1. Introduction

The "P112" board implements a stand-alone Z180-based computer system. Although primarily designed as a CP/M platform, it can support other software.

The board is designed to fit on a 3.5" diskette drive (some drives may require spacers). It requires only +5V power, so that, with a 5V-only diskette drive, no other power supplies are required. Power consumption (board only) is around 150mA. The board supports two RS-232 serial and one parallel IO port, besides a capability for bus expansion. An expansion socket permits 3 further serial ports to be used, by providing off-board level converters. All ports are fully interrupt-capable.

The following goals were set, in selecting components:

- Parts should be current production, recommended for new designs
- No specially programmed logic devices should be necessary

2. Logic Description

The following descriptions should be read in conjunction with the accompanying schematics. Data sheets on the various chips are obtainable from the manufacturers’Web sites.

2.1. CPU

Zilog recommend two parts for new designs: the Z84C15 and Z80182. For the present project, the Z80182 is appropriate, providing a Z180 core, with enhanced memory mapping features. This part is available in a range of clock speeds: the initial boards are clocked at 16MHz. This speed, while the highest the CPU chip can support, does limit the choice of serial-port speeds, as it does not factor well. This has minimal impact on the standard serial port, which uses the in-built baud-rate generator. This generator is very flexible in its range of possible divisors. If however, the expansion ports (see P16 below) are used, they have a much reduced set of divisors available, and will normally require a carefully-chosen clock frequency. The boot software can recognise and adjust to the following clocks:

12.288MHz
16.0MHz
18.432MHz
24.576MHz

The last two will require a faster CPU chip to be fitted.

The Z80182 includes several in-biult peripheral functions. All these are fully supported by the Z80 vectored interrupt system. The following internal peripherals are used:

2.1.1. Serial Port

The Z80182 has a total of 4 serial ports. Of these, one is brought out as the default terminal port, at normal RS232 levels. The remaining 3 ports are available as un-buffered TTL signals, for use with off-board level converters. One of these additional ports features full SDLC functionality, and can be configured for DMA control.

The Z80182 serial ports do not support the DSR and RTS modem-control signals, so these are provided (in the main-terminal port) by two lines from the on-chip parallel port (see below). These lines are not used by the standard software.

2.1.2. Parallel Port

The Z80182 has one 8-bit parallel port available (the others have been overridden for other functions). This is used to support on-board facilities. The pin assignments are:

A0	RTC data I/O line (bidirectional)
A1	RTC Clock line
A2	RTC Reset line
A3..A4	Not used
A5	Set low to enable the 12V Vpp generator for flash ROMs
A6	DSR input from Serial Port 1
A7	RTS output to Serial Port 1

2.1.3. DMA

By default, DMA Channel 0 is used for the diskette controller, and Channel 1 is available to an expansion card (if fitted). Jumper P2 may be altered if required, to provide DMA support for Serial Port 1.

The Z180 DMA only provides the TENDx (end-of-block) signal during its *write* cycle. This makes it unusable with the floppy disk controller, when reading from the disk (the FDC requires TEND asserted during the DMA cycle addressed to it, not to the memory). Consequently, the FDC is programmed not to use TENDx, which implies that all transfers will post a "end of cylinder" error. This is allowed for by the software.

2.1.4. Memory Mapping

The Z80182 includes two levels of memory mapping logic. The first maps the 64kB logical address space into a maximum of 3 "zones" in the 1MB physical address space. The second decodes the RAMCS\ and ROMCS\ outputs from the translated physical address.

In normal use, the first map may be changed frequently, as the operating system switches tasks. The second will normally be initialised at reset, in terms of the amount of memory actually fitted, and not changed thereafter. The boot-code includes a "smart" memory initialisation routine, which examines the chips actually fitted, and locates them in the physical space in an optimal manner. The sign-on message includes a report of the amount of RAM available, and its location in physical address space. If sufficient RAM is present, the ROM will be copied into low RAM, and afterwards disabled. This enables the CPU to be run faster.

