Don Barrett

TEXAS INSTRUMENTS
INCORPORATED
POST OFFICE BOX 53 • LUBBOCK TEXAS 79408

[illegible]

14 August 1978

John Poore
[illegible] Systems
12[illegible] San Fernando Road
Sylmar, Calif. 91342

Dear Mr. Poore

Please find attached the interface description and external communications document for the Texas Instruments SR-52 and SR-56 calculators and the PC-100 print cradle. This information is intended for your use only, and Texas Instruments shall not be liable for any incidental or consequential cost, expenses, or damages incurred through use of this information.

The warranties on Texas Instruments products shall not be enlarged, diminished, or affected by, and no obligation or liability shall arise or grow out of the rendering of this information to you by Texas Instruments. Please be advised that any modification or alteration, or damage caused by attempted modification or alteration of a Texas Instruments product is not within the scope of the standard warranty on that product, and any repair of such damaged or modified equipment will be in accordance with normal out-of-warranty repair charges and may also result in the equipment being restored to an as-manufactured condition as a part of the repair action. Also, please note that Texas Instruments reserves the right to change, alter, or discontinue products without notification.

I am sure that proper use of this information, in the spirit in which it is given, will help you better utilize your equipment.

Sincerely,

Christine Johnston

Christine Johnston
Consumer Relations

/deb
Enclosure

TEXAS INSTRUMENTS
INCORPORATED
POST OFFICE BOX 10508 • LUBBOCK, TEXAS 79408
Consumer Products Operations

TEXAS INSTRUMENTS PC-100A
INTERFACE DESCRIPTION

TEXAS INSTRUMENTS INCORPORATED
CALCULATOR PRODUCTS DIVISION
LUBBOCK, TEXAS

JUNE 1, 1975

INTRODUCTION

The purpose of this document is to provide an example of a low cost interface system to control the PC-100A. The PC-100A is normally controlled by one of the following calculators: TI-59, TI-58, SR-52, and SR-56. A switch under the front panel door must be set to the corresponding calculator. The third position, labeled (C) "other" is used for the TI-58 and TI-59. This position must also be used by this interface circuit.

The PC-100A was designed to be controlled by one of the above calculators which contain a specific set of instructions that the PC-100A recognizes. Since no other controller contains these codes and the proper format for sending them, an interface circuit must be built to generate these instructions. To encompass as many controllers as is feasible, this interface was designed such that data can be asynchronously transferred from the controller to the interface circuit. The interface circuit will then assemble the data and instructions into the proper format for the PC-100A.

This interface CKT is to be used in place of a calculator, not in conjunction with a calculator. The method of connecting the interface circuit to the PC-100A is left to the user. If a calculator is to be used on the PC-100A interchangeably, the interface circuit must be removeable.

The PC-100A is a thermal printer, printing up to 20 characters per line. All characters are configured by a 5 x 7 dot matrix. The PC-100 A converts a serial stream of data into the proper format such that five dots, when stepped seven times, will print the desired alpha numeric character. To utilize these characteristics, one must understand the capabilities and limitations of the system. This should be accomplished by a brief description of the following:

1. The I/O requirements of the PC-100A.
2. The I/O contraints of controlling machine.
3. An explanation of how the interface circuit works.
4. Operating Instructions - The sequence of events necessary to control the PC-100A.
5. Schematic of PC-100A.

PC-100A I/O REQUIREMENTS

CLOCK 1 A two-phase, non-overlapping clock (Ø 1) and (Ø 2) as described by Figure 1 is required so both systems operate at the same rate.

IDLE 2 The system was designed to operate by instruction cycles. An instruction cycle is 16 clock periods. Each clock period is defined as a State Time. S_0 thru S_{15} = 1 instruction cycle. Both systems must keep their State Times in sync; therefore, the PC-100A will set (S_0) as the first clock period after a falling edge on IDLE. The controlling system must generate a falling edge on the IDLE line each time it reaches its State Time Zero (S_0). Thus, the two systems will remain in sync.

EXT 3 EXT is a serial stream of 16 bits of data. Only S_3 thru S_9 are used as data to establish the desired character. The other bits are Don't-Care conditions.

IRG 4 IRG is a serial stream of 16 bits of data. S_3 thru S_{15} are used as data to establish the desired operation to be performed. S_0 thru S_2 are Don't-Care conditions. Each bit is clocked onto IRG by Ø1 and is clocked into the PC-100A by Ø2.

BUSY 5 A status line called BUSY is provided in order that peripherals can communicate various levels of activity to the calculator. This BUSY line is common to many peripherals and thus each unit lets the line float when not active. A peripheral can indicate an active status by pulling the BUSY line to V_{SS} (there is a pull down to V_{DD} in the calculator).

CONTROLLING MACHINE I/O REQUIREMENTS
HEREAFTER REFERRED TO AS A MICRO PROCESSOR OR µP

BUSY 1 This is an output from the interface CKT which is monitoring the PC-100A's activity. This line must be monitored by the micro-processor. If it is a logic "1," no load pulse is allowed. When a logic "0" is detected, this is the signal that the PC-100A has completed its last command and it is ready for the next command.

LOAD 2 When the μP is ready to load data and the BUSY line is at logic "0," the μP must generate a rising edge on the load line. This clocks the six bits of data and the two control bits into the holding registers of the interface CKT. The μP is now free to assemble new data. Data and control bits only need to be valid during the rising edge of LOAD.

DATA 3 Data is a six bit wide parallel Bus. Six bits are required for each character that is to be printed. Six bits allows 64 possible combinations. See Table I for the code to produce each of the 64 alpha-numeric characters.

CONTROL 4 Control is a two bit wide parallel Bus. Two bits allows 4 possible combinations. See Table III for the code to execute each of the 4 operations.

Note: All μP I/O must be TTL levels, positive logic.

INTERFACE CIRCUIT

1. VOLTAGE REQUIREMENTS are +5V and +16V. All chips have VCC = +5V. Only the open collector buffer (SN7417), pull up resistors use +16V. The level translation, TTL to MOS, is accomplished because the PC100A ground reference was designed to float. The MOS GND. (VSS) is connected directly to the +16V line of the interface circuit.

2. CLOCK - The Biphase clock is generated by an oscillator and a monostable multivibrator alternately gated by a Flip Flop. The period should be adjusted to be between 2.2 μsec and 2.8 μsec. The 1K pot adjusts the period (see schematic).

   Note: The higher the frequency, the faster the response time but the print intensity suffers.

   The pulse width (Tpw) should be adjusted, by the 10K pot, to 1 μsec minimum. Maximum will be determined by frequency and (T_{SW}). Switchint Time (T_{SW}) must be 800 N sec. minimum. (T_{SW}) is what is left over after 4(T_{RC}) + 2(T_{PW})'s are taken out of two periods. See Figure 1.

   Note: If frequency limits are exceeded significantly, one can't meet the T_{PW} and T_{SW} specifications and some of the MOS will malfunction.

3. PROM - Program the PROM per Table II. This contains the IRG instructions and timing events. The PROM is a 256 x 4 bit (SN74S287) PROM. An instruction cycle formats it into 16 words of 16 bits duration, 4 bits wide. The 4 bits are used as serial data streams y_1 thru y_4. Y_3 contains the IRG codes. Y_2 causes a step command to be executed when appropriate. Y_4 loads the EXT data at State Time 3 (S_3) on character load command only. Y_1 clears BUSY line at S_{15} for clear and load commands.

4. INSTRUCTION CYCLE - The (SN74293) is used as a divide-by-16 counter. It establishes the chip synchronization pulse on IDLE and steps the PROM through 16 addresses each instruction cycle.

OPERATING INSTRUCTIONS

A 16 bit code will come in serially on the IRG line. One of the six possible commands will be executed. Five of the commands are utilized by the interface processor and are listed on Table IV. Four of these five commands are determined externally by the two bit wide control BUS; the other command (Step) is automatically executed after a paper advance and after a print command. These five codes are generated by the PROM (SN74S287) in the interface circuit.

