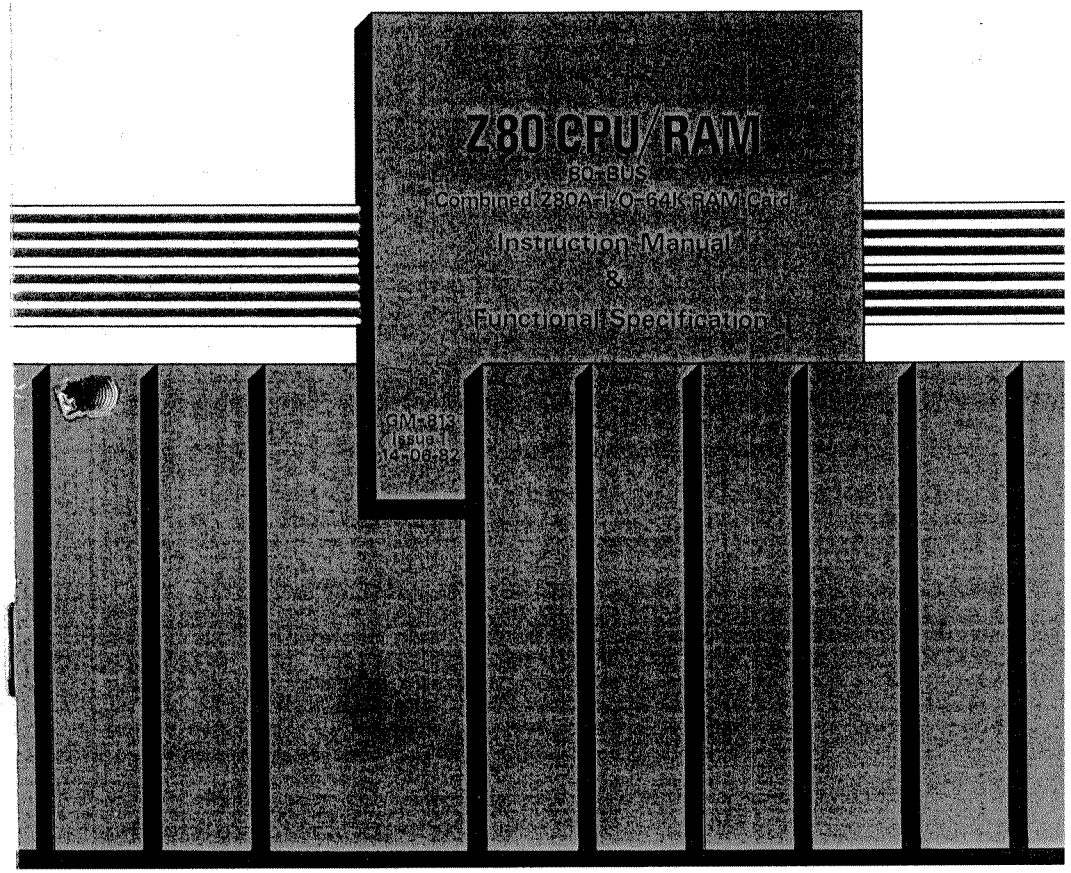
# The Gemini MultiBoard Microsystem






Gemini Microcomputers Ltd

18 Woodside Road Amersham Bucks HP7 0BH England.

## Introduction

The GM813 is a Z80 CPU card for use on GEMINI Multiboard systems. A great deal of flexibility has been built into the card to allow it to be used in a wide variety of configurations. It is possible to use the card by itself as, for example, a controller, or in combination with other 80-BUS compatible cards as a very powerful business or software development system including disk storage, video output and EPROM programming capability.

The CPU section of the card uses a Z80A processor, the Z80 being fully buffered to the 80-BUS specification and able to run at either 2 or 4 MHz. At power-on, or if reset is pressed, the program in the on-board 4K x 8 EPROM is entered. This may contain the main monitor for the system, or just the initialisation routines to load and execute an alternative system. In the latter case the EPROM may be switched out of the memory map. If necessary Wait states can be inserted in all memory cycles that access the EPROM.

A full 64K of RAM is included on the card. For those requiring more memory the RAM can be used in a "paged" system of up to a four banks of memory. The Gemini GM802 64K RAM card and the Gemini GM803 EPROM card both support this page addressing mode. In addition the card incorporates a memory-mapping system that enables it to address up to 512K of memory directly. Combined with the page-mode of operation this allows a total of 2Mbytes of memory to be handled by the card.

The input/output section of the card consists of a dual parallel interface in the form of a Z80A PIO, and a serial interface which includes programmable baud rates, full modem support signals, inputs and outputs at RS232 levels, and a 1200 baud Kansas city/CUTS tape interface. Switching between RS232 and cassette interfaces may be done under software control.

This manual is in three sections, a user section instructing the user in commissioning GM813, a functional description section detailing the design, and a copy of the 80-BUS specification.

Contents



## Commissioning

Carefully unpack your GM813 and examine it for any mechanical damage. In the event of any damage please inform your dealer immediately.

Your GM813 will have been shipped to you fully tested and working, all that may be required is for the board to be plugged into the bus. However please take the time to read through this manual as it should prove useful.

When plugging GM813 into the bus please take great care, excessive force should not be required, any difficulty that is encounted will (in all probability) be due to the keyway of the edge connector not slotting accurately into the slot in the edge of the card. Ensure that the card is plugged in with the edge connector going in first and the correct way around, it is not possible to plug the board in the incorrect way around because of the keyway. Power is connected to the card through the bus - refer to the 80-BUS description for further details.

## Input/Output Connections

There are four I/O connectors on the board edge - PIO, Serial, Aux. and Tape. The connections are detailed below.

### PIO Connector

The PIO (Parallel Input/Output Controller) has two eight bit data ports, one known as A and one as B, so they are numbered A0 to A7 and B0 to B7, A0 and B0 represent the least significant bits, A7 and B7 represent the most significant bits. In addition each port has two handshake lines, one for input and one for output. Below are the details of the connector on GM813 (marked 'PIO'). +5 volts and ground are also provided on the connector to drive a small amount of external circuitry. However, users should beware of drawing excessive currents. Further details of the operation of the PIO are given in the GM813 functional description. Full details can be found by obtaining an MK3881 PIO manual.

| | | | |
|---|---|---|---|
| B5 | 1 | 2 | B4 |
| B6 | 3 | 4 | B3 |
| B7 | 5 | 6 | B2 |
| /ARDY | 7 | 8 | B1 |
| /BSTB | 9 | 10 | B0 |
| /ASTB | 11 | 12 | /BRDY |
| A0 | 13 | 14 | Ground |
| A1 | 15 | 16 | Ground |
| A2 | 17 | 18 | Ground |
| A3 | 19 | 20 | +5 volts |
| A4 | 21 | 22 | +5 volts |
| A5 | 23 | 24 | A7 |
| A6 | 25 | 26 | Ground |

### Serial Connector

The Serial connector (marked 'Serial') is used for connecting RS232 type printers, terminals or modems to GM813. It is quite involved and so it will be explained pin by pin. Selection between RS232 and cassette interfaces is controlled by the /OUT 1 output of the 8250 UART. This means that the selection is under software control. See the 8250 section of the functional description, and the software manual for further details.

Pin 1 RLSD (Received Line Signal Detector). An input to GM813, this would normally come from a modem or similar piece of equipment. When positive it indicates that the modem etc, is receiving a signal. This signal is also known as Data Carrier Detect. On a DB25 type connector this would be pin number 8. The signal is at RS 232 levels.

Pin 2 +5 volt output for driving external circuitry. Maximum allowable current is 250mA.

Pin 3 SIN (Serial INput). Marking (high) when negative and spacing (low) when positive, this is the RS 232 data input to GM813. This signal is also known as RxD. The DB25 pin number is 2 if GM813 is to be the host, or 3 if GM813 is to be a peripheral. RS 232 levels.

Pin 4 /BAUDOUT. This signal, from the WD8250 UART, is a clock signal of 16x the 3250 receive clock. It is at TTL levels, is not symmetrical, and has not been buffered. Also detailed in the description of the link options.



Pin 5 RCLK. This is connected to the receive clock input pin of the 8250 and should be 16x the receive clock. It is normally connected to /BAUDOUT, but is available to allow an external clock input. This signal should be at TTL levels - do not drive with RS 232 levels. Also detailed in the description of the link options.

Pin 6 SOUT (Serial OUT). This is the RS232 data out from GM813, also known as TxD (Transmitted Data). DB25 pin number is 2 if GM813 is a peripheral or pin 3 if GM813 is the host. This is an RS232 level output, and must not be connected directly to TTL.

Pin 7 -12 volts output for powering external circuitry. Maximum load 100mA.

Pin 8 CTS (Clear To Send). An input to GM813, this would originate at a modem or similar device and indicates that the modem etc is ready to transmit data on behalf of GM813. Positive when clear to send, negative when not clear. RS 232 levels. DB25 pin 5.

Pin 9 DSR (Data Set Ready). An input to GM813, this would come back from a modem after it has attempted to establish a communications channel. It does not however indicate the existance of a communications channel. Positive to indicate the attempt has been completed. RS 232 levels. DB25 pin 6.

Pin 10 RTS (Request To Send). An output from GM813 which, when positive, tells the modem to transmit. If the link is half-duplex the receive will be supressed. When the RTS line goes low to high the modem will respond by taking the CTS line high and data may then be transmitted. When RTS goes low it must remain low until the modem has taken CTS low in response. RTS is DB25 pin 4.

Pin 11 Ground. This is the system ground and is not to be connected to safety ground. DB25 pin 7. (Not DB25 pin 1, which is safety ground.)

Pin 12 DTR (Data Terminal Ready) An output from GM813 this has to be positive to maintain the communications channel. When taken negative the modem will complete the current transmission, and then close down the channel. RS 232 levels. DB25 pin 20.

Pin 13 +12 volts for powering external circuitry. Maximum load 100mA.

Pin 14 Ground. Please see comments re. pin 11.

Pin 15 Ground. See above.

Pin 16 +5 volts for powering external circuitry. Maximum load 250mA.

In different manufacturers equipment the RS232 standard has been implemented in a variety of different ways. The instruction manual for the equipment being interfaced should be carefuly read. As the protocol has to be implemented on GM813 in software, it can easily be changed to suit individual requirements.

