

Standard Interfacing to the S - 100 Bus

There are two major classes of interface connected to the Imsai/ Altair / Sol ... S - 100 bus. These are :

- 1) Devices that are ' Memory Mapped '
- 2) Devices that appear as ' Input/Output Ports

The following is a description of the signal lines used for these interfaces and their possible circuit implementation.

I) Input ports

There are 256 potential input ports requiring eight address lines. The bus sequence generated by the Cpu is as follows :

- 1) Place the address of the input port on address lines A0 - A7.
- 2) Assert Sinp , signifying this address is an Input Port.
- 3) Assert Pdbin
This signal informs the addressed Input Port to place its data on the Data Input lines $\text{DIO} - \text{DI7}$, for the duration of PDBIN .

II) Output ports

There are 256 potential Output ports addressed by lines A0 - A7. The following Cpu bus sequence is generated:

- 1) Place address of output port on lines A0 - A7.
- 2) Assert SOUT

SOUT is a status signal identifying A0 - A7 address as that of an Output Port.

- 3) Place data on data bus $\text{D00} - \text{D07}$

- 4) Assert $\overline{\text{PWR}}$ (low)

$\overline{\text{PWR}}$ (processor write) is a write strobe pulse signifying data lines $\text{D00} - \text{D07}$ contain valid data for the duration of this pulse.

Interfacing of I/O ports usually involves decoding of Address lines A0 - A7 (by jumpers to inverters and 8 input And gates, Exclusive or gates...) And signals coincidence of Sinp and PDBIN , or Sout and $\overline{\text{PWR}}$

Standard Interfacing to the S100 Bus (continued)

Memory Mapped Input / Output

Memory mapping allows for up to 65K addresses to be addressed for input output operations (Read/Write). Supporting this greater number of addresses, requires extra decoding of the top eight address lines A08 - A15. In addition it is 'Good practice' to also decode SOUT, and SINP (both unasserted) for proper operation. Some systems use these signals during DMA, ~~and other~~ Interrupt sequences, or other arbitrary weirdnesses; so decoding of these two lines is an advised precaution.

III) Memory Write (Output to Memory)

The Cpu bus sequence is as follows :

- 1) Place the address of the memory location to be written into, on address lines A0 - A15.

- 2) Assert MWrite

This signal identifies a memory write operation.

- 3) Place data on Output data lines D00 - D07.

- 4) PWR is asserted (low) informing the addressed ~~data~~ device, data on lines D00 - D07 is valid and may be taken in for the duration of this pulse.

IV) Memory Input (Read from Memory)

The Cpu bus sequence is as follows:

- 1) Place the address of the location to be read from, on address lines A0 - A15.

- 2) Assert SMEMR, identifying this as a memory read operation.

- 3) Assert PDBIN

This signal informs the addressed device to place the requested data on data input bus lines DIO - DI7

If slow memories or input output circuits are used (slower than the fastest bus cycle) 'Wait States' must be inserted by the slower device. The wait state holds the current bus signals ~~in~~ until the device 'catches up' To request a Wait state the PRDY line is brought low by the slower device. Timing involving PSync, PWait and and the Theta two (ϕ_2) clock is used (see later description of Addition of Wait States)

Requirements for a Flexible Audio/Video/Control System

To meet the requirement of flexibility , important functions should be maleable (programmable) . It is taken as truth, that ~~any~~ 'userable' digitally controlled system, will need the following structures.

1) A Coordinating Microcomputer

This is a programmable, small computer, to be used for coordination and overall operation of the system : Activating devices , accepting user input as character strings, knobs, etc. to determine parameters for 'system orientation'.

2) A Flexible Bus Structure

Used for control of information and control between devices. The majority of these transfers will be to determine characteristic 'Element' operation.

3) An Input/Output Connection Scheme

This refers to the method of interconnecting final output or input from/to the controlled Elements'. These signals may be analog in nature.

Note that the Control signals are intentionally limited to digital signals. This is important to ensure reliable operation in a bus structured environment. Analog to digital conversion(A/D) or digital to analog conversion may be used inside the 'Elements'to modify operation, from outside signals.

Intelligent Bus Controller/Arbitrator Signal Paths

The Bus Controller is responsible for joining signal paths from separate buses. It also can decide what/when/and where data will be transferred. Once programmed it will time data transfers and coordinate signals across buses. Different programming can change the operation of these transfers(see Bus Controller Modes).

