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

by
Steven S. Pietrobon, Ph.D.
13 March 2023.

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 input and two output ports. To increase its usability, Jamieson Rowe in his EDUC–8 book gave a schematic that added 16 input and 16 output ports up to a maximum of 128 input and 128 output 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 E8/CE) and adding a new horizontal board (E8/B) to the bottom of E8/CE. This new board buffers various signals from EDUC–8 to up to eight EDUC–8ME units. Each EDUC–8ME adds eight serial input ports, eight serial output ports and 8KB of static RAM. With eight EDUC–8ME units this allows an expansion of up to 64 serial input ports, 64 serial output ports and 64KB of RAM.

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 (E8M/C), decoder (E8M/D), I/O port multiplexer (E8M/I) and memory (E8M/M). The decoder provides the interface to EDUC–8 and the other boards. The multiplexer provides the interface to the eight input/output (I/O) ports. Up to four memory boards can be used, each containing up to 16 1Kx1 static RAM chips. The design philosophy is to use technology available in 1974, that is 74 Series TTL and memories equivalent to the Intel 2102, such as the Signetics 2602B and Mostek MK4102P, as recommended by Jamieson Rowe in his EDUC–8 book. If desired, more modern technology such as 74LS series TTL and higher density RAMs can be used.

The memory expansion method used is similar to that in the Digital PDP–8. Each bank of memory addresses the full 256 bytes available to EDUC–8. Each EDUC–8ME has up to 32 banks for 8 KB of RAM. Two 8–bit registers are used to control access to which bank 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 bank 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 bank data. A multiplexer is used to select either the IDA or IFA to produce the instruction field (IF), which is used to select a bank of RAMs (consisting of two 1Kx1 memories). The lower 11 bits to the RAMs is used to serially access one byte of the 256 bytes in a bank. These address bits are provided from E8/M to E8/B.

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 IF. A three bit DIP switch is used to compare the EDUC–8ME unit number (0 to 7) with the top three bits of IF. If the values are the same then the appropriate memory, input port or output port is enabled.

Like 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 bank, it would also be convenient to 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 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 EDUC–8ME, this is limited to 64 I/O ports.

Buffer Board (E8/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 E8/CE and E8/B boards, as shown in schematic BUFFER1. 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 E8/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 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.

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

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.

$\overline{\text{EIFL}}$ and $\overline{\text{SDF}}$ 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 enable instruction field latch ($\overline{\text{EIFL}}$) 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

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

where $\overline{\text{S}}$ is the active low set input (which sets Q high and Q_B low), $\overline{\text{R}}$ is the active low reset input (which sets Q_B high and Q low), with Q_B 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 BUFFER2. 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 $\overline{\text{EIFL}} = \overline{\text{Q.(JMS+JMP).T0}\cdot\text{5}}$, as shown in schematic BUFFER1. 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 $\overline{\text{EIFL}}$ to the EDUC–8ME units. At the end of EXECUTE for the JMS or JMP instruction, we need to reset Q using the signal $\overline{\text{(JMS+JMP).T22}\cdot\text{5}}$. We also need to reset Q at power up using the $\overline{\text{MR}}$ signal. That is we have

$$\text{Q} = \overline{\overline{\overline{\text{MB4}}.\overline{\text{MB3}}.\text{CLR.SHF.SKP.T13}}.\text{Q\_B}}$$
$$\text{Q\_B} = \overline{\overline{(\text{JMS}+\text{JMP}).\text{T22}\cdot 5}.\overline{\text{MR}}.\text{Q}}$$
$$\overline{\text{EIFL}} = \overline{\text{Q.(JMS+JMP).T0}\cdot 5}$$

For the two bank 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 bank. 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 $\overline{\text{EIFL}}$. That is

$$\text{P} = \overline{\overline{\text{FETCH.T22}\cdot 5.\overline{\text{MB7}}.\text{MB4}}.\text{P\_B}}$$

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

$$\text{P\_B} = \overline{\overline{\text{EXECUTE.T22}\cdot 5}.\overline{\text{MR}}.\text{P}}$$

We thus generate the select data field ($\overline{\text{SDF}}$) active low signal using a 7437 buffer during EXECUTE and if P is high. That is

$$\overline{\text{SDF}} = \overline{\text{EXECUTE.P}}$$

The logic for generating $\overline{\text{EIFL}}$ and $\overline{\text{SDF}}$ is shown in schematic BUFFER1. Signals MB3, MB4 and MB7 are obtained from the E8/M memory board. Signals $\overline{\text{T0·5}}$, T22·5, T13, FETCH, EXECUTE and $\overline{\text{MR}}$ are obtained from the E8/T timing board.