The RS232 outputs swing from -12 volts to +12 volts. The inputs will accept the same voltage swings, but they can also work with TTL levels. However, it must be pointed out that in this case the noise rejection will be impared. The threshold charactistics of the RS232 receivers can be modified by varying the values of the resistors R11 to R14 on GM813. The standard value is no resistor. (For further details see the 75189A (Texas) data sheet.)

The connector details of GM813s Serial connector are given below:

| | | | |
|---|---|---|---|
| RLSD | 1 | 2 | +5 volts |
| SIN | 3 | 4 | /BAUDOUT |
| RCLK | 5 | 6 | SOUT |
| -12 volts | 7 | 8 | CTS |
| DSR | 9 | 10 | RTS |
| Ground | 11 | 12 | DTR |
| +12 volts | 13 | 14 | Ground |
| Ground | 15 | 16 | +5 volts |

Aux. pins
---

Two pins are provided (adjacent to the 5 pin DIN socket and marked 'AUX') for switching an external device. The pin furthest from the board edge is connected to +5 volts, the pin closest to the board edge is connected to the collector of an NPN transistor, TR1. TR1 is switched "on" whenever the /OUT 1 of the 8250 goes high, /OUT 1 is also used to switch the 8250 input between the RS232 interface and the tape interface circuitry. The transistor could therefore be used to switch a tape recorder on or off. Precise details are not provided as the external circuitry will vary from tape recorder to tape recorder. Please, however, ensure that any relay used has an anti-backlash diode and does not sink more than 250mA into the transistor.

Five pin DIN socket
---

The DIN socket is provided for the tape interface and has two sets of links associated with it. Selection between RS232 and tape interfaces is controlled by the /OUT 1 output of the 8250 UART. This means that the selection is under software control. See the 8250 section of the functional description, and the software manual for further details. Details of the links are given below.

LK1 is to set the output level of the tape interface. By connecting the centre pin to the pin nearest the edge of the card the output will be about 100mV; by connecting the centre pin to the pin nearest the centre of the card the output level will be decreased to about 10mV. The correct pin can be selected by experimentation. GM813 is supplied with the output level set to 10mV. Data should be recorded at about OVU.

LK2 selects the input channel. If a mono recorder is in use this will not matter, however if a stereo recorder is used it is extremly important that the two channels are not mixed when the tape is played back. The data can, however, be recorded on both channels. This is because of the phase variances between the channels, which will result (if the two are mixed) in a "lumpy" frequency response. The right hand channel is recorded nearest the centre of the tape, so it is recommended that the centre pin of LK2 is linked to the pin nearest the edge of the card if a stereo recorder is used. Many mono machines however, only output on pin 3 of their DIN connectors. If this is the case then the centre pin should be connected to the pin nearest the centre of the card. As supplied GM813 has the left hand channel connected.

The output levels of tape recorders will vary over a very wide range. If GM813 is found to be insensitive R9 (47K) can be decreased. Any value of resistor (within reason) may be used.

| | | | |
|---|---|---|---|
| Ground | 2 | | |
| Output | 4 | 5 | Input |
| Output | 1 | 3 | Input |

View from back of plug.

VR1
---

The potentiometer on the edge of GM813, VR1, is to adjust the centre frequency of the cassette data recovery circuit. This has been electronically set at manufacture and should not be altered.

Reset Switch and HALT LED
---

Also provided on the edge of the GM813 card is a small push button, marked 'Reset'. Depressing this causes a reset pulse to be applied to the Z80 CPU, PIO, 8250 UART and line 14 of the bus (/RESET). A reset pulse is also generated when power is first applied to the card, and can also be generated by connecting a switch between line 10 of the bus (RESET SWITCH) and Ground. The result of a Reset will be to reinitialise the system.

Alongside the Reset Switch is a red LED that monitors the state of the Z80 HALT output. Refer to the Z80 manual for further details.

Link Options
===

There are a wide variety of link options on GM813. The appropriate links for most applications have been made during manufacture, however full details are provided below. All links should be made by soldering wire from one connection on a header plug to the other, but please take great care as an incorrect link may do damage and a poor connection will result in an unreliable system. Care must also be taken when links "crossover" on the linkblocks to ensure that shorts do not occur.

LK1 & LK2
---

Details are given above under "Five pin DIN socket".

LK3 UART Receive Clock
---

The 16x receive clock used in the 8250 is brought out of the device, and in most cases is fed straight back in. However, in some applications it is useful to have the receive clock connected to an external source. The RCLK input of the 8250 may be connected to the /BAUDOUT of the 8250 by LK3. LK3 will be found adjacent to pin 1 of the 8250 (IC7). For further details see the section on the serial connector.

TP2 UART Interrupts
---

The WD8250 UART has an interrupt output that indicates when certain operations have been completed. This line is connected to TP2 on the board which is adjacent to the 8250 (IC7). If it is required to use the 8250 under interrupt control this point may be cross connected to one of the inputs on the PIO (IC4) so that the 8250 can generate vectored interrupts via the PIO. The appropriate software for this would have to be incorporated in the system monitor.

LK4 & LK5 Memory device selection
---

The EPROM socket on the CPU card will accept either a 2716 type EPROM (2K x 8bits) or a 2732 type (4K x 8bits). The board will only accept the single-supply versions of these EPROMs,(eg I2716, TMS2516), not the three-rail versions (eg TMS2716). If a 2732 type EPROM is used then LK5 should be disconnected and LK4 should be inserted to connect the address line A11 through to pin 21 of the IC. If a 2716 type EPROM is used then LK4 should be disconnected and LK5 inserted to connect +5volts to pin 21 of the IC.

| EPROM type | LK4 | LK5 |
|---|---|---|
| 2716 | out | in |
| 2732 | in | out |

## LKB1 (IC35) Main System links

### System Configuration Option

Pin 14 on LKB1 is connected to the /RI (ring indicator) input of the 8250, the logic state of which can be read by the CPU via the 8250. If this pin is left unconnected the CPU will read a low, if linked to ground (pin 1) the CPU will read a high. This pin can be used by the system software to determine a power-on option of the card. Where implemented the appropriate position will be given in the software manual.

### Clock Options

There are a variety of clock options available on GM813, by the appropriate linking on LKB1. The 'Master Clock' is determined by the crystal (XTAL1, normally 16MHz) and various divisions of this frequency are present at LKB1 - /4 (4MHz) at pin 9 and /8 (2MHz) at pin 11.

The CPU clock, pin 6 of LKB1, is normally 4MHz and should therefore be connected to pin 9.

The WD8250 UART requires a clock input of 2MHz at pin 4 of LKB1, and this should therefore be connected to pin 11.

The 80-BUS clock line should be driven by GM813, and this is enabled by connecting the clock driver (at pin 7) to the bus line (at pin 8). In certain situations the 80-BUS clock line may be driven by another card. In this case the clock drive link (9-6) would be removed, and the CPU clock input (6) connected to the bus clock (8).

Finally, a 4MHz clock signal must be provided on the bus if the bus master (GM813) is not running at 4 MHz. If the bus master is running at 4 MHz there is no need to connect this pin. If the bus master is running at a different frequency then the 80-BUS AUX CLK line (at pin 5 on LKB1) should be connected to a 4 MHz clock. For further details refer to the 80-BUS specification.

### Wait States

With some EPROMs it is necessary to provide wait states to the Z80 during memory accesses. The wait states are enabled by linking pin 3 to pin 12 on LKB1. The wait states are only inserted in memory read cycles that access the on-board EPROM.

A summary of the functions of LKB1 is given below:

| | | | |
|---|---|---|---|
| GND | 1 | 14 | /RI input on 8250 |
| Not connected | 2 | 13 | Not connected |
| Z80 Wait input | 3 | 12 | EPROM Wait signal |
| 8250 clock | 4 | 11 | 2MHz clock (Xtal/8) |
| 80-Bus aux clock | 5 | 10 | Not connected |
| Z80 clock | 6 | 9 | 4MHz clock (Xtal/4) |
| Buffered 4MHz | 7 | 8 | 80-Bus clock |



## Memory Expansion

The GM813 supports two modes of increasing the memory capacity of the system in which it is being used beyond the conventional 64K limit of the Z80 processor. The following pages describe the two modes and their use.

### Page-Mode Memory expansion

With page-mode an entire memory board can be switched into or out of the memory map under software control. One particular IO port, port OFFH, is reserved to control this function. The bits of the port are divided into two halves, the upper four bits are used to write-enable a board, and the lower four bits to read enable a board. To simplify the amount of logic necessary on the memory boards to implement this feature the four-bit fields are used directly and are not decoded further. The functions of the bits are shown below:

| Bit | Function if set |
|---|---|
| 7 | Write enable page 3 |
| 6 | " " " 2 |
| 5 | " " " 1 |
| 4 | " " " 0 |
| 3 | Read enable page 3 |
| 2 | " " " 2 |
| 1 | " " " 1 |
| 0 | " " " 0 |

The 64K of memory on GM813 is placed in page 0.

As there is no subsequent decoding of the data fields the onus lies on the programmer not to set up an invalid condition. For instance if the page-mode system is being used with all four pages in use, in general only one board can be read enabled at any one time. If a value such as 1FH is written into port OFFH then when the Z80 attempts to read data from memory all four boards will attempt to put a byte on the data bus resulting in conflict between their data bus buffers and garbled data into the CPU. However it is perfectly possible to write-enable all of the boards simultaneously (by writing say OF1H into port OFFH) so that the same information can be written into each board.

In use the page-mode system requires a little thought to utilise in an application as the entire memory of the processor is switched. Either a common area of memory which is not switched has to be used to effect the transfer, or the identical program has to be written into the all the memory boards so that when the switch occurs the control program continues to run correctly.

### Extended addressing - Memory mapping

GM813 supports an alternative method for extending the addressing capability of the Z80 - memory mapping. GM813 supplies 19 address lines to the 80-Bus rather than the 16 of the Z80. This means that the board can directly address 256K bytes of memory, although the Z80 can only "see" 64K of it at any one time. The extension in the addressing capability is achieved by putting mapping registers on the top four address lines. These registers are used to translate or "map" the top four address lines to seven address lines.

In GM813 the mapping is done by two 74LS189 TTL RAMs which are connected to give a 16 word x 8 bit register array. These RAMs, which are addressed by the top four address lines (A12-A15) of the Z80, supply the seven high address lines (A12-A19) to the 80-Bus from their data outputs. As any value may be written into the RAM registers it possible to arrange for any 4K segment of the Z80's 64K physical address space to access any 4K segment in the 256K address space of the 80-Bus.