On a load character command, a 16 bit code will come in serially on the EXT line and one of the 64 possible alpha-numeric characters will be stored in the print register.

Characters are stored from right to left. As each character is stored, the pointer will move one position to the left. If more than 20 characters are entered consecutively, the pointer will wrap around to Position One and store over the previously stored character. See Table I for the character codes. The interface circuit takes 6 bits from the six bit wide parallel data BUS and inserts the seventh (MSB) bit into the serial data stream going to EXT.

The following is an explanation of the events for a typical load/print sequence. This sequence normally will be repeated for each line printed.

The first command should be control code 10. This clear command will store blanks in all 20 positions and set the pointer to the extreme right position.

The next command will be to enter the desired characters in the print register. Control code 01 loads the character and moves the pointer left to the next position. If a space is desired, load a blank. As previously noted, no more than 20 consecutive load character commands should be executed.

The next command will be to print the contents of the print register. Control code 11 will print the entire line. The contents of the print register are not altered.

The last command of the sequence is to advance the paper. Control code 00 will advance the paper 3/7ths of the height of the printed line.

> Note: Control code 00 can be executed from the PC-100A's front panel but the distance the paper will advance will be proportional to the time the button is held down.

See Table III for the approximate times required to execute these four commands.

See Table V for a typical load/print sequence. Study it to understand the significance of a right justified print register.

CHARACTER CODES

TABLE I

HEX CODE (SIX BITS)	EXT S_9 CODE S_3 MSB LSB	CHARACTER PRINTED	HEX CODE (SIX BITS)	EXT S_9 CODE S_3 MSB LSB	CHARACTER PRINTED
00	0000000	(blank)	20	0100000	.
01	0000001	0	21	0100001	U
02	0000010	1	22	0100010	V
03	0000011	2	23	0100011	W
04	0000100	3	24	0100100	X
05	0000101	4	25	0100101	Y
06	0000110	5	26	0100110	Z
07	0000111	6	27	0100111	+
08	0001000	7	28	0101000	×
09	0001001	8	29	0101001	*
0A	0001010	9	2A	0101010	Γ
0B	0001011	A	2B	0101011	π
0C	0001100	B	2C	0101100	≘
0D	0001101	C	2D	0101101	(
0E	0001110	D	2E	0101110	)
0F	0001111	E	2F	0101111	,
10	0010000	-	30	0110000	↑
11	0010001	F	31	0110001	%
12	0010010	G	32	0110010	:
13	0010011	H	33	0110011	/
14	0010100	I	34	0110100	=
15	0010101	J	35	0110101	'
16	0010110	K	36	0110110	×
17	0010111	L	37	0110111	$\bar{x}$
18	0011000	M	38	0111000	≥
19	0011001	N	39	0111001	?
1A	0011010	O	3A	0111010	÷
1B	0011011	P	3B	0111011	!
1C	0011100	Q	3C	0111100	Ⅱ
1D	0011101	R	3D	0111101	∴
1E	0011110	S	3E	0111110	Π
1F	0011111	T	3F	0111111	Σ

PROM CODES

SN74S287

TABLE II

	Y_1	Y_2	Y_3	Y_4
00→7F	0	1	0	1
	PAPER ADVANCE			
80	0	1	0	1
1	0	1	0	1
2	0	1	0	1
3	0	1	0	1
4	0	1	0	1
5	0	1	0	1
6	0	1	1	1
7	0	1	1	1
8	0	1	1	1
9	0	1	0	1
A	0	1	1	1
B	0	1	0	1
C	0	1	1	1
D	0	1	0	1
E	0	1	1	1
8F	0	0	0	1
	LOAD CHARACTER			
90	0	1	0	1
1	0	1	0	1
2	0	1	0	1
3	0	1	0	0
4	0	1	0	1
5	0	1	0	1
6	0	1	1	1
7	0	1	0	1
8	0	1	1	1
9	0	1	1	1
A	0	1	0	1
B	0	1	0	1
C	0	1	1	1
D	0	1	0	1
E	0	1	1	1
9F	1	1	0	1

	Y_1	Y_2	Y_3	Y_4
	CLEAR			
A0	0	1	0	1
1	0	1	0	1
2	0	1	0	1
3	0	1	0	1
4	0	1	0	1
5	0	1	0	1
6	0	1	1	1
7	0	1	0	1
8	0	1	0	1
9	0	1	0	1
A	0	1	1	1
B	0	1	0	1
C	0	1	1	1
D	0	1	0	1
E	0	1	1	1
AF	1	1	0	1
	PRINT			
B0	0	1	0	1
1	0	1	0	1
2	0	1	0	1
3	0	1	0	1
4	0	1	0	1
5	0	1	0	1
6	0	1	1	1
7	0	1	0	1
8	0	1	1	1
9	0	1	0	1
A	0	1	1	1
B	0	1	0	1
C	0	1	1	1
D	0	1	0	1
E	0	1	1	1
BF	0	0	0	1

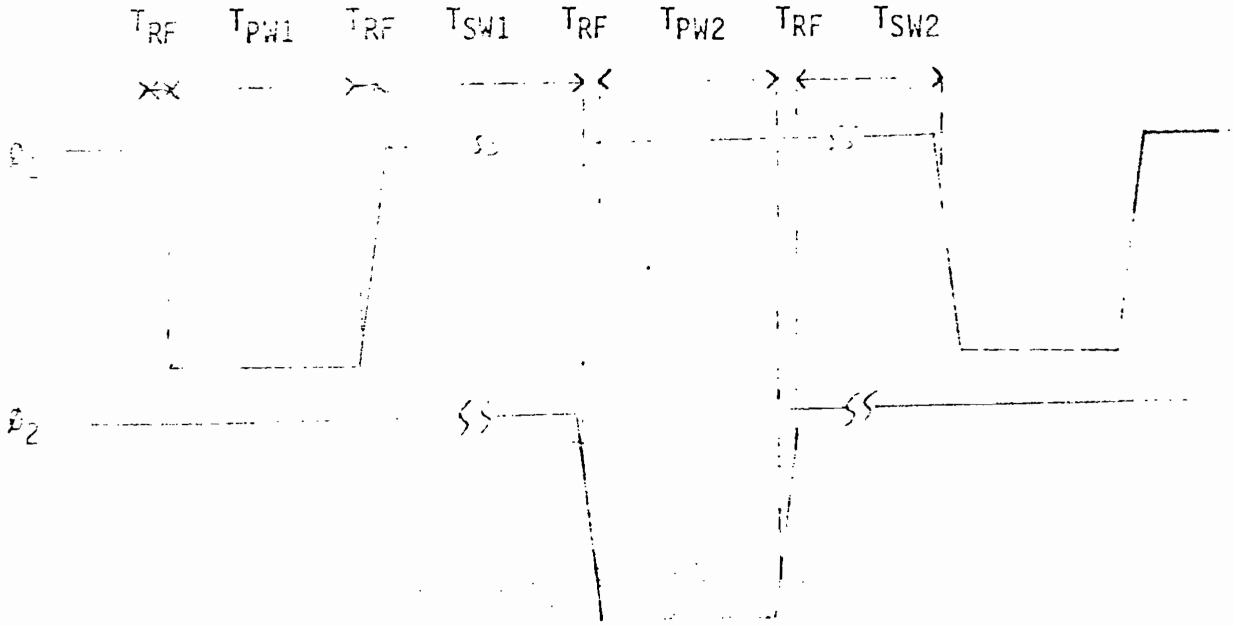
	Y_1	Y_2	Y_3	Y_4
	STEP			
C0	1	1	0	1
1	1	1	0	1
2	1	1	0	1
3	1	1	0	1
4	1	1	0	1
5	1	1	0	1
6	1	1	1	1
7	1	1	1	1
8	1	1	0	1
9	1	1	0	1
A	1	1	1	1
B	1	1	0	1
C	1	1	1	1
D	1	1	0	1
E	1	1	1	1
CF	1	1	0	1
D0→EF	1	1	1	1
	STEP			
F0	1	1	0	1
1	1	1	0	1
2	1	1	0	1
3	1	1	0	1
4	1	1	0	1
5	1	1	0	1
6	1	1	1	1
7	1	1	1	1
8	1	1	0	1
9	1	1	0	1
A	1	1	1	1
B	1	1	0	1
C	1	1	1	1
D	1	1	0	1
E	1	1	1	1
FF	1	1	0	1

