TEXAS 27C256 JC.
27256I.
SGS M27256F1

P9030 Users manual

GP INDUSTRIAL ELECTRONICS 1984

GP Industrial Electronics Ltd
Unit E, Huxley Close,
Newnham Ind. Estate, Plymouth
Tel: (0752) ~~332964~~
342961

P9030 Manual - Table of Contents

P9030 Manual - Table of Contents

Introduction

The P9000 series of PROM Programmers has been designed to efficiently handle a wide range of 24 and 28 pin MOS EPROMs and EEPROMs.
Personality cards and hardware changes are not required. Eight zero insertion force sockets (ZIFs) are provided for programming along with a single master socket.
Operating the machine is very straightforward.
The 16 character display shows the selected device type and function mode and also gives details of any operator errors, PROM faults and system faults.
The discrete ZIF LED indicators show mode PASS or FAIL for each copy socket.
Only 6 keys are required to operate the machine - to select the device, select the mode and start the operation.
All the machines in the range are protected against common programming faults such as device reverse insertion, shorted pins, and misplaced insertion.

On the P9020 and P9030 machines an 8k x 8 RAM buffer is incorporated to facilitate programming from sources other than the Master socket such as a development system or computer.
This buffer can be expanded to 32k x 8 internally.

The P9030 includes a sophisticated RAM editor and special RAM programming functions along with a remote control facility.

General Operating Instructions

Supply Voltage

Machines supplied in the UK and Europe are set to operate at 240v, 50Hz supply. A ready wired cable, complete with fused plug is supplied. The cores of the cable are colour coded as follows:

Live: Brown   Neutral: Blue   Earth: Green/Yellow

The mains cable plugs into the P9000 via the connector marked "MAINS", located on the righthand side, rear of the unit. The pins on this connector are:

Earth

Live   Neutral

The unit is additionally protected by a 500mA Antisurge fuse located within the unit. Access to this fuse is gained by first disconnecting the mains supply, and then removing the top of the unit. The fuse is located next to the transformer.

Opening the P9000 Case

If the case needs to be opened to gain access to the fuse, or for any other reason,the following procedure should be used. Disconnect the machine from the mains supply. Remove the six crosshead screws - 3 along the front edge and three along the back face.

Lift the rear of the top off the base and then slide the top forwards, taking care to clear the keyboard and the LEDs.

The reverse procedure should be used to reassemble the unit.

Using the Machine - Points to Note

To ensure trouble free operation, please observe the following points:

a/ Operate the machine on a vibration free surface

b/ Do not locate the machine near any source of heat or in direct sunlight

c/ Ensure no metal parts can fall into the machine

d/ Disconnect from the mains supply when not in use

e/ DO NOT switch the machine on or off with EPROM devices in the ZIF sockets

f/ Check the device type setting when inserting EPROMs into the ZIF sockets

g/ Periodically clean the ZIF sockets with a stiff bristle brush to ensure good contact

h/ Never force an EPROM into or out of ZIF sockets - they are zero insertion force sockets

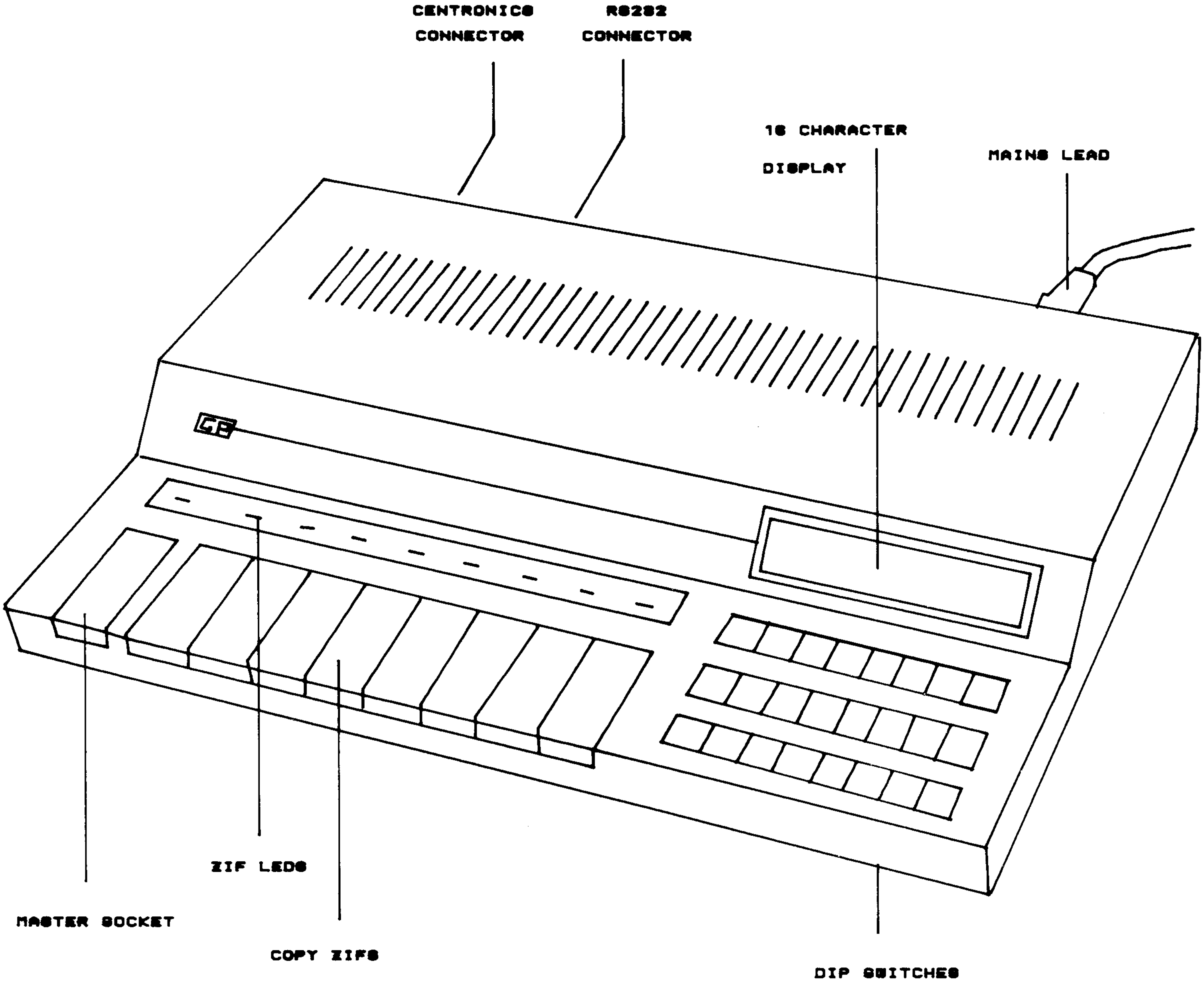


FIGURE 2. LAYOUT FOR THE PS030

Discrete LED Indicators

Each copy ZIF has associated with it a corresponding ZIF LED. These are used to indicate a mode PASS/FAIL as described in the table below:

ZIF LED	Status
On	Socket powered down, mode PASS
Off	No information
Flashing	Socket powered down, mode FAIL

When any key is depressed after a mode cycle, all ZIF PASS/FAIL information is lost.

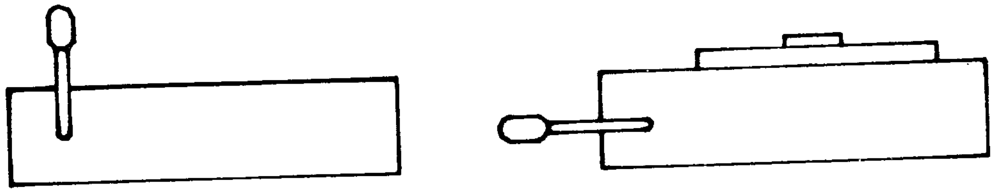
16 Character Fluorescent Display

This is divided into two sections.
A device type display and a mode display.
Apart from its use for setting up the MODE and EPROM type, it also displays messages as described below:

Message	Meaning
P9000	machine has just been powered up, has completed a self test successfully. Mode and Device have defaulted to CHECKSUM - 2764N.
BUSY	machine is actioning a mode cycle. Also indicates when the ZIF sockets are powered up. Any other message on display, ZIF sockets are powered down
DONE	indicates completion of selected mode
FAIL	means mode could not be completed due to a copy EPROM failure to PASS a pre-check
ERROR	indicates a fault has occured during a mode cycle. The nature of the fault is shown by the fault code in the mode display

Zero Insertion Force Sockets (ZIFs)

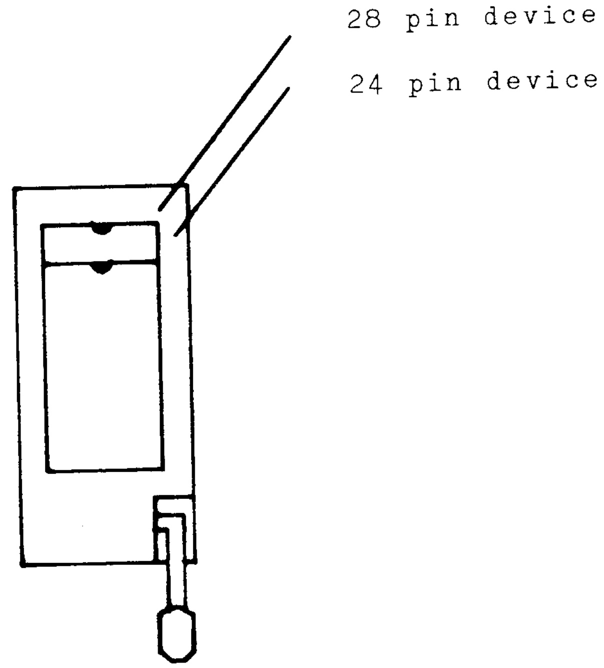
Each socket is zero insertion force type and will give reliable service provided they are kept clean and are used in the proper way. The diagram below shows the correct way to load a PROM into the socket.