On Reset the system monitor initialises each register of the 74LS189s with its own address, thus the 16 registers 0-F contain the data bytes 00-0F. This means that initially there is no difference between the Z80's output address, and the address put out on the 80-Bus. However if we program say 2E into mapping register 0, then whenever the Z80 accesses an address in the range 0XXX the actual physical address put onto the bus will be 2EXXX.

This approach has a distinct advantage over page mode in that the system's memory can be re-allocated in amounts of 4K at a time, not entire memory boards. In fact the mapping scheme can work quite successfully with just the standard 64K of memory. Consider a real-time task that has to handle say seven identical machines. Let us assume that the control program is contained in the bottom 12K of memory and requires additional workspace for each machine controlled. When switching tasks the program has to switch workspaces. Without memory mapping this would have to be done by either saving the current data and copying in the new, or by writing the program so that all data was accessed indirectly via one of the index registers which could then be changed to point to the new data area. However with memory mapping, the program can be written to use a fixed area of memory at say 3000H. The data areas for the tasks can then be allocated to 4000H, 5000H, 6000H, 7000H, and so on. Then to switch to task 3 it is only necessary to write a 6 into mapping register 3. Thereafter any memory reference in the range 3XXX will actually be mapped to a physical address of 6XXX. To switch to the next task it is only necessary to write a 7 into mapping register 3, and so on.

Writing to the mapping registers
---

To simplify the hardware required to implement the memory-mapping feature of GM813, use is made of the special Z80 OUT instruction "OUT r,(C)", where the output port is indirectly addressed by register C. When this instruction is executed, the port address (register C) is placed on the low address lines (A0-A7) of the address bus, and register B appears on the high address lines (A8-A15). This means that the four most significant bits of register B will address the memory-mapping RAMs. If an OUT instruction is performed to port OFEH then a write strobe will be generated for the 74LS189s to latch whatever is on the data bus into the memory-mapping register selected by A12-A15. For example to write 2EH into memory mapping register 7 the following sequence of instructions should be executed:

```
LD   A,2EH          ;Data to be written
LD   B,70H          ;B high nibble = register address
LD   C,0FEH         ;C = memory-mapping port
OUT  (C),A          ;Write data into the selected register
```

The 74LS189s actually provide inverted data at their outputs, so an inverting buffer (IC45) is used for the 7 high address lines to correct for this. (It saves the programmer from some mental effort!)



GM813 Functional description
==========



Section 1 - Processor Circuitry
==========

1.1 Clock
---

The master clock is a crystal oscillator formed by two 74S04 inverters (IC's 37b/c), the inverters being biased into a linear region of operation by the two 820R resistors Rs 27 & 28. A crystal is connected across the two inverters to provide feedback and hence oscillation at the desired frequency. The oscillator will oscillate at the series resonant frequency of the crystal. The 47nF capacitor (C32) is used to couple the two inverters together, the 10pF capacitor (C31) being an anti-jitter capacitor. The output of the oscillator is divided by two 74LS113 flip-flops (IC 36) and one 74LS74 flip-flop (IC 29a) in succession to yield the oscillator frequency divided by 2, 4, & 8, all three clocks being available at LKB1. LKB1 is provided with clocks from four sources, three from the division chain and one from the bus. It also has provision for outputting to the CPU, 8250 UART, and to the bus. There is provision for one of two clocks on the bus, the first being the system clock (CLOCK) which must be the same clock as that used to drive the CPU, and the other an auxilliary clock (AUX CLK) for use by expansion cards when the CPU is not running at 4MHz. Many permutations are possible. However in the standard 4MHz system the master oscillator would run at 16MHz, the CPU at 4MHz, the 8250 at 2MHz, and the bus clock would come from the GM813. If the GM813 was run at 6MHz the master oscillator would run at 16MHz as before and the results would feed both the 8250 (at 2MHz) and the AUX CLK line on the bus (at 4MHz). The 6MHz for the CPU would come from the system clock line on the bus. The CPU clock is also connected to the wait state generator/dynamic RAM timing IC (IC 10) and the PIO. For both the CPU and PIO the clock has to swing from 0 volts to 5 volts, this swing being provided by a push-pull transistor arrangement (TR2 and TR3) driven by an 74S04 (IC 37e).

## 1.2 Reset

Reset can come from one of two sources. On power up a 47uF capacitor (C29) will be charged up via a 22K resistor (R23). While the set input of the 74LS74 (IC 29a) which is connected to the junction of the resistor and capacitor is less than approximately 2 volts, the Q output will remain high and this will ensure that the output of the 7407 (IC30a) is low. As a consequence, Bus line 14 and the reset input of the Z80 will be low. Once the 47uF capacitor (C29) has charged past the threshold of the LS74, bus line 14 and the reset input to the Z80 will go high.

The second source of a reset is either a high to low on line 10 of the bus or the closing of switch SW1. When a reset occurs during operation two criterions must be met.

a) The reset pulse must not be so long as to prevent dynamic RAM from being correctly refreshed.
b) The reset pulse must not occur during T3 of an /M1 cycle. If the reset pulse was to occur during T3 of an /M1 cycle, /MREQ will go indeterminate for one T state some 10 T states later. This could damage the contents of dynamic RAM.

The reset switch is connected between bus line 10 and ground. Bus line 10 goes via a 470 ohm resistor (R17) (which limits the current) to the input of a 74LS14 (IC 23a), the input having a 22K pullup resistor (R18) and a 47uF capacitor (C10) to ground to act as a debounce circuit. Users are however cautioned that this may not be sufficient in all cases and that in extreme cases two or more resets may occur. While the reset switch is open the input to the 74LS14 (IC 23a) is high and the output low. When the switch is closed the ouput of the LS14 goes high triggering the 74LS221 monostable (IC 24a). The monostable produces a pulse of about 50us duration on its Q output. When the Q output goes high it removes the reset signal from the 74LS74 (IC 29b) and puts a high on its D input. At the start of the next M1 cycle, signified by the output of IC23b going high, the high on the D input of the 74LS74 is clocked through to its Q output thus producing a Reset signal to the CPU via IC37f and IC30a When the monostable (IC24a) times out and its Q returns to the low state, a reset signal is once again applied to the 74LS74 (IC29b) taking its Q output low and thus removing the Reset signal from the CPU. It should also be noted that the reset is triggered on the high to low transition of line 10. If line 10 or the reset switch is held low multiple resets will not occur.

## 1.3 Reset jump

GM813 does not have the usual "reset jump" facility to force the instruction fetch following a reset to a particular address. Instead the Reset pulse to the Z80 also sets the set-reset flip-flop formed by IC16a/b. This holds one input of IC16c low, thus ensuring that the on-board EPROM (IC 33) is permanently selected. This condition persists until the set-reset flip-flop is reset by performing an "OUT" instruction to port OFFH, following which the EPROM will only respond to addresses in the range OF000H-OFFFFH. Thus following a reset the high address lines of the Z80 are effectively ignored, and the on-board EPROM appears throughout the entire memory space of the Z80. For example if a 4K x 8 EPROM is being used, the EPROM will appear from 0000-0FFF, 1000-1FFF, 2000-2FFF,.. .. E000-EFFF, F000-FFFF.

This approach allows the initialisation routines in the EPROM to set-up the memory map in the system correctly, before enabling the system RAM.



When a reset occurs the sequence of events is as follows.

a) A low on the /RESET line sets the set-reset flip-flop formed by IC16a and IC16b. This ensures, via IC16c, that the EPROM is permanently selected. It also puts the Z80 into its reset state and all control and address lines float.
b) The /RESET line goes high, the Z80 fetches the first byte of the first instruction with an /M1 cycle, the Z80 "thinks" it is fetching the instruction from 0000, however the EPROM is selected and returns the byte at its first address.

Normally the EPROM will start with an instruction to set the program counter in the Z80 to the correct address by executing JP OF003H (for example). This ensures the program counter will continue to fetch instructions from the EPROM when the normal address decoding of GM813 is enabled. Next the EPROM should contain the instructions to set up the memory-mapping RAMS (see below). Following this the monitor will write 11H to port OFFH to ensure that the on-board RAM is paged into the system (see below). The act of performing this final setting up instruction will reset flip-flop IC16a and IC16b and enable the normal memory decoding of GM813.

## 1.4 Wait state generation

If required GM813 can insert a wait state into any memory cycle which accesses the on-board EPROM. This option is enabled by connecting pins 3 & 12 on LKB1. If wait states are enabled the sequence of events is as follows.

a) /MREQ goes low and as a consequence the D input of the 74LS175 (IC 10c) goes high.
b) The /Q output of IC 10c goes low when the flip-flop is clocked by the low to high of the system clock at the beginning of T2. The /Q output of IC 10c is ORed with the Q output of IC 10d and with the EPROM select signal (/EOE) from IC28 by the 74LS27 (IC 14b). The output of the NOR gate is inverted in IC14c and goes to a 7407 which takes the /WAIT line of the bus low.
c) During the high to low transition of T2 the Z80 will sample the wait line and if it is low a wait cycle will be inserted.
d) The rising edge of the clock at the beginning of Tw (the extra cycle inserted by the wait state generator) will clock the second half of the 74LS175 (IC 10d), causing the Q output to go high. This will result in the /WAIT line being taken high.

## 1.5 NMI pulser

Provision has been made for the generation of an /NMI pulse from a high to low transition on Bus line 6. Line 6 goes via a 470 ohm resistor (R26) to the input of a 74LS14 (IC23c), the input having a 22K pullup resistor (R25) and a 47uF capacitor (C30) to ground to act as debounce circuit, the intention being that Bus line 6 would be connected to a switch in the same manner as the reset switch is connected to line 10. The output of the LS14 is connected to a 74LS221 (IC24b) which generates a negative going pulse from the low to high output of the LS14. The /NMI pulse is buffered onto Bus line 21 and into the Z80's /NMI pin with a 7407 (IC30b). As with the reset input, switch bounce may give more than one /NMI.

