EDUC–8ME: Memory and Input/Output Port Extension for the EDUC–8 Microcomputer

by
Steven S. Pietrobon, Ph.D.
29 June 2024.

Abstract

To reduce cost and complexity, the original Electronics Australia EDUC–8 microcomputer from 1974 was limited to 256 bytes of static RAM and two serial input and two serial output ports (two I/O ports). To increase its usability, Jamieson Rowe in his EDUC–8 book gave a schematic that added 16 I/O ports up to a maximum of 128 I/O ports. To increase memory, 256 bytes could be added to each I/O port extension. Data could be stored in this memory or copied to internal RAM for execution, but this would be very slow. We present a port extension similar to Rowe's scheme, but a memory extension similar to that used in the Digital PDP–8, which EDUC–8 is based on. We name this extension EDUC–8ME (pronounced "Educate Me").

Introduction

The EDUC–8ME requires changing the EDUC–8 motherboard (to E8L/C, which also includes modifications to include a link bit) and adding a new horizontal board (E8ME/B) to the bottom of E8L/C. This new board buffers various signals from EDUC–8 to up to eight EDUC–8ME units. Each EDUC–8ME adds eight I/O output ports and 8KB of static random access memories (SRAM), using 64 1Kx1 SRAMs. With eight EDUC–8ME units this allows an expansion of up to 64 I/O ports and 64KB of RAM. Alternatively, 64KB can achieved wih two EDUC–8ME units using 4Kx1 SRAMs or one unit using 32 16Kx1 SRAMs .

EDUC–8ME consists of five different boards. The front panel board is the same as EDUC–8 (E8/F) and is used only on the first EDUC–8ME unit. The other boards are the motherboard (E8ME/C), decoder (E8ME/D), I/O port multiplexer (E8ME/I) and memory (E8ME/M). The decoder board (used only in the first EDUC–8ME unit) provides the interface from EDUC–8 to all the EDUC–8ME units. The multiplexer provides the interface to the eight I/O ports and the memory. Up to four memory boards can be used, each containing up to 16 SRAM chips. The memory board is designed to accept either the Intel 2102 or 2125 1Kx1, 2147 4Kx1 or 2167 16Kx1 SRAMs, even though they all have different pinouts.

The main design philosophy is to use technology available in 1974, that is 74 series TTL. If desired, more modern technology such as 74LS series TTL can be used. 1Kx1 memories available in 1974 include the Intel 2102, Signetics 2602B and Mostek MK4102P which all have the same pinout. These RAMs use NMOS technology and were recommended by Jamieson Rowe in his EDUC–8 book. If desired, more modern SRAMs can be used, such as the 2125 from 1976, 2147 from 1977 and 2167 from 1981. Note that the 2125 is pin equivalent to the Fairchild bipolar 93425 (the tristate version of the open collector 93415 or 2115, as used by EDUC–8) and Harris CMOS 6508 from 1976.

The memory expansion method used is similar to that in the Digital PDP–8. Each field of memory addresses the full 256 bytes available to EDUC–8. Each EDUC–8ME has up to 32 fields for 8 KB of RAM using 1Kx1 SRAMs (128 fields with 4Kx1 and the full 256 fields with 16Kx1). Two 8–bit registers are used to control access to which field is being used. These are the Instruction Field Address (IFA) and Data Field Address (DFA) registers. To prevent wild address jumps, the IFA is only updated during JMP or JMS instructions and is used to select which field of memory is being used to access instructions and direct data. The DFA is only used during indirect AND, TAD, ISZ and DCA instructions to select the indirect field data. A multiplexer is used to select either the IDA or IFA to produce the field address (FA), which is used to select a field. The lower 11 bits to the RAMs is used to serially access one byte of the 256 bytes in a field. These address bits are provided from E8/M to E8ME/B to E8ME/D to E8ME/I to E8ME/M.

The I/O expansion uses two 6–bit registers, the input data address (IDA) and output data address (ODA). The lower three bits are used to select one of the eight input or output ports. To reduce complexity, the IDA and ODA are also multiplexed to FA. A three bit DIP switch is used to compare the EDUC–8ME unit number (0 to 7) with the top three bits of FA. If the values are the same then the appropriate memory, input port or output port is enabled.

Similar to that proposed by Rowe, one EDUC–8 input port (ID1) and one output port (OD1) is used for access to the EDUC–8ME input and output ports. Note that Rowe used ID0 for the input port which used a custom 16–bit interface for access to a paper tape reader. However, we use ID1 instead of ID0, so that the ports have the same number. This also allows access to the paper tape reader for original EDUC–8 machines. In order to program the IDA or ODA, OD0 was used with an LDB instruction (load output buffer, reset flag) and a 74164 address buffer (AB) serial to parallel converter. The most significant bit of AB was used to select writing the data to IDA or ODA during the flag reset operation.

In our case, we have four registers we need to program (IFA, DFA, IDA, ODA). Using Rowe's scheme, we could use the top two bits to select the register, but this would limit us to 64×256 = 16KB of RAM. In order to implement subroutines with different I/O ports and to return from subroutines in a different field, it would also be convenient to also read these four registers. We could use ID0 for this, but this now leaves us with no EDUC–8 ports for other purposes (such as a keyboard or paper tape reader).

A solution to this problem is noting that for an IOT instruction, the least three significant bits are used for the instruction type, but there are only four valid instructions: 6x1, 6x2, 6x4 and 6x6, where x is a two bit value for the port. We call x the memory expansion address (MEA). We propose using 6x3 and 6x7 to read and write to the four registers, respectively, where x is used to select the register (0 for IFA, 1 for IDA, 2 for DFA and 3 for ODA). We selected 1 for IDA and 3 for DFA as these are the same values for ID1 and OD1. This scheme frees up ID0 and OD0 to be used for other purposes and increases the amount of available memory to 64KB. Theoretically, we could have up to 256 I/O ports, but due to space limitations at the back of each EDUC–8ME unit, this is limited to a total of 64 I/O ports.

Buffer Board (E8ME/B)

E8/IOT Modification

In order to use 6x3 and 6x7 instructions, we need to disable E8/IOT when these instructions are used. This is done using the new E8L/C board (described in a separate article [1]) and E8ME/B board, as shown in schematic E8ME/B1. Three signals are used to control the IOT board, $\overline{\text{CLR\_IOT\_FLAG}}$, $\overline{\text{IOT\_SHIFT}}$ and SKP_ON_IOT_FLAG from the E8/D decoder board. We simplify these signal names to $\overline{\text{CLR}}$, $\overline{\text{SHF}}$ and SKP, respectively. These three signals are sent from E8/D to E8ME/B through some logic and then to E8/IOT as $\overline{\text{CLRM}}$, $\overline{\text{SHFM}}$ and SKPM.

For the 6x3 and 6x7 instructions, the least significant bit is always one, implying that the $\overline{\text{T14–21.SKP}}$ signal from E8/D will be active. However, since SKPM stays low, the skip instruction is never implemented as $\overline{\text{IOT\_SKP}}$ from E8/IOT remains high. The second least significant bit is also one, implying that $\overline{\text{IOT\_SHIFT}}$ from E8/D will be active, which is desired since all 6x3 and 6x7 instructions require the shift operation. However, the $\overline{\text{SHFM}}$ input stays high to prevent the E8/IOT board from shifting data. For the 6x7 instruction, the $\overline{\text{CLR\_IOT\_FLAG}}$ signal from E8/D will be active, however $\overline{\text{CLRM}}$ is forced high, to prevent the E8/IOT board from resetting an I/O port.

The Karnaugh maps for the signals are below, where we have used positive logic. Inputs 0 (needed for non–IOT instructions), 1, 2, 4 and 6 should remain the same, while inputs 3 and 7 should be mapped to 0. Input 5 is left undefined as it is not a valid instruction. Thus we have $\overline{\text{CLRM}} = \overline{\text{CLR}.\overline{\text{SKP}}}$, $\overline{\text{SHFM}} = \overline{\text{SHF}.\overline{\text{SKP}}}$ and SKPM = $\overline{\text{SHF}}$.SKP.

CLR \ SHF,SKP	00	01	11	10
0	0	0	0	0
1	1		0	1

CLRM = CLR.$\overline{\text{SKP}}$

CLR \ SHF,SKP	00	01	11	10
0	0	0	0	1
1	0		0	1

SHFM = SHF.$\overline{\text{SKP}}$

CLR \ SHF,SKP	00	01	11	10
0	0	1	0	0
1	0		0	0

SKPM = $\overline{\text{SHF}}$.SKP

/IFAE *and* /SDF *Signal Generation*

After the IFA has been written to the AB, we have to wait until a following JMP or JMS instruction to write the output of AB into the IFA register at the start of EXECUTE. This is performed using set reset flip flop (SRFF) Q to generate the instruction field address enable (/IFAE) signal. In general, an SRFF can be implemented with two NAND gates, each with two or more inputs. For two input NAND gates we have

$$\text{Q} = \overline{\overline{\text{S}}./\text{Q}}$$
$$/\text{Q} = \overline{\overline{\text{R}}.\text{Q}}$$

where $\overline{\text{S}}$ is the active low set input (which sets Q high and /Q low), $\overline{\text{R}}$ is the active low reset input (which sets /Q high and Q low), with /Q being the inverse of Q. With more than two inputs to each gate, additional set or reset signals can be used.

We first set Q using the $\overline{\text{SQF}}$ signal when the instruction 607 (IFA write) is used during T13 of EXECUTE. That is $\overline{\text{SQF}} = \overline{\overline{\text{MB4}}.\overline{\text{MB3}}.\text{CLR.SHF.SKP.T13}}$, as shown in schematic E8ME/B2. As CLR = IOT.MB2, SHF = IOT.MB1 and SKP = IOT.MB0, where IOT = EXECUTE.IB7.IB6.$\overline{\text{IB5}}$, where IB7, IB6 and IB5 are the instruction decoder register values for MB7, MB6 and MB5, respectively, the condition is satisfied.

If the Q register is set, then during a JMP or JMS instruction, the IFA register is loaded at the start of EXECUTE. That is /IFAE = $\overline{\text{Q.(JMS+JMP).T0·5}}$, as shown in schematic E8ME/B1. We have JMS+JMP = $\overline{\overline{\text{JMS}}.\overline{\text{JMP}}}$, where $\overline{\text{JMS}}$ and $\overline{\text{JMP}}$ are obtained from E8/D. As JMS = EXECUTE.IB7.$\overline{\text{IB6}}$.$\overline{\text{IB5}}$ and JMP = EXECUTE.IB7.$\overline{\text{IB6}}$.IB5, the condition is satisfied. A 7437 buffer is used to distribute /IFAE to the first EDUC–8ME unit. At the end of EXECUTE for the JMS or JMP instruction, we need to reset Q using the signal $\overline{\text{(JMS+JMP).T22·5}}$. We also need to reset Q at power up using the $\overline{\text{RST}}$ signal. That is we have

$$\text{Q} = \overline{\overline{\overline{\text{MB4}}.\overline{\text{MB3}}.\text{CLR.SHF.SKP.T13}}./\text{Q}}$$
$$/\text{Q} = \overline{\overline{(\text{JMS}+\text{JMP}).\text{T22·5}}.\overline{\text{RST}}.\text{Q}}$$
$$/\text{IFAE} = \overline{\text{Q.(JMS+JMP).T0·5}}$$

For the two field address registers, IFA and DFA, we need to generate a signal to select one of the addresses that is used to select the appropriate memory field. The DFA is selected during EXECUTE when performing an indirect AND (0xx), TAD (1xx), ISZ (2xx) or DCA (3xx) instruction. The indirect bit MB4 for the instruction needs to be available during EXECUTE. However, as MB4 to MB7 are overwritten during T23 of FETCH with the four bit page address, this information is lost. Thus, we use an SRFF P, to store this information before T23 of FETCH, but after T14–21 when the instruction is written to MB.

That is, we want to set P during FETCH and T22·5 and if MB7 is low (for an AND, TAD, ISZ or DCA instruction) and if MB4 is high (for an indirect instruction). Note that for an IOT instruction, MB3 and MB4 are not overwritten and thus can be used for /IFAE. That is

$$\mathrm{P} = \overline{\overline{\mathrm{FETCH.T22{\cdot}5.}\overline{\mathrm{MB7}}\mathrm{.MB4}}\mathrm{./P}}$$

At the end of EXECUTE we want to reset P. We also want to reset P on power up. Thus

$$\mathrm{/P} = \overline{\overline{\mathrm{EXECUTE.T22{\cdot}5}}.\overline{\mathrm{RST}}\mathrm{.P}}$$

We thus generate the select data field (/SDF) active low signal using a 7437 buffer during EXECUTE and if P is high. That is

$$\mathrm{/SDF} = \overline{\mathrm{EXECUTE.P}}$$

The logic for generating /IFAE and /SDF is shown in schematic E8ME/B1. Signals MB3, MB4 and MB7 are obtained from the E8/M memory board. Signals $\overline{\mathrm{T0{\cdot}5}}$, T22·5, T13, FETCH, EXECUTE and $\overline{\mathrm{RST}}$ are obtained from the E8/T timing board.

Note that E8/T needs to be modified so that $\overline{\mathrm{RST}}$ is connected to unused pin 25 of the edge connector. Unfortunately, the 7420 4–input NAND gate used for the output of $\overline{\mathrm{RST}}$ is already fully loaded (in fact, it is overloaded!). It drives the $\overline{\mathrm{MR}}$ input of the 9316/74161 timing counter with 1 UL (unit load, 40 μA high and 1.6 mA low), reset inputs of the defer and run flags (two 7400 with 1 UL each) and four $\overline{\mathrm{C_D}}$ inputs of two 7473 dual JK FFs with 2 UL for each $\overline{\mathrm{C_D}}$ input. This gives a total of 11 UL with the 7420 able to drive 20 UL high and 10 UL low (20/10 UL). Thus, adding an additional 2 UL for E8ME/B will further overload the 7420. To fix this problem, the 7420 needs to be replaced with a 7440 dual 4–input NAND buffer which can drive 30 UL and is pin compatible with the 7420.

For 74LS, the load is 0.5/0.25 UL for the 74LS161, 4×1.5/0.5 UL for the two 74LS73, 2×0.5/0.25 UL for the 74LS00 and 2×0.5/0.25 UL for the 74LS10 used in E8ME/B. This gives a total load of 8.5/3.25 UL. The 74LS20 can however drive 10/5 UL and thus does not need to be changed.

EXECUTE is also fully loaded, with 10 UL in E8/D (eight 7400 and two 7410 inputs). The driver is a 7404 hex inverter on E8/T. E8ME/B requires an additional 2 UL (7400 and 7437 inputs). Unfortunately, there are no buffer circuits equivalent to the 7404. However, a 74H04 or 74S04 can be used (both available in 1974), which can drive 12.5 UL. The 7404 that needs to be replaced on E8/T is closest to the edge connector side. A 74LS04 can drive 10/5 UL with the total load being 12×0.5/0.25 = 6/3 UL, so does not need to be changed.

MEA Input and Output Port Signals

Similar to how the E8/IOT board has $\overline{\mathrm{DATA}}$ for output, DATA for input, $\overline{\mathrm{CLOCK}}$, $\overline{\mathrm{FLAG\_RESET}}$ and $\overline{\mathrm{FLAG}}$ signals, we generate two custom ports to read and write the field address to the four ME registers. As we don't need to read a flag, the $\overline{\mathrm{FLAG}}$ signal is omitted. However, we do need $\overline{\mathrm{FLAG\_RESET}}$ for both writing and reading, equivalent to LDB (as used by Rowe for his IOT expansion) and KRB instructions. We use T13 for a write, so that data is transferred from AB to one of the IDA, ODA or DFA registers. /IFAE is used to transfer AB to IFA. For a read, T1 is used so that data is transferred from one of the four MEA registers to a 74165 parallel to serial converter, so that data can be shifted during T2–9.