Lever open, insert or remove PROM

Lever closed, PROM is firmly held

The ZIFs are designed to accomodate both 24 pin and 28 pin PROMs. The diagram below illustrate correct PROM orientation.



DEVICE TABLE 1

Device Menu	2508	2758A	2758B	2716	2815	2816	48016	9716	2532	2732	2732A	2564
Manufacturer												
AMD				2716DC						2732DC	2732ADC	
EUROTECHNIQUE				ET2716						ET2732		
FUJITSU				8516						MBM2732	MBM2732A	
HITACHI				HN462716			HN48016		HN482532	HN482732		
INTEL		2758A	2758B	2716	2815	2816						
MITSUBISHI				M5L2716K						M5L2732K		
MOTOROLA				MCM2716 MCM27A16					MCM2532			
NATIONAL				MM2716 NMC27C16		NMC2816		NMC9716	NMC2532	NMC2732 NMC27C32		
NEC				UPD2716D						UPD2732D	UPD2732AD	
OKI				2716							2732A	
ROCKWELL						R5213				R87C32		
SEEQ												
SGS						2816A 5516A						
TEXAS INST	TMS2508								TMS2532	TMS2732		TMS2564
TOSHIBA										TMM2732D		

This table has been compiler from manufacturers data and is correct to the best of our knowledge

DEVICE TABLE 2

Device Menu	2764N	2764I	2764A	2764Q	27128N	27128I	27128A	27128M	27128Q	27256I	27256H	27256Q	27512I	27512M
Manufacturer														
AMD	2764DC		AM2764A			27128DC		AM27128ADC						27512DC
EUROTECHNIQUE	ET2764													
FUJITSU				MBM2764 MBM27C64					MBM27128			MBM27C256	MBM27C512	
HITACHI	HN27C64 HN482764				HN4827128						HN27256			
INTEL	2764		2764A			27128	27128A			27256			27512	
MITSUBISHI		M5L2764K												
MOTOROLA														
NATIONAL			NMC27C64				NMC27C128			NMC27C256		NMC27C256*	NMC27C512	
NEC	UPD2764D UPD27C64					UPD27128D								
OKI	MSM2764RS													
ROCKWELL	R2764 R87C64													
SEEQ		2764				27128								
SGS	M2764													
TEXAS INST	TMS2764													
TOSHIBA	TMM2764D				TMM27128D									

This table has been compiled from manufacturers data and is correct to the best of our knowledge

(Y) = Intelligent Identifier
* First batches of NMC27C256 have no identifier

EPROM Selection

The P9000 must be set up to correspond to the particular type of EPROM to be programmed. The device type is selected using the two arrow keys and the DEVICE SELECT key.

By depressing the DEVICE SELECT key, the machine will display the current EPROM and MODE selected. (The P9000 defaults to 2764 at power on). The DEVICE display decimal points flash to indicate that the machine is ready for selection.

Depressing either of the arrow keys will step the DEVICE display through the EPROM list, either up or down. Once the required device type appears in the window, no further action is required for selection.

To select the correct device from the device menu, refer to the two tables listed overleaf.

PROM Manufacturers are listed on the left hand side of the page, and their respective devices are listed to the right. The correct selection for the P9000 is listed at the top of the page in the line labelled Device menu.

Some devices are apparently duplicated in the device menu E.g. 2764N, 2764I, 2764A, & 2764Q.
The suffixes (N, I, A, or Q) refers to the programming method required by those devices as stipulated by the EPROM manufacturers -

N = Normal program (50ms pulse)
I = Intelligent Programming
A = INTEL 'A' version of standard part
Q = Fujitsu Quickpro programming method

It is important to match the P9000 with the devices you are programming - E.g. a 2764A does NOT program in the same way as a 2764. **Damage to the devices, or inadequate programming may result if an incorrect setting is used**

Mode Selection

By depressing the mode key, the P9000 will display the current mode and PROM type. (The machine defaults to CHECKSUM when AC power is first applied.) The mode display decimal points flash to indicate the machine is ready for selection. Depressing either of the arrow keys will step the display up or down through the mode menu.
Once the required mode appears in the display, no further action is required for mode selection.
The table below briefly describes all the possible modes.

MODE DISPLAY	DESCRIPTION
CHECKSUM	Checksum master and compare with copy sockets
PROGRAM	Program using master data
VERIFY	Compare master and copy devices
CRC VALUE	Calculate CRC of master and compare with copy
BIT CHECK	Check for programmability
ERASE	Chip erase EEPROMs
SELF TEST	Execute self test
BLANK CK	Check copy sockets are blank
*SIN PRG	Program using data from serial port
*SOUT	Send data to serial port
**SIN LOAD	Load RAM with serial data
INVERT	Invert all RAM data
**REMOTE	Enter Remote control operation

* Not on P9010
** Not on P9020
A detailed description is given later in this section

START/STOP

Once the required device type and mode have been selected, the START key can be pressed to start the cycle.
Prior to actioning the selected mode. E.g. blankcheck,the P9000 will power up all the ZIF sockets and perform a socket test to check for misplaced or reverse inserted PROMs.
During any mode function, the machine will continually monitor the power supply, ports, and check for correct address and data set-up.
Any faults encountered will make the P9000 stop the cycle, power down the ZIF sockets and display an error message - see ERROR MESSAGES.
The STOP key can be used at any time to stop a mode cycle and return to normal operation.

DESCRIPTION OF MODES

CHECKSUM

Calculates the checksum of the master device and then compares each copy socket with the Master.

PROGRAM

The machine checks for programmability (illegal bit check) using data from the Master device, then programs the copy devices, then compares each device with the master PROM, and displays the Master checksum. During programming a decrementing count is displayed in order to give an indication of the time left to complete the cycle.
NOTE: Any EEPROMs which fail the illegal bit check will automatically be erased and re-checked before programming.

VERIFY

Compares the master data with each copy socket. Any PROM not containing identical data is indicated by a flashing ZIF LED.

CRC VALUE

Executes a complex algorithm to produce a unique number to describe the master PROM. The CRC value is similar to the checksum but is more secure.

BIT CHECK

This is the illegal bit check or test for programmability.
It checks that all the bits in the copy devices can be set to the required pattern stored in the master. A programmed '0' cannot be set to a '1' without exposure to UV light (EPROMs) or electrical erasure (EEPROMs).

ERASE

Erases the selected EEPROM. The device type selected must be an EEPROM - any other selection will give an error message

SELF TEST

The machine tests its ROM, RAM, ports, power supply, display and ZIF LEDs.

BLANK CHECK

Reads through all the devices (except the Master) and checks that all address locations contain hex FF. At the end of the cycle, non blank PROMs are indicated by flashing LEDs.

SIN PRG (P9020, P9030 only)

Programs the copy devices with data received from the RS232C Serial port (full details are given in a later section).

SOUT (P9020, P9030 only)

Outputs RAM data to the serial port. Format and speed are the same as for serial in and are set up on the parameter DIL switches.

SIN LOAD (P9030 only)

Receives data from the RS232 and loads it into the RAM

INVERT

Inverts all the data in the RAM. The function is useful for microsystems where PROMs are driving inverting (faster) buffers.

REMOTE (P9030 only)

This is the remote control facility. Commands can be sent to the P9030 from the Serial port to provide an interactive programming system.
(Further details are given in the REMOTE control section of the manual).

ERROR MESSAGES

During the course of normal operation the P9000 will generate Error messages in the event of a system, PROM, or operator error. Error messages are detailed below:

ERROR NUMBER	MESSAGE	MEANING
	PSU ERROR	Power supply error- Error number gives details
1	PSU ERROR 00	Vpp error, 21v
2	PSU ERROR 01	Vcc error, 6v (intelligent devices)
3	PSU ERROR 02	12v error, erase supply,RS232 +12v
4	PSU ERROR 03	Vpp error, 12v (intelligent devices)
5	PSU ERROR 04	Master Vcc error
6	PSU ERROR 05	Vpp error, 25v
7	PSU ERROR 06	Copy supply, 5v Vcc
8	PSU ERROR 07	Vpp error, 5v
	PORT ERROR	ZIF or system port error, error number gives details. The first two digits tell where the error is, the last two digits give the actual bits in error
9	PORT ERROR E1xx	Master ZIF data lines
10	PORT ERROR E5xx	Logic drive to ZIFs - pin1, 23 and A13, (pin 26)
11	PORT ERROR EAxx	High voltage drivers
12	PORT ERROR ECxx	ZIF copy data bus (driver)
13	PORT ERROR EDxx	ZIF address high A8 - A15 (reader)
14	PORT ERROR EExx	ZIF address low A0 - A7 (reader)
15	PORT ERROR F0xx	ZIF copy data bus (reader)
16	PORT ERROR F1xx	ZIF address high (driver)
	PORT ERROR F2xx	ZIF address low (driver)
17	**PROM ERROR**	(ZIF power up test) device is upside down or misplaced

NOTE:

1/ Vpp errors (1,4,6,8) may occur during the course of programming and are usually associated with 1 or more copy socket devices drawing excessive current from the Vpp source. If persistent errors occur (i.e. with devices not in the sockets) then the machine may need re-calibrating.

2/ Vcc errors may occur if an upside down device fails to be detected by the 'power on socket test'.

3/ Port errors - The driver ports to the ZIF are constantly checked to ensure that address and data information has been properly set up. Errors 12,13,14,15 and 16 occur when a device in the ZIF sockets is faulty. E.g. shorted pin or is upside down or misplaced.

4/ Port errors - if a port error occurs whose number does not appear in the table, then an internal system fault has occured.

Production Programming - Recommended Procedure

The following is a general guide to programming with the P9000 series programmers.

1/ When a master is first created, take a checksum and note this on a label affixed to the device.

2/ Store masters in conductive, non-corrosive foam. (Some conductive foams are corrosive and will eventually give problems due to unreliable ZIF contact.