### 1.6 Bus control

The address lines from the Z80 are buffered onto the bus via 74LS244 type buffers (A0 to A7 by IC48, A8 to A11 by 1/2 IC 46) for the least significant 12 lines. The most significant 7 lines are buffered onto the bus via a 74LS240 inverting buffer which compensates for the inverted data from the memory-mapping RAMS IC34 and IC40. The only control is the /BUSAK control signal which, when asserted, tristates all the address lines. The control signals from the Z80 are buffered with a 74LS244 (IC 47) buffer which is controlled in the same manner by the /BUSAK, the only difference being that the outputs of the buffers are pulled up to +5 volts with 10K resistors to ensure that in the event of the bus being "floated" all the control lines go high. The data bus is bidirectional and uses a 74LS245 (IC 39) and as a consequence the direction of operation and the state has to be controlled. The state is controlled by the same /BUSAK which controls the other buffers, but is also disabled whenever the on-board EPROM is accessed. The direction in which the buffers "buffer" is determined by IC28, a programmable logic array, which selects the appropriate direction dependent on the states of three of its inputs, /RD, /IORQ and /M1. The data bus buffer will buffer "out" unless /RD is low (data into the Z80) or an /INTAK (interrupt acknowledge) cycle is being executed. The Z80 will execute an /INTAK cycle when it wants an interrupt vector. An /INTAK cycle is signalled by the Z80 taking both /IORQ and /M1 low. The direction input is low when the Z80 is inputting data from the bus and high when outputting data to the bus. The input signals to the Z80 (/RESET, /WAIT, /INT, /NMI, /BUSAK) are input directly to the Z80 from the bus with 2k2 pullup resistors.

## Section 2 - Memory Circuitry

### 2.1 Memory mapping and paging

On GM813 there are two methods of increasing the amount of memory available beyond the normal 64K limit of the Z80. The use of this facility, and the way in which the hardware operates in this area has already been descibed in the first section of this manual.

### 2.2 Memory enabling and decode

The on-board memory consists of an EPROM (either 2K X 8 or 4K x 8), and 64K of RAM. The EPROM is permanently decoded for addresses OF000H to OFFFFH, but may be switched entirely out of the memory map under software control. The RAM occupies memory in the range 00000-OFFFFH in page 0 of the paging system.

The three extended address lines (A16-A18) are connected to the NOR gate IC14a. This provides a high level whenever the A16-A18 are 0, and this signal (AEO) is used to enable the on-board memory. The NAND gate IC30 provides the address decoding for the on-board EPROM. Its output goes low if AEO is high, A12-A15 are high, and EEN is high. (EEN - EPROM ENable - is control signal froma spare output of the 8250 UART. If EEN is programmed low then the EPROM address decoding is disabled). This signal passes via IC16c and the inverter IC37a to the /CS input of the EPROM and also forms the /EPROM signal to the memory control circuitry (see below). Thus, if the EPROM is enabled (EEN is high , the EPROM responds to addresses in the range OFXXXH. The other input to IC16c is the one that forces the EPROM to be permanently selected on reset (see 1.3 above).



### 2.3 Memory & I/O data bus control.

The on-board memory and I/O share a common data bus. The control of the devices and the buffer is necessarily complex and in the case of GM813 the majority of the work is done by a single IC, IC28 which is a programmable logic array. Its four outputs are programmed to give control signals derived from the logical combination of its input signals. Its operation is best understood by considering the input signals connected to IC38, and then the ouputs can be examined, and the necessary input conditions for each will be given.

PAL input signals.

/MREQ,/IORQ,/RD,/WR,/M1,/RFSH - The standard Z80 control signals.
IEI,IEO - Standard Z80 interrupt enable daisy chain signals.
/RAMDIS - Goes low to disable any RAM in the system. Used by EPROM boards to overlay RAM in a system.
/EPROM - Goes low when the on-board EPROM socket is addressed.
/IO - Goes low when any on-board I/O device is addressed.
AEO - Goes high if the extended address line A16-A18 are low.
RDEN - Is high when the on-board RAM is paged in for reading.
WREN - Is high when the on-board RAM is paged in for writing.

PAL output signals.

In general the signals are generated from the standard Z80 control signals plus the additional ones noted below.

BDIR
Buffer direction. This is the simplest of the control signals, and sets the direction of transmission of the on-board data-bus buffers. Normally the Z80 buffer (IC39) is set to output data to the 80-Bus, and the memory/IO buffer (IC44) to input data from the 80-Bus. However if the Z80 executes a Read cycle, an Input cycle, or an interrupt acknowledge cycle, then the direction of the data-bus buffers is reversed. (In the latter case the Z80 will be reading the interrupt vector from the interrupting device).

/BUFEN
Buffer enable. This signal is used to enable the local data-bus buffer (IC44). The buffer must be enabled whenever a valid on-board RAM read/write cycle is executed,(RAM is read/write enabled, AEO is high and the EPROM is not being read), an on-board I/O cycle is being executed,(/IO is low), or an interrupt acknowledge cycle is in progress and the interrupt was generated on-board, (determined by IEI and IEO).

/EOE
EPROM Output Enable. This signal goes low whenever a memory read cycle is performed on the on-board EPROM. (/EPROM is low, AEO is high). It enables the output of the EPROM and also disables the Z80 data-bus buffer (IC39) via IC16d.

/CASEN
CAS enable. This signal is used to complete a valid memory read/write cycle on the on-board RAM,(AEO is high,not an EPROM read,/RAMDIS is high, and the RAM is Read/write enabled).

## 2.4 Dynamic RAM refresh

Refreshing of the on-board dynamic ram is carried out automatically by the Z80. However the refresh address counter in the Z80 (register R) is only 7 bits wide and can only handle RAMs with a 128-cycle refresh requirement. Some of the recent 64K dynamic RAMs (eg those from Texas Instruments) are designed for 256-cycle refresh in 4ms rather than the usual 128-cycles in 2ms. In order not to restrict the range of devices usuable with GM813 additional circuitry (IC26 and IC27) is included to extend the automatic refresh capability to encompass the 256-cycle refresh dynamic RAMs.

The 74LS00 (IC26) is connected as a multiplexor which switches the A7 address line to the memory between the A7 bus line, and the /Q output of the 74LS74 (IC27b). This multiplexor is controlled by the /RFSH control signal from the Z80, and so during refresh cycles the eigth address line comes from IC27b rather than the Z80. The D-type flip-flop IC27a is clocked by the OR of /MREQ and /RFSH and so samples the A6 address line during each refresh cycle, and thus follows bit 6 of the Z80's refresh counter. The Q output of IC27a is connected to the clock input of IC27b, and so whenever A6 in the Z80 refresh counter goes from low to high IC27b is clocked and changes state. In this way the refresh address range of the Z80 is extended to 8 bits.

## 2.5 Dynamic RAM timing

A memory access to the on-board 64K dynamic RAMs is started on every /MREQ cycle. The /MREQ signal passes through the OR gate IC5b to form the /RAS signal to the dynamic RAMs, latching A0-A7 into the RAMs. When /MREQ goes low it also removes the clear signal from the 74LS175 (IC10) and places a high on the D input of IC10c. On the next rising edge of the system clock the Q and /Q outputs of IC10c will change state. The Q output switches the 74LS257 address multiplexors to select the 8 high-order address lines, while the /Q signal passes through the two 74LS14s (IC23e & IC23f) and the OR gate IC5d to form the /CAS signal to the RAMs. The 74LS14s are included to provide the necessary delay to give the address multiplexors time to switch, and the OR gate allows the /CAS signal to be controlled by the /CASEN signal from the PAL (IC28).

The /CASEN signal from IC28 performs two functions. If it is not a valid read/write cycle for the on-board ram /CASEN remains high and no /CAS signal is generated for the RAMs. In this case their outputs remain in a high impedance state, and they only internally refresh the row selected by the A0-A7 address strobed in by the /RAS signal. If it is a valid cycle the /CASEN will go low to enable the /CAS strobe to the RAMs. However in the case of a write cycle the /CASEN signal is delayed until /WE enable is low. This is done so that the 64K dynamic RAMs perform an "early write cycle" in which their outputs remain in a high impedance state, thus allowing their DINs and DOUTs to be connected to a common bi-directional data bus.

IC10a and IC10b are used in a conventional manner to provide a precharge extender for the dynamic RAMs. Betwen memory cycles the dynamic RAMs require their /RAS and /CAS inputs to be high for a minimum period during which they precharge certain internal nodes in preparation for the next memory cycle. With the Z80 there is a very short time between the memory read in an M1 cycle and the following refresh cycle. The "precharge extender" is used to anticipate the end of the /MREQ signal in the M1 istruction fetch, and switches of the /RAS signal to the dynamic RAMs via the input to pin 5 of IC5c, thus ensuring that the specified precharge times can be met.



# Section 3 - Input/Output Circuitry

## 3.1 I/O address decoding

When the Z80 executes an input or output instruction the port address appears on the bottom eight address lines (A0 to A7). The bottom eight address lines are connected to the 256 X 4 bipolar PROM (IC 19). The D0 output of the PROM is used to enable the PIO. The D1 output decodes the "page-mode" set-up port of OFFH. The D2 output decodes port OFEH which is used to set-up the memory-mapping RAMs, and D3 is used to enable the 8250. In addition to the device enables, two further signals are generated, /IORD which is /IORQ ORed with /RD by the 74LS32 (IC 20c) and /IOWR which is /IORQ ORed with /WR by the 74LS32 (IC20d), these two signals being used by the 8250 UART.

The PROM is coded in the following manner.

a) Select the ports to which you wish the various functions to be assigned. The "page-mode" port should be OFFH and the "memory-mapping" port should be OFEH.
b) The 256 PROM locations are directly mapped to the 256 I/O ports which are available. One merely has to load the particular location in the PROM with the contents which will take the appropriate data output of the PROM low and as a consequence enable the specific device.
c) D0 will go low if the location contains an E, D1 will go low if the location contains a D, D2 will go low if the location contains a B and D3 will go low if the location contains a 7.

