the cards will be loaded;

3 insert the program card at the punch/read station or feed it from the hopper. In either case, this card *MUST NOT* be registered;

4 press READ.

Steps 2, 3 and 4 will be repeated for each different program level.

The assignment of a program level is arbitrarily made by the operator. There are no rules as to what program level each punched program is entered into and it is therefore left to the convenience of the operator.

Loading the program memory can be repeated as many times as necessary. Any new program stored at any level, "erases" the previous program existing at the same level. Assume that the operator is loading a new program at level 2, where another program was already in memory. The new program will "erase" the old one. Another way to blank out the different program levels (1-6) is by reading blank cards at the different levels (instead of punched out program cards). This results in 6 blank levels and the machine is shifted into 80 single-columns fields, all of them in Numeric shift. Blanking out the program memory also results from flipping OFF/ON, the mainline switch below the desk. In this case the CDR is automatically set at level 0, regardless of the PROGRAM MODE setting.

Normally, program switching should be performed at the first position of a field. There is the risk in switching programs in the middle of a field that the machine may combine the original program with the newly selected one, producing incorrect results. Manual changes, therefore, should be effected in the first position of a field before any data has been entered in it.

Switching program levels can be effected by the use of a procedure in which the PROG SEL key is involved. Since there were only 2 possible program levels in the 029, and the CDR has a capacity of 6, the procedure enlarges enormously the capacity of the machine, and while the procedure is a little more elaborate than in the 029, it is still simple.

Program level can be changed any number of times while punching a card (with the REC ADV/CARD FEED switch set to AUTO) by keying PROG SEL followed by a digit from 1 through 6. Regardless of the program setting indicated at the PROGRAM MODE dial, the new program level is effective immediately, overriding the program level previously selected (*via* PROG SEL) or dialed (PROGRAM MODE). The new program level remains in force

until the machine advances to column 01 of the next card. At this time, it reverts back to the level indicated by the PROGRAM MODE dial.

Another way to change program levels is by setting REC ADV/CARD FEED to MANUAL. On reaching column 80 of the card being keyed, the column indicator will show 00. At this point, a new program level may be selected by keying PROG SEL followed by a digit, and then setting REC ADV/CARD FEED to AUTO. A FEED cycle will occur and the new card will be under the new program level selected.

The operator should be aware that the depression of a valid digit (1 to 6) for a program level, MUST follow immediately after pressing PROG SEL. Otherwise the keyboard locks. Should this happen, the keyboard is unlocked by depressing CHAR BKSP. This automatically reverts the situation and the operator must again select the new program level by depressing PROG SEL followed by a valid digit. In other cases, if after depressing PROG SEL a change of program level is not desired, press either the digit corresponding to program level in use, or FIELD BKSP to restore the program-select condition.

The function of *Stat Pch* on the PROGRAM MODE dial, is related to the punching out of cards for production statistics, data for use in measurements of work load, analysis of errors and job accounting.

Data Read is another position of this dial which is used when new data (maybe a new master card) has to be read into output storage for duplication into the following cards. To read a new data card into output storage, the procedure is as follows:

1 remove all cards from the card bed with the spring-loaded CLEAR switch;

2 set the Dial at Data Read;

3 insert the new data card manually at the punch/read station or feed manually from the hopper. In either case the new card *MUST NOT* be registered;

4 press the READ pushbutton. At this point the new card will be registered and read, the data will be transferred into output storage and the machine is automatically set at program level 0. In this situation, the new data in storage can be duplicated or corrected. Or, if a new program level is desired, the change can be effected at this point.

If the new card was registered, the READ cycle would not take place.

Prog Pch is extremely useful to obtain a copy of any program stored in memory, at any level. The punchout, which will produce a copy of the selected program on a blank card, is easily carried out in a procedure very similar to the one described for Data Read:

1 remove all cards from the card bed with CLEAR;

2 set: a) the CDR to PUNCH,
b) the AUTO SKIP/DUP switch to OFF,
c) the REC ADV/CARD FEED to MANUAL,
d) the PROGRAM MODE dial to PROG PCH;

3 insert a blank card manually at the punch/read station or feed from the hopper;

4 register this card with REG and then depress REL (this sets the column indicator at 00);

5 depress, in succession, PROG SEL and a digit, for the program level desired.

The blank card will be punched with the data in program storage and will be sent to the stacker. Regardless of the position of the PRINT switch, all program cards will be printed. Do not select the digit 0 (in step 5) because a blank card will be produced.

PRINT
This functional switch can allow or suppress printing on top of each card. No double printing occurs when cards are being verified because PRINT is non-operational during the VERIFY mode. However, when a Verify Correct Card has to be punched out, during verification, PRINT is operational, if the PRINT switch is ON.

CHARACTER MODE
This switch allows the use of a 48- or a 64-character set. The characters which are not operative in the 48-character mode are those outlined in Figure 17.1 when the CDR is in numeric shift. Pressing these keys locks the keyboard in the 48-character mode.

CLEAR
This spring-loaded switch has the same function as in the 029 and does not duplicate fields programmed for auto skip or auto dup. This effect should be considered in comparison with the effect of REL key.

REC BKSP (pushbutton on Console)
CHAR BKSP (functional key on keyboard)
FIELD; WORD BKSP (functional tab on keyboard)
These 3 functions are interrelated and make the CDR capable of backing a column, a field or a whole card. These alternate possibilities were not available in the 029.

Depressing REC BKSP returns the keyboard all the way back to the first column of a card. Input storage is completely erased, and is ready again to accept new input. The column indicator will show 01. If automatic skipping or duplication was programmed for that column 01, it is temporarily inactivated, but can be immediately reactivated by pressing SKIP or VER DUP.

CHAR BKSP causes a backspace of one column. This is similar to the backspace button on the 029. But in the 029 this pushbutton had to be kept depressed to obtain a continuous backspacing. With the CDR the operator can back a column, a field or a card by a single stroke of the proper key. When the REC ADV/CARD FEED switch is set to AUTO, by depressing CHAR BKSP, all automatic functions (dup, etc.) under program control are bypassed. If REC ADV/CARD FEED is set to MANUAL, CHAR BKSP will backspace to the last column entered manually. If the CDR is being operated

in the Verify Mode, CHAR BKSP resets the error-condition (before rekeying) and allows additional verify attempts. CHAR BKSP is also used to unlock the keyboard.

FIELD; WORD BKSP, when depressed under program control, returns the CDR to the first column of the field being keyed.

There is one particular situation which has to be considered in detail when this functional key is actuated. Assume there are two adjacent fields, the first one normally programmed (that is with any function but auto skip or auto dup) and the second field programmed for auto skip or auto dup. Assume that the operator, after keying the last column of the first of these two fields, senses an error. Since the program calls for an automatic function immediately after the erred information is keyed, these functions will be carried out anyway; nevertheless FIELD; WORD BKSP will jump back two fields when depressed, that is, the second field (the auto skip/dup) will be jumped over and the card will go back to the first column of the first field. When REC ADV/CARD FEED is set to AUTO and a keying error is effected in the last column of the last manual field, the next blank card will be automatically fed. Consequently in this case FIELD; WORD BKSP can not be used since the new card is already at the read/punch station. The column indicator will show 01; but if REC ADV/CARD FEED is set to MANUAL before column 80, the column indicator will show 00 and FIELD; WORD BKSP can still be used.

The use of FIELD; WORD BKSP is especially convenient in alphabetic fields, particularly when names, parts or program words are entered. In these cases, when corrections have to be made, the operator should depress both the ALPHA key and the FIELD; WORD BKSP tab. This will cause the CDR to return to the column following the last manually-keyed space, or to the first column of a manual field, whichever occurs first. The operator should keep in mind at all times that, although an alphabetic data is being keyed in an alphabetic field, he must still depress both keys.

Any backspace operation (character, field, word or card), does not change the program level under which the job is being punched.

How to Prepare Program Cards

When the power switch is turned OFF, any information in memory is erased, including the programs which may have been in storage. Since most of the characters needed to prepare program cards are in numeric shift (except one), it is convenient to set the machine at any 1-6 program level (using the PROGRAM MODE dial), with no program in storage. This sets the machine for a single 80-column field in numeric shift. The only code to be entered in alphabetic shift is the skip code. The 11-punch has to be keyed in alphabetic shift since the same key, in numeric shift, produces a negative LZ. The next section deals with the most common operations carried out with the CDR, in punch and verify modes.

PUNCH MODE

A number of operations can be carried out in punch mode. These are listed

below in the order in which an operator carries out a job. This is particularly true of the first 3 operations:

1 punch a program card;
2 store a program in memory (program load);
3 punch cards;
4 punchout (to obtain a copy of a stored program);
5 read data for a new master card;
6 add a punch;
7 makeover card.

To Punch Program Cards

1 Set the machine in the punch mode;
2 place blank cards in the hopper;
3 set AUTO SKIP/DUP to OFF;
4 CLEAR the bed;
5 set REC ADV/CARD FEED to AUTO and PROGRAM MODE to the digit 1;
6 set PRINT to ON/OFF as desired;
7 select the 48- or 64-CHARACTER MODE as desired;
8 press FEED and start punching the program cards with the program codes.

To Load Programs in Memory

1 CLEAR the bed;
2 set the AUTO SKIP/DUP to OFF;
3 select the PROGRAM MODE to the desired level (1-6);
4 place manually a program card at the punch/read station or FEED it from the hopper. In either case, DO NOT REGISTER;
5 depress the READ pushbutton.

To Punch Cards

1 Load the program in memory;
2 turn the PROGRAM MODE dial to the desired level. Depress PROG SEL and then the digit key corresponding to the program level under which this job will be punched;
3 with AUTO SKIP/DUP set to OFF, proceed to punch the first (master) card manually. After the first card is punched,
4 switch AUTO SKIP/DUP to ON and continue punching.

Punchout

1 Set the machine in the Punch mode;
2 CLEAR the bed, if necessary;
3 switch AUTO SKIP/DUP to OFF;
4 set REC ADV/CARD FEED to Manual;
5 dial PROGRAM MODE to Prog Pch;
6 insert manually a blank card at the punch/read station or feed it from the hopper;

7 key REG to register this card;
8 key REL;
9 key PROG SEL followed by the digit of the program level which the operator wants to duplicate.

Read Data for a New Master Card

1 CLEAR the bed;
2 dial PROGRAM MODE to Data Read;
3 insert the new card, carrying the new data (new master card) at the punch/read station or feed it from the hopper. In either case DO NOT REGISTER the new card;
4 press the READ pushbutton on the console.

The data is transcribed into output storage and the new master card passed on to the stacker. Punching continues by setting the PROGRAM MODE and keying PROG SEL followed by the desired program level (0-6).

Add a Punch

To add a punch to an already punched card:

1 CLEAR the bed;
2 turn the AUTO SKIP/DUP to OFF and REC ADV/CARD FEED to Manual;
3 advance the input to the desired column (use the space bar) and key the characters to be added (PRINT is set to ON or OFF, as desired);
4 press REL;
5 place the punched card at the punch/read station and *REG*ister it;
6 switch REC ADV/CARD FEED to AUTO.

The card will receive the new data from output and then the card will be released into the stacker. The existing punches remain unaffected. To Re-punch, or make a card which is to be corrected in certain columns, proceed as follows:

1 set the PROGRAM MODE to Data Read;
2 turn REC ADV/CARD FEED to AUTO (PRINT to ON or OFF as desired);
3 place the erred card in the read/punch station. Duplicate manually up to the column(s) requiring a makeup. Key the correct data and then duplicate manually the rest of the card;
4 register a blank card at the punch/read station;
5 press the READ pushbutton at the Console.

VERIFY MODE

The general principle under which the CDR functions as a verifier is quite simple. The punched cards to be verified, are "read" at the punch/read station and their data is sent into input storage. Then, as the operator keys the data from the source document, for verification purposes, each key-stroke is entered into output storage and simultaneously compared with the

data in input storage. If the data is identical and no errors are found, the verification operation continues and a 2- and 3-punch is entered into column 81, before each correct card is released.

Verification under program control uses the same programs and the same program codes as those used for punching, except for LZ verification. These LZ fields, for verification purposes ONLY, must have programs specially coded with a 3-hole at the first and last columns of that LZ field and with field definition punches (&s) in all other columns.

Procedure for Verification under Program Control

1 Select a program level (1-6) and store the program in memory at that level. This procedure has been described in PUNCH mode. At this time, the Punch/Verify switch can be at either setting;
2 load the punched cards to be verified in the hopper;
3 CLEAR the bed;
4 set the Punch/Verify switch to Verify;
5 set AUTO SKIP/DUP to OFF and REC ADV/CARD FEED to AUTO (for the first card);
6 set PRINT and CHARACTER MODE (48 or 64 characters) as desired;
7 key FEED (one card only will be fed);
8 dial PROGRAM MODE to the level where the program has been stored. Key PROG SEL followed by the proper digit (1-6) that corresponds to the level where the program is stored;
9 verify the first card (duplication and skipping not active);
10 switch AUTO SKIP/DUP to ON after the first card and continue verifying.

If keying errors are found during verification, the VERIFY light turns ON and the keyboard locks. To unlock the keyboard, depress the VER RES key. This, simultaneously, turns the VERIFY light off. A second verification tryout is attempted. The VERIFY light will turn ON, the keyboard will lock and it will be unlocked with the VER RES key, when the light will turn OFF.

The third verification tryout will overwrite the correct character into output storage, "erasing" the erred character. At this point, the operator backspaces with the CHAR BKSP key to reverify the new character entered into output storage. If CHAR BKSP is not keyed, the keyboard locks at the end of the field and the column indicator will show 88, to indicate that a complete reverification of the total field is mandatory. In this situation, instead of CHAR BKSP, key FIELD; WORD BKSP and *re*verify the entire field.

Assume that only one error has been found and that this error has been corrected (at the third tryout), corrected (rewritten into output storage) and reverified (CHAR BKSP). The card will be released into the stacker (in a pre-stacked position) but without the 2-3 punches in column 81. This card can be immediately removed from the stacker since it is very easily identified in view of the lack of punches in column 81. The column indicator will now show CC, meaning that a *C*orrection *C*ard should be punched out. The corrected data is still in output storage but *no* card has been fed for the hopper. Insert manually a blank card at the punch/read station and depress the VER

CORR key. The blank card is punched out with the correct and verified data from output storage. This corrected card will be released into the stacker with a 2-punch in column 81 (instead of a 2-3 punch). A new card will now be automatically fed, registered and verification will continue.

These corrected cards will be printed only if the PRINT Switch is ON. This switch, however, is generally inoperative in verify operations so that overprinting does not occur.

Procedure to Correct Erred Cards

1 When a column is found in error, unlock the keyboard by depressing VER RES;
2 verify that column for the second time. Keyboard locks. Depress VER RES;
3 rekey the data for the third time;
4 key CHAR BKSP;
5 rekey (reverify) the data for the fourth time;
6 at the end of this card, place a blank card at the punch/read station;
7 depress VER CORR;
8 continue verifying.

If a corrected card is not desired, after verifying the erred card, key FEED or actuate the CLEAR switch. Using CLEAR inhibits the duplication into the next record; FEED permits it.

Verification can also be interrupted with REL. REL is used when there are numerous errors in the card or the correction has been improperly made. The use of REL allows the duplication or skipping of data fields which have been so programmed.

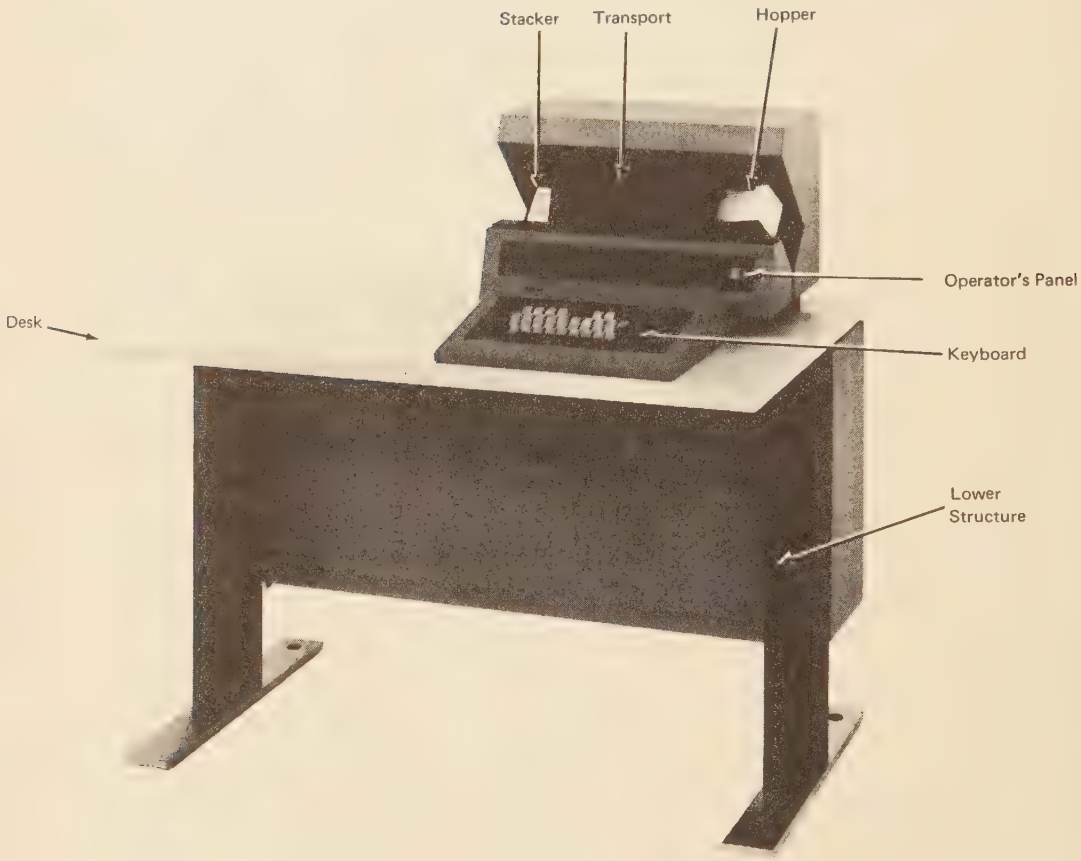


Figure 18.1 The IBM 5496 Data Recorder Components

18 The IBM 5496 Data Recorder

This machine, like the 129, is a self-contained unit which permits both punching and verifying, has an electronic memory which permits the correction of data while they are in storage and a system which stores programs in memory with a capacity of up to 4 programs. Physically the 5496 resembles the 024, 026, 029 and the 129. There are, however, a few structural differences. A hinged lid covers a number of visible, essential parts, such as the hopper, the stacker and the card transfer. This lid makes this cardpunch quieter than other models. Below the lid, on the transfer, there are four stations: the punch, read, verify wait and print stations. The main switch and paper chip box are, in this model, on the panel and not below the desk.

The cards used by the 5496 are of a different size (smaller) and shape than the regular, rectangular cards used by the previously described punches. Nevertheless, these cards can hold 20% more information than the other ones (96 instead of 80 columns), although they are one third their size.

The 5496 is used mainly in conjunction with the System 3 computer. System 3 is a small installation which does not have the capabilities of IBM 370. Nevertheless it has become a boon to small business since it can carry out the most common accounting operations (payroll, billing, sales, etc.).

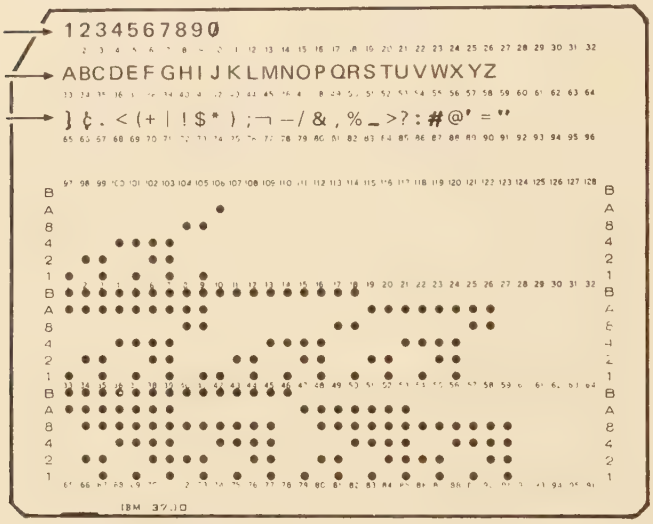
The Data Recorder has three memory areas: the output, the input and the program storage. Like the 129, keyed data enters into storage, to be actually punched only when the last column of each record passes through the column indicator. At this point the actual punchout takes place. There is a right-adjust function with left-filled blanks but there is no left-adjust, like the one present in the 129.

THE CARD

There are four differences between the cards used in the 5496 and those used by the 029 and 129:

a number of columns and positions within the column;
b size;
c there are 4 printing lines;
d the data punches are round instead of rectangular.

The 5496 punches data into 96-column cards of a different size and shape than the previous cards. These cards are 3 5/16" by 2 31/32". These cards, with upper left or upper right corner cuts, have 96 punching positions and 128 printing positions. The printing and punching positions are arranged in tiers of 32 columns each. Therefore, the groups occupy positions 1-32, 33-64, 65-96 and 97-128. The first 3 groups of printing and punching tiers, correspond to one another. That is, data printed in the first tier corresponds to data punched on the first tier and so forth. Each column has six punching positions named B, A, 8, 4, 2 and 1, respectively, from the top of the tier to the bottom. Each tier of the punch area is more than twice as wide as the corresponding tier of the printing area. The 4th printing area is reserved for other uses.



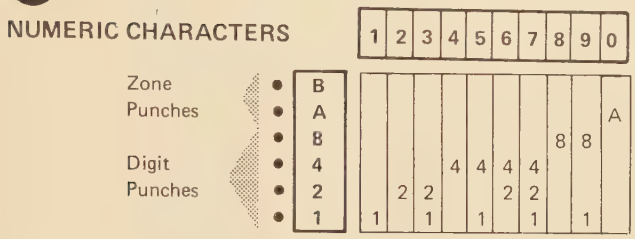
It can be said that the code used by the 5496 for the representation of letters, digits and special characters, is an extension of the binary representation used by the paper tapes employed in some computer systems. Basically, it is an application of the law of probabilities. In order to give an example, just assume that you have two numbers: 1 and 2. These can be combined in pairs to have 4 possible numbers 11, 12, 21 and 22. Mathematically this is expressed as the exponential function $2^2 = 4$, where the exponent is the number of digits involved. This can also express the number of combinations that can be obtained with two holes in six positions. The general mathematical expression $2^6 = 64$ is then obtained, and 64 is the total number or characters that can be punched in the 5496. There are, however, character combinations which require three holes instead of two. Then the mathematical expression is different. Suffice to say that the number of combinations used by the 5496 are generously covered by the total number of possible combinations available.

The representation of digits is clear. If we assume that each position has actually the values of 1, 2, 4 and 8, the representation of the digits 1 through 9 is simply a combination of the proper punches. The digit 0 produces a punch in zone A. This does not mean that in order to obtain the character 0 (zero) printed, you have to key the letter A: this is a concept that the reader should keep in mind. In order to punch 0, you key 0, regardless of the holes and where they are produced.

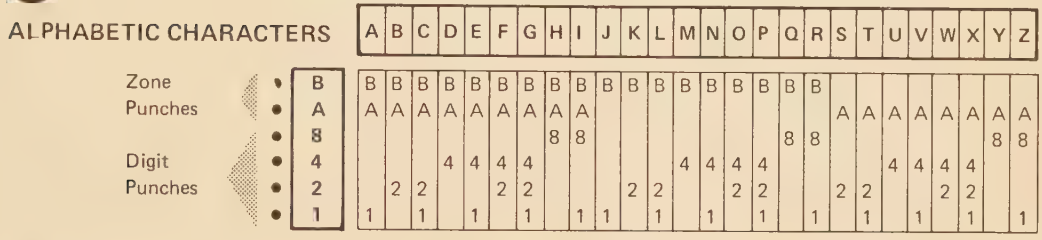
The following table gives the holes punched for the different characters used by the 5496.

Table 18.1

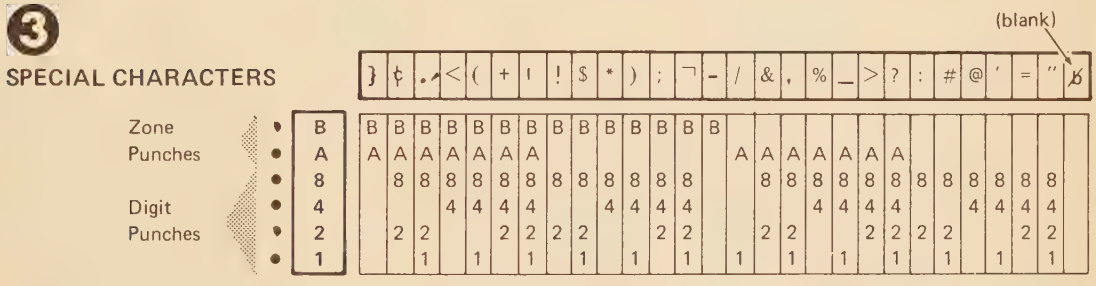
1 NUMERIC CHARACTERS



2 ALPHABETIC CHARACTERS



3 SPECIAL CHARACTERS



The process through which the cards are punched, is very similar to that of the 024/026, 029 and 129. Blank cards are placed in the hopper (with a capacity of 350). The card passes from the hopper, *via* the transport, into the stacker. There are 4 stations in the transport: the punch, read, verify wait and print stations. Punching does not occur simultaneously with keying, just like in the 129: the keyed data for each card is sent to memory where it is stored. Punchout takes place at the end, all at once, whereupon all the stored data is punched out. At this point all the data stored in electronic memory is discharged from the output memory. It is for this reason that any data keyed in error can be corrected in memory before the actual punchout takes place.

There are two data memories, like the 129, one input and one output memory (besides the program memory with 4 levels). The input memory is where the data is actually keyed in. At the end of a record, the data is discharged into the output memory and simultaneously punchout takes place.

The data stored in the output memory can be used when it has to be duplicated.

PROGRAM CARDS

A card may contain one field of 96 columns or 96 single-column fields or any of the many possible combinations in-between: there are no limitations to the length (up to 96 columns) of a field in the 5496 card.

The codes used to program the 5496 differ radically from those used in the 029 and the 129. For instance, the program codes for the 029, used the first column of each field, "the high priority punch", to indicate the type of field; the rest of the field was filled with field definition punches. The 5496 is exactly the opposite: all the columns of each field carry the so called high priority punch and only the last column of each field (and only this column) also carries the "end-of-field" code, in addition to the field code. That is, the last column of every field carries two punches. The following table shows the different codes used by the 5496.

Table 18.2 Codes used for program cards

Function	Use key	Punch at	Remarks
End of field	-(hyphen)	zone B	Last column on each field
Automatic skipping	0(zero)	zone A	Every column, including the last one
Automatic duplication	8	8	Every column, including the last one
Lower shift	4	4	Every column, including the last one
Upper shift	Space	blank	Every column, including the last one
Numeric shift*	2	2	Every column, including the last one

*Only the 0-through-9 digits and the blank can be entered in these fields.

The codes punched in positions 2, 4 and 8 will be produced by keying the respective digits (2, 4 and 8). However, the program codes for positions A and B, will *NOT* be produced by keying the letters A and B. As a matter of fact, the "A code" and the "B code" will be produced by keying zero and the hyphen, respectively. Self check number codes are coded at the 1 level.

Since the last column of every field carries a double punch, the function code and the "end of field" code, these combinations are, in fact, the representation of other characters as per our Table 18.1. Consequently for these last columns in program cards, a convenient short cut can be used as per the following table.

Table 18.3

Codes for the last column of a field coded for	Punched at	Equivalent to
Automatic skipping	A and B	} bracket
Automatic duplication	8 and B	Q
Lower shift	4 and B	M
Numeric shift	2 and B	K

There is still another approach to multi-punch these last columns: using the MLT PCH key, the respective punches can be entered. Any alphabetic or numeric field can be used as a regular left-zero or left blank field. Here they are called Right-Adjust fields for which no coding in the program card is necessary. The decision to make of any entry a right-adjust entry rests solely with the operator. In order to move all the keyed data to the right-most column of that field, the operator simply depresses the Right-Adjust key tab after keying the data, and the field will be entered with blanks to the left, up to the first non-blank character. The only precaution the operator has to take is in the verification stage: a "1" code must be punched in those program cards for fields which will be verified as right-adjusted.

There are three shifts in which data can be entered in cards using the 5496: the lower, upper and numeric shift. The lower shift corresponds to the usual alphabetic shift in which the characters shown in the lower parts of the punching keys can be keyed; the upper shift enters the characters shown in the upper parts of the punching keys. In the numeric shift only the 0 to 9 digits can be entered, as well as the blank (space bar). This last shift is very convenient when punching numeric data. Any attempt to enter any other character in these numeric shift fields, causes a keyboard lockup (which is unlocked by pressing the ERROR RESET tab key to the left).