The output ports are enabled if SHF and SKP are high. Signal CLR is used to select the output port and $\overline{\mathrm{CLR}}$ the input port. Data is output during T2–9 from the AC_BIT_0 signal. For the output port we have

$$\overline{\mathrm{ME\_OUT\_DATA}} = \overline{\mathrm{CLR.AC\_BIT\_0.T2{-}9.SHF.SKP}}$$
$$\overline{\mathrm{ME\_OUT\_CLOCK}} = \overline{\mathrm{CLR.MCPB.T2{-}9.SHF.SKP}}$$
$$\overline{\mathrm{ME\_OUT\_RESET}} = \overline{\mathrm{CLR.T13.SHF.SKP}}$$

For the input port data is input non–inverted, but is transmitted inverted to the A–BUS using a 7401 open collector gate. We have for the input port

$\text{A–BUS} = \overline{\overline{\text{CLR}}\text{.ME\_IN\_DATA.T2–9.SHF.SKP}}$
$\overline{\text{ME\_IN\_CLOCK}} = \overline{\overline{\text{CLR}}\text{.MCPB.T2–9.SHF.SKP}}$
$\overline{\text{ME\_IN\_RESET}} = \overline{\overline{\text{CLR}}\text{.T1.SHF.SKP}}$

All outputs use 7437 buffers. The AC_BIT_0, T1, T2–9, T13, MCPB and A–BUS signals can be obtained from the connector used for E8/IOT. To reduce the amount of logic and minimise power consumption, 7408 two input and 7411 three input AND gates are used. The logic is shown in the E8ME/B2 schematic.

The MB3 and MB4 signals are buffered using 7437 gates to $\overline{\text{MB3O}}$ and $\overline{\text{MB4O}}$, respectively, in order to select one of the four registers during a 6x3 or 6x7 instruction. For a memory reference instruction during EXECUTE, we force $\overline{\text{MB3O}}$ and $\overline{\text{MB4O}}$ high using the OPR_IOT signal. This is equivalent to selecting the IFA register. The logic for this is shown in the E8ME/B1 schematic.

Below is a table showing all the loads for all the signals from EDUC–8. For the board columns the first number is the pin used on the board, followed by the number of ULs.

Signal Name	E8/T	E8/D	E8/M	E8/P	E8/A	E8/IOT	E8/F	E8ME/B	Total
$\overline{\text{T0·5}}$	8–0	8–1	7–3					19–1	5
T22·5	3–0	3–1						18–3	4
FETCH	21–1	19–2						21–1	4
EXECUTE	23–0	21–10						22–2	12
$\overline{\text{RST}}$	25–11							24–2	13
T1	16–0		12–1	14–1	13–2	15–1		15–1	6
T2–9	15–0	12–3	11–1	13–2	12–1	14–1		14–1	9
T13	14–0		10–1	12–3		13–1		13–1	6
MCPB	11–1			11–2	9–1	10–1		10–1	6
AC_BIT_0/AC0					11/J–7	12–1	X–1	12–1	10
$\overline{\text{CLR\_IOT\_FLAG}}$		6–0				7–0		7–4	4
$\overline{\text{IOT\_SHIFT}}$		11–0			10–1	11–0		11–2	3
SKP_ON_IOT_FLAG		7–0				8–0		8–5	5
$\overline{\text{JMS}}$		E–2		D–2			X–1	L–1	6
$\overline{\text{JMP}}$		F–1					X–1	M–1	3
OPR_IOT	20–1	16–0	16–1					20–2	4
$\overline{\text{MB3}}$		4–0				5–2		5–1	3
$\overline{\text{MB4}}$		5–1				6–2		6–1	4
MEMORY_ENABLE		23–0	23–4					23–2	6
MB3		M–3	M–3				X–1	N–1	8
MB4	31–1	N–2	N–3				X–1	P–2	9
MB7		R–1	R–3				X–1	R–1	6

For $\overline{\text{CLR\_IOT\_FLAG}}$, $\overline{\text{IOT\_SHIFT}}$ and SKP_ON_IOT_FLAG these signals are diverted to E8ME/B and thus have zero loads for E8/IOT. For MB3, MB4 and MB7 we have included one additional load each for the Page Zero circuit in E8/M where MB3 goes to one input of a 7405 dual input NAND with the other input tied to VCC via a resistor. As can be seen, only EXECUTE and $\overline{\text{RST}}$ need to increase their output load from the standard 10 ULs. All the other signals satisfy the requirement of 10 or less ULs.

$\overline{\text{WE}}$ Signal Generation

For the Intel 2102, data must be held for 100 ns after the write enable $\overline{\text{WE}}$ goes high. This is a problem since $\overline{\text{WE}}$ = $\overline{\text{MCPA.(T2–9.(JMS+DCA)+T14–21.(ISZ+DEP))}}$ and the RAM data DIN = $\overline{\text{C–BUS}}$ where

$\overline{\text{C–BUS}}$ = D–BUS.$\overline{\text{T2–9.ISZ}}$.MEMORY_ENABLE.$\overline{\text{(T2–9.(JMS+DCA)+T14–21.(ISZ+DEP))}}$
+ T14–21.(JMS+JMP).MA0 + T14–21.(ISZ+DEP).MB0 + DCA.AC0 + T2–9.ISZ.SUM
+ (T2–9.JMS+T2–9.(FETCH+DEP+EXM)).PC0

where

D–BUS = DOUT0.DOUT1
MEMORY_ENABLE = T2–9.(JMS+DCA+TAD+AND+ISZ) +
T14–21.(ISZ+FETCH+DEP+EXM+DEFER)

and DEP is the deposit cycle, EXM is the examination cycle and DOUT0 and DOUT1 are the RAM open collector data outputs, where disabled outputs are always high. The first term of $\overline{\text{C–BUS}}$ outputs data from the RAM during MEMORY_ENABLE, except during a write operation or T2–9.ISZ, where the serial adder SUM is output. For JMS, PC0 is written to the RAM, which changes on the falling edge of MCPB. For DCA, AC0 is written to the RAM, which changes on the falling edge of T1 for a RAL instruction (when data is not written) or the falling edge of MCPB. For ISZ or DEP, MB0 is written to the RAM, which changes on the falling edge of MCPA. Both MCPA and MCPB are driven from the Main Gate and are approximately coincidental.

We have that the rising edge of $\overline{\text{WE}}$ corresponds to the falling edge of MCPA, thus data can change soon after the rising edge of $\overline{\text{WE}}$, possibly violating the 100 ns hold time. As 74 series data sheets do not provide minimum delay times, we scale low to high (LH) delay times by 2.4/5 = 0.48 where the minimum and maximum delay times for a 74F00 gate are 2.4 and 5 ns, respectively. Similarly, we scale high to low (HL) delay times by 1.5/4.3 = 0.35.

The minimum delay from MCPB to the clock input for PC0 is 22×0.48 (7400 LH) = 10.56 ns. We then have the following LH and HL delays:

40×0.48 (7496 LH) + 15×0.35 (7401 HL) + 22×0.48 (7400 LH) = 35.01 ns
40×0.35 (7496 HL) + 45×0.48 (7401 LH) + 15×0.35 (7400 HL) = 40.85 ns

This gives a total minimum delay of 10.56+35.01 = 45.57 ns. The maximum delay from MCPA to $\overline{\text{WE}}$ is 22 ns (7400 LH). This gives a minimum effective hold time of 45.57–22 = 23.57 ns, which violates the 100 ns requirement. We have for AC0 and MB0

MCPB/MCPA to 7495: 22×0.48 (7400 LH) + 15×0.35 (7400/7404 HL) = 15.81 ns
7495 to DIN:
27×0.48 (7495 LH) + 15×0.35 (7401 HL) + 22×0.48 (7400 LH) = 28.77 ns
32×0.35 (7495 HL) + 45×0.48 (7401 LH) + 15×0.35 (7400 HL) = 38.05 ns

which gives a minimum delay of 15.81+28.77 = 44.58 ns and an effective hold time of 44.58–22 = 22.58 ns. Thus, the critical path for writing data to DIN is for AC0 or MB0, which has a minimum delay of 44.58 ns. For EDUC–8ME we also need to add the delays for the 7437 buffer in E8ME/B and 7404 driver in E8ME/D. Since the 7437 and 7404 have the same combined LH and HL delays, the delay is 22×0.48 (LH) + 15×0.35 (HL) = 15.81 ns. This gives a minimum delay of T_{hmin} = 44.58+15.81 = 60.39 ns.

The address must also be held for 50 ns after $\overline{WE}$ goes high. The minimum delay is from the falling edge of MCPA to the 7493 strobe counter in E8/M to A0 in the E8M/M memory board. We have

MCPA to 7493: 22×0.48 (7400 LH) + 15×0.35 (7404 HL) = 15.81 ns
7493 to A0 (E8M/M):
16×0.48 (7493 LH) + 15×0.35 (7437 HL) + 22×0.48 (7404 LH) = 23.49 ns
21×0.35 (7493 HL) + 22×0.48 (7437 LH) + 15×0.35 (7404 HL) = 23.16 ns

This gives a minimum delay of 15.81+23.16 = 38.97 ns. Since 60.39–100 = –39.61 ns is less than 38.97–50 = –11.03 ns, the critical path for the hold times is determined by AC0 or MB0 to DIN.

To fix the data hold problem we use a 74122 multivibrator where input $\overline{A_1}$ is connected to $\overline{WE}$ and $\overline{A_2}$, B_1, B_2 and $\overline{C_D}$ are connected to logic high. On the falling edge of $\overline{WE}$ the gate is triggered to generate an active low clock pulse from output $\overline{Q}$. We want the pulse to go back high before the rising edge of MCPB, with a minimum pulse width of T_w = 750 ns, as required by the 2102. We now need to determine the values for the external resistor Rx and capacitor Cx. Using Rx = 10K and the 74122 data sheet, this gives a nominal capacitance of Cx = 180 pF. Assuming Cx has a 10% tolerance, the resistance for Rx is 10/1.1 = 9.1K and 10/0.9 = 11.1K, when Cx is at its maximum and minimum variation, respectively. This range can be achieved using an 8.2K 5% resistor and 5K 10% trimpot. The maximum fixed resistance is 1.05×8.2 = 8.6K, which is less than the 9.1K minimum required value, as required. The minimum maximum resistance is 0.95×8.2+0.9×5 = 12.3K which is greater the 11.1K requirement, as required.

The clock frequency is determined using a 10K (5%) feedback resistor and a 220 pF (10%) capacitor in the E8/T board. For our EDUC–8 unit, we measured a clock frequency of 430 MHz where the MCPA high width was 1147 ns. We measured the resistance to be 9.87K. Assuming the frequency changes linearly with R and C, this gives a minimum width of 1147×(9.5/9.87)× (0.9/1.1) = 903 ns (about 550 kHz).

The maximum delay from the rising edge of MCPA to the falling edge of the $\overline{WE}$ at the memory chips is 15 (7400 HL) + 40 (74122 HL) + 22 (7437 LH) + 15 (7404 HL) = 92 ns. This leaves 903–92–750 = 61 ns data hold time from the rising edge of the buffered $\overline{WE}$ to falling edge of MCPA. Adding the T_{hmin} = 60.39 ns delay time available from the data, this gives 61+60 = 121 ns hold time, satisfying the 100 ns constraint. If the high width of MCPA is T_{ch} and the maximum low width of $\overline{WE}$ is T_{wmax}, then we have T_{ch}–92–T_{wmax}+60 = 100, or T_{wmax} = T_{ch}–132. For example, if T_{ch} = 903 ns, then T_{wmax} = 771 ns.

The circuit for the $\overline{WE}$ pulse generation is shown in schematic E8ME/B2. To adjust the variable resistor, move the jumper so that $\overline{A1}$ is connected to MCPB. The cable to the E8/M board and the first EDUC–8ME unit should be disconnected. Insert a 93415 or equivalent RAM into the lower position of the E8/M board, so that a RAM without a hold time restriction can be used. As MCPB is only active when EDUC–8 is running, load the instruction JMP 0 (500) into address 000 and run the program. By measuring the high width of MCPB to determine T_{ch}, the maximum low width T_{wmax} = T_{ch}–132 in ns can be calculated. With E8ME/B connected to the extender board, adjust the 5K trimpot so that the $\overline{Q}$ output of the 74122 has a low width between 750 ns and T_{wmax}. This allows for any variations due to temperature, voltage and component drift.

If a 74LS122 multivibrator is used, the capacitor Cx = 150 pF.

Memory Signal Buffers

In order to use the external memory in EDUC–8ME, we need to buffer the signals from the first memory chip in E8/M, which are effectively pin locations A0 to A9, $\overline{CS}$, $\overline{WE}$, DIN, DOUT and

GND. As $\overline{\text{CS}}$ is connected to $\overline{\text{MEMORY\_ENABLE.}\overline{\text{MA7}}}$ in the lower memory position, we let A10 = $\overline{\text{CS}}$ = MA7 when MEMORY_ENABLE is high. Note that the upper memory position could also be used where $\overline{\text{CS}}$ = $\overline{\text{MEMORY\_ENABLE.MA7}}$. We also need to access MEMORY_ENABLE from E8/D, in order to generate the chip select signals for the EDUC–8ME memory boards E8ME/M. Pin $\overline{\text{WE}}$ is sent to the 74122 (U14) circuit to generate an active low pulse, as described in the previous section. Note that pin DIN of X4 is actually an output, corresponding to the data to be written to the RAMs. Pin DOUT is connected to D–BUS and is an input, corresponding to the data that is read from the RAMs.

For pins A0 to A9, $\overline{\text{CS}}$, DIN, QB of U14 and MEMORY_ENABLE, we use 7437 buffers to output /A0 to /A10, /DIN, WE and /ME to the EDUC–8ME units as shown in schematics E8ME/B2 and E8ME/B1. For pin DOUT was use a 7401 open collector gate to connect to the D–BUS. The inputs to the gate are /DOUT from the EDUC–8ME units and MEMORY_ENABLE, to ensure the output is active only while the RAMs are being accessed.

A single resistor R1 is used to tie any unused inputs to V_{CC}, in order to reduce power consumption and prevent component failure in case of overvoltage. This includes 15 7437 inputs and $\overline{\text{A2}}$, B1, B2 and $\overline{\text{C}_\text{D}}$ of the 74122. This resistor should range between 1K and 5K, where the maximum resistance is

$$R_{\text{max}} = \frac{V_{\text{CC,min}} - V_{\text{OH}}}{0.04U_{\text{IH}}} = \frac{4.75 - 2.4}{0.04N_{\text{I}}} = \frac{58.75}{N_{\text{I}}}. \tag{1}$$

All the inputs have $N_{\text{I}} = 1$, except for $\overline{\text{CD}}$, which has $N_{\text{I}} = 2$. The total value is $N_{\text{I}} = 15+3+2 = 20$, which gives $R_{\text{max}} = 2.938$K. To calculate the average we let $R_{\text{av}} = (R_{\text{max}}+R_{\text{min}})/2 = 1.969$K and a recommended value of R1 = 1.8K. For 74LS series logic a pullup is not recommend, R1 should be shorted or 0R.