Buses:

- 1)Minicomputer bus
- 2)S-100 bus
- 3)2nd Port bus

A) Connecting the 16 bit Minicomputer bus directly to S-100

This allows the minicomputer complete control of the S-100 bus in either eight bit or sixteen bit mode. It allows for 'loading' Elements with control programs without having to funnel through time buffers. Device/Elements are activated after they are 'programmed' allowing dense memories to be setup in one video field.

B) Joining 16 bit Minicomputer Bus to 2nd Port bus

This connection allows the S-100 Element Bus to run independently of the Port bus. The only time conflicts occur when simultaneous access to the same device is attempted by both buses. The controller tests and arbitrates this condition.

... Bus Controller Signal Paths (continued)

c) S-100 to 2nd Port bus

This is probably the least used of all possible connections. The reason is this ties up both buses for a single transfer. It might be a way to simultaneously read and write between the two buses possibly saving time on fast transfers. This connection still needs exploration.

Some Economic Considerations

The S-100 16/8 bit bus, has numerous capabilities. Some flexibility may be sacrificed, for use in smaller lower cost systems. Later additions may be accomplished with no major physical or conceptual changes.

Buses :

Three buses are used in the full system. These are :

- 1) Sixteen (16) bit micro/mini-computer bus
- 2) Second Port bus
- 3) S-100 16/8 bit bus

Programmable Controllers :

Three programmable controllers are used in the fully implemented system. Their tasks are shared, with their divisions determined by the system's use

- 1) A Sixteen(16) bit mini/micro-computer.
- 2) An 'intelligent' bus controller/arbitrator
- 3) An eight (8) bit microprocessor

By juggling around combinations of these components, powerful economy systems may be devised.

Economic Considerations (continued)

1) No mini/micro-computer (sixteen bit) used.

Since this is the most expensive part of most systems, this could be a large savings. Elimination of the mini-computer also removes the need for it's interface.

In this case the eight bit microprocessor would take over the function of being the major controller. The microprocessor now is responsible for coordinating the entire system, and human interaction. The microprocessor could be placed on a seperate bus , like the minicomputer, but the distinction between mini or micro computer is crossed. The major change is in the number of bits (16 Or 8 Etc.) The single microprocessor system is an excellent starting point. Later as more complex tasks are defined, a sixteen bit mini/micro computer may be added.

2) No second Port bus

This removes the need for extra circuitry on the element cards to accomodate this bus. This would place all changes in control functions, with the S-100 bus. This limits time critical application (on line video) . / Carefully thought out timing changes could somewhat get over this limitation.

Economic Considerations (continued)

3) No Bus Controller/Arbitrator

If no Minicomputer(16 bit) is used, and the 2nd Port is unnecessary, all control will reside on the S-100 bus. A microprocessor is necessary on the S-100 bus. The functions of the Bus Controller would now be handled by Software. The Microprocessor executes this software to allow transfers of information to elements from/to Buffer Memory. The S-100 bus is kept in eight bit mode(the standard S-100 bus). This arrangement would limit the speed of operation, but is the cheapest in ' hardware' to implement.