Below is a table of contents for the standard I/O PROM for GM813.

| PROM location | Contents | Function |
|---|---|---|
| 00 - B3 | F | No function |
| B4 - B7 | E | PIO |
| B8 - BF | 7 | 8250 UART |
| C0 - FD | F | No function |
| FE | B | Memory-mapper |
| FF | D | Page-mode |

## 3.2 PIO

The PIO provides two sets of eight bit input/output lines. The PIO is selected by a /CE from the port address decode PROM (IC 19). It also inputs the control signals /IORQ, /M1 & /RD from the system. The PIO uses these signals to determine the particular cycle that is being executed. /M1 is gated with the reset pulse from bus line 14 by the 74LS21 (IC21b). When a reset occurs /M1 is interrupted and the PIO is reset. It must be pointed out that the actual reset only takes place on the restoration of /M1. The PIO has two further outputs and four inputs from the system. /INT goes low to generate an interrupt. Interrupt priority arbitration with other devices in the vectored interrupt structure takes place via the IEI and IEO lines. The PIO is clocked with the system clock in the same manner as the Z80 itself from the same clock driver circuitry. Address lines A0 and A1 are input to the PIO to determine the particular port that is being addressed. The eight bit data bus to the PIO is buffered by the same 74LS245 (IC44) that is used to buffer the memory section of the GM813. The two sets of eight bit input/output lines appear on a 26 way IDS connector along with four ground lines and two +5 volt lines.

The PIO is divided into two eight bit ports, port A and port B, each port having a data port and a control port. The port allocations are as listed below. It must be explained that the location of the PIO in the logical port address map is described as a port and the input/output lines are often also described as ports, please do not get confused.

| | |
|---|---|
| Port A data | B4 |
| Port B data | B5 |
| Port A control | B6 |
| Port B control | B7 |

The data ports are read/write. The control ports are write only. The PIO has four modes of operation.

0) Output mode. The data lines are configured as outputs, data written to the data port will appear on the output lines.
1) Input mode. The data lines are configured as inputs, and data can be read via the data port.
2) Bidirectional mode. The configuration depends upon the state of the handshake lines.
3) Control mode. Inputs and outputs amongst the eight lines can be mixed.

When the PIO is reset it goes into input mode and the inputs float (users should however note that when the PIO is not powered the input/output lines are grounded). Changing the configuration is effected by writing to the control port pertaining to the appropriate data port. The sequence is as follows.

1) First load the interrupt vector. This is the least significant eight bits of the memory address where the Z80 expects to find the address of a routine which it will execute. The least significant bit (D0) must be 0. If you are not going to use vectored interrupts this stage can be omitted.
2) Tell the PIO its mode of operation.

| Mode number | Type of mode | Code |
|---|---|---|
| 0 | Output | 0F |
| 1 | Input | 4F |
| 2 | Bidirectional | 8F |
| 3 | Control | CF |

Load the control port with one of the above codes.

3) This next stage is only required if you have selected mode 3 at stage 2. In mode 3 you select which lines are going to be outputs and which lines are going to be inputs. This is done by selecting the lines which are going to be inputs and adding up their binary weighted value. If, for example, you wish bits 7, 6 & 2 to be inputs you add 80 + 40 + 04 giving C4. You then output C4 to the control port assigning the appropriate bits as inputs. The remaining bits will default to being outputs.
4) The next byte is only necessary if interrupts are being used. Its value is obtained by adding up the numbers in the table below appropriate to what is required.

| | | |
|---|---|---|
| Enable interrupts | 80 | (disable interrupts) |
| AND function | 40 | (OR function) |
| High state | 20 | (low state) |
| Mask will follow | 10 | (no mask) |
| + | 07 | |



On the left hand side of the table is the effect of adding in a particular number and on the right hand side (in brackets) is the effect of not adding in a particular number. 07 must always be added in.

5) The final byte is the mask. This is obtained in the same manner as the byte used to program which lines are inputs and which are outputs in step 3. The bits you do not wish to monitor should be added up in proportion to their binary weightings. For example, if you do not wish to monitor bits 6, 3 & 2, adding 40 + 08 + 04 will give 4C, and loading the PIO with 4C would mask out bits 6, 3 & 2 which as a consequence would not be monitored.

It is not possible, in this somewhat brief description, to go into complete detail on the use of a most versatile device. For complete details please see the PIO manuals from Mostek or Zilog.

However, here are three examples of the use of a PIO on the GM813.

1) PIO into output mode, load the control port with OF. Any data sent to the data port will be output.
2) PIO into input mode, the PIO will reset into input mode. To read the inputs simply read the data port. If, however, it is already in another mode, load the control port with 4F.
3) Running vectored interrupts from the 8250 UART.
   a) Load the Z80's I register with the eight most significant bits of the interrupt vector.
   b) Put the Z80 into interrupt mode 2 (IM 2).
   c) Load the locations of the interrupt vector with the address of the UART service routine.
   d) Load the PIO control port with the least significant eight bits of the interrupt vector. Note that the least significant bit has to be a 0.
   e) Put the PIO into mode 3 by loading the control port with CF.
   f) Make all the lines inputs by loading the control port with FF.
   g) We wish to, enable interrupts, OR all the bits, look for a high, and tell the PIO that a mask will follow, so 80 + 20 + 10 + 07 = B7, so load the control port of the PIO with B7.
   h) Now we wish only to monitor the least significant bit so the control port is loaded with FE.
   i) Now enable interrupts.

Points to remember, the IEI and IEO daisy chains on the system back plane must be set up, otherwise the PIO will never return from the interrupt (see the Gemini 80-BUS specification). TP2 (the 8250 interrupt output) should be linked across to A0 on the PIO. An interrupt will disable any further interrupts so the keyboard service routine should re-enable them.

NOTE: When the PIO generates an interrupt it takes the IEO line low to prevent any other device that is further down the interrupt daisy chain line from interrupting. ie Only devices with a higher interrupt priority will be able to interrupt its interrupt service routine. Once interrupted and under service, the PIO monitors the data bus, and when it sees the Z80 fetching a RETI instruction (return from interrupt) it finally takes its IEO line high to enable the lower priority devices. With GM813 the on-board PIO will only be able to detect the RETI instruction if it is fetched from the on-board RAM. If the interrupt service routine is located in the EPROM or another memory board, then the IEO output will remain low. If this is likely to be a problem the PIO should be reset by the service routine.

## 3.3 8250 UART

The 8250 is a very sophisticated asynchronous communications element. It includes a baud rate generator and full modem control facilites.

The 8250 receives its chip enable from the I/O decode PROM (IC19). It has been allocated ports B8 to BF. The read enable input /DISTR is fed from the 74LS32 (IC20c) which generates /IORD (I/O read). The write enable input /DOSTR is fed from the 74LS32 (IC20d) which generates /IOWR (I/O write). Reset from bus line 14 is inverted by the 74LS04 (IC13c) and connected to the reset input, reset being active high. The clock input to the 8250 comes from LK1 and in a 4MHz system would be 2MHz. Address lines A0, A1 & A2 are also input and are used to select between the registers. The interrupt output is is taken to TP2.

### 8250 registers

#### Line Control Register (port BB read/write)

Bits 0 & 1 determine the number of bits in each transmitted or received character.

| Character length | Bit 1 | Bit 0 |
|---|---|---|
| 5 bits | 0 | 0 |
| 6 bits | 0 | 1 |
| 7 bits | 1 | 0 |
| 8 bits | 1 | 1 |

Bit 2, determines the number of stop bits in each character. If a logic 1, 1.5 stop bits for a 5 bit character, 2 stop bits for the 6, 7 & 8 bit character lengths. If a logic 0, 1 stop bit whatever the character length.

Bit 3, parity enabled when logic 1, causes parity generation on transmit and verification on receive.

Bit 4, parity select, logic 0 for odd parity, logic 1 for even parity. Odd parity means that the character will have an odd number of ones. Even parity means the character will have an even number of ones.

Bit 5, stick parity, inverts the effect of bit 4.

Bit 6, set break, when logic 1 the serial output from GM813 is forced to the spacing state, i.e. negative. This is irrespective of transmitter activity.

Bit 7, DLAB bit, this bit is used to switch ports B8 and B9 between the receive/transmit buffers (B8), the interrupt enable register (B9), and the baud rate divisor registers. When logic 1 the divisors are accessable and when logic 0 the receive/transmit buffers and the interrupt enable register are accessable.



#### Line Status Register (port BD read/write)

Bit 0, data ready indicator, logic 1 whenever a complete character has been received and is ready to be read. Reset either by a read of the receiver data register or by writing a 0 to this location.

Bit 1, overrun error indicator, set to logic 1 if the receive buffer was not read prior to the current character being loaded in. Reset by a read of the line status register.

Bit 2, parity error indicator, set if a parity error is detected. Reset by a read of the line status register.

Bit 3, framing error, set if the received character did not have a valid stop bit. Reset by a read of the line status register. This is the most common type of error from tape interfaces.

Bit 4, break interrupt indicator, set to a logic 1 whenever the serial input to GM813 is held in spacing (-12 volts) for longer than a full character period.

Bit 5, transmitter holding register empty indicator, set to a logic 1 whenever the holding register is ready to accept a new character. Reset by loading the transmitter holding register (port B8).

Bit 6, transmitter shift register empty indicator, set to a logic 1 whenever the transmitter shift register becomes empty. Reset by the transmit shift register becoming active. This bit is read only.

Bit 7, there is no bit 7, this location is permanently set to logic 0.

#### Modem Control Register (port BC read/write)

Bit 0, DTR output, set to logic 1 to set the DTR output from GM813 positive. Reset to take the DTR output negative.

Bit 1, RTS output, set to logic 1 to set the RTS output from GM813 positive. Reset to take the RTS output negative.

Bit 2, /OUT 1 output, set to logic 1 to select RS 232 input and output, reset to 0 to enable the tape interface. Also controls the transistor TR1, which is switched on with the tape interface.

Bit 3, /OUT 2 output, used to enable the GM813 onboard memory, set to logic 1 to disable the onboard memory, reset to logic 0 to enable the onboard memory.

Bit 4, loopback when set to logic 1, normal operation when reset to logic 0. When set to logic 1 the following events occur.

a) The serial output is set to marking, logic 1.
b) The serial input is disconnected.
c) The output of the transmitter shift register is connected to the receive shift register.
d) The following status bits are linked, CTS to DTR, DSR to RTS, RLSD to OUT1, RI to OUT2.

This enables all the 8250's functions and software to be checked. It should be noted that the interrupts are still operational.

Bits 5 to 7, there are no bits 5 to 7, they are all set to logic 0.