The outputs for $\overline{\text{CLRM}}$ and $\overline{\text{SHFM}}$ also use 7401 open collector gates, to reuse some spare gates. The minimum and maximum pullup resistor values (in K or kΩ) are

$$R_{\text{min}} = \frac{V_{\text{CC,max}} - V_{\text{OL}}}{I_{\text{OL}} - 1.6U_{\text{IL}}} \qquad R_{\text{max}} = \frac{V_{\text{CC,min}} - V_{\text{OH}}}{N_{\text{O}}I_{\text{OH}} + 0.04U_{\text{IH}}} \tag{2}$$

where

$V_{\text{CC,max}}$ = maximum power supply voltage (5.25 V)
$V_{\text{CC,min}}$ = minimum power supply voltage (4.75 V)
V_{OL} = output low voltage level (0.4 V for 74, 0.5 V for 74LS)
V_{OH} = output high voltage level (2.4 V for 74, 2.7 V for 74LS)
I_{OL} = output low open collector current (16 mA for 7401, 8 mA for 74LS01)
I_{OH} = output high open collector leakage current (0.25 mA for 7401, 0.1 mA for 74LS01)
U_{IL} = summation of input low unit loads being driven (typically 1 for 74 and 0.25 for 74LS)
U_{IH} = summation of input high unit loads being driven (typically 1 for 74 and 0.5 for 74LS)
N_{O} = number of open collector outputs connected together

We thus have for 74 series TTL

$$R_{\text{min}} = \frac{4.85}{16 - 1.6N_{\text{I}}} \qquad R_{\text{max}} = \frac{2.35}{0.25N_{\text{O}} + 0.04N_{\text{I}}} \tag{3}$$

where N_{I} is the number of standard 1.6 mA input low and 0.04 mA input high inputs. For 74LS series TTL we have

$$R_{\text{LS,min}} = \frac{4.75}{8 - 0.4N_{\text{I}}} \qquad R_{\text{LS,max}} = \frac{2.05}{0.1N_{\text{O}} + 0.02N_{\text{I}}} \tag{4}$$

where N_{I} is the number of standard 1.6×0.25 = 0.4 mA input low and 0.04×0.5 = 0.02 mA input high inputs.

In this case we have $N_O = 1$ and $N_I = 2$ and 1 for $\overline{\text{CLRM}}$ and $\overline{\text{SHFM}}$, respectively. The table below gives the recommended values, where R2 and R3 are the pullups for $\overline{\text{CLRM}}$ and $\overline{\text{SHFM}}$, respectively.

Table 1. E8ME/B Pullup Resistor Values

			74 series TTL		74LS series TTL		74/74LS
	N_O	N_I	R_{min} (Ω)	R_{max} (Ω)	R_{min} (Ω)	R_{max} (Ω)	R_{rec} (Ω)
R2	1	2	379	7121	660	14643	3.9K
R3	1	1	337	8103	625	17083	4.7K

Decoder Board (E8ME/D)

E8ME/D1

The first task for the decoder board is to receive and write the MEA register values. Figure 1 shows the timing diagram for /ME_OUT_CLOCK, /ME_OUT_DATA and /ME_OUT_ RESET, where $\overline{\text{CLK}}$, $\overline{\text{DATA}}$ and $\overline{\text{RESET}}$, respectively, are used as the signal names. TC is the timer counter value in E8/T. Data in the accumulator (AC) register is shifted out least significant bit first during T2–9. The reset signal goes low at T13.

As AC changes after the rising edge of $\overline{\text{CLK}}$, either the rising or falling edge of $\overline{\text{CLK}}$ can be used to clock data into the address buffer (AB) shift register. The 74164 8–bit shift register has a rising edge clock input, so data is shifted in using the rising edge. The circuit for the AB register is shown in schematic E8ME/D1, where an inverter is used for $\overline{\text{DATA}}$ to restore it to its correct polarity.

Two six bit 74174 registers (for IDA and ODA) and two 8–bit 74273 registers (for IFA and DFA) are used to write AB to IDA, ODA or DFA (MEA write), depending on MB3A and MB4A during the low to high edge of /ME_OUT_ RESET and AB to IFA during the low to high edge of /IFAE. We also use MB3A and MB4A to read one of the four register values (MEA read) so these signals are modified according to the Karnaugh tables below.

MB4 \ OPR_IOT,SDF	00	01	11	10
0	0	1		0
1	0	1		1

MB4A = SDF + MB4.OPR_IOT

MB3 \ OPR_IOT,SDF	00	01	11	10
0	0	0		0
1	0	0		1

MB3A = MB3.OPR_IOT

Let MB3 and MB4 be the memory buffer values from the E8/M board. If OPR_IOT is high then MB4A = MB4 and MB3A = MB3. That is, for an IOT 6x3 or 6x7 instruction we either MEA read or write to address x = 2MB4+MB3. The exception is instruction 607 (write IFA) where MEA write is not performed since /IFAE is used to write AB to IFA. An MEA read onto the field address bus FA[7:0] also occurs for 61y and 63y instructions (read or write to port 1 with y = 1, 2, 4 or 6), where

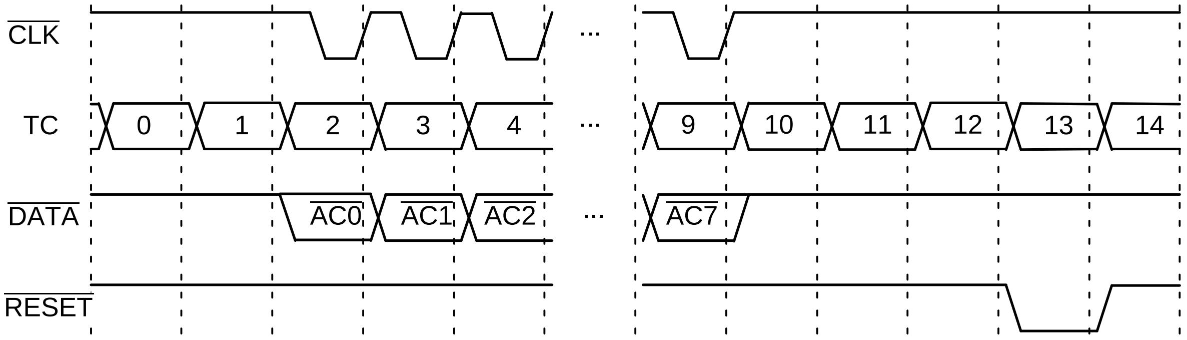


Figure 1: MEA Output Timing

x = 1 has FA = IDA and x = 3 has FA = ODA. FA is sent to the E8ME/I boards where FA[7:5] is used to select the EDUC–8ME unit and FA[2:0] one of the eight input or output ports. For other OPR and IOT instructions, MEA read or write does not occur during EXECUTE, when these signals are active.

If OPR_IOT is low and /SDF (select data field) is low, MEA write of AB to IFA occurs (when /IFAE goes low at the start of EXECUTE following a JMP or JMS instruction) and FA read of IFA (during DEPOSIT, EXAMINE, FETCH, DEFER and EXECUTE for JMS and JMP or direct AND, TAD, ISZ or DCA instructions). That is, MB4A = MB3A = 0 and we MEA read or write to address x = 0.

If OPR_IOT is low and /SDF is high (during EXECUTE for indirect AND, TAD, ISZ or DCA instruction), then an FA read of DFA is performed with MB4A = 1 and MB3A = 0 or address x = 2. Note that /SDF and OPR_IOT can not go high simultaneously.

The bus for the register that is read is called FA (field address), which goes to the MEA read shift register, the extended memory and the I/O ports. By using the same FA read address, we reduce complexity since we don't need separate multiplexers for an MEA read and for addressing the extended memory. As /MB3O = $\overline{\text{MB3.OPR\_IOT}}$ and /MB4O = $\overline{\text{MB4.OPR\_IOT}}$, we implement the signals as MB3A = $\overline{\text{/MB3O}}$ and MB4A = $\overline{\text{/SDF./MB4O}}$, as shown in schematic E8ME/D1.

One half of a 9321 or 74LS139 dual 1 of decoder is used to generate /IDAE (input device address enable active low), /DFAE (data field address enable active low) and /ODAE (output device address enable active low) from MB3A, MB4A and /ME_OUT_RESET. These signals go to the clock inputs of the IDA, DFA and ODA registers where the data from AB is written during the rising edge of /IDAE, /DFAE and /ODAE, respectively. This is shown in schematic E8ME/D1 for IDA and ODA and E8ME/D2 for DFA.

The other half of the 9321/74LS139 1 of 4 decoder is used to decode MB3A and MB4A to one of four active low outputs. Signals /IFA, /DFA, /IDA and /ODA should be passed to the /AND, /TAD, /ISZ and /DCA inputs of E8/F, respectively.

The signal /MR is used to reset the four registers on power up, similar to the master reset signal signal used in E8/T. This allows EDUC–8ME to be used for programs written for the EDUC–8, without having to set up the four MEA registers. Instead of two inverters, a spare 7411 three input AND gate is used.

Signals ID1_DATA, /ID1_FLAG, /OD1_FLAG and /DOUT are 7438 open collector outputs and thus require pullup resistors (R3 to R6). With up to N_{O} = 8 open collector outputs, I_{OL} = 30 UL (48 mA) and N_{I} = 1 standard inputs, this gives R_{min} = 105 Ω and R_{max} = 1152 Ω. For the 74LS37, we have I_{OL} = 15 UL (24 mA), which gives $R_{\mathrm{LS,min}}$ = 201 Ω and $R_{\mathrm{LS,max}}$ = 2500 Ω. For 74 and 74LS compatibility, a resistor value between 201 Ω and 1152 Ω (average 676.5 Ω) can be used. Choosing the closest standard resistor value this gives R3 to R6 = 680R.

The pullup for unused inputs R1 has $N_{\mathrm{I}} = 24$, which gives R_{max} = 58.75/24 = 2.448 kΩ. Averaging this with a 1K minimum value gives 1.724 kΩ for a recommended value of 1.8K.

E8ME/D2

In order to externally program the separate fields and for running programs, we need to initialise the IFA and DFA registers. We perform this using the switch register (SR) from EDUC–8 and the load field address signal (called /EXT_LFA) from the front panel of the first EDUC–8ME unit. As shown in E8ME/D2, two 74157 quad 2 to 1 multiplexers are used to either select AB or SR for input to IFA and DFA.

Two 7411 three input AND gates are used to select either /EXT_LFA or /IFAE for the IFA or /EXT_LFA or /DFAE for the DFA. In order to meet the data hold requirements of the IFA and DFA registers, /EXT_LFA is delayed using a spare 7412 three input NAND gate. The delayed EXT_LFA signal is then used to select SR when high or AB when low.

The maximum delay of /EXT_LFA to the clock input plus the data hold time is 7411 LH + 74273 (D hold) = 27+5 = 32 ns. The minimum delay of /EXT_LFA to the data inputs is 7412 HL + 74157

(S to Z) = 25×0.35+min(23×0.48, 27×0.35) = 8.75+min(11.04,9.45) = 18.2 ns, which is 32–18.2 = 13.8 ns short of the requirement. To ensure that /EXT_LFA is sufficiently delayed we add a 15 pF load capacitor to the output of the open collector NAND gate. Since the maximum delay of the NAND is specified for a 15 pF load and 400 Ω (2K for 74LS) pullup, using the same capacitor and higher pullup value of 3.9K should give a delay of 25+9.45 = 34.45 ns, which is 34.45–32 = 2.45 ns greater than the requirement.

As the SR register will be driving an additional load, we need to check that the 10K switch pullup resistors used on E8/F in EDUC–8 are not too high. The number of loads for the SR registers is three (two for EDUC–8 and one for EDUC–8ME). This gives $R_{\max}$ = 58.75/3 = 19.583 kΩ. Thus, the 10K switch pullups on EDUC–8 can continue to be used. The /EXT_LFA switch input also has three loads and thus can also use a 10K pullup. As recommended by Rowe for /EXT_LA, the switch capacitor should be 1 nF in value [4, page 66]. Note that if a paper tape reader is attached, parallel pullup resistors will need to be added as $R_{\max}$ is 6.351 kΩ with 7401 or 7437 OC gates.

As E8/F does not have OC inverters to drive the IFA7 and DFA7 LEDs, a 7412 triple 3–input NAND OC is used to output /IFA7 and /DFA7. Note that the 74LS12 with U_{OL} = 5 (8 mA) has a maximum I_{OL} that is twice this value, or 16 mA [2, page 3–4]. The output low voltage of a 74LS open collector output transistor at room temperature is about $0.09+29.3I_c$ volts (25°C curve for V_{OL} versus I_{OL} from [1, fig. 2–13]). Thus, the voltage across the LED is $V_{\mathrm{cc,max}}–0.09–(29.3+180×0.9)I_c = 5.16–191.3I_c$ using 10% tolerant resistors. The diode voltage is about $1.84+17.6I_c$ (I_f vs V_f curve for the 9 mcd 5 mm red LED from Altronics [3, fig. 3]). Solving, this gives I_c = (5.16–1.84)/(191.3+17.6) = 15.9 mA, which is just within specification.

The pullup R7 for EXT_LFA has one output and two inputs for 74, with a range of 379 to 7121 Ω. For the 74LS157, the select line has two inputs, giving a total of 1 output and 4 inputs, with a range of 742 to 11389 Ω. For 74/74LS compatibility the pullup should range from 742 to 7121 Ω (average 3932 Ω) for a recommended value of 3.9K.

E8ME/D3

Four 74153 dual 4 to 1 multiplexers are used to select one of the IFA, DFA, IDA and ODA registers to the FA[7:0] bus using the MEA address MB4A and MB3A. A 74165 8–bit parallel to serial converter is used to output the FA data to the AC register. The timing diagram is shown in Figure 2.

During T1, FA[7:0] is asynchronously loaded into Q[0:7] of the internal shift register using the active low input of /ME_IN_RESET. Since the output is from Q7 and the input is in reverse order, the first bit to be output is FA0. On the rising edge of /ME_IN_CLOCK (equivalent to the falling edge of MCPB) this bit is shifted into AC. Simultaneously, the next bit is output. Note that we can't use the falling edge of /ME_IN_CLOCK to shift data out, as this would cause the first bit to be shifted into AC being FA1, instead of FA0, with the last bit being a 1, as DS (serial input in) is high. A 7437 buffer is used to output the data to ME_IN_DATA.

7437 buffers are also used to distribute /FA0 to /FA7 to the E8ME/I boards of each EDUC–8ME unit via an IDC 40–pin connector, including the first unit.

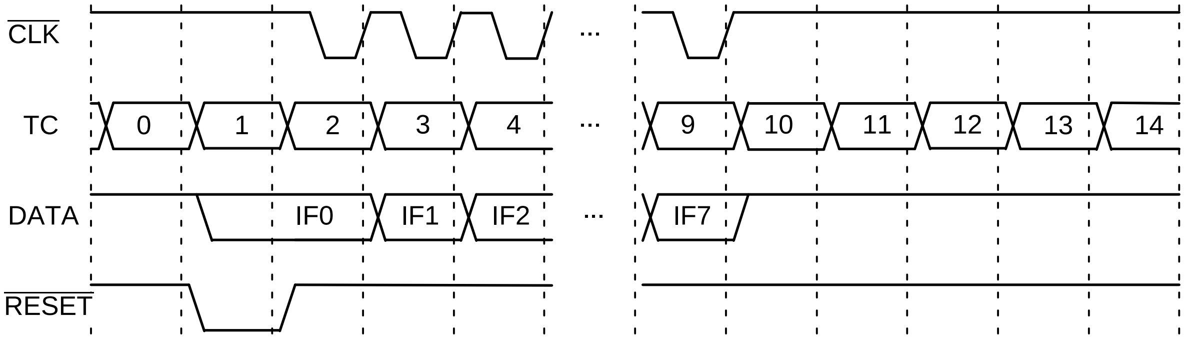


Figure 2: MEA Input Timing

Input/Output Port Multiplexer Board (E8ME/I)

The E8ME/I board provides decoding and access to the memory boards and interfacing to eight I/O ports. When accessing memory, three different configurations can be selected, depending on the type of RAMs used. We summarise this below.