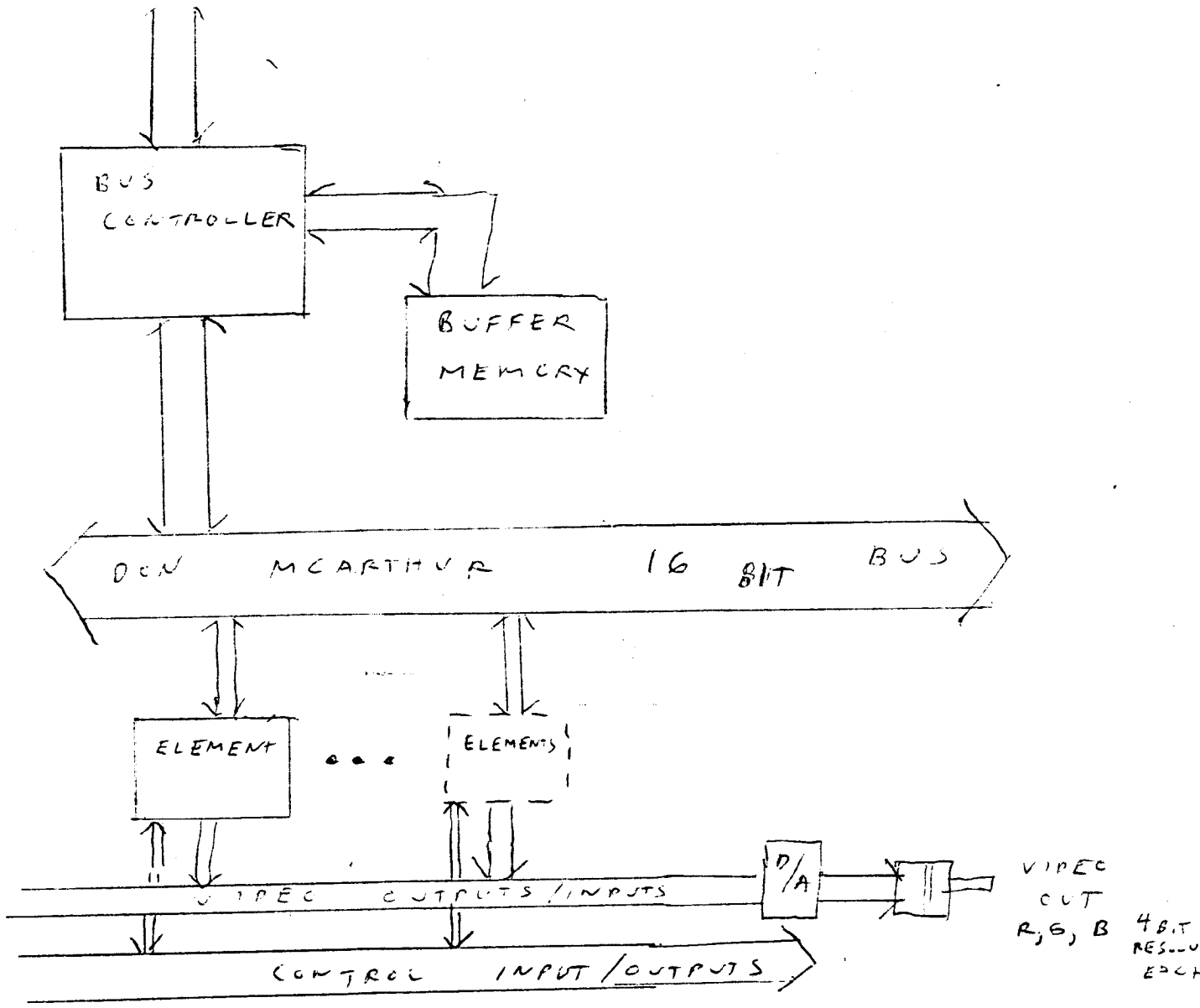
4) No Eight bit Microprocessor

This would place all control functions with the 'Intelligent Bus Controller/Arbitrator and the Minicomputer(16 bit). Omission of the microprocessor is of questionable value, since it the lowest cost device in the system. But many functions involving transfer of data can easily be handled by the Bus Controller.