When recording under program control, the keyboard is normally in upper shift mode. In this shift all upper shift characters (including the digits) can be entered. Alphabetic or lower shift characters can be keyed in fields programmed for upper shift characters simply by depressing the LOWER SHIFT key while entering the character. This was also done in the 029 and 129. The same procedure, reversed, is used to enter upper-shift characters into otherwise alphabetic-coded fields.

Fields coded for automatic skipping and automatic duplication present the same characteristics we have described previously for the same fields in the 029 and 129. From the codes punched in the program card, which the 5496 scans, the machine determines the functions to be performed and the columns where they will be carried out. This scanning is carried out three columns at a time. The first time, the recorder scans columns 1, 33 and 65. We call these 3 columns, a column group. Each column group (1, 33 and 65; 2, 34 and 66, etc.) are scanned and punched at one time. Printing also takes place in column groups.

How to Prepare a Program Card

The power switch, in the 5496, is located on the control panel. When it is

turned on, any information in the three memory levels is erased and the machine is placed in manual control, and, as such, in the lower shift mode. We are assuming that, except for the power switch in the panel, all the other switches are off. Since all the characters to be keyed for the preparation of program cards are in the upper shift mode, it is very convenient to place the machine in this shift. This can easily be performed by feeding a blank card into the program level 1, which puts the recorded under the upper shift control. This is carried out as follows:

1 place blank cards in the hopper, face up (the print area should be facing the operator);
2 turn the PROG switch ON;
3 press PROG 1;
4 check that the column indicator shows 01. Otherwise flip the RECORD ERASE switch once;
5 flip the PROGRAM LOAD switch once. This will cause the blank card to be fed as the program card for level 1. Its blank codes will place the 5496 in the upper shift mode.

With the recorder in the upper shift mode it is very easy to prepare program cards.

How to Load Program Cards into Memory

As we have mentioned before, the 5496 has a capacity of four memory levels. That is to say that up to 4 programs can be stored although you can only use one program at a time. Assume that you have already punched 4 program cards and you want to store them in memory, ready for your next job. The procedure to load these programs in memory is easy, and similar to the one described above:

1 place the program cards in the hopper, face up. The order in which these programs will be fed at the different level is decided by the operator, depending on the frequency with which they are used;
2 the AUTO SK/DUP switch must be OFF;
3 turn the PROG switch ON;
4 make sure that the column indicator shows 01;
5 press the desired program level key. PROG 1 if the first program is to be stored at level 1, and so forth;
6 flip the PROG LOAD switch once. This will feed the program card and have its codes stored in memory;
7 repeat the procedure for the other desired programs.

Feed only one program at a time, going through these steps. If the operator is loading more than one program into memory, these steps will have to be repeated as many times as there are programs. These codifications will remain in memory until another program is loaded at the same level or the main power switch is turned off.

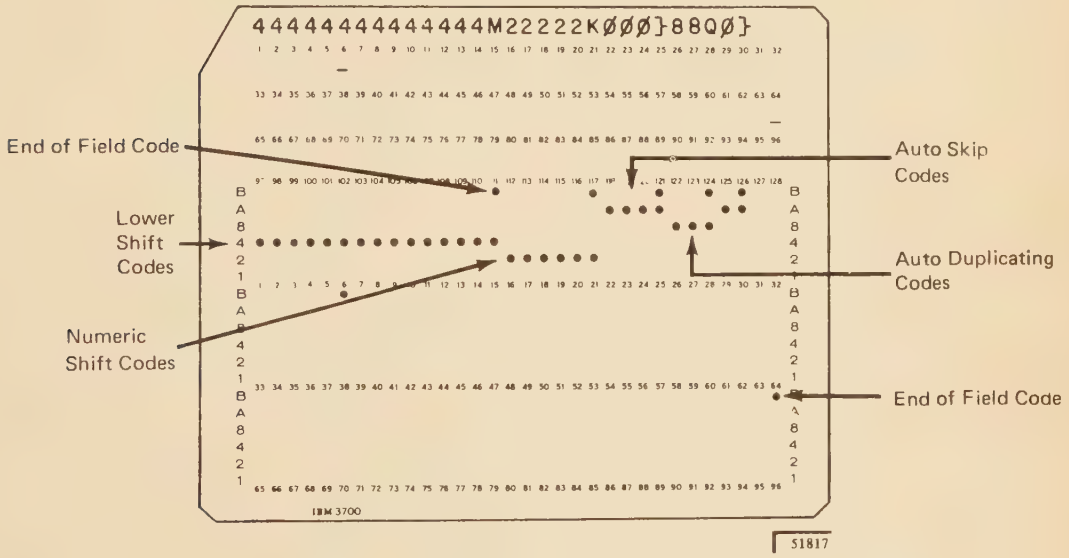
Program level 4 is also used to store information for auxiliary duplication. This added feature is used when seldom used data fields have to be fed at certain intervals. This data can be fed instead of a program, that is, a program and auxiliary data cannot be kept in memory, at level 4, simultaneously. Auxiliary duplication can only be fed at level 4 and at no other level.

How to Key Data Under Program Control

The procedure to key data with the 5496 is no different than the procedures used with the 029 and the 129. A master card has to be keyed manually first and the following cards will be punched under program control.

We will assume that PROG 1 is in control, and already in memory. The preparation of the master card is carried out with the PROG switch on. The AUTO REC REL and PRINT switches should also be ON. Blank cards are placed in the hopper and with the column indicator set at 01, the operator begins to record the information from the source document into the input memory. The AUTO SK/DUP switch, at this stage is OFF, since there is no data to be duplicated or fields to be skipped. This switch, however, should be turned ON when the last field is about to be punched in the master card. After the master card is finished, turn ON the AUTO SK/DUP switch and continue keying as usual.

At the end of the card the column indicator will have advanced through 96 and 00, and now shows 01. At this moment, the information in input memory is discharged into output memory and the actual punchout takes place. If the AUTO REC REL had been OFF, the column indicator would have stopped at 00 and the REL would have to be pressed in order to get the punchout. A copy of an actual program used in the 5496 is shown below.



L–Z or Right Adjust Fields

Any field of data can be made into a Right Adjust field simply by depressing the Right Adjust tab key, after the data has been keyed. There is no need to code the program card to this effect. All the data will be moved to the right so that the last character keyed falls in the last column of that field, while the data keyed will be preceded by blanks.

To Change Program Levels

The AUTO REC REL switch in the panel serves the function of auto-

matically feeding blank cards. Under program control and with this switch ON, when the REL key is depressed this will cause the column indicator to advance to 96, to 00 to 01 (and punchout), when a new card is fed from the hopper. If, however, this switch is OFF, pressing REL will cause the column indicator to advance to 96 and to 00. At this point, another depression of REL will cause punchout and feeding of a new card. Changing program levels, therefore, can be effected while this AUTO REC REL switch is OFF. If the program level has to be changed at the end of a card, turn this switch OFF, press REL once, and then key the PROG level desired. This will produce the punchout, advance a blank card from the hopper and put the machine under control of the new program level. Changes of program levels are to be carried out at the end of a card, when the column indicator shows 00 and not at 01.

This change can also be effected while keying a field. However the net effect in this case will differ if the operator is at the beginning or at any other position within the field. If we assume that the field 1-10 is being keyed under program level 1, and PROG 2 is depressed, this program level is effective immediately if the PROG 2 key was depressed at column 1; but if the PROG key was depressed anywhere between columns 2 to 10, the change will become effective at the beginning of the next field.

Program card codes can be easily erased from memory. If there is one particular program at level 1 and the operator loads another program card at the same level, the new codes will erase the existing ones and the new program codes will now be stored at level 1. This applies to the four program levels. A blank card fed into any memory level is treated as one upper shift command, although it may not contain the "end-of-field" punch in column 96. The 5496 will punch exclusively in the upper shift mode in this case.

Credit

There are some numeric fields which carry a hyphen over the last digit, indicating that this information is a credit. When the MULT PCH key is pressed, the recorder is in upper shift mode, (regardless of program card codes) and the column indicator will not advance while this key is kept depressed. This procedure is used to punch a credit. However the 5496 can not distinguish a negative number (say $\bar{1}$) from the alphabetic representation that a double punch entails. Consequently, when keying a credit, instead of a negative number (which involves keeping a key depressed) a letter may be punched. The 1-through-9 negative digits combinations can be represented by letters. Since the negative hyphen on top of the number is a punch in position B, the following combinations can be established:

Table 18.4

Negative number	Punches at	Equivalent to letter
$\bar{1}$	1 and B	J
$\bar{2}$	2 and B	K
$\bar{3}$	1, 2 and B	L
$\bar{4}$	4 and B	M

(cont.)

$\bar{5}$	1, 4 and B	N
$\bar{6}$	2, 4 and B	O
$\bar{7}$	1, 2, 4 and B	P
$\bar{8}$	8 and B	Q
$\bar{9}$	1, 8 and B	R
$\bar{0}$	A and B	} (bracket)

To Erase Keyed Data

The Field Erase key (FLD ERASE) is extremely useful to erase data keyed in error from memory. When this key is pressed under manual control (PROG switch OFF), only one column at a time will be backspaced and erased. However, under program control, keying FLD ERASE will erase all the characters keyed in the whole field and the machine will go back (erasing) to the first column of the field where it was depressed. Assuming the operator has a job in which the field 1-10 is to be manually entered; 11-15 is to be skipped and 15-30 automatically duplicated. Then, while keying the next field, the operator realizes that information keyed in the first field (1-10) is wrong. By depressing FLD ERASE, the column indicator will backspace to column 1. Fields to be skipped or duplicated automatically are not erased from memory, because FLD ERASE does not erase output, only the input memory. Therefore, after keying the right information on columns 1-10, the column indicator will automatically go back to column 31.

The simultaneous depression of FLD ERASE and LOWER SHIFT keys will result in a word erase function. This is useful to erase any miskeyed alphabetic information.

A more complete "erase" function can be obtained with the RECORD ERASE switch in the panel. By flipping this switch, all the characters already keyed will be erased and the column indicator will backspace (erasing) to 01. However, if the first field, say columns 1-10, is programmed for automatic duplication, the first field will not be erased. This spring loaded switch is inoperative when the column indicator is set at 01.

Duplication and Auxiliary Duplication

To duplicate information already punched in cards, there are three procedures: one under manual control and two under program control. In all cases, the information to be duplicated has to be fed into the output memory by placing the punched card in the hopper, and after checking that the column indicator is at 01, pressing the READ key. After this, the blank cards into which this information is to be duplicated are placed in the hopper. Under manual control, the operator holds the DUP key depressed (the use of the PRINT switch is optional) and duplication will take place at the rate of 10 columns per second.

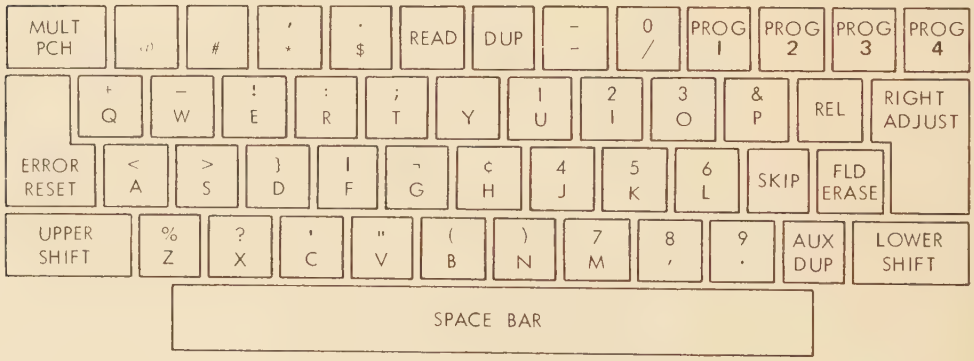
To duplicate under program control, there are two ways: in the first case, a program card coded for automatic duplication in the field (or fields) to be duplicated, is entered in program memory. After this, if the AUTO REC REL switch is ON, the duplication process occurs automatically, without the intervention of the operator. The other approach is to feed a blank program into the program memory. Then duplication is effected by depressing the

DUP key, which will cause duplication at high speed. On the other hand if one wishes to limit the manual duplication to certain fields, "end-of-field" codes must be punched in the program card for each field to be duplicated.

Cards punched in error can be corrected using the same approach: fields keyed correctly are duplicated as indicated above, using the DUP Key. The fields in error are then keyed manually into the new card.

There are times when the normal process of duplicating a field has to be altered because the information to be entered in a card differs from the information being duplicated. It is in these cases when it is convenient to have the seldom used information stored in memory. For these cases, the use of the AUX DUP key is very convenient. Assuming the information 123 is being duplicated in columns 30-32 of a job. Then, in a random fashion, the information 456 is to be entered instead of 123. In these cases, it is convenient to store the new information (456) in memory at the program level 4. To effect this, a new card should be punched with the new information in the specified columns (30-32) and this information should be loaded in the program level 4 using the same procedure used to load a program. Then, when this information is needed, before reaching the first column of that field, the AUTO SK/DUP switch is turned OFF, the PROG switch is turned ON and AUX DUP is pressed. After this, return the AUTO SK/DUP switch back ON. AUX DUP can only be used under program control and the only level at which this information can be stored is at level 4.

THE KEYBOARD



The 5496 keyboard is very similar to that of the 029 and 129. The total number of keys is the same. The punching keys are identical and, with a few variations, are located at the same location as before. Some functional keys have already been discussed. We will therefore review these functional keys in a rapid fashion.

Error Reset (ERROR RESET) Key: The use of this key permits the unlocking of the keyboard whenever certain error conditions occur.

Upper Shift (UPPER SHIFT) Key: Permits entering any upper shift character, regardless of the coded program.

Multiple Punch (MULT PCH) Key: Permits entering more than one upper shift character per column. When this key is depressed the keyboard is automatically placed in upper shift mode and prevents the column indicator from advancing. This key can be used for entering "Credit" information.

Read (READ) Key: This key allows the transfer of punched information from a card into the recorder output memory. Pressing this key will cause a card to be fed from the hopper into the read station of the card bed while, at the same time, its information is read and stored. This key is very frequently used when new master cards are used for a job already in progress. This key is operational only when the column indicator displays 01.

Duplication (DUP) key: Causes the duplication of information already available in output memory into input memory so that information can be duplicated from card to card. Duplication takes place at the rate of 10 columns per second when this key is held depressed.

Program keys (PROG 1, PROG 2, PROG 3, and PROG 4): These keys are used to place the 5496 under control of one program level and to feed program card codes into memory. These functions are dependent on the position of the PROG switch, which must be ON to be operational. Program level 4 is also used to store auxiliary duplication information.

Release (REL) key: When a card has been keyed and the information keyed takes less than the 96 columns of each card, these can be released by depressing REL. The column indicator will advance to 00 or to 01, depending on the position of the AUTO REC REL switch. If this switch is ON the column indicator will advance through columns 96 to 00 to 01, where punchout will occur and a new card will be fed from the hopper. If the switch is OFF, the column indicator will advance to 00 and stop. In this case press the REL key for a second time to advance the column indicator to 01. Punchout and feed will then occur.

Right Adjust (RIGHT ADJUST) key: This key is operational under program control only. It causes a shift of all the characters (numeric, alphabetic, special, blanks and negative numbers) to the right-most column of the assigned field. Columns at the left are filled with blanks.

Field Erase (FLD ERASE) key: Used to correct keying errors. Under manual control, it backspaces and erases at the rate of one column per depression. Under program control it backspaces, erasing, to the first column of the field being keyed. In both cases, the column indicator also backspaces.

Lower Shift (LOWER SHIFT) key: Used to enter alphabetic information and a few special characters. If this key is depressed simultaneously with the FLD ERASE key, a word erase function results.

Skip (SKIP) key: Causes skipping to occur over columns. Under manual control, this skipping occurs at the rate of about 10 columns per second while the key is held depressed. Under program control, depressing this key

permits the skipping of columns not to be used in a field, up to the first column of the next field.

Auxiliary Duplication (AUX DUP) key: This key permits the use of information previously stored at the program level 4.

THE PANEL CONSOLE

The 5496 panel console is somewhat different from others previously discussed. Besides the ON-OFF switches, some of which are spring-loaded, there is a lighted window and column indicator. The chip box and the power switch are also in this panel. This is a departure from previous models. A number of the switches have interrelated functions and for that reason they will be discussed together. Of these interrelated switches some have a higher hierarchy that others.

The Power Switch
This is located on the right of the console and it is an ON-OFF type of switch. When the power is turned ON all the data in storage, program codes or actual data, is erased from memory. Therefore, at the beginning of a job, when the power is switched ON all the memories are automatically cleared. If, at this time, the program control (PROG) switch is OFF, the recorder operates under manual control, on lower shift mode. On the other hand, if the PROG switch is ON, the 5496 is under control of program level 1. Since there is no program recorded on level 1, the 5496 operates in the upper shift mode. The program level indicator will have the 1 lighted.

The Program Level Indicator
This automatically lights the number indicative of the program level in control. There are four program level indicators, one per program. Immediately below this lighted window there are three indicators which become visible when one of the conditions indicated therein occurs.

The ERROR Light
This light comes on when there is a keyboard lockup or when there is a verification error. When the capacity of the stacker (350 cards) is filled, the "Stacker Full" (STKR FL) light is on and there is a momentary keyboard lockup. There is also keyboard lockup when there are no cards to be fed from the hopper. The feed check (FD CHK) indicator lights up. FD CHK also comes on when there is a card misfeed or a card jams.

The Column Indicator
The lights come on when the power switch is turned on. Besides the numbers

01 to 96, indicative of the card's columns, there also is 00. This 00 indicates that all the information has been keyed into the card, up to and including column 96, but the information, in input storage, has not been discharged and no punchout has taken place. (Punchout takes place when the column indicator advances from 00 to 01.) In punching mode, the number displayed by the Column Indicator indicates the next column where the data is going to be entered. In verify mode, the number shown indicates the column which is being currently verified.

The PRINT Switch
This switch allows or suppresses printing. Printing, on the card, takes place in groups of 3 columns: 1, 33 and 65; 2, 34 and 66 and so on.

During verify operations, PRINT is inoperative (except in one case) although it may be ON. During VERIFY REPUNCH, printing will occur on the new card being repunched.

The VERIFY/PUNCH Switch
The position of this switch indicates in which mode the 5496 will operate.

The Program Control (PROG) and Program Load (PROG LOAD) switches
Both these switches have interrelated functions. PROG is used to place the machine under program control. PROG LOAD is used to load program codes into memory but is inoperative if PROG is OFF. That is to say that in order to load programs both switches must be ON.

When PROG is OFF the recorder is under manual control and, as such, in lower shift mode. When PROG is switched ON, the 5496 is under control or program level 1, whatever program may be stored there. When the power switch is turned OFF and ON, this clears all memories. If the PROG switch is turned ON at this point, it will place the machine under program level 1, which has no program in memory. In these circumstances, the 5496 operates in the upper shift mode.

To load programs in memory, using PROG LOAD, the PROG switch must be ON. Then the program (or programs) are placed in the hopper. The program level key desired is then depressed (1 through 4) followed by a flip of PROG LOAD. This will send the codes into memory.

The Automatic Skipping and Duplicating (AUTO SK/DUP) Switch
This switch allows the skipping and duplication of fields which have been so coded in the program card. The PROG switch must also be ON for the codes to be recognized. If the PROG switch is OFF and AUTO SK/DUP is ON, the program codes for automatic skipping and duplication are not recognized. Right adjust verification is also suppressed if PROG is OFF.

The Automatic Record Release (AUTO REC REL) Switch
This allows automatic card feeding and advances the column indicator to 01. In order to be operational, the PROG switch must also be ON. Under manual control, with PROG OFF and AUTO REC REL OFF, the column indicator advances from 96 to 00 when the card has been keyed. To have the column indicator advance to 01 and obtain a punchout the REL key must be depressed.

With PROG ON and AUTO REC REL OFF, the column indicator

advances to 00. The REL key or PROG 1, 2, 3 or 4, must be depressed to advance to 01.

With both PROG and AUTO REC REL ON, the column indicator automatically advances to 01.

The Record Erase (RECORD ERASE) Switch

The function of this switch is to erase from memory all the recorded data, backspacing the column indicator to 01. This switch is operational regardless of the position of the PROG switch, that is to say, recorded data can be erased under program or manual control. Since punchout takes place when the column indicator advances from 00 to 01, record erasing can be carried out before the card is actually punched out, allowing the operator the possibility of correcting any miskeyed (or misverified) data.

This spring-loaded switch not only erases recorded data but also erases change-of-program-level commands. Assume that data is being keyed at level 1, and before the end of the card being keyed, the program level is changed to 2, by depressing PROG 2. If, before the column indicator reaches 01, the RECORD ERASE switch is operated this will not only erase all the keyed data but the change of program level from 1 to 2 is cancelled.

The Verify Field Correct (VER FLD CORRECT) and Verify Repunch (VERIFY RPCH) Switches

These two switches are used when the 5496 is in the verify mode. The PROG switch must obviously be ON since the verification will proceed field by field. If the PROG switch is OFF, the 5496 backspaces to column 01 when the VER FLD CORRECT is operated.

When the spring-loaded VER FLD CORRECT is used, the column indicator, under program control, backspaces to the first column of the field being verified erasing the data keyed in. At the same time, and for the length of this field only, the 5496 changes its mode back to punch. This will permit the operator to rekey that field. At the end of the corrected and rekeyed field, the 5496 operates in the verify mode again. The field then must be reverified for accuracy. The VER FLD CORRECT switch also permits the rekeying of right-adjust fields where too many or too few characters have been keyed in.

If fields have been found in error while verifying, and VER FLD CORRECT has been used to correct data, a blank card, placed in the hopper can be punched with the new data and will be notched to signify that it has been verified correct. To effect this, after the blank card has been placed on top of the remaining cards in the hopper, the operator moves the VERIFY REPCH switch. The old card, without the notch, can therefore be discarded from the stacker. The new, corrected and notched card, will arrive at the stacker and a new card will be automatically fed from the hopper and transported to the verify wait station. The VERIFY REPCH is operational only after column 96 has been verified and the data has been corrected in output memory. The column indicator will remain at 00 and the keyboard will lock until the new card is repunched and column 1 of the next card is recognized.

The PRINT switch, which is normally non-operational during verify operations, becomes operational when the VERIFY REPCH is used. Therefore, the new, corrected and notched card, will be printed.

VERIFICATION WITH THE 5496

The 5496 recorder is also used to verify the accuracy of the data punched with the same machine. To use the 5496 in the verify mode the VERIFY PUNCH switch in the panel must be turned to VERIFY. Verification is mostly carried out under program control.

The program codes used for verification are exactly the same ones used for punching, with one exception. The same program cards can be used to punch and verify the same job except for fields which have been right-adjusted. Only in these cases, provisions must be made in the verify program card to tackle this small problem. The verify code for right-adjusted fields is the digit 1 punched in the first column only of the right-adjusted field. This code, in the punch mode, is a self-check function. Since there are no specific codes for the right-adjust punch function, the operator should establish, for each job, which fields will be right-adjusted and proceed to enter and verify data in these fields accordingly. However, to avoid confusions, the verify program card for the same job should be carefully prepared so it can be used to verify. A decision must be made, before starting a job, as to which fields will be right-adjusted and then prepare the punch and verify program cards for that job.

The procedure to verify cards which have already been punched is very simple. In the first place, the operator must check if the program card used for the punching job can be used to verify. If not, because of the right-adjusted fields, a new program card for verification must be prepared. Whatever the case, the program card is loaded in memory using the same technique used to load programs in memory for punching. Up to four verify programs can be loaded in memory. Then the operator selects the program level under which that particular job will be verified using the same procedure used when the data was punched and a program level was selected. The PROG and AUTO REC REL switches should be ON.

The punched cards to be verified are placed, face up, in the hopper. The first card is handled like a master card, that is, the AUTO SK/DUP must be OFF because there is yet no data to be duplicated or skipped. Depress the READ key: the first punched card will enter the transport, its data read into memory and the card will stop at the verify wait station. The operator verifies the data. As each column is rekeyed, each character is automatically compared with the data in storage. If this card is found to be correct, it is notched and transported into the stacker. At this point, the AUTO SK/DUP switch can be turned ON. The next card is automatically read and stops for verification at the verify wait station.

If the characters do not agree the keyboard locks and the error light comes on. The keyboard is unlocked and the light turned off by depressing ERROR RESET. The operator is allowed two more tryouts. The character keyed at the last tryout is entered in output storage and the verification can continue.

When the verification is carried out under program control, in every field where a character has been changed, the 5496 returns for reverification to the first column of that field after verifying the last column of the same field. The field will therefore have to be reverified. However, when the verification is carried out under manual control, any corrected character will cause the 5496 to backspace to column 1 and the complete card will have to be reverified. It is then evident that it is more convenient to verify under

program control. The VER FLD CORRECT switch is extremely useful under program control, for correcting a field (or fields) where numerous errors have been found. Setting this spring loaded switch ON, places the 5496 in punch mode during verification. Then the correct data can be rekeyed.

At the end of a field where VER FLD CORRECT has been used, the 5496 automatically returns to column 1 of that field and back to verify mode. This field must then be reverified again. The VERIFY REPCH is then used to obtain a corrected and notched card of a verified correct record.

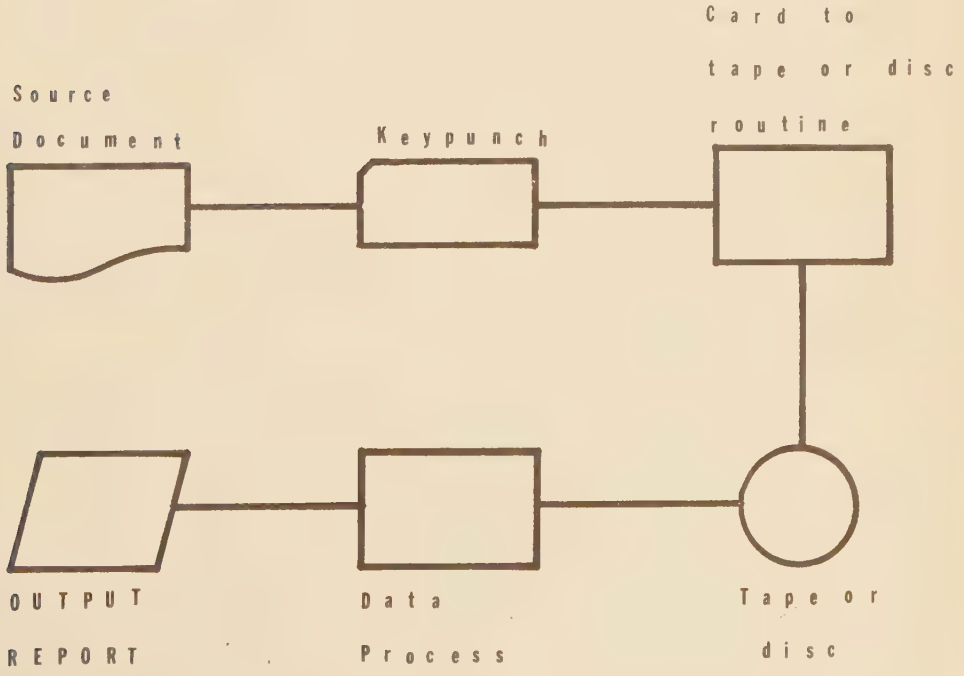
If the last column of a field has not been verified, a field, two fields, a word or a complete record can be corrected by using the proper combination of switches and keys.

19 The keypunch and the computer

by Lester Fox and J. Blum

Once the operation of the keypunch becomes more or less natural to the operator through education and experience, it is important to understand the effect of keypunching on an overall computer operation. By becoming aware of the repercussions arising from mispunched or out of sequence cards, a keypunch operator realizes the interdependency of the various elements of a computer system.

In this section the objective is to establish the position of the keypunch in the total scheme of electronic data processing. To do so let us follow the path of data throughout the system. There are certain clearly defined steps in the process of data. The path of data through a system varies greatly from installation but, in the main, the data proceeds as follows:



To further illustrate the data path let us recall the example wherein the payroll cards were prepared. First there was the source document that contained the required information for keypunching. The column assignments and a short description of the data comprised the instruction sheet specifications. The source documents which the operator uses to transcribe the data into the cards was also shown in that section. We now assume that all the cards have been punched.

The cards are then read on a card reader. One of many devices used for input and output on computers. While these devices are part of a computer system they are not located physically within or attached to the central processor and are therefore known as peripheral devices. Some commonly used peripheral devices are the card reader, card punch, paper tape reader and punch, magnetic tape reader and disc drive. Perhaps the most commonly used peripheral is the printer. These devices are only a few of the large number of devices which are used in conjunction with the computer.

It may be helpful to a new keypunch operator to become familiar with other devices so that there can be a full understanding of the total system. Magnetic tapes for example are extremely common in the field of data processing and bear some further discussion.

As is now obvious, the operator punches coded holes in a card to signify numbers, letters and special characters. In much the same way, magnetic impulses are positioned to indicate these characters. The peripheral devices (tape, card reader) are so designed as to translate the codes from one device to another so that the continuity of data is not interrupted.