1) For 2102/2125 1Kx1 SRAMs, FA[7:5] is used to select one of eight EDUC–8ME units, FA[4:3] is decoded to /CS[3:0] for selecting one of four memory boards, FA[2:0] and A10 going to the memory board decoders to select one of 16 SRAM chips and A[9:0] going to each SRAM chip.

2) For 2147 4Kx1 SRAMs, FA7 is used to select one of two EDUC–8ME units, FA[6:5] is decoded to /CS[3:0] for selecting one of four memory boards, FA[4:1] going to the memory board decoders to select one of 16 SRAM chips and FA[0] and A[10:0] going to each SRAM chip.

3) For 2167 16Kx1 SRAMs, FA7 is decoded to /CS[1:0] for selecting one of two memory boards, FA[6:3] going to the memory board decoders to select one of 16 SRAM chips and FA[2:0] and A[10:0] going to each SRAM chip.

Two jumpers X3 and X4 are used to select the appropriate address inputs for the above three configurations so as to generate /CS[3:0]. Inputs /A[10:0], WE, /DIN and /FA[2:0] are inverted and sent to the memory boards. For the 2147 or 2167, U22 is optionally used to invert /FA[7:3] and /A10 (using the three input jumper X5) and sent to the memory boards.

A DIP switch and three 74136 exclusive OR (XOR) open collector gates are used to generate the FAE (FA enable) signal. With 1Kx1 SRAMs we have FAE = (S2⊕/FA7).(S1⊕/FA6).(S0⊕/FA5) where (S2,S1,S0) is the EDUC–8ME unit selected. The 2x4 jumper X6 is use to select FAE and output to FAEM. For example for the first unit (number 0) we have (S2,S1,S0) = (0,0,0), which is selected by having the DIP switches in the closed position. The next unit (number 1) would be (0,0,1), etc. With 4Kx1 SRAMs, the FAE signal is not used. This allows FAE to independently select the I/O ports. Instead, we use X6 to select either /FA7 for the first EDUC–8 unit or FA7 for the second EDUC–8 unit. For 16Kx1 SRAMs X6 is used to select the logic HI signal (tied to VCC via pullup resistor R9). FAEM is used along with /ME from E8ME/B (via E8ME/D) to produce the /CS[3:0] signals using a 74155 dual 1 of 4 decoder.

If the I/O ports are being used, then FA[2:0] and FAE are used to select one of the eight I/O ports in each EDUC–8ME unit. As FAE is not dependent on the type of RAMs used, the full number of 64 I/O can be obtained with additional EDUC–8ME units. For example, with 16Kx1 SRAMs, the first EDUC–8ME unit could contain all 64KB of RAM and eight I/O ports, while a second unit could contain another eight I/O ports.

7438 OC buffers are used to send FA[2:0] to the I/O circuits and memory. A spare 74136 OC XOR gate is used to send A10 to the memory boards. Due to the three memory configurations, the high and low loads are different for A10, FA0 and FA[2:1], as given in Table 2. R5, R6, R7 and R8 are the pullups for A10, FA0, FA1 and FA2, respectively. The term $U_{\mathrm{IH}}/U_{\mathrm{IL}}$ is used to indicate the number of standard input high and input low units used, with 0.04 mA per unit for high and 1.6 mA per unit for low signals. The 7438 has U_{OL} = 30, the 74LS38 has U_{OL} = 15, the 74136 has U_{OL} = 10 and the 74LS136 has U_{OL} = 5. N_{O} = 1 for all signals.

We have that FA[2:0] drives four standard inputs for the I/O ports (one 74259 and three 74151A). With 1Kx1 SRAMs, FA[2:1] drives an additional eight standard inputs (select lines of four 74251 8 to 1 multiplexers and address lines for four 74154 1 of 16 decoders), and FA0 and A10 drive four standard inputs (74154 address lines). With 4Kx1 SRAMs, FA[2:1] drive four standard inputs (A1 and A0 of the 74154) and FA0 and A10 drive 64 SRAM address inputs (four boards with 16 RAMs on each board) with U_{IH} = 0.25 (0.01 mA) and U_{IL} = 0.00625 (0.01 mA) each. With 16Kx1 SRAMs, FA[2:0] and A10 drive 32 SRAM address inputs with the same loading per input as the 4Kx1 SRAMs.

To simplify resistor selection we choose the largest R_{min} and smallest R_{max} for each pullup for 74/74LS compatibility. The average and recommended values for R5 to R8 are given in Table 3.

Table 2. E8ME/I FA[2:0] and A10 R_{min} and R_{max} pullup values

		74 series TTL			74LS series TTL		
Pullup	SRAM	$U_{\mathrm{IH}}/U_{\mathrm{IL}}$	R_{min} (Ω)	R_{max} (Ω)	$U_{\mathrm{IH}}/U_{\mathrm{IL}}$	R_{min} (Ω)	R_{max} (Ω)
R5 (A10)	1Kx1	4/4	505	5732	2/1	742	11389
	4Kx1	16/0.4	316	2640	16/0.4	645	2770
	16Kx1	8/0.2	309	4123	8/0.2	618	4881
R6 (FA0)	1Kx1	8/8	138	4123	4/2	228	7885
	4Kx1	20/4.4	118	2238	18/1.4	218	2500
	16Kx1	12/4.2	117	3219	10/1.2	215	4100
R7 (FA1) R8 (FA2)	1Kx1	12/12	168	3219	6/3	247	6029
	4Kx1	8/8	138	4123	4/2	228	7885
	16Kx1	12/4.2	117	3219	10/1.2	215	4100

Table 3. E8ME/I FA[2:0] and A10 recommended 74/74LS pullup values

Pullup	R_{min} (Ω)	R_{max} (Ω)	R_{av} (Ω)	R_{rec}
R5	742	2640	1691	1.8K
R6	228	2238	1233	1.2K
R7/R8	247	3219	1733	1.8K

For FA[7:5] and FAE which use the 74136 OC XOR gates, pullups R1 to R4 are used. For the switch select inputs S0, S1 and S2 we use pullups R1, R2 and R3, respectively. With $U_{\mathrm{IH}} = 1$ this gives $R_{\mathrm{max}} = 58.75$ kΩ. Like the SR switch register, we let R1 = R2 = R3 = 10K.

For FAE, pullup R4 is used and has $N_{\mathrm{O}} = 3$ (for the three XOR gates) and $N_{\mathrm{I}} = 6$ (one 74259 and three 7438 for the I/O ports, one 74155 for /CS[3:0] and one 7438 for /DOUT from the memory). Table 4 gives the recommended values for R4.

Table 4. E8ME/I R4 and R10 Pullups

			74 series TTL		74LS series TTL		74/74LS	
	N_{O}	N_{I}	R_{min} (Ω)	R_{max} (Ω)	R_{min} (Ω)	R_{max} (Ω)	R_{av} (Ω)	R_{rec}
R4	3	6	758	2374	848	4881	1611	1.5K
R10	1	8	138	4123	228	7885	2178	2.2K

For A0 to A9, /WE and DIN, the 2102, 2147 and 2167 SRAM inputs have an input current of 0.01 mA or $U_{\mathrm{IH}} = 0.25$. With 4×16 = 64 SRAM chips in total with 1Kx1 or 4Kx1 SRAMs, this gives a total input load of 16 UL. The 7404 for U1 and U2 has $U_{\mathrm{OH}} = 20$ which is sufficient. However, the 74LS04 has $U_{\mathrm{OH}} = 10$ which is not sufficient. Thus, the 74LS04 must not be used for these 12 signals. Similarly, U22 driving A10 and FA3 to FA7 have similar loads and thus must use the 7404. As only 32 chips are used with 16Kx1 SRAMs, the 74LS04 can be used if desired. For the 2125 and 93425 $U_{\mathrm{IH}} = 1$ ($U_{\mathrm{IL}} = 0.025$ and 0.25, respectively). This means that only 20 SRAM chips can be used with the 7404.

A 7438 NAND open collector buffer is used to output DOUT from the memory boards to the /DOUT bus. FAEM is used to select the output. No pullup is required on DOUT, since the outputs from E8/M are tristate.

A 74259 3 to 8 decoder is used to generate IO[7:0] from FA[2:0] and FAE. Each of the corresponding outputs is used to select an appropriate output signal (using 7437 buffers). This is different to Rowe’s design, which used 74154 1 to 16 decoders for the outputs. However, as the 74154 (or

74138 1 to 8 decoder) only has standard outputs, these chips may have problems driving the long lines to the external I/O devices. Output signals /OD1_CLOCK, /OD1_RESET, /OD1_DATA, /ID1_CLOCK and /ID1_RESET (from the EDUC–8 OD1 and ID1 output and input ports) are inverted and used to enable the appropriate output or input port signal via the 7437 buffers.

For the input signals we use 74151A 8 to 1 multiplexers instead of the 74150 16 to 1 multiplexer used by Rowe. For the output of the multiplexer, this is output to shared lines from all the EDUC–8ME units to the /OD1_FLAG, ID1_DATA and /ID1_FLAG lines from EDUC–8. The pullups for these signals and /DOUT are on the E8ME/D board in the first EDUC–8ME unit. A spare 7438 OC buffer is used to invert /ID1_RESET. This has $N_{\mathrm{O}} = 1$ and $N_{\mathrm{I}} = 8$ with $U_{\mathrm{OL}} = 30$ for the 7438 and $U_{\mathrm{OL}} = 15$ for the 74LS38. Table 4 gives the recommended pullup values.

Pullup R9 is used to tie up to 14 inputs to VCC using the HI signal. We have $R_{\mathrm{max}} = 58.75/14 = 4.196$ kΩ. Averaging this with $R_{\mathrm{min}} = 1$ kΩ gives a recommended value of R9 = 2.7K. Note that the Fairchild 74LS136 has emitter type inputs and a HI is used on a spare 74LS136 gate to produce A10. For 74LS we have $R_{\mathrm{max}} = (4.75–2.7)/(0.02×14) = 7.321$ kΩ, then the 0 Ω R9 should be replaced with a 3.9K resistor.

Memory Board (E8ME/M)

To allow each memory board to be used in any location, a four way jumper X2 is used. For example, the first memory board would place a jumper across pins 1 and 2, connecting signal /CS0 to the enable input of the 74154 1 of 16 decoder and 74151A 8 to 1 multiplexer. For 1Kx1 SRAMs, signals FA[2:0] and A10 go to the address inputs of the decoder, to select one of the 16 memory chips when /CS is low. For 4Kx1 and 16Kx1 SRAMs, FA[4:1] and FA[6:3] go the 74154 address inputs A3 down to A0, respectively. Special jumpers X11 to X12 are used to select the appropriate signals A3D down to A0D as shown in schematic E8ME/M1. Each of these special jumpers are two 1x3 jumpers that are offset by one position.

The four memory chips that are supported each have different pinouts as shown in Figure 3.

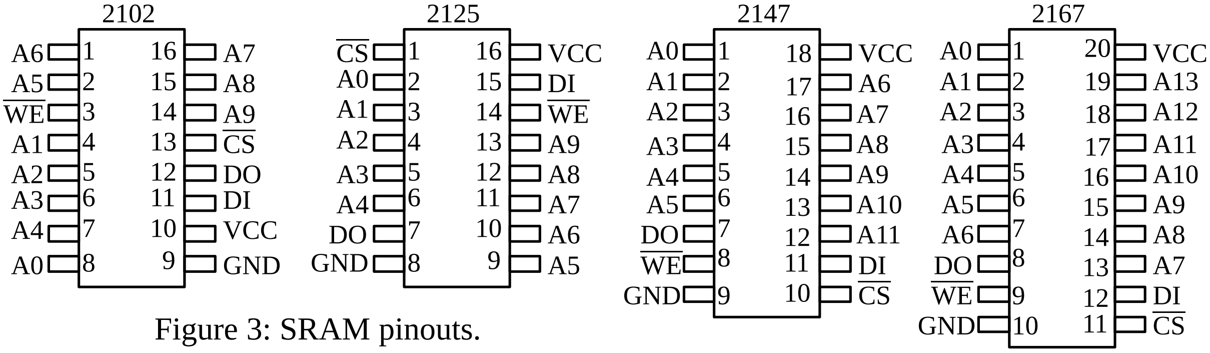


Figure 3: SRAM pinouts.

In order to allow overlap of the pins we first note that $\overline{\text{CS}}$ should not be shared with other pins, since there is a single line to each pin. We also rename the address pins as shown in Figure 4 so that the same address names are used between chips. To combine the chips into a single pinout, we first construct a 28 pin pattern where the 2102, 2147 and 2167 have their $\overline{\text{CS}}$ inputs aligned. As the 2125 has the $\overline{\text{CS}}$ input in the top left hand corner, we increase the pattern to 30 pins and add the 2125 to

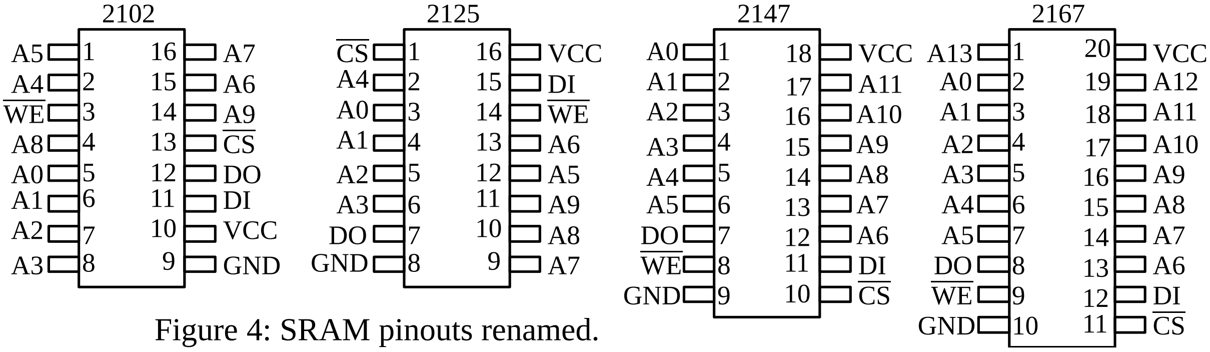


Figure 4: SRAM pinouts renamed.

the top of the pinout. As some pins are not common, we give them a combined name, e.g., A4–A13 where A4 is used for the 2125 and A13 for the 2167. Figure 5 gives the combined 30 pin pinout with symbol name 21XX. As two pins can’t have indentical names, some pins have an A suffix added, e.g., VCCA which is the same input as VCC.

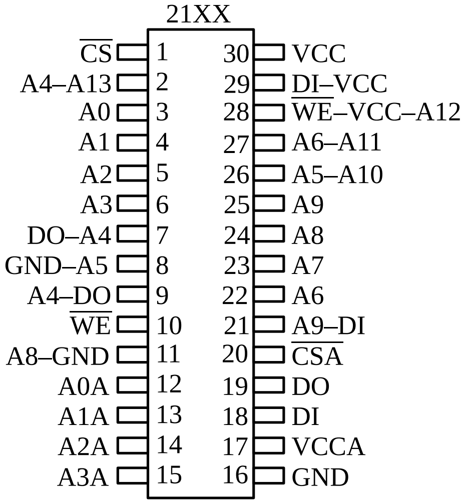


Figure 5: 21XX combined pinout.

Figure 6 shows the locations for the 2102, 2125, 2147 and 2167 SRAMs within the 21XX pinout, where red is used to indicate the pin labels selected. For pins with combined names, Table 5 gives the lines that are selected depending on the SRAM used.

