

Midterm #1

- Test corrections due Tuesday
 - MC: in your own words, explain the correct answer
 - Essay: if 21 or below (B), please rewrite your essay

Four assignments

- History of Computing reading; Chapter 3
- Midterm corrections; History of Computing essays

History of computation, pt. 1

Calculators; ancient and semi-modern computing

Abacus (ancient)



Antikythera mechanism



Mid 1800's — mid 1900's

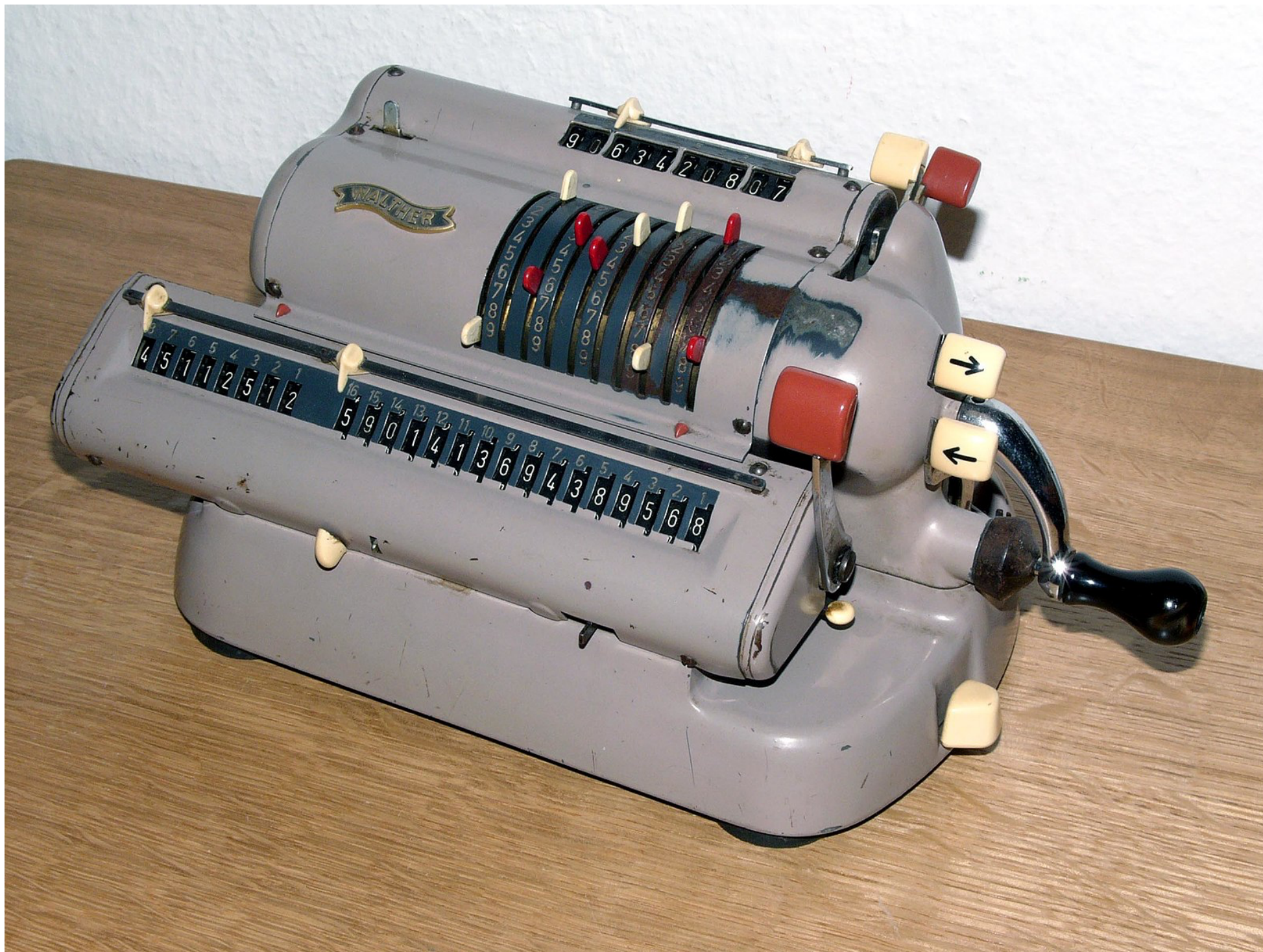
Slide rule



Mechanical calculators (Odhner ~1900)



Mechanical calculator (Walther 1960)



Curta calculator (1947-1972)

- Developed in concentration camp
- Addition, subtraction, multiplication, division