TABLE III

CONTROL CODE MSB LSB	EXPLANATION	TIME TO EXECUTE
10	Clears the print buffer. Data = Don't Care	170 µs
01	Loads a character (defined by the DATA inputs) into the line buffer.	170 µs
11	Prints the buffer contents. Data = Don't Care	300 ms
00	Inititates a single paper advance which is 3/7ths character height. Data = Don't Care	100 ms

FIGURE I

CLOCK SPECIFICATIONS



		MIN	MAX	MEASURED AT
T_{PW}	=	1.0µs		90%
T_{RF}	=		300ns	10% TO 90%
T_{SW}	=	800ns		10%

Instruction ReGister (IRG)

TABLE IV

INSTRUCTION MNEMONIC	INSTRUCTION CODES															
	S_{15}	S_{14}	S_{13}	S_{12}	S_{11}	S_{10}	S_9	S_8	S_7	S_6	S_5	S_4	S_3	S_2	S_1	S_0
LOAD CHARACTER	0	1	0	1	0	0	1	1	0	1	0	0	0	0	0	0
CLEAR	0	1	0	1	0	1	0	0	0	1	0	0	0	0	0	0
STEP	0	1	0	1	0	1	0	0	1	1	0	0	0	0	0	0
PRINT	0	1	0	1	0	1	0	1	0	1	0	0	0	0	0	0
PAPER ADVANCE	0	1	0	1	0	1	0	1	1	1	0	0	0	0	0	0

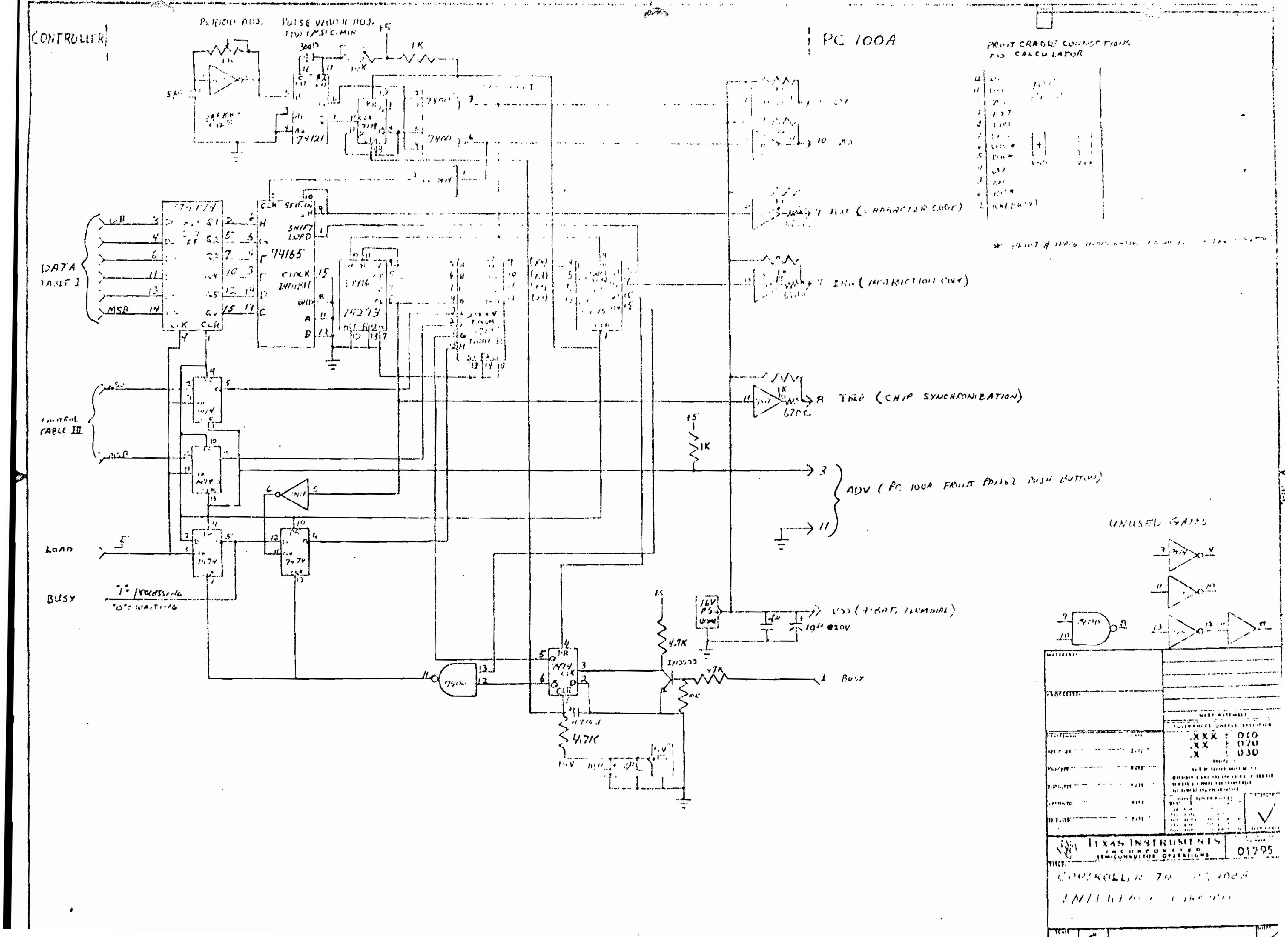
TABLE V

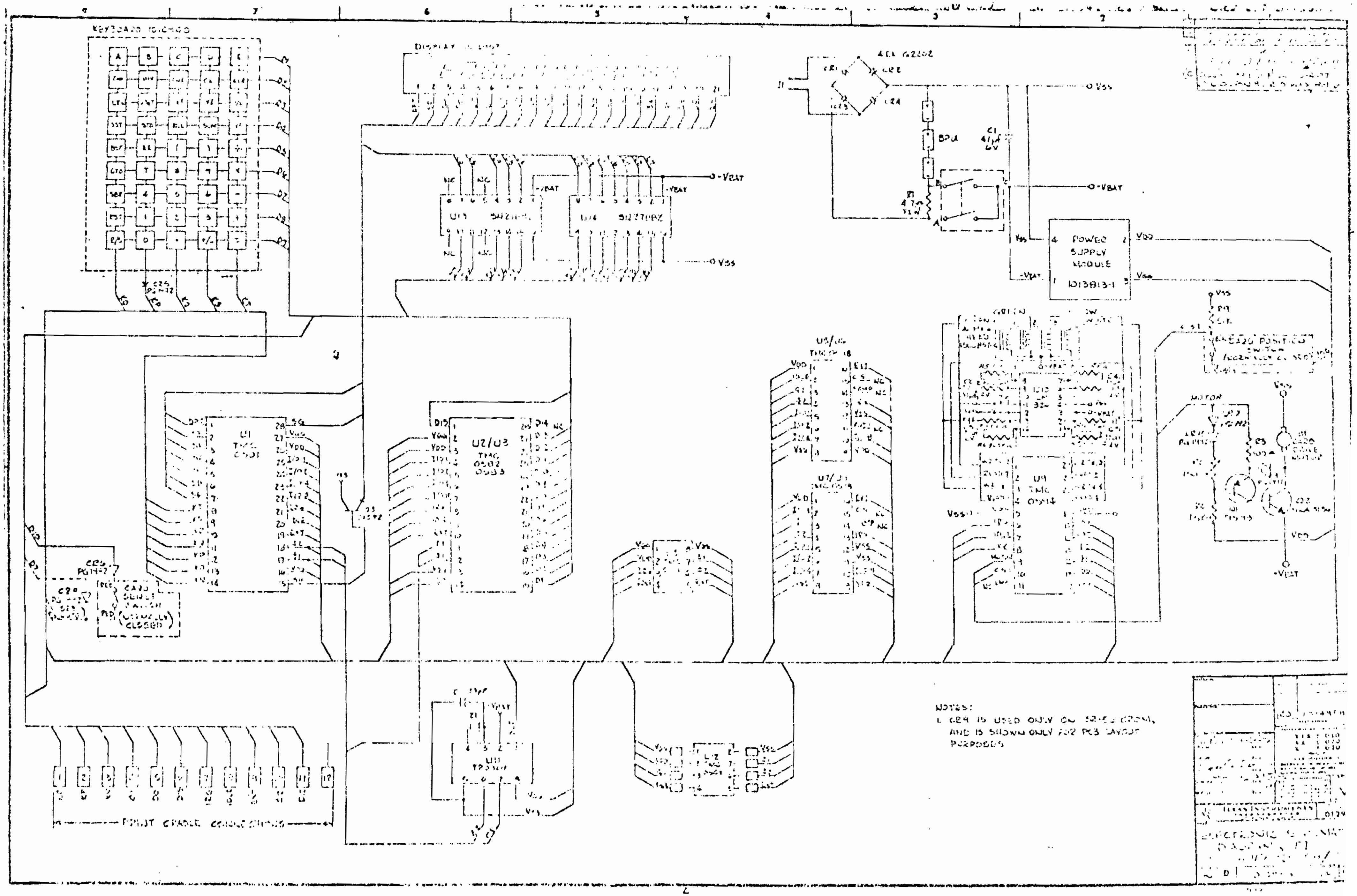
An example of a typical load - print sequence is shown below:

Desired Output	T H E S I N O F 3 0 D E G I S
	0 . 5

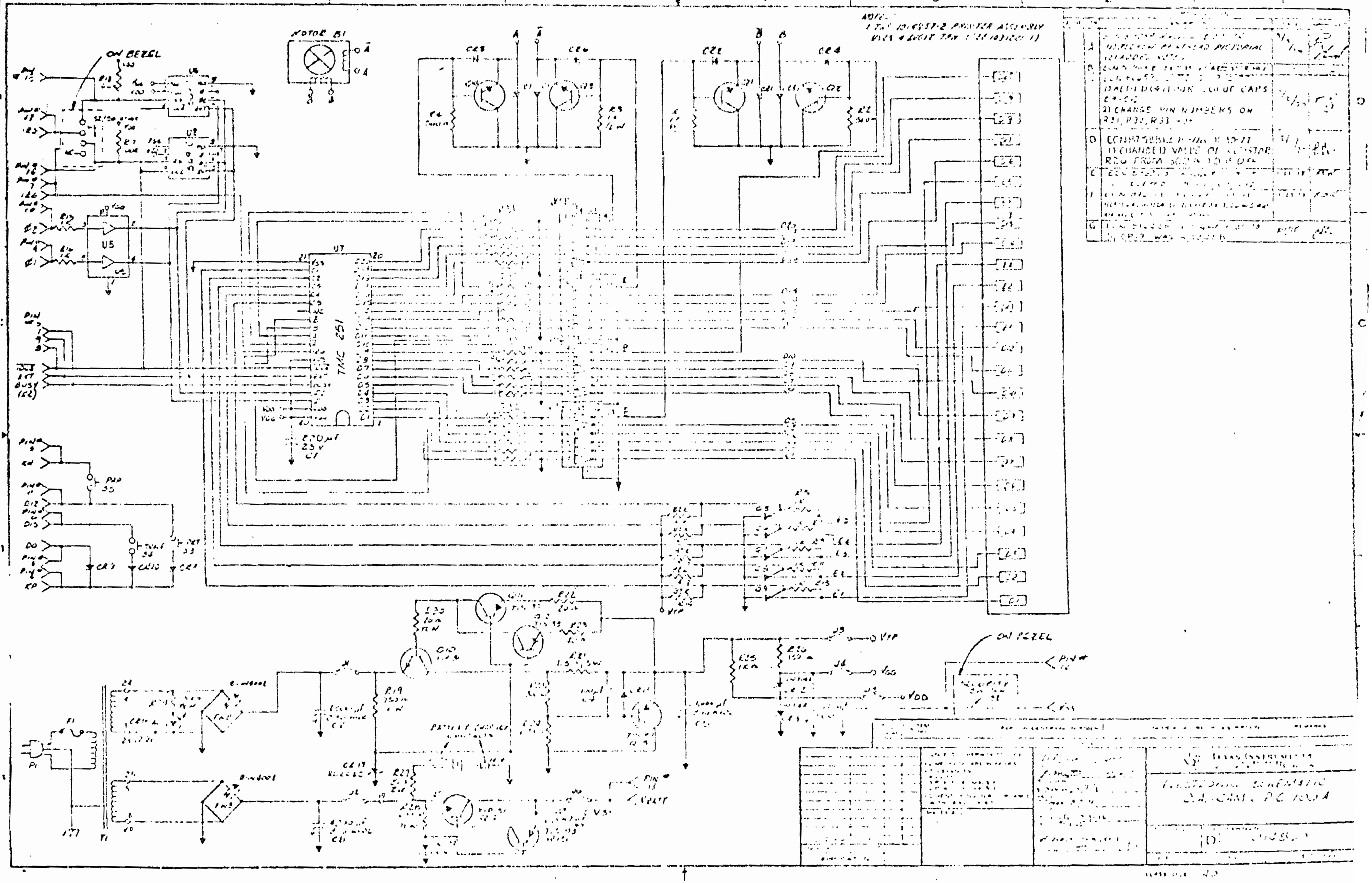
Note: Generate LOAD If and only If Busy = 0

	DATA MSB LSB	CONTROL MSB LSB	LOAD	RESULTS		
	000000	1 0	__⎸‾‾	CLEARS REGISTER		
	011110	0 1	"	LOADS	CHARACTER	S
	010100	0 1	"	"	"	1
	000000	0 1	"	"	"	BLANK
	010010	0 1	"	"	"	G
	001111	0 1	"	"	"	E
	001110	0 1	"	"	"	D
	000000	0 1	"	"	"	BLANK
	000001	0 1	"	"	"	0
	000100	0 1	"	"	"	3
	000000	0 1	"	"	"	BLANK
	010001	0 1	"	"	"	F
	011010	0 1	"	"	"	0
	000000	0 1	"	"	"	BLANK
	011001	0 1	"	"	"	N
	010100	0 1	"	"	"	1
	011110	0 1	"	"	"	S
	000000	0 1	"	"	"	BLANK
	001111	0 1	"	"	"	E
	010011	0 1	"	"	"	H
	011111	0 1	"	"	"	T
	XXXXXX	1 1	"	PRINTS LINE		
	XXXXXX	0 0	"	ADVANCES PAPER		
NEXT SEQUENCE	XXXXXX	1 0	"	CLEARS REGISTER		
	000000	0 1	"	LOADS	CHARACTER	BLANK
	000000	0 1	"	"	"	"
	000000	0 1	"	"	"	"
	000000	0 1	"	"	"	"
	000110	0 1	"	"	"	5
	100000	0 1	"	"	"	.
	000001	0 1	"	"	"	0
	XXXXXX	1 1	"	PRINTS LINE		
	XXXXXX	0 0	"	ADVANCES PAPER		


CONTROLLER
PERIOD ADJ.
PULSE WIDTH ADJ.
+5
1K
300Ω
74121
7400
74174
74165
SHIFT LOAD
CLOCK INHIBIT
74173
7474
7400
7404
DATA TABLE I
MSB
CONTROL TABLE II
LOAD
BUSY
"1" PROCESSING
"0" WAITING
PC 100A
PRINT CRADLE CONNECTIONS TO CALCULATOR
7 TEXT (CHARACTER CODE)
9 IRG (INSTRUCTION CODE)
8 TRG (CHIP SYNCHRONIZATION)
3 ADV (PC 100A FRONT PANEL PUSH BUTTON)
11
16V PS
VSS (PRINT TERMINAL)
10µ @20V
4.7K
47K
2N3393
1 BUSY
UNUSED GATES
TEXAS INSTRUMENTS INCORPORATED SEMICONDUCTOR OPERATIONS
.XXX ± .010
.XX ± .020
.X ± .030
01795
CONTROLLER TO PC 100A
INTERFACE CIRCUIT