Table 5: Signals selected for 21XX

Signal Name	2102	2125	2147	2167
A4–A13		A4		A13
DO–A4		DO	A4	A4
GND–A5	A5	GND	A5	A5
A4–DO	A4		DO	DO
A8–GND	A8		GND	GND
A9–DI	A9		DI	DI
A5–A10		A5	A10	A10
A6–A11		A6	A11	A11
$\overline{WE}$–VCC–A12		$\overline{WE}$	VCC	A12
DI–VCC		DI		VCC

Jumpers X3 to X10 are used to select eight of the ten signals, except for DO–A4 and A4–DO. For these two signals, a special 2x2 jumper is used that combines DO–A4 and A4–DO with A4 and DO (labelled DOUT in the schematics). For the 2102 we jumper pins P3 and P4, connecting A4 to A4–DO. For the 2125 we jumper pins P1 and P2, connecting DO–A4 to DOUT. For the 2147 and 2167 we jumper pins P1 and P3 (connecting A4 to DO–A4) and jumper pins P2 and P4 (connecting A4–DO to DOUT). As shown below, only four DOs of each SRAM can be tied together. Thus, four DO–A4 and A4–DO 2x2 jumpers are required, labelled X13 to X16.

For the 2102, the output driving capability is limited to an output high of 2.2 V and I_{OH} = 0.1 mA and output low of 0.45 V and I_{OL} = 1.9 mA. The tristate leakage current is I_{LOH} = 0.01 mA high and I_{LOL} = 0.1 mA low. Driving a single input, the input high currents are I_{IH} = 0.04 mA and I_{IL} = 1.6 mA, respectively. With one output enabled we have $I_O = (N–1)I_{LO}+I_I$, where N is the number of outputs. Thus, using $N = (I_O–I_I)/I_{LO}+1$, the maximum number of outputs tied together

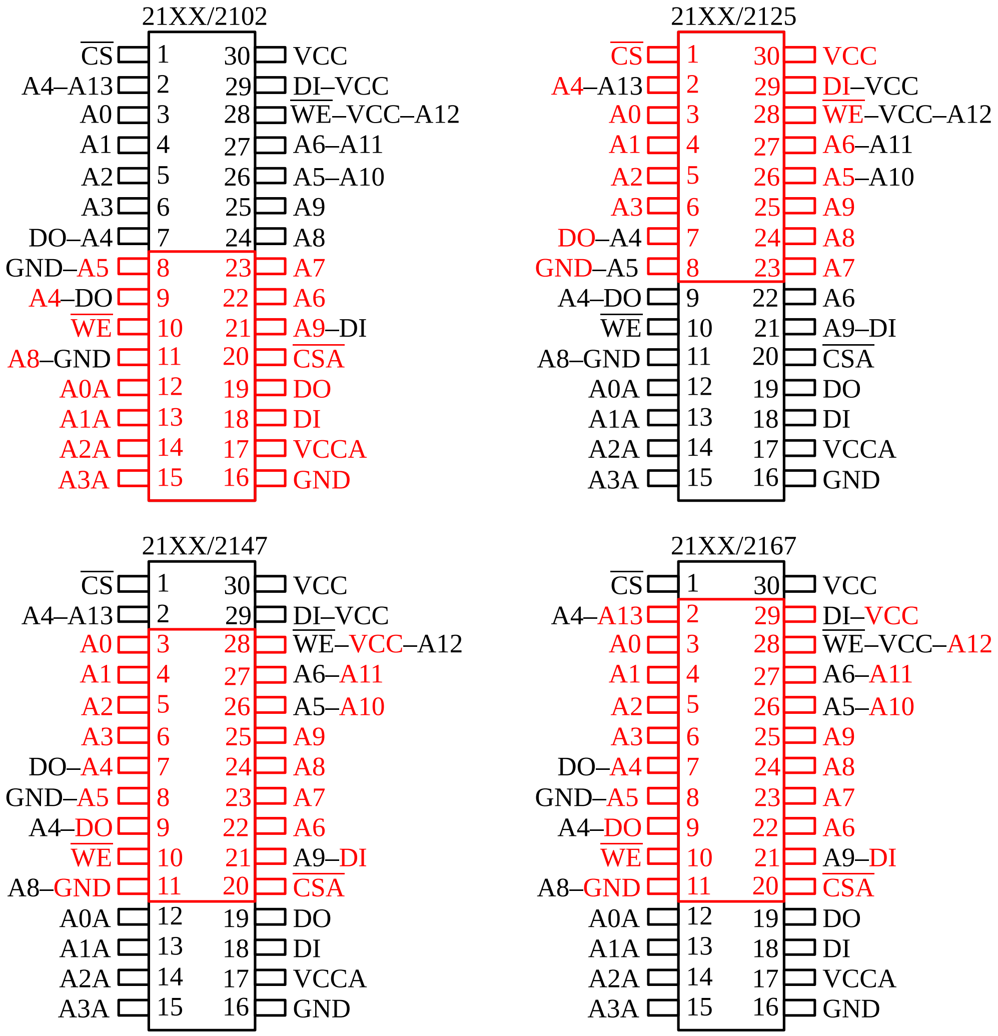


Figure 6: Pins selected for 21XX.

with one output high is N_H = (0.1–0.04)/0.01+1 = 7 and with one output low is N_H = (1.9–1.6)/0.1+1 = 4. Thus, the maximum number of 2102s that can tied together is only four, whereas we have up to 16 2102s on each board.

To overcome this problem, a 74251 8 to 1 multiplexer is used to select one of four outputs, with each output corresponding to four 2102s tied together.. The 74251 was chosen as it has a tristate output, which is enabled when /CS goes low. For 1Kx1 SRAMs, signals FA[2:1] are used to select the output, which is why these signals have an addition four UL compared to FA0. For 4Kx1 and 16x1 SRAMs, the select inputs are FA[4:3] and FA[6:5], respectively. These signals are the same as A3D and A2D.

Five inputs of the 74251 are tied high which results in R_{max} = 58.75/5 = 11.75 kΩ and R_{av} = 6.375 kΩ assuming R_{min} = 1K. As a maxium of 5K is recommended for pullups, we let R1 = 4.7K.

Implementation Notes

The recommended pullups for open collector outputs can be used either with 74 or 74LS series TTL. For 74LS, pullups used to tie unused inputs to HI should be 0R, except for the Fairchild 74LS136 if used in E8ME/I.

E8/T Timing Board: For 74 series TTL replace 7404 closest to edge connector with 74H04 or 74S04. Replace 7420 used for $\overline{\text{RST}}$ with 7440. No changes required for 74LS. Connect $\overline{\text{RST}}$ from pin 6 of 7440/74LS20 to pin 25 of edge connector.

E8/M Memory Board: The 1Kx1 memory chips are not loaded. Connect cable from the lower position 16–pin DIL socket (the 93415 closest to the 7493) used for the memory chips to the E8ME/B board. Make sure that VCC is not connected to the cable.

E8ME/B Buffer Board: Connect pins 1 and 2 of jumper X2 for normal operation. To adjust $\overline{\text{WE}}$ timing, disconnect EDUC–8ME units and cable to E8/M. Insert 93415 or 2115 1Kx1 RAM into lower position of E8/M and place jumper across pins 2 and 3 of X2. Load instruction JMP 0 (500) into address 000 and run program. Adjust trimpot R4 so that low width of QB (pin 6 of 74122) is at least 750 ns and at most T_{ch}–132 ns, where T_{ch} is the high width of MCPB (pin 1 of 74122).

E8ME/I Multiplexer Board: The positions of switches A3/B3, A2/B2 and A1/B1 in the 1st unit are ON,ON,ON, 2nd is ON,ON,OFF, 3rd is ON,OFF,ON, 4th is ON,OFF,OFF, 5th is OFF,ON,ON, 6th is OFF,ON,OFF, 7th is OFF,OFF,ON and 8th is OFF,OFF,OFF. U1, U2 and U22 must use 7404 inverters for 2102 and 2147 SRAMs. The 74LS04 does not have sufficient drive for memory boards, except for the 2167. For 2125 and 93425 SRAMs, there is a limit of 20 chips.

E8ME/M Memory Board: For connector X2, place jumper across pins 1 and 2 for first, 3 and 4 for second, 5 and 6 for third and 7 and 8 for fourth memory board.

References

[1] S. S. Pietrobon, "EDUC–8L: Adding a link bit to the EDUC–8 microcomputer," 2024. http://www.sworld.com.au/steven/educ–8/educ–8l.pdf

[2] Fairchild, "TTL data book," 1978. https://bitsavers.org/components/fairchild/_dataBooks/1978_Fairchild_TTL_Data_Book.pdf

[3] Bright LED Electronics Corp., "5 mm bright red LED," BL–B5134, Dec. 2004. https://download.altronics.com.au/files/datasheets_Z0800.pdf

[4] J. Rowe, "Interfacing EDUC–8 with punched paper tape," Electronics Australia, Mar. 1975, pp. 64–73.

Revision History

13 Mar. 2023. First release.

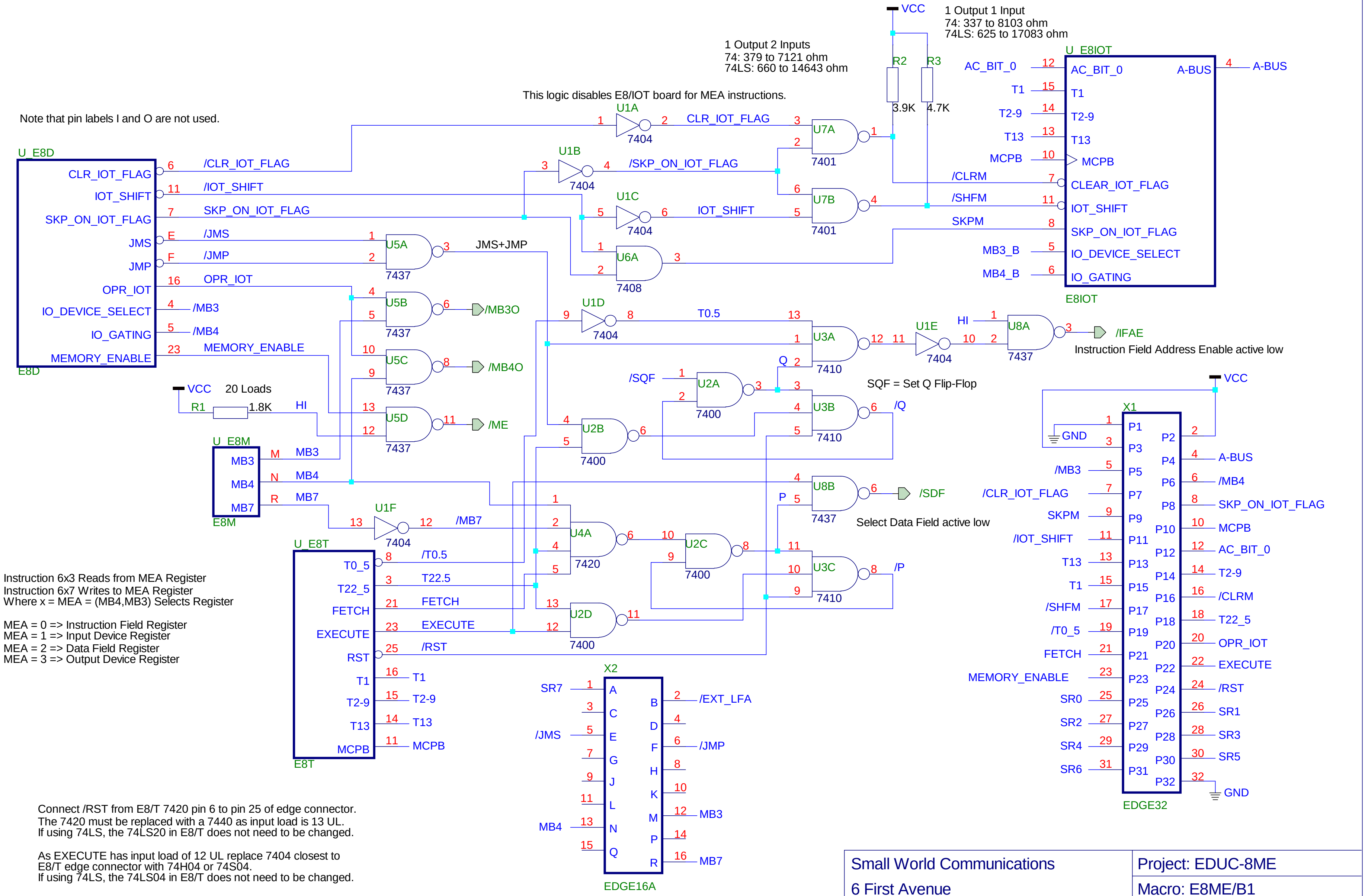
14 Mar. 2023. Changed E8ME/D U20 from 9321/74LS139 to 7400. Added Implementation Notes.

15 Mar. 2023. Corrected resistor calculation for R9 of E8ME/D. Added Revision History.

17 Mar. 2023. Updated pullup resistors R1 for E8ME/B, E8ME/D and E8ME/M. Added optional R7 SIL for E8ME/B.

21 Mar. 2023. Added 7422 dual 2–input NAND OC to E8ME/D for $\overline{\text{IFA7}}$ and $\overline{\text{DFA7}}$ outputs to E8/F.

29 Jun. 2024. Moved unit selection and address decoding logic from E8ME/D to E8ME/I. This means that E8ME/D is only required in the first EDUC–8ME unit, reducing complexity if more than one unit is used and reducing the load on SR[7:0]. Updated E8ME/M boards to allow option of using 2125 1Kx1, 2147 4Kx1 or 2167 16Kx1 SRAMs. Added references.



1 Output 1 Input
74: 337 to 8103 ohm
74LS: 625 to 17083 ohm

1 Output 2 Inputs
74: 379 to 7121 ohm
74LS: 660 to 14643 ohm

This logic disables E8/IOT board for MEA instructions.

Note that pin labels I and O are not used.