Language

- Computer used to mean the person

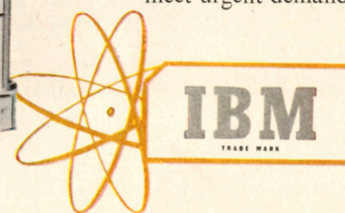
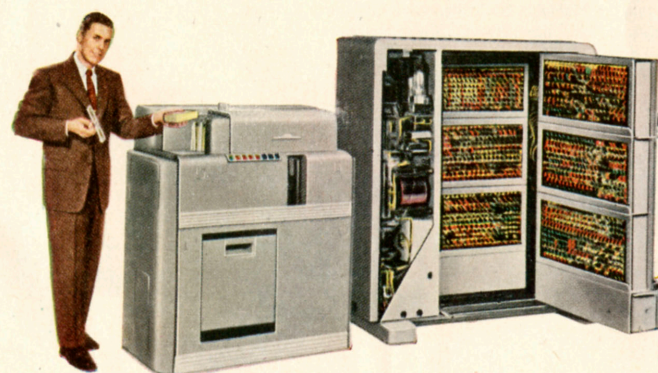


150 Extra Engineers

An IBM Electronic Calculator speeds through thousands of intricate computations so quickly that on many complex problems it's like having 150 EXTRA Engineers.

No longer must valuable engineering personnel . . . now in critical shortage . . . spend priceless creative time at routine repetitive figuring.

Thousands of IBM Electronic Business Machines . . . vital to our nation's defense . . . are at work for science, industry, and the armed forces, in laboratories, factories, and offices, helping to meet urgent demands for greater production.