Since the card reader is ONE of the slower devices on a computer, data is rather transmitted using some other device. In this case it is tape. This does not imply that the cards may not be used directly for processing. They certainly may. Because of speed, space and security considerations, however, card data is not as efficient as other devices. For example, the IBM card reader (2540) reads approximately 800 cards per minute while the average IBM tape (2400) reads over 60,000 characters per second. In order to equal the tape's transfer rate, a card reader would have to read 45,000 cards per minute. In terms of space the data contained on one reel of tape equals the data contained on a stack of cards 10 stories high. One last point: security. Anyone can read a card. Only the machine can read a tape.

After card to tape or other faster device has been accomplished, the system now processes the data and the final output is produced. It would seem that the keypunch played a very small part in the running of the system. This is misleading because the full effect of the keypunch effort is not apparent. The keypunch operator not only produced the payroll data cards but provided the source of the card to tape program and the reporting program.

If at any point in the system a card was found in error this could have a serious effect on the rest of the data flow. If we remember that the computer can only process the data exactly as it finds it, performing exactly as the programmer has instructed it, then the problems caused by mispunching are obvious.

There are tremendous advantages to the keypunch operator who takes the time to investigate fully the systems that the company employs. In this way the keypunch operator is able to make a greater contribution to the processing function. The operator is able to determine the source of the printed document which is used to prepare card input. A knowledge of

programming syntax provides the operator with the ability to correct the transcription errors on program coding forms.

By far the greatest advantage to the operator is knowing the systems of the company; because of this knowledge, the operator is unharried, efficient and able to produce in a professional manner.

20 Keytape

by Jaime Blum and Lester Fox

INTRODUCTION

The previous chapters have no doubt convinced you of the fact that the preparation of source data is an important and exacting procedure. There is no room in the electronic processing of data for even one "minor" error. Since errors are so costly, a great deal of time and effort is required to insure the correctness of data.

In addition to the problems of editing punched data there are the problems of proper sequencing, space requirements for card decks, easily damaged cards and relative slowness of transmission. Every installation has, at one time or another, been on the verge of strangulation as a result of card storage space requirements. The approximate cost of $1.00 per 1000 cards can be doubled or tripled when storage, handling, record keeping and other functions associated with cards are added. Since the number of card images on a single 2400 foot tape (10 1/2" reel diameter) can equal a stack of card one hundred feet high the space required to store the data is relatively infinitesimal.

One rather new device has been developed which successfully dispels the aforementioned data preparation problems. The device is the keytape. Instead of cards, the data is transcribed on to tape. The benefits of the keytape device are such that they manifest themselves in every facet of the source data preparation process. The list of card problems mentioned at the beginning of this section are resolved immediately.

The data will be on tape now rather than on cards. The correct sequence of records is therefore assured. If a deck of cards was accidentally dropped, resequencing was at best a major effort and was at worst impossible. Furthermore, through the use of keytape sequence, problems, should they arise, can be corrected dynamically.

Insofar as the susceptibility to damage inherent in cards is concerned (e.g. fraying at the edges, folds, bends etc.) these problems do not exist on tapes since there are 25 feet of tape before the first record is written. This leader provides adequate protection from the effects of improper handling. As you know, once a card has been punched it can no longer be used for anything else. Conversely the tape can be rewritten countless times.

Possibly the most important benefit accompanying the use of keytape over keypunch is the speed at which data can be transmitted to the computer. As you read in the chapter describing the keypunch and the

computer you, no doubt, recall the additional card-to-tape step in the system; of course, through the use of keytape, that step is omitted.

What is a keytape? The concept of a keytape is easy to understand since the main components of a keytape have already been mentioned. A keytape system is made up of keyboards, each linked (in many cases) to a visual device like a TV screen where the keyed data is displayed, and a tape carrier where the data is actually entered.

Data on keytape systems can be entered from keystation-to-tape or keystation-to disc-to tape (shared processor data entry system). In systems which use disc, a number of keystations are linked and share a minicomputer that collects, controls, edits and organizes input data into tape files that are then ready for the mainframe computer. This, therefore, eliminates the pooling process required in stand-alone keystation-to-tape entry. While both systems are being widely used in electronic data processing, the illustration in this chapter will be geared to a keystation-to-disc-to-tape system.

Every keyboard in a shared processor system can be used for entry, update search or verification for any of the jobs processed. Contrasted to the keypunch equipment, keystations are noticeably more quiet and attractive, making the data entry room a more efficient place to work.

Keyboards on keytapes are in all cases, compatible with the cardpunches we have studied. All the "punching" keys are located in the same position. There are however, differences between different "makes" of keytapes as to the position and function of the different functional keys. This is to be expected since each manufacturer provides different functions for their respective units.

MAGNETIC TAPE

The tape used in data processing consists of a base of polyester film, coated on one side with a thin layer of a magnetic substance together with binders, solvents, lubricating and dispersing agents and additives. The ferromagnetic material is arranged in a random fashion when the tape is not polarized.

Tapes are generally available in rolls 2400 ft. long with a standard width of 1/2". The reels on which the tape is wound, are 10 1/2" in outside diameter, although there are others of different diameters for several different applications.

All the signals which are recorded on tape result in the magnetization of the ferromagnetic oxide dispersed in the emulsion, in an area of the tape which is adjacent to the recording system ("Write Head"). There also is a "Reading System" which is part of the magnetic transducer called head: this is the Read Head. Consequently the transducer holds a Read/Write Head, which will either "read" or "write" magnetic impulses on tape. All recording on tape whether for radio, TV or electronic data processing (EDP) are based on the same principle.

The data keyed by the operator is recorded on the tape on tracks which can be compared to phonograph grooves, except that in here there is no measurable depth. Present EDP tapes are recorded on either 7 or 9 parallel tracks, arranged symmetrically and in a longitudinal fashion all across the length of the tape. Table 20.1 can be thought of as an example of a 9-track tape where each character can be represented on magnetic tape by a combination of magnetic dots in the proper position. These dots or bits are

recorded by the Head. Since the surface that each bit takes on the tape is extremely small, a lot of information can be recorded on small lengths of tape. This amount of information per inch is generally referred to as density. A 2400 ft. reel can carry more than 100,000,000 bits of information. Since the tape is fed at speeds of 200 to 300 inches per second, a transfer of up to 100,000 characters per second can be read by the head and 100,000,000 bits of alphabetic characters can be read in less than 3 minutes. Compare this speed with that of the regular cardpunch.

The length of the magnetic tape can be almost completely used. The first 10 feet are reserved for threading on the reel. The last 14 feet (approximately) are left at the end of the tape. That is, of the 2400 ft., only 24 feet (1%) is not used.

Data recorded on tape is arranged in groups or fields, each comprising one or more bits, stored in sequence; fields can be combined to form records of words of various lengths. These records are grouped into blocks of data of various lengths, depending on the system. Non-recorded gaps separate recorded data blocks to avoid loss of data or the introduction of errors.

KEYTAPE AND KEYDISC

In general terms, keytapes consist of a number of keyboard terminals (called keystations). Each keystation has a display mounted on the desk which, in many cases, resembles a TV screen. One cable that runs from keystation to keystation connects all of them to the control unit. Each keystation, however, has its own power supply, that is, each can be turned on and operated independently of the others.

The video display permits the operator to see each character of the record as it is being keyed. An entire record, of a variable number of keyed characters, can be displayed on the screen. There also is, in most cases, a character counter and a cursor. The cursor on the screen indicates exactly where the next character will be entered. Once the character is entered, the cursor will automatically move to the next field position on the screen.

Since the operator can see on the screen the data that is being keyed, the operator will immediately be able to determine whether the correct data has been entered at the correct location. If an error has been made, the operator can backspace the cursor to the erroneous character and rekey, insert or delete.

It should be pointed out that the terminal and display screen only serve as an intermediate depository of the data and the format; data will not be recorded on magnetic tape until the operator causes it, once that the required data has been completely and accurately entered.

The magnetic disc unit is a temporary storage device that allows the information being entered to be processed and checked. The disc unit is similar to a number of phonograph records stacked one on top of the other. Information may be accessed from any point on the disc tracks instantaneously.

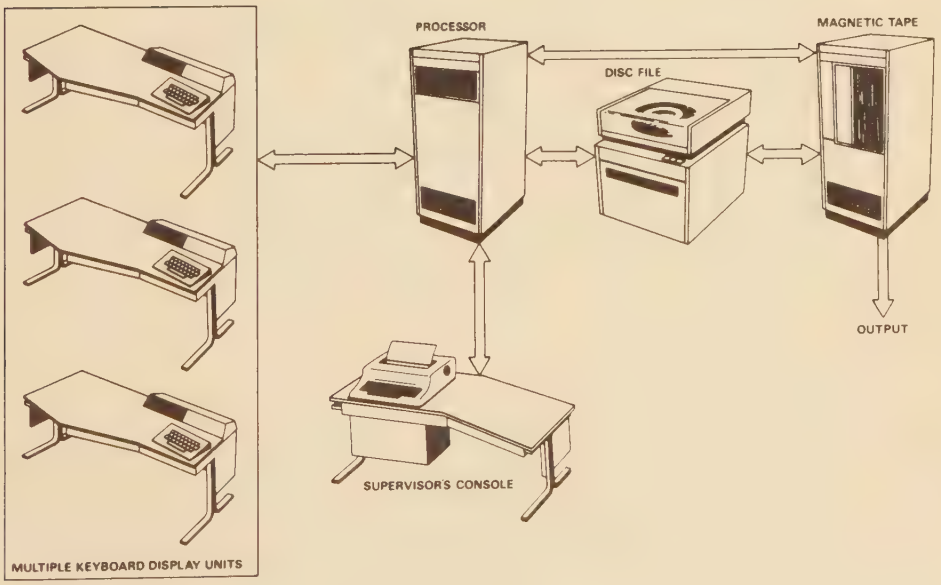
The magnetic tape unit is the final output of the keytape system. The tape functions in the same fashion as a home tape recorder. The difference is that the operator enters data characters in the form of magnetized spots (bits). Once the information is stored on tape, this tape may be used to enter data to a data processing system, a computer for instance. The rate of entry of

data on magnetic tape is much greater than on cards. Therefore the economics of using tapes, instead of cards, become obvious once again. Besides it is more economical to use than the cards, since the tape can be used numerous times, while the cards can only be used once. One reel of tape can store as much information as 100,000 thousand cards.

Keytape applications is the support of entry programs which require a large bank of keypunch or keystations terminals. It collects formats, analyzes, verifies, corrects and stores data being entered simultaneously from many keystations.

The number of operators which have access to the control unit vary from unit to unit. Nevertheless, each individual keystation is linked to the supervisor's console. Using a series of simple commands, the supervisor can assign tasks; generate program formats; permit inquiries as to the status of individual jobs, operators or control file content; and dump completed jobs from the disc to magnetic tape.

Data entry or verification takes place at any of the keystations. These keystations are operated in a manner identical to a card keypunch or verifier. All control capabilities inherent to a program care are also available in the keytape. These programs fall under the general category of "software". The illustration below shows the data flow in these systems.



The data entered from individual keystations flows to a processing unit from which they are stored temporarily on a magnetic disc. The information is then corrected and verified from the keystation before it is stored on magnetic tape. Once it has been transferred from the magnetic disc to the magnetic tape, it may be used for input to the computer.

Data being entered in each keystation is monitored under a format or program, which is based on the same principle as a program drum and card of a cardpuncher, except that the formats are stored on disc and are called for use by the operators by indicating the number of the format the operator wants to use.

It is conceivable that, due to the many advantages to be realized through the use of the keytapes for data preparation, the overwhelming majority of installations will in a few years convert their systems to keytape. As far as keypunches are concerned, the transition should be a smooth and gentle one.

MAGNETIC DISCS

Magnetic discs are thin metal discs coated, on both sides, with magnetic recording material. These discs are mounted on a vertical shaft, separated from each other so as to provide space for the movement of the access arms, which carry the Read/Write assemblies. Just as with tape, data is stored on magnetized spots on concentric tracks on each surface of the disc. Some units have 500 tracks on each surface. The number of discs mounted on each shaft varies with each system and can reach up to 25 magnetically coated, 2-foot discs.

The surface of the magnetic disc can be used repetitively, as with magnetic tape. And, as with magnetic tape, each time new information is recorded, the old information is erased. There is an advantage, however, of discs over tape, namely sequentiality. In order to search for a block of data on tape, the operator must start with record #1 and follow one by one, sequentially, until the data looked for is found. With discs, there is direct access. That is, the operator can get to any specific record regardless of location. This high-speed access to data storage location provides the operator with an easy upkeep of data files since two read/write heads are mounted between spinning discs.

COMPUTERESE

Computers function in binary states: this means the computer components can indicate only two possible states or conditions. Such as the presence or absence of a magnetized spot on magnetic tape or magnetic disc. In a computer, the values assigned to a specific number of binary combinations becomes the language to represent the data. We call this language Computerese.

The binary notations of 0 and 1 are commonly called bits. The system of coding decimal digits in an equivalent binary way is known as *B*inary *C*oded *D*ecimal or BCD, for short. A code used to represent numeric and alphabetic characters may use both positions of binary indication. By proper arrangement of the binary indications (0 bit, 1 bit), all characters can be represented

by a combination of bits. When 8 bits are used, we have an eight-bit alphanumeric code.

In some codes, each character is generally represented by a specific number of bit positions that must always contain an even or odd number of 1 bits. The decision about even or odd number of 1 bits, rests solely with hardware manufacturers.

Let's assume we are dealing with a computer where all characters must have an odd number of 1 bits: an error is indicated when characters with an even number of 1 bits are detected. This type of checking is known as a parity check. Codes that use an odd number of 1 bits are said to have odd parity. For instance, in an eight-channel tape, if we call the channels DCBA 8421, the representation of the letter A will be 1100 0001.

The Eight-bit Alphanumeric Code (*E*xtended *B*inary *C*oded *D*ecimal *I*nterchange *C*ode or EBCDIC) uses eight binary positions (bit) for each character format, plus a position for parity checking, a total of nine channels.

By using eight bit positions, 256 characters can be coded. This number of possible combinations is extremely useful for applications related to computer work, where not only alphanumeric characters are used, but where a much wider range of special characters and many control characters are needed. The following Table and Appendix 3 presents some of these combinations. Other bit combinations still have no assigned function. They are reserved for future assignment.

Data is recorded in parallel channels or tracks along the length of the tape, as previously mentioned. The tracks across the width of the tape provide one row of data: a byte. A byte is the representation of a character, such as the letter A. A byte is a combination of bits. Spacing between the vertical rows is automatically generated during the writing operation and varies, depending on the character density used for recording. Character densities as high as 1600 per inch are available (1600 bytes per inch or 1600 bpi). A longer space is automatically generated between blocks of records on tape; this space is called the interblock gap.

The following table shows the representation of the most frequent graphics using the EBCDIC representation. These correspond to the bits on tape.

Table 20.1 EBCDIC code representation

Graphic representation	Punched card code	8-bit code DCBA 8421
A	12-1	1100 0001
B	12-2	1100 0010
C	12-3	1100 0011
D	12-4	1100 0100
E	12-5	1100 0101
F	12-6	1100 0110
G	12-7	1100 0111
H	12-8	1100 1000
I	12-9	1100 1001
J	11-1	1101 0001

(cont.)

K	11-2	1101 0010
L	11-3	1101 0011
M	11-4	1101 0100
N	11-5	1101 0101
O	11-6	1101 0110
P	11-7	1101 0111
Q	11-8	1101 1000
R	11-9	1101 1001
S	0-2	1110 0010
T	0-3	1110 0011
U	0-4	1110 0100
V	0-5	1110 0101
W	0-6	1110 0110
X	0-7	1110 0111
Y	0-8	1110 1000
Z	0-9	1110 1001
0	0	1111 0000
1	1	1111 0001
2	2	1111 0010
3	3	1111 0011
4	4	1111 0100
5	5	1111 0101
6	6	1111 0110
7	7	1111 0111
8	8	1111 1000
9	9	1111 1001

The presence of a magnetized bit in any channel is represented by a 1. The absence of a magnetized bit in any channel is represented by a Ø.

21 Keytape: The Mohawk Data Recorder (6400 Series)

INTRODUCTION

The Mohawk (6400 Series) Data Recorders are exclusively of the key-to-tape type, that is, the data are entered directly (upon the operator's command) into tape. Discs, as temporary storage devices, are not employed in these systems.

The Data Recorder has two major components: the keyboard (which is very similar to the ones the operator is familiar with) and the tape deck which houses the tape feed mechanism, the Read/Write and Erase heads, and the movable parts to mount magnetic tape reels.

Data Recorders can perform three different types of operations: Entry, Verification and Search. The operating modes can be easily selected from the keyboard by using the appropriate switches.

Entry Mode

These machines have two independent types of electronic memories: one for programs and the other one for data. The program memory can store two programs at the main and alternate level. The data are stored in the data memory with a capacity of 100 positions per record. Provisions can be built in the machine for shorter or longer records.

As with other machines, a number of common, repetitive operations are simplified and hastened by the use of programs. These programs are entered directly in program memory directly from the keyboard or by using programs previously prepared on magnetic tape. Automatic Skip and Dup, are performed at the rate of 80 microseconds per position.

There is a position counter which advances one position each time there is a key depression. The display also houses other "status" indicators. All the data are temporarily stored in data memory until the record is completed. At this time, the information is released ("written") into magnetic tape as the tape passes the Read/Write (and Erase) head. Then the data memory is left with a "copy" of the same data; in other words the contents of the data in memory are not destroyed until new data is keyed in.

The Read/Write head, in the tape deck, is immediately adjacent to the Erase head. The tape passes first under the erase head and then under the read/write head. In the write (or entry) cycle, the erase head is also turned on so as to erase the area of the tape where the new record will be "written" on.

Figure 21.1

After a complete record is keyed, the tape performs a backwards-and-forwards movement to check the "written" record with data in the data memory. This is called a check-after-write movement. During the second forward movement, the erase head is turned off. All these movements take approximately 150 milliseconds. The new record will be written at approximately 0.6 inch from the previous one. This is called the interrecord gap. An 80-column card can be written on approximately 1/4 inch length of tape.

The Display on the Console indicates:

1 the position where the next character will be keyed;

2 the program code controlling that position;
3 number of data blocks recorded;
4 the character code (in computerese) of the previous record keyed at the same position displayed.

Verify Mode

On Verification, the tape movement is different: the read-after-write back and forth movement of the tape does not take place, provided that the record being verified has been found without errors. When errors are found, however, a forward-backward-forward tape movement takes place not only once but twice: the first time to enter the corrected record on tape, the second time, as the usual read-after-write check.

Programs used for verification are exactly the same as those used for entry. As before, these programs can be entered by direct keying or by tape. A "dup verify" feature permits the operator the use of the DUP key to verify adjacent fields which contain identical data.

On Verification mode, the Display on the Console indicates:

1 the position where the next character will be verified;
2 the program code controlling that position;
3 number of data blocks verified;
4 the character code (in computerese) of the previous record keyed at the same position displayed.

Search Mode

The Data Recorder can search one block of data, within a group of blocks in a tape. This is performed to enter new data, to correct existing data or to locate the ending tape mark of the previous job.

Searches are carried out at an extremely fast speed of about 1100 blocks of data (comprising many, many records) per minute. A distinctive mark in the data memory, called the identifier (key entered by the operator) locates the block to be matched with the identifier in memory. At this point, regardless of the position in the tape of the block of interest, the search stops.

THE KEYBOARD

The keyboard consists of three sections: the keyboard proper, the Memory Display panel and the switches.

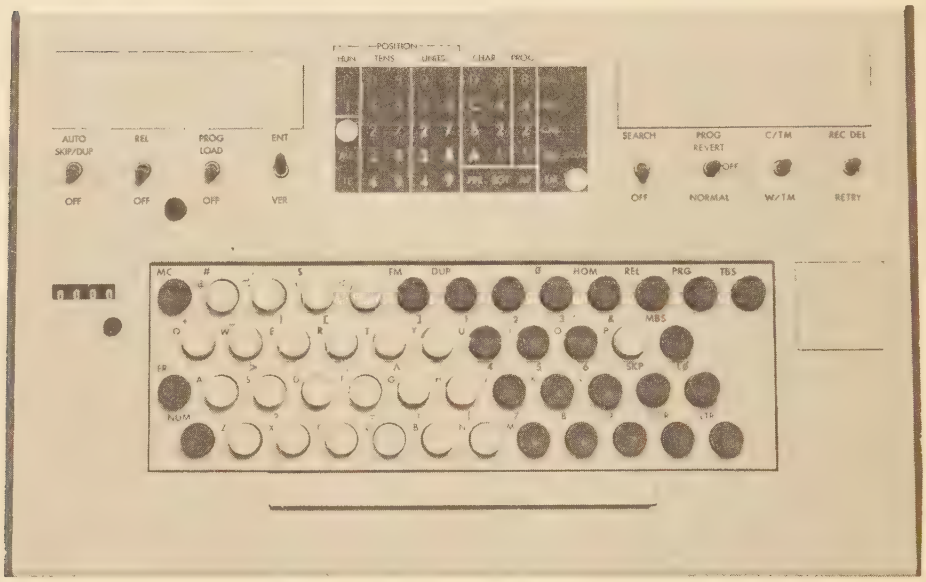
The keyboard consists of 34 character keys, 14 functional keys and the space bar. Every key has a square platform from which the round key proper is elevated. The character keys and their position on the keyboard are identical (compatible) with the characters on a regular keypunch keyboard. They are color coded and the characters they enter on tape are not printed on the round key itself, but rather on the base.

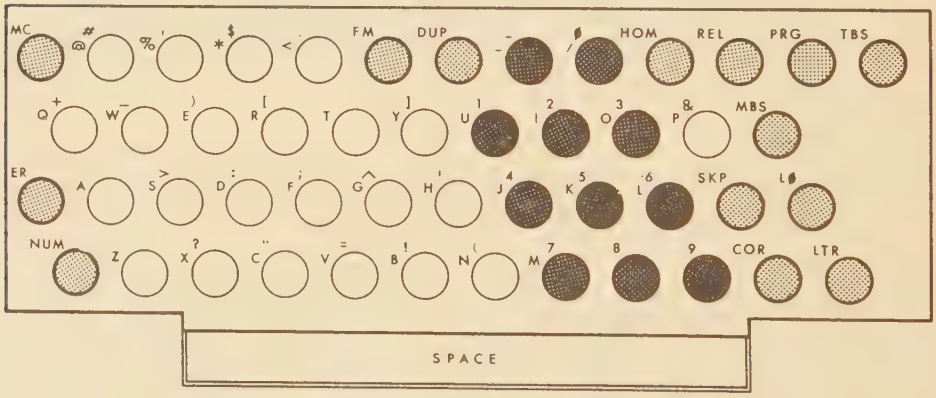
The 34 character keys enter alphabetic characters, numeric characters and special characters which, in all, permit the entry of 62 different characters. The character keys are coded by colors: blue and white tops for alpha, numeric or special characters; red top keys for functional keys.

With three exceptions, the position of the keys on the keyboard are

compatible with other keyboards described previously. The three exceptions refer to characters entered in numeric shift as follows:

Character entered in the data recorder	Numerical shift of letter	Character entered by a keypunch
⊏	R	¢
⊐	Y	|
Λ	G	¬





The keyboard is always in alpha shift unless under program control. Even if a program is loaded, it will still be in alpha shift if the PROG REVERT switch is OFF. As usual, even under alpha program control, the operator can enter numeric characters by depressing the NUM key. Special numeric characters can only be keyed while the NUM key is depressed, regardless of

program in memory. The space bar and the minus (or dash) key are functional *both* in alpha and numeric shifts. All other character keys are either alpha or numeric, not both. A numeric symbol is any numeric character other than the digits Ø (zero) to nine, space or – (dash or minus). An alphabetic symbol is any alphabetic character entered in alphabetic shift which is not any of the characters from A to Z. (See Table 1.1.)

The 14 functional keys have, in general, similar functions are previously discussed. The only functional differences are described below:

MC (*M*ulti-*C*ode)
This is equivalent to Multiple Punch and permits entering several single bits in one memory position. With one limitation: when the MC key is kept depressed, only the keys labelled D, C, B, A, 8, 4, 2, and 1, are functional. Since each of these keys enter one single bit at specific channels, on tape, by selecting the proper combination of keys, any 256 combinations of the EBCDIC code (Appendix 1) can be entered.

FM (*F*ield-*M*odify)
This is one of the "protected" keys on the keyboard which is functional when ER (*E*rror *R*elease) is kept depressed. This FM key is extensively used on verification: in these cases, no data can be entered unless this key (or COR) is depressed. As its name implies, it is used to modify whole fields on verification.

DUP (*DUP*licate)
Similar to the DUP key on other keyboards. It is also used to enter into program memory, those programs manually keyed.

HOM (*HOM*e)
This is another "protected" key: it becomes functional only when ER is kept depressed. This key is used to backspace the tape to the very first position of a record, position 001.

REL (*REL*ease)
This key discharges all the keyed data in data memory on to tape. If the operator has keyed say 50 (out of 100) positions, and this key is depressed, the first 50 keyed positions are written on tape, and the remaining 50 positions are filled on tape with spaces. This key is functional only when the *REL* (*REL*ease) *switch* on the keyboard is ON. If the switch is OFF, this key is inoperative. In other words, the REL switch can override and render inoperative the REL key.

If the REL key is depressed when the operator is at position 001, all the 100 positions are filled with spaces, not only on tape but also in Data Memory. Since this may erase from memory important data which may have to be duplicated on to the next record, the depression of this key at position 001 must be carefully weighed. If the AUTO SKIP/DUP switch is ON, data so programmed will be saved.

PRG (*PR*o*G*ram)
Two programs can be stored in program memory in the Data Recorder: the main and the alternate program. By keying PRG, the operator can switch from the main to the alternate program or vice versa.

The PROG REVERT-OFF-NORMAL switch can override this key. When the switch is at the NORMAL position, programs can be selected manually as indicated above, by keying PRG. If we assume that there are two programs in program memory, and the Data Recorder is under control of the alternate program, setting the switch at PROG REVERT will cause the Recorder to switch to the main program automatically when a data release occurs (manually, using the REL key or at the end of a record, column 100). From then on, the Recorder will be under the control of the main program, unless the alternate program is selected again from the keyboard, by using the PRG key. The PROG REVERT switch is functional only when the alternate program is in control. When the switch is OFF, the program codes in memory DO NOT control the Recorder.

TBS (*T*ape-*B*ack-*S*pace)
Another protected key which is functional only when ER is depressed. This key is used to backspace tape the length of one block of records. In this step, the data on tape is NOT erased.

MBS (*M*emory-*B*ack-*S*pace)
This is another protected key, which is functional only when ER is depressed. Its function is similar to TBS, but instead of backspacing the length of one record, this key backspaces just one position. Keeping both keys (ER and MBS) continuously depressed, backspaces the tape continuously until the keys are released.

ER (*E*rror-*R*elease)
There is, on the keyboard, one unlabelled, red ERROR LIGHT, underneath the AUTO SKIP/DUP and REL switches. This light comes on and flashes when there is an error condition. To turn the light off, the ER key must be depressed. When the light is flashing the keyboard is locked.

This key also protects other keys from being inadvertently depressed. We have already mentioned some of the protected keys. The ER must be depressed to permit keying the "protected" keys.

SKP (*SKiP*)
Its function is identical to SKIP in a cardpunch.

LØ (*L*eft-Zer*Ø*)
Zero is represented in computerese as Ø. Its function is identical to the same function in other machines.

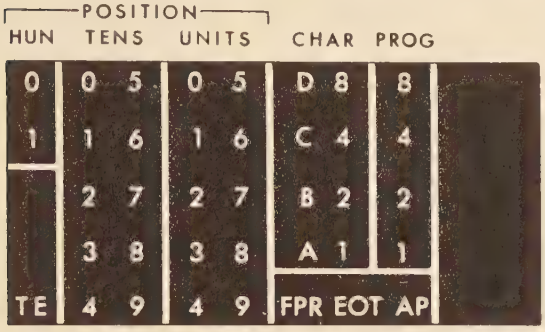
NUM (*NUM*eric Shift)
Its function is identical to the same function in other machines. It should be added that for fields programmed for numeric entries, this key must nevertheless be depressed to enter special characters appearing in upper shifts.

COR (*COR*rection)
Another protected key (ER must be kept depressed for this key to become functional). It is used during verification to correct a character in a position found to be in error.

LTR (*L*e*T*te*R*-Shift)
Its function is identical to the ALPHA key in other machines.

Memory Display Panel

The Memory Display is a piece of plastic with inscribed letters and numbers, which is located above the keys. It is divided in sections labelled "POSITION", "CHAR"(acter), "PROG"(ram) and STATUS INDICATORS.