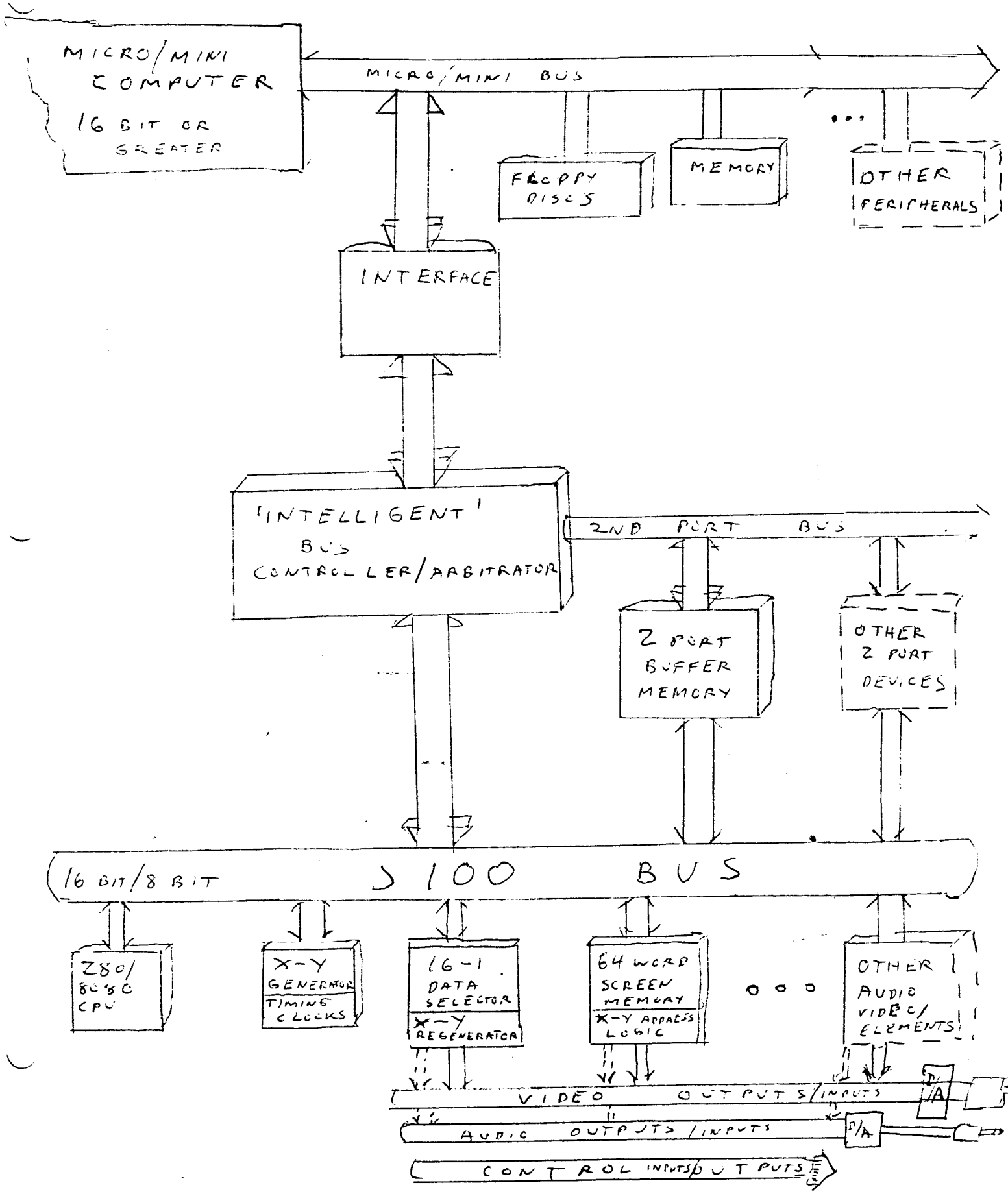
Caution should be exercised in appearant savings found by ommitting hardware. Unless a roaming software expert is known, software is the most time consuming and expensive part of any reasonably complex system. The hardware -Software tradeoffs should be considered before saving on parts.