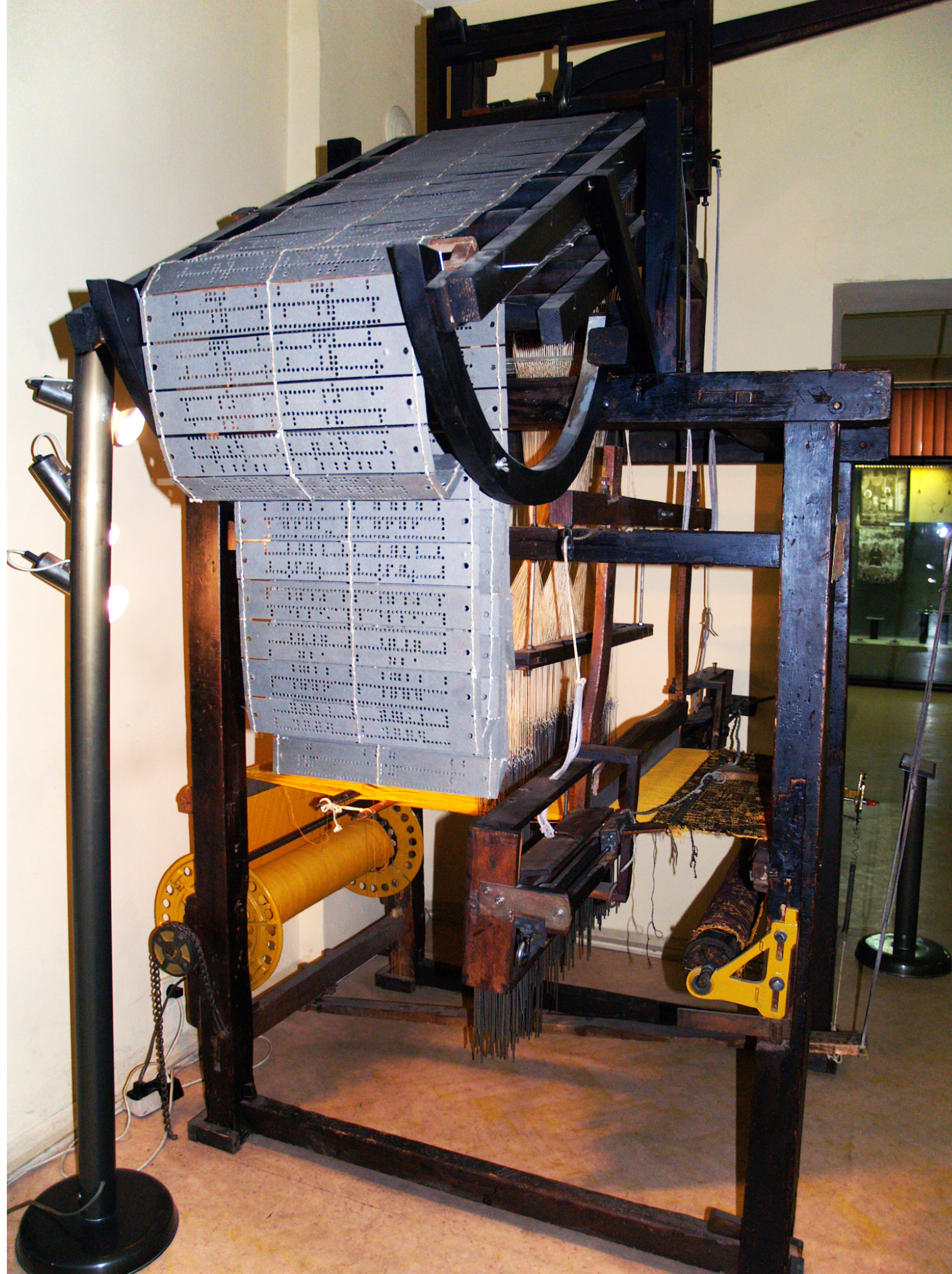


INTERNATIONAL BUSINESS MACHINES

Data

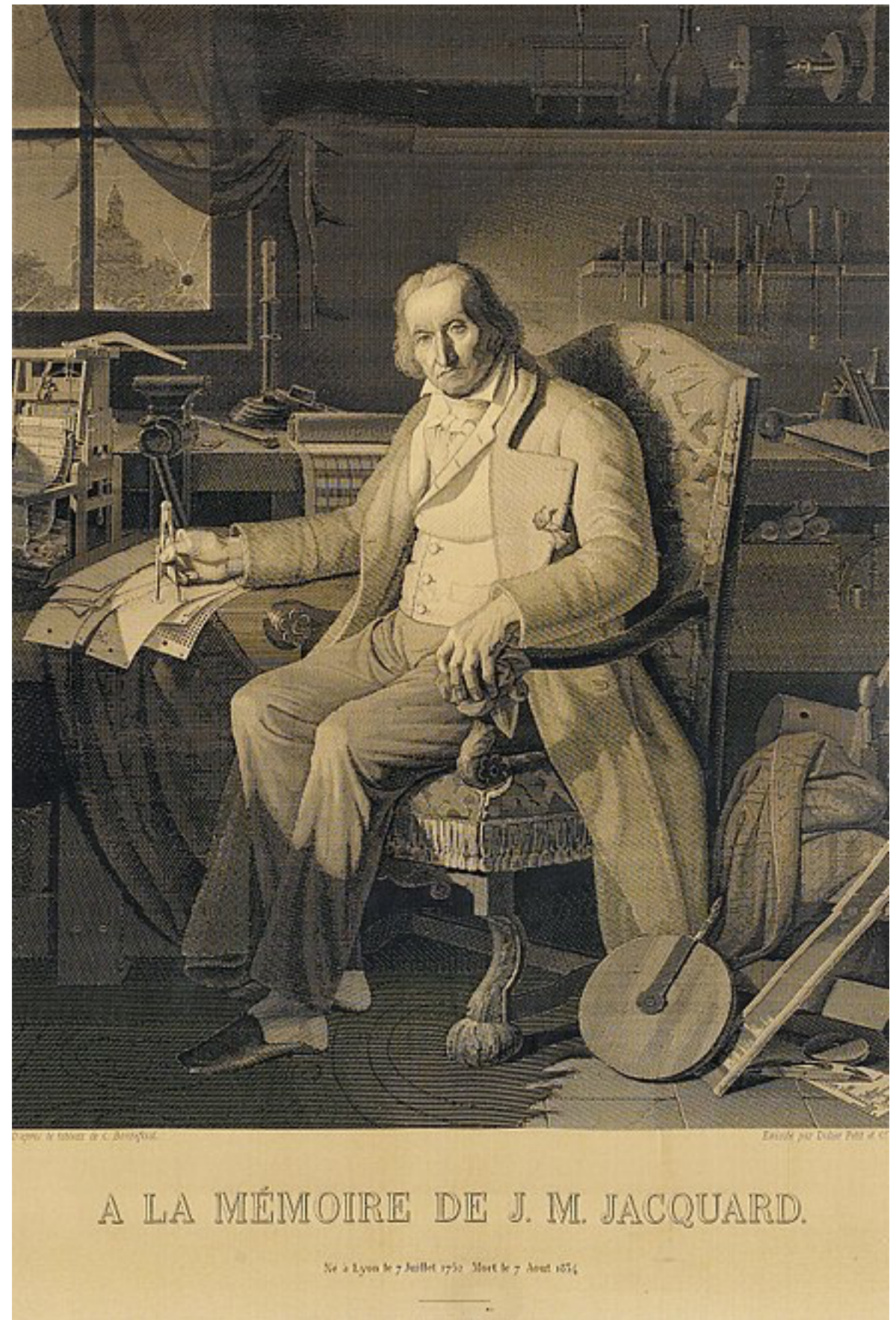