The Position Display indicates the next position that may be keyed. Since the length of a record is generally 100 positions, there will be 3 lights in the Position Display to indicate which position is to be keyed. Three lights are lit simultaneously in any other combination. The display from 001 to 100 should be read vertically and not horizontally. One light per digit will be lit to show the column but the three lights will be constantly on; positions therefore may be displayed in this fashion: 100, 092 and 005. The count advances one unit with each character keyed, except where there is an MC (Multiple Code) entry. Home position is 001. Consequently when the data is released from data memory into tape, the counter automatically returns to 001 (except when there is a programmed SKIP/DUP field). In these cases, the counter will stop at the next position with a Stop code.

CHARacter display has 8 positions labelled D, C, B, A, 8, 4, 2 and 1; behind each character there is light which will be lit according to the character being keyed. The character being displayed does not represent the alphabetic being keyed but its computerized representation.

When numbers are represented in the character display, all the letters D, C, B and A are lit plus the combination of the necessary digits to represent any digit from 1 to 9. When all the letters are lit, but no number is shown, this represents the digit zero Ø.

Alphabetics are represented by combinations of two or three letters and at least one number. For instance, to represent the letter A the letters D and C are lit as well as number 1. Letter J is represented as DC A and the number 1. The following table shows the most frequent character combinations used.

A : DC 1	H : DC 8	O : DC A 42
B : DC 2	I : DC 8	P : DC A 421
C : DC 21	J : DC A 1	Q : DC A8
D : DC 4	K : DC A 2	R : DC A8 1

(cont.)

E : DC 4 1	L : DC A 21	S : DCB 2
F : DC 42	M : DC A 4	T : DCB 21
G : DC 421	N : DC A 4 1	U : DCB 4

V : DCB 4 1	3 : DCBA 21
W : DCB 42	4 : DCBA 4
X : DCB 421	5 : DCBA 4 1
Y : DCB 8	6 : DCBA 42
Z : DCB 8 1	7 : DCBA 421
Ø : DCBA	8 : DCBA8
1 : DCBA 1	9 : DCBA8 1
2 : DCBA 2	Sp : C

Compare the letters lit with the EBCDIC codes in the previous chapter

The Program Display shows the 1, 2, 4 and 8 numbers; each of them can be lit according to the program code in storage, individually or in combinations. Therefore, the program code can be any of the digits from Ø to 9. The program code displayed is the code that controls the next position to be keyed. These program codes will be discussed subsequently.

Once a program is loaded, the operator can use the display to tell the position she is in (POSITION), the character in Data Memory (CHAR) and the program code for that position (PROG). Consequently the combination of lights on the Memory Display at a simple glance gives the operator an overall picture.

The STATUS INDICATORS are four 2- or 3-letter combinations located at the bottom of the Memory Display. These Status Indicators, which may be lit with different colors, indicate other status of the run.

TE (*T*ape-*E*rror) (Red)
This indicator turns on when one of the following conditions exist: (1) errors during entry, verify or search operations; (2) a non-compare in the read-after-write check that takes place with the tape movement; (3) the Data Recorder read a parity-error block, a short data block or blank tape.

FPR (*F*ile-*P*rotect-*R*ing)
The File Protect ring can be easily mounted or removed on the back of the tape reel. No entries can be keyed on a reel without the ring on, during the ENTRY mode. This will prevent accidental erasures, since the presence of the ring turns the erase head off. Any attempt to enter data without the ring on, will turn the FPR light on. The ring is removed for the verification mode.

EOT (*E*nd-*O*f-*T*ape)
Tape marks, which will be discussed later, can be entered on the tape. When a mark is read into memory position 001, the EOT indicator will turn on.

AP (*A*lternate-*P*rogram)
There are two programs that can be loaded in the Program Memory: the main and the alternate program. The AP light will be on when the alternate program is in control.

Switches

There are eight switches on the panel, which have different functions. Some of these switches have identical functions as corresponding switches on other machines.

AUTO SKIP/DUP-OFF
This switch allows automatic skipping and duplication of programmed skip and dup fields. If the switch is OFF, the skip and dup codes in program memory are not recognized. This switch is NOT functional when programs are being loaded into program memory and the program load switch is set to PROG LOAD.

REL-OFF
This switch, which allows the release of magnetic data, is associated with the Entry and Verification procedures and its functioning will depend on whether the operator is in the Entry or Verification mode.

On an Entry operation (ENT/VER switch set to ENT), after a record has been keyed, the data in data memory is released to tape when the release switch is set to REL. What actually happens is that a copy of the each data position is kept in data memory until replaced by another character (this is important in Skip/Dup operations). On Verification mode (ENT/VER set to VER) the process is exactly the opposite when the release switch is set to REL. The data is read from the tape into data memory, where it will be verified by the operator. If errors are found during verification, release does not occur until those positions are corrected and reverified. This will be discussed further down.

Manual release can be done by keying the REL key, on the keyboard, in any of both bodes, Entry or Verify. All these actions are prevented when the release switch is set at OFF.

PROG LOAD-OFF
As we know, there are two program levels in program memory: the main and the alternate level. In order to enter program codes into the program memory, this switch must be set to PROG LOAD. Programs can be entered, at either level, directly by keying the codes into memory or by using program tapes.

When data is being keyed or verified, this switch must be OFF.

ENT-VER
Entry or Verify. These are two of the three modes of operation of the Data Recorder.

SEARCH-OFF
This is the third mode of operation. The meaning of search will become clear when tape marks are discussed.

PROG REVERT-OFF-NORMAL
This is a 3-position control switch which has a direct bearing on the program in control in the Data Recorder. Data Entry (or Verification, since the program codes are identical for Entry and Verification), is under control of only one of the program levels when this switch is set to NORMAL: the

operator can select manually either level. However, if keying is effected under the alternate program, with the switch set at PROG REVERT, the control level will automatically revert to the main program when a data release occurs. The program in control will then continue to be the main program until the alternate level is again selected from the keyboard. This position of the switch is effective only when the alternate program is in control.

At OFF *all* program codes are NOT recognized and the keyboard is in alpha shift.

C/TM-W/TM (*C*lear/*T*ape *M*ark-*W*rite/*T*ape *M*ark)
These switches are used to enter or clear marks on tape. This switch and these marks are important in searches and will be discussed under Tape Marks.

REC DEL-RETRY (*REC*ord *DEL*ete-RETRY)
This switch is used to erase one record in either the entry or verify mode. To erase one record, while the ER key is being held down, this switch is momentarily set at REC DEL. The operator should consider this switch or the TBS (Tape Backspace) key to correct operator-sensed errors already released onto tape.

RETRY is used to rewrite a record on tape on ENTRY mode. On using this switch, a 3-inch length of tape is completely erased. REC DEL erases only the length of one record.

TAPE DECK

The tape deck houses three parts: (1) the tape feed mechanism (with the read/write and erase heads) and the BOT (*B*eginning-*O*f-*T*ape) sensor adjacent to it; (2) the rollers and (3) the mounting hub for the magnetic tape reel.

The tape feed mechanism (upper left, Figure 21.2) has a hinged glass door which protects the feed mechanism and the heads from dust. The door must be kept constantly closed while the Data Recorder is in operation, and should be opened only for threading tape. There is a FEED/REWIND switch, under the tape feed mechanism, which has four functions: LOAD (to load tape), FEED (to start automatic tape feed, this is a spring-actioned position), REWIND (when the tape is to be rewound on the reel) and OPERATE (to position the tape for the next operation).

There are no provisions for a take-up reel, for the recorded tape. Consequently the recorded tape falls freely into a bin. In REWIND operations, the tape is just as easily picked up from the bin by the tape-feed mechanism. When program tapes are used to enter programs in program memory, a loop is made at one end of the tape and the latter placed so that the loop fits the roller, thus avoiding its falling into the bin after the feeding step.

The File Protect Ring (FPR) must be mounted on the back of the reel in all Entry operations. The ring is removed for verifications. The magnetic tape reel is positioned over the center hub to mount it on the deck, and, applying pressure to it, is pushed down firmly. It will lock in place. (To remove the reel, apply pressure to the hub with the thumb and pull out the reel.)

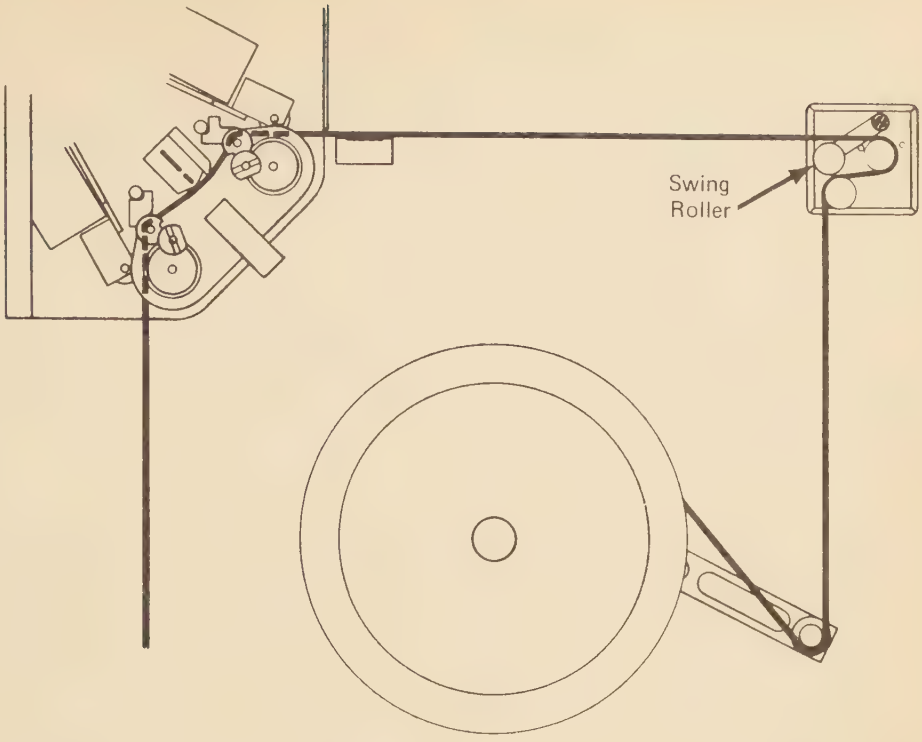


Figure 21.2

On mounting a reel on the mounting hub, it must be so positioned that the tape comes off to the right of the reel. On a feed operation, the tape is threaded as shown in Figure 21.2 above. Note the way the tape is threaded around its nearest roller and the position (to the left) of the swinging roller.

On a rewind operation, (see Figure 21.3), the tape is not wound around the nearest roller; the movable roller has been swung to the right, and the tape is now threaded.

TAPE AND TAPE MOVEMENT

The Data Recorder uses IBM-compatible tape reels of up to 10.5 inches in diameter. The tape is standard 1/2 inch wide in odd parity, with a density of 800 bpi (bits per inch).

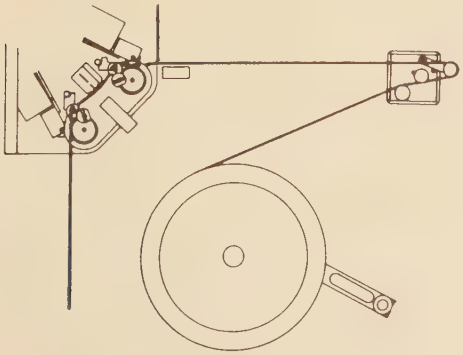


Figure 21.3

At the end of Release (of data into data memory), three actions (and tape movements) take place:

1 The tape moves forward and any old, unusable data is automatically erased. At the same time, while the tape is moving forward, a copy of the data in data memory is written. The data memory is left with a copy of its own data. At this point

2 the tape moves backwards the length of one record, the record which has just been "written".

3 The tape moves forward again and while the record is passing under the Read/Write head, it is being automatically compared with a copy of the record in data memory. The tape stops slightly beyond the Read/Write head, with the newly written record to its left. The data memory still has a copy of the original record.

The tape is now ready for entry of the next record. Certain fields in data memory will be used in the following records, under a programmed duplication. Other information in other fields will be erased as new data is keyed in.

PROGRAM CODES

The functions that can be carried out with the Data Recorders are similar to those in machines previously discussed: Auto Dup, Auto Skip, Alpha field, Numeric field, Left Zero (LØ). The code characters for the first and last position and for the rest of the field, may be different from other characters previously used, but their functions are similar to those reviewed in previous chapters. Consequently, these program codes and their functions will be reviewed briefly here.

Function	First position	Rest of the field	Last position
Numeric Field	2	spaces	space
Alpha field	3	1	1
Auto Dup	4	spaces	4
Auto Skip	5	spaces	space
Left Zero (LØ)	6	spaces	space
Left Zero (LØ)	7	spaces	space

To enter data in LØ fields, the operator, as usual, first keys the significant digits and then presses the LØ key. The keyed digits are right-justified (shifted to the right) and the field is filled with zeros from the beginning of the field.

TO LOAD PROGRAMS INTO PROGRAM MEMORY

Programs can be loaded into the main or alternate program levels in program

memory (1) keying the program directly into memory or (2) using programs already on tape.

1 To load a single program into program memory directly from the keyboard. (This can be done at any time without dismounting the reel.) The position of the switches is as follows:

AUTO SKIP/DUP	OFF
RELEASE	OFF
PROG LOAD	ON
ENT/VER	ON (Entry)
SEARCH	OFF
PROG REVERT/NORMAL	NORMAL

The Data Recorder is in Alpha mode. Consequently, to enter the digits which are the program codes, the NUM key must be kept depressed. The program codes are entered from the first position (001) on, consequently the HOM key is depressed. At this point the decision at which level to enter the program must be made and the operator, consequently, selects from the keyboard the program to be used. If the Alternate level is used the Memory Display will show the letters AP lit.

After the program has been keyed, it MUST be verified. Consequently, the mode is changed from Entry to Verify, by setting the switch to VER. After the program has been verified correct, the PROG LOAD switch is turned Off: the program is already loaded in memory.

2 To load program from program tapes (which will be described immediately hereafter), we assume that we have already available the prepared program tape. The necessary steps to load this program into memory are very similar to the steps described above for programs entered from the keyboard.

The program tape loop is isolated on the roller. The reflective marker is facing down, with the tape extended to the left and threaded into the tape feed mechanism. The switches are positioned as follows:

AUTO SKIP/DUP	OFF
RELEASE	ON
PROG LOAD	ON
ENT/VER	ON (Verify)
SEARCH	OFF
PROG REVERT/NORMAL	NORMAL

After selecting from the keyboard the main or alternate program level, proceed to load the program. As will be noted, there are two differences in the position of the switches, when compared with their position when the program is loaded from the keyboard. The Release switch is ON to allow the immediate release and the mode is Verify because we are reading from tape, which is assumed verified—correct before use. Depressing the REL key releases the program into program memory.

PROGRAM TAPES

A program tape is a short length of tape which carries one, or more frequently-used programs. These strips, properly labelled, can be stored and kept handy for future work.

Program tapes can be used at any moment, even in the middle of a run, since there is no need to clear program memory or perform any keying when loading any such programs. It is generally recommended, however, to load programs at the beginning of a run, before the tape reel is mounted. If more than one program is loaded on one tape strip, the proper program can be selected by using the search function of the Data Recorder.

Approximately four feet of tape are needed. A loop is made at one end of the tape (using about 3/4 of an inch of tape) so that the loop fold is inside the non-oxide (shiny) part of the tape. A reflective marker is placed midway between the two ends of the tape strip. This reflective marker will help to position the program tape for reading. The seal used for the loop can also be used to label the tape.

To prepare a program tape is like entering keyed data, the only difference being that the tape used is not coming from the reel, but is a short strip, not connected with the reel. Nevertheless, since this is an entry operation, a tape reel, with a File Protect Ring mounted (and the tape NOT threaded, but hanging loose), must be on the reel hub. The operator should also be aware that, at this point, the program is being written on tape but NOT loaded into program memory. This is done as indicated earlier in this chapter.

The position of the switches, on writing program tapes, is as follows:

AUTO SKIP/DUP	OFF
RELEASE	ON
PROG LOAD	OFF
ENT/VER	ON (Entry)
SEARCH	OFF
PROG REVERT/NORMAL	NORMAL

Keeping the NUM key down, the operator proceeds to enter the program codes, just like any other program. After keying the program, the operator MUST make sure that there are no errors and consequently the program tape MUST be verified. A tape mark is generally entered on these program tapes: this serves as a control when reading programs into memory.

The same program strip can be used in machines other than the one in which it was prepared. Besides, other information can also be entered on the same tape strip, such as a block containing constant information or data for a data label. This "constant block" can be read into memory after loading the program, by placing the PROG LOAD switch to OFF and depressing the REL key.

Programs entered into memory, by either procedure, must be checked for program codes, position by position.

TAPE MARKS: LABELS AND IDENTIFIERS

There are two types of tape markers. One type is a reflective marker, which is positioned manually at the beginning of the tape, on the shiny side. This

reflective marker is required to position the tape properly for enter or verify operations. When the tape is threaded in the tape deck and is being fed fast forward, the *B*eginning-*O*f-*T*ape (BOT) reflective marker is sensed by a sensor which, very adequately, is called the BOT sensor. The forward tape movement stops here.

The other type of tape mark is positioned magnetically on tape by using the C/TM-W/TM switch on the keyboard. Magnetic marks are generally placed following the last data block on tape, for computer recognition. To *w*rite a *t*ape *m*ark (W/TM) the switch must be set to the W/TM position and the code entered is 0001 0011 (A21). Seven (7) inches of erased tape follow a TM. When the switch is set to C/TM (*C*lear/*T*ape *M*ark) the TM is cleared.

To signify the end of a block of records an *E*nd-*O*f-*T*ape (EOT) mark is entered. This TM is also an A21 code. When the last record has been read and the tape moved to the next position (Home, 001), if there is a TM in this position, the EOT display, on the keyboard, lights up and the keyboard locks up. Movement of the switch to C/TM clears the EOT light and unlocks the keyboard.

Header and trailer labels are placed on tape for identification purposes. The header label identifies the beginning of a block of records. It can have a number of identifying characters or a sequence of characters like a date or a serial number. The trailer label identifies the end of a file or block of records. This identifier generally is different from the header label and may contain a record count, totals or other identifying characters.

A tape mark is generally interposed between the last record and the trailer label. In turn the trailer label is also followed by a TM. Consequently, the tail end of a block of data records is made up, sequentially, of the last data record, a tape mark, the trailer label and a TM. This last TM indicates to the computer to stop reading because it has read all the records for that particular block.

At times a job can not be finished within the day. In other cases there is a block of records that has to be modified or a tape must be searched to find the last record entered on it. In these cases, identifiers are used. An identifier may be a group of characters in one particular record. For example, the last record keyed in one particular job, was keyed under a program which featured an alpha field between columns 1 and 35, followed by other fields. The last record entered in that job, was "Manor Travel". In order to find this record, on resuming the job, the Search capabilities of the Data Recorder are used. To this effect, the same words, or part of it, say Manor, are keyed and verified into data memory between columns 1 and 35. This is the identifier and must be keyed in the same positions as on the record being searched. Spaces must be keyed to fill the rest of the record.

This identifier is kept in memory (and will not be transferred on to tape in a search operation) and when the actual search is done, each record that is read will be compared with the identifier in data memory. When a match is found, the tape stops: the search is completed.

The File Protect Ring (FPR) may or may not be inserted on the tape reel depending on whether an entry will follow the Search (in which case the ring is on), or is removed if a verification follows the search. The decision as to whether the ring is on or off the reel must be made before the search commences. Since Entry or Verification runs follow Search, and a TM is placed after the last record, the TM must be cleared (by placing the TM switch to C/TM) before continuing with the job.

Tape Marks can be erased by depressing the ER and TBS keys.

ERRORS

Errors can be made during data entry. There errors can be sensed (1) immediately after they are keyed, while they still are in the data memory; (2) after the erroneous record is already written on tape and (3) when the data in memory and the record read from tape do not agree.

1 Sometimes, the operator senses when the wrong key has been depressed. This will produce a wrong entry for that position in the data memory. Errors of this type are very easy to correct. The ER and MBS (Memory Back Space) are depressed simultaneously. The machine will backspace one position. The new, correct character is then keyed: this will erase whatever was in data memory in that position and the new character will enter in its place.

2 The erred character is on tape and in data memory. By depressing ER and TBS (Tape Back Space) the tape will back up one record length. The character entered in error still is in the data memory and on tape. Now, the record has to be rekeyed without errors. Upon completing the new, corrected data, the new data will be released from data memory into tape (while keeping a "copy" of such data) the tape will move forward, erasing the old, incorrect record, simultaneously writing the new, correct data in its place.

3 "Error Recovery Procedure": say a record has been released to tape with errors (although the data in memory is correct). The ER light will come on as well as the TE letters on the memory display. The procedure to follow in such a case is (a) to depress the ER key to extinguish the light and (b) depress the RETRY switch. This will cause the tape to backspace the length of one record: the error is still on tape and the correct record is still in memory. The tape will now move forward again the length of one record and, in the process, the erroneous record will be erased (while the correct data is still kept in memory). At the same time the record will be released from memory into tape. The difference between this case and the preceding one, is the existence of correct data in memory, in the present case.

Scratched, damaged or dusty tapes can cause the described errors. Repeated tape errors indicate the need for a new reel of tape.

VERIFICATION

This is one of the three modes in which Data Recorders can be used. The programs and program codes needed for verification, are the same as used for entry programs. Therefore, a program that applies to a particular verification job can be readily used if it is already in program memory. If the program is also available on tape, it can just as easily be loaded for job verification. Otherwise the program can be key-entered from the keyboard.

The File Protect Ring (FPR) on the back of the reel which carries the job to be verified, must be removed: this is done to avoid erasing the tape on verification, since there can not be key-entering of data without the ring on.

Corrections can be entered, on verification, only when certain keys are depressed.

During verification, as each record passes under the Read/Write head a copy is read into the Data Memory. Then, as each key is depressed by the operator, to verify, the keyed character is compared with the character in the same position in data memory; during verification, as each key is depressed, the character is NOT entered in data memory. If there is agreement between the key depressed and the character in memory, the position indicator will advance one position. If the verified record is found without errors, the tape moves forward to the next record to be verified.

Errors may be committed, however, which may or may not be on the tape. When there is no agreement between the keyed character and the one in data memory, i.e. a "no compare" situation, the Error Light comes on, the position indicator does not advance to the next position and the keyboard will lock out. These no-compare conditions may result from:

1 an error in the character on tape, or
2 the keyed character is incorrect while the character on tape is correct.

Whichever the case is, the operator must be sure that it is the proper character which is being keyed, that is, the character being keyed must be the same as the character in the source document. There is still another case: when the correct record on tape is read incorrectly into data memory.

These two "no-compare" cases mentioned above can be verified as follows:

Case 1
Error in the same position(s) as the characters on tape. Consequently the data read into the data memory is incorrect. The procedure discussed below will correct the data in memory. However, the data on tape must also be corrected by transferring the data in data memory on to tape. Therefore, verification and correction of the data in memory, is followed by transfer to tape.

Procedure
a Assume that there is only a single position in a field of data memory that has to be corrected. When the operator depressed the correct key, the error light came on. The operator now depresses two keys: ER and COR (which is functional only on verification). COR is a protected key, that is, ER must be depressed simultaneously to make it functional. Furthermore, COR locks out release: consequently, in order to relase the new, verified record from data memory into tape, it requires reverification of the entire record before it can be released.

Therefore: on finding a character in error, the operator depresses ER and COR, and then keys in the correct character.

When the record has been completely verified, the correct data is in data memory, but cannot be released because the COR key has been used.

b Depress ER and HOM. The record is backspaced to position 001. The entire record must be reverified. An automatic release writes the corrected record on tape when all the positions have been reverified without further corrections. And then,

c Depress REL, to read the next record into Data Memory for verification.

Sometimes, instead of one position, a complete field, within a record, must be corrected. In these cases, instead of using ER and COR, the operator uses ER and FM (Field Modify) and follows the same steps outlined above. FM is another protected key.

When a position of a field is corrected during verification, the forward-backwards-forward movement of the tape takes place twice. The first backspace movement takes place when the HOM key is depressed and the entire record has been verified. At the end of the verification, an automatic release takes place and the new record is written in place of the old, incorrect one. Then the usual, read-after-write check takes place: the tape moves backwards again to check with the data in data memory. Finally, when REL is keyed, the next record is read into data memory for verification.

Case 2

The keyed character is incorrect (while the character on tape is correct). The operator, on verification, depressed an incorrect character. Assume the character on the display, shows an alpha character, letter A, in the next position to be verified. The operator, inadvertently, keys a B for the same position. The error signal comes on and the keyboard locks out. To stop the error light, the operator depresses the ER key. This unlocks the keyboard. The operator must assure herself that the character to be keyed is the right one, by checking with the source document.

Case 3

There are times when the Data Recorder may make errors on reading from the tape. The net effect is that the record is read incorrectly into data memory.

In these cases, the error light starts to flash. The operator must depress the ER key to extinguish the light and then the RETRY switch to backspace the record and re-read it, this time, correctly.

22 Key-to-disc-to-tape: The Mohawk (2400 Series) Key Display System

INTRODUCTION

The Mohawk Key Display Systems introduce a radical departure from other systems reviewed herein before. The end product is an IBM-compatible tape, carrying all the keyed data. The similarity stops here. Between the keying operator and the tape, a number of other pieces of computer hardware are now interposed. They facilitate the operator's job enormously despite appearances to the contrary. The system comprises a regular keyboard (here called keystations), with a TV-like screen, a Multiplexer (within a Processor), a Magnetic Disc drive and finally a tape drive where data entered *via* the keystations is finally recorded on magnetic tape. A number of keystations (up to twenty, depending on the installation) are hooked up to the same Processor. The Processor polls each keystation in sequence, requesting data; then, automatically, without any intervention from the Operator, the Processor regulates, consolidates and coordinates the data influx sent by the keyboard. Another keystation, also hooked up to the same Processor, supervises all these activities and, ultimately, upon command from the Supervisor's Console, verified data is finally dumped from the disc onto tape.

A number of common, well known operations can be carried out from these keystations: Entry, Verify and Search. Besides, a new mode the *S*ingle *O*perator *R*esponsibility (SOR) has been added to this system capability.

One of the features of the Key Display System is a display screen, called the *C*athode *R*ay *T*ube or CRT; the CRT provides immediate and constant reference of all keyboard activities, not only of keyed data but also of other commands which can be initiated from the keystation. The beginning of a job always entails a "dialogue" between the Operator and the system. Statements are flashed on the CRT screen with inquiries, from the system, on the nature of the job. These inquiries are answered by the Operator by depressing numbers or single alphabetic characters.

The graphics shown on the screen are the regular characters as we know them, not in computerese. Although the keystations are linked with the others and the Supervisor's console, each individual console can be operated by itself, regardless of the status of the other keystations integrated in the same circuit, i.e. each keystation has its own power supply which can be turned on or off independently. This independence permits more than one

Operator to be involved in one job. For instance, if there are several pages of a source document, which is going to be entered under the same program, the data can be key-entered into the Processor by different Operators. On the other hand, working on the same page of a source document, one Operator can start key-entering its data and, from another keystation, another Operator can start its immediate verification (after the first three records have been recorded). There is still the further possibility that an Operator can verify each record immediately after Entry. This single operator mode is a process of entry/verify each record until the complete batch has been keyed and verified.

Each keystation has manual controls to regulate the picture brightness. There also is a switch which enables the Operator to completely blank off the keyed data on the screen. Furthermore, since each keystroke (data or functional) generates a small "click", there is a volume control to adjust the loudness. There also is a tone when a complete record (here called a page) of 125 columns, is released into the disc.