3/ Always set up the P9000 to match the type of device you are using - refer to the PROM cross reference chart.
Special Note: Some EPROMs of the same type often have different programming requirements - E.g. the 2764 is available in three different versions, so make sure the correct device type is selected.

4/ With the ZIF lever up, position the master PROM into the master ZIF taking care to ensure it is properly seated.
Close the ZIF lever - lever down, horizontal, parallel with the surface of the socket.
(When the master is first inserted at the start of a prodcution run, it is a good idea to CHECKSUM the device to ensure the P9000 is reading it correctly.)

5/ Select PROGRAM from the mode menu and press STOP.

6/ Load the required number of devices (1-8) into the copy ZIF sockets - make sure they are seated properly. If a socket does not contain a device it will always appear to the P9000 as a blank device.

7/ Press START to commence programming. During programming a count is decremented and displayed to give an indication of the time remaining to complete the cycle.

8/ When programming is complete the display will show PROGRAM = xxxx, where xxxx is the checksum of the Master device. Make sure this corresponds to the Master checksum (on its label) before removing the copy devices and affixing labels. (If the checksum is different then the P9000 is not reading the master correctly - so remove it, clean its pins, replace into the Master socket and repeat the programming procedure).

9/ At the end of a production run, quickly brush the ZIF sockets with a stiff antistatic bristle brush to remove any dirt.

NOTE: In order to ensure continued reliable programming, it is of paramount importance that the ZIF sockets are kept clean at all times.

P9000 Serial Data Transfers

Introduction

The P9030 has a bidirectional RS232C port as standard. This port may be used to receive data for device programming from a host computer, transmit data to a host computer or printer, or used as a communications link to an RS232C terminal for remote operation.

The P9020 has a bidirectional RS232C port as standard. This port may be used for transmission and reception of device data.

The RS232C port will support transmission/reception baud rates between 110 and 19K2 baud. The data may be received in any one of 15 formats, and transmitted in 16.

These formats are:

1	MOS TECHNOLOGY	
2	SIGNETICS ABSOLUTE	
3	TEKTRONIX HEXADECIMAL	
4	BINARY	
5	DEC BINARY	
6	ASCII HEX COMMA	
7	ASCII HEX APOSTROPHE	
8	ASCII HEX PERCENT	
9	ASCII HEX SPACE	
10	BIOF	
11	BHLF	
12	BPNF	
13	LIST	(Output only)
14	MOTOROLA EXORCISER	
15	INTEL HEX	
16	GP BINARY	

These formats are all described in detail in Appendix A.
A full specification of the serial will be found in the appendix.

The speed, format and word format for the serial transmission is selected by means of the DIP switches located on the base of the machine. The P9000 reads the switches each time the serial mode is selected, so that it is necessary to have the correct format selected before each data transfer. The setting of the switches is shown on the following page.

DIP Switch Table for Serial Data Transfers

Condition SW No:	Off	On	Remarks
SW 10	0	1	Baud rate
SW 9	0	1	Baud rate
SW 8	0	1	Baud rate
SW 7	0	1	Format
SW 6	0	1	Format
SW 5	0	1	Format
SW 4	0	1	Format
SW 3	7 bits	8 bits	No of bits in data word
SW 2	Parity off	Parity on	Parity on/off
SW 1	Parity odd	Parity even	Parity odd/even

<u>Serial Transfer Baud Rate</u>
This is selected with SW 10, SW 9, SW 8 as follows:

Baud Rate	SW10	SW9	SW8
110	1	1	1
300	1	1	0
600	1	0	1
1200	1	0	0
2400	0	1	1
4800	0	1	0
9600	0	0	1
19k2	0	0	0

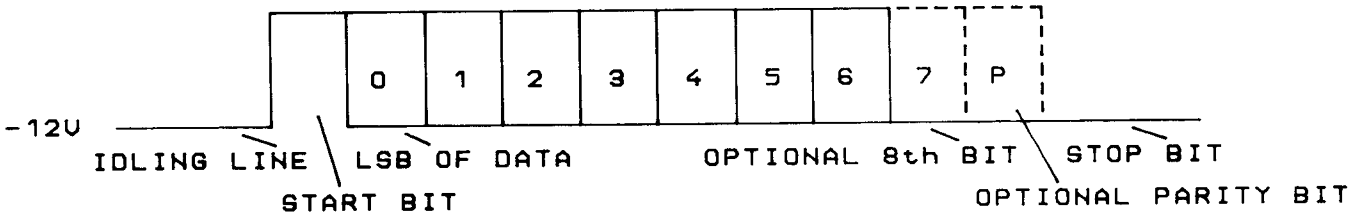
<u>Serial transfer Format</u>
This is selected with SW 7, SW 6, SW 5, SW 4 as follows:

Format	SW7	SW6	SW5	SW4
MOSTEK	0	0	0	0
SIGABS	0	0	0	1
TEKHEX	0	0	1	0
BINARY	0	0	1	1
DEC BIN	0	1	0	0
ASC ","	0	1	0	1
ASC "'"	0	1	1	0
ASC "%"	0	1	1	1
ASC SP	1	0	0	0
B10F	1	0	0	1
BHLF	1	0	1	0
BPNF	1	0	1	1
LIST	1	1	0	0
MOTOROLA	1	1	0	1
INTEL	1	1	1	0
GP BINARY	1	1	1	1

Word Format The P9000 word format is:

START BITS	:	1
STOP BITS	:	2 OUT 1 OR MORE IN
DATA BITS	:	7/8
PARITY	:	ON/OFF/ODD/EVEN

The Serial Word:



Handshaking: The P9000 uses hardware handshaking via CTS/DTR and DSR/RTS. When the P9000 is receiving, the DTR and RTS line (pin 20 and pin 4) must be used to control the data flow into the programmer.
A high level (+12v) on the RTS & DTR line indicates ready to receive. A low level (-12v) indicates not ready.
Before the P9000 will output data, the input handshake lines CTS and DSR, (pins 5 and 6) must taken to a high level (> 3v).
If a handshake line changes state during a byte, the P9000 expects the transfer to continue until the end of the next stop bit.
At speeds below 2400 baud it is not necessary for the sending machine to respond to the hardware handshake.

Operation of the Serial Transfer Modes for the P9020

SOUT This is the serial output mode for the P9020. It outputs the contents of the master device to the RS232C port in the format selected by the DIP switches.
The function will always output the entire MASTER contents and will only stop if the handshake line indicates that the receiving machine is not ready to receive, or if the STOP key is pressed.
The only error message which can be generated by the serial transfer is a "TIMEOUT ERROR" if the handshake line remains at not ready for too long.

SIN PRG This is the serial input and serial programming mode. When this mode is selected the P9020 will perform a blank check on the ZIF sockets. If there are any programmed devices in the sockets, the P9020 will stop and display "FAIL BLANK CHECK". If all sockets pass, the P9020 will display "BUSY SIN PRG" and set the handshake line to ready.

Operation of the Serial Transfer Modes for the P9030

The Serial Out mode

When selected the P9030 prompts for a Start Address which is the address of the first byte to be output. It then asks for a length which is the number of bytes to be output by the P9030. If the data format specified by the DIP switches is one which includes addresses in the data records then the P9030 will also prompt for an Offset address. This address will be added to the address in each record so that the data will end up in the desired position within the machine which is receiving the data from the P9030. Once the last prompt has been satisfied the P9030 display "BUSY SOUT". This message will be left displayed until either the data has all been transmitted or a time-out error occurs. It must be noted that the P9030 will only transmit data from the RAM. It is therfore necessary to ensure that the data which is to be sent is in the RAM prior to starting the data transmission.

The Serial Load and Programme Modes

These two modes operate in virtually the same way and so will be dealt with together. The only difference between the two modes is that in Serial Programme mode (SINPRG) the P9030 programmes Devices with the incoming data, where as in Serial Load mode (SINLOD) the data is just loaded into the Ram. The only other difference is a consequence of this and is that SINPRG allows amounts of data larger than the Ram to be input. This is not possible in SINLOD mode.

Once the mode is selected the P9030 will prompt for different parameters depending on which format is selected:
If the selected format is one of:

MOS TECHNOLOGY
SIGNETICS ABSOLUTE
TEKHEX
MOTOROLA EXORCISER
INTEL HEX

the P9030 will prompt for an offset address. This offset address is in effect the base address of the device into which the data will be placed. The offset address is subtracted from the addresses within the data records, before they are loaded into the system. For these formats the offset is the only parameter required.

If the selected format is one of the other formats:

BINARY
DEC BINARY
ASCII SPACE, APOS,
COMA, PERCENT
BPNF
BHLF
B10F
GP BINARY

then the P9030 will prompt for a start address, this being the address at which the first byte is to be loaded into memory. Additionally DEC BINARY and BINARY formats will request a length parameter, indicating the number of bytes which the

P9030 is to load.

Once the P9030 has received all necessary parameters it will perform a Blank Check on the Zif sockets. If any socket contains a programmed device then the P9030 will display "FAIL BLANK CHECK". If all devices are blank then the P9030 will display "BUSY SINPRG".
Eventually if no errors, the message

"BUSY PROGRAM" will appear indicating that the P9030 is programming the received data into the devices.

When all data has been loaded the P9030 will display "DONE SINPRG" and give pass fail information on the LED's.

In the event of an error the P9030 will give a warning beep and display information in accordance with the section on serial errors. In addition, if for any reason the P9030 fails to programme all 8 devices (e.g. no devices correctly programmed) the transfer will be aborted and the message "FAIL VERIFY" will be displayed.