Modem Status Register (port BE read/write)

Bit 0, delta Clear To Send (CTS), if set to logic 1 indicates that the CTS line has changed state since the modem status register was last read.
Bit 1, delta Data Set Ready (DSR), if set to logic 1 indicates that the DSR line has changed state since the modem status register was last read.
Bit 2, trailing edge Ring Indicator (RI), if logic 1 indicates that the /RI input to the 8250 has gone from logic 1 (high) to logic 0 (low) since the modem status register was last read. For further details on the ring indicator please refer to bit 6 of the modem status register.
Bit 3, delta Received Line Signal Detector (RLSD), if logic 1 indicates that the RLSD has changed state since the modem status register was last read.
Bit 4, Clear To Send (CTS) input, logic 1 if the CTS input to GM813 is positive, logic 0 if CTS is negative. If bit 4 (loopback) of the modem control register is a logic 1 this bit indicates the state of RTS (bit 1) in the modem control register.
Bit 5, Data Set Ready (DSR) input, logic 1 if the DSR input to GM813 is positive, logic 0 if the DSR input is negative. If bit 4 (loopback) of the modem control register is logic 1, this bit indicates the state of DTR (bit 0) in the modem control register.
Bit 6, Ring Indicator (RI) input, logic 0 if the /RI input is high, logic 1 if the input is low. If bit 4 (loopback) of the modem control register is logic 1, this bit indicates the state of OUT 1 (bit 2) in the modem control register. It should be noted that on the GM813 the /RI input to the GM813 is pulled up by a 10k resistor R59 and it is also connected to pin 7 of link block 1 where it may be grounded. It is intended that this should be used by the software to indicate the board status. If this is implemented further details will be in the software manual.
Bit 7, Received Line Signal Detect (RLSD) input, logic 1 if the RLSD input to GM813 is positive, logic 0 if RLSD is negative. If bit 4 (loopback) of the modem control register is set, this bit indicates the state of OUT 2 (bit 3) in the modem control register.

Interrupt Enable Register (port B9 read/write)

Bit 7 of the line control register (port BB) must be logic 0 for access.

Bit 0, Received Data Available interrupt, when set to logic 1, an interrupt will be generated whenever data is available to be read.
Bit 1, Transmitter Holding Register Empty interrupt, when set to logic 1, an interrupt will be generated whenever the transmit holding register becomes empty.
Bit 2, Receiver Line Status interrupt, when set to logic 1, an interrupt will be generated whenever any of bits 1 through 4 of the line status register go high to indicate an error condition.
Bit 3, Modem Status interrupt, when set to logic 1, an interrupt will be generated whenever any of bits 0 through 3 of the modem status register go high to indicate a change in modem status.
Bits 4 through 7, always set to logic 0.



Interrupt Identification Register (port BA read only)

Bit 0, Interrupt Pending, when logic 0 an interrupt is pending. When the interrupt identification register is accessed all interrupts are frozen and no further interrupts will be acknowledged until the cause of the interrupt has been serviced or cleared.
Bit 1 = logic 1 & bit 2 = logic 1,
Highest level of interrupt priority, caused by a Receiver Line Status Register interrupt. Source of error, Overrun error, Parity error, Framing error or Break interrupt. Reset by reading the Line Status Register.
Bit 1 = logic 0 & bit 2 = logic 1,
Second highest level of interrupt priority, caused by receive data becoming available. Reset by reading the receiver buffer register.
Bit 1 = logic 1 & bit 2 = logic 0,
Third highest level of interrupt priority, caused by the transmitter holding register becoming empty. Reset by either reading the interrupt identification register or by writing into the transmitter holding register.
Bit 1 = logic 0 & bit 2 = logic 0,
Fourth highest level of interrupt priority, caused by a modem status register interrupt. Source of the interrupt is a change in status of one of the following signals, CTS, DSR, RI or RLSD. Reset by reading the modem status register.
Bits 3 through 7, set permanently to logic 0

Baud Rate Generator

Least significant byte port, B8 (read/write)
Most significant byte port, B9 (read/write)
Bit 7 of the Line Control Register (port BB) must be set to 1 for access.

The baud rate is generated by dividing down the clock input of the 8250. The equation for determining the baud rate is as below.

$$\text{Divisor} = \frac{\text{8250 clock frequency}}{\text{(Baud rate x 16)}}$$

The divisor should be split into two bytes. The most significant byte should be loaded into the most significant divisor latch, the least significant byte being loaded into the least significant divisor latch. It should be noted that,
a) The divisor is in Hex.
b) The bytes should be loaded even if zero (some 8250's have been observed to power up loaded with FF).

The divisors for a 2MHz clock are tabulated below. The numbers in the hex column should be loaded.

| Baud Rate | Divisor (decimal) | Divisor (hex) MSB | LSB |
|---|---|---|---|
| 50 | 2500 | 09 | C4 |
| 75 | 1667 | 06 | 83 |
| 110 | 1136 | 04 | 70 |
| 134.5 | 929 | 03 | A1 |
| 150 | 833 | 03 | 41 |
| 300 | 417 | 01 | A1 |
| 600 | 208 | 00 | D0 |
| 1200 | 104 | 00 | 68 |
| 1800 | 69 | 00 | 45 |
| 2000 | 63 | 00 | 3F |
| 2400 | 52 | 00 | 34 |
| 3600 | 35 | 00 | 23 |
| 4800 | 26 | 00 | 1A |
| 7200 | 17 | 00 | 11 |
| 9600 | 13 | 00 | 0D |

Data Holding Register (port B8 write only)
---

Bit 7 of the line control register (port BB) must be logic 0 for access.

Data from this register is output from the 8250 in serial form.

Receiver Buffer Register (port B8 read only)
---

Bit 7 of the line control register (port BB) must be logic 0 for access.

This register contains the received data.

Further data on the 8250 can be obtained from the relevant data sheet. (e.g. National Semiconductor INS8250), or see An Introduction to Microprocessors Volume 3, by Osborne.

3.4 RS 232 interface
---

The RS 232 interface is implemented with the use of two IC's. A 75188 (IC 3) is used for output and converts from TTL levels to RS232 levels (+/- 12 volts). The 75188 also inverts the signal in that a high from the 8250 produces a -12 volt signal and a low produces +12 volt signal. It should be noted that the discussion of the 8250 (section 3.3) referred to the signals output from the GM813 and not from the 8250. The RS 232 data output from GM813 can be inhibited by the /OUT 1 signal, which is also used for switching between the tape and RS 232 interfaces.

The RS 232 input buffering is handled by a 75189A (IC 2). The 75189A translates RS 232 levels to TTL levels. The signal is inverted in the same manner as the 75188, a negative signal producing a TTL high output. The discussion of the 8250 (section 3.3) made reference to signals into the GM813. The signal will be inverted between the 75189A and the 8250. The hysteresis of the 75189A can be adjusted by varying the values of Rs 12, 13, 14 and 15. The GM813 will have been shipped without any resistors in these positions as under normal circumstances none should be required. If however particular operating conditions required different hysteresis, resistors can be added. Please first read the data sheet for the 75189A. It is possible to input at TTL levels to the GM813 but the noise margin will be reduced. Further data on the 75188 and and the 75189A may be obtained from Texas Instruments Data Sheets.



3.5 Tape interface
---

The tape interface is based on the original Kansas City format (aka CUTS). The original specification called for a low (from the UART) to be represented by four cycles of 1200 Hz and a high to be represented by eight cycles of 2400 Hz. This resulted in a transmission rate of 300 baud (300 bits per second (30 characters per second)). To speed up the data rate this has been changed to 1200 baud by representing a low with one cycle of 1200 Hz and a high with two cycles of 2400 Hz. The encoding is done in the following manner by the 74LS113 (IC 6), which consists of two JK flip-flops. If the J input of an LS113 is held high and the K input is also high the falling edge of the clock pulse will cause the Q output to change state. If the K input is then taken low the Q output will go high on the next falling edge of the clock. If both inputs are low the outputs will not change on the falling edge of the clock. The net result is a synchronous programmable frequency divider. The serial data from the 8250 goes via a 74LS04 inverter (IC13f) to the K input of the 74LS113 (IC6a). When the data output of the 8250 is high the input to the LS113 is low. This has the effect of causing the Q output of the LS113 (on the next high to low of the clock) to go high. The Q output of IC6a is connected to both the J and K inputs of IC6b. The net effect of taking both the inputs high is that the Q output of IC6b will "toggle" on each high to low of the clock. This will result in a frequency of half the clock frequency being output. If, however, the serial data output of the 8250 is low the K input of the 74LS113 will be high. This will have the effect of causing IC 6a to "toggle" i.e. the output will change state each time the clock goes high to low. This will have the net effect of dividing the clock to the LS113 (both parts) by four, resulting in an output frequency of 1200 Hz. The output of IC 6b is connected to a combined low pass filter and attenuator. The low pass filter is formed by the 47k resistor R7 and the 10nF capacitor C7. The low pass filter prevents the harmonics in the encoded data signal causing problems. The encoded data signal is connected to one of the pins of link 1 and to a potential divider formed by the 4k7 resistor R11 and the 470 ohm resistor R10. A second pin on link 1 is connected to the junction of the two resistors to provide the option of a low level signal out of the GM813.

The clock to the LS113 is the 16x baud rate clock used to clock the shift register in the 8250, divided by four by the 74LS74 (IC 11) dual D-type flip-flop.

Tape decode circuitry
---

The signal from the tape is attenuated by Rs 8 and 9 which form a potential divider. In the standard GM813 these resistors will both be 10k, however they may be changed to suit individual conditions. The signal is coupled to the input of the XR2211 by a 100nF capacitor (C5).

The XR2211 is a FSK (Frequency Shift Keying) decoder consisting of a phase comparator, a VCO (voltage controlled oscillator) (which together with the phase detector forms a phase lock loop), a voltage comparator and a lock detector. The signal is input to the 2211 which amplifies and limits it. It then goes to a phase detector which generates a voltage which is proportional to the phase difference between an incoming signal and the onboard oscillator. The free running frequency of the oscillator is determined by the 22nF capacitor (C1) and can be fine tuned by the 10k variable (RV1). The output of the phase comparator appears at pin 11 of the 2211. The output of the phase comparator is filtered with a 4n7F capacitor (C3) and coupled back to the VCO via a 33k resistor (R3), thus forming a phase locked loop. The output of the phase comparator is further filtered by the 100k resistor (R4) and the 2n2F capacitor (C4). It is then input to the voltage comparator (pin 8).