The keystrokes apparently are immediately displayed on the screen; however the signal generated by each stroke goes first to the Processor, which controls the handling of the keystrokes, by the proper programs. Only then the signal is returned to the keystation screen, *via* the Multiplexer, and it is then shown on the display. At the same time, the keyed data is temporarily

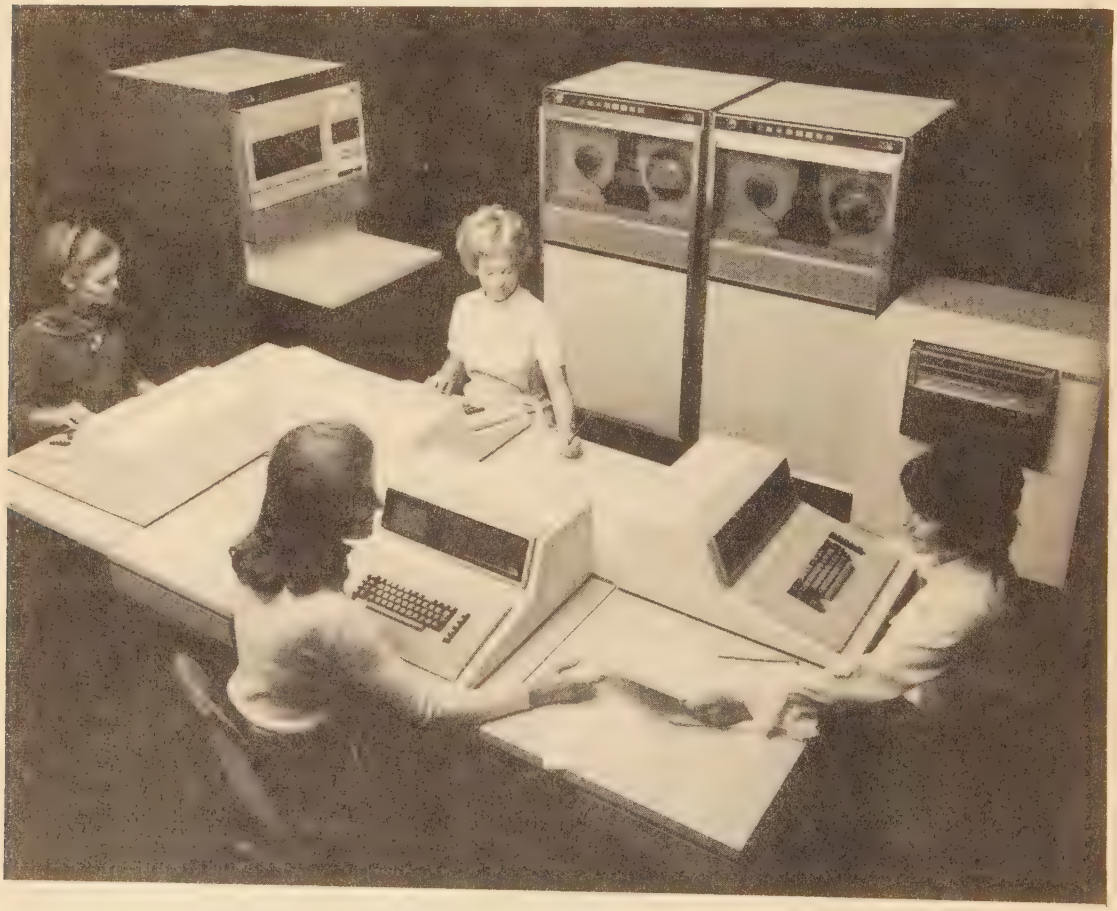


Figure 22.1 The Key Display System

Figure 22.2 The keyboard

stored on disc. Therefore, there is no direct connection between the keyboard and its own screen. The signal returns by an indirect route to the screen. This process, however, takes only milliseconds and no time lapse is apparent to the operator.

The keystations work under the control of programs (which here are called formats). The data polled from each keystation by the Multiplexer (in the Processor), is accepted and organized by the Processor under the control of the same format used by the keystation. Then the data is sent to, and temporarily stored by, the disc. The formats (programs) used can be key-entered into the keystation or retrieved from the disc (*via* the Processor) where they may be temporary or permanent residents of the program library. The Processor also regulates the output of data from the disc to the tape, since the Processor is directly linked to the Supervisor's keystation.

The disc has two magnetic surfaces with 203 tracks each. Each track, in

turn, has 40 sectors. Into each sector, 128 characters can be keyed: 125 for data and 3 for control. Each sector is the equivalent of a punched card but with 125 positions instead of 80. A single disc drive has a storage capacity of 2,048,000 characters. The disc normally stores 256 formats. Since formats can have subformats (up to 4 subformats per format) the storage capacity of the disc is of up to 1024 subformats. Each sector can be directly accessed by the Processor *via* the Read/Write head. The data on the disc, under commands, can be dumped onto magnetic tape. The tape reels are housed in tape drive.

The Operator will be mainly concerned with activities centered around the keystation which will be the main topic of discussion of this chapter. The keystation, as mentioned above, comprises the keyboard and the display screen.

THE DISPLAY SCREEN

The *C*athode *R*ay *T*ube (CRT) or display screen can display up to 240 characters, 8 lines with 30 characters each. Job status information is displayed on the first two rows: the third is generally blank to help separate the status data from the job data.

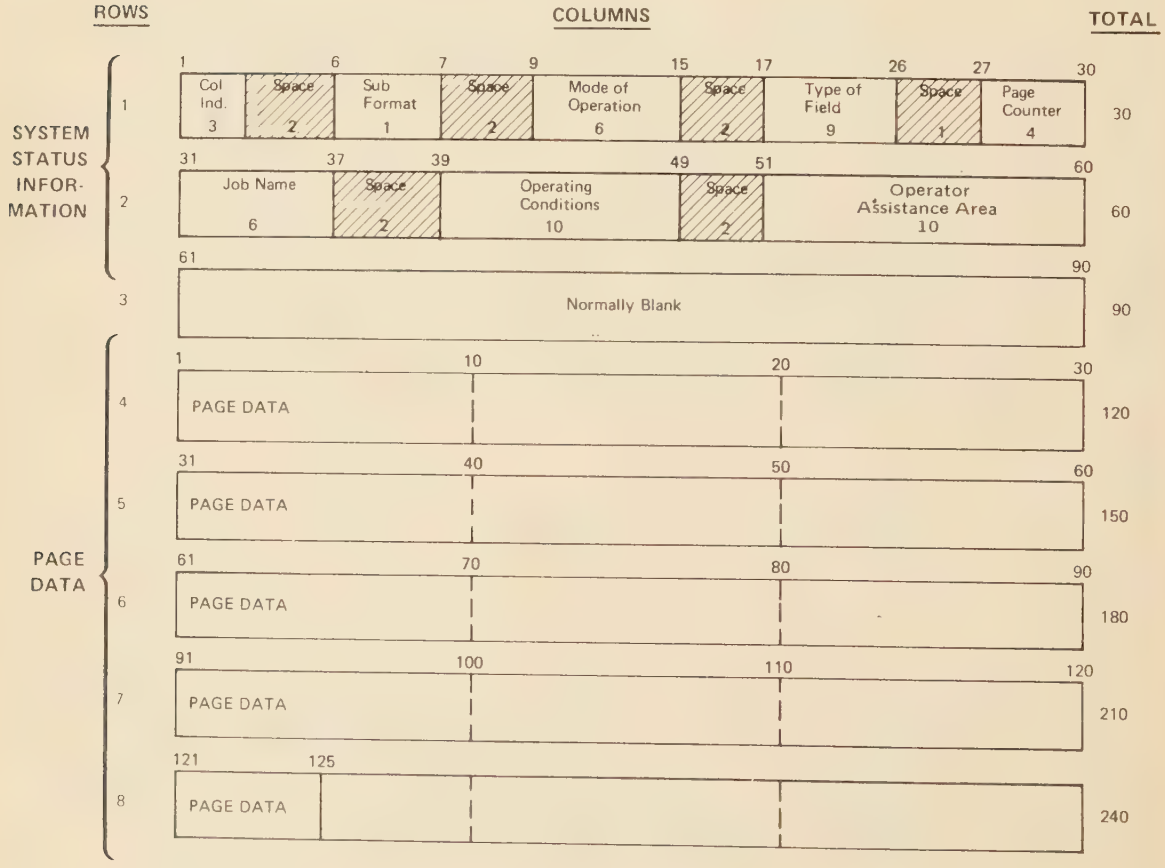


Figure 22.3

On the next four rows job data is actually key entered, up to a maximum of 125 characters: this is the maximum number of characters allowed per page (record). The last 25 positions on the last row are not used. The position on the screen where the next character will be entered is shown by a "Cursor". This is a bar of low illumination which "travels" from position to position when data is key entered.

The first 60 columns on the first two rows, display status information and will be discussed in the eight items below, making references to Figures 22.3 and 22.4.

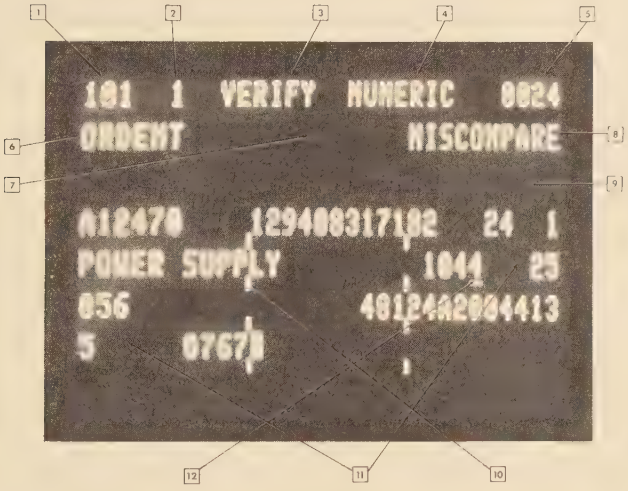


Figure 22.4

1 This is the position of the cursor on the page being entered. The number shown, in Enter mode, indicates where the next character will be entered. In Verify mode, the position to be verified is shown. This is a 3-column field separated from the next one by a 2-space blank.

2 The concept of subformat is similar to the idea of main and alternate programs available in card punches. When a job required two programs at different levels, they were both punched on the program card. Either program could be made to control the whole punching operation by a simple depression of one key on the keyboard. In the Key Display system, there may be a main program and three alternate ones. In this system, they are called formats and subformats. The subformat number, a 1-column field, identifies which of the four (optionally eight) formats is controlling the key entry. These subformats are chained to each other within the same format. This field is also separated from the next one by a one-column blank field.

3 The mode of operation: this 6-column field indicates to the Operator the mode being used. The modes used in the Key Display system are as follows:

ENTRY	Entry Mode
VERIFY	Verify Mode
SOR E	Single Operator Responsibility in the Entry mode
SOR V	Single Operator Responsibility in the Verify mode
FORMAT	Indicates a new format is being key entered
SEARCH	Search mode. (This mode was discussed in the previous chapter).

This field is also separated from the next one by a 2-position blank field.

4 Type of field. The format in control determines the type of field which is controlling the Key Display. This 9-column field in the display, can show one of the following messages:

DUP	Duplicate Field
LEFT O	Mandatory Left Zero Field
LOWER SH	Alpha Field
MAN NEG	Mandatory negative Left Zero Field
NUMERIC	Pure Numeric; only the 0 through 9 digits and the minus (–) sign can be entered.
UPPER SH	Special Characters, upper shift symbols, can be entered.
SKIP	Skip Field.

The Key Display can be optionally prepared so that other field types may be shown. These can be:

EXT CK (for EXTension CheckK field), VALD CH (VALiDation CheckK field), CK DIGIT (CheckK DIGIT field), ACCUM (ACCUMulator field) and ACC/CRF (ACCumulator plus CRossfooter Field).

5 The page counter. This is a 4-position field. Its value increases by one each time a complete page of data has been key entered or verified. In the Search Mode, it displays the page number when the desired record is found. Batch-Header and End-Of-Batch records (which could be compared to tape labels) are not counted.

This field completes the first row of status indicators.

6 The job name. Each format is entered in the program library under a given name. As such, it can be recalled at any time. It defines a chain of one of four subformats.

This is a 6-column field separated from the next by a 2-column blank field.

7 Operating Conditions. This 10-column field shows which functional key was depressed last or the status of certain system operations (it is blank in Figure 22.4). The messages can read as follows:

NEG FIELD	Negative Field being key entered of verified
CH CONST	The dup field change (D/S OFF) functional key has been depressed.
CH SKIP	D/S OFF functional key depressed for a Skip field change.
FIELD CORR	FBS/COR functional key being depressed
CHAR CORR	CBS/COR functional key being depressed
PAGE BKSP	PAGE BS/DEL functional key depressed in Entry Mode
PAGE DELT	PAGE BS/DEL functional key depressed in Verify or Search Mode
PAGE INSRT	PAGE INS functional key depressed in Verify or Search Mode
MULTI CODE	MC functional key depressed

SEARCH FOUND	Message to indicate the successful conclusion of a Search Operation
NOT FOUND	Message to indicate the unsuccessful conclusion of a Search Operation
SEARCH CORR	Message to indicate that a page is being re-entered in conjunction with a Search Operation
SCAN	SCAN functional key depressed in Verify Mode.

"Operating Conditions" is separated from the next field by a 2-column blank field.

Additional messages can be generated by the Operator's actions on the keyboard. These messages continue the "dialogue" between operator and machine to facilitate and round-off a certain list showing some of these messages:

ENTRY COMPLETED	NUMBER OF PAGES
ENTRY STARTED	NUMBER OF DELETIONS
VERIFY STARTED	NUMBER OF INSERTS
VERIFY COMPLETED	NUMBER OF KEYSTROKES
VERIFY NOT REQUIRED	NUMBER OF CORRECTIONS
BATCH ALREADY ENTERED	CLEAR DISC ?
BATCH NOT IN SYSTEM	TAPE LABEL ?
INCORRECT FORMAT	EOF CODE ?
INCORRECT BATCH NAME	STANDARD ?
VERIFY NOT INITIATED	SELECT MODE
MULTI BATCH	END OF DAY ?
FORMAT NOT IN SYSTEM	BATCH NAME
BATCH IS NOT DORMANT	VERIFY ALREADY STARTED
ENTRY PHASE NOT COMPLETED	FUNCTION COMPLETED

8 Operator's Errors. This 10-column field show messages which are generated by the Operator's keying errors. These messages may be as follows:

DUP ERROR	NUMERIC
SKIP ERROR	INVAL KEY
MISCOMPARE	REL ERROR
LEFT ZERO	BACKSPACE
FORMAT	NEG FIELD

Simultaneously with the message, the error light on the keyboard begins to flash. Additional error messages can be displayed in machines especially prepared.

9 The previous items are the "status conditions" of the job being keyed. The next row, columns 60 through 90, is blank. The following 5 rows are left for the display of data key entered.

10 Two vertical lines appear on the screen, dividing the last 5 rows into three groups of ten positions each.

11 Left zeros appear as blanks on the screen. This eases screen readability.

12 Negative fields are shown with the last digit of that field underscored.

The entire page is shown on the screen until the first character of the next page is keyed. At this time, the previous page disappears from the screen. Data on these last 5 rows can also be blanked off by the use of the blanking switch located in the Operator control area of the keystation.

THE KEYBOARD

The keyboard comprises 34 character keys (capable of producing 64 alphameric graphic characters), 20 functional keys (including the alpha and numeric shift keys) and the space bar. There are no switches and there is an error (flashing) light. The character keys are located on the keyboard in an identical manner with other keyboards previously discussed. Characters, other than the regular alphameric graphics, but part of the 256 EBCDIC combinations, can be generated with the functional key MC.

Some of the functional keys described below have been discussed extensively in Chapter 21, when the same keys were described in detail. Therefore, the function of identical keys will be described succinctly.

SEL MODE (*SEL*ect *MODE*)
Any keyboard procedure, at all times, must be initiated or terminated with a depression of this key. This causes the screen to display the System Procedures, as shown in Figure 22.5 below.

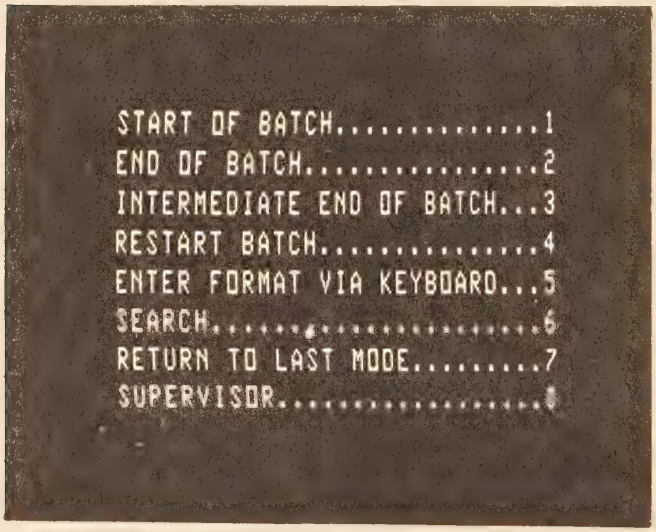


Figure 22.5

At this point, any new procedure may be commenced by depressing the appropriate digit; or, if this is not desired, by depressing the "7" digit, the system is returned to whatever procedure the system was at.

REL (*REL*ease)
The function of this key is very similar to the REL key described in Chapter 21: by pressing it, all the keyed data is released into the Processor. If the Operator, on Entry mode, has keyed say 50 columns (out of 125), depressing this key will cause the data to be transferred to the Processor. The remaining positions are filled with blanks. The cursor disappears from the

screen and the page, with its keyed-in record, is displayed until the first character of the next record is entered.

On Verify mode, the remaining positions are checked for spaces (except for the formatted dup field). If one of the columns being verified contains a character other than a space, an error condition is produced.

Should REL be accidentally keyed, on Entry mode, the previous page can be returned to the keystation by keying FBS/COR or HOM. In Verify mode, an error condition is produced.

HOM (*HOM*e)
This key may be used during Entry and Verify modes; it backspaces both the cursor and column indicator to the Home position, column 001. In the process, all keyed in data is deleted (except for data in formatted dup fields).

MC (*M*ulti-*C*ode)
This is equivalent to Multiple Punch and permits entering two single bits into one column. While this key is kept depressed, the cursor does not advance. Only a few of the character keys on the keyboard remain activated when MC is depressed. These are the letters A, B, C, D, E and F, and the digits 0 through 9. This will allow the Operator to enter any of the 256 hexadecimal combinations listed in Appendix 3. When the second character is depressed, the cursor advances to the next column. The graphic equivalent of the combined character will be shown on the screen, if such a character exists.

DUP (*DUP*licate)
Similar to the DUP key on other keyboards. The duplicated data is displayed on the screen.

SKIP (*SKIP*)
Its function is identical to SKIP on other keyboards.

DIS/OFF (Automatic *D*uplicate/*S*kip *OFF*)
This new function permits the operator to change the data in fields formatted for automatic duplication or to enter data in blank fields formatted for automatic skip. The effect of depressing the D/S OFF key is to ignore the dup and skip functions of fields which are to be modified.

This key may be used in either the Entry or Verify modes, although with different effects. During Entry mode, D/S OFF can be used to enter or change data in fields already passed over or before the field of interest is reached. After new data is entered in these fields, the system backspaces to the first column of the field just rekeyed, forcing its verification.

During the Verify mode, the D/S OFF key permits the verification of new data in a dup or skip field.

LEFT ZERO
Similar to other LEFT ZERO functions previously described.
— (Minus). This key is used to enter a negative number in a numeric field only. Depression of this key causes an automatic Left Zero and the negative number will be right justified.

SUB 1 (*SUB*format 1)
SUB ADV (*SUB*format *ADV*ance)
These two keys can be used only in the Enter mode and its effects will be better understood when the Operator becomes familiar with the concepts of format, subformat, Format Field Descriptors (FFD) and Procedures, described later on.

SUB ADV permits the Operator to escape from one subformat into the next one. SUB 1 permits the Operator to return to Subformat 1.

ERR RSET (*ERR*or *ReSET*)
Whenever an error condition is detected by the system, the keyboard locks up, the error flashing light comes on and there generally is an ERROR MESSAGE on the screen. In order to continue operating, the ERR RSET key must be depressed: this unlocks the keyboard and turns the light off.

CBS COR (*C*haracter *B*ack*S*pace-*COR*rect)
FBS COR (*F*ield *B*ack*S*pace-*COR*rect)
PAGE B/DL (*PAGE B*ackspace/*DeL*ete)
These three keys enable the Operator to correct a "sensed" error in data entry: a single character, a complete field or an entire page can be corrected or deleted depending on what key is depressed in Entry mode. All characters, fields or page backed over, are deleted.

In the Verify mode, corrected character (or field or page) is rekeyed twice: once for entry then for verification.

Page backspacing can be repeated over and over until the page of interest is found. In the process, the Operator may delete numerous pages which will eventually have to be rekeyed. In this situation, the Operator may instead consider performing a Search Operation which will be described later under Operations. Therefore, instead of backspacing pages, the Operator initiates an intermediate-end-of-batch operation which will maintain the integrity of the data and then perform the Search. The Operator can return later to the last page keyentered, without losses.

SCAN (*SCAN*)
The effect of pressing this key is to display the current page. This key is operational only on Verify or Search modes and permits the sight check of all the characters of a page. When the next character key is depressed, the screen will display only the characters which have already been verified. This key "protects" the PAGE INS key.

PAGE INS (*PAGE INS*ert)
This key is also operational in Verify mode and can be depressed only after SCAN has been activated. This key has the effect of overriding the Verify mode, returning the system to the Entry mode.

Depressing SCAN will permit the operator to sight check if one or more pages are missing. If this is the case, upon depressing PAGE INS, data can be entered into a page. The system forces the immediate verification of the data so entered by reverting automatically to the Verify mode.

⟶ (*CURSOR FORWARD*)
This key is used to advance the Cursor to the first position of a field where an "identifier" will be entered. This operation is performed only in Search

mode. The meaning of an "identifier" in conjunction with a search has been detailed in Chapter 21. The Search operation is described later in this chapter.

This key is also used for a sequential view of all the pages of a batch, searching for a non-existent identifier generated by this key.

CONT SRCH (*CONT*inue Sea*RCH*)
The function of this key is directly associated with searches. When the page searched for has been found, this key permits the search operation without having to re-enter the search identifier.

FORMAT CODES, FIELD DESCRIPTORS, FORMATS AND SUBFORMATS

Formats and subformats, stored on the disc, control the Entry and Verify procedures of the Key Display System. The type of data and how it is arranged within each page is governed, as we have seen before, by programs which now are called formats. Coding data such as type of field, length of the field and whether or not the field will be verified, is entered on disc as Format Field Descriptors (FFD). The format (and subformat) organization does not represent a radical change from concepts learned previously; the only difference is how the information is organized before being entered onto disc.

If all the fields of a source document fit on one page, the operator needs just one program (format). Therefore the format consists of one subformat. However, if the document is such that it requires two programs, the format would consist of two subformats, chained one to the other. This is similar to the concept of main and alternate programs, with which the Operator is familiar. With the Key Display System, however, there is the possibility of having four (and optionally eight) subformats automatically chained to each other. Any of these may be understood to be the "main" subformat. Therefore, when the Operator enters a subformat from the keyboard, a chaining number must also be entered (see Figures 22.10, .11 and .12, where the chaining numbers are 3, 3 and 1, respectively).

If the new format contains one subformat, the chaining number is 1, that is, it will be chaining the format to itself; when there are two subformats in a format, subformats 1 and 2, the first subformat key entered must carry the chaining number 2, and the second subformat, the chaining number 1. The reason is simple: when the Operator is entering (or verifying) data under control of subformat 1, this subformat will automatically yield control to subformat 2 at the end of the page. Therefore, the chaining number 2. Consequently, on keying under control of the second subformat, the chaining number must be 1.

When one subformat (say subformat 3, in a 4-subformat job) is repetitively used ("main program"), its chain number will be 3. That is, it will be chained to itself. When all the data under this subformat has been keyed, the Operator escapes from subformat 3 to subformat 4 by depressing the SUB ADV (*SUB*format *ADV*ance) functional key. The chaining number in subformat 4, is 1.

Each operator can enter new subformats into the disc from any keystation without interfering with other operations. Subformats can also be loaded

from tape. A maximum of 256 formats may be stored on disc. If the disc is full, the Supervisor can make room on the disc for new formats, by deleting others from it. This function can only be initiated from the Supervisor's keystation.

Subformats may be entered from the keyboard. This is described in detail under Procedures. At the present time the Operator should be made aware on how these subformats are organized, before they are entered on disc. A format comprises a number of statements called *F*ormat *F*ield *D*escriptors (*FFD*). Each FFD corresponds to one field of data. FFDs comprise three parts: the first is an alphabetic character; the second is a digit and the third part is comprised of three more digits. Therefore, an FFD may look like this: D2015. The same formats and FFDs are used for entry and verification.

The first symbol in an FFD, the alpha character, defines one of the thirty two field types available in Key Display Systems, as shown in Table 22.1 below.

Table 22.1

FIELD TYPE	CODE	FIELD TYPE	CODE
DUP	A	VAL. CHECK FIELD 3	Q
SKIP	B	VAL. CHECK FIELD 4	R
UPPER SHIFT (NUMERIC)	C	CHECK DIGIT 1	S
PURE NUMERIC	D	CHECK DIGIT 2	T
LOWER SHIFT (ALPHA)	E	ACCUMULATOR 1	U
MANDATORY NEGATIVE	F	ACCUMULATOR 2	V
COMPULSORY LEFT ZERO	G	ACCUMULATOR 3	W
EXT. CHECK OP.1 (DEFINITION)	H	ACCUMULATOR 4	X
EXT. CHECK OP.2 (ADDEND)	I	EXT. CHK OP.3 + ACCUM 4	Y
EXT. CHECK OP.2 (SUBTRAHEND)	J	ACCUMULATOR 1 + CRF	Z
EXT. CHECK OP.2 (MULTIPLICAND)	K	ACCUMULATOR 2 + CRF	<
EXT. CHECK OP.2 (DIVISOR)	L	ACCUMULATOR 3 + CRF	/
EXT. CHECK OP.3 (AUTO DISPLAY)	M	ACCUMULATOR 4 + CRF	>
EXT. CHECK OP.3 (KEYED ENTRY)	N	Not Used	_
VAL. CHECK FIELD 1	O	Not Used	_
VAL. CHECK FIELD 2	P	Not Used	_

The first seven codes (A through G) are field descriptors which are familiar to the Operator. Therefore, their description will be very succinct. The difference between Numeric and Upper Shift characters has been described previously in this chapter. The possibility of left- or right-justification for alpha or numeric characters has been described in Chapter 17 under LEFT ZERO (or LEFT BLANK) fields. Consequently, in the Key Display, words in Lower shift (alpha) fields can be either left-justified (using the SKIP key) or right aligned (using the LØ key).

Any upper shift numeric field can be made into a left-justified field (zeros or blanks after the significant digits) using the SKIP key, or a Left Zero field using the LØ key. However, when credit data is entered, the advantage of using negative fields is apparent. Consequently, the use of negative fields. Data keyed in negative fields (F) can be made negative by the depression of

the minus (–) key which, in this type of field, also performs the LØ function.

For compulsory LØ (G) fields, data keyed can be entered by depressing the LØ key or the minus (–) key. Leading zeros are automatically verified in these fields.

The second character of the FFD, always a digit ranging from 2 to 7, described the field's shift state and whether or not the field will be verified. Table 22.2 describes the Shift/Verify Codes.

Table 22.2

SHIFT STATE	KEY-VERIFY	NON-VERIFY
Upper (NUM)	2	3
Lower (ALPH)	4	5
Special Option	6	7

This Table shows the codes used for the second character in the FFDs. All the fields labelled non-verify are automatically skipped in the Verify Mode.

The last three characters of a FFD describe the length of a field, in columns.

PROCEDURES

There are eight operations that can be originated from any keystation. There are a few which can be originated from the Supervisor's console only. These operations are very simple to start and, in all cases, entail a "dialogue" between the Operator and the screen. The Operator depresses certain functional keys. As a consequence, messages appear on the screen, which must be answered by the Operator from the keystation by depressing certain keys. When these preparations are finished (which take very little time) the system is ready to accept a number of operations, like entering a format from the keyboard, entering the data under that format from the same or other keystations, verification, searches, etc.

The system Supervisor, at the start of each workday, must perform a procedure which will ready the Processor for multiple keystation entry, prepare the disc and, optionally, load formats stored on magnetic tape onto the disc. These initialization procedures take between three and five minutes, after which all keystations are ready for operation. At the end of the day, the Supervisor must perform, from her console, an end-of-day procedure to bring all operations to an orderly close.

We will now describe a typical Operator's procedure, starting with the generation of the format from the keystation, its verification and loading into disc; and the data entry under such format.

Starting a Procedure

In *all* cases, for any procedure, the Operator must first depress the SEL MODE key. Whatever was on the screen, prior to depressing the key, disappears and a new message, the System Operations, Figure 22.5 is displayed. This figure illustrates the procedures available to the Operator.

In a way, this message is asking the Operator which procedure will be carried out. Since the answer must be a digit, the keyboard is automatically

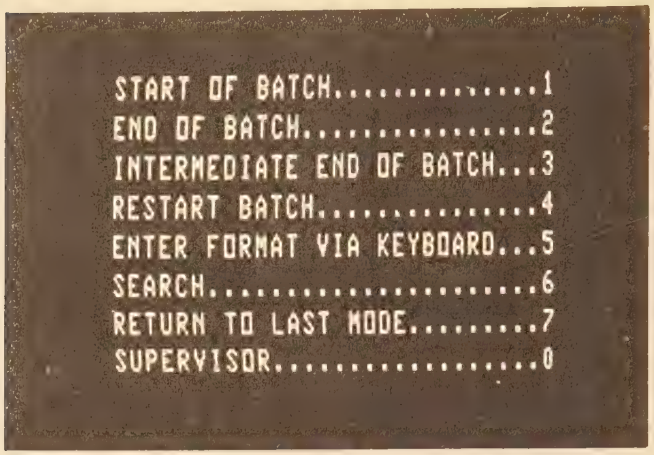


Figure 22.5 Systems operating procedures

placed in Numeric shift. After the digit has been keyed, it must be *REL*eased into the Processor by depressing the REL key to initiate the desired operation. Since we are describing the start of a procedure, from the creation of the format to the verification of data, we will assume that the Operator has depressed the key to enter the "5" digit, followed by REL to "ENTER FORMAT VIA KEYBOARD" procedure. Data Entry and Verification will follow.