NOTES:
1. CR4 IS USED ONLY ON SR-52 CRSM, AND IS SHOWN ONLY FOR PCB LAYOUT PURPOSES


NOTE:
1 TMI 1016537-2 PRINTER ASSEMBLY
2) ADDED DRILL SIZE LOGIC CAPS
2) CHANGE PIN NUMBERS ON
R31, R32, R33
D ECN 1875851
1) CHANGED VALUE OF RESISTOR
R26 FROM
G
CR12 WAS
MOTOR B1
CR8
CR6
CR2
CR4
R4
R3
R2
U6
U8
U5
U7
TMC 251
ON BEZEL
ϕ2
ϕ1
BUSY
C7
25V
CR9
CR10
CR11
C1
C2
C4
C5
C6
R19
R20
R21
R22
R25
R26
R27
Q10
BATTERY CHARGE
VBATT
VPP
VGG
VDD
VSS
ON BEZEL
PIN
T1
F1
P1
TEXAS INSTRUMENTS
ELECTRICAL SCHEMATIC
PC 100A
D

TEXAS INSTRUMENTS
INCORPORATED
POST OFFICE BOX 10508 • LUBBOCK, TEXAS 79408
Consumer Products Operations

August 28, 1978

PACESETTER SYSTEMS
12740 San Fernando Road
Fylmar, California 91342

Attention: Mr. John Poore

Dear Sir:

Please find attached the interface communications document for the Texas Instruments PC-100A print cradle. This information is intended for your use only, and Texas Instruments shall not be liable for any incidental or consequential cost, expenses, or damages incurred through use of this information.

The warranties on Texas Instruments products shall not be enlarged, diminished, or affected by, and no obligation or liability shall arise or grow out of the rendering of this information to you by Texas Instruments. Please be advised that any modification or alteration, or damage caused by attempted modification or alteration of a Texas Instruments product is not within the scope of the standard warranty on that product, and any repair of such damaged or modified equipment will be in accordance with normal out-of-warranty repair charges and may also result in the equipment being restored to an as-manufactured condition as a part of the repair action. Also, please note that Texas Instruments reserves the right to change, alter, or discontinue products without notification.

I am sure that proper use of this information, in the spirit in which it is given, will help you better utilize your equipment.

Sincerely,

Dan J. Ezzone
Product Manager
Programmable Calculators

DJE:pn
Attachment

TEXAS INSTRUMENTS PC 100

INTERFACE DESCRIPTION

TEXAS INSTRUMENTS INCORPORATED

CALCULATOR PRODUCTS DIVISION

DALLAS, TEXAS

SEPTEMBER 30, 1976

The Texas Instruments PC 100 Print Cradle can be interfaced to mini/micro computer systems to provide low cost alphanumeric 20 column thermal print capability. The interface described herein allows a wide range of microprocessors to be interfaced to the print cradle with a minimum of additional hardware.

The interface talks to the processor by way of a six bit wide data bus, (DATA) a two bit wide bus (CONTROL), a load line (LOAD) and a status (BUSY) line. The interface loads the contents of Control and Data when the processor signals with a rising edge on LOAD. Busy is generated by the interface to communicate its operational status to the processor. When the processor signals the interface with a LOAD, Busy is switched high, and the data bus and control bus from the processor must remain valid until Busy goes low. The twenty print characters are loaded into the interface sequentially, from right to left, and then the entire line is printed. Prior to loading and printing a line the interface must be cleared by the processor. When the interface is cleared it loads blanks into the print buffer, therefore any positions not loaded by the processor will be blanks (i.e., if less than twenty characters are loaded the remainder will be blank). If more than twenty characters are loaded the interface loads the excess over the beginning positions. The data bus and control bus are defined below.

Control Code	Explanation
00	Initiates a single paper advance which is approximately half a print line. Data = don't cares
01	Enters a character (defined by the data bus) into the line buffer.
10	Clears the print buffer. Data = don't cares
11	Print the buffer contents. Data = don't cares

The power supply voltages required by the interface are +5 and +16 as indicated on the schematics.

PROM CODES (IC [illegible])