The memory setup routine makes two requirements:
1. If two RAM chips are fitted, they must be of the same size.
2. The address-decode jumper P1 must be correctly set.

If these requirements are not met, the startup code will malfunction.

2.2. Memory

The board is designed to accept a wide variety of memory parts, in both 28 and 32-pin packages. The 0.6" DIP format was chosen as being compatible with the widest range of parts.

One ROM socket and two RAM sockets are fitted. A pre-programmed ROM can be fitted in a RAM socket if desired. On-board RAM capacities from 32kB to 1MB are available.

The jumper P3 can interchange the select signals for the ROM and RAM-1 sites. This permits a pre-programmed ROM to be fitted in RAM-1 (U3), to program a blank flash device in U4. (See below).

2.2.1. ROM

The ROM socket (U4) can accept a 32kB ROM part. A pre-programmed EPROM may be used, or a "flash" ROM. The board includes facilities for in-system programming of 5V and 12V flash ROMs. For normal use, the header P12 should be jumpered across Pins 2-3 (WE\). For 12V parts (Intel or AMD 28F256), the 12V converter U12 is required. At power-up, this converter will be disabled, causing a Vpp of 5V to be supplied via D2. This makes the circuit safe with 5V ROMs also. To program a ROM, set the CPU parallel port pin A5 low: this enables the 12V supply.

For 5V ROMs (eg Atmel AT29C256), U12, Q1, R3 and R4 may be omitted. D2 may be replaced by a shorting link, to deliver standard 5V power.

2.2.2. On-Board RAM

Each RAM socket may accept a 32kB, 128kB or 512kB SRAM part. The address decoding must be set for the parts in use, as follows:

RAM size	RAM part (typical)	P1 setting	P13 setting
32kB / 64kB	HM62256	1-4 (A15)	1-2 (Vcc)
128kB / 256kB	HM628128	1-2 (A17)	1-2 (Vcc)
512kB / 1MB	HM628512	1-3 (A19)	2-3 (A17)

It is not possible to mix RAM chips of different sizes.

For zero wait-state operation, 70nS parts are required.

2.2.3. Expansion RAM

Provision is made for an expansion board to carry up to 32kB of RAM, which may be mapped into the main memory space. By convention, such memory will be at the top of the 1MB physical space. The purpose of this is to provide for dual-port memory for video drivers and similar devices. Of course, the 32kB may be extended by a bank-switching arrangement on the expansion board.

Expansion board memory should be selected when the signals ROMCS\ and RAMCS\ are both high (ie no on-board memory is selected). The read and write enables are MRD\ and MWR\, which enable bus transactions.

If necessary, the WAIT\ line may be driven low to delay the CPU.

2.3. IO Cycle Control

The multi-function IO chip requires two additional wait-states in every IO cycle, to satisfy its timing. This is provided by setting the IWI bits in the DCNTL register of the Z801821 CPU (IO address 32H).

2.4. Real-Time Clock

The Dallas DS1202 has a simple bit-serial interface. This is supported by 3 bits from the CPU's internal PIO device (see above). The DS1202 also provides a block of battery-backed RAM, which may be useful for storing BIOS setup parameters (memory size, serial communications setups, etc.).

2.5. Expansion Socket

The CPU bus is brought out to the expansion socket J1. Access to expansion memory has been described above. For external IO devices, IORQ\ low and M1\ high should be decoded to validate an IO access. Valid addresses for expansion boards are in the ranges 40..7F and C0..D7. The IO data strobes are RD\ and WR\.

Support is provided for expansion boards using the Z80182 internal DMA. The EXTRQ\ request line may be jumpered (at P2) to one of the internal DMA channels.

2.6. Serial Port 1

This is implemented using one of the Z80181's internal SCC channels. These have enhanced capabilities over standard PC channels, including SDLC operation, and fully vectored interrupt support. See the Z80182 documentation for details.