ENTER FORMAT VIA KEYBOARD (Procedure 5)

This procedure permits the Operator to enter from the keyboard a new format (which we will assume was not previously available or not stored on magnetic tape) to control the Entry or Verification of data in a source document. This format, at the end of the keying procedure, will be temporarily stored on disc.

As a result of the depressing of 5 and REL, a new message appears on the screen: Figure 22.6

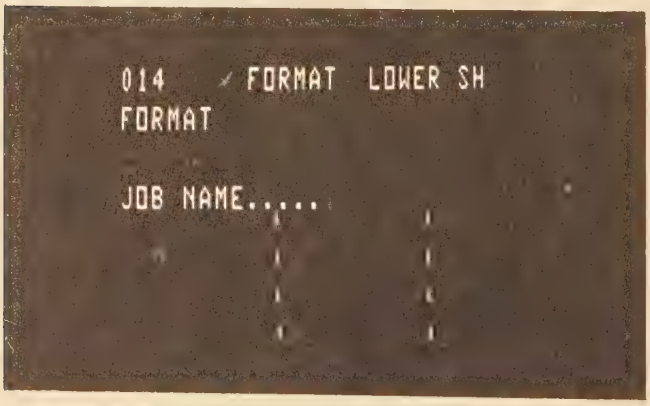


Figure 22.6

The keyboard is automatically in lower shift (alpha) to allow the Operator to enter alpha data i.e. the job name. A name MUST be given to each job so that when the job's format and data are stored on disc and/or tape, its location (address) can be easily located, for future reference or other procedures. We will assume that this job will be named ORDER. Job names consist of 6 characters (maximum). After the Operator has keyed ORDER, the screen will appear as shown in Figure 22.7.

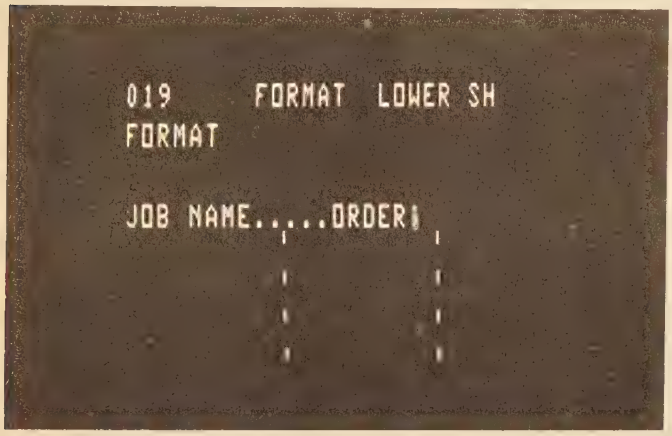


Figure 22.7

After the job name has been keyed, the system immediately changes its mode to Verify, forcing the verification of the job's name. If an error condition is present, the system will detect a miscompare and the MISCOMPARE message will be flashed on the screen. If the job name is verified correct, the system proceeds to check with the disc for duplicate names. If a duplicate name exists on disc, an error message (FORMAT ERR Ø) is flashed on the screen. The procedure will have to be started again. (Format errors will be discussed later.) On the other hand, if there is no duplicate job name on disc, a new message will be displayed on the screen.

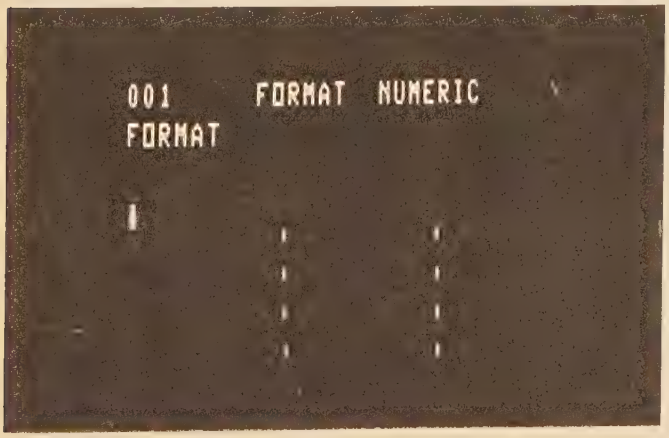


Figure 22.8

This message shown above indicates that:

a the cursor is in the "Home" position;
b the operation being performed *via* the keyboard is a FORMAT;
c the keyboard is in Numeric shift. This first character is always a numeric

character (the chaining number) and 4 out of 5 FFD characters are numeric characters.

The Operator is now ready to enter the new format.

When format codes and FFDs were discussed, the organizational aspects of formats and subformat were explained. Consequently, the first digit key-entered must be the subformat chaining number (from 1 to 4). If we assume that the new format being entered is comprised of 4 subformats, the subformat code appearing with the field descriptors for the first subformat, will be the digit 2, indicating that subformat 1 is chained to subformat 2. See Figure 22.9.

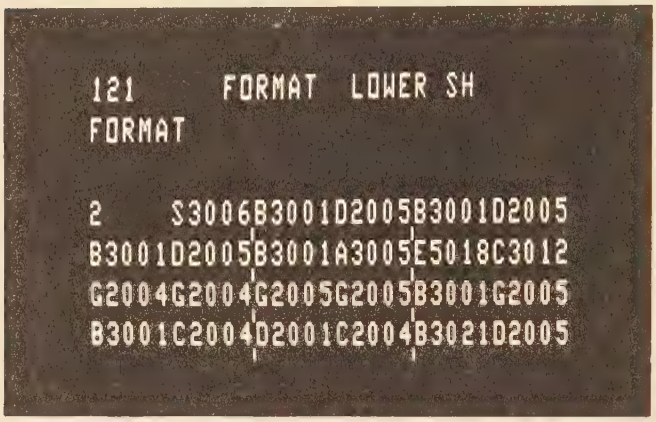


Figure 22.9

For the second subformat, see Figure 22.10.

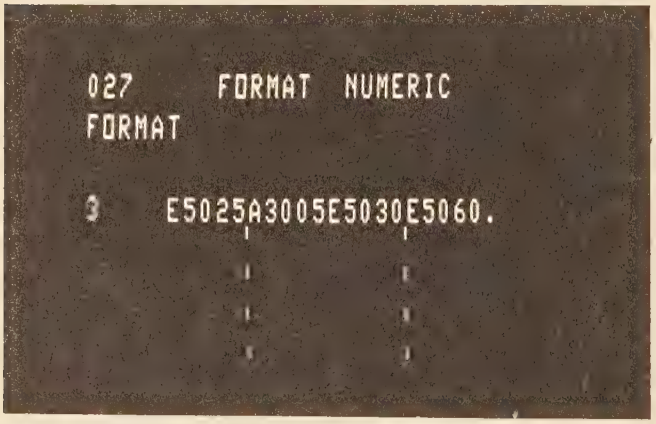


Figure 22.10

After all subformats' FFDs have been keyed, the Operator keys a period (.) and then REL to indicate the end of a subformat to the system. In *ALL* cases, at this point, the system automatically reverts to the Verify mode whereby the Operator is forced to key verify the subformat chaining number, the FFDs and the period (.).

If the new format contains only one subformat, the chaining number is 1 (thereby chaining subformat 1 to itself); if another subformat, say subformat

#3 is used repetitively, this subformat is chained to itself a shown in Figure 22.11.

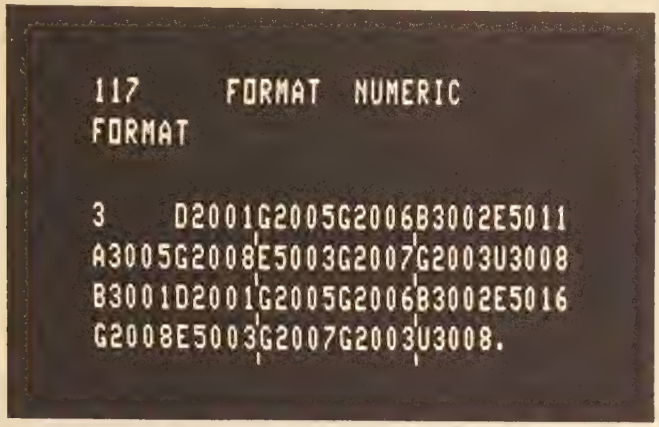


Figure 22.11

When the repetitive data is constantly entered under the same subformat, the Operator will be entering that data form from the source document, until the data is exhausted. Then, the Operator may escape from this subformat by keying the functional key SUB ADV.

After the field descriptors for the final subformat have been entered, the Operator keys an asterisk (*) instead of a period. This signifies to the system that the last subformat has been keyed. Figure 22.12.

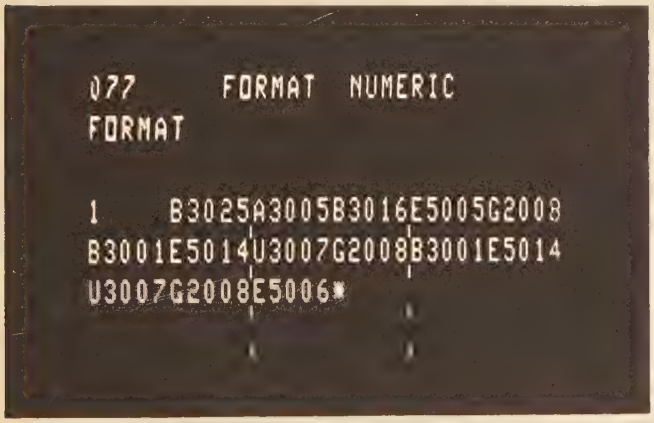


Figure 22.12

The REL key is depressed, and the Operator must verify this subformat, as usual. This procedure is now finished and the screen will display a new message. Figure 22.13.

Figure 22.13

The new format is already stored on disc. It can be used immediately by the same Operator who keyed it and/or simultaneously by any other Operator in the same System. It can be kept temporarily on disc or dumped into magnetic tape. We will assume, for the next discussion, that this format will be used immediately for the procedure of data entry.

START OF BATCH (Procedure 1), Entry

This operation will permit the Operator to enter and/or verify data keyed under a format already on disc. Data entered by one operator under one format, can be almost immediately verified by another Operator under the same format after the first three pages have been key-entered by the first Operator. There are three modes of operation to start a batch:

a Entry (E);
b Verify (V);
c *S*ingle *O*perator *R*esponsibility (SOR).

In the last mode, SOR, Entry and Verification of each page is carried out sequentially by the same Operator. When a single page of data has been key-entered, the Key Display System automatically switches to the Verify mode. The Operator must key verify the page. This page-by-page entry/verify sequence proceeds until the entire batch has been completed.

To perform a START OF BATCH, the Operator keys SEL MODE, as in all cases where a new procedure is being started. The screen displays the Operating procedures shown in Figure 22.5. The Operator selects the digit "1" followed by REL. The following "dialogue" messages will be displayed on the screen, one line at a time. The first line will read

START OF BATCH.....1
FILL IN-

And then the Operator, with the system automatically in alpha shift, will key the proper characters to signify the next operation. At the end of this step, the screen will show the following display (Figure 22.14).

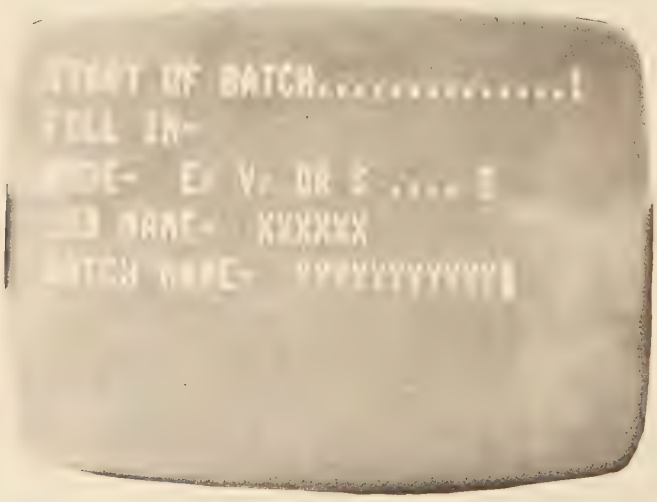


Figure 22.14

The job name is the 6-character label, such as ORDERS, which corresponds to the same format already stored on disc, and will control data Entry. The batch name identifies a group of documents which can be entered under the same format. This batch name consists of a unique group of 12-characters which has not been used previously.

These items of information are called "Header Record Items". At this stage, the screen may appear like this

```
START OF BATCH .....1
FILL IN-
MODE– E, V, or S.....E
JOB  NAME– ORDERS
BATCH NAME– ORDERS 053001
```

Since during this step the keyboard is automatically in alpha shift, the Operator must depress the NUM key in order to enter the digits appearing in the Header Items. When the JOB NAME or BATCH NAME do not fill the 6 or 12 required positions, the remaining positions are filled with spaces (space bar).

In all cases after the Header Record Items have been keyed, the system automatically switches to the Verify mode and the Operator MUST key-verify this information in order to continue with the job.

After the Header Record Items have been verified correct, the system retrieves the format from the disc and a new message appears on the screen

```
001          1    ENTRY      UPPER SH      0001
ORDERS
```

which indicates

001	The cursor is already at home position
1	the batch is under control of subformat 1
ENTRY	The system is in Entry mode
UPPER SH	"Type of Field" indicator. The first field, according to the format, will be an upper shift field.
0001	Number of the page the data will be entered in.
ORDERS	Job name

The system is ready for data entry and the Operator may begin to key the data. As the job progresses, two of the items displayed on the screen i.e. the position number, and the type of field indicator, will change sequentially to reflect the format code. Besides, the cursor itself will begin to move on the screen as each position is keyed-in.

After keying all the data for the first page, the screen may look like this (Figure 22.15).

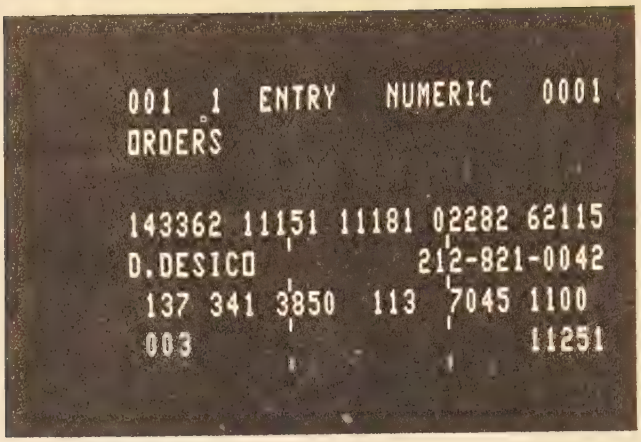


Figure 22.15

The cursor is not visible any longer. Note the position indicator 001: this indicates that the next keystroke will be entered at the first position of the next page. However, the Operator still has access to the page being shown on the screen to make any corrections necessary.

As with other data machines discussed previously, *in all cases* where data does not fill in all spaces assigned to each field, the Operator depresses the SKIP key to fill in the remaining positions. If it is a Numeric field, the LEFT ZERO (LØ) or Minus (–) keys may be used, depending on the type of field.

In all cases, if the data available does not fill 125 columns, keying REL fills the rest of the page with spaces, but the page on the screen does not change until the first keystroke of the next page has been entered. At this time, the screen removes the image of the last record, a tone sounds, the page count is increased by one and the next page will be ready for entry at the first keystroke.

In all cases, during Entry of the first page, fields which are programmed to be duplicated, force their own verification. If, for instance, the Operator is keying a dup field in the first page, when the last character has been keyed in, the cursor returns automatically to the first column of that field, forcing its key verification in order to continue with the job. Once all these dup fields are entered and verified in the first page, this information will be automatically duplicated in the following pages, of the same or different subformats of the same format. Consequently the Operator must code the different subformats so that dup info will be entered in the same fields of the different subformats. Changes in dup data or entering data in fields otherwise formatted for automatic skip, can be performed by the use of the D/S OFF key.

We now assume that Entry of this job is finished. The Operator, in this procedure as well as in others, depressed SEL MODE to signify the end of the Entry. This will cause the screen to display the System Operating Procedures: Figure 22.5.

At this point, the operator can verify her own job entry or start another job. Verification of the same job can be carried out from the same keystation or from another one after the first three pages have been entered initially. In all cases, the data must be verified before it can be dumped onto tape (this can be done from the Supervisor's keystation only). Even before this, an end-of-data procedure must be carried out.

In all cases each batch must be closed immediately after it has been

entered using an END OF BATCH (procedure #2) or an INTERMEDIATE END OF BATCH (procedure #3). When a batch has been closed using the END OF BATCH procedure it may not be re-opened. Careful consideration should be given in these cases to the INTERMEDIATE END OF BATCH, instead. This is done to protect the integrity of the data. We will now assume that the Operator (the same or another) is ready to verify the data entered under a job and batch name.

START OF BATCH (Procedure 1), Verify

This operation will permit the verification of data already entered on disc, under a coded format also on disc. Upon a depression of the SEL MODE key, the System Operating Procedures (Figure 22.5) will be displayed on the screen. The Operator keys 1 and REL and the message

```
START OF BATCH ..... 1
FILL IN–
```

will appear on the screen. The next steps are identical to those described above under START OF BATCH, Entry (Figure 22.14). The only difference being that V will be entered instead of E. Job name and batch name must be the same 6- and 12-characters used in the Entry mode. The screen may appear like this

```
START OF BATCH .....1
FILL IN–
MODE– E, V, or S.....V
JOB NAME – ORDERS
BATCH NAME – ORDERS 053001
```

Since the keyboard is in alpha mode during the entry of the Header Record Items, the NUM key must be depressed to enter the digits.

In all cases when the Header Record Items have been keyed in, the system switches automatically to the Verify mode and the Operator MUST key-verify the data in order to continue with the job.

After the Header Record Items have been verified correct, the system retrieves the format from the disc and a new message appears on the screen.

```
001          1    VERIFY     UPPER SH     0001
ORDERS
```

This, practically, is the same message displayed in the Entry mode, except for the word Verify.

A functional key SCAN permits the Operator the display of the entire page. In this case the position indicator and cursor remain fixed. (The word SCAN is displayed on the Status Conditions on the screen.) Error corrections will be discussed later.

The system is ready for data verification. All data must be verified except in fields programmed for non-verification, which have been so programmed in the format. These fields are automatically skipped.

At the end of the verify operation, the SEL MODE key is depressed to terminate the procedure. And, again, the batch must be closed again with the

END OF BATCH or INTERMEDIATE END OF BATCH procedure. Verification and/or Entry of an entire batch need not be done all at one time; it can be temporarily suspended (break). In all cases, however, the integrity of the data must be protected with the END OF BATCH or INTERMEDIATE END OF BATCH procedure.

During verification of the first page, fields formatted for automatic duplication, force their own verification. If for instance, the Operator is verifying a dup field on the first page, when the last character has been keyed in, the cursor returns automatically to the first column of that field, forcing its key verification in order to continue the job. Once all these dup fields are verified on the first page, this information will be automatically bypassed for verification of the following pages, under the same or different subformats of the same format.

START OF BATCH (Procedure 1) Single Operator Responsibility

It has already been mentioned that the Single Operator Responsibility (SOR) allows the Operator to perform, sequentially, the Entry and Verify functions in one single run. In order to perform this procedure, the SEL mode key is depressed first causing the Systems Operations to be displayed (Figure 22.5) on the screen; the Operator will then depress 1 and REL and the customary fill-in display (Figure 22.14) will be shown on the screen. The Operator will key in the proper data. In this case S (for SOR) is entered at MODE. At the end, the screen may appear like this

```
START OF BATCH .....1
FILL IN–
MODE– E, V, OR S..... S
JOB NAME – ORDERS
BATCH NAME – ORDERS 053001
```

This information must be verified. After the Header Record Items have been verified correct, the system retrieves the format from the disc and a new message appears on the screen

```
001          1    SOR E    UPPER SH'    0001
ORDERS
```

This practically is the same message displayed at the beginning of Entry (Figure 22.6) or Verify modes, except for the code word SOR E.

The system is ready for data entry in the SOR mode. After one page of data has been keyed, the Key Display reverts automatically to the verify mode: the screen will display the word SOR V, while the cursor goes back to position 001. The Operator must verify this data. This entry/verify process is continued until the batch is finished. At this point, the SEL MODE key is depressed and the Operator proceeds to close the batch.

Although this presentation of entering format, data and verification does not follow numerically the order of the Operations Display, it gives a sequential view of how these operations are carried out in the Key Display System. The other procedures are described below.

END OF BATCH (Procedure 2)

INTERMEDIATE END OF BATCH (Procedure 3)

Two similar procedures allow the Operator to close a batch, temporarily or otherwise. A batch closed by Procedure 2, cannot be re-opened. The INTERMEDIATE END OF BATCH procedure (#3) permits the Operator to close temporarily a batch that has not been completed. A batch, closed with procedure 3, can be reopened by using the RESTART BATCH (Procedure 4) described below.

Upon depressing of the SEL MODE key, Operations Display (Figure 22.5) is shown; the Operator then depresses 2 (or 3) and REL. One of the following messages appear on the screen:

END OF BATCH 2
BATCH NAME – ORDERS 053001

or

INTERMEDIATE END OF BATCH 3
BATCH NAME – ORDERS 053001

Upon verification that the batch name is correctly entered, the Operator depresses REL and a new message appears on the screen.

FUNCTION COMPLETED

These two operations can be aborted if SEL MODE is keyed before REL. The INTERMEDIATE END OF BATCH Procedure is very convenient when an Operator temporarily stops an operation to resume it later. It should be pointed out that ORDERS and ORDERS053001 have been used as examples. Job and batch names can be assigned any name.

RESTART BATCH (Procedure 4)

This is the only procedure that permits the resumption of an operation closed by the INTERMEDIATE END OF BATCH procedure.

As usual, after keying SEL MODE, Operations Display (Figure 22.5) will be shown on the screen; the operator depresses 4 and REL. The following "dialogue" message appears on the screen:

RESTART BATCH 4
FILL IN–
MODE – E, V or S
JOB NAME – ORDERS
BATCH NAME – ORDERS 053001

This information must be key-verified (the system reverts automatically to the Verify mode upon entering the info). The Operator fills in the appropriate data. The Mode (E, V or S) remains the one in use when the batch was temporarily closed. The batch may be reopened from the same keystation where it was closed, or from any other in the system.

SEARCH (Procedure 6)

The principle and application of Searches and Identifiers was discussed in Chapter 21. The SEARCH procedure in Key Display Systems is based on the same principle with the advantage that each page can be sight-checked. Any data in temporary storage on disc can be searched. An identifier must be keyed into the same field as on the page being searched for. The cursor forward (——▶) functional key may be used to position the system.

To start a search, the SEL MODE key is depressed, as usual. The System Operations is shown on the display (Figure 22.5). The Operator then keys 6 and REL. The following message is then displayed on the screen:

SEARCH..... 6
BATCH NAME – ORDERS 053001

After the batch name is entered, the system reverts automatically to Verify mode by pressing REL and the batch name must be verified. The Operator keys the identifier, which must also be verified. At this point, though, there is no automatic switch to the Verify mode; consequently, after keying the identifier, REL is depressed and the identifier is verified. REL is keyed again at this stage after the identifier has been verified correct. The use of REL in conjunction with the identifier may be obviated by selecting an identifier which includes column 125 of the page. In this case, it will not be necessary to depress REL after entering and verifying the identifier.

After the identifier has been verified, the system begins the search. If the search is successful the message SEARCH FND is displayed on the status indicators (second row of the display). At the same time, the page counter indicator (first row, right) indicates the page number in reference to the page sequence of the batch. If the search is not successful, the message NOT FOUND is displayed and a new search must be initiated using a different identifier.

The function of the SCAN key has been previously described. When a successful search has been accomplished, a depression of the SCAN key produces a display of the entire contents of that page. At this point, the page can be verified by sight-check.

If errors are noted, that page can be completely deleted by using the PAGE B/DL or HOM keys. The data is then rekeyed in its entirety. The message SCH CORR represents the status conditions on the screen when HOM is keyed. After keying new data, following a search, the system reverts to the Verify mode, and the page must be reverified.

Other pages in the batch, following the first one and carrying the same identifier, can be searched for using the CONT SEARCH key. Pages which follow the first one and may or may not carry the identifier, may be viewed by depressing simultaneously the cursor forward (——▶) and SCAN keys.

RETURN TO LAST MODE (Procedure 7)

Sometimes the Operator depresses accidentally the SEL MODE key. On other occasions, SEL MODE is depressed voluntarily interrupting one particular function. As we know, depressing SEL MODE displays the System Operating Procedures (Figure 22.5) on the screen. However, at

this stage, if the Operator keys 7 and REL, the system returns to the mode it was operating in before. No data is lost and the Operator can resume her activity immediately.

SUPERVISOR (Procedure 8)

This procedure is reserved for the Supervisor.

ERRORS

A number of errors can be committed during the different procedures and in different modes. These errors can be of two types: keying errors and procedural errors (derived from the different functions).

Keying Errors

These errors may result from keying during Entry or Verify modes. Sometimes the operator feels or senses that a keying error has been committed and, therefore, before releasing the data onto the disc, these errors can be corrected. These errors may be one-character long, one-field long or one-page long. If the character just entered is erroneous, the operator depresses CBS/COR: the cursor backspaces over the incorrect character and, in the process, deletes it. When a field is incorrect, FBS/COR is depressed and the cursor goes back to the first position of the field in operation (the cursor backspaces one field per depression; successive depressions of FBS/COR backspaces one field at a time). The field is completely deleted.

When a page must be corrected, in its entirety, HOM is depressed: the cursor is backspaced to position 001 of the current page and deletes it completely (fields formatted for dup and skip are inactivated). *In all cases*, character, field or page, the new correct characters may then be rekeyed. Furthermore, if the page to be corrected is several pages behind, the PAGE B/DL key is depressed. All the pages so backspaced are completely deleted. HOM must be now depressed so as to start rekeying from position 001.

During verification, when errors are detected, the keyboard locks up, the red flashing light comes on and, in some cases, there is an error message on the screen (second row, status indicators). To reactivate the keyboard and kill the light, the operator first depresses the ERR RSET key. Then, depending on the error message, the operator takes subsequent action.

Error Messages

These appear on the second row of the status indicators on the screen. The error light is flashing. The operator keys ERR RSET to turn the light off and reactivate the keyboard. At the same time, the error message is deleted from the screen. A list of the most common error messages and how the data is corrected (these are called "recovery procedures"), is shown below.

Error Message: MISCOMPARE
Mode: Verify
Indicates that the character entered during verification does not match the character entered during Entry.

The MISCOMPARE may have occurred for two reasons: the operator may have depressed the wrong key (while the data on file is correct) or the data on file is incorrect. The same MISCOMPARE message may appear when there are discrepancies not only in characters, but also in fields.

The procedure to follow has been described under Keying Errors. The difference is that, since we are verifying, the data has to be keyed twice: the first time to enter it and the second time to verify the data just entered.

The use of the SCAN key (whereby all the page contents are displayed on the screen) is sometimes recommended to determine if the page is correct, has been duplicated or is missing.

Error Message: DUP ERROR
Mode: Verify
Data newly entered in fields formatted for duplication, require depression of the D/S OFF key before the data is verified. The error light and message are cancelled by keying ERR RSET and D/S OFF.

Error Message: SKIP ERROR
Mode: Verify
Data entered in fields formatted for automatic skip require depression of the D/S OFF key before they are verified. Use the same recovery procedure indicated for DUP ERROR.

Error Message: LEFT ZERO
Mode: Entry
The Operator has forgotten to depress the LEFT ZERO or Minus (—) key.
Keys to be depressed: ERR RST and LEFT ZERO (or minus).

Error Message: LEFT ZERO
Mode: Verify
Same as above.

Error Message: FORMAT
Modes: Entry and Verify
Indicates the presence of a formatted field which is not an available feature in that installation. Check with the Supervisor.

Error Message: NUMERIC
Mode: Entry
Operator may have tried to enter a character other than the 0 through 9 digits, the LEFT ZERO or minus keys.
Keys to be depressed: ERR RSET and the correct digit.

Error Message: INVAL KEY
Modes: Entry and Verify
Operator may have depressed a functional key.
Keys to be depressed: ERR RSET, and the correct character.

Error Message: REL ERROR
Mode: Entry
Operator may have improperly depressed REL (say in the middle of a LØ field).

Keys to be depressed: ERR RSET and the correct character.

Error Message: REL ERROR
Mode: Verify
Same as above.

Error Message: BACKSPACE
Mode: Entry
Operator may have depressed CBS/COR or FBS/COR when the system was at position 001.
Keys to be depressed: ERR RSET and the correct character.

Error Message: NEG FIELD
Mode: Entry
Operator has forgotten to depress the minus key in a formatted negative field.
Keys to be depressed: ERR RSET and minus.

Error Message: NEG FIELD
Mode: Verify
Same as above.

There are some other error messages which are displayed when the operator is performing a certain procedure or at the end of it.

Error Message: BATCH ALREADY ENTERED
Indicates that the selected batch name is already in use and that another batch name must be entered.