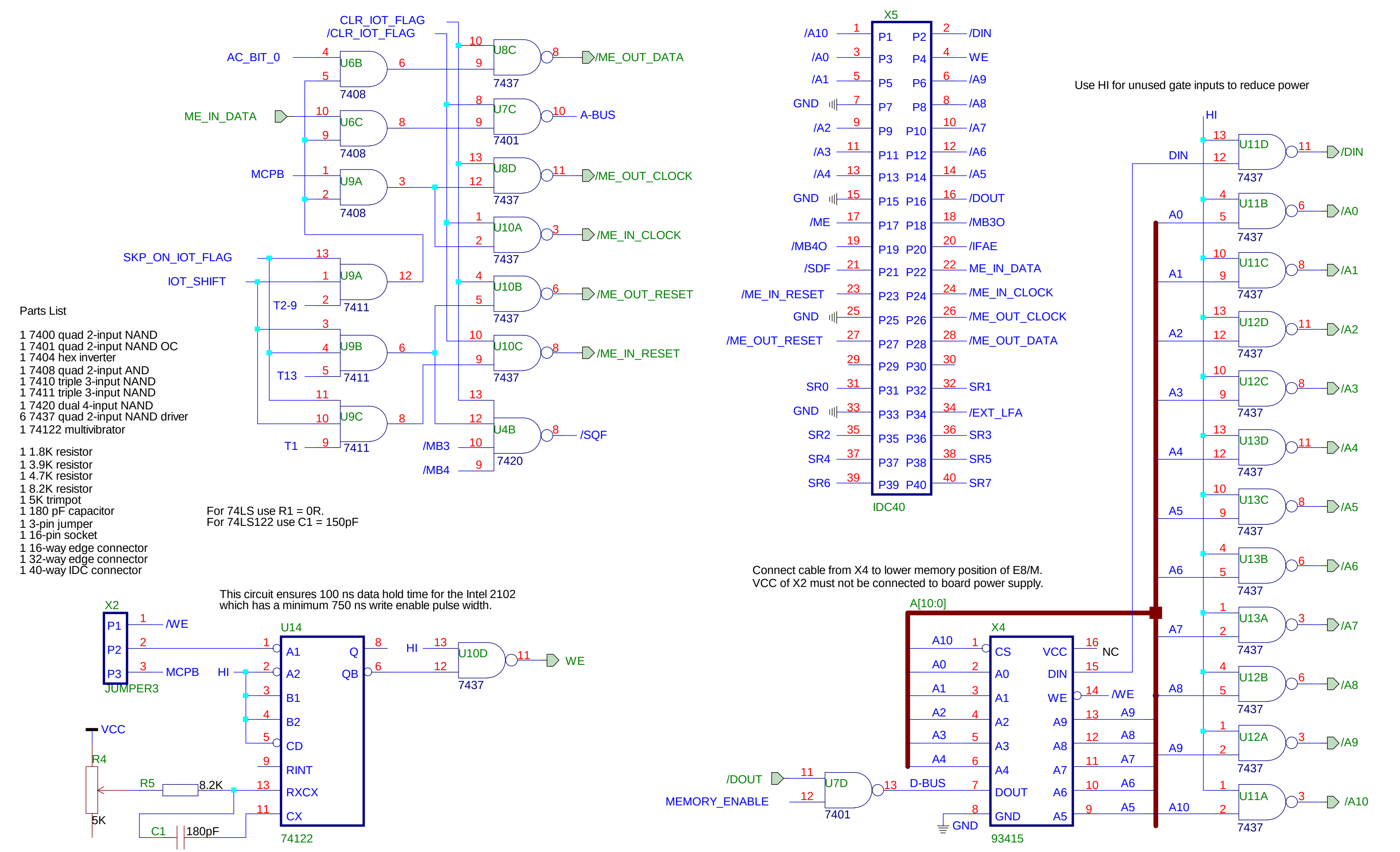
Instruction 6x3 Reads from MEA Register
Instruction 6x7 Writes to MEA Register
Where x = MEA = (MB4,MB3) Selects Register

MEA = 0 => Instruction Field Register
MEA = 1 => Input Device Register
MEA = 2 => Data Field Register
MEA = 3 => Output Device Register

Connect /RST from E8/T 7420 pin 6 to pin 25 of edge connector.
The 7420 must be replaced with a 7440 as input load is 13 UL.
If using 74LS, the 74LS20 in E8/T does not need to be changed.

As EXECUTE has input load of 12 UL replace 7404 closest to
E8/T edge connector with 74H04 or 74S04.
If using 74LS, the 74LS04 in E8/T does not need to be changed.

Small World Communications	Project: EDUC-8ME
6 First Avenue	Macro: E8ME/B1
Payneham South SA 5070	Date: 1/22/22
Date Last Modified: 6/20/24	



Use HI for unused gate inputs to reduce power

Connect cable from X4 to lower memory position of E8/M.
VCC of X2 must not be connected to board power supply.

This circuit ensures 100 ns data hold time for the Intel 2102
which has a minimum 750 ns write enable pulse width.

For 74LS use R1 = 0R.
For 74LS122 use C1 = 150pF

Parts List

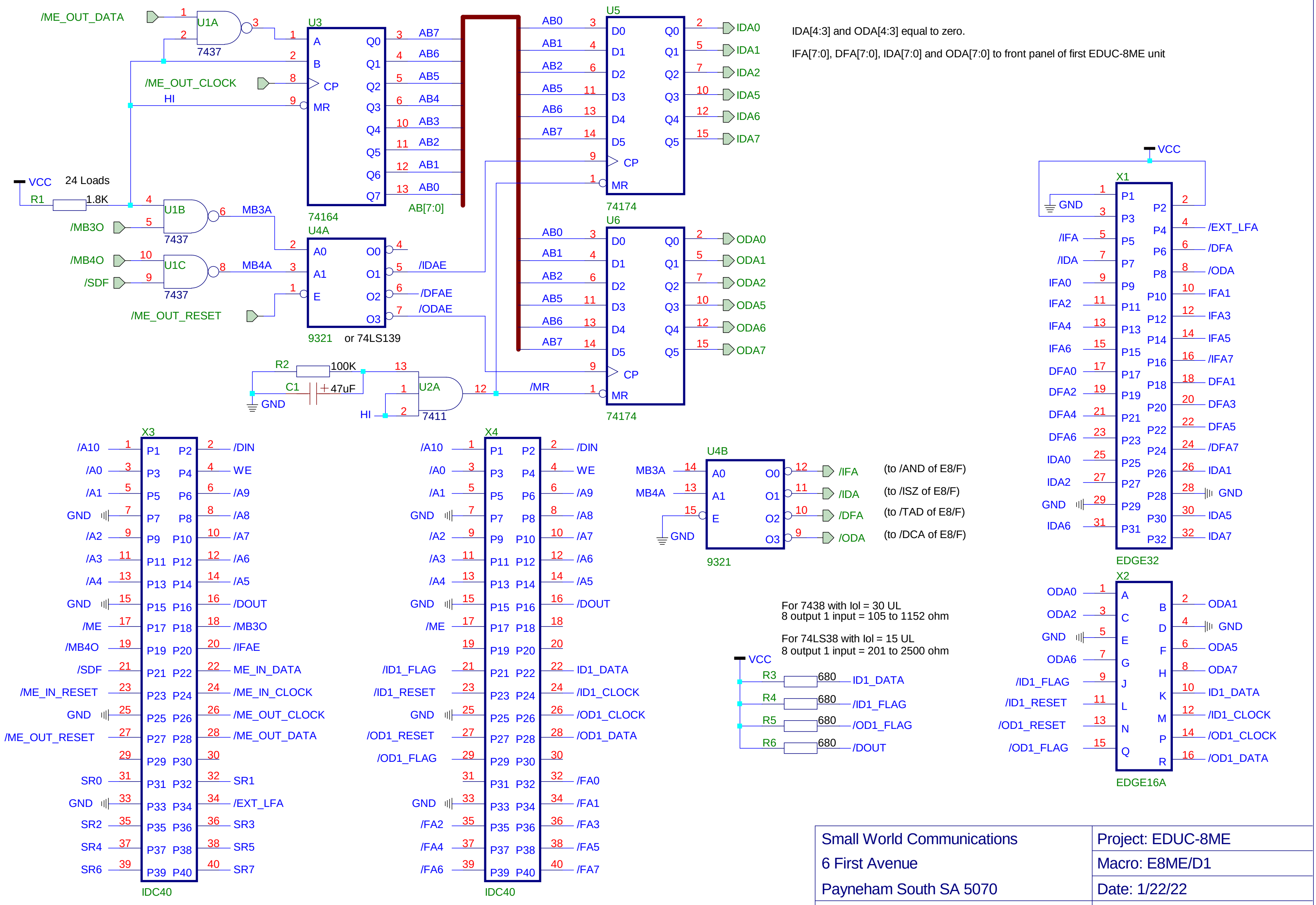
1 7400 quad 2-input NAND
1 7401 quad 2-input NAND OC
1 7404 hex inverter
1 7408 quad 2-input AND
1 7410 triple 3-input NAND
1 7411 triple 3-input NAND
1 7420 dual 4-input NAND
6 7437 quad 2-input NAND driver
1 74122 multivibrator

1 1.8K resistor
1 3.9K resistor
1 4.7K resistor
1 8.2K resistor
1 5K trimpot
1 180 pF capacitor
1 3-pin jumper
1 16-pin socket
1 16-way edge connector
1 32-way edge connector
1 40-way IDC connector

For normal operation jumper pins 1 and 2 of X2.

To adjust WE pulse width, disconnect EDUC-8ME and cable from E8/M.
Insert 93415 into lower position of E8/M and jumper pins 2 and 3 of X2.
Load instruction JMP 0 (500) into address 000 and run program.
Adjust trimpot R4 so that the low width of QB is at least 750 ns
and at most Tch-132 ns where Tch is the MCPB high width in ns.

Small World Communications	Project: EDUC-8ME
6 First Avenue	Macro: E8ME/B2
Payneham South SA 5070	Date: 1/22/22
Date Last Modified: 6/29/24	



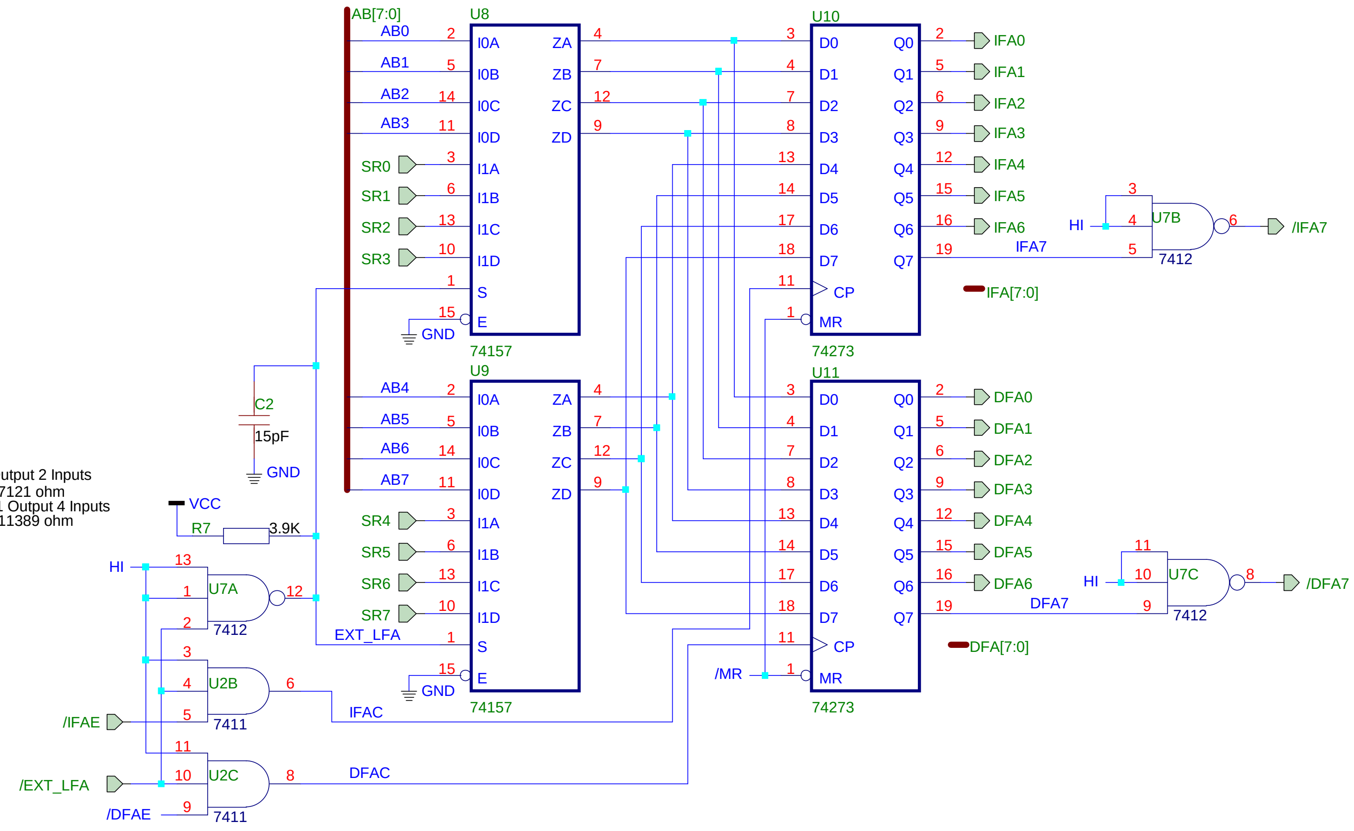
IDA[4:3] and ODA[4:3] equal to zero.

IFA[7:0], DFA[7:0], IDA[7:0] and ODA[7:0] to front panel of first EDUC-8ME unit

For 7438 with Iol = 30 UL
8 output 1 input = 105 to 1152 ohm

For 74LS38 with Iol = 15 UL
8 output 1 input = 201 to 2500 ohm

Small World Communications	Project: EDUC-8ME
6 First Avenue	Macro: E8ME/D1
Payneham South SA 5070	Date: 1/22/22
Date Last Modified: 6/20/24	



74: 1 Output 2 Inputs
379 to 7121 ohm
74LS: 1 Output 4 Inputs
742 to 11389 ohm

/EXT_LFA goes to /EXT_LA of front panel of first EDUC-8ME unit
with 10K pullup and 1nF capacitor.

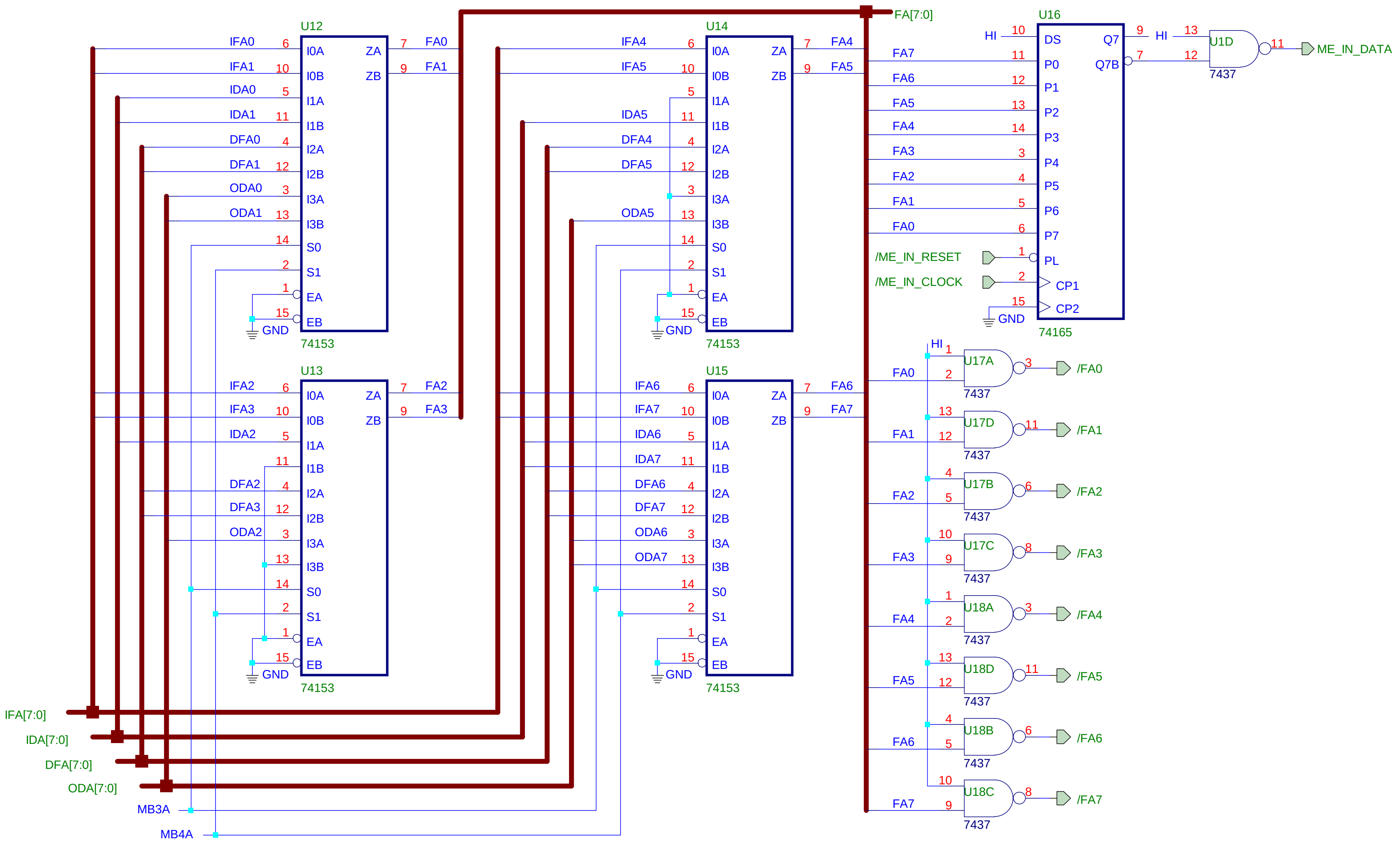
Parts List

1 7411 triple 3-input AND
1 7412 triple 3-input NAND OC
3 7437 quad 2-input NAND buffer
1 74139/9321 dual 2 to 4 decoder
4 74153 dual 4 to 1 multiplexer
2 74157 quad 2 to 1 multiplexer
1 74164 8-bit S/P shift register
1 74165 8-bit P/S shift register
2 74174 6-bit D-FF
2 74273 8-bit D-FF