P9030 RAM EDITING FACILITIES

This section gives a detailed description of the P9030 editing facilites, taken one key at a time. Examples are given on the use of each key by itself, and in conjunction with other keys. The table below gives a list of the available RAM editing functions:

KEY	DESCCIPTION
MEM	Define memory address for edit mode
INSERT	Insert data at address
DELETE	Delete data at address
FILL	Fill block with same data value
COPY	Copy source block to destination
SEARCH	Find occurance of data strings
CHANGE	Change data string(s) to new string
SPLIT	16 bit to 8 bit split
MERGE	8 bit to 16 bit shuffle
ENTER	Load HEX entry from display buffer
UP ARROW	Increment address
DOWN ARROW	Decrement address
STOP	Return to normal mode
CODELOCK	Lock or unlock RAM editor (and special function keys)

STOP

This will stop any function and return the machine to normal mode - I.e. ready to accept new keyboard commands. The display now shows the current mode and device selected from the menus. Subsequent stops will have no effect.

HEXADECIMAL KEYS (0123456789ABCDEF)

These lower case keys are only enabled when the P9030 requires entry of hexadecimal data, otherwise they are not directly accessible.

CODELOCK

This useful command will lock out the RAM editor and special function keys to prevent accidental use or use by unauthorised personnel. A 2 digit code is required to lock the keyboard, and this same code must be entered in order to unlock it.

Example: Lock and unlock the keyboard with code AA.

KEYPRESS	DISPLAY	MEANING
CODELOCK	UNLOCKED	Keyboard is unlocked, Enter code AA
AA	UNLOCKED AA	Code AA is entered
ENTER	LOCKED	Keyboard now locked
STOP	CHECKSUM 2764	Return to normal mode, RAM disabled
CODELOCK	LOCKED	Keyboard is locked, enter code
AA	LOCKED AA	Unlock code is AA
ENTER	UNLOCKED	Keyboard is unlocked
STOP	CHECKSUM 2764	Return to normal mode, RAM enabled

If the wrong unlock code is entered, the beeper will sound continuously and the display will show an ILLEGAL ACCESS message - Press STOP before entering the correct unlock code. When the keyboard is locked, the P9030 will only respond to the CODELOCK, DEVICE SELECT, MODE SELECT, UP, DOWN ,STOP and START keys.

ENTER

This key is used during the course of hexadecimal data entry. E.g. address and data information, Fill parameter, codelock word.

When the hex keys are pressed, the values are loaded into a variable length buffer depending upon the number of digits required by the P9030 (E.g. 4 digits required for address, 2 digits required for data). You can 'overflow' the buffer by pressing as many hex keys as you like - the P9030 will only recognise the last 4 digits you pressed, in the case of address entry, or 2 digits in the case of data entry.
This means that mistakes can be rectified simply by continuing to enter hex digits until the displayed value is what you want.
The P9030 will only act on the data once the 'ENTER' key has been pressed.

Example: View data at address 1234

KEYPRESS	DISPLAY	MEANING
MEM	ADDRESS	Prompt for address
1255	ADDRESS 1255	Made a mistake so keep entering
1234	ADDRESS 1234	Address now correctly entered
ENTER	@1234 FF EDIT	Data at 1234 is FF, Edit mode selected

Note: If ENTER is pressed without entering any hex values at all, the P9030 will substitute 0 as the value.

Example: Change data at addresses 0000,0001,0002 to 00

KEYPRESS	DISPLAY	MEANING
MEM	ADDRESS	Prompt for address entry
ENTER	@0000 FF EDIT	Data at 0000 is FF
ENTER	@0001 FF EDIT	Data changed to 00, skip to next address
ENTER	@0002 FF EDIT	Data changed to 00, skip to next address
DOWN ARROW	@0001 00 EDIT	Review previous address
DOWN ARROW	@0000 00 EDIT	Review previous address

MEM (Memory address)

This defines a RAM address for data modification.
Any address can be defined, provided it is within user RAM.
The base address of the RAM is always 0000, the last address depends upon the amount of RAM in your machine.
For a P9030 equipped with 8k bytes, the last address is 1FFF.

Example: View data at RAM address 0000

KEYPRESS	DISPLAY	MEANING
MEM	ADDRESS	Prompt for address
0000	ADDRESS 0000	Enter address
ENTER	@0000 FF EDIT	Data is FF, edit mode selected

When edit mode is selected, the P9030 will only respond to the ENTER key (enter data), the STOP key (terminate edit mode) and the HEX keys.

Example: Change data at address 1000 to 3A.

KEYPRESS	DISPLAY	MEANING
MEM	ADDRESS	Prompt for address
1000	ADDRESS 1000	Enter address
ENTER	@1000 FF EDIT	Data is FF, edit mode selected
3A	@1000 3A EDIT	Enter 3A as new data
ENTER	@1001 FF EDIT	Data is entered, skip to next address
DOWN ARROW	@1000 3AEDIT	Move back to view previous address

The UP & DOWN ARROW keys can be used to move through memory as required.
NOTE: If the 'ENTER' key is pressed when the P9030 requires HEX entry, but no such entry has been made, then the data entered is 0. (See ENTER function explanation).

COPY

This command will copy blocks of data within the RAM. When a copy has been completed, the source data has not been changed, but has been duplicated at the destination address.

The copy command is 'intelligent' in that it will not overwrite data provided that the source block and destination block do not overlap.

NOTE: If the destination block overlaps the source block, then a complete and accurate copy of the source block will be made, but the overlapped part of the source block will be overwritten.

Example: Copy the block from 0000-0800 to the area starting at address 1000

KEYPRESS	DISPLAY	MEANING
COPY	COPY @	Prompt for source start address
0000	COPY @ 0000	Enter start address
ENTER	COPY @ 0000-	Prompt for source end address
0800	COPY @ 0000-0800	Enter end address
ENTER	0000-0800 TO	Prompt for destination start address
1000	0000-0800 TO 1000	Enter destination address
ENTER	DONE	Operation complete

To return to normal mode, press STOP.

FILL

Memory Fill is used to fill all or part of the RAM with a specified value.

Example: Fill the RAM block 0123-0234 with OA

KEYPRESS	DISPLAY	MEANING
FILL	FILL @	Prompt for start address
0123	FILL @ 0123	Enter start address
ENTER	FILL @ 0123-	Prompt for source end address
0234	FILL @ 0123-0234	Enter end address
ENTER	0123-0234 WITH	Prompt for FILL pzrameter
OA	0123-0234 WITH OA	Enter parameter
ENTER	DONE	Fill complete

SPLIT

This divides the whole RAM in half, and stores all data at even address in the lower half and all data at odd addresses in the top half of the RAM. The effect is that if 16 bit data had been loaded into the RAM (from the serial port) it can be split so that 2 EPROMs can be programmed: one containing the data at even addresses, the other containing data at odd addresses.

MERGE

This is the converse of SPLIT.
The effect of Merge is to interleave the data in the top half of the RAM with the data in the lower half I.e. a 16 bit to 8 bit shuffle.

When this command is invoked, the entire RAM (including any expansion RAM) is merged.

SEARCH

The P9030 can be made to search the RAM for the occurance of a data string. Three parameters need to be entered. These are:
-The address at which the search will start
-The number of bytes in the string to be found
-The data comprising the string

The search begins at the start address and proceeds until a match is found with the specified string data, the display now showing the first address of the first occurance of the string. Use the UP ARROW key to find subsequent occurances of the string.

Example: Search the RAM for the data string 30 31 starting at 0000.

This example assumes that the RAM is filled with FF apart from 2 strings of 30,31 at addresses 0432 and 18EE.

KEYPRESS	DISPLAY	MEANING
SEARCH	FIND @	Prompt for start address of search
0000	FIND @ 0000	Enter the search start address
ENTER	LENGTH	Prompt for the string length
02	LENGTH 02	Enter the string length
ENTER	FIND @ 01 D	Prompt for first byte in the string
30	FIND @ 01 D30	Enter the first byte
ENTER	FIND @ 02 D	Prompt for the 2nd byte of string
31	FIND @ 02 D31	Enter the 2nd byte
ENTER	SEARCHING	Delay whilst P9030 searches
	@0432 30	First string has been found at 0432
UP ARROW	@18EE 30	Use the ARROW key to find the next data string at 18EE
UP ARROW	NOT FOUND	No more strings found

Note: The maximum string length that can be searched for is 255 bytes.

CHANGE

Change a data string to a new data string. Any number of occurances of a data string can be found (see SEARCH) and changed to a new string.

<u>Example:</u> Change the data strings at 0432 and 18EE to 42, 43. This example assumes that the RAM is filled with FF except for 2 strings of 30, 31 at addresses 0432, 18EE. (See the example in SEARCH).

KEYPRESS	DISPLAY	MEANING
CHANGE	FIND @	Prompt for start address of search
0000	FIND @ 0000	Enter the search start address
ENTER	LENGTH	Prompt for the string length
02	LENGTH 02	Enter the string length (02)
ENTER	FIND @ 01 D	Prompt for first byte in the string
30	FIND @ 01 D30	Enter the first byte
ENTER	FIND @ 02 D	Prompt for the 2nd byte of string
31	FIND @ 02 D31	Enter the 2nd byte
ENTER	CHNG @ 01 D30 T	Prompt for 1st byte of new string (CHANGE FIRST BYTE, 30, TO)
42	CHNG @ 01 D30 T42	Enter new first byte
ENTER	CHNG @ 02 D31 T	Prompt for 2nd byte of new string
43	CHNG @ 02 D31 T43	New 2nd byte is 43
ENTER	SEARCHING	All data has been entered, so search begins
	@ 0432 42	1st string at 0432 has been found and changed to the new string
UP ARROW	@ 18EE 42	The next occurance of the string has been found and changed
UP ARROW	NOT FOUND	No more strings occur in the RAM

<u>NOTE:</u> The maximum string length that can be changed is 255 bytes.

DELETE (See also INSERT)

Deletes any byte in the RAM provided there are at least 5 bytes of free space above the delete address. Once the delete address has been entered, the P9030 will search from the address to the top of the RAM for the occurance of at least 5 bytes set to FF. Once this has been found, it will discard the byte at the delete address (it will be lost).
The data block between the delte address and the free space is now shifted down one address and an FF is added to the block of free space.