Error Message: BATCH NOT IN SYSTEM
Indicates that this batch is not on disc or has already been dumped into tape.

Error Message: BATCH IS NOT DORMANT
Indicates the Operator that there is in progress an active action of dumping the batch on tape.

Error Message: FORMAT NOT IN SYSTEM
Indicates the Operator that job name is not in the system. Either a new format must be entered into the system or the wrong name was keyed.

Error Message: INCORRECT BATCH NAME
Indicates that the batch name does not correspond to the one which is current.

Error Message: VERIFY NEAR
Indicates that the Operator has backspaced to within 9 pages of a formatted verify field.
Keys to be depressed: ERR RSET and REL.

One of the operations which was discussed in detail was entering a format from the keyboard. These error messages are all related to formatting and the operator receives the message with a number. In *all* formatting errors, the

operator must key ERR RSET, SEL MODE and restart the ENTER FORMAT PROCEDURE from the keyboard.

Error Message: FORMAT ERR Ø
There is a job with the same name on disc. Depress ERR RSET and enter new name.

Error Message: FORMAT ERR 1
More than 4 subformats have been defined.

Error Message: FORMAT ERR 2
One of the fields in any subformat exceeds 125.

Error Message: FORMAT ERR 3
One of the fields in any subformat has been entered with a zero length.

Error Message: FORMAT ERR 4
The total number of characters for one subformat exceeds 125.

Error Message: FORMAT ERR 5
Indicates that the total number of fields in a subformat exceed the allowed 64.

Error Message: FORMAT ERR 6
The Operator has programmed a subformat with fields that are all dup or skip, which do not require any entry or verification of data.

Error Message: FORMAT ERR 7
The operator has coded the subformat chaining number incorrectly.

23 Key-to-disc-to-tape: The Inforex (Intelligent Key Entry) System

INTRODUCTION

The principle governing these systems is basically the same as the keytapes described in Chapter 22: the signals generated by the keystrokes are entered directly into a shared processing and control unit (the Control Unit), from where they can be dumped onto magnetic tape. The Control Unit contains control and logic functions, a disc and tape unit and indicators and switches for effective operation. The usual IBM-compatible keyboard, which is part of the keystation, has a few, new functional characters. The TV-like (*C*athode *R*ay *T*ube CRT) display is also part of the keystation. All these systems have been discussed in detail in Chapter 22.

The Control Unit contains the control and logic functions, a disc and tape unit, and indicators and switches for efficient operation. Its primary function is to gather the keyed data from the keystations during data entry and/or data verification, form them into records and place them into an intermediate storage device, the disc. Commands that are initiated at the keystation cause the control unit to transfer the records for a job/batch from disc to magnetic tape. The tape unit is easily accessible for tape loading and unloading. The Control Unit is a separate entity from the keystation.

Any keystation can be used to enter data or to verify data from any data file. Data may be entered in free format or under the control of up to any of 128 predefined programs. Any stored program can be used simultaneously by each of sixteen different operators, or, in turn, each operator can use a different program. One cable, that runs from keystation to keystation, connects all of them to the Control Unit. Each keystation, however, has its own power supply. That is, each can be turned on independently of the others.

On every job performed with the Inforex, there is constant reference to "jobname" and operator ID. These controls guarantee the continuity of data files. Once the disc has been used to gather the data from the entry points, the collected data files, or parts of them, are transferred to magnetic tape. This is usually the final operation at the Inforex end of the system (total computer installation). The tape is then transferred manually to the computer room where it is fed into the daily system for processing.

The beginning of a job entails a "dialogue" between the Operator and the system. Statements are flashed on the CRT screen, with inquiries about the

nature of the job. These inquiries are answered by the Operator with simple alphanumeric characters. The graphics shown on the screen are the regular characters as we know them, not computerese. A list of messages which can be displayed is included in Appendices 4, 5, 6 and 7.

THE KEYSTATION

The keystation comprises the keyboard and the Display Screen.

Figure 23.1 The Inforex System

The Display

The display or CRT (for Cathode Ray Tube) permits the operator to see each character of the record as it is being keyed. An entire 125-character record can be displayed in the form of four rows of 32 characters each. The last three positions of the fourth row are used as character counter. This character counter, in conjunction with a moving indicator, called the cursor, shows the next position to be keyed.

At the end of any keyed record, the character counter returns to 001, and the cursor returns to the first character position of the first row. The

controls for the display are located under the screen of the CRT. The ON-OFF knob, located to the left, is not only the main switch but it also controls the brightness of the display. The knob on the right controls the contrast.

The Keyboard

Figure 23.2 shows the Inforex keyboard and panel of control switches. There are five switches, three shift keys, 15 functional keys (7 single-function and 8 double-function keys) and 35 character keys. The 35 character keys are located in the same positions as in many of the systems previously reviewed, that is, it is IBM-compatible. These character keys enter the same characters as the IBM keyboards. The 35 character keys can generate 26 alpha characters, the 0-9 digits and 26 special characters.

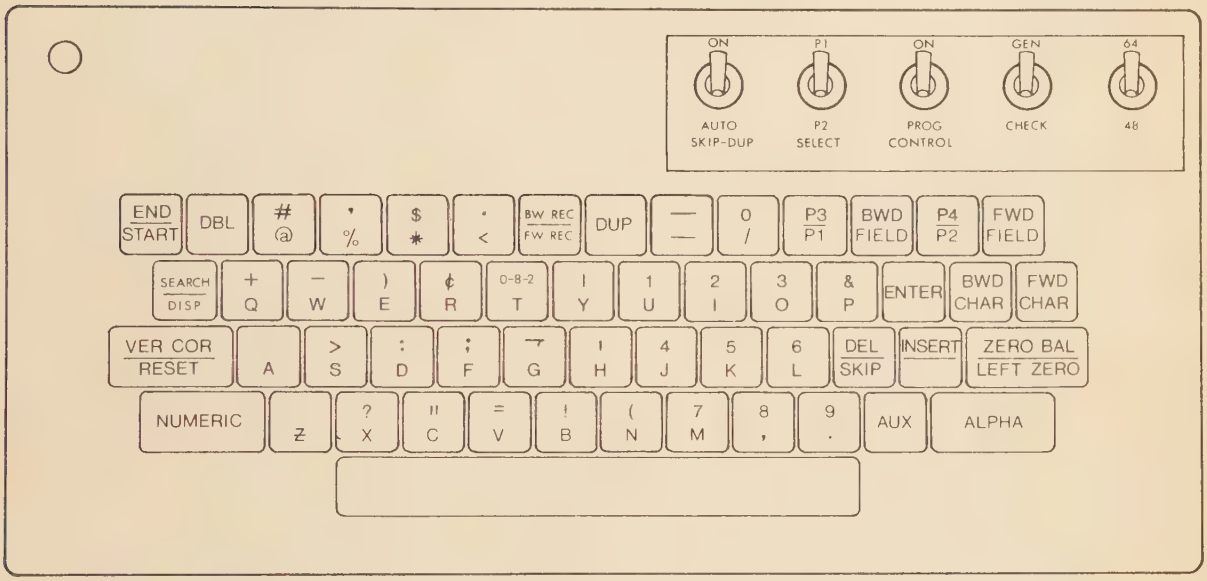


Figure 23.2 Keyboard of the Inforex System

The keyboard can perform as a 64- or 48-character keyboard. The sixteen characters which are non-functional when the keyboard is functioning as a 48-character unit, are all in the upper or numeric shift. These prohibited characters are listed in Table 4.1 (Chapter 4). The 64/48 character switch, on the panel, determines the number of characters available to the operator. Should these keys be accidentally depressed when using the keyboard in numeric shift on the 48-character set, an "INVALID KEY" message will be displayed on the CRT. These messages, occasionally flashed on the screen, can be turned off by depressing the functional key RESET which will, at the same time, clear the accompanying red light on the keyboard.

The characters A and Z when keyed in the 64-character set, regardless of shift, enter A and Z (there was a keyboard lock in the IBM cardpunches). The space bar enters a blank, regardless of shift. As indicated, there also is a red (attention) light on the keyboard. When this light turns ON, it is always accompanied by a message displayed on the screen. The keyboard locks up and the only way to erase the message, unlock the keyboard and turn off the red light, is by depressing the functional key RESET.

Shifts

When the machine is not under program control, the keyboard is in alphabetic shift. The shift, under program control, will be determined by the program control codes. As before, the depression of either shift key (NUMERIC or ALPHA) will override the shift specified in the program. In other words, we have the regular numeric and alphabetic shifts.

The Inforex system, however, offers another special shift, which is directly linked to the many, new functions available with the keytape. The *DBL* (double) shift key can be compared to the multiple punch key since it permits entering multiple data. The DBL shift key on the Inforex Keyboard is used to obtain all of the 256 EBCDIC codes (except X'00'). Appendix 3 shows the character displayed on Hexadecimal keying. The basic communication with IBM computers is in the hexadecimal numerating system. Codes and storage formats for assembler programs are listed in hexadecimal notation. The binary system of computer number systems and conversions has been *E*xtended to *B*inary *C*oded *D*ecimal *I*nterchange *C*ode (EBCDIC).

There is, however, another application of the *DBL* shift key. There are a number of double-functional keys with commands associated with program codes or keyboard activities. One of these double functional keys, for instance, is *END/START*. *START* can be entered directly and is one of the statements that has to be keyed when a job is started. The operator may interrupt a job at any point during data entry or verification by the use of the functional key *END*. To do this, the operator, first depresses *DBL* and simultaneously *END/START*. *END* will be displayed on the screen and the job will be interrupted.

In the same fashion, there are other functions which require the simultaneous depression of *DBL* and other functional keys. These functions will be described later.

Functional Keys

There are 15 functional keys, eight of which are double-function keys. The upper function (above the line on the key) is operational when the *DBL* is held depressed. Of these keys, *NUMERIC, ALPHA, END/START* and *DBL* have been mentioned.

P1/P3; P2/P4
The Inforex system has the option of 4 programs (while on a regulard card-punch the main and alternate were available). These four program levels are called, P1, P2, P3 and P4. This feature allows for a wide program selection. These two double-function keys are associated with the PROGRAM SELECT switch on the panel. This switch controls the original level of program control under which the keystation is functioning. The program level may be changed at will by using either *P1/P3* or *P2/P4* in combination with the DBL shift key. The new selected program level becomes operational at the first position of the next field. These keys override the setting of the PROGRAM SELECT switch at the beginning of each new record. This switch is effective only when the PROGRAM CONTROL switch is ON.

DISP
SEARCH
Display, used in combination with a code, causes the information requested

to be displayed on the screen. The meaning of the Search function in conjunction with identifiers has been explained in Chapters 21 and 22.

VER COR
RESET
VER(ify)*COR*(rect) is used in the correction of a field in the Verify mode. Upon depression, the system changes temporarily to the Entry mode, allowing the Operator the possibility of correcting errors for the duration of a field. RESET turns off the red attention light, clears the message off the display and unlocks the keyboard.

DEL
SKIP
DEL(ete) is used to blank a complete record from the disc in a data set. The entire record to be deleted must be visible on the CRT and must be displayed by the use of a *BW REC* Operation. Skip has the same functions with which the Operator is familiar.

ZERO BAL
LEFT ZERO
ZERO BAL(ance) is used to set the balance accumulator to zero and subtotalling line items in jobs with balance fields. It may be used at any time during data entry. The Left Zero function has the same functions with which the Operator is familiar. LØ fields can be entered unsigned, or signed (positive or negative) and will be discussed under codes.

END
START
This key signals the system to END or START a function. Whenever the Operator starts a job, be it data entry, program coding, etc., the START key must be depressed. A dialogue between the Operator and the systems follows. Let us assume that the Operator will enter program codes at level P1 into the keystation. The START key is depressed. The message START appears on the screen. The Operator keys P1 (which is displayed on the screen). Finally, to signal the end of this function, the Operator keeps depressed the DBL key and presses END/START. The final message on the screen reads

START P1 ENTER

END is depressed to terminate the Data Entry mode. The system is closed to further data input and a display message containing the number of records entered and the balance total will apeear. To enter new data, a new operation must be initiated.

BW REC
FW REC
*B*ack*W*ard and *F*or*W*ard RECord allows the Operator to page backward and forward by record units. *FW REC* moves the cursor to the first position of the next record while the character counter is advanced to 001.

BW REC has the opposite function, moving the cursor and counter to the

first position of the previous record. Both movements are non-destructive, that is, previous or dup info is not erased.

During these operations the AUTO SKIP/DUP switch must be OFF and the DUP key cannot be used.

The seven single-function keys are:

DUP
The *DUP* key, as before, is used to duplicate data from a previous record into the one being keyed. Under program control, duplication continues to the end of the field. With no program control, one character per key depression will be duplicated. This key is non-operational in the *BW REC* (Backward Record) mode.

BWD FIELD; *FWD FIELD*
The *BWD FIELD* (Backward Field) moves the cursor backwards to the beginning of the nearest previous field, *not* erasing the data in it. Simultaneously, the character counter on the screen moves backwards also.

FWD FIELD (Forward Field) has the opposite effect.

ENTER
ENTER is a very important functional key. It is always keyed at the end of a command, on specific keyboard activities. These activities (jobtypes) commence with the depression of *START* followed by the jobtype and the jobname, and terminate with a depression of the functional key *ENTER*. This is an example of a *J*ob *C*ontrol *S*tatement: (JCS):

*START*S, PAYROLL*ENTER*

which indicates to the Control Unit to Store (S) the program (progname) PAYROLL. Note that there are no spaces between START and S; or between PAYROLL and ENTER; note also that a comma separates S from PAYROLL. These are meaningful commands to the Control Unit.

ENTER is also used at the end of a record so the data is transferred to the disc and the display is cleared. It is used, too, in conjunction with Job Control Statements (JCS) and as a release key in certain jobtypes. JCS and jobtypes will be discussed under Operations.

BWD CHAR
*B*ack*W*ar*D CHAR*acter moves the cursor and the character counter backward one column (character) at a time, not erasing the keyed data.

FWD CHAR
*F*or*W*ar*D CHAR*acter has the opposite movement function as BWD CHAR.

AUX
*AUX*iliary causes the data at the Auxiliary level of programming to be duplicated into the record being keyed.

Switches

There are five control switches on the keyboard. Three of them have

identical functions as in the IBM cardpunches and will not be discussed. These switches are the AUTO SKIP/DUP, the PROGRAM SELECT and the PROGRAM CONTROL switches.

The *64/48 CHARACTER* switch has already been mentioned in reference to the number of available characters on the keyboard. In the 64-character position all the character keys are active.

The *GENERATE/CHECK* switch, in CHECK position, calculates a check digit that is compared with the keyed check digit. In GENERATE, the calculated check light is inserted in the last position of the check digit field.

PROGRAMS AND CODES

The preparation and use of programs in conjunction with the Inforex is an activity (jobtype) which is just as easy (or even easier) than with cardpunches. The program is prepared by the operator for immediate use or, if the program exists in the program library (in the Control Unit), it is just as easy to call it from storage into the keystation.

There are four levels of program control (P1, P2, P3 and P4) and one Auxiliary Duplication level resident at the Keystation at any time. Each program level may be of up to 125 positions long (instead of the regular 80) and, as before, the operator can switch (within a record) among different levels by the use of the proper switches and keys.

Each program will have a label or identification "sticker". This will help in its identification and/or retrieval from library for use at the keystation. The program library has a capacity of 128 sets of programs. Each program can be used individually by one operator or simultaneously by all the operators hooked into the same Control Unit. Besides, since the programs can be coded for data entry and data verification, the data entered by one operator at any keystation may, almost immediately, be called by another operator to start its verification.

When a program is created (or generated) they are stored as "temporary residents" of the keystation. If this program will be used often, it is very easy to store it in the library, from which it can just as easily be recalled.

There is a large number of functions which can be carried out by the keytape. Most of them are already known to the Operator. There are only four new functions which have been added, which give the Inforex its present, larger latitude.

The preparation of programs involves concepts with which we are already familiar: there is the high priority code (or punch as it was called in the cardpunches) or field start code, entered at the first position of each field only, and there are the field definition (continuation) codes to be entered in the rest of the field. These field continuation codes not only define the length and shift of a field, but also indicate whether the field is to be verified or not. That is to say, when a program is created the field continuation codes will indicate if these fields will be used just for data entry or data verification. There also is the new concept of field balance for "total" calculations and the incrementation of fields. All of these new concepts are easy to understand. The following table lists all the functions which can be carried out with the keytape, their field start and continuation codes. The symbol ⌀ in this table, stands for blank (space).

Table 23.1

Function	Field start code	Field continuation code
1) Numeric	⌀	&
2) Numeric Verify	⌀	V
3) Alpha	1	A
4) Alpha Verify	1	C
5) Alpha Right Boundary (Right Justify)	2	A
6) Alpha Right Boundary Verify	2	C
7) Left Zero	2	&
8) Left Zero Verify	2	V
9) Auto Dup Numeric	0	&
10) Auto Dup Numeric Verify	0	V
11) Auto Dup Alpha	0	A
12) Auto Dup Alpha Verify	0	C
13) Auto Skip	—	V
14) Balance	2	P
15) Check Digit 10	3	&
16) Check Digit 10 Verify	3	V
17) Check Digit 11	4	&
18) Check Digit 11 Verify	4	V
19) Auto Increment	0	P
20) Auto Auxiliary	4	C
21) Auto Forward Field	3	P
22) End of Record	E	

This last code, E, is used to end the record definition, when the record is less than 125 positions. If, for instance, a record is 95 characters long, the record termination E is entered in position 95. Single position fields can also be prepared by entering the field start code in that single position as was the case in programs prepared with the card punches. All these single position fields, except Auto Forward Field, require verification. We will now review these functions. We will pass briefly on those functions where concepts and applications are the same as for similar functions explained previously.

Numeric, Numeric Verify, Alpha and Alpha Verify Functions

Fields not prepared for verification will automatically be displayed on the screen during verification without any required keying. As before, positions which will not be entered, in every field, can be skipped over by depressing *SKIP* to advance to the next field. The system checks the remaining positions of the field for blanks. If the remaining positions are not blank, the message ERROR will be flashed on the screen (with the accompanying red light indicator). A depression of RESET will clear screen and red light, and will display the character position that is in error; the character counter will indicate the position that is not blank with the cursor beneath it.

Alpha Right Boundary (Right Justify) and Alpha Right Boundary Verify Functions

If the operator enters more alphabetic characters than is allowed (say trying to enter 11 characters for a field of 10 positions) the message FIELD FULL will be displayed and the light turned ON. After depressing *RESET*, the operator can rekey the field by keying *BWD FIELD* and abbreviating the alphabetic data. Or key *SKIP* to terminate the field in the last allowed position.

Left Zero and Left Zero Verify Functions

We have previously dealt with LZ fields. The minimum field length for LZ fields in the Inforex is 2 positions.

A LZ field can be entered signed or unsigned. If a sign is not required, after entering the digit, the operator regularly depresses the *LEFT ZERO* functional key.

On entering or verifying numeric data (not when preparing the programs for LZ fields), if the LZ field is to carry a sign, the *LEFT ZERO* key is not depressed. Instead, after the last digit is keyed, the & (indicating plus or positive) or the — (dash) (indicating minus or negative) keys are depressed. A sign will be placed over the units and all the necessary zeros will be placed to the left of the first significant digit. The LZ field will be entered as a signed LZ field. This means, regardless of what the last digit is, the depression of & or — will automatically make a positive or negative LZ field. The graphic which appears on the screen, however, is an alphabetic character, as per this table.

Table 23.2

Positive numbers	Negative numbers
$\overset{+}{0}$ = ƀ	$\bar{0}$ = &
$\overset{+}{1}$ = A	$\bar{1}$ = J
$\overset{+}{2}$ = B	$\bar{2}$ = K
$\overset{+}{3}$ = C	$\bar{3}$ = L
$\overset{+}{4}$ = D	$\bar{4}$ = M
$\overset{+}{5}$ = E	$\bar{5}$ = N
$\overset{+}{6}$ = F	$\bar{6}$ = O
$\overset{+}{7}$ = G	$\bar{7}$ = P
$\overset{+}{8}$ = H	$\bar{8}$ = Q
$\overset{+}{9}$ = I	$\bar{9}$ = R

Therefore, on entering a positive or negative 001234 the character displayed may be 00123D or 00123M, respectively. A similar situation, for another application, was shown under "Credit" in Chapter 18 and Table 18.4. Consequently, the *LEFT ZERO*, & and — keys can be used to enter data in LZ fields. If, however, the capacity of the field is exceeded prior to depressing

these three keys, the FIELD FULL message will be displayed and the red light will turn on. As explained in the previous function, RESET clears the screen (and the red light) and by keying *BWD FIELD* the operator can start to rekey the same field. If, after depressing RESET, the LEFT ZERO, & or — keys are depressed, the system advances to the next field. On the other hand, if SKIP is depressed as the first key in a LZ field, blanks will be placed in it; if LEFT ZERO is depressed as the first character in a LZ field, the field will be filled with zeros. In both cases, the system will stop at the first position of the next field.

In LZ fields prepared for verification, skipping to the first position of the next field occurs automatically without depression of the LEFT ZERO key, if this is to be an unsigned LZ field. For signed LZ fields, in verification, the last digit is verified by keeping *DBL* depressed and keying, in succession, the sign (& or —) and then the last digit.

Auto Dup Numeric, Auto Dup Numeric Verify, Auto Dup Alpha, Auto Dup Alpha Verify and Auto Skip Functions

In order for the programming codes to be functional in these fields, the *AUTO SKIP/DUP* switch must be ON. Data, within these fields, can be changed. To effect this, the *AUTO SKIP/DUP* must be turned OFF before entering the new data. The new data will automatically be duplicated into the next records when the *AUTO SKIP/DUP* is turned ON again.

Balance Function

This is the first of the new functions available in the Inforex System. The System has the capability of accumulating balance totals as the job/batch is being keyed. At the completion of data entry for a job using a balance total, that total is usually checked against a control supplied in the source document. If the totals agree, there is probably no need to rekey the balance field and Verification can be used to rekey all other verify fields. If, on the other hand, the totals do not agree, there is another jobtype called Balance Verify that can be used to perform the rekeying of all the verify fields *and* the balance field. This case will be discussed later under jobtypes.

The *ZERO BAL/LEFT ZERO* is a double-function key on the Inforex keyboard which is used to clear the balance accumulator, setting it to zero. It may be used at any time during data entry since the source document may provide sub-totals. It is generally used, however, at the beginning of a job. In all cases the cursor must be in the first position of the balance field that is to be made zero.

There is another double function key on the keyboard labelled *DISP* (display). This key is used to question the system as to the status of any particular job. Since this is a double key, the operator has to depress first *DBL* followed by *DISP* when using *DISP*. This action has the net effect of halting *any* keying operation, to blank the screen and freeze the cursor, wherever it may have been when *DISP*(lay) was requested. If, after depressing *DBL* and *DISP* the operator keys B, the screen will display the status of the *B*alance accumulator, and the red light will turn ON. To continue keying, the operator depresses *RESET*. The operator can request information at any time concerning the status of the balance accumulator for that particular job. This balance accumulator can be up to 12 positions

long. If the number is greater than 12 digits, the high order position digits are disregarded and discarded.

On specially equipped units, a second accumulator can be installed. In these cases the field start and continuation codes are 2 and Q, respectively.

Check Digit 10, Check Digit 10 Verify, Check Digit 11 and Check Digit 11 Verify Functions

Under program control, self-checking numbers can be generated during data entry. The use of Modulus 10 or Modulus 11, mentioned in Chapter 15, is available for these check-digit fields. In both cases, the generated digit is associated with the position of the *CHECK/GENERATE* switch on the panel.

With this switch at GENERATE, the system calculates a check digit using both Modulus (the field length must provide one extra position to accept this check digit). On entering, under GENERATE, in these programmed fields, any character key should be entered, since there is one extra position available for entry. The system will display, instead, the computed check digit. On the other hand, with the switch in CHECK, the system calculates the check digit for the digits entered by the operator. On keying the last digit, the system will compare the keyed check digit with the computed one. If these two figures agree, the last keyed character is displayed in the units position. If they do not agree, an ERROR message is displayed. As usual RESET is depressed, whereupon the keyed digit is displayed. The error can then be visually located.

To correct errors in these fields there are two possibilities:

1 the field can be completely rekeyed depressing *BWD FIELD* and re-keying the complete check digit field, or
2 the character can be corrected by using *BWD CHAR* to backspace to the digit position in error, correct it and then depress *FWD FIELD* to advance the system to the next field.

Appendices 1 and 2 show the math used to calculate Modulus 10 or 11.

Auto Increment Function

This programmed field causes a ONE to be automatically added to a field. The switches PROGRAM CONTROL and AUTO SKIP/DUP must be ON for the program codes to be recognized. The counting process continues for every record keyed in a job under program control. Minimum length, in these fields, is two positions.

Auto Auxiliary Function

This type of automatic duplication has the advantage of keeping information that is common to a number of certain records available in auxiliary storage. For instance, on billing, headings like the Company name, address, etc. etc., may be repeated. In these cases the information is entered at the Auxiliary Level at the positions where they will appear in the record. The program codes are recognized when the PROGRAM CONTROL and the AUTO SKIP/DUP switches are ON.

Auto Forward Field Function

This is a special case of Auto Skip. Where Auto Skip enters blanks in the skipped field, blanking existing information, this function skips over the entire field leaving preexisting information intact.

End of Record Function

This function is used to end the record definition when the maximum available positions (125) are not used. In these cases the record termination code E is entered.

OPERATIONS

Jobtypes and Job Control Statements (JCS)

We have reviewed, so far, the system components and the preparation of program levels. In that process, we mentioned that the operator "maintains a dialogue" with the Control Unit using messages displayed on the CRT. These messages indicate, to the operator, the status of their job as well as some other activities. There is still another area which we will discuss now. This forthcoming discussion will enable the Operator to undertake jobs on her own. This area pertains to the control messages the operator has to "send" to the Control Unit to be able to perform any activity (jobtype).

There is a large number of activities which can be carried out with the Inforex System. These activities, which are called jobtypes, fall under the following general categories:

1 jobtypes to generate, store and retrieve programs;
2 jobtypes to control data entry and data verification;
3 jobtypes of a supervisory nature.

In order to have a general overview of the system, we will briefly describe an entry operation. This will be followed by a detailed discussion of the different steps taken. We will assume that (a) a program will be entered from the keystation at the P1 level and (b) it will be followed by data entry under control of the program just entered. In both cases, the operation must be started with the cursor at position 001.

Key	Remarks
1) START	START key.
2) P 1	Depress P and 1. P1 is the jobtype code to enter a program from the keyboard. The "1" indicates that it will be entered at level 1.
3) ENTER	ENTER key
4) Operator keys program codes for the desired jobname "ORDERS"	

5)	ENTER	ENTER key. The program codes are now residents of the keystation and may be used for data entry.
6)	START	START key
7)	D;ORDERS(85)	Note the semicolon.
8)	ENTER	ENTER key. At this point the message PROCEED is displayed on the CRT. Depress RESET and proceed to enter the data.

It may be noticed that the operations, prior to entering data, are short and simple. We will now proceed to describe the jobtype and jobtype codes.

Any activity or jobtype is initiated by the operator keying a *Job Control Statement* (JCS). This statement includes a code that indicates to the system the jobtype to be performed. In these *Job Control Statements* the code is generally (although not always) preceded by the word *START* and terminated with the word *ENTER*. *START* and *ENTER* do not have to be keyed by spelling each word. On the keyboard there are two functional keys, labelled accordingly. To enter these command-words, these functional keys have to be depressed just once.

The code, which varies from jobtype to jobtype, can be alphabetic or numeric, or a mixture of those characters. The JCS must be keyed using a comma (or a semicolon) and parenthesis, when indicated. Spaces (blanks) and/or slashes, in general, are prohibited. If the space bar or any other forbidden character is entered while keying any JCS, an ERROR message will appear on the screen (and the red light will turn ON) when the JCS is finished and entered by depressing *ENTER*. The JCS can then be entered properly, by depressing RESET and rekeying the correct JCS.

Each job activity (or jobtype) may require one program set (with one or more program levels) and an auxiliary (AUX) duplication level. When such a set is stored in library, a "PROGNAME" is assigned to it. This progname may consist of up to eight (alpha or numeric) characters without blanks. (A program directory PROGDIR of the stored programs can be displayed using the adequate JCS.) A large number of activities can be initiated from the keyboard, besides program-related jobtypes.

There actually are two forms of JCS for program-related jobtypes:

D,ORDERS(85)
D;ORDERS(85)

Note the use of a comma or semicolon as a separator. The use of a comma causes the system to automatically refer to the program library and load into keystation the program used for the job name. If a semicolon is used, no reference is made to the program library: the keystation is in free form, using the program in residence at the keystation at that time. The use of the semicolon avoids the erasure of the temporary program in residence.

A selection of keyboard-initiated activities is listed in Appendix 7.