The voltage comparator compares the voltage from the phase detector with a reference and causes the output to go high or low depending on whether the incoming frequency is lower or higher than the free running frequency of the VCO. The data output is connected back to the input to the voltage comparator via a 510k resistor (R5), this acting to speed up the transitions. The data output is also connected to a 5k1 pull up resistor (R1) and to the input of the 74LS04 (IC13e), the inverter being required as the decoder produces a low for a high frequency input and a high for a low frequency input.

The output of the LS04 is connected to the 74LS00 (IC12). IC12 is connected as a multiplexor, and under the control of the /OUT1 signal from the 8250, switches either the output of the 2211 (IC1) or the RS232 input to the serial input of the 8250.

The reference voltage against which the output of the phase comparator is compared is available at pin 10, where it is also decoupled by the 100n capacitor (C2). The 2211 also has a lock detector which will inhibit the output of spurious data from the 2211. The output of the 2211 will go low when the phase lock loop goes out of lock. The lock detection circuitry has a time constant formed by the 100k resistor (R6) and the 33nF capacitor (C6) associated with it.

### 3.6 Memory mapping RAMs & Page mode latch

The memory mapping RAMs (IC34 & IC40) and the page-mode latch (IC22) are both clocked in the same manner. The I/O port decode signals from IC19 are ORed with the /IORQ./WR signal from IC20d in IC20a and IC20b to produce the appropriate clock signals.

### 3.7 I/O Data Bus Control

The I/O section of GM813 shares the same 74LS245 (IC44) data bus buffer as the memory section. The description of the control circuitry can be found in section 2.3

## Section 4 - PROM Types

The recommended part is the Texas 74S287, however here is a list of equivalents. If you require a custom PROM for a particular application please contact a Gemini distributor.

| Brand | Part number |
|---|---|
| AMD | 27S21C |
| Fairchild | 93427DC |
| Futitsu | MB7057 |
| Harris | 7611-5 |
| Intel | 3601 |
| Intersil | 5623C |
| MMI | 6301-1 |
| Nat semi | 74S287 |
| NEC | uPB423D |
| Signetics | N82S129 |
| Texas | 74S287 |



## 80-BUS - a functional description

The Multiboard range of 8"x8" cards have been designed so that they may be used together in a wide variety of configurations. All of the cards have a 77 way PCB edge connector which carries signals to the 80-BUS specification, as detailed below. Cards are simply connected together by soldering the edge connectors supplied with the cards into a motherboard, and then plugging the cards into the connectors. The motherboard provides interconnection of each of the edge connector pads on each card to the same pad on every other card. The only exceptions to this are the `daisy chain' lines, and details of these are given in the specification.

Power is supplied to the cards by connecting the relevant supply rails to the motherboard. Each power supply rail occupies two or more bus lines, and in these cases should therefore be connected to all of the relevant bus lines, this is particularly true of the Ground (0V) rail.

80-BUS has many similarities to Nascom's Nasbus. The Nasbus, unlike many other bus systems, has had a very ordered development. However, when we started developing the Multiboard range of cards it became apparent that a new revision to the Nasbus specification was urgently required. After a great deal of careful thought and many hours of deliberation the following document was drawn up. It expands on the third issue of the Nasbus functional specification giving timing and other details never previously published and also attempts to anticipate some of the possible future developments of the bus.

The original Nasbus specification made provision for the extra address and data lines of 16 bit processors. Careful consideration reveals that the bus would not be suitable for this, and so a number of new signals have been defined for the lines made free. The importance of good ground signals can not be overemphasised, and so extra ground lines have also been added.

When defining this bus a great deal of thought went into deciding whether or not to maintain the NAS MEM, NAS IO, and DBDR signals. These signals are particular to Nascom 1 (and NAS IO also to Nascom 2) and are unlikely to be required by any future cards. They therefore constitute a `nuisance'. However, for the sake of compatibility, to avoid the S100 situation, and with pressure from INMC80, the Nascom Users' Club, it was decided that 80-BUS would maintain support for these signals.

Because of the above considerations 80-BUS remains fully Nascom 1 and 2 compatible. It is Gemini Microcomputers intention to allow any manufacturer who produces a card that fully complies with the 80-BUS specification to advertise accordingly. Cards should be submitted for approval. The 80-BUS is a Z80 bus and no attempt has been made to make it compatible with any other processor.

One final point relevant to the design of 80-BUS compatible cards is that all cards, including the bus master, should provide a means for being switched out of the memory map under software control. This may be by means of implementing the Page Mode structure, or by some alternative method. This condition also applies to any I/O card that is memory mapped.

80-BUS pin allocation
---

| PIN | | SIGNAL | DESCRIPTION |
|---|---|---|---|
| 1 | | GND | Ground |
| 2 | | GND | Ground |
| 3 | | GND | Ground |
| 4 | | GND | Ground |
| 5 | | CLOCK | System clock |
| 6 | * | /NMI SW | A low on this line initiates a short pulse on line 21 |
| 7 | | RSFU | Reserved for future use |
| 8 | | AUX CLK | 4MHz clock signal (optional) |
| 9 | * | /RAM DIS | Ram disable |
| 10 | * | /RESET SW | Reset switch |
| 11 | * | /NAS MEM | Memory decode to Nascom 1 |
| 12 | * | /NAS IO | I/O decode to Nascom 1 and 2 |
| 13 | * | /DBDR | Data bus drive, used to change the direction of the data bus buffers on the Nascom buffer board or Gemini `Supermum'. |
| 14 | * | /RESET | 50uS reset pulse, resets entire system. |
| 15 | | /HALT | Z80 halt signal |
| 16 | | /BAI | DMA |
| 17 | | /BAO | daisy chain |
| 18 | | /BUSRQ | Z80 bus request |
| 19 | | IEI | Interrupt |
| 20 | | IEO | daisy chain |
| 21 | * | /NMI | Z80 NMI line, (not used by Nascom 1) |
| 22 | * | /INT | Z80 interrupt line |
| 23 | * | /WAIT | Z80 wait line |
| 24 | | /RFSH | Z80 refresh signal |
| 25 | | /M1 | Z80 opcode fetch signal |
| 26 | | /IORQ | Z80 input/output signal |
| 27 | | /MREQ | Z80 memory signal |
| 28 | | /WR | Z80 write signal |
| 29 | | /RD | Z80 read signal |
| 30 | | A0 | |
| 31 | | A1 | |
| 32 | | A2 | |
| 33 | | A3 | |
| 34 | | A4 | |
| 35 | | A5 | |
| 36 | | A6 | |
| 37 | | A7 | |
| 38 | | A8 | Z80 16 bit |
| 39 | | A9 | address bus |
| 40 | | A10 | |
| 41 | | A11 | |
| 42 | | A12 | |
| 43 | | A13 | |
| 44 | | A14 | |
| 45 | | A15 | |
| 46 | | A16 | Optional implementation ) |
| 47 | | A17 | for extended ) Implemented on GM813 |
| 48 | | A18 | addressing. ) |
| 49 | | GND | Ground to seperate the data and address busses. |
| 50 | | D0 | |
| 51 | | D1 | |
| 52 | | D2 | |
| 53 | | D3 | Bidirectional data bus. |
| 54 | | D4 | |
| 55 | | D5 | |
| 56 | | D6 | |
| 57 | | D7 | |
| 58 | | RSFU | Reserved for future use |
| 59 | | INT 0 | Interrupt |
| 60 | | INT 1 | request |
| 61 | | INT 2 | lines |
| 62 | | INT 3 | |
| 63 | | /PWRF | Powerfail warning |
| 64 | | AUX PWR | Backup power |
| 65 | | NDEF 1 | Not to |
| 66 | | NDEF 2 | be defined |
| 67 | | GND | Ground to seperate power and signal lines. |
| 68 | | -5V | |
| 69 | | -5V | |
| 70 | | -12V | |
| 71 | | -12V | |
| 72 | | keyway | |
| 73 | | +12V | |
| 74 | | +12V | |
| 75 | | +5V | |
| 76 | | +5V | |
| 77 | | +5V | |
| 78 | | +5V | |

Notes
---

1) * is an open collector line.
2) IEI to be linked to IEO on cards not using the interrupt daisy chain.
3) /BAI to be linked to /BAO on cards not using the DMA daisy chain.
4) Bus drivers must be able to drive 75/15 U.L.
5) Bus receivers must not load the bus past 1/0.25 U.L.
6) Bus master to pull up all open collector lines with 2k2.
7) Bus master to pull up the following lines with 10k, /HALT, /MREQ, /IORQ, /RD, /WR, /M1, /RFSH.
8) Bus timing reference point is pin 6 of the Z80. As the bus is in essence a buffered Z80, the timing of bus signals is as the Zilog/Mostek Z80 data book. All Z80 signals are buffered onto the bus with 20nS +/- 10nS buffers, the sole exception being the bus clock which should be 20nS (+/- 10nS) ahead of the Z80 clock (pin6). The timing of other signals is detailed in the description of the particular signal. All expansion card timing must, however be referenced to the bus.
9) Cards using /BAI, /BAO, IEI & IEO should pull them up with 2k2.
10) Bus termination. Long buses may require termination. 220R on each line to a 2.6V low impedance source should solve 99% of problems.
11) Grounding. The ground line to the PSU should be as short as possible and as thick as possible.
12) The names of the various bus signals are as detailed above, please do not change them or abbreviate them, ie AUX CLK not AUX CLOCK or A CLOCK etc.

Comments
---

The following is a line by line description of the bus and should help resolve any ambiguities.

Lines 1-4, GND.

The quality of the system ground cannot be overemphasised. Ground noise problems were at the root of the now infamous Nascom "Memory plague". The faster that systems go the more critical the noise problem will become. Noise problems will manifest themselves as a generally unreliable system with a predilection to do "odd" things.

Line 5, CLOCK.

This line is important as all the bus timing is derived from it. It should spend at least 46% of its time below VOL (0.4V) and at least 46% of is time above VOH (2.4V), it has the other 8% spare to go up and down. The clock on the bus should be 20nS (+/- 10nS) ahead of the clock on pin 6 of the Z80.

Line 6, /NMI SW.

Provision has been made on the bus for an NMI switch and this line is to be held high by the bus master. Grounding it will initiate a short pulse on line 21 and the Z80 /NMI input. Users are cautioned that switch bounce may cause more than one NMI.

Line 7, RSFU.

This line is reserved for allocation at a later date, please do not use.

Line 8, AUX CLK.