Example: Delete data at 0005
This example assumes the RAM is completely filled with FF except for a data block at 0000 - 0007.

KEYPRESS	DISPLAY	MEANING
DELETE	DELETE @	Prompt for delete address
0005	DELETE @ 0005	Enter address
ENTER	@ 0005 00 INS/DEL	Data has been deleted, the data at 0006, 0007 has been shifted to 0005, 0006
DELETE	@ 0005 CC INS/DEL	Data at 0005 is deleted again, the data at 0006 (CC) has been shifted to 0005.
DELETE	@ 0005 FF INS/DEL	All data from the delete address to the top of RAM is now at FF
DELETE	NO FREE SPACE (beep)	Delete now no longer possible

INSERT (See also DELETE)

Inserts a free byte (FF) at any address in the RAM. The P9030 searches the RAM starting at the specified insert address, for the occurance of 5 unused bytes (5 bytes of FF). If free space is found, the 1st byte of FF is shifted back through the intervening data to the insert address. The data at the insert address can then be modified using the (MEM) edit function. Once the Insert mode has been entered, the delete key is also enabled. Delete works in a similar way but shifts the free byte from the insert address back to the first block of 5 free bytes it can find. If there are no free bytes or the RAM is completely cleared, a 'NO FREE SPACE' message is displayed.

Example: Insert then delete a byte at address 0010
This example assumes that the RAM is completely filled with data except for 6 free bytes starting at address 0200.

KEYPRESS	DISPLAY	MEANING
INSERT	INSERT @	Prompt for insert address
0010	INSERT @ 0010	Enter address
ENTER (beep)	@ 0010 FF INS/DEL	FF has been inserted at 0010, there are now only 5 free bytes starting at 0201, intervening data block has been shifted up one address
DELETE (beep)	@ 0010 00 INS/DEL	The data block has been shifted back by one address. (The first byte of the block is 00). There are now 6 free bytes starting at address 0200.

Press STOP to return to normal mode
In the example given above, no data has been lost or added: The delete key could have been pressed until all the data from 0010 to the top of the RAM was cleared: because there were 6 free bytes, only two insertions could be made (provided the Delete function had not created more free space).

P9030 Special Function Keys

These are very powerful commands which operate on the RAM, but in conjunction with the Master and Copy devices.
Typical uses include:-

1/ Transferring from one device to another E.g. from a 2532 to a 2732
2/ Combining a PROMset into a single device E.g. combine data from two 2732's into a single 2764
3/ Transferring data from a single device into smaller devices E.g. transfer from a 2764 into four 2716's
4/ Combine Master data and serial data to create a new Master
5/ Send Master data to the Serial Port
6/ Comparing sections of one or more Masters with the RAM or each other
7/ Calculating the checksum and CRC of blocks of RAM or Master

The table below briefly describes the Special Function Keys (a detailed explanation is given later in this section).
Each function operates on a user-defined block of data in the RAM and master & copy sockets. The result of the operation is shown on the ZIF LEDs in the usual way:
PASS = LED lit: FAIL = LED flashing

Key	**Description**
CSUM	Checksum RAM block & compare with the master and copy block
CRC	Calculate CRC of RAM block & compare with master and copy
STORE	Copy master block to specified RAM address then verify
PROG	Program copy starting at specified address with RAM block
VERIFY	Compare master and copy block with RAM block
IBC	Illegal Bit Check Master and copy block with RAM block

Each of the functions mentioned above operates using a PROM start address, a RAM start address and a block length - the P9030 prompts for each of these as required:

Note: 1/ If the defined length of the block is larger than the RAM size in your machine, it will be rejected and requested again

2/ If the RAM start address is outside the RAM it will be rejected and requested again

3/ If the PROM start address is outside the range of the PROM it will be rejected and requested again

PROG (Program)

Programs the copy sockets with data from the RAM block after performing an illegal bit check to check for programmability. Once programming is complete, all the devices are verified against the RAM block and the result indicated by the ZIF LEDs.

EXAMPLE: Program the copy socket devices starting at address 0034 with RAM data block at 1100-1134

KEYPRESS	DISPLAY	MEANING
PROG	ROM START @	Prompt for ROM start address
0034	ROM START @ 0034	Enter start address
ENTER	RAM START @	Prompt for RAM start address
1100	RAM START	Enter start address
ENTER	LENGTH	Prompt for block length
34	LENGTH 34	Enter length
ENTER	BUSY PROGRAM	Start program sequence
	DONE PROGRAM	Programming complete
	FAIL VERIFY	Programming complete but no device programmed

STORE

Copies data from the MASTER socket into the user-defined RAM block then verifies the RAM block with both the Master & Copy sockets.

EXAMPLE: Store data from the Master, block limits 0100-01FF to the RAM starting at address 1000

KEYPRESS	DISPLAY	MEANING
STORE	ROM START @	Prompt for ROM start address
0100	ROM START @ 0100	Enter ROM start address
ENTER	RAM START @	Prompt for RAM start address
1000	RAM START @ 1000	Enter RAM start address
ENTER	LENGTH	Prompt for block length
200	LENGTH 200	Enter length
ENTER	BUSY STORE	Stores the data block & verifies
	DONE STORE	Store function complete, store successful

CSUM (Checksum)

Calculates the 2 byte checksum of any length RAM block and then compares this block with the Master and copy blocks. Flashing LEDs show when the RAM and PROM blocks are different.

The checksum is the 16 bit addition of all the bytes in the block. The carry from the 16th bit is discarded to give a 2 byte value.

Example: Calculate the checksum for the RAM block 0000-1FFF and compare with the Master and Copy over the same range

KEYPRESS	DISPLAY	MEANING
CSUM	ROM START @	Prompt for ROM start address
0000	ROM START @ 0000	Enter ROM start address
ENTER	RAM START @	Prompt for RAM start address
0000	RAM START @ 0000	Enter RAM start address
ENTER	LENGTH	Prompt for block length
2000	LENGTH 2000	Enter length
ENTER	BUSY CHECKSUM	Calculates checksum value of RAM block
	CHECKSUM = E000	Display result (E000 in this case)

IBC (Illegal Bit Check)

Performs an Illegal Bit Check on both the Master and Copy sockets using RAM data over a specified address range.
The IBC is a test for programmability - it checks that all bits in the device can be set to the required pattern in the RAM. A programmed '0' cannot be set to a '1' without exposure to UV light (EPROMs), or electrical erasure (EEPROMs).

Example: Illegal bit check an entire device with RAM data at 1000-1800. (Select the required device type from the Device Menu).

KEYPRESS	DISPLAY	MEANING
IBC	ROM START @	Prompt for ROM start address
0000	ROM START @ 0000	Enter ROM start address
ENTER	RAM START @	Prompt for RAM start address
1000	RAM START @ 1000	Enter RAM start address
ENTER	LENGTH	Prompt for block length
800	LENGTH 800	Enter length
ENTER	BUSY BIT CHECK	IBC on Master and copy sockets using RAM data
	DONE BIT CHECK	IBC complete result on ZIF LEDs

Devices which FAIL the IBC are shown by a flashing ZIF LED, PASS devices are shown by a constantly lit ZIF LED

VERIFY

Compares a user-specified RAM block with the master and Copy sockets. The start address of the block in the RAM and the PROM must be specified along with a block length.
Any device where block data differs from the RAM block data is indicated by a flashing ZIF LED.
If the master device fails to verify then a "VERIFY ERRORS" message is generated. The errors can be viewed by using the UP & DOWN ARROW keys.

Example: Verify a PROM block 0800-09FF against a RAM block 0000-01FF

KEYPRESS	DISPLAY	MEANING
VERIFY	ROM START @	Prompt for ROM start address
0800	ROM START @ 0800	Enter ROM start address
ENTER	RAM START @	Prompt for RAM start address
0000	RAM START @ 0000	Enter RAM start address
ENTER	LENGTH	Prompt for block length
200	LENGTH 200	Enter length
ENTER	BUSY VERIFY	Verify Master and Copy with RAM data
	DONE VERIFY	Verify complete. No errors in the Master
	VERIFY ERRORS	Verify complete, but Master has errors
UP ARROW	SEARCHING	Searching for the first occurance of an error
	0000 M=00 R=FF	1st error is at 0000 In this example Master = 00, RAM = FF

If no verify errors occur, then the function is shown to be completed with the "DONE" message. Any errors that are present can be viewed with the UP & DOWN ARROW keys.

CRC (Cyclic Redundancy Check)

The Cyclic Redundancy Check is a complex algorithm which produces a unique number to 'describe' a block of data. it is similar in many respects to a checksum, but is more reliable as a check value since any change in the data will always produce a new CRC value. (This is not always the case with checksum).

The CRC function will calculate a value for any length RAM block, then perform a Verify (for identical data) between the RAM block and the Master and Copy blocks.

Example: Calculate the CRC for the RAM block 0100-01FF, and compare with ROM block 0000-00FF

KEYPRESS	DISPLAY	MEANING
CRC	ROM START @	Prompt for ROM start address
0000	ROM START @ 0000	Enter ROM start address
ENTER	RAM START @	Prompt for RAM start address
0100	RAM START @ 0100	Enter RAM start address
ENTER	LENGTH	Prompt for block length
200	LENGTH 200	Enter length
ENTER	BUSY CRC VALUE	Calculating the CRC of the RAM block
	CRCVALUE = 826E	CRC value for the block is displayed (826E in this case)

The Print Key

When the print key is pressed the P9030 will display "1=LABELS 2=HEX", this is a request to the user to select which of the two print options is required. The options being a hexadecimal memory dump or the label printing option.

Label Printing

When label printing is selected the P9030 prompts "1=NEW 2=OLD" which is a prompt for the user to decide whether to define a new label or to reprint / modify the old label. On power up the old label buffer is set to nulls (00). If a new label is selected then the buffer is filled with ASCII spaces (20h). In either case the machine will display "CODE XX CHAR Z" where XX is the hex code for the character and Z is the ASCII character represented.
The display device will not display the entire ASCII character set and may use alternativ characters instead. Hence the CHAR information should not be used as anything more than a guide to what will appear in the final message.