PRESENT DON MCARTHUR VIDEQ / CONTROL BUS

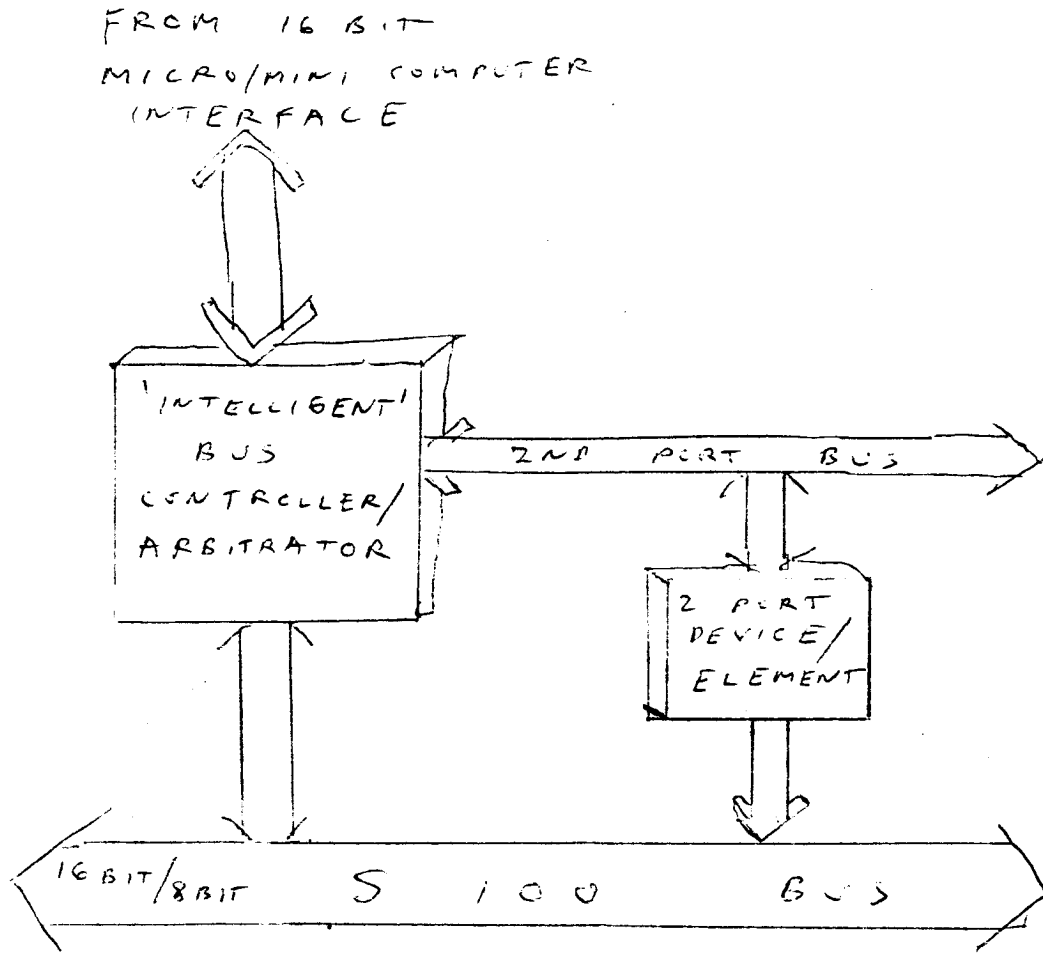
FROM
LSI-11
16 BIT COMPUTER
INTERFACE



PROPOSED 16 BIT / 8 BIT S100 AUDIO / VIDEO / CONTROL BUS



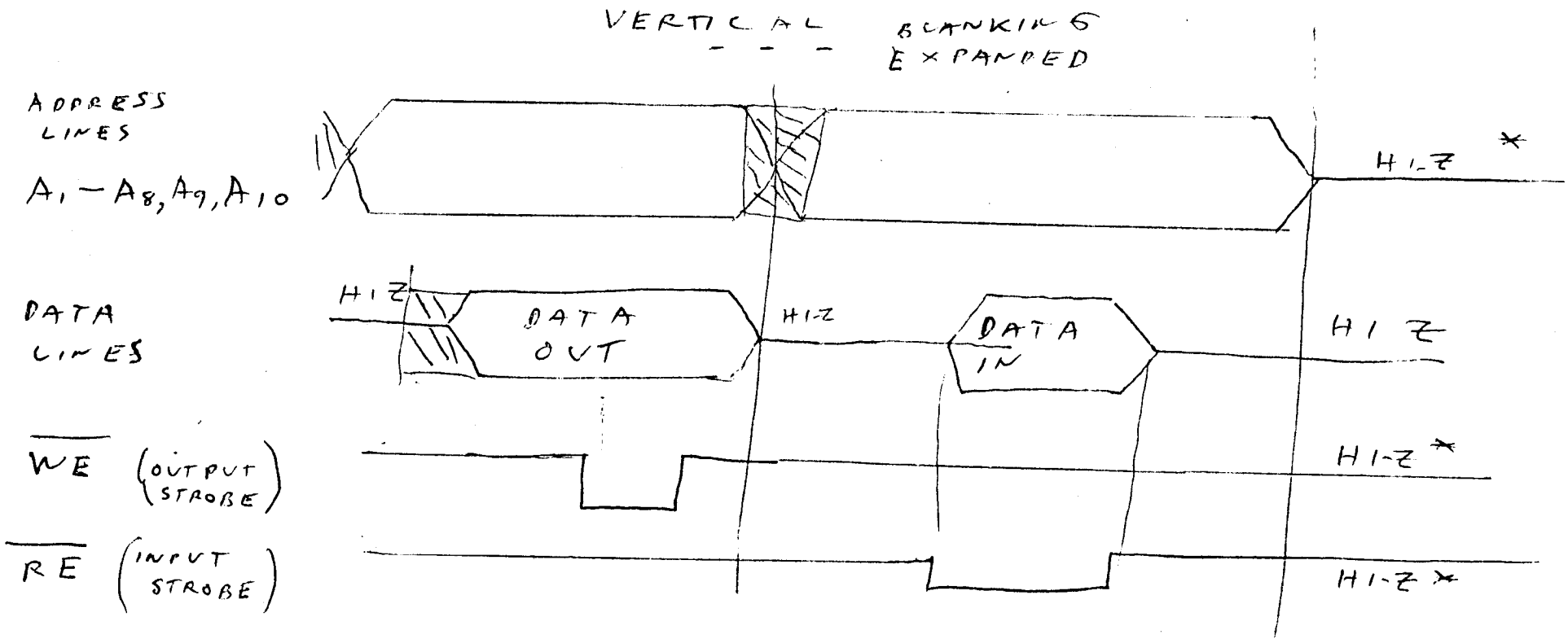
INTELLIGENT BUS CONTROLLER SIGNAL PATHS



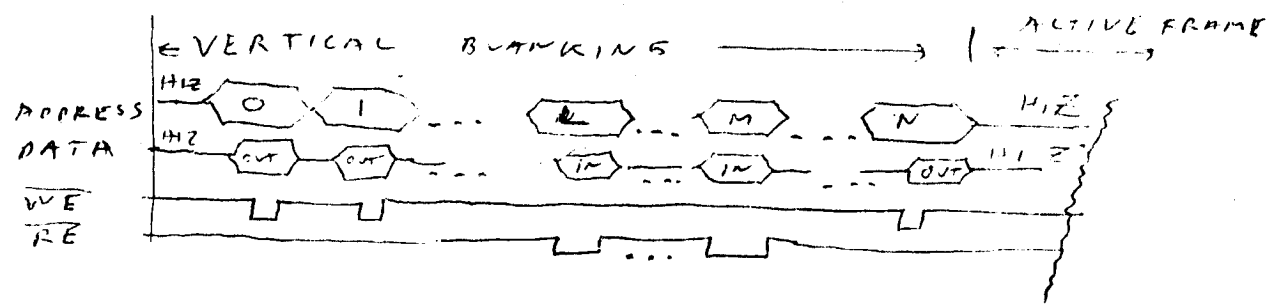
POSSIBLE SIGNAL PATHS

- 1) 16 BIT INTERFACE \longleftrightarrow S100 BUS
- 2) 16 BIT INTERFACE \longleftrightarrow 2ND PORT BUS
- 3) S100 \longleftrightarrow 2ND PORT BUS

DON 'BUS SIGNAL DESCRIPTION



SHADED REGIONS REPRESENT 'DON'T CARE' CONDITIONS



BUFFER MEMORY

$n = 256$ IN 256 WORD SYSTEM

$n = 1024$ IN 1024 WORD SYSTEM

L - M IS THE FEATURE AREA

* NOTE: PRESENT DON 'BUS DOES NOT TRISTATE THESE SIGNALS

Pinouts for Don Bus on the S100 bus

Name Pin Number

A0 79
A1..... 80
A2..... 81
A331
A4..... 30
A5..... 29
A6 82
A7 83
A8..... 84
A9 34
A10 37
A11 87
A1233
A13 85
A14..... 86
A15..... 32

Address lines : Notice the bizarre numbering on these pins. In the 256 word Don bus pins A1 - A8 are used. In the 1K version (That's you Binghamton) lines A1 - A10 are used

~~XXXXXXXXXX~~

D00 (DI00)..36