Note that E8/T needs to be modified so that $\overline{\text{MR}}$ is connected to unused pin 25 of the edge connector. Unfortunately, the 7410 4–input NAND gate used for the output of $\overline{\text{MR}}$ is already fully loaded (in fact, it is overloaded!). It drives the $\overline{\text{MR}}$ 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 $\overline{\text{C}_\text{D}}$ inputs of two 7473 dual JK FFs with 2 UL for each $\overline{\text{C}_\text{D}}$ 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 E8/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 E8/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. E8/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 $\overline{\text{DATA}}$ for output, DATA for input, $\overline{\text{CLOCK}}$, $\overline{\text{FLAG\_RESET}}$ and $\overline{\text{FLAG}}$ signals, we generate two custom ports to read and write the bank address to the four ME registers. As we don't need to read a flag, the $\overline{\text{FLAG}}$ signal is omitted. However, we do need $\overline{\text{FLAG\_RESET}}$ 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. $\overline{\text{EIFL}}$ 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 $\overline{\text{CLR}}$ the input port. Data is output during T2–9 from the AC_BIT_0 signal. For the output port we have

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

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 BUFFER2 schematic.

The MB3 and MB4 signals are buffered using 7437 gates to MB3_O and MB4_O, 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 MB3_O and MB4_O high using the OPR_IOT signal. This is equivalent to selecting the IFA register. The logic for this is shown in the BUFFER1 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. For $\overline{\text{CLR\_IOT\_FLAG}}$, $\overline{\text{IOT\_SHIFT}}$ and SKP_ON_IOT_FLAG these signals are diverted to E8/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 $\overline{\text{MR}}$ need to increase their output load from the standard 10 ULs. All the other signals satisfy the requirement of 10 or less ULs.

Signal Name	E8/T	E8/D	E8/M	E8/P	E8/A	E8/IOT	E8/F	E8/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
$\overline{\text{MR}}$	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

$\overline{\text{WE}}$ 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 (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. The minimum delay from MCPB to the clock input for PC0 is 22×0.48 (7400 LH) = 10.56 ns. 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. 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 E8/B and 7404 driver in E8M/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 $\overline{\text{WE}}$ 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 $\overline{\text{WE}}$ and $\overline{A_2}$, B_1, B_2 and $\overline{C_D}$ are connected to logic high. On the falling edge of $\overline{\text{WE}}$ 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 minimum 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 $\overline{\text{WE}}$ 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 $\overline{\text{WE}}$ 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 $\overline{\text{WE}}$ 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 $\overline{\text{WE}}$ pulse generation is shown in schematic BUFFER2. To adjust the variable resistor, move the jumper so that $\overline{\text{A1}}$ is connected to MCPB. The cable to the E8/M board and any EDUC–8ME units 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 E8/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 one of the two memory chip locations in E8/M, which are effectively pin locations A0 to A9, $\overline{\text{CS}}$, $\overline{\text{WE}}$, DIN, DOUT and GND. As $\overline{\text{CS}}$ is connected to $\overline{\text{MEMORY\_ENABLE.}\overline{\text{MA7}}}$ in the lower memory position and $\overline{\text{CS}}$ = MA7 when MEMORY_ENABLE is high, we let A10 = $\overline{\text{CS}}$. Note that the upper memory position could also be used where $\overline{\text{CS}} = \overline{\text{MA7}}$ when MEMORY_ENABLE is high. We also need to access MEMORY_ENABLE from E8/D, in order to generate the chip select signals in E8M/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 X3 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_B to A10_B, DIN_B, WE and ME_B to the EDUC–8ME units as shown in schematics BUFFER2 and BUFFER1. For pin DOUT was use a 7401 open collector gate to connect to the D–BUS. The inputs to the gate are DOUT_B from the EDUC–8ME units and MEMORY_ENABLE, to ensure the output is active only while the RAMs are being accessed. Signal DOUT_B is actually an open collector bus and thus requires a pullup resistor R6. The minimum and maximum pullup resistor values (in K or kΩ) are

$$R_{\text{min}} = \frac{V_{\text{CC,max}} - V_{\text{OL}}}{I_{\text{OL}} - 1.6U_{\text{IL}}} \qquad R_{\text{max}} = \frac{V_{\text{CC,min}} - V_{\text{OH}}}{N_{\text{O}}I_{\text{OH}} + 0.04U_{\text{IH}}} \tag{1}$$