2.7. Multi-IO Chip

The non-CPU IO functions are implemented in a SMC multi-function IO chip. These parts are available in several variants, and the board is designed to accept 4 different types: FD37C651, '652, '665 and '666. Further details on configuration options are given below.

2.7.1. Address Decoding

The multi-IO chip occupies the IO address space 80..BF. These addresses are mapped to appear to the IO chip as standard PC addresses. Programming of the multi-IO features follows standard PC practice. The address mapping is described below.

2.7.2. Cycle Timing

The IO chip latches address data at the beginning of IOR\ or IOW\. This can cause problems with the Zilog timing, which regards the RD\ and WR\ signals as basically clock-enables. This is overcome by use of the CPU's "E" clock output, which goes high during the valid part of a bus cycle. This is used as a local enable to IOR\ and IOW\, via the decoder U11A. This decoder also provides the interlock from the safety-latch described below.

2.7.3. Safety Latch

As pointed out by Claude Palm (The Computer Journal, No. 76) the Z180-series devices can generate spurious IO selections during interrupt-acknowledge cycles. These are normally suppressed, since neither of the RD\ or WR\ data dtrobes appear. However systems which decode the DACK\ (DMA acknowledge) from the address may malfunction, due to these spurious IORQ\ signals. In the present design, this is prevented by the latch U8.

This is set at the beginning of a interrupt acknowledge cycle (M1\ and IORQ\ both low), and blocks any address decodes. The latch is only cleared when IORQ\ again goes high, at the end of the acknowledge cycle.

2.8. Serial Port 2

This is the "primary" serial port on the multi-IO device (the secondary port is not used). It is strictly PC compatible.

2.9. Diskette Port

The diskette interface emulates a standard PC-type diskette controller. The IO chip is configured to logically "swap" Drives 0 and 1. This assumes the drives have the DS1 jumper set, as is normal for PC-AT usage. Given this, Drive-0 connects directly to the PCB, with no ribbon-cable cores swapped. This is to enable the PCB to be mounted directly on the disk drive, with the PCB and drive connectors adjacent. Drive-1 will be the further away, and will have cores 10-16 swapped in the cable.

It should be noted that there is no universal standard regarding the placement of the data connector on diskette drives: it is not possible to have this card fit directly to **every** drive available. The direct fitting has been tested with Teac FD235 drives.

2.10. Printer Port

The printer port provides PC-like functions, in basic bidirectional mode. Enhanced ECP and EPP modes are not supported.

3. Major Components

The principal components are briefly introduced below, with particular reference to their use in the present application.

3.1. CPU Chip

The Z80182 provides a Z180 CPU core, multi-function serial IO port, and two DMA channels. Extensive use is made of the Z180's memory mapping ability, to fill in "holes" in the physical address space, and to switch the ROM in and out of circuit.

This part is available in several speed grades: the initial build will be fitted for 16MHz clocking.

3.2. I/O Combination

Most of the IO functions are implemented in a SMC multi-IO part, designed for use in PC's. The board is multi-capable, and can accept any of the following parts:

- FDC37C651
- FDC37C652
- FDC37C665
- FDC37C666

The '651 and '665 are fully software configurable. However the '652 and '666 have some functions configured by external resistors. These are not otherwise fitted, and are the only surface-mount resistors on the board. Space limitations preclude putting reference designators on the board, however they are all 1206 size, 27kΩ parts. Their reference designators are R101..R111. No harm will occur if they are fitted with a software-configurable IO chip.

These parts can be programmed to generate active-high or active-low interrupt signals. In this application, active-low is required, and is selected by the start-up code.

3.2.1. Port Addressing

The IO chip may be configured for several different internal address decoding schemes, correponding to different IO assignments in a PC environment. In the present application, the following addresses are used:

Function	Address (CPU)	Address (IO chip)
Parallel Port	8C..8F	3BC..3BF

Diskette (program access)	90..97	3F0..3F7
Serial Port	98..9F	3F8..3FF
Diskette (DMA access)	A0..BF	N/A

3.3. Flash ROM

The board is designed to accept a variety of flash ROM parts. The inital build is fitted with Intel parts, which need a 12V programming voltage. If 5V-only parts (eg Atmel) are used, the voltage converter U12 and its associated components may be omitted.