D01 (DI01)..35
D02 (DI02)..88
D03 (DI03)..89
D04 (DI04)..38
D05 (DI05)..39
D06 (DI06)..40
D07 (DI07)..90
DI08 56
DI09 57
DI010..... 58
DI011..... 59
DI012..... 60
DI013 61
DI014 62
DI015 63
Byte 64

Data lines : note again the enjoyable scrambled numbers. The numbers on the left are S100 designations when available. Parenthesized names are for Don Bus. Note also lines on pins 56- 67 are blank (unused on the standard S100 bus.)

Pinouts for Don Bus on the S100 bus

| <u>Name</u> | <u>Pin number</u> | | |
|--------------------------|-------------------------|--|-------------------|
| Xclock | 12 | (9.7 Mhz clock) | **pins 12 to 17 |
| Xload | 13 | | are unused on the |
| Yclock | 14 | (2 x H clock) | standard S100 bus |
| Yload | 15 | | |
| Hdrive | 16 | (Horizontal drive ,TTL) | |
| Vdrive | 17 | (Vertical drive , TTL) | |
| S _{ubcar} | 66 | (Subcarrier , TTL) | |
| \overline{PWR} | (\overline{WE}) ..77 | | |
| Pdbin ,. | (\overline{RE}) .. 78 | Notice the current level of \overline{RE} is the | |
| + 8 Volts | 1,51 | reverse of PDBIN | |
| + 16 Volts .. | 2 | | |
| - 16Volts .. | 52 | | |
| Ground | 50, 100 | | |