New character information should be entered in HEX. Each character may be entered with either the ENTER key or the UP or DOWN arrows. If the ENTER key is used the character pointer will continue to point at the existing character. If the UP / DOWN keys are used the character pointer will be moved backwards or forwards through the message.
Once the message has been built up the GO key should be used to end message entry. Once the GO key has been pressed the complete message will be displayed until the ENTER key is pressed.

Beware if the message is all spaces then the display will appear blank, this is no cause for alarm and pressing ENTER or STOP will rectify the situation.

Once ENTER has been pressed the P9030 will prompt "HOW MANY?" to which the user must reply with the number of labels to be printed. N.B. The labels are supplied in rows of 8, so as not to waste labels the P9030 will always round up the number of labels to the next complete row. Once this number has been entered the P9030 will print the required labels.

The Hexadecimal Memory Dump

The P9030 will prompt for a start and end address for the memory dump. Once both have been entered the P9030 will print the required data .

Operation of the P9030 in remote mode

If the remote option in the manual is selected the control of the P9030 is transffered to whatever device is attached to the RS232 port of the P9030. As soon as the remote mode is selected the P9030 checks the settings of the DIP switch for the serial parameters, configures the serial port appropriately and transmits a command menu. The menu is followed by the system prompt ">", which is displayed whenever the P9030 is ready to receive a command.

All of the P9030 commands are sent as a 3 character string and must be terminated with a carriage return (ODH). The command menu is displayed whenever remote mode is selected and whenever the HLP command is issued. The command menu is listed below:

CHR	Checksum on Ram
CRR	CRC on Ram
STR	Store
PGR	Programme from Ram
VFR	Verify against the Ram
IBR	Illegal bit check against the Ram
CHP	Checksum on Prom
CRP	CRC on Prom
PGP	Programme from master Prom
VFP	Verify against the master Prom
IBP	Illegal bit check against the master Prom
PRT	Print function
MEM	Memory address function
INS	Insert
DEL	Delete
FIL	Fill
CPY	Copy
SRC	Search
CHG	Change
SPT	Split
MRG	Merge
ERA	Erase
STS	Self Test
BCH	Blank Check
SPG	Serial Programme
SOT	Serial Output
SLD	Serial Load
INV	Invert
DEV	Device Select
LOC	Local
HLP	Help Menu
DMP	Dump Memory.

All functions work in the remote mode in the smae way as in the normal operation, with the following exceptions:

1/. All functions which operate on the Zif will return a string of data givng the pass/ fail information .

2/. When a function which has operated on a Zif ends, after returning the pass/ fail data it will not display the system prompt until <CR> is entered. This is so that the P9030 may display the pass fail information on the Zif LED'S.

3/. DEV, LOC, HLP and DMP are not available out of remote mode, there functions are given below:

DEV Device select, Lists all possible device types and then prompts for a number indicating which device is to be operated on.

LOC Returns the P9030 to local command level.

HLP Displays the Command Menu.

DMP Allows the display of hex data from the ram. It prompts for a start and end address, then displays the data.

4/. When data entry is requested typing "Q"(uit) will cause the P9030 to return to command level.

5/. During data entry the P9000 will resond to both <BS> and <RUBOUT> in addition only the last two/ four digits will actually be entered when carriage return is hit. Any digits not entered will be taken as leading zeroes.

6/. The cursor functions for memory etc. are implemented by the upper case keys 'N' (Next) and 'P' (Previous).

Appendix A - Serial Data Transfers

Word Format The P9000 word format is:

START BITS	:	1
STOP BITS	:	2 OUT 1 OR MORE IN
DATA BITS	:	7/8
PARITY	:	ON/OFF/ODD/EVEN

Data sent LSB first, Parity last

Handshaking: Hardware only

Data output from the P9000 will occur when Pins 5 & 6 of the "D" connector (CTS & DSR) are at a voltage of > +3v.
Data output will be halted when the level on Pins 5 or 6 is < -3v.
Between +3v and -3v, the handshaking status is indeterminate.

To control the flow of data into the P9000, it outputs a +12v level on Pins 20 & 4 (DTR & RTS) when it is ready to receive.
When the P9000 is not ready to receive, the level on Pins 20 & 4 is dropped to -12v.
At the end of a serial transfer, the P9000 leaves the handshake lines at the +12v (ready) state until the next transfer is initiated. This avoids the problems of "hang up" in the sending machine.

Parity Even or odd

Parity is used to adjust the sum of ones in the data word plus the parity bit to be an even or odd number as selected. it may also be disabled.

Timeout

If the handshake or data lines are set up in such a way that the P9000 cannot receive/transmit for a long period of time, the P9000 will indicate a "timeout error" and return control to the keyboard. This will not happen in remote mode, where the timeout facility is disabled.

RS232C Signal specification

Input levels: +/- 3v to +/- 18v

Output levels: +/- 12v

Current limits: 15mA Sink or Source @ 12v

RS232C Connector pinout

Pin	Name	Direction	Description
1	Protective Ground		
2	TXD	OUT	Output data from P9000
3	RXD	IN	Input data from P9000
4	RTS	OUT	Paired with DTR
5	CTS	IN	Handshaking Input (controls data output)
6	DSR	IN	Paired with CTS
7	Signal ground		
20	DTR	OUT	Handshaking output (controls data input)

Serial Error Messages

"Serial Error" - Indicates that the checksum received is different to the calculated checksum. (Either a transmission error or an incorrect data format)

"Parity Error - The received parity failed to agree with the selected parity

"Illegal Address" - The data currently being received will not be within the current device. This data will not be loaded, but the P9C00 will continue to scan the line for valid data.

Use of the Buffer RAM

The software of the P9000 has been designed to make it possible to programme any size device from the serial port, even with the minimum RAM of 8k x 8. At the same time, the P9000 will use as much of the RAM as possible in order to free the sending machine.

To explain this more fully, the types of serial format must be split into two groups, those which have addresses included in them, and those that don't. The formats which have addresses are:

MOS TECHNOLOGY
SIGNETICS ABSOLUTE
TEKTRONIX HEXADECIMAL
MOTOROLA EXCRCISER
INTEL HEXADECIMAL

With these formats, the loading process takes place in one of two ways: If the PROM is larger than the RAM then the data is loaded and programmed, one record at a time. If the PROM is smaller than, or the same size as the RAM, then the entire data stream is loaded before programming takes place.

The other formats may be regarded as being continuous - that is, there are no implied address records.
With these formats data is loaded into the RAM until the RAM is full. The data is then programmed before the P9000 loads more data into the RAM. This process is repeated until all data is sent and programmed.

Appendix B - Serial Data Transfer Formats

Intel Hex Data Format

General

The Intel Hex format is a widely used format for the transfer of binary data. It transmits the data as short data records in ASCII code each, record having a checksum in order to ensure integrity of the data.

There are several record types within the definition of Intel Hex, but the P9000 only uses three of them. These are: type 00 -data record, type 01 the end of file record and type 02 the extended address record. If the P9000 receives any other records it just discards them.

Intel Data Record Format (type 00)

Byte 1 — Colon (:) delimiter

2 - 3 — Number of binary bytes of data in this record. The maximum is 32 binary bytes (64 ASCII bytes).

4 - 5 — Most significant byte of the start address of the data.

6 - 7 — Least significant byte of the start address of the data.

8 - 9 — ASCII zeroes. The "record type" for a data record.

10 - — Data bytes. Each binary byte is sent as two ASCII characters each one representing one nibble of the Hex representation of the byte.

Last two bytes — Checksum of all bytes in the record, excluding the delimiter, carriage return and line feed.The checksum is the negative of the modulo 256 binary sum of all of the bytes in the record.

CR,LF — Carriage return, line feed.

Intel Extended address record (Type 02)

Byte 1	Colon (:) delimiter
2 - 3	"02" The record length
4 - 5	ASCII zeroes
6 - 7	Record type "02"
8 - 9	USBA Upper segment base address (The top 16 bits of a 24 bit address) It is used in Intel's 16 bit data records. If no 02 records are sent the USBA is set to zero. If a USBA is specified then the bottom 12 bits are added to the offset address of the data records.
10 - 11	Checksum of all bytes in the record, excluding the delimiter, carriage return and line feed. The checksum is the negative of the modulo 256 binary sum of all the bytes in the record.
CR,LF	Carriage return,line feed.

Intel End of File Record (Type 01)

Byte 1	Colon (:) delimiter.
2 - 3	ASCII zeroes.
4 - 5	Most significant byte of transfer address (not used by P9000 ; set to zeroes).
6 - 7	Least significant byte of transfer address (not used by P9000 ; set to zeroes).
8 - 9	Record type 01. Indicates end of record
10 - 11	Checksum.
CRLF	Carriage return, line feed.

Note: all ASCII code is sent as seven bits

An Example of Intel Hex.

Given the data stream 23 45 AF B1 DO 77 to be sent as an Intel Hex Record to start at address 0000. The Record would be:
:060000002345AFB1D077EB<CR><LF>

Which may be broken down as:

Delimiter	:
Number of Bytes in the Record	06
Start Address High	00
Start Address Low	00
Record Type	00
Data	23
	45
	AF
	B1
	DO
	77
Checksum	EB
CR,LF	OD
	OA

Where the Checksum is calculated as follows:

```
CS= 06+00+00+00+23+45+AF+B1+D0+77+ =315
Modulo 256                         =15
Negative                           =EB
```

N.B.: The above checksum calculation was performed in Hexadecimal.

Upper Segment Base Addresses (USBA)

The Intel Hex records which may be received by the P9000 may be either the standard 8 bit format (record types 0 & 1) or the extended 16 bit format (additional record type 2).