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 low 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 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

$$R_{\text{min}} = \frac{4.85}{16 - 1.6N_{\text{IL}}} \qquad R_{\text{max}} = \frac{2.35}{0.25N_{\text{O}} + 0.04N_{\text{IH}}} \tag{2}$$

where N_{IL} and N_{IH} is the number of standard 1.6 mA input low and 0.04 mA input high units, respectively. For 74LS series TTL we have

$$R_{\text{LS,min}} = \frac{4.75}{8 - 0.4N_{\text{IL}}} \qquad R_{\text{LS,max}} = \frac{2.05}{0.1N_{\text{O}} + 0.02N_{\text{IH}}} \tag{3}$$

where N_{IL} and N_{IH} is the number of standard 1.6×0.5 = 0.8 mA input low and 0.04×0.25 = 0.01 mA input high units, respectively.

With up to $N_{\text{O}} = 8$ standard open collector outputs and $N_{\text{IL}} = N_{\text{IH}} = 1$ standard input for DOUT_B this gives $R_{\text{min}} = 337\ \Omega$, $R_{\text{max}} = 1152\ \Omega$, $R_{\text{LS,min}} = 625\ \Omega$ and $R_{\text{LS,max}} = 2500\ \Omega$. The average values are 744 Ω and 1562 Ω, for 74 and 74LS, respectively. Choosing the closest standard resistor value this gives R6 = 680R and 1.5K, for 74 and 74LS series TTL, respectively.

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

$$R_{\text{max}} = \frac{V_{\text{CC,min}} - V_{\text{OH}}}{0.04U_{\text{IH}}} = \frac{4.75 - 2.4}{0.04N_{\text{IH}}} = \frac{58.75}{N_{\text{IH}}}. \tag{4}$$

All the inputs have $N_{\text{IH}} = 1$, except for $\overline{\text{CD}}$, which has $N_{\text{IH}} = 2$. The total value is $N_{\text{IH}} = 15+3+2 = 20$, which gives $R_{\text{max}} = 2.938\text{K}$. Averaging with $R_{\text{min}} = 1\text{K}$, gives $R_{\text{av}} = 1.969$ 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. In this case we have $N_O = 1$ and $N_{IL} = N_{IH} = 2$ 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. We also give the values for R6.

Table 1. E8/B Pullup Resistor Values

			74 series TTL				74LS series TTL			
	N_O	N_I	R_{min} (Ω)	R_{max} (Ω)	R_{av} (Ω)	R_{rec}	R_{min} (Ω)	R_{max} (Ω)	R_{av} (Ω)	R_{rec}
R2	1	2	379	7121	3750	3.9K	660	14643	7651	8.2K
R3	1	1	337	8103	4220	3.9K	625	17083	8854	8.2K
R6	8	1	337	1152	744	680R	625	2500	1562	1.5K

Decoder Board (E8M/D)

DECODER1

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_B, ME_OUT_DATA_B and ME_OUT_RESET_B, where $\overline{\text{CLK}}$, $\overline{\text{DATA}}$ and $\overline{\text{RST}}$, 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 DECODER1, 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 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_B and AB to IFA during the low to high edge of $\overline{\text{EIFL}}$. 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