Jacquard loom 1804



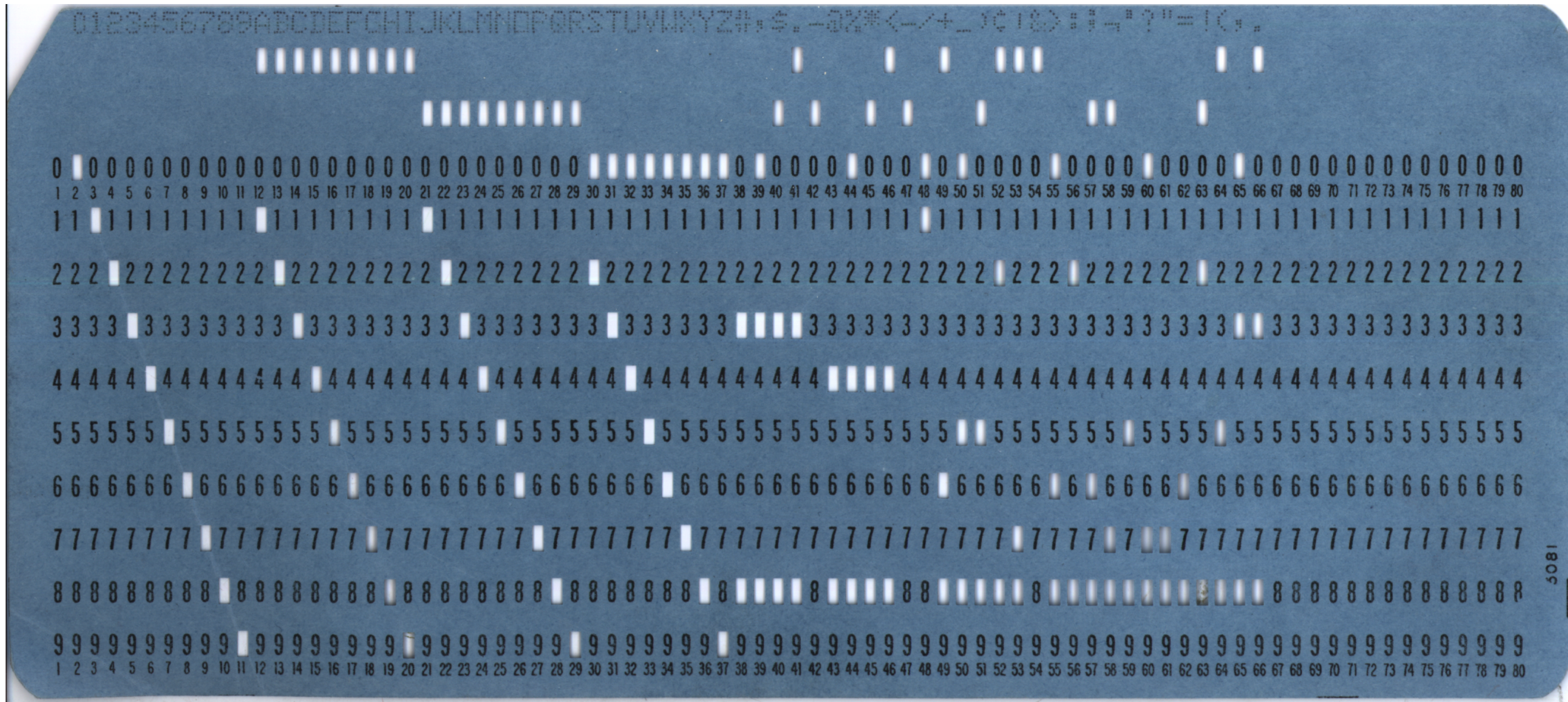


Silk weaving

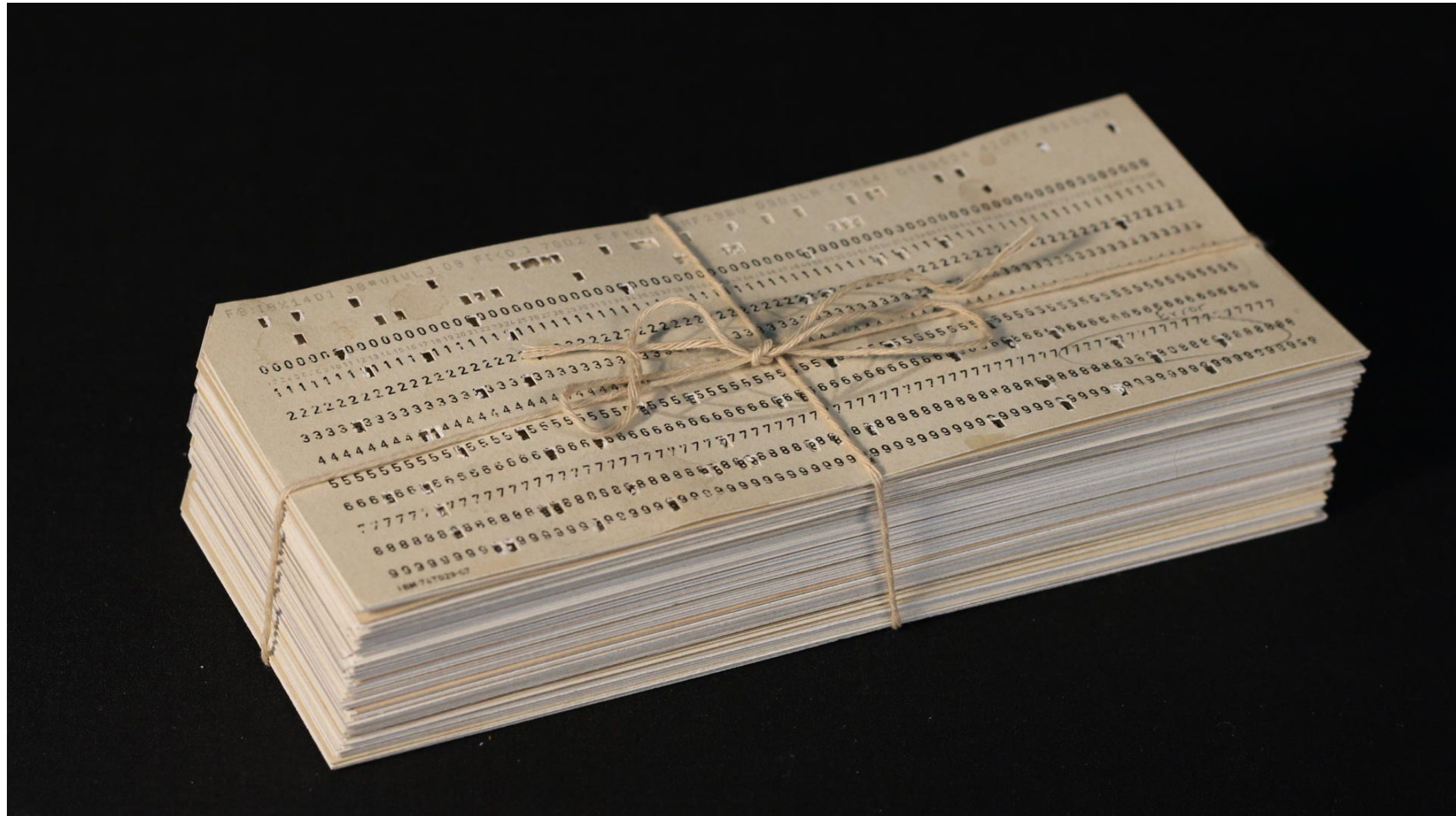
- 24,000 cards!