3.4. Realtime Clock

A Dallas DS1202 realtime clock/RAM is used. This has a serial interface, which is implemented using 3 spare parallel-port pins from the CPU chip.

Be aware that this chip, when first powered-up, defaults to write-protected and oscillator disabled. Before it will run normally, you must first turn off write-protect, then activate the oscillator. Each is a single-byte write operation.

4. Connector Pins

The pin assignments of the various connectors are tabulated below. All are 0.1" pitch headers.

4.1. P1 RAM Size Selector

Pin Assignment			
1	Address decoder input	2	A17 (128kB chips)
3	A19 (512kB chips)	4	A15 (32kB chips)

This jumper selects the address at which the address logic switches between the RAM chips. In effect, it defines the capacity of each RAM chip. The pin assignment is as follows:

	4	
3	1	2

4.2. P2 DMA Requests

Pin Assignment			
1	Diskette controller request	2	DMA request 0 (to CPU)
3	Expansion skt. request	4	DMA request 1 (to CPU)
5	SIO request		

This header selects the sources for DMA requests. The Z180 core includes 2 DMA channels, which may be connected among 3 possible sources: diskette, expansion socket, and serial IO channel. The standard boot code requires Pins 1 and 2 jumpered.

4.3. P3 Flash Bootload Selector

Pin Assignment			
1	RAM select from CPU	2	RAM socket 1 select
3	ROM socket select	4	ROM select from CPU

For normal use, jumper pins 1-2, 3-4. For ROM duplicating, jumper pins 1-3, 2-4.

This jumper allows the ROM (U4) and RAM-1 (U3) sockets to be logically interchanged. The effect is that the CPU will boot from RAM-1. This enables the board to serve as a flash-ROM duplicator, using the following procedure.

Set P3 in the "normal" position, and fit the standard boot ROM in U4, with a 32kB RAM in U2 and U3. Boot the system: it will report RAM available at 40000..4FFFF (physical locations). The logical space is mapped as 0000..7FFF to U4 (ROM), and 8000..FFFF to U3 (RAM). U2 is currently inaccessible (being overlaid by the ROM).

NB The following procedure will be superseded by future software.

Now copy the existing boot code into U3 as follows. Manually enter the following code into high memory:

```
FFF0   di                    F3
FFF1   ld      hl,0          21 00 00
FFF4   ld      de,8000       11 00 80
FFF7   ld      bc,FFF0       01 F0 FF       ;Copy up to this code
FFFA   ldir                  ED B0
FFFC   halt                  76
```

Set SP=FFFF, and Go FF00. The ROM code will be copied into U3 RAM (*overwriting the Debugger workspace*), and the CPU will then halt.

Without removing power, hold the Reset input pin low. Then interchange the J3 jumpers, and fit the new ROM in U4 (*do this carefully: you are "hot-plugging" the ROM*). Finally remove the Reset. The board will re-boot, and will report RAM at 40000..47FFF (physical). The code now sees logical addresses 0000..7FFF in U3, and 8000..FFFF in U2. The ROM is currently inaccessible.

The system is now live, with 64kB of logical RAM available. You can boot the operating system normally, and use the ROM-burning program (*not yet available*) to program the ROM. *Remember to use an OS version with a RAM-resident BIOS!*

To boot from your new ROM, don't forget to reset the P3 jumpers before resetting. If you do not, it will boot from the RAM image: this may be a useful technique for checking-out a new boot program before burning it in ROM.

The above procedure is not, of course, limited to building boot ROMs. In this mode, the board serves as a general-purpose flash-ROM programmer, for 32kB parts.