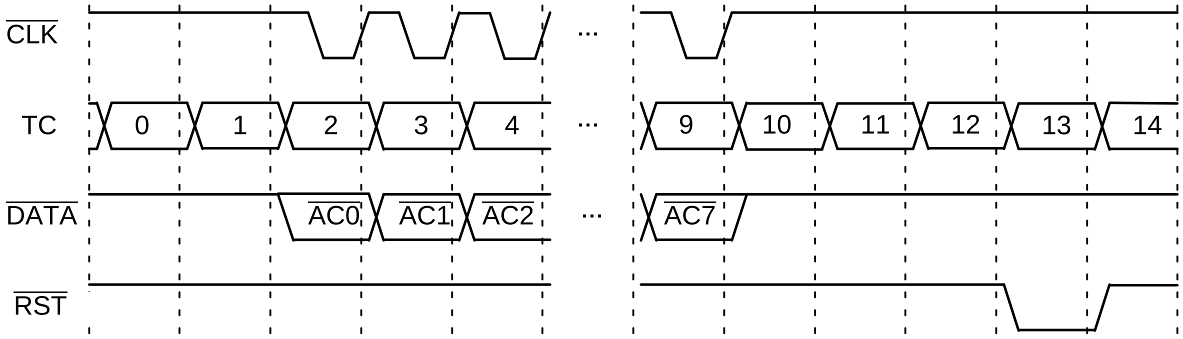


Figure 1: MEA Output Timing

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 (MB4,MB3). The exception is instruction 607 (write IFA) where MEA write is not performed since $\overline{\mathrm{EIFL}}$ is used to write AB to IFA. An MEA read is also performed for 61x and 63x instructions (read or write to port 1). 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 $\overline{\mathrm{EIFL}}$ goes low at the start of EXECUTE following a JMP or JMS instruction) and MEA read of the 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 (0,0).

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

The bus for the register that is read is called IF, which goes both to the MEA read shift register and to the extended memory. By using the same IF read address, we reduce complexity since we don't need separate multiplexers for an MEA read and for addressing the extended memory. As MB3_O = $\overline{\mathrm{MB3.OPR\_IOT}}$ and MB4_O = $\overline{\mathrm{MB4.OPR\_IOT}}$, we implement the signals as MB3A = $\overline{\mathrm{MB3\_O}}$ and MB4A = $\overline{\mathrm{SDF\_B.MB4\_O}}$, as shown in schematic DECODER1.

One half of a 9321 or 74LS139 dual 1 of decoder is used to generate EIDL_B (enable input device latch active low), EDFL_B (enable data field latch active low) and EODL_B (enable output device latch active low) from MB3A, MB4A and ME_OUT_RESET_B. These signals go to the clock input of the IDA, DFA and ODA registers where the data from AB is written during the rising edge of EIDL_B, EDFL_B and EODL_B, respectively. This is shown in schematic DECODER 1 for IDA and ODA and DECODER2 for DFA.

The signal MR_B 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 EDUC8, without having to set up the four MEA registers. Instead of two inverters a spare 7408 AND gate is used.

The three most significant bits of IF, that is IF[7:5] are used to select one of eight EDUC–8ME units. The IDA address is in the form (IDA7,IDA6,IDA5,0,0,IDA3,IDA1,IDA0) with the three most significant bits used for selecting the EDUC–8ME unit, like that used for IFA and DFA. The ODA has a similar form. A DIP switch and three 74136 exclusive or (XOR) open collector gates are used to generate the IFE (IF enable) signal. We have IFE = (S2⊕IF7).(S1⊕IF6).(S0⊕IF5) where (S2,S1,S0) is the inverse of the EDUC–8ME unit selected. For example for the first unit (number 0) we have (S2,S1,S0) = (1,1,1), which is selected by having the DIP switches in the open position. The next unit (number 1) would be (1,1,0), etc.

DECODER2

In order to externally program the separate banks and for running programs, we need to separately initialise the IFA and DFA registers. We perform this using the switch register (SR) from EDUC–8 and the load address signal (called ME_EXT_LA_B) from the front panel of the first EDUC–8ME unit. As shown in DECODER2, two 74157 quad 2 to 1 multiplexers are used to either select AB or SR for input to IFA and DFA.

Two 7408 two input AND gates are used to select either ME_EXT_LA_B or $\overline{\mathrm{EIFL}}$ or $\overline{\mathrm{EIDL}}$ for the clock inputs of IFA and DFA, respectively. As ME_EXT_LA_B goes to up eight EDUC–8ME units, this signal is first buffered using a spare AND gate. In order to meet the data hold requirements of the IFA and DFA registers, ME_EXT_LA_B is delayed using a spare 74136 XOR gate and 7404 inverter. The delayed signal is then used to select SR when low or AB when high.

Let $\overline{\text{LA}}$ be the buffered ME_EXT_LA_B signal. The maximum delay of $\overline{\text{LA}}$ to the clock input plus the data hold time is 7408 LH + 74273 (D hold) = 27+5 = 32 ns. The minimum delay of $\overline{\text{LA}}$ to the data inputs is 74136 HL + 7404 LH + 74157 (S to Z) = 55×0.35+22×0.48+min(23×0.48, 27×0.35) = 29.81+min(11.04,9.45) = 39.26 ns, which satisfies the requirement of being greater than 32 ns. To further ensure that $\overline{\text{LA}}$ is sufficiently delayed we add a 15 pF load capacitor to the output of the open collector XOR gate.