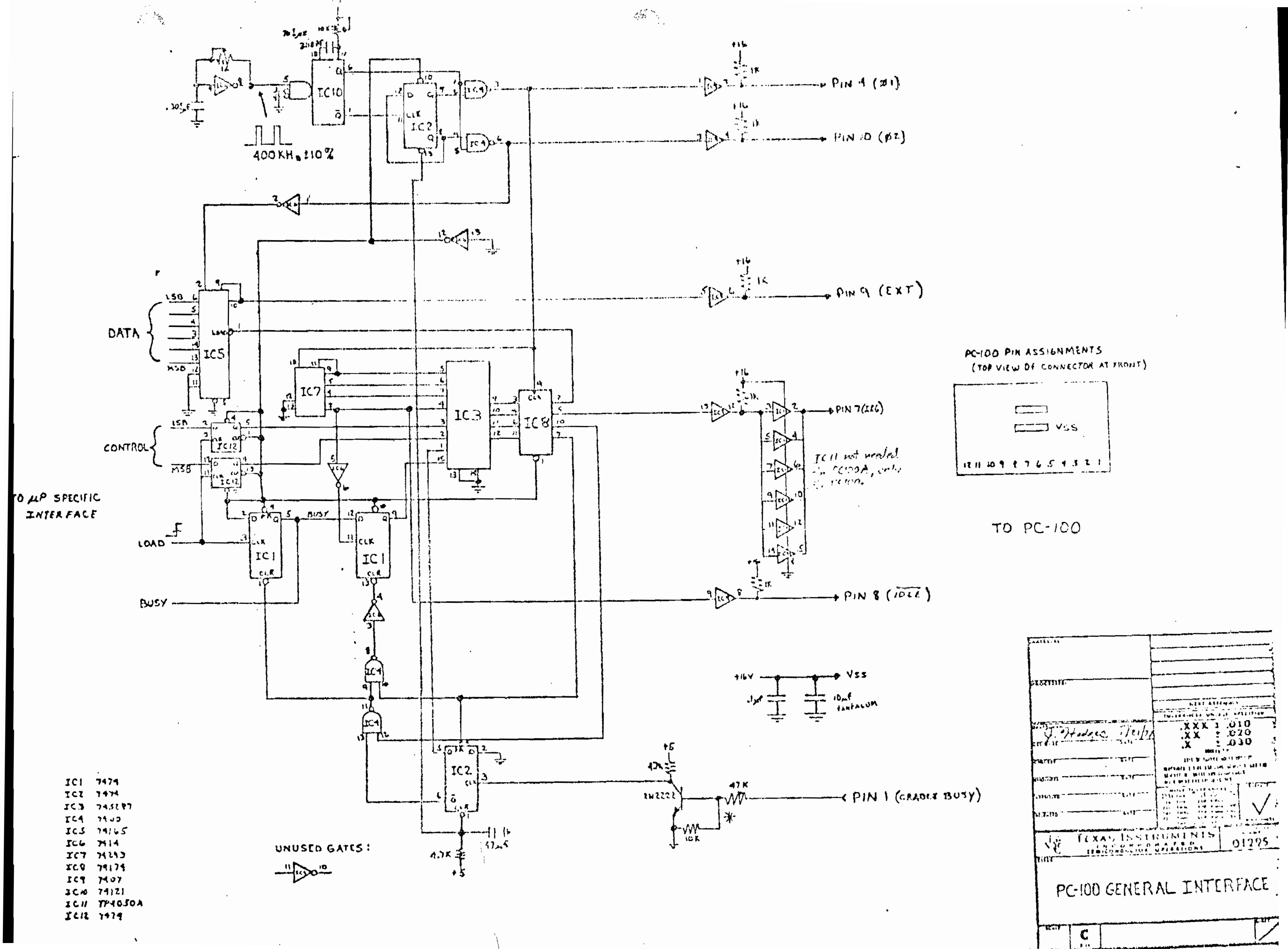
0 = burn

		Y_1	Y_2	Y_3	Y_4
	00 → 7F	0	1	0	1
pap adv	80	0	1	0	1
	1	0	1	0	1
	2	0	1	0	1
	3	0	1	0	1
	4	0	1	0	1
	5	0	1	0	1
	6	0	1	1	1
	7	0	1	1	1
	8	0	1	1	1
	9	0	1	0	1
	A	0	1	1	1
	B	0	1	0	1
	C	0	1	1	1
	D	0	1	0	1
	E	0	1	1	1
	F	0	1	0	1
load char	90	0	1	0	1
	1	0	1	0	1
	2	0	1	0	1
	3	0	1	0	0
	4	0	1	0	1
	5	0	1	0	1
	6	0	1	1	1
	7	0	1	0	1
	8	0	1	1	1
	9	0	1	1	1
	A	0	1	0	1
	B	0	1	0	1
	C	0	1	1	1
	D	0	1	0	1
	E	0	1	1	1
	F	1	1	0	1

		Y_1	Y_2	Y_3	Y_4
clr	A0	0	1	0	1
	1	0	1	0	1
	2	0	1	0	1
	3	0	1	0	1
	4	0	1	0	1
	5	0	1	0	1
	6	0	1	1	1
	7	0	1	0	1
	8	0	1	0	1
	9	0	1	0	1
	A	0	1	1	1
	B	0	1	0	1
	C	0	1	1	1
	D	0	1	0	1
	E	0	1	1	1
	F	1	1	0	1
print	B0	0	1	0	1
	1	0	1	0	1
	2	0	1	0	1
	3	0	1	0	1
	4	0	1	0	1
	5	0	1	0	1
	6	0	1	1	1
	7	0	1	0	1
	8	0	1	1	1
	9	0	1	0	1
	A	0	1	1	1
	B	0	1	0	1
	C	0	1	1	1
	D	0	1	0	1
	E	0	1	1	1
	F	0	0	0	1

	Y_1	Y_2	Y_3	Y_4
C0	1	1	0	1
1	1	1	0	1
2	1	1	0	1
3	1	1	0	1
4	1	1	0	1
5	1	1	0	1
6	1	1	1	1
7	1	1	1	1
8	1	1	0	1
9	1	1	0	1
A	1	1	1	1
B	1	1	0	1
C	1	1	1	1
D	1	1	0	1
E	1	1	1	1
F	1	1	0	1
D0 → DF	1	1	1	1
F0	1	1	0	1
1	1	1	0	1
2	1	1	0	1
3	1	1	0	1
4	1	1	0	1
5	1	1	0	1
6	1	1	1	1
7	1	1	1	1
8	1	1	0	1
9	1	1	0	1
A	1	1	1	1
B	1	1	0	1
C	1	1	1	1
D	1	1	0	1
E	1	1	1	1
F	1	1	0	1

EXT CODE	CHARACTER PRINTED	EXT CODE	CHARACTER PRINTED
S_9 S_3		S_9 S_3	
0000000	(blank)	0100011	W
0000001	0	0100100	X
0000010	1	0100101	Y
0000011	2	0100110	Z
0000100	3	0100111	+
0000101	4	0101000	x
0000110	5	0101001	*
0000111	6	0101010	✓
0001000	7	0101011	π
0001001	8	0101100	e
0001010	9	0101101	(
0001011	A	0101110	)
0001100	B	0101111	,
0001101	C	0110000	↑
0001110	D	0110001	%
0001111	E	0110010	$
0010000	-	0110011	/
0010001	F	0110100	=
0010010	G	0110101	,
0010011	H	0110110	x
0010100	I	0110111	x
0010101	J	0111000	2
0010110	K	0111001	?
0010111	L	0111010	÷
0011000	M	0111011	!
0011001	N	0111100	II
0011010	O	0111101	Δ
0011011	P	0111110	π
0011100	Q	0111111	Σ
0011101	R		
0011110	S		
0011111	T		
0100000	.		
0100001	U		
0100010	V		



IC1	7479
IC2	7474
IC3	74S287
IC4	7400
IC5	74165
IC6	7414
IC7	74293
IC8	74174
IC9	7407
IC10	74121
IC11	TP4050A
IC12	7479

J. Hodges 7/21/77

.XXX ± .010
.XX ± .020
.X ± .030

TEXAS INSTRUMENTS
INCORPORATED
SEMICONDUCTOR OPERATIONS

01295

PC-100 GENERAL INTERFACE

C

EXTERNAL COMMUNICATIONS

FOR THE SR-52/56 CALCULATORS

TEXAS INSTRUMENTS INCORPORATED

CALCULATOR PRODUCTS DIVISION

DALLAS, TEXAS

SEPTEMBER 27, 1976

GENERAL:

Signal description (for exact voltage ranges and timing, consult 0500 family spec.)

VOLTAGE:

V_{SS} = 0V; system GND and logic level "1" V_{DD} = -10V and is logic "0" V_{GG} = -16V.

CLOCKS:

A two phase non-overlaping clock is used throughout the system; labeled $\phi 1$, $\phi 2$. The clocks are negative pulses from V_{SS} to V_{GG} of approximately 1μs pulse width and 20% duty cycle as shown in Figure 1. The negative transition is therefore the "leading edge" of the clock.

Internally these clocks are further divided into 16 state times, S_0 - S15, each representing a full cycle of the clocks, i.e., from leading edge of $\phi 1$ to the leading edge of the next $\phi 1$. A full cycle of state times is defined as an instruction cycle, approximately 80 μsec long. See Figure 1.

SYNCHRONIZATION:

The various calculator chips and external devices must be in sync, i.e., in the same state time in order to communicate. This function is provided by the $\overline{\text{IDLE}}$ signal, an output from the calculator. $\overline{\text{IDLE}}$ is a logic level signal, i.e., it switches between V_{SS} and V_{DD}. The negative transition of IDLE ($V_{SS} \rightarrow V_{DD}$) sets the leading edge of S_0 as shown in Figure 1. When the positive transition of $\overline{\text{IDLE}}$ occurs at S_1 (leading edge) and remains high (V_{SS}) until S_0 the calculator is in the calculate mode. When $\overline{\text{IDLE}}$ occurs at the leading edge of S_{15} the calculator is in the display mode and is scanning the keyboard.

Thus any external device wishing to communicate with the calculator must monitor the negative transition of $\overline{\text{IDLE}}$ and set its state time counter accordingly. Any external device wishing to control a calculator peripheral (like the P.C. 100) must provide a negative transition on $\overline{\text{IDLE}}$ (an input to peripherals) every 16th $\phi 1$.

INSTRUCTIONS:

All calculator instructions are transmitted serially on the IRG line. Each instruction is 16 bits long, one per state time. Each bit is clocked onto IRG at $\phi 1$ and devices monitoring IRG clock it in with $\phi 2$. IRG bits at state times S_0 thru. S_2 are don't cares. The LSB is clocked onto IRG at S_3 and the MSB at S_{15}. IRG is a logic level signal.

DATA:

Data is transmitted between calculator chips and from the calculator to peripherals on the EXT line. EXT is a logic level signal with one bit of data each state time. S_3 thru S_9 are the state times when data is sent to most peripherals. The LSB is sent during S_3 and the MSB during S_9.

STATUS:

A status line called BUSY is provided in order that peripherals can communicate various levels of activity to the calculator. This BUSY line is common to many peripherals and thus each unit lets the line float when not active. A peripheral can indicate an active status by pulling the BUSY line to V_{SS} (there is a pull down to V_{DD} in the calculator).