This line is a new allocation. Many boards (eg disk controllers) require a 1, 2 or 4 MHz signal. This was easily provided when the CPU clock was 2 or 4 MHz, however the advent of the 6 MHz Z80 changes the situation. Any bus master not running at either 2 or 4 MHz must provide a 4 MHz clock on this line. Designers of expansion cards should take note of this and provide a link to allow the board to use this line instead of line 5.

Line 9, /RAM DIS.

This signal is intended to prioritise memory. Normally this signal would be generated by memory on the bus master, an EPROM card or any other high priority memory when a memory read took place. A RAM card would normally gate /RAM DIS with the output buffer, so that in the event of /RAM DIS being asserted the output buffer would fail to be enabled, this would have the effect of "overlaying" RAM with EPROM/ROM. /RAM DIS should not inhibit a write cycle; it should also remain high for any cycle apart from a memory read.

Line 10, /RESET SW.

A high to low on this line will initiate a reset cycle. It is intended that a switch be connected between this line and ground. The actual RESET line is line 14.

Line 11, /NAS MEM.

This signal is only used by Nascom 1 and is asserted when a Nascom memory address is detected. It would normally be provided by a memory board and would typically be 0000H to 0FFFH or F000H to FFFFH. This is an obsolete signal and no new boards that require it should be designed. This line used to be called MEMEXT.

Line 12, /NAS IO.

This line used to be called IOEXT and many people persist in still calling it that, even though the two are different. In its original form (note that it was active high) it would be taken high to indicate an I/O address external to the Nascom, in its current form it is taken low to indicate a Nascom I/O address, that is to say that it looks for a Nascom I/O address as opposed to looking for an external address. /NAS IO should be taken low within 50nS of a Nascom I/O address and /IORQ, (referenced to the bus). In its original form the onboard ports on the Nascom would remain enabled for a short fraction of an external I/O cycle (the time taken to detect an external address and assert IOEXT) and this was the cause of many obscure problems. If you have problems a good test is to write a short machine code routine to continuously write 80H to port 08H. If the breakpoint register display comes up you have a problem, if not you don't. /NAS IO is a obsolete signal utilised by Nascom 1 and 2 and all new designs should incorporate full I/O decoding.

Line 13, /DBDR.

This signal is used by the Buffer board and Supermum with Nascom 1. It controls the direction of the data bus buffers. When an expansion card outputs data to the bus this line must be taken low, normally this is the same signal as used to turn on the output buffers. Despite being only used by expanded Nascom 1s it is felt that this signal must be provided on all expansion cards. While /DBDR should ideally go low before the data bus driver is enabled, it must be low within 30nS of the data bus driver being enabled and must release /DBDR within 30nS of the data bus drivers being disabled.

Line 14, /RESET.

This line is the "cleaned up" version of line 10. It is important that the falling edge of the reset pulse on this line be synchronised with the falling edge of /M1, and the bus master must provide the appropriate logic to take care of this. The last issue of the Nasbus specification called for a 10uS reset pulse. This has now been extended to 50uS as chips in the 179X family require a 50us pulse. Deep investigation of the matter has yet to yield a 179X chip that can tell the difference between a 10uS pulse and a 50uS pulse. N2 owners who are concerned by this should substitute a 10nF capacitor for C1 (1nF). Supermum owners need not worry as this has been taken care of.

Line 15, /HALT.

Z80 halt signal. Up to this point in time nobody has used it, but it is there.

Lines 16 17 18, /BAI /BAO /BUSRQ.

/BAI and /BAO are the DMA daisy chain. If an expansion card wishes to take control of the bus (an expansion card is any bus card which is not a bus master) it asserts /BUSRQ; the bus master will respond by taking /BAO low. The mother board connects line 17 to 16 between each slot, ie line 17 of the bus master will go to 16 of the adjacent card, line 16 at the bus master is not used although it can be a test point. Between all subsequent cards line 17 goes to line 16 (for full connection details see the section on daisy chains). Cards that do not use the DMA facilities should connect 17 to 16. /BAO will be fed into the /BAI of any potential DMAing device, and the /BAO of the same device will go into the /BAI of the next and so on. If a potential DMAing device has not asserted the /BUSRQ line it will pass on the signal. When the signal reaches the device which originally asserted the /BUSRQ line it will hold /BAO high, at this point it will have taken contol of the bus. The highest priority device is that nearest the bus master.(CPU card).

Lines 19 20, IEI IEO.

Interrupt daisy chain for vectored interrupts. The IEI input of the highest priority device is held high, the IEO output of that device goes to the IEI input of the device with the second highest priority, the daisy chain is continued until the device with lowest priority is reached, its IEO is not connected. It is recommended that line 20 of the bus master is linked to line 19 of the adjacent card and so on down the bus. As the interrupt daisy chain does not involve the Z80 it is possible to move the bus master from slot to slot and vary the level of interrupt priority of the devices on the bus master. The DMA daisy chain however does involve the Z80, and the bus master must always be to one side of the expansion cards which may generate a /BUSRQ. If the motherboard is linked in the recommended manner the device with highest interrupt priority is nearest to the bus master. If the daisy chain is connected the other way arround a problem could arise as the Z80-DMA can also generate interrupts. For further details on connections see the section on daisy chains.

Line 21, /NMI.

A short pulse will be generated on this line by the bus master from a low on line 6 (/NMI SW). On Nascom 1 the NMI is used in the single step

Line 22, /INT.
Used for the Z80 maskable interrupt. For full details see the book "Z80 family program interrupt structure" available from Zilog.

Line 23, /WAIT.
Used to insert wait states into Z80 machine cycles. Expansion cards that require wait states should provide them.

Line 24, /RFSH.
Used to control the refreshing of dynamic RAM. It should be noted that a refresh cycle is a memory cycle and designers should take appropriate steps. The I register contents will appear as the top eight bits on the address bus during a refresh cycle.

Line 25, /M1.
Z80 /M1 used to indicate an opcode fetch, also used (in conjunction with /IORQ) to indicate an interrupt acknowledge cycle.

Line 26, /IORQ.
Used to indicate a Z80 I/O cycle. The port address will be on the bottom eight address lines (A0 to A7). The top eight will have the contents of the A register on them. If /IORQ is asserted with /M1 it indicates an interrupt acknowledge cycle and the Z80 will expect to receive an interrupt vector.

Line 27, /MREQ.
Used to indicate a Z80 memory cycle.

Line 28, /WR.
Used to indicate a Z80 write cycle, asserted in conjunction with /IORQ or /MREQ.

Line 29, /RD.
Used to indicate a Z80 read cycle, asserted in conjunction with /IORQ or /MREQ. It should be noted that /RD is not asserted during an interrupt acknowledge.

Lines 30 to 45, A0 to A15.
Z80 address lines, should be tristated during a /BUSAK.

Lines 46,47,48, A16, A17, A18.
Optional implementation for extended addressing, should be tristated during a /BUSAK. These lines are impemented on GM813 CPU/RAM board.

Line 49, GND.
An additional ground line to reduce system noise. Must be implemented on both the mother board and on expansion boards.

Lines 50 to 57, D0 to D7.
Z80 data lines, should be tristated during /BUSAK.

Line 58, RSFU.
This line is reserved for allocation at a later date. Please do not use.

Lines 59 60 61 62, INT 0 1 2 3.
Interrupt request lines, used to generate interrupt vectors from devices that are not capable of generating their own interrupts. These lines would be monitored by an interrupt controller which would be capable of generating interrupt vectors, the controller must be capable of being programed with the sense of a particular line (i.e. whether its active high or



active low) and the vector. A device unable to generate a vector would assert one of these lines when it required to interrupt, expansion cards availing themselves of this facility should provide it via links so that the particular line can be selected by the user.

Line 63, /PWRF.
Powerfail warning. This line is to be taken low 100mS before the power rails drop by more than 5% and held low until 100mS after the power on reset. For use by backup memory etc. An optional signal which would be provided by the power supply circuitry if implemented.

Line 64, AUX PWR.
An auxilary +5 volt supply for the use of backup devices. Absolute maximum current when the main power supplies are off is 100mA. Implementation is optional.

Lines 65 66, NDEF1 NDEF2.
Not to be defined. These are lines for users to allocate as they require, there are only two restrictions and one provision.

a)TTL levels only, ie no voltage greater than 5 volts and no voltage less than 0 volts.

b) No transition until 100nS after the previous transistion, ie don't put a 16MHz clock on this line.

c) A link must be provided to disable the use of NDEF 1,2.

Line 67, GND.
An additional ground line to separate the power lines from the rest of the bus.

Lines 68,69, -5 volt supply.

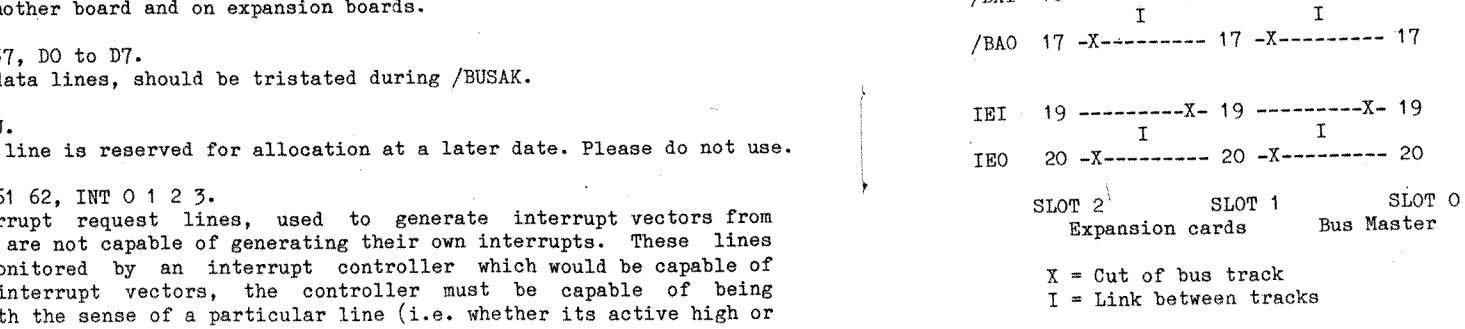
Lines 70,71, -12 volt supply.

Line 72, Keyway

Lines 73,74, +12 volt supply.

Lines 75,76,77,78, +5 volt supply.

Daisy chains
---




X = Cut of bus track
I = Link between tracks