The other half of the 9321/74LS139 1 of 4 decoder is used to generate four board select signals CS0_B to CS3_B. This uses signals IF[4:3] to select one of the four boards and enable input $\overline{\text{MEMORY\_ENABLE.IFE}} = \overline{\overline{\text{ME\_B}}\text{.IFE}}$ to select the EDUC–8ME unit when IFE is high. Signals A0_B to A10_B, WE and DIN_B are buffered using 13 7404 inverters to generate the signals A0 to A10, WE_B and DIN. Signals IF[2:0] and A10 are to select one the 16 RAMs on each memory board.

For A0 to A9, WE_B and DIN, the 2102 1Kx1 RAM has an input current of 10 μA. With 4×16 = 64 2102 chips in total, that is a total input current of 640 μA. The 7404 has an output high current of 20 UL = 20×40 = 800 μA which is sufficient. However, the 74LS04 is 10 UL = 400 μA and is not sufficient. Thus, the 74LS04 must not be used for these 12 signals. The 74LS37 used to drive these signals from E8/B has a 30/15 UL output, sufficient to drive eight EDUC–8ME units with 7404 loads.

A 7438 NAND open collector buffer is used to output DOUT from the memory boards to the DOUT_B bus. IFE is used to select the output. No pullup is required on DOUT, since the outputs from E8/M are tristate. As the SR register will be driving additional loads, we need to check that the 10K pullup resistors used on E8/F in EDUC–8 are not too high. The maximum number of loads is $\lfloor(4.75-2.4)/(0.04\times10)\rfloor = \lfloor 58.75/10 \rfloor$ = 5 for 74 series TTL and $\lfloor(4.75-2.7)/(0.02\times10)\rfloor = \lfloor 102.5/10 \rfloor$ = 10 for 74LS. With one load in EDUC–8, and eight loads for EDUC–8ME, the total load is 9. This implies that the 10K pullup will work with 74LS, but with only four EDUC–8ME units with 74 series TTL. With R_{max} = 58.75/9 = 6.528 kΩ, R_{min} = 1 kΩ and R_{av} = 3.764 kΩ we recommend using R_{rec} = 3.9K resistors to replace the 10K SR pullup resistors on E8/F.

DECODER3

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

During T1, IF[7:0] is asynchronously loaded into Q[0:7] of the internal shift register using the active low input of ME_IN_RESET_B. Since the output is from Q7 and the input is in reverse order, the first bit to be output is IF0. On the rising edge of ME_IN_CLOCK_B (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_B to shift data out, as this would cause the first bit to be shifted into AC being IF1, instead of IF0, with the last bit being a 1, as DS (serial input in) is high. A 7438 buffer is used to output the data via the DIP switch in DECODER1. As

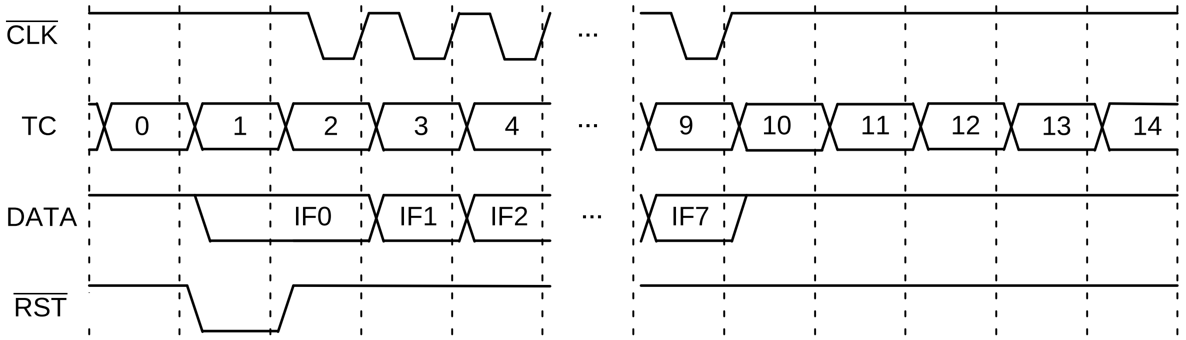


Figure 2: MEA Input Timing

only one EDUC–8ME unit is needed for an MEA read, the DIP switch is on for the first unit and off for the other units.