Punch cards



Data



Data



Data

NARA-A Brief History



Storage of IBM record cards at the Federal records center in Alexandria, Virginia, November 1959.

Post-war computing

My family and computers

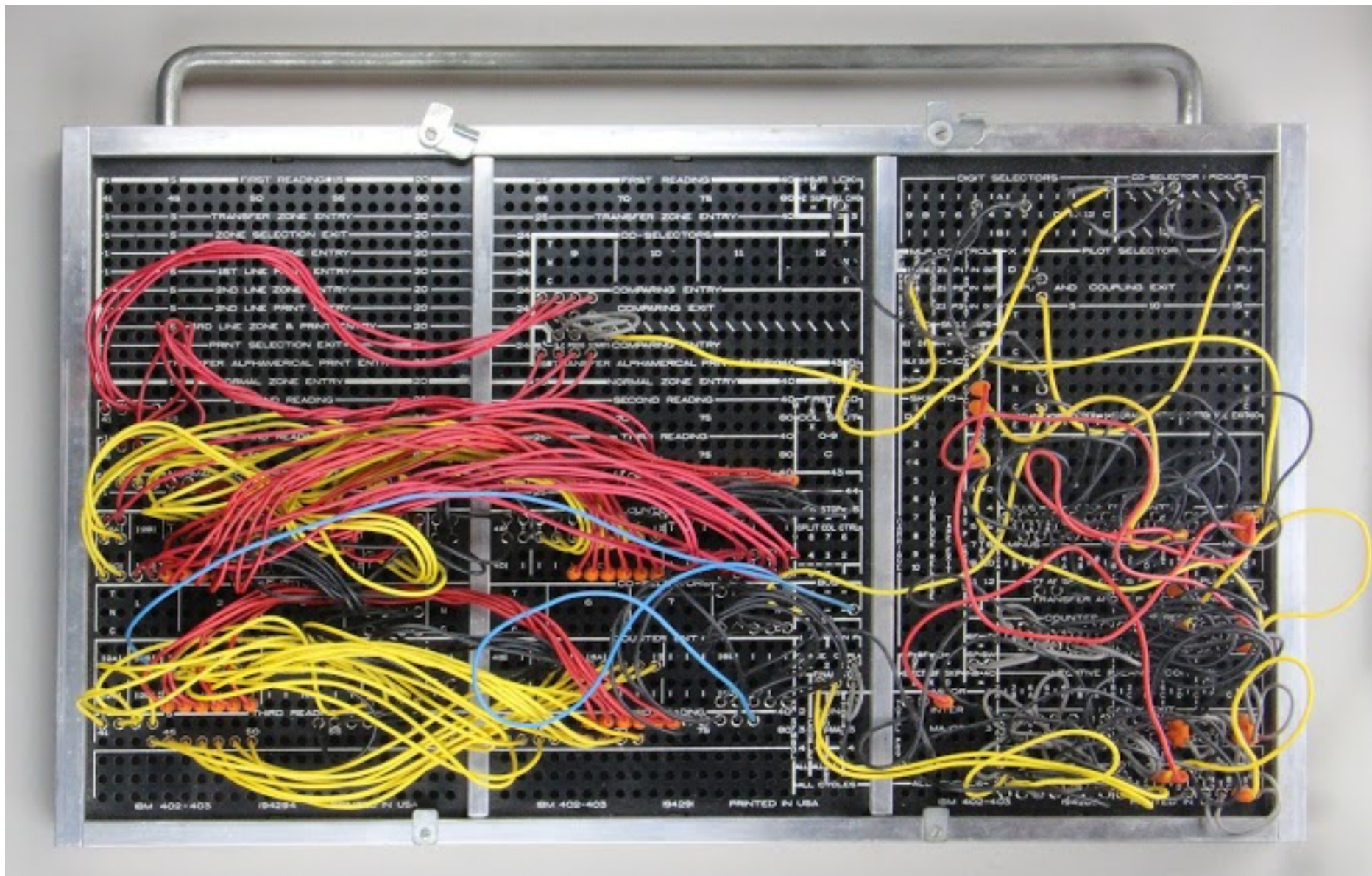


Pete Morales



Barbara Hosier

- Data on punch cards
- Program hard wired



IBM 403 & 403



'Software'