The USBA is a 16 bit number which is used to set the current segment base. (This terminology is derived from the Intel 8086). In effect this means that the 16 bit number is shifted right four times and added to the 16 bit address of the type 0 data records. This results in a 24 bit address. The P9000 only actually uses the 16 least significant bits.

```
E.g.:
USBA =                         1263H
ADDRESS IN DATA RECORD=        3334H
ACTUAL ADDRESS OF DATA=       12340H
                             + 3334H
                             -------
                              15674H

IN THE P9000 THIS WOULD BE      5674H
```

Motorola Exorciser or "S" Format

General

The Motorola "S" format provides for the transmission of data in printable ASCII format. The data is divided into records. The P9000 recognises and uses two types of record, these are: "S1" the data record and "S9" the end of file record.

Exorciser Data Record Format (type S1)

Byte 1 — "S" character delimiter

2 — ASCII 1. The record type for data.

3 - 4 — Byte count. The number of binary data bytes in the record plus three (1 for checksum and 2 for address).

5 - 6 — Most significant byte of the start address of the data record.

7 - 8 — Least significant byte of the start address of the data record.

9 - — Data bytes. Each byte is sent as two ASCII characters, each representing one nibble of the Hex representation of the byte.

Last two bytes — Checksum of all bytes in the record excluding the delimiter and record type. The checksum is the 2's complement (NOT) of the modulo 256 binary sum of the bytes in the record.

CR,LF — Carriage return and line feed are output from the P9000, but are not checked when input.

Exorciser Data Record Format (type S2)

Byte 1 — "S" character delimiter

2 — ASCII 2. The record type for data.

3 - 4 — Most significant byte of start address of the data record

5 - 6 — Next most significant byte of start address of the data record

7 - 8 — Least significant byte of start address of the data record

9 - — Data bytes. Each byte is set as two ASCII characters, each representing one nibble of the hex representation of the byte.

Last two bytes

Checksum of all bytes in the record excluding the delimiter and record type. The checksum is the 2's complement (NOT) of the modulo 256 binary sum of the bytes in the record.

CR,LF Carriage return and line feed are output from the XP640, but are not checked when input.

Exorciser End of File Record

Byte 1 "S" delimiter

2 ASCII 9 Indicates end of file record

3 - 4 Byte count = 03 in end of file record

5 - 6 Most significant byte of start address (not used in the P9000 ; set to zero)

7 - 8 Least significant byte of start address (not used by the P9000 ; set to zero).

9 - 10 Checksum

CR,LF Carriage return and line feed are output from the P9000, but are not checked when input.

An Example of Motorola Format.
A Motorola record consisting of the data 67 A0 4A 2B to start at 213F would be:

S107213F67A04A2B1C<CR><LF>

Which consists of:

Delimiter	S
Record Type	1
Byte Count (Data + 3)	07
Start Address High	21
Start Address Low	3F
Data	67
	A0
	4A
	2B
Checksum	1C
CR	0D
LF	0A

Where the Checksum is calculated as follows:
CS = 07+21+3F+67+A0+4A+2B = 1E3
Modulo 256 E3
1's Complement 1C

N.B.: The above calculations were performed in Hexadecimal

GP Binary Format

General

This is a simple format devised by GP specifically for users writing there own formats. It is designed to be as simple as reasonably possible to write drivers/ receivers for. All data is sent in 8 bit binary, LSB first.

Format of GP binary

The data is preceded by a 4 byte block consisting of a block-length and a checksum:

Byte 1	Least significant byte of the block length.
2	Most significant byte of block length.
3	Least significant byte of the checksum.
4	Most significant byte of the checksum.
5 -	Data bytes.

The block length is the number of bytes in the data record.

The checksum is the modulo 65536 binary sum of the data being transferred.

An Example of GP Binary .

A GP Binary record to send the following data 23 67 8F 2A would be:

Low Block Length	04
High Block Length	00
Low part of Checksum	43
High part of Checksum	01
Data	23
	67
	8F
	2A

Where the Checksum was calculated as follows:
CS= 23+67+8F+2A =143

N.B.: The above calculation was performed in Hexadecimal

Format of Serial List

This format is an output only format designed primarily to drive a serial printer. Data is output as ASCII characters in rows of 16 characters, each row being preceeded by the address of the first character in the row. Each row is terminated by carriage return and line feed. The data is sent in blocks of 256 bytes. After every third block a form feed is sent to prevent data being printed on the perforations of the paper.

Example of serial list output

```
0000 E4 AA CD 00 99 C9 E5 F5 E1 F1 4F 7D ED CF 21 01
0010 21 FF FF 0A E4 C4 01 C9 22 FD 22 E4 14 C3 FF FF
```

The Tektronix Hexadecimal format (TEKHEX)

This format provides for the transfer of data blocked into records of printable ASCII characters. There are 2 types of records used and recognised by the P9000. These are the data record and the end of file record.

Tekhex Data Record.

Byte 1	"/" character; delimiter
2 - 3	Most significant byte of the start address of the data record.
4 - 5	Least significant byte of the start address of the data record.
6 - 7	Byte count. The number of binary data bytes in the record .
8 - 9	First checksum, sum of all bytes, modulo 256 of the six hex digits of the load address and byte count.
10 -	Data bytes. Each byte is sent as two ASCII characters, each representing one nibble of the Hex representation of the byte.
Last two bytes	Checksum of all of the data bytes in the record, calculated as the modulo 256 sum of all the nibbles making up the data bytes.
CR,LF	Carriage return and line feed are output from the P9000, but are not checked when input.

Tekhex End of File Record

Byte 1	"/" delimiter
2 - 3	Most significant byte of start address (not used in the P9000 ; set to zero)
4 - 5	Least significant byte of start address (not used by the P9000 ; set to zero).
6 - 7	Byte count = 00 in end of file record
9 - 10	Checksum of all bytes in the record excluding the delimiter and record type. The checksum is the modulo 256 binary sum of the NIBBLES making up the bytes in the record.
CR,LF	Carriage return and line feed are output from the P9000, but are not checked when input.

An example of TEKHEX data format

To send the data 23,00,A8,A9,17,04 the data format would look like:

/000006062300A8A9170436<CR><LF>

Which consists of:

Delimiter	/
Start Address	0000
Byte Count	06
Checksum of Address field	06
Data	23,00,A8,A9,17,04
Checksum	36

Where the checksums were calculate as:

Checksum of Address = 0+0+0+0+6= 6
Checksum of data = 2+3+0+0+A+9+1+7+0+4=36H

MOS Technology data format

In this format the data is divided into records and sent as printable ASCII characters. There are two types of record used and recognised by the P9000. These are the data record and the end of file record.

MOS Data Record

Byte 1 — ";" character; delimiter

2 - 3 — Byte count. The number of binary data bytes in the record .

4 - 5 — Most significant byte of the start address of the data record.

6 - 7 — Least significant byte of the start address of the data record.

8 - — Data bytes. Each byte is sent as two ASCII characters, each representing one nibble of the Hex representation of the byte.

Last four bytes — Checksum ,sum of all data bytes in the record. The checksum is the modulo 65536 binary sum of all the bytes in the record including the block length and address, but excluding the delimiter and the checksum itself. It is transmitted high byte then low byte.

CR,LF — Carriage return and line feed are output from the P9000, but are not checked when input.

MOS End of File Record

Byte 1	";" delimiter
2 - 3	Byte count = 00 in end of file record
4 - 5	Most significant byte of sum of total bytes sent in all records
6 - 7	Least significant byte of sum of total bytes sent in all records
8 - 9	Most significant byte of checksum
10 - 11	Least significant byte of the checksum of all bytes in the record excluding the delimiter and record type. The checksum is the modulo 65536 binary sum of the bytes in the record.
CR,LF	Carriage return and line feed are output from the P9000, but are not checked when input.

Example of MOS TECHNOLOGY data records.

To send the data record 86 AF E5 64 98 99 99 00 the MOS record would be:

;08000086AFE564989999000448<CR><LF>

Which consists of:

Delimiter	;
Byte Count	08
Start Address	0000
Data	86AFE564989999900
Checksum	0448

Where the checksum is calculate as:

Checksum = 86+AF+E5+64+98+99+99+00=0448

Signetics Absolute Data Transmission Format

In this format data is divided into records of printable ASCII characters. The P9000 uses and recognises two types of data record. The data record and the end of file record.

Signetics Absolute Data Record

Byte 1	":" character; delimiter
2 - 3	Most significant byte of the start address of the data record.
4 - 5	Least significant byte of the start address of the data record.
6 - 7	Byte count. The number of binary data bytes in the record .
8 - 9	Checksum of all the bytes in the address and data fields, calculated by EXORing each byte with the previous byte, then rotating the resultant byte left one bit.
10 -	Data bytes. Each byte is sent as two ASCII characters, each representing one nibble of the Hex representation of the byte.
Last two bytes Checksum	sum of all data bytes in the record the checksum is calculated in the same way as the first checksum.
CR,LF	Carriage return and line feed are output from the P9000, but are not checked when input.

Signetics Absolute End of File Record

Byte 1	":" delimiter
2 - 3	Most significant byte of start address (not used in the P9000 ; set to zero)
4 - 5	Least significant byte of start address (not used by the P9000 ; set to zero).
6 - 7	Byte count = 00 in end of file record
8 - 9	Checksum of all the bytes in the address and data fields, calculated by EXORing each byte with the previous byte, then rotating the resultant byte left one bit.
CR,LF	Carriage return and line feed are output from the P9000, but are not checked when input.