4.4. P4 Serial Port 1

Pin Assignment			
1	DCD (in)	2	DSR (spare in)
3	RXD (in)	4	RTS (spare out)
5	TXD (out)	6	CTS (in)
7	DTR (out)	8	SYNC/RI (in)
9	Ground	10	Not used

The pin assignments are such that a 10-way ribbon cable (with the 10th core removed) can connect a 10-way header to a DB-9 plug, and generate the standard PC/AT pin arrangement. This port is implemented as a Z180 port, which does not support the DSR and RTS lines. These are therefore implemented using parallel port pins. This port can support synchronous modes, in which the CPU's SYNC pin (normally used as the RI input) becomes an output. To prevent contention with the RS232 receiver, the jumper at P5 should be removed in synchronous working.

The Z180 uses the CTS and DCD pins as hardware enables. For the port to operate, these must either be strapped to the DTR output, or connected via (for example) a "null modem" cable to another PC-AT compatible port. *If left open, the port will be disabled.*

4.5. P5 SYNC Jumper

Pin Assignment			
1	From SYNC receiver	2	To SYNC pin on CPU

See P4 above. This link may be removed to prevent contention between the SYNC output of the CPU chip, and the RS232 receiver output. The link should be fitted in asynchronous working, and removed for synchronous.

4.6. P6 Power Supply/Reset

Pin Assignment			
1	Ground	2	+5V
3	Reset	4	+5V
5	Ground		

The reset input is intended for a switch to ground. It may be left open, the board will automatically reset when powered up.

4.7. P7 Parallel Printer

Pin Assignment			
1	Strobe\	2	AutoFeed\
3	D0	4	Error\
5	D1	6	Init\
7	D2	8	Select-In\
9	D3	10	Ground
11	D4	12	Ground
13	D5	14	Ground
15	D6	16	Ground
17	D7	18	Ground
19	Ack\	20	Ground
21	Busy	22	Ground
23	PaperEnd	24	Ground
25	Select	26	Ground

The pin assignment is such that a 26-way ribbon cable (with line 26 removed) may run direct to a DB-25 connector, presenting a standard IBM parallel printer interface.

The interface offers the basic bidirectional capability; expanded (ECP/EPP) features are not available.

4.8. P8 Serial Port 2

Pin Assignment			
1	DCD (in)	2	RXD (in)
3	TXD (out)	4	DTR (out)
5	Ground	6	DSR (in)
7	RTS (out)	8	CTS (in)
9	RI (in)	10	Not used

See P4 for details. This port is implemented on the combination IO chip, and is strictly PC compatible.

4.9. P9 Disk Drives 0,1

Pin Assignment			
2	DENSEL\	4	Not used
6	DRATE0	8	INDEX\
10	MTR0\	12	DR1\
14	DR0\	16	MTR1\
18	DIR\	20	STEP\
22	WDATA\	24	WGATE\
26	TK0\	28	WPROT\
30	RDATA\	32	HDSEL\
34	DSKCHG\		

All odd-numbered pins are Ground. Drive wiring is the usual PC/AT type, with lines 10..16 switched around between the drives. The Drive 0 & 1 signals are logically interchanged within the IO chip, so that the drive connected *without* a twist in the cable is Drive 0. This will typically be the drive on which the board is mounted.

4.10. P10 Disk Drives 2,3

Same as P9, but decodes Drives 2 & 3.

4.11. P11 Battery Isolator

Pin Assignment			
1	Ground	2	Battery -ve

This serves as a safety-isolator while working on the board, and to force a reset to the real-time clock chip.

4.12. P12 ROM Pin-3 Function

Pin Assignment			
1	+5V	2	U4 pin 3
3	Memory Write\		

This jumper connects Pin 3 of the ROM socket either to MWR\, or to Vdd. For normal use, connect to MWR\.

4.13. P13 RAM Pin-30 Function

Pin Assignment			
1	+5V	2	U2/3 pin 30
3	A17		

For 512kB RAM chips, this pin should connect to A17. For smaller parts, connect to Vdd.