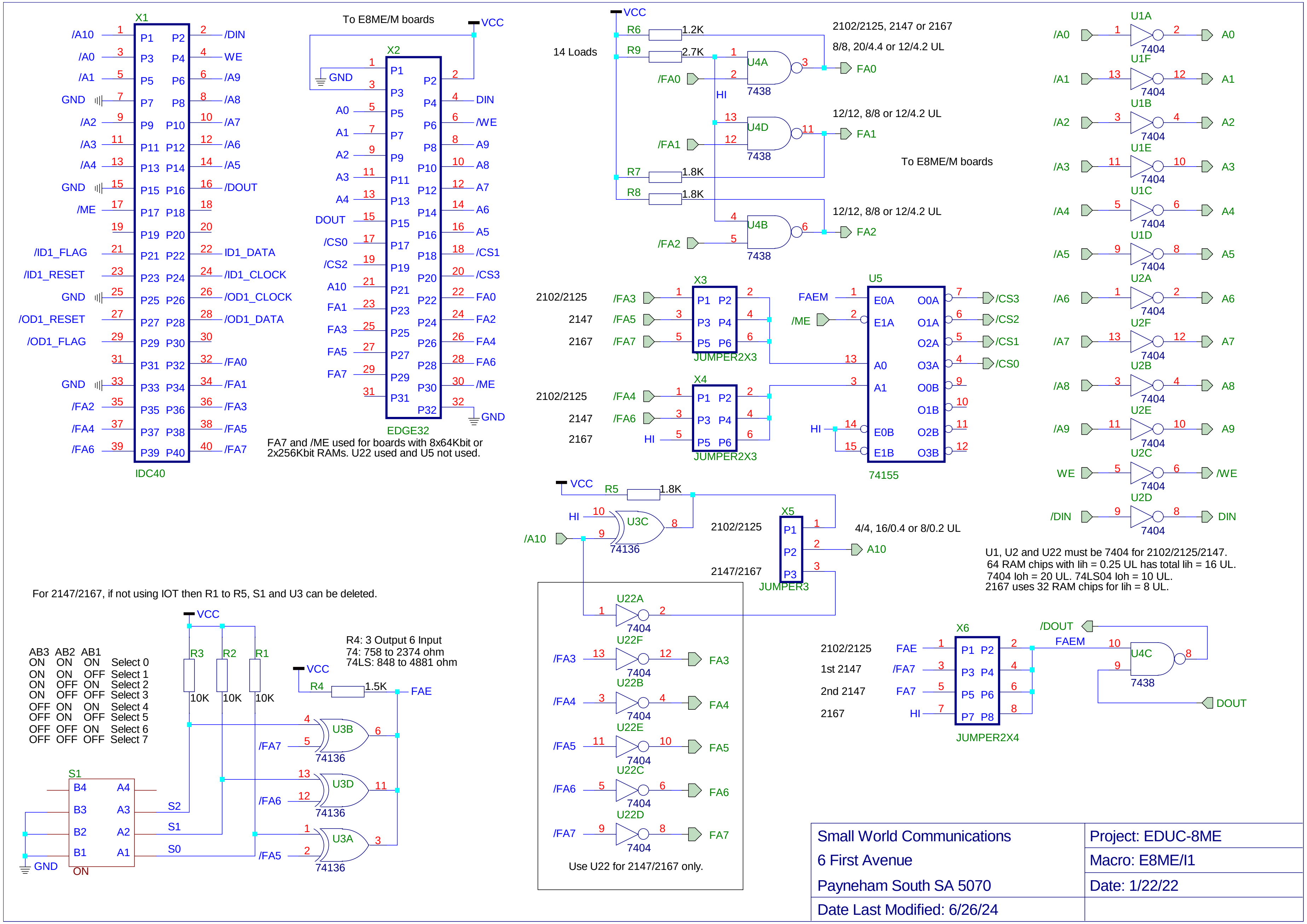
4 680R resistors
1 3.9K resistor
1 100K resistor
1 15 pF capacitor
1 47 uF capacitor
1 16-way edge connector
1 32-way edge connector
2 40-way IDC connectors

For 74LS use R1 = 0R.

Small World Communications	Project: EDUC-8ME
6 First Avenue	Macro: E8ME/D2
Payneham South SA 5070	Date: 1/22/22
Date Last Modified: 6/29/24	



Small World Communications	Project: EDUC-8ME
6 First Avenue	Macro: E8ME/D3
Payneham South SA 5070	Date: 1/22/22
Date Last Modified: 6/7/24	



To E8ME/M boards

14 Loads

2102/2125, 2147 or 2167
8/8, 20/4.4 or 12/4.2 UL

12/12, 8/8 or 12/4.2 UL

To E8ME/M boards

12/12, 8/8 or 12/4.2 UL

FA7 and /ME used for boards with 8x64Kbit or 2x256Kbit RAMs. U22 used and U5 not used.

4/4, 16/0.4 or 8/0.2 UL

U1, U2 and U22 must be 7404 for 2102/2125/2147.
64 RAM chips with Iih = 0.25 UL has total Iih = 16 UL.
7404 Ioh = 20 UL. 74LS04 Ioh = 10 UL.
2167 uses 32 RAM chips for Iih = 8 UL.

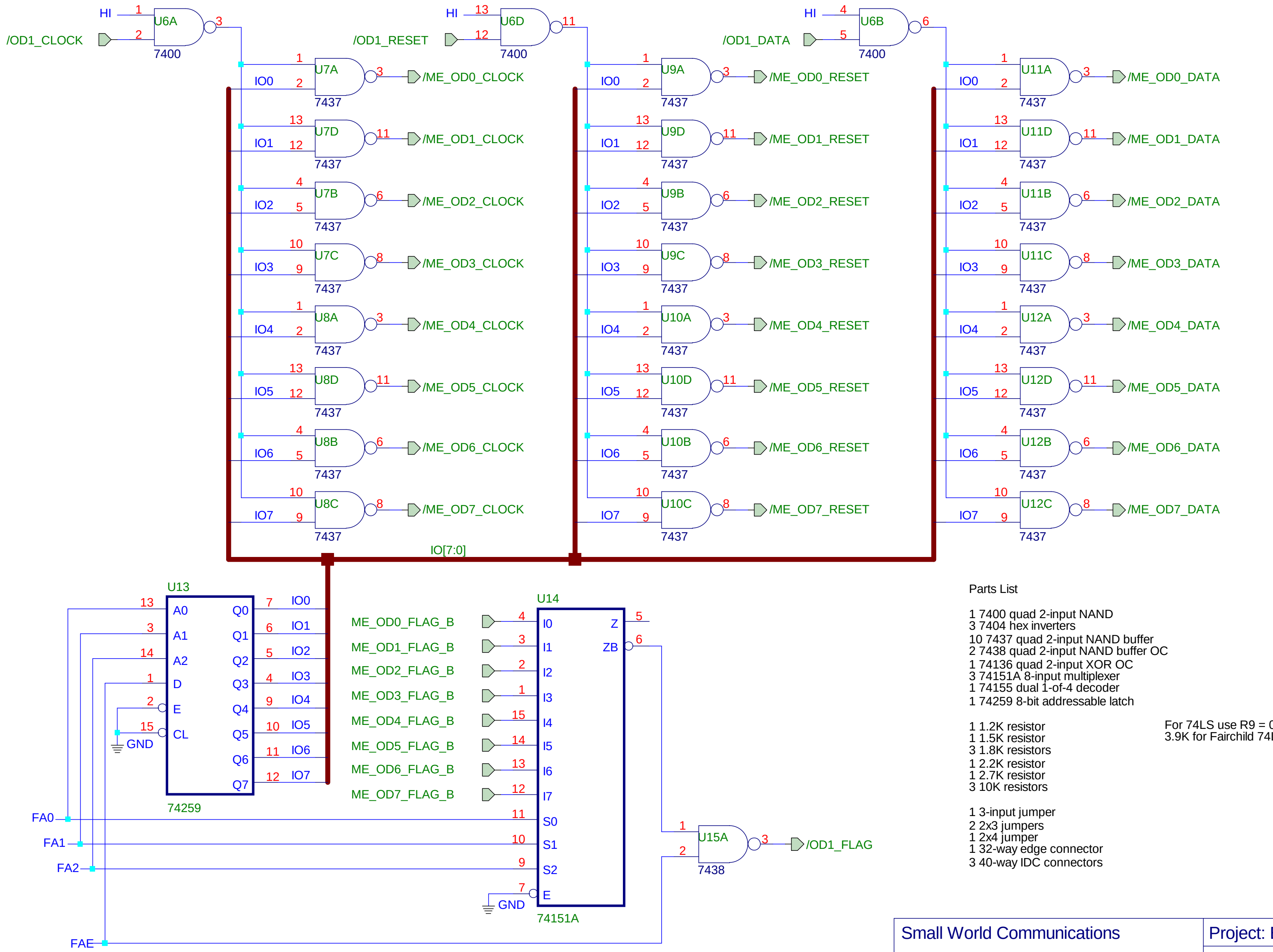
For 2147/2167, if not using IOT then R1 to R5, S1 and U3 can be deleted.

R4: 3 Output 6 Input
74: 758 to 2374 ohm
74LS: 848 to 4881 ohm

AB3	AB2	AB1	
ON	ON	ON	Select 0
ON	ON	OFF	Select 1
ON	OFF	ON	Select 2
ON	OFF	OFF	Select 3
OFF	ON	ON	Select 4
OFF	ON	OFF	Select 5
OFF	OFF	ON	Select 6
OFF	OFF	OFF	Select 7

Use U22 for 2147/2167 only.

Small World Communications	Project: EDUC-8ME
6 First Avenue	Macro: E8ME/I1
Payneham South SA 5070	Date: 1/22/22
Date Last Modified: 6/26/24	


HI
/OD1_CLOCK
U6A
7400
HI
/OD1_RESET
U6D
7400
HI
/OD1_DATA
U6B
7400
U7A 7437 IO0 /ME_OD0_CLOCK
U7D 7437 IO1 /ME_OD1_CLOCK
U7B 7437 IO2 /ME_OD2_CLOCK
U7C 7437 IO3 /ME_OD3_CLOCK
U8A 7437 IO4 /ME_OD4_CLOCK
U8D 7437 IO5 /ME_OD5_CLOCK
U8B 7437 IO6 /ME_OD6_CLOCK
U8C 7437 IO7 /ME_OD7_CLOCK
U9A 7437 IO0 /ME_OD0_RESET
U9D 7437 IO1 /ME_OD1_RESET
U9B 7437 IO2 /ME_OD2_RESET
U9C 7437 IO3 /ME_OD3_RESET
U10A 7437 IO4 /ME_OD4_RESET
U10D 7437 IO5 /ME_OD5_RESET
U10B 7437 IO6 /ME_OD6_RESET
U10C 7437 IO7 /ME_OD7_RESET
U11A 7437 IO0 /ME_OD0_DATA
U11D 7437 IO1 /ME_OD1_DATA
U11B 7437 IO2 /ME_OD2_DATA
U11C 7437 IO3 /ME_OD3_DATA
U12A 7437 IO4 /ME_OD4_DATA
U12D 7437 IO5 /ME_OD5_DATA
U12B 7437 IO6 /ME_OD6_DATA
U12C 7437 IO7 /ME_OD7_DATA
IO[7:0]
U13
A0 A1 A2 D E CL
Q0 Q1 Q2 Q3 Q4 Q5 Q6 Q7
IO0 IO1 IO2 IO3 IO4 IO5 IO6 IO7
GND
74259
ME_OD0_FLAG_B
ME_OD1_FLAG_B
ME_OD2_FLAG_B
ME_OD3_FLAG_B
ME_OD4_FLAG_B
ME_OD5_FLAG_B
ME_OD6_FLAG_B
ME_OD7_FLAG_B
U14
I0 I1 I2 I3 I4 I5 I6 I7 S0 S1 S2 E
Z ZB
GND
74151A
U15A
7438
/OD1_FLAG
FA0
FA1
FA2
FAE

If not using MUX do not use R10, U6 to U21, X7 and X8.

Parts List

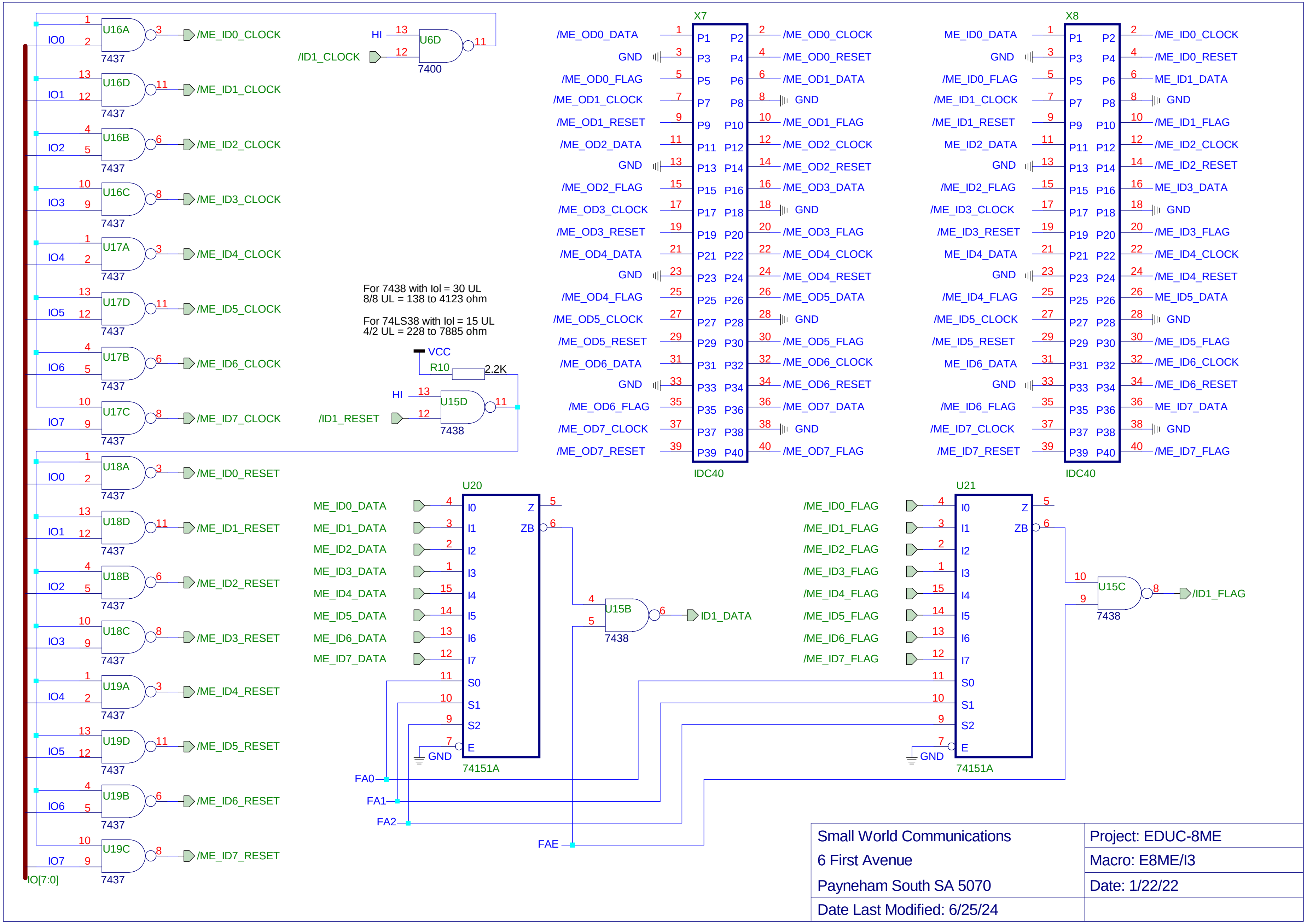
1 7400 quad 2-input NAND
3 7404 hex inverters
10 7437 quad 2-input NAND buffer
2 7438 quad 2-input NAND buffer OC
1 74136 quad 2-input XOR OC
3 74151A 8-input multiplexer
1 74155 dual 1-of-4 decoder
1 74259 8-bit addressable latch

1 1.2K resistor
1 1.5K resistor
3 1.8K resistors
1 2.2K resistor
1 2.7K resistor
3 10K resistors

1 3-input jumper
2 2x3 jumpers
1 2x4 jumper
1 32-way edge connector
3 40-way IDC connectors

For 74LS use R9 = 0R or
3.9K for Fairchild 74LS136.