Example of SIGNETICS ABSOLUTE data format

To send the data 23 EE F1 2A D4 55 99 the data record would be as follows:

:0000070E23EEF12AD4559946<CR><LF>

Which consists of:

Delimiter	:
Start Address	0000
Byte Count	07
First Checksum	0E
Data	23EEF12AD45599
Second Checksum	46

Where the checksums are calculated as follows:

First Checksum (((00 EXOR 00)*2 EXOR 00)*2 EXOR 07)*2=0E

Second Checksum ((((((23 EXOR EE)*2 EXOR F1)*2 EXOR 2A)*2 EXOR D4)*2 EXOR 55)*2 EXOR 99)*2=46

The ASCII Space, Comma, Apostrophe and Percent

Data in these formats is transmitted in sequential, two character groups representing hex bytes followed by the execute code space, percent, apostrophe or comma. Data may be transmitted as either 4 or 8 bits. The P9000 assumes that the two characters prior to the execute code were a valid character. If only one character was received prior to the execute code then a leading zero is assumed.

When the P9000 is receiving in these formats it recognises 3 types of information; these are Address information, Data and Checksum.

General
The data transmission must be preceeded with an <STX> character (02H) which may then be followed immediately with data or by an address field. The transmission must be terminated with an <ETX> (03H) followed by either a checksum field or at least 16 nulls.

Data field

Each time an execute code is received the two previous bytes are assumed to be valid data. If there have not been two valid ASCII Hex bytes prior to the execute code then the programmer assumes leading zeroes.

Address field

When the P9000 receives a "$" followed by an "A" it then expects 4 ASCII Hex digits giving the address of the first data field. This address must be terminated by a comma (except in the "Comma" format where it is terminated by a full stop).The input data will then be loaded, starting at this address.

Checksum field

The data field must be terminated with an <ETX> this may optionally be followed with a checksum. The checksum is expected as "$" followed by "S" followed by the four bytes of the checksum. The checksum must be terminated with a comma (or for the comma format a full stop). The checksum is calculated as the modulo 65536 sum of all of the data sent since the previous <STX>. If the checksum is not sent then at least 16 characters must follow the <STX> to prevent a time-out error.

An example of an ASCII SPACE data transmission

<STX>$A0000,<CR><LF>
31 FF 77 C3 FF FE 76..........<ETX><CR><LF>
$S1234,<CR><LF>

An example of an ASCII COMMA data transmission

<STX>$A0000.<CR><LF>
31,FF,77,C3,FF,FE,76..........<ETX><CR><LF>
$S1234.<CR><LF>

An example of an ASCII PERCENT data transmission

<STX>$A0000,<CR><LF>
31%FF%77%C3%FF%FE%76..........<ETX><CR><LF>
$S1234,<CR><LF>

An example of an ASCII APOSTOPHE data transmission

<STX>$A0000,<CR><LF>
31'FF'77'C3'FF'FE'76..........<ETX><CR><LF>
$S1234,<CR><LF>

ASCII BPNF,BHLF,B1OF Formats

In these formats each byte of data is transmitted as an ASCII "B" followed by eight ASCII bytes representing the bits of the data byte. Zeroes and ones are represented respectively in the two formats by: "N", "P" or "L", "H", or "0", "1". Each byte is terminated with the ASCII character "F". The data is transmitted least significant bit first. The entire data stream must be started with a non-printable <STX> and ended with a non-printable <ETX>. The data output from the P9000 is formatted to suit a list device by outputting a space between each byte, and a <CR><LF> at the end of each line of six bytes.

An example of BPNF format.

The data stream OF,84,73,21 would be sent as:

<STX>BPPPPNNNNF BNNPNNNNPF BPPNNPPPNF BPNNNNPNNF<ETX>

An example of BHLF format.

The data stream OF,84,73,21 would be sent as:

<STX>BHHHHLLLLF BLLHLLLLHF BHHLLHHHLF BHLLLLHLLF<ETX>

An example of B1OF format.

The data stream OF,84,73,21 would be sent as:

<STX>B11110000F B00100001F B11001110F B10000100F<ETX>

DEC Binary and Binary formats

In both of these formats data is transmitted as a string of binary information. The only difference in the two formats is the start of record. For Binary the record starts with any number of nulls followed by a rubout (FFH). In DEC binary the format starts with any number of rubouts followed by a null. The data after the record start is a string of binary data with no checksums, no byte counts and no print formatting. As there is no end of file delimiter the receiving machine must have been told how many bytes to expect. In the P9030 this is entered from the keyboard. In the P9020 this is assumed to be enough data to fill the currently selected device.

Appendix C

The Printer Interface

General

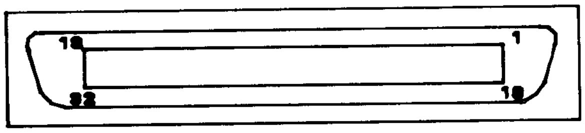
The P9000 printer interface is a parallel interface.It is compatible with the Centronics type port which the majority of printers are equipped with. The data is transmitted in standard ASCII code with the 8th bit set to a zero. Carriage Returns and Line Feeds are sent at the end of each line.

Connection

The printer port is the **AMP CHAMP 36 BAIL LOCK TYPE** connector on the rear of the P9000. It may be connected to any **CENTRONICS** type printer via a **CENTRONICS/CENTRONICS** cable.The pinout of the connector is shown in the table below:

PIN	SIGNAL	PIN	SIGNAL
1	STROBE	19	TWISTED PAIR GROUND (PIN 1)
2	DATA 1	20	TWISTED PAIR GROUND (PIN 2)
3	DATA 2	21	TWISTED PAIR GROUND (PIN 3)
4	DATA 3	22	TWISTED PAIR GROUND (PIN 4)
5	DATA 4	23	TWISTED PAIR GROUND (PIN 5)
6	DATA 5	24	TWISTED PAIR GROUND (PIN 6)
7	DATA 6	25	TWISTED PAIR GROUND (PIN 7)
8	DATA 7	26	TWISTED PAIR GROUND (PIN 8)
9	DATA 8	27	TWISTED PAIR GROUND (PIN 9)
10	NC	28	TWISTED PAIR GROUND (PIN 10)
11	BUSY	29	TWISTED PAIR GROUND (PIN 11)
12	NC	30	GND
13	NC	31	NC
14	GND	32	NC
15	GND	33	GND
16	GND	34	NC
17	GND	35	NC
18	NC	36	NC

The Centronics Type Printer Port

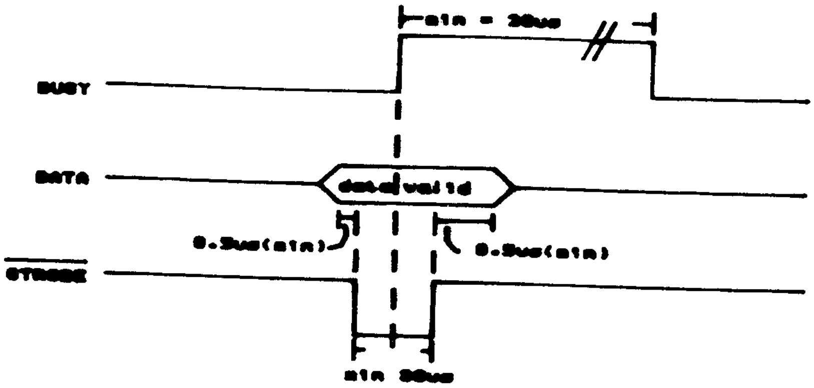


Pin Out of The Centronics Connector

Description of Signals on the Centronics Interface

STROBE an active low output signal which is output to indicate that there is valid data on the port.
BUSY when this input is high the P9000 will not output data. It is used to indicate that the printer is not ready to receive data.
DATA 1-8 these lines carry the output data.

GND all of the ground lines are linked to the P9000 system ground. Problems with parallel interfaces often stem from bad grounds, so ensure that all grounds are connected.



Timing Diagram for the Centronics Port

GP Industrial Electronics Ltd.,

P9000 Production EPROM/EEPROM Programmer
Calibration Procedure

The P9000 is a precision made machine.

All timing for program pulses, set up times etc. are software controlled by a Z80 Microprocessor and are therefore crystal controlled and fixed.

The power supply voltages are preset and computer tested before they leave the factory.

These voltages may need adjustment from time to time.
Before attempting to calibrate the P9000, first check that it is required:-
Select CALIBRATE from the mode menu, and press 'START'.

1/ Move to step number 7 initially by pressing the UP ARROW key.
Measure the voltage as specified under step 7, and if out of range adjust using preset No: 7.

2/ Exit from calibrate mode by pressing 'STOP'.
The re-enter the calibrate sequence by pressing 'START'.

Follow the sequence of steps listed below and measure the voltage as specified on all steps apart from step 7 which you should pass over. - **This has already been set on your first pass.**

If one or more of the measured voltages are outside those specified in the table then adjust the voltage using the preset potentiometers numbered below.

To gain access to the potentiometers, remove the P9000 top cover - Please follow the instructions on its removal as given in the P9000 Users Manual.

NOTE

1/ There are dangerous voltages inside the P9000 and calibration should only be carried out by a competent electronics engineer or technician.

2/ When reassembling the P9000 please follow the procedure given in the users guide.

3/ Damage caused by incorrect calibration or inexpert dismantling of the P9000 will void the warranty.

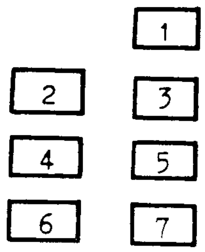
4/ It is essential to check step 7 first before checking any of the other settings.

Calibration table

Step Number	Pin Number	Socket	Lo Volt Limit	Hi Volt Limit	Adjust
One	28	Master	4.80	5.20	3
Two	28	Copy	5.80	6.20	2
Three	28	Copy	4.80	5.20	4
Four	1	Copy	20.70	21.50	6
Five	1	Copy	11.70	12.40	5
Six	1	Copy	4.80	5.20	7
Seven	1	Copy	24.70	25.50	1
Eight	22	Copy	10.70	12.00	No adjust
Nine	Measure the pulses on pin 27 of a copy socket to be mark space TTL pulses of 570us (approx). This checks that the system clock (crystal controlled) is OK to quarantee software timing. No adjustment is possible or should ever be necessary.				

To exit from CALIBRATE mode, press 'STOP'.

Potentiometer Identification



P9030 INDEX

P9030 Index