Kelley Air Force Base (IBM 402 & 403)

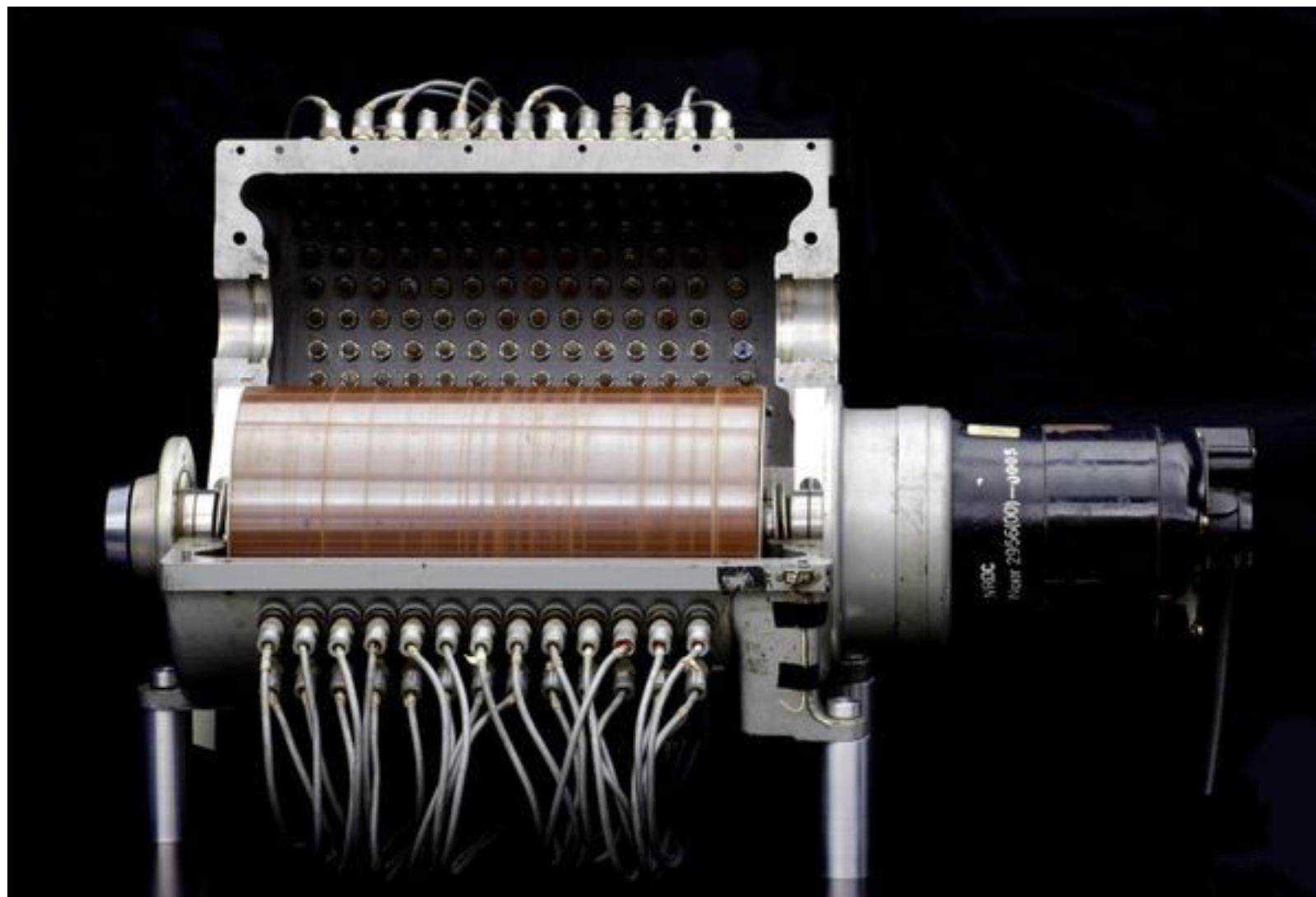


IBM 650



Magnetic drum (IBM 650)

- Program stored on drum (along with memory)
- Data on punch cards or magnetic tape



Grace Hopper and COBOL



Machine language

; Main program

```
-----
Start
; **** YOUR CODE GOES HERE ****

      MOVLW      b'00001111'
      MOVWF      H'20'
      CLRF       H'21'
      MOVLW      b'11001100'
      MOVWF      H'21'
LoopPoint RRF      H'20',F
      INCF       H'21',F
      GOTO       LoopPoint

Finish
      END                ; End of program
.
```