Print Cradle Communications

The PC 100 is provided with a 14 contact connector for interface to the calculator. Figure 2 shows the pin assignments of this connector. The PC 100 recognizes six distinct instructions on IRG. These instructions enable the print buffer to be cleared or loaded, a line of loaded characters to be printed and a paper advance to be actuated. The data on EXT (S_3 - S_9) is clocked into a shift register each instruction cycle and selectively loaded into a memory upon appropriate IRG command. When the printer decodes a Clear command the print buffer is set to all zeros (blanks on printout) and the character load pointer is set on the right most position. When the first LOAD command is decoded, the data from EXT is loaded into the right most print position of the memory and the character load pointer is moved one position to the left. Each character load command received thereafter loads the current EXT into the memory and moves the character load pointer one position left. If this sequence is repeated for more than 20 character loads, the pointer wraps around to the initial position and the right most character is written over. A full twenty characters can be loaded by this character load sequence and subsequetly printed. When the left most character in the desired output has been loaded no further character loads are required; the Clear instruction loaded the entire memory with the code for a blank. The Print instruction causes the PC 100 to initiate printing of the current contents of the memory. The Paper Advance command will cause the paper to advance one half a line. The Step command has two distinct functions. If Step is decoded during the character load sequence it moves the pointer one position left and leaves the memory unchanged (if a Clear was used to begin the sequence a blank will be in the memory) regardless of the code on EXT. The second function of the Step command occurs when the PC 100 is printing a line, i.e., in a print cycle. During a print cycle the Step command causes the PC 100 to pull the BUSY status line to V_{SS} at S_2 of the instruction cycle following the Step command. A loop is used by the processor controlling the PC 100 to find when the print or paper advance sequence is finished. The FUNCTION command causes the EXT code to be converted by the PC 100 into three character codes for easy loading of often used alpha strings like SIN, COS, PRT etc.,. There are forty fixed three character codes available as shown in Figure 3. The character load pointer is moved over three places by the FUNCTION command. This instruction should only be used in the right most positions of the printout.

The codes for the instructions and the codes for the 64 possible characters are shown in Figures 4 and 5.

An example of a typical load - print sequence is shown below:

	THE SIN 0F 30 DEG IS
Desired Output	0.5

Note: Each line represents one instruction cycle.

IRG	EXT
CLEAR	X
X	S CODE
LOAD	S CODE
X	I CODE
LOAD	I CODE
STEP	X
X	G CODE
LOAD	G CODE
X	E CODE
LOAD	E CODE
X	D CODE
LOAD	D CODE
STEP	X
X	0 CODE
LOAD	0 CODE
X	3 CODE
LOAD	3 CDOE
STEP	X
X	F CODE
LOAD	F CODE
X	0 CODE
LOAD	0 CODE
STEP	X
X	N CODE
LOAD	N CODE
X	I CODE
LOAD	I CODE
X	S CODE
LOAD	S CODE
STEP	X
X	E CODE
LOAD	E CODE
X	H CODE
LOAD	H CODE
X	T CODE
LOAD	T CODE

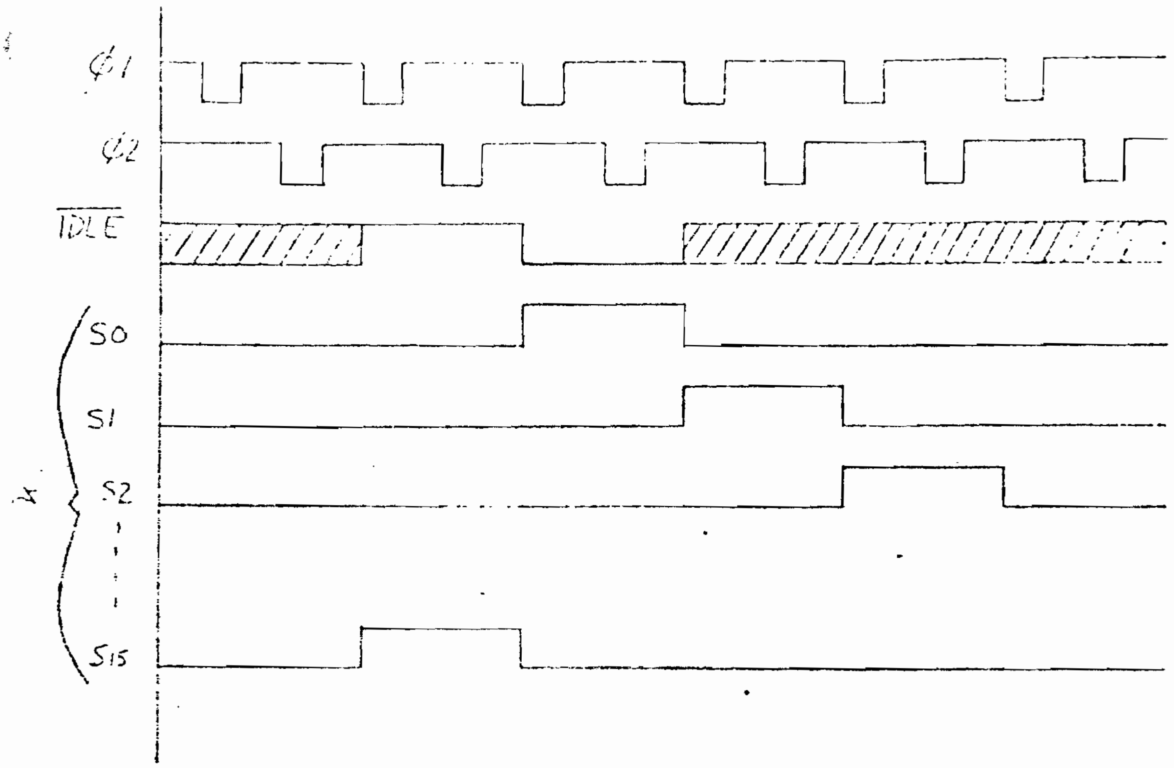
```
                      IRG                                EXT

                      PRINT                              X
                  ┌─→ STEP                               X
IF BUSY = 1 ──────┘   X                                  X
   BUSY = 0 ────────→ CLEAR                              X
                      PAPER ADVANCE                      X
                  ┌─→ STEP                               X
IF BUSY = 1 ──────┘   X                                  X
   BUSY = 0 ────────→ CLEAR                              X
                      PAPER ADVANCE                      X
                  ┌─→ STEP                               X
   BUSY = 1 ──────┘   X                                  X
   BUSY = 0 ────────→ CLEAR                              X
                      STEP                               X
                      STEP                               X
                      STEP                               X
                      STEP                               X
                      X                                  5 CODE
                      LOAD                               5 CODE
                      X                                  · CODE
                      LOAD                               · CODE
                      X                                  0 CODE
                      LOAD                               0 CODE
                      PRINT                              X
                  ┌─→ STEP                               X
   BUSY = 1 ──────┘   X                                  X
   BUSY = 0 ────────→ CLEAR                              X
```

Calculator Communication

The calculator may be adapted to drive other peripherals by interfacing to the print crade connections in the battery pack. The calculator can be programmed to calculate data intended for a peripheral and then a PRT key actuated to transfer the contents of the display and possible function trailer over the battery pack connector to the external device. With knowledge of the PC 100 communications lines described earlier the peripheral designer can decode appropriate instructions and data for his specific use. For the calculator to send this data it must percieve a proper connection to the PC 100. The allusion of a proper connection can be created by the diode between pins 5 and 2 of the battery pack connector as shown below.