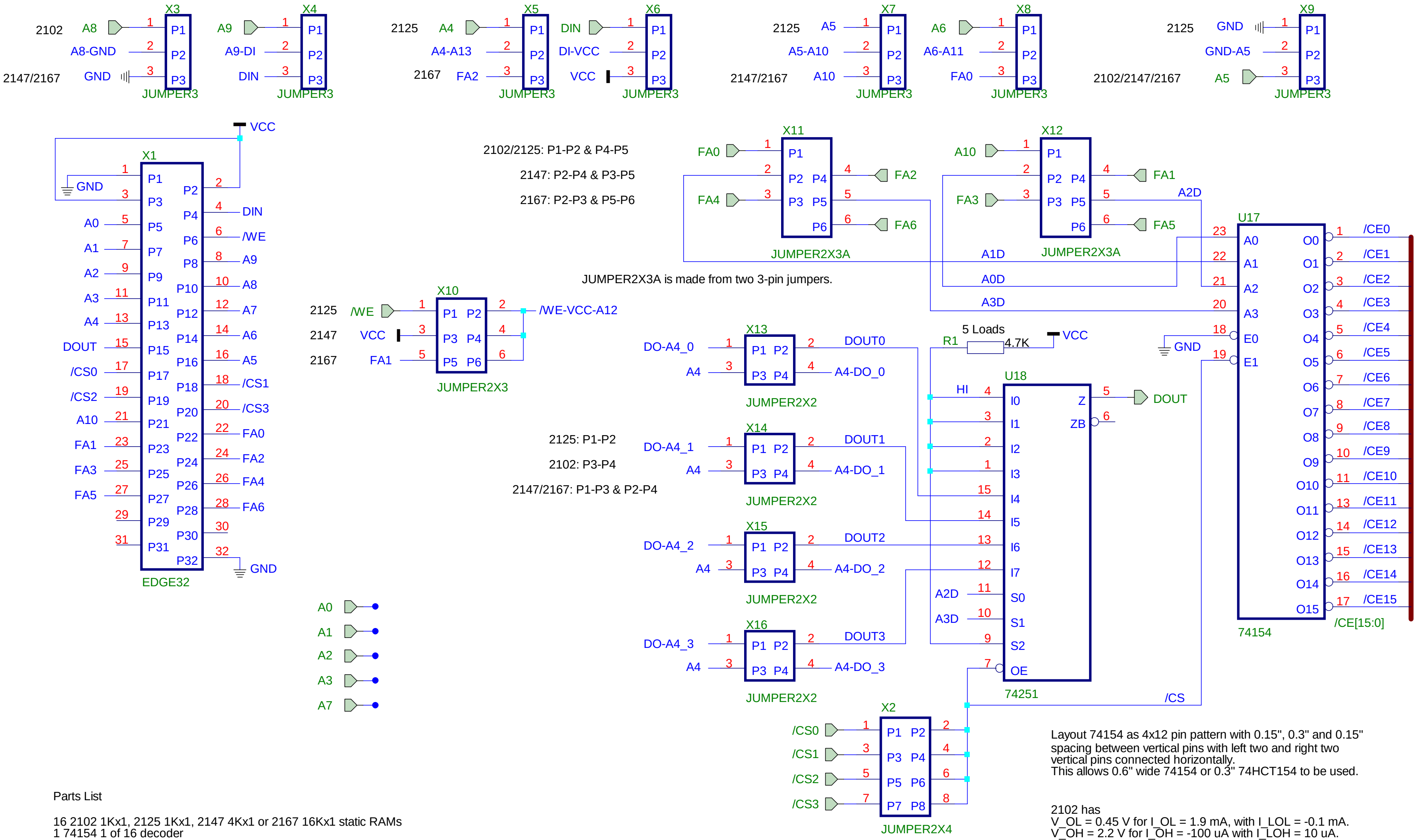
Small World Communications	Project: EDUC-8ME
6 First Avenue	Macro: E8ME/I2
Payneham South SA 5070	Date: 1/22/22
Date Last Modified: 6/29/24	



For 7438 with Iol = 30 UL
8/8 UL = 138 to 4123 ohm

For 74LS38 with Iol = 15 UL
4/2 UL = 228 to 7885 ohm

Small World Communications	Project: EDUC-8ME
6 First Avenue	Macro: E8ME/I3
Payneham South SA 5070	Date: 1/22/22
Date Last Modified: 6/25/24	



2102/2125: P1-P2 & P4-P5

2147: P2-P4 & P3-P5

2167: P2-P3 & P5-P6

JUMPER2X3A is made from two 3-pin jumpers.

2125: P1-P2

2102: P3-P4

2147/2167: P1-P3 & P2-P4

Connect one jumper for board selection.

Layout 74154 as 4x12 pin pattern with 0.15", 0.3" and 0.15"
spacing between vertical pins with left two and right two
vertical pins connected horizontally.
This allows 0.6" wide 74154 or 0.3" 74HCT154 to be used.

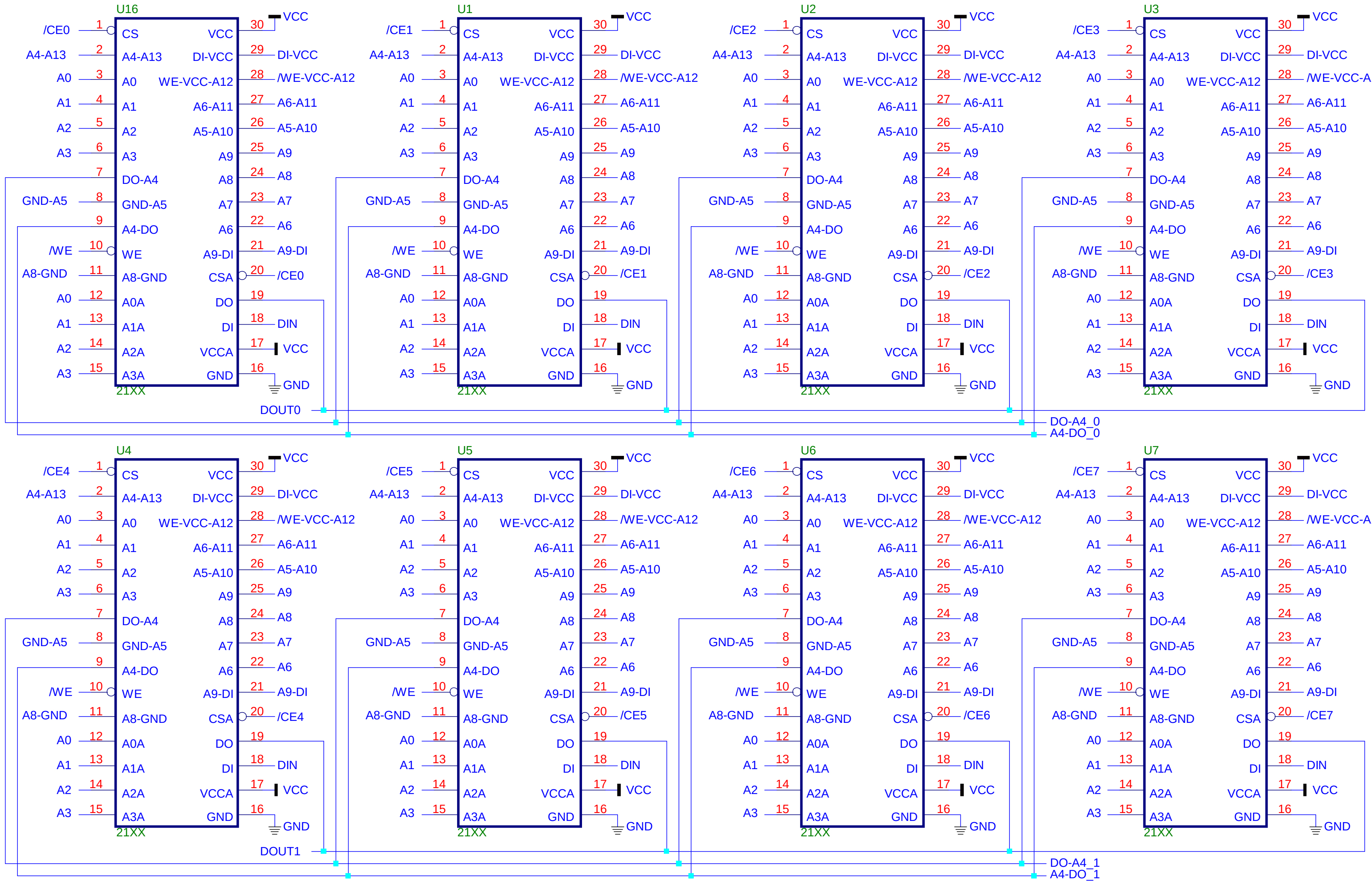
2102 has
V_OL = 0.45 V for I_OL = 1.9 mA, with I_LOL = -0.1 mA.
V_OH = 2.2 V for I_OH = -100 uA with I_LOH = 10 uA.
Thus four 2102 outputs driving one TTL input has
I_L = -1.6-3*0.1 = -1.9 mA and I_H = 40+3*10 = 70 uA.

Parts List

16 2102 1Kx1, 2125 1Kx1, 2147 4Kx1 or 2167 16Kx1 static RAMs
1 74154 1 of 16 decoder
1 74251 8-input multiplexer 3S
1 4.7K resistor
4 2x2 jumpers
1 2x3 jumper
1 2x4 jumper
11 3-pin jumpers
1 32-way edge connector

For 74LS use R1 = 0R.

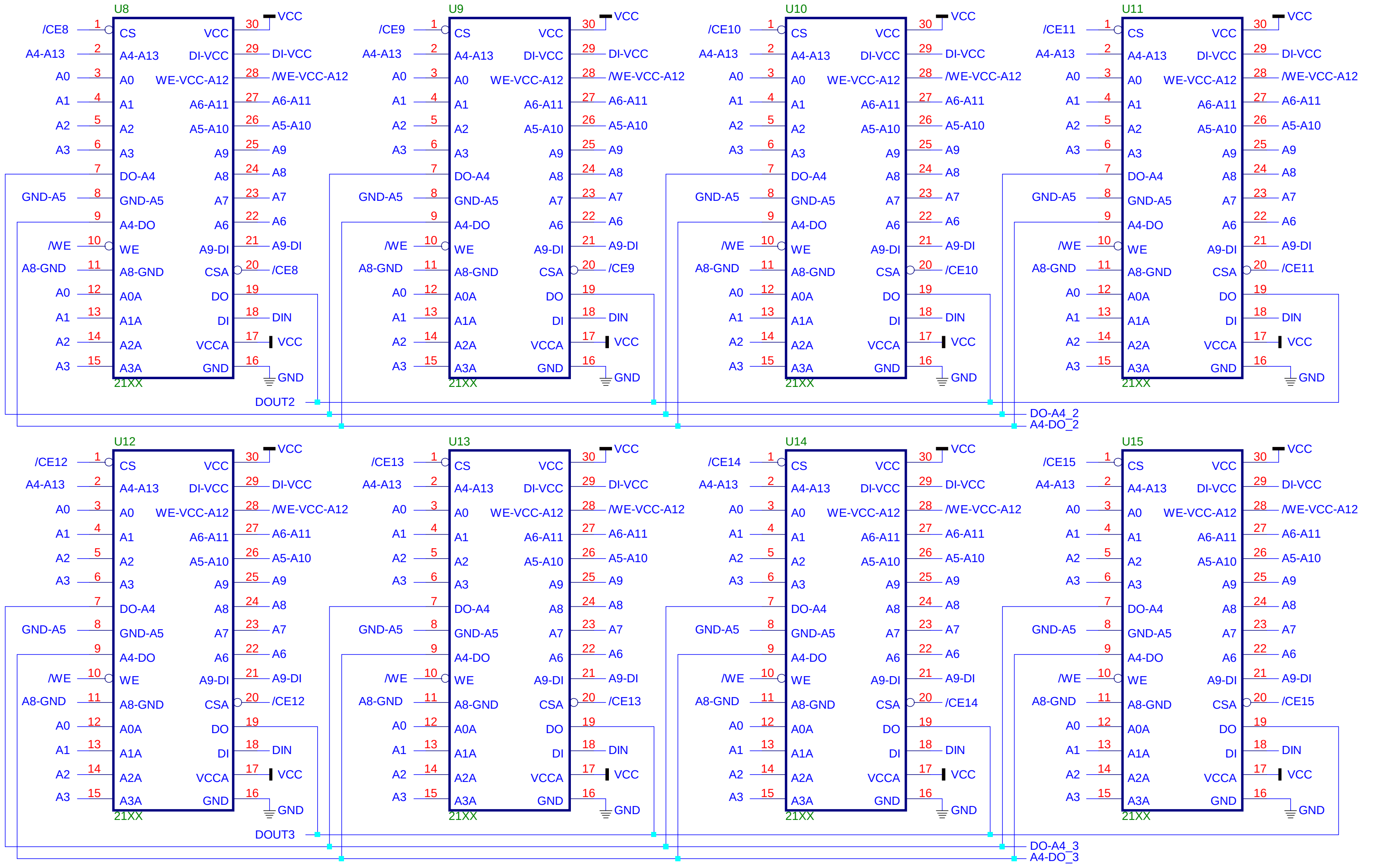
Small World Communications	Project: EDUC-8ME
6 First Avenue	Macro: E8ME/M1
Payneham South SA 5070	Date: 1/22/22
Date Last Modified: 6/29/24	



2102 1Kx1 SRAM. Fit 16-pin DIP to pins (8-15) & (23-16)
2125 1Kx1 SRAM. Fit 16-pin DIP to pins (1-8) & (30-23)
2147 4Kx1 SRAM. Fit 18-pin DIP to pins (3-11) & (28-20)
2167 16Kx1 SRAM. Fit 20-pin DIP to pins (2-11) & (29-20)

2125 compatible with TTL 93425 and CMOS 6508.

Small World Communications	Project: EDUC-8ME
6 First Avenue	Macro: E8ME/M2
Payneham South SA 5070	Date: 1/22/22
Date Last Modified: 6/7/24	



Small World Communications	Project: EDUC-8ME
6 First Avenue	Macro: E8ME/M3
Payneham South SA 5070	Date: 1/22/22
Date Last Modified: 6/7/24	