Lastly, half of a 9321 1 of 4 decoder is used to decode MB3A and MB4A and passed onto the front panel board. Signals IFA_B, DFA_B, IDA_B and ODA_B should be passed to the AND_B, TAD_B, ISZ_B and DCA_B inputs of E8/F, respectively. Again, this circuit is only used on the first EDUC–8ME unit.

Pullup Values

The following table shows the recommended values for the various pullup resistors used in the decoder board. For R2, R8 and R11, the values were calculated using higher output low values of 30 and 15 UL for 7438 and 74LS38, respectively. There are seven inputs tied high which gives R_{max} = 58.75/7 = 8.393K. Averaging with R_{min} = 1K gives R_{av} = 4.696K and R1 = R_{rec} = 4.7K. For 74LS, R1 = 0 Ω. However, if using a Fairchild 74LS136 then the 0 Ω R10 should be replaced with a 10K resistor, as emitter instead of diode inputs are used. For DIP switch outputs S0, S1 and S2, 10 K pullup resistors are used for R4, R5 and R6, respectively.

Table 2. E8M/D Pullup Resistor Values

			74 series TTL				74LS series TTL			
	N_{O}	N_{I}	R_{min} (Ω)	R_{max} (Ω)	R_{av} (Ω)	R_{rec}	R_{min} (Ω)	R_{max} (Ω)	R_{av} (Ω)	R_{rec}
R2	1	6	126	4796	2461	2.7K	220	9318	4769	4.7K
R7	3	6	758	2374	1566	1.5K	848	4881	2865	2.7K
R8,R11	1	1	105	8103	4104	3.9K	201	17083	8642	8.2K
R9	1	1	337	8103	4220	3.9K	625	17083	8854	8.2K

Input/Output Port Multiplexer Board (E8M/I)

The inputs IF[2:0] are used to select the input or output port and IFE the EDUC–8ME unit selected. Excluding this board IF1 and IF2 both have a load of 1 UL for E8M/D and 8 UL for four E8M/M memory boards. IF0 has 1 UL for E8M/D and 4 UL for four E8M/M boards. For this board, IF[2:0] would have a load of 4 UL each. Thus, to prevent overloading, IF1 and IF2 are buffered using 7404 inverters. This gives total loads of 9 UL for IF0 and 10 UL each for IF1 and IF2. The inversion of IF1 and IF2 changes the order of the I/O ports selected. That is, ports 0, 1, 2, 3, 4, 5, 6 and 7 become 6, 7, 4, 5, 2, 3, 0 and 1, respectively.

A 74259 3 to 8 decoder is used to generate IO[7:0] from IF[2:0] and IFE. 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_B, OD1_RESET_B, OD1_DATA_B, ID1_CLOCK_B and ID1_RESET_B (from the EDUC8 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_B, ID1_DATA and ID1_FLAG_B lines from EDUC8. Link X2 is used to select the pullups for the first EDUC–8 unit. The table below gives the pullup resistor values. For R1 to R4, 7438 OC buffers are used with an output low of 30 UL or 15 UL for 74LS.

Table 3. E8M/I Pullup Resistor Values

			74 series TTL				74LS series TTL			
	N_O	N_I	R_{min} (Ω)	R_{max} (Ω)	R_{av} (Ω)	R_{rec}	R_{min} (Ω)	R_{max} (Ω)	R_{av} (Ω)	R_{rec}
R1–R3	8	1	105	1152	628	680R	201	2500	1351	1.5K
R4	1	8	433	4123	2278	2.2K	228	7885	4056	3.9K
R5	0	1	1000	58750	–	10K	–	–	–	0R