A program can be used for data entry and for data verification only when it is in storage or "resides" at the keystation. If a program has been prepared and is still a keystation resident (it has not yet been sent to the library), it can be used immediately. If such a program (or set of programs) is seldom

used, it may not pay to have it in the program library. In these cases, it is erased after data entry (or data verification). However, if such a program (or set of programs) is used often it may then be stored in the library (Program Library will be referred as PROGLIB).

The most common activities are: the creation of programs, data entry, interrupting data entry mode, continue data entry mode, ending data entry mode, data verification, interrupting data verify mode, continue interrupted verify mode, ending data verify mode, single character correction, field correction in verify mode, delete in verify mode and some supervisory and data modification activities (like file paging, file search, file updating and display functions). Some of these common activities will be described succinctly.

The seven jobtypes and codes used for Data Entry and Verify are:

Job type	Code
Data Entry	D
Data Verify	V
Balance Verify	B
Continue Data Entry	CD
Continue Verify	CV
Continue Balance Verify	CB
Compute Balance	K

Jobtype F (File) is used for paging, search and updating an existing data set. The use of identifiers, called arguments, has been discussed previously. The functional key DIS (Display) can be compared to SCAN in the previous chapter and permits the interrogation of the system. The codes used for Display are:

Operator display functions

DISP 1	Program level 1.
DISP 2	Program level 2.
DISP 3	Program level 3.
DISP 4	Program level 4.
DISP A	Auxiliary programmed data.
DISP R	Data entry or verify record counts.
DISP E	Number of records corrected during verify.
DISP O	(Alpha O) Original record to be verified. In data entry DISP O shows only those columns of the previous record, that were keyed beyond the present position of the cursor.
DISP B	Balance total accumulated.
DISP C	Negative balance total accumulated. (Credit Balance).
DISP L	Program level being used.
DISP J	Last JCS entered at the keystation

As indicated before, a "dialogue" occurs between the Operator and the

system. Some of the messages generated, require a certain "action" from the Operator. The nature of the message will vary depending whether the Operator is in Entry, Verify or Balance modes. A list of messages is shown in Appendices 4, 5 and 6.

To initiate a job in Data Entry Mode, we will assume that the Progname is available on disc.

Key	Remarks
1) START	START key
2) D,ORDERS(85)ANN	Data will be entered in the system under the progname ORDERS. (85) is the batch number and ANN is Operator's identification.
3) ENTER	ENTER key.

The system proceeds to search for the ORDERS program in library, which is loaded into the keystation. If the program is found the message

PROCEED

appears on the CRT. The Operator depresses RESET and starts entering the data.

There may be other messages, besides Proceed. These are listed under Data Entry Messages in Appendix 4. If Proceed is not displayed the Operator depresses RESET and carefully repeats the above operations. If, again, Proceed is not displayed, the Supervisor should be called.

Data Entry may sometimes be interrupted for diverse reasons (coffee break, lunch time, etc.). Interrupting Data Entry is generally performed at the end of one (any) record. At this point, with the cursor at position 001, the following operation takes place:

Key	Remarks
1) I	The alpha character I is depressed followed by
2) END	which is done keeping DBL depressed and depressing END.

The message

INTERRUPTED

is displayed on the CRT, with the balance count total and record count. At this point, the program is returned to the library and is no longer a resident at the keystation. The functional key RESET is depressed.

At the end of the interruption the Operator is ready to resume Data Entry. The operation that follows is to re-initiate the job.

Key	Remarks
1) START	START key
2) CD,ORDERS(85)ANN	CD is the code for Continue Data Entry
3) ENTER	ENTER key

The PROCEED message is displayed, meaning that the same program has been reloaded on the keystation. The Operator re-initiates the keying operation.

At the end of a batch of work, the Operator depresses END (holding DBL depressed). The record count and balance total will be displayed. The Operator then depresses RESET to clear the CRT.

We will now assume that key entry for ORDERS(85) is completed. While this job was in operation, a verifier, from another keystation, could have started the verification. The same program is used by both Operators. To perform verification, the verify Operator proceeds as follows:

Key	Remarks
1) START	START key
2) V,ORDERS(85)MARY	
3) ENTER	ENTER key

The appearance of the PROCEED message on the CRT is an indication that the program is already loaded on the keystation and the verifier is ready to start. There may be other messages. These are listed under Verification Messages (Appendix 5).

Temporary interruption (for a break) in the Verify operation can be effected by depressing END, whereupon the INTERRUPTED message is displayed along with the verify record count, verify error count and the balance total. RESET clears the CRT.

The Verify Operation can be resumed, as described on "Continue Interrupted Data Entry" above. Here, the procedure is as follows:

Key	Remarks
1) START	START key
2) CV,ORDERS(85)MARY	
3) ENTER	ENTER key

Message displayed: PROCEED

When the verifying operator has verified the last record of a batch the END procedure takes place automatically.

On Verification, if the character keyed does not compare with the one originally keyed, the red light comes on and the ERROR message is displayed. The Operator depresses RESET to clear the CRT. Two possibilities exist:

1 the character depressed by the verify operator is the wrong one. On keying the character which agrees with the original one, the verification job is allowed to continue;

2 the character originally entered is in error. The verify operator has three try-outs to enter the correct character. At the third time, the correct character is entered.

Sometimes, complete fields have to be corrected. To effect this, the VER COR key is used. This key temporarily changes the system mode, from Verify to Entry, over the length of the field to be corrected.

Appendix 1

MODULUS 10

Modulus 10 is designed to detect either incorrect keying of a single digit or a single transposition. The following arithmetic process is used to generate the check digit, regardless of the method of generation employed (i.e. generator, computer, or manual):

1 The units position and every alternate position of the basic code number is multiplied by 2.

2 The digits in the product and the digits in the basic code number that are not multiplied by 2 are crossfooted.

3 The crossfooted total is subtracted from the next higher number ending in zero.

4 The difference is the check digit.

Example:

Basic code number	6 1 2 4 8	
Units and every alternate position of basic code number	6 2 8	
Multiply by 2	X2	
Product	1 2 5 6	
Digits not multiplied by 2	1 4	
Cross add	1+2+1+5+4+6	= 19
Next higher number ending in zero		20
Subtract crossfooted total		−19
Check digit		1
Self-checking number	6 1 2 4 8 1	

Other examples:

	Self-Checking Number	
Basic Code Number	Basic Code	Check Digit
45626	45626	9
30759	30759	5
73074	73074	7

Appendix 2

MODULUS 11

Modulus 11 is designed to detect a single digit miskey, single transpositions, and double transpositions. The main feature of this system, distinguishing it from other self-checking number systems, is that it is based on a weighted checking factor for each digit in the basic number being tested. Regardless of how the self-check digit is generated, the following arithmetic process is used:

1 Each digit position of any basic number is assigned a weight (checking factor). These factors are: 2, 3, 4, 5, 6, 7, 2, 3, 4, 5 starting with the units position of the number and progressing toward the high-order digit.

2 In the example shown below, write the number as illustrated, leaving space between the digits.

3 Below each digit, starting at the right and working left, place the corresponding check (weighting) factor.

4 Multiply each digit by its checking factor and add the products.

5 Because this is a Modulus 11 operation, divide the sum of the products by 11, and subtract the remainder from 11. The result is the check digit.

In Modulus 11 operation, basic numbers that require a check digit of 10 cannot be used as self-checking numbers. The accounting system must be adjusted to eliminate such numbers from codes that are to be self-checked. If an operator is generating check digits and keys a basic number requiring a check digit of 10, a message "CK DIT #10" is displayed. *RESET* clears the message. A *BWD FIELD* function must be performed and a new base number, that will not generate a check digit of 10, must be keyed.

When the calculations on the basic number result in a check digit of 11, the digit 0 can be substituted and appended to the basic number to make a valid self-checking number. Eleven and zero have the same value in this part of the calculation.

Basic number: 943457842

Write digits of basic number:	9	4	3	4	5	7	8	4	2

From right to left, write checking factors:	4 3 2 7 6 5 4 3 2
Add the products:	36 + 12 + 6 + 28 + 30 + 35 + 32 + 12 + 4 = Total 195

Divide: 195 divided by 11 = 17 plus a remainder of 8

Subtract: 8 subtracted from 11 = 3 (the check digit)

The new self-checking number is: 9434578423

Appendix 3

CHARACTER DISPLAY FROM HEXADECIMAL KEYING

The chart shows the characters that will appear on the Display when the Double Key (DBL) is used to key hexadecimal data. Two hex characters (0-9, A-F) must be keyed before a character is displayed. The first key (09, A-F) is represented across the top of the chart and the second key on the left of the chart.

To determine the character that is displayed, trace the intersection of the first and second hexadecimal characters keyed. The displayed character is in the intersecting square.

Example: The keying of the hexadecimal E4 displays U.

↓	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	SP	&	-	Ø	SP	&	-	Ø	SP	&	-	Ø	SP	&	-	Ø
1	A	J	/	1	A	J	/	1	A	J	/	1	A	J	/	1
2	B	K	S	2	B	K	S	2	B	K	S	2	B	K	S	2
3	C	L	T	3	C	L	T	3	C	L	T	3	C	L	T	3
4	D	M	U	4	D	M	U	4	D	M	U	4	D	M	U	4
5	E	N	V	5	E	N	V	5	E	N	V	5	E	N	V	5
6	F	O	W	6	F	O	W	6	F	O	W	6	F	O	W	6
7	G	P	X	7	G	P	X	7	G	P	X	7	G	P	X	7
8	H	Q	Y	8	H	Q	Y	8	H	Q	Y	8	H	Q	Y	8
9	I	R	Z	9	I	R	Z	9	I	R	Z	9	I	R	Z	9
A	¢	!	▮	:	¢	!	▮	:	¢	!	▮	:	¢	!	▮	:
B	.	$	,	#	.	$	,	#	.	$	,	#	.	$	,	#
C	<	*	%	@	<	*	%	@	<	*	%	@	<	*	%	@
D	(	)	_	'	(	)	_	'	(	)	_	'	(	)	_	'
E	+	;	>	=	+	;	>	=	+	;	>	=	+	;	>	=
F	¦	¬	?	"	¦	¬	?	"	¦	¬	?	"	¦	¬	?	"

Appendix 4

DATA ENTRY MESSAGES

Message	Cause	Action
"PROCEED"	System is ready to accept data.	*RESET* – begin keying.
"NAME USED"	Jobname and batch number already used.	See supervisor to assign another batch number or use JCS to continue data entry (CD).
"NO JCS"	ENTER depressed prior to depressing START key.	RESET – depress START key and rekey JCS.
"NO JCS END"	1 Data entry is already started and the START key was depressed.	1 RESET – begin keying data.
	2 DBL was not held down during depression of END.	2 RESET – DBL END.
"95% FULL"	No more available disc space.	Notify Supervisor.
"INVALID KEY"	64/48 switch in 48 and a special character was keyed.	RESET – adjust switch if necessary and rekey.
"NO PROGNAME"	There is no program in the system with that name.	Check spelling of progname or notify supervisor.
"STAT NOT I"	"CD" attempted on job that was not interrupted.	Rekey JCS using appropriate statement.
"ERROR"	1 The JCS is incorrect.	1 Be sure all commas are keyed.
	2 A check digit field is incorrect.	2 Key correct check digit number.
"NOT LZERO"	LEFT-ZERO key depressed in a field not designated for Left-Zero.	RESET – use the appropriate key.
"FIELD FULL"	Left-Zero, Balance or Alpha Right Boundary Field contains the maximum number of characters.	In a Left-Zero or Balance Field, depress LEFT-ZERO, Dash or Ampersand key as desired. In an Alpha-Right Boundary field, depress the SKIP key.
"DBL KEY"	DBL key held down and one	Hold DBL key and depress

	character was keyed. System is waiting for second character to be keyed. (Hexadecimal).	the numeric "1". BWD CHAR and rekey correct character.
"NO RECORDS"	CD attempted on a batch that was ended before the first record was entered.	Notify supervisor for cancellation of the batch.

Appendix 5

VERIFICATION MESSAGES

Message	Cause	Action
"PROCEED"	System is ready to verify data.	RESET – begin keying.
"NOT IN JOBFILE"	Specified Jobname (XX) is not in the Jobfile.	Correct missing batch number or wrong batch number or wrong Jobname.
"NO JCS"	ENTER depressed prior to depressing START key.	RESET – depress START key.
"NO JCS END"	1 Verification is already started and the START key was depressed.	1 RESET – begin verifying data.
	2 DBL key was not held down during the depression of the END key.	2 RESET – DBL END.
"NO PROGNAME"	There is no program in the system with that name.	Check spelling of Progname or notify supervisor.
"JOBNAME BUSY"	1 "CV" attempted while specified job is already being verified.	1 Check for correct batch number or see supervisor.
	2 Batch has been verify interrupted.	2 Key START CV instead of START V.
"DBL KEY"	DBL key held downward and only one character was keyed. System is waiting for second character to be keyed (Hexadecimal)	Hold DBL key and depress the number "1". "ERROR" message will appear. RESET and key correct character.
"ERROR"	1 The JCS is incorrect.	1 Be sure all commas are keyed.
	2 A key was depressed that was different from the original character keyed during data entry.	2 Display the original record. If a correction is necessary, refer to page entitled "VERIFICATION CORRECTIONS". If no correction is necessary, RESET – rekey.

"STAT NOT I"	"CV" attempted on job that was not verify interrupted.	Rekey JCS using appropriate statement.
"STAT C"	"V" attempted on job that is completely verified.	Rekey JCS using appropriate statement.
"STAT NOT C"	"RV (RB)" attempted on job that was not completely verified.	Rekey JCS using appropriate statement.
"INVALID KEY"	1 64/48 switch in 48 and a special character was keyed. 2 DBL VER COR used twice in the same field.	1 RESET – adjust switch if necessary and rekey. 2 RESET – key correct data.
"NO RECORDS"	"B" attempted on a job that contains balance fields that was never interrupted.	Notify supervisor to have data entry operator momentarily interrupt the job.

Appendix 6

BALANCE MESSAGES

Message	Cause	Action
"FIELD FULL"	Too many digits were keyed in the balance field.	RESET. Depress LEFT ZERO and (if amount is wrong) BWD FIELD and DBL VER COR and rekey field.
"INVALID KEY"	1 FWD CHAR or FWD FIELD key was depressed. 2 VER COR key was depressed twice in a row without the use of the LEFT ZERO key.	1 RESET and proceed with appropriate key. 2 RESET and proceed with appropriate key.

Appendix 7

JOBTYPES AND JOB CONTROL STATEMENTS (JCS) IN THE INFOREX SYSTEM

Jobtype	JCS
1 To define Program control levels.	*START*P1*ENTER* *START*P2*ENTER* *START*P3*ENTER* *START*P4*ENTER* *START*A*ENTER*
Following each START JCS, the program control should be keyed. If the program control is less than 125 characters, *ENTER* must be pressed before proceeding.	
2 To store Program Controls in ProgDir The progname must not be longer than 8 alphanumeric characters with no blanks.	*START* S,ORDERS *ENTER*
3 To retrieve Program Controls from ProgDir	*START* P,ORDERS *ENTER*
4 To begin Data Entry	*START* D,ORDERS(85) *ENTER*
5 To interrupt Data Entry	I *END*
6 To end Data Entry	*END*
7 To continue Data Entry	*START* CD,ORDERS(85) *ENTER*
8 To begin Verification	*START* V,ORDERS(85) *ENTER*
9 To interrupt Verification	*END*
10 To end Verification	The system will automatically END the job upon completing the verification of the last record entered.
11 To begin Continue Verification	*START* CV,ORDERS(85) *ENTER*
12 To interrupt Continue Verification	*END*
13 To end Continue Verification	See JCS #10

14	To begin Balance Verification	*START* B,ORDERS(85) *ENTER*
15	To interrupt Balance Verification	*END*
16	To end Balance Verification	See JCS #10
17	To continue Balance Verification	*START* CB,ORDERS(85) *ENTER*
18	To recompute the Balance Total	*START* K,ORDERS *ENTER*
19	To end Balance Recompute	*END*
20	To display the complement of a negative Balance Total	*DISP* C
21	To begin Paging	*START* F,ORDERS *ENTER*
22	To begin a File Search	*START* F,ORDERS(85) *ENTER*
23	To specify the Search Argument	*SEARCH* ARGUMENT *ENTER*
24	To continue Search for the next or preceding record	*FW REC* *BW REC*
25	For another record with same argument	*SEARCH* FW REC
26	To end File Search	*END*
27	To Display	
	Program Controls	*DISP* X (X is 1,2,3 or 4)
	Auxiliary Dup Data	*DISP* A
	The Balance Total	*DISP* B
	Complement of a negative Balance Total	*DISP* C :
	The Original Entry	*DISP* O
	The Record Count	*DISP* R
	The Error Count	*DISP* E
28	To page the Jobfile	*START* F;JOBFILE *ENTER*
29	To search the Jobfile	*START* F;JOBFILE,99 *ENTER*
30	To page the ProgDir	*START* F;PROGDIR *ENTER*

Answers to selected problems

CHAPTER 1

1 To observe if the cards are properly placed.
2 80
3 12
4 Zone punches
5 Digit punches
6 X
7 1
8 2
9 2 and/or 3
10 12-4
11 11-6
12 0-2

CHAPTER 2

1 The punching station is where the actual card perforation takes place. The reading station is where these punched cards are "read" in order to duplicate this information into the card at the punching station.
2 Facing the operator in the hopper, with its back in the stacker.
4 500-500
5 2- Keying FEED twice or keying FEED and REG alternately.
6 Hopper, card bed, punching station, reading station, stacker.

CHAPTER 4

3 Printing
5 Keying more than one digit per column.
8 Depressing the Space bar.
13 To engage the reading mechanism.

CHAPTER 5

2 11-12
4 0-12
6 /-A
8 Master Card

CHAPTER 6

1 6
3 0
5 6
7 6-7
9 12

CHAPTER 7

1 Keep the ALPHA key depressed while keying the alphabetic characters.
2 Keep NUM (NUMERIC) depressed while keying the digits.
3 5-4-

Glossary

ALPH (ALPHA) Functional key which shifts the cardpunch to alpha shift.

ALPHABETIC CHARACTERS Graphic representation of characters which require two punches per column.

ALPHABETIC FIELDS Fields where only alpha characters are entered.

ALPHABETIC FIELDS, DUPLICATED Fields which, under program control, duplicate alpha information from card to card.

ALPHABETIC KEYS Keys used to punch alpha characters, which require two punches per column.

ALPHABETIC SHIFT Punches in the program card to key alpha information under program control.

ALPHAMERIC Contraction of alphabetic and numeric.

ALPHANUMERIC Characters which can be alphabetic, digits or special symbols.

ALTERNATE LEVEL Codes in the program card to alternate programs.

ALTERNATE PROGRAM (card) Program card codes with digits 4, 5, 6, 7, 8 and 9.

ANALOG Data presented in continuously variable quantities. Opposite to digital.

ANALOG COMPUTER A computer which receives data in analog form and produces an output in analog form.

AUTO FEED Functional switch that allows for the automatic card feeding.

AUTOMATIC DUPLICATION Duplication of information carried out under program control.

AUTO SKIP/DUP Functional switch which, under program control, allows or suppresses the automatic skipping or duplication.

BACKSPACE KEY Key used to backspace either one column at a time or continuously.

BINARY A digit expressed using the digits zero (0) and one (1).

BIT A *bi*nary digi*t*: an abbreviation of the italic letters.

BLANK A column or position without data.

BLOCK A group of computer words.

BYTE A sequence of two adjacent bits.

CAPACITY A number of cards that can be placed in the hopper and stacker in a card-punch.

CARD Piece of cardboard of special thickness, size and shape where coded information is entered.

CARD BED The three machine positions through which each card passes in its way from the hopper to the stacker.

CARD FIELD A set of columns where the same information is entered.

CARDPUNCH Machine that punches holes on cards in a logical fashion.

CHAIN Segments linked in tandem.

CHARACTER The letters A to Z, the digits 0 to 9 and special symbols accepted by the computer.

CHECK DIGIT Digits, often called self-checking numbers, used for checking purposes (Modulus 10 and Modulus 11).

CLEAR Spring-loaded functional switch used to clear all cards from the card bed.

COBOL Acronym for COmmon Business Oriented Language.

CODE Sets of rules to convert data from one form or representation to another.

CODE, BINARY Code where the ten decimal digits are represented by a combination of bits.

COLUMNS Number of places where information can be keyed in. Synonym with position.

COLUMN INDICATOR Metal pointer, at the base of the program drum (generally) which points to the next column to be keyed.

COMBINED KEYS Keys with a double representation for alpha, numeric and special characters.

COMPUTER An electronic system capable of accepting information in coded form (input), performing required mathematical and logical processes, and supplying the results in written form (output).

COMPUTER CHARACTERS Special characters entered in alpha and numeric fields which are used in computer programming.

CONTROL (PANEL) Electronic part of any hardware system, where functions may be selected, status determined and switched.

CONSOLE Control panel.

CRT Cathode Ray Tube, an electronic tube with a TV-like screen, a picture tube.

DATA A collection of letters, numbers and symbols used to describe an idea.

DIGIT PUNCHES Perforation of digits which require one punch per column.

DISC Rotating device with a magnetized surface where information is recorded as magnetic pulses.

DOCUMENT Representation of information on paper, in a form which is readable by Operators.

DRUM Device where the program card is mounted.

DUP Functional key which allows the duplication of information.

EDP Electronic Data Processing, hardware devoted to the automatic handling of a variety of information.

EOF End-Of-File, point of completion of a group of data on tape.

ERROR RESET Functional key which allows the clearout of stored digits in L-Z Fields and to unlock keyboards.

FEED Functional key which allows the feeding down of cards from the hopper.

FEEDING, AUTOMATIC Switch which allows the automatic feeding from the hopper, without any intervention from the Operator.

FEEDING, MANUAL Operations by means of which cards are fed down from the hopper, by depressions of the FEED key.

FERROMAGNETIC Property of certain minerals of becoming polarized under magnetic fields.

FIELD Number of columns (or positions) assigned to a specific function.

FIELD LENGTH Number of columns or positions where certain information is entered.

FIELD DEFINITION PUNCHES Program card punches which limit the length of each field.

FORMAT Arrangement of information in fields.

FORTRAN FORmula TRANslation, a programming language.

FUNCTIONAL CONTROL SWITCHES Switches located above the keyboard, used in numerous functions.

FUNCTIONAL KEYS Non-punching keys with special functions.

FUSES Protection devices placed underneath the desk top (generally) which allow for electric flow.

HARDWARE Electronic equipment or devices, physical parts of a computer and other peripheral equipment. Opposite to software.

HEAD, READ-WRITE A small magnet used to read, record or erase information on tape, as polarized spots.

HIGH-PRIORITY PUNCH Perforation on the first column of each programmed field in the program card.

HOLLERITH CODE Combination of holes to represent alphabetic, numeric and special characters.

ILLEGAL CHARACTER Combination of holes or representation not accepted nor readable by hardware.

INPUT Data to be entered and handled by hardware.

I/O Abbreviation for Input/Output.

KEY Labelled lever on the keyboard.

KEYBOARD ENTRY Element of data entered via the keyboard.

KEYBOARD LOCKING Condition resulting from keying forbidden characters.

KEYBOARD VERIFY Elements of data verified via the keyboard.

KEYBOARDS Different combinations of keys for multiple data-entering functions.

LANGUAGE System to represent information and present data in a form intelligible to EDP machines and humans.

LEFT JUSTIFICATION In alphabetic fields, keying alphabetic characters so that the first letter falls in the first column of those assigned to this field.

LIBRARY Collection of data and/or programs available to the Operator. Generally on drum or magnetic tape.

L-Z Feature available in certain machines, which allows the insertion of zeros before the first significant digit of a numeric field.

L-Z CODE Punches in the program card to key numeric information so that it is automatically preceded with zeros.

L-Z FIELDS Numeric fields which, under program control, insert zeros before the first significant digit.

L-Z PRINT Functional switch which permits or suspends the printing of zeros on card, before the first significant digit. Printing is allowed only when the PRINT switch is ON.

MAGNETIC CORE A magnetized device to store data.

MAGNETIC DRUM A magnetized cylinder to store data.

MAIN LINE SWITCH Device that allows the flow of electric current. Generally located near the desk top.

MANUAL CARD INSERTION Insertion, by hand, of individual cards, in the punching and/or reading stations.

MANUAL DUPLICATION Duplication carried out by manually depressing the DUP key.

MARK, TAPE Magnetic character entered on tape at the end of a block of data.

MASTER CARD The first card of every punching job.

MEMORY See Storage.

MULT PCH Functional key which permits entering more than one digit per column.

NORMAL LEVEL Codes in the program card for normal programs.

NORMAL PROGRAM Program card coded in positions 12, 11 and 0 and in digits 1, 2 and 3.

NUM (NUMERIC) Functional key which shifts the cardpunch from alphabetic to numeric shift.

NUMERIC FIELDS Fields where numeric or special (but not alphabetic) characters are entered.

NUMERIC FIELDS, DUPLICATED Fields which, under program control, duplicate numeric information from card to card.

NUMERIC KEYS Keys used to punch numeric characters with one punch per column.

NUMERIC SHIFT Provision of the program card to key numeric information under program control.

ONE TWO PROG SEL Functional switch which, in combination with the program card, allows for program selection.

OUTPUT Data transferred from the internal storage to the outside of the hardware.

OVERPUNCH Add more punches per column than allowed. Add extra columns to fields of known length.

PARITY CHECK A check, assigned by hardware manufacturers, whereby the addition of magnetized spots should always be odd or even.

PRINT Functional switch which permits or suppresses printing on the card's upper edge. This switch has a higher priority than the L-Z PRINT switch.

PRINTING Graphic characters printed under the card's upper edge, in machines with printing mechanism.

PROGRAM Method to solve a problem, using computer languages and procedures.

PROGRAM CARD Coded card for the execution of certain punching functions.

PROGRAM CONTROL When the hardware system is operating under a coded program.

PROGRAM CONTROL LEVER Mechanism which controls the operation of the program unit by lowering or raising the sensing mechanism.

PROGRAM DRUM Round metal support of the program card.

PROGRAM UNIT System in the cardpunch which controls a number of automatic functions.

PUNCH POSITIONS Digits 1 through 9 on the card.

PUNCHING STATION Position at the card bed where the actual perforation takes place.

READING STATION Position at the card bed where cards are "read" in order to duplicate their information.

RECORD Data or groups of related information placed in storage.

REG Functional key used to register cards at the punching station. It is also used to engage the sensing mechanism of the program unit.

REL Functional key used for the release of a card with automatic duplication.

RIGHT JUSTIFICATION In numeric fields, keying digits so that the last one falls on the last column assigned to that numeric field.

ROWS One of the 12 horizontal lines in a card, each with 80 possible punching positions.

SEARCH Procedure to examine stored data on records, to select one particular item.

SHIFT Level at which data is entered in EDP machines: alphabetic, numeric.

SKIP Functional key used for the manual column spacing.

SOFTWARE All the programs used to enter data in computers. Opposite to hardware.

SPACE BAR Can be pressed at any time to blank one column.

SPECIAL CHARACTERS Graphic representation of special characters which require up to three perforations per column. Some special characters are used for computer programming.

STACKER Holder of keyed cards stacked in the original punching sequence.

STORAGE Device where data is entered or from where it is recalled. Generally magnetic. Synonymous with memory.

STROKE Act of depressing a key.

SWITCH Manually operated electric switch which may be in either two positions: "ON" and "OFF".

TAB Keyboard key of certain shape.

TAPE A strip of magnetized paper used to input data.

VERIFIERS Special machines used to check the accuracy of entered information.

X-PUNCH Punch at the 11-position.

ZONE PUNCH Positions 12, 11 and 0, on the card.

Index

A complete and concise manual for students about to study electronic data processing. It is, in fact, the latest manual on keypunch, keytape and keydisc, and it includes IBM's 129 Card Data Recorder, 50 Keytape, and the 5496 Data Recorder for small installations.

Machines, components and program codes are shown in detail, together with figures, keyboards and switches. The authors describe several applications of keypunch, keytape and keydisc principles, and teach the student to create programs and to know the meaning of the job being done.

The simple format of the book will allow students to follow other reference manuals without difficulty.

Recommended further reading

INTRODUCTION TO DIGITAL COMPUTER PLOTTING
Edited by T.C. Smith and Y.C. Pao

STUDY AND COMPILATION OF COMPUTER LANGUAGES
By Y.A. Wallach

COPING WITH INCREASING COMPLEXITY: Implications of General Semantics and General Systems Theory
Edited by Donald E. Washburn and Dennis R. Smith

AUTOMATED DESIGN OF CONTROL SYSTEMS
By C.W. Merriam III

ANALYSIS AND DESIGN OF DIGITAL SYSTEMS
By Vasil Uzunoglu

INTRODUCTION TO LOGIC AND SWITCHING THEORY
By Nripendra N. Biswas

0 677 03950 6

Gordon and Breach
One Park Avenue, New York, NY 10016
42 William IV Street, London WC2
7-9 rue Emile Dubois, Paris 75014