4.14. P14 Extra Serial Ports

This connector carries un-buffered signals for the spare CPU serial ports. Uncommitted inputs have 47kΩ pull-up resistors. The signal names follow those used in the Z80182 data-book, which should be consulted for the capabilities of these ports. The pin-outs are:

Pin	Signal	Pin	Signal
1	TXA1	2	RXA0
3	RTS0	4	CTS0
5	RTSB = TEND1	6	DTRB
7	CTS1	8	TXA0
9	RXA1	10	DCD0
11	CTSB	12	DCDB
13	Vcc	14	TXDB
15	TRXCB	16	RXDB
17	RTXCB	18	SYNCB
19	GND	20	GND

The 3 ports are designated by the final characters in the names, thus:
0 & 1 Basic asynchronous ports
B Full-function a/sycnhronous port

Note that RTSB and TEND1 share a pin. Either role is possible, by suitably programming the CPU chip. The standard software does not use this pin at all.

4.15. J1 Bus Expansion

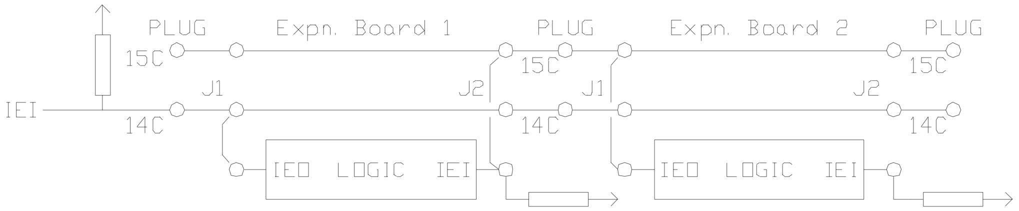
Pin Assignment			
1A	Ground	1B	Ground
1C	+5V	2A	D0
2B	A8	2C	D1
3A	D2	3B	A9
3C	D3	4A	D4
4B	A10	4C	D5
5A	D6	5B	A11
5C	D7	6A	A0
6B	A12	6C	A1
7A	A2	7B	A13
7C	A3	8A	A4
8B	A14	8C	A5
9A	A6	9B	RAMCS\
9C	A7	10A	Mem-Read\
10B	Mem-Write\	10C	Ground
11A	M1\	11B	WAIT\ (input to CPU)
11C	RD\	12A	IORQ\
12B	ROMCS\	12C	WR\
13A	RST\ (reset)	13B	E
13C	INT0\ (dedicated interrupt)	14A	TEND0\ (DMA Ch-0 End signal)
14B	TEND1\ (DMA Ch-1 End signal)	14C	IEI (daisy-chain int. enable in)
15A	EXTRQ\ (Z-80 vectored interrupt)	15B	PHI (main CPU clock)
15C	IEO (daisy-chain out)	16A	+5V
16B	Ground	16C	Ground

In essence, a subset of the CPU bus is brought out on these pins. The lines are unbuffered, and are not intended to drive long backplane lines.

The pin assignments are such that for basic IO functions, only the two outer rows of the connector are used. This enables the cheaper 2-row connectors to be used. The centre row is needed only when a memory page is mapped on to an external board.

The intended mating connector is the same half-DIN41612 format, with wire-wrap tails. These are soldered through the expansion board, and are plugged into J1. This technique enables several expansion boards to be stacked on each other, if required.

The IEO pin is not connected on the CPU card. It is assigned for use by expansion cards, to enable the daisy-chain to be propagated through a maximum of 2 expansion cards. The method is shown below:



Each expansion board has 2 jumpers, J1 and J2. It may be seen that by setting these jumpers as shown, a daisy-chain of up to two vectored-interrupt expansion boards is possible.

The "E" clock output may be used to gate the RD\ and WR\ signals, as described for the IO combination chip. It also provides a convenient clock to control a logic analyser. For this purpose the *falling* edge is appropriate, and will give exactly one clock (with valid data available) per bus cycle. If a logic analyser is used, bear in mind that CPU accesses to internal chips usually do not generate external bus cycles.