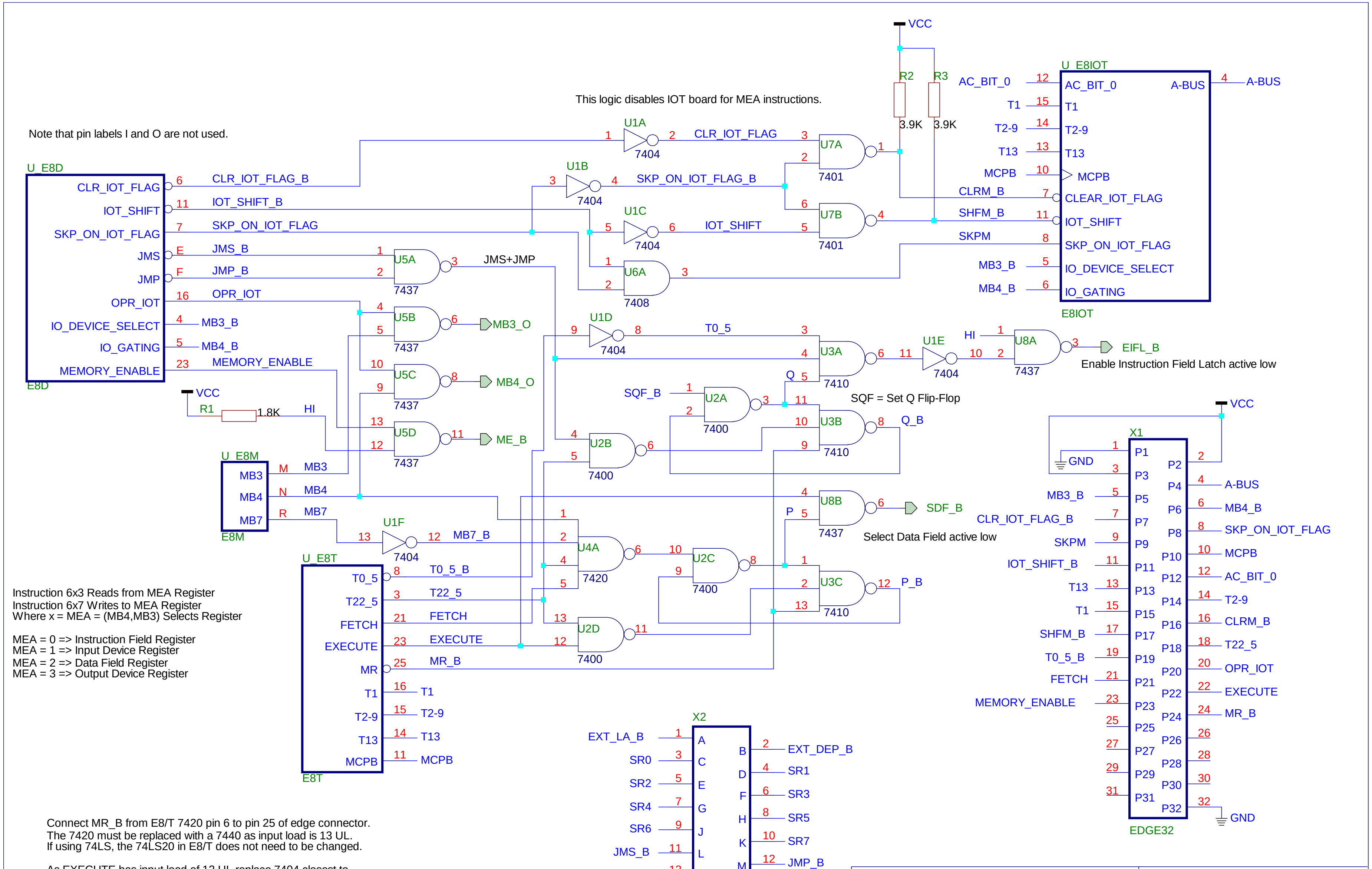
Memory Board (E8M/M)

To allow each memory board to be used in any location, a four way jumper X1 is used. For example, the first memory board would place a jumper across pins 1 and 2, connecting signal CS0_B to the enable input CS_B of the 74154 1 of 16 decoder. Signals IF[2:0] and A10 go to the address inputs of the decoder, to select one of the 16 memory chips when CS_B is low.

For the 2102, the output driving capability is limited to an output high of 2.2 V and I_{OH} = 100 μA and output low of 0.45 V and I_{OL} = 1.9 mA. The tristate leakage current is I_{LOH} = 10 μA high and I_{LOL} = 0.1 mA low. Driving a single input, the input high currents are I_{IH} = 40 μA 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 with one output high is N_H = (100–40)/10+1 = 7 and with 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_B goes low. Signals IF[2:1] are used to select the output, which is why these signals have an addition four UL compared to IF0.

Five inputs of the 74251 are tied high which results in R_{max} = 58.75/5 = 11.75 kΩ. With R_{min} = 1000 Ω and R_{av} = 6,375 Ω we choose R1 = 6.8K.



This logic disables IOT board for MEA instructions.

Note that pin labels I and O are not used.