Pin 5 ———▷|——— Pin 2

TMS 0250 TIMING



* STATE TIME DEF

Notes:

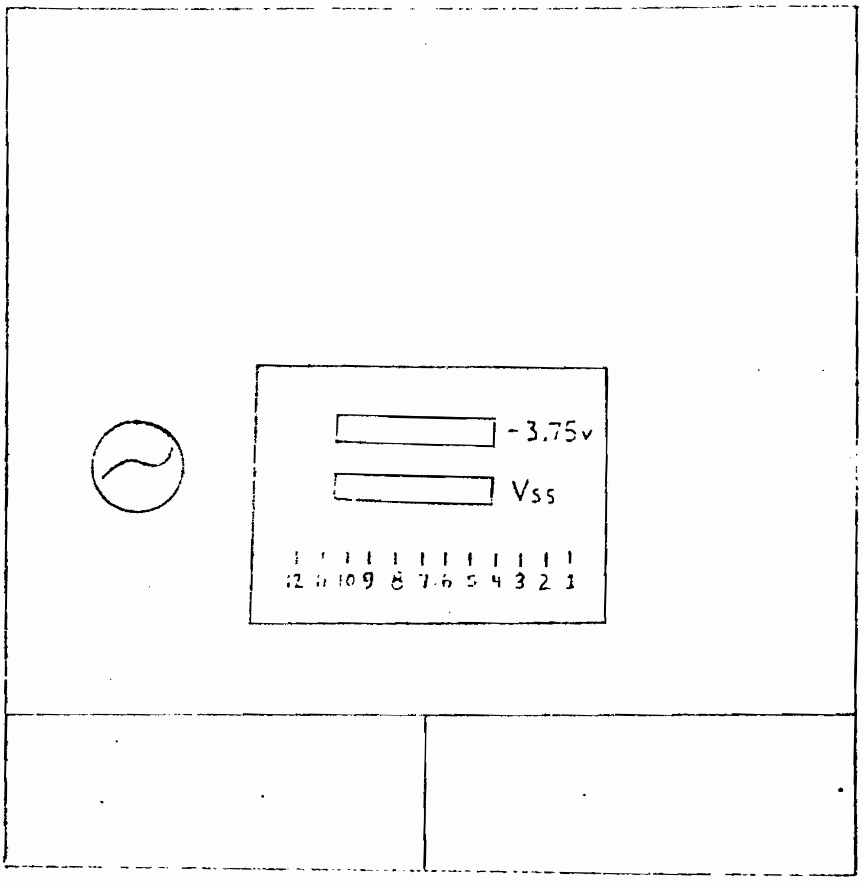
1. Character Data must be present at EXT for the entire state time as follows:

LSB	S_3
D2	S_4
D3	S_5
D4	S_6
D5	S_7
D6	S_8
MSB	S_9

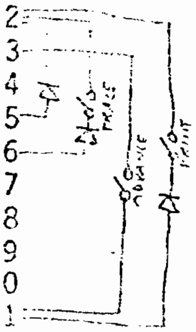
2. Instruction Word data must be present at IRG for the entire state time as follows:

LSB	S_3
MSB	S_{15}

Figure 1



Contact #	Function
1	Busy
2	*
3	*
4	Ø1
5	*
6	*
7	IRG
8	IDLE
9	EXT
10	Ø2
11	*
12	*



* Involved with PC 100 PRINT, PAPER ADVANCE, TRACE KEYS

1 Battery voltage is applied to the calculator through these large contacts.

Figure 2

FUNCTION CODES

FIGURE 3

EXT CODE (BIT PATTERN)	EXT (CODE) (HEX)	PRINTED CHARACTERS		
S_9 | S_3		EXT CODE (OCTAL)		HEX
1 1 1 1 1 0 0	7C	174	IF	7C
0 0 1 0 0 0 1	11	021	=	11
1 0 1 0 1 1 1	57	127	SIN	57
1 0 1 0 1 1 0	56	126	COS	56
1 0 1 1 1 0 1	5D	135	TAN	5D
1 1 0 0 0 0 1		141	SUM	61
1 1 0 1 0 0 1		151	Σ +	69
1 1 0 0 1 1 0		146	STO	66
1 1 0 1 0 0 0		150	RCL	68
1 0 1 0 0 1 1		123	PRM	53
1 0 1 0 0 0 1		121	LNX	
0 1 1 1 1 0 0		074	√X	3C
0 1 1 1 1 0 1		075	X ⇄ Y	3D
0 0 1 0 0 1 0		022	−	12
0 0 1 0 0 1 1		023	+	13
0 0 1 0 1 1 0		026	÷	16
0 0 1 0 1 1 1		027	×	17
0 0 1 1 0 1 0		032	x √Y	1A
0 0 1 1 0 1 1		033	Y^x	1B
0 1 0 0 0 0 1		041	CLR	21
0 1 0 0 0 1 0		042	INV	22
0 1 0 0 0 1 1		043	DPT	23
0 1 0 0 1 1 1		047	+ / -	27
0 1 0 0 1 1 0		046	CE	26
0 1 0 1 1 0 1		055	EE	2d
0 1 1 0 0 0 1		061	e^x	31
0 1 1 0 0 1 1		063	x^2	33
0 1 1 0 1 1 0		066	1 / x	36
1 0 1 0 1 0 0		124	%	54
1 1 0 0 1 1 1		147	π	67
1 1 1 0 0 0 0		160	ERR	70
1 1 1 0 0 0 1		161	(	71
1 1 1 0 0 1 0		162	)	72
1 1 1 0 0 1 1		163	LRN	73
1 1 1 0 1 0 0		164	RUN	74
1 1 1 0 1 1 0		166	HLT	76
1 1 1 1 0 0 0		170	STP	78
1 1 1 1 0 1 0		172	GTO	7A
[illegible]		[illegible]	[illegible]	04
[illegible]		[illegible]	[illegible]	0B
[illegible]		012	[illegible]	0A
[illegible]		014	XM	0C
[illegible]		114	FD	4C

INSTRUCTION CODES

INSTRUCTION MNEMONIC	S_{15}	S_{14}	S_{13}	S_{12}	S_{11}	S_{10}	S_9	S_8	S_7	S_6	S_5	S_4	S_3	S_2	S_1	S_0	
LOAD	0	1	0	1	0	0	1	1	0	1	0	0	0	X	X	X	5390 (5397)
FUNCTION	0	1	0	1	0	0	1	1	1	1	0	0	0	X	X	X	53C0 (53C7
CLEAR	0	1	0	1	0	1	0	0	0	1	0	0	0	X	X	X	5440 (5447
STEP	0	1	0	1	0	1	0	0	1	1	0	0	0	X	X	X	54C0 (54C7
PRINT	0	1	0	1	0	1	0	1	0	1	0	0	0	X	X	X	5540 (5547
PAPER ADV	0	1	0	1	0	1	0	1	1	1	0	0	0	X	X	X	55C0 (55C

Figure 4

CHARACTER CODES

EXT CODE (BIT PATTERN)	EXT CODE (OCTAL)	CHARACTER PRINTED	EXT CODE (BIT PATTERN)	EXT CODE (OCTAL)	CHARACTER PRINTED
S_9 S_3			S_9 S_3		
0000000	00	(blank)	0100011	43	W
0000001	01	0	0100100	44	X
0000010	02	1	0100101	45	Y
0000011	03	2	0100110	46	Z
0000100	04	3	0100111	47	+
0000101	05	4	0101000	50	x
0000110	06	5	0101001	51	*
0000111	07	6	0101010	52	√
0001000	10	7	0101011	53	π
0001001	11	8	0101100	54	e
0001010	12	9	0101101	55	(
0001011	13	A	0101110	56	)
0001100	14	B	0101111	57	,
0001101	15	C	0110000	60	↑
0001110	16	D	0110001	61	%
0001111	17	E	0110010	62	$
0010000	20	-	0110011	63	/
0010001	21	F	0110100	64	=
0010010	22	G	0110101	65	,
0010011	23	H	0110110	66	X
0010100	24	I	0110111	67	x
0010101	25	J	0111000	70	2
0010110	26	K	0111001	71	?
0010111	27	L	0111010	72	÷
0011000	30	M	0111011	73	!
0011001	31	N	0111100	74	II
0011010	32	O	0111101	75	Δ
0011011	33	P	0111110	76	Π
0011100	34	Q	0111111	77	Σ
0011101	35	R			
0011110	36	S			
0011111	37	T			
0100000	40	.			
0100001	41	U			
0100010	42	V			

Figure 5