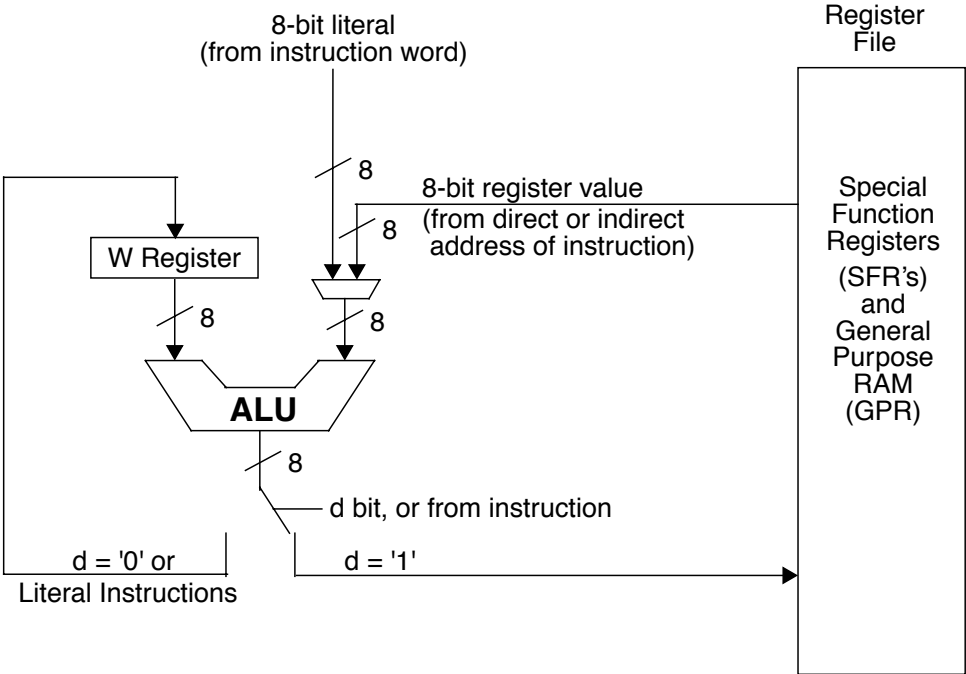
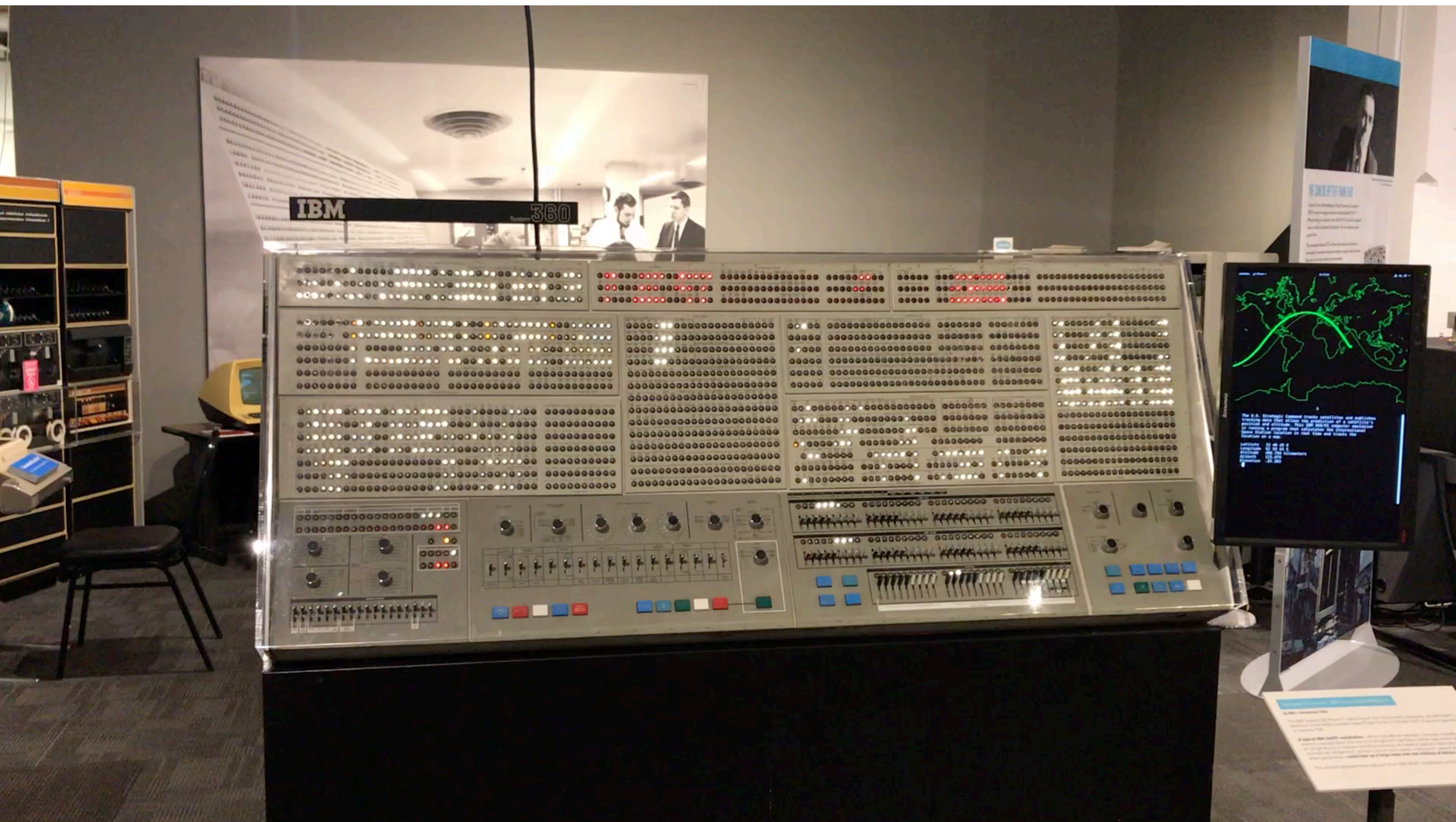


TABLE 16-2: PIC16F87/88 INSTRUCTION SET

Mnemonic, Operands		Description	Cycles	14-Bit Opcode				Status Affected	Notes
				MSb		LSb			
BYTE-ORIENTED FILE REGISTER OPERATIONS									
ADDWF	f, d	Add W and f	1	00	0111	dfff	ffff	C,DC,Z	1,2
ANDWF	f, d	AND W with f	1	00	0101	dfff	ffff	Z	1,2
CLRF	f	Clear f	1	00	0001	1fff	ffff	Z	2
CLRW	-	Clear W	1	00	0001	0xxx	xxxx	Z	
COMF	f, d	Complement f	1	00	1001	dfff	ffff	Z	1,2
DECf	f, d	Decrement f	1	00	0011	dfff	ffff	Z	1,2
DECFSZ	f, d	Decrement f, Skip if 0	1(2)	00	1011	dfff	ffff		1,2,3
INCF	f, d	Increment f	1	00	1010	dfff	ffff	Z	1,2
INCFSZ	f, d	Increment f, Skip if 0	1(2)	00	1111	dfff	ffff		1,2,3
IORWF	f, d	Inclusive OR W with f	1	00	0100	dfff	ffff	Z	1,2
MOVF	f, d	Move f	1	00	1000	dfff	ffff	Z	1,2
MOVWF	f	Move W to f	1	00	0000	1fff	ffff		
NOP	-	No Operation	1	00	0000	0xx0	0000		
RLF	f, d	Rotate Left f through Carry	1	00	1101	dfff	ffff	C	1,2
RRF	f, d	Rotate Right f through Carry	1	00	1100	dfff	ffff	C	1,2
SUBWF	f, d	Subtract W from f	1	00	0010	dfff	ffff	C,DC,Z	1,2
SWAPF	f, d	Swap nibbles in f	1	00	1110	dfff	ffff		1,2
XORWF	f, d	Exclusive OR W with f	1	00	0110	dfff	ffff	Z	1,2
BIT-ORIENTED FILE REGISTER OPERATIONS									
BCF	f, b	Bit Clear f	1	01	00bb	bfff	ffff		1,2
BSF	f, b	Bit Set f	1	01	01bb	bfff	ffff		1,2
BTFSC	f, b	Bit Test f, Skip if Clear	1 (2)	01	10bb	bfff	ffff		3
BTFSS	f, b	Bit Test f, Skip if Set	1 (2)	01	11bb	bfff	ffff		3
LITERAL AND CONTROL OPERATIONS									
ADDLW	k	Add literal and W	1	11	111x	kkkk	kkkk	C,DC,Z	
ANDLW	k	AND literal with W	1	11	1001	kkkk	kkkk	Z	
CALL	k	Call subroutine	2	10	0kkk	kkkk	kkkk		
CLRWDT	-	Clear Watchdog Timer	1	00	0000	0110	0100	$\overline{TO}, \overline{PD}$	
GOTO	k	Go to address	2	10	1kkk	kkkk	kkkk		
IORLW	k	Inclusive OR literal with W	1	11	1000	kkkk	kkkk	Z	
MOVLW	k	Move literal to W	1	11	00xx	kkkk	kkkk		
RETFIE	-	Return from interrupt	2	00	0000	0000	1001		
RETLW	k	Return with literal in W	2	11	01xx	kkkk	kkkk		
RETURN	-	Return from Subroutine	2	00	0000	0000	1000		
SLEEP	-	Go into Standby mode	1	00	0000	0110	0011	$\overline{TO}, \overline{PD}$	
SUBLW	k	Subtract W from literal	1	11	110x	kkkk	kkkk	C,DC,Z	
XORLW	k	Exclusive OR literal with W	1	11	1010	kkkk	kkkk	Z	

Grace Hopper & COBOL

- Human write instruction in 'English-like language' (COBOL)
- A program translates to machine language (compiler)
- Same program (in COBOL) can be compiled (translated) to run on different computers
 - Don't have to rewrite software for each computer



My family and computers



Pete Morales



Barbara Hosier