Enable Instruction Field Latch active low

SQF = Set Q Flip-Flop

Select Data Field active low

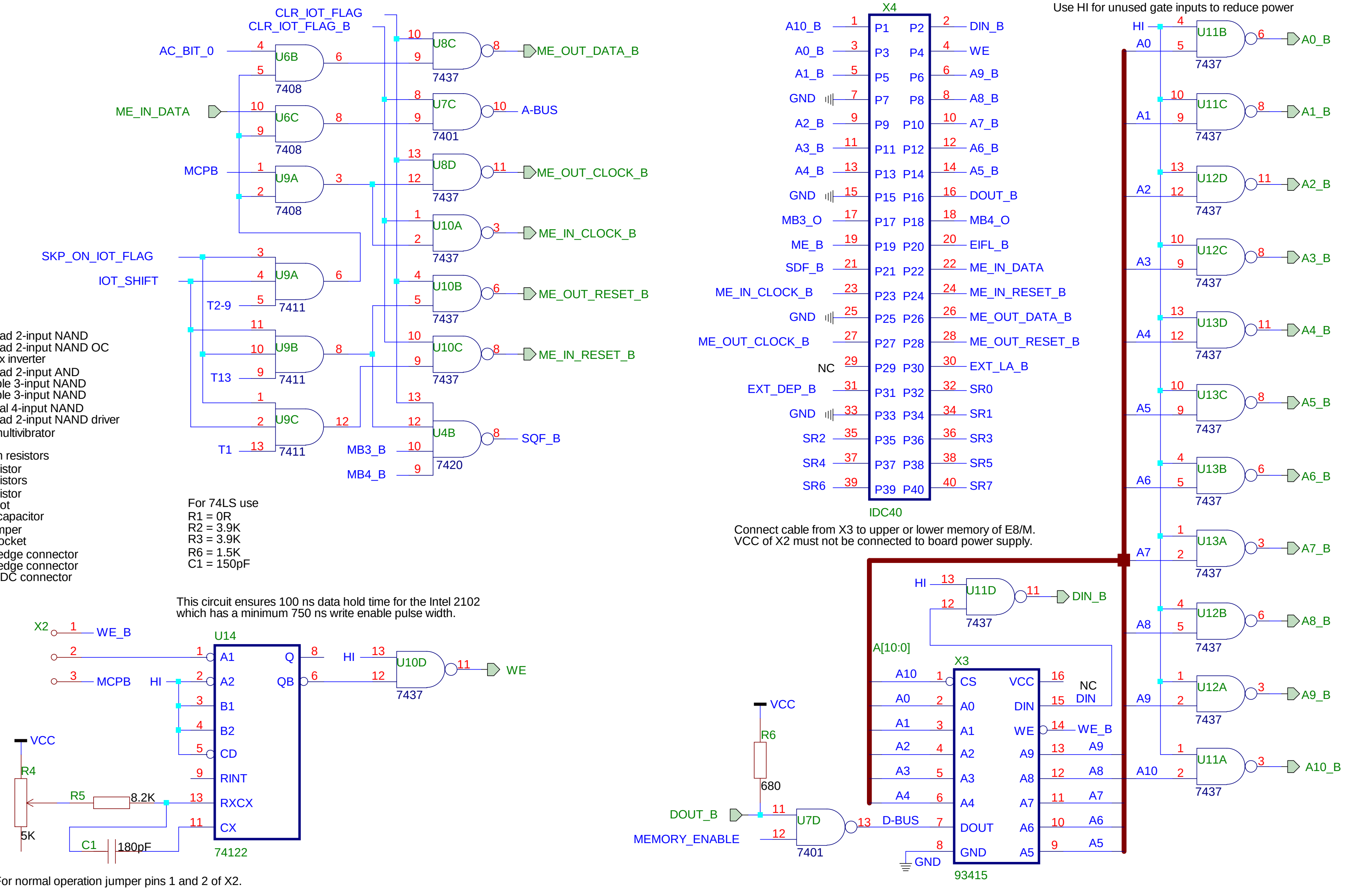
Instruction 6x3 Reads from MEA Register
Instruction 6x7 Writes to MEA Register
Where x = MEA = (MB4,MB3) Selects Register

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

Connect MR_B 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: EDUC8ME
6 First Avenue	Macro: BUFFER1
Payneham South SA 5070	Date: 1/22/22
Date Last Modified: 3/10/23	



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

1 680 ohm resistors
1 1.8K resistor
2 3.9K resistors
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 74LS use
R1 = 0R
R2 = 3.9K
R3 = 3.9K
R6 = 1.5K
C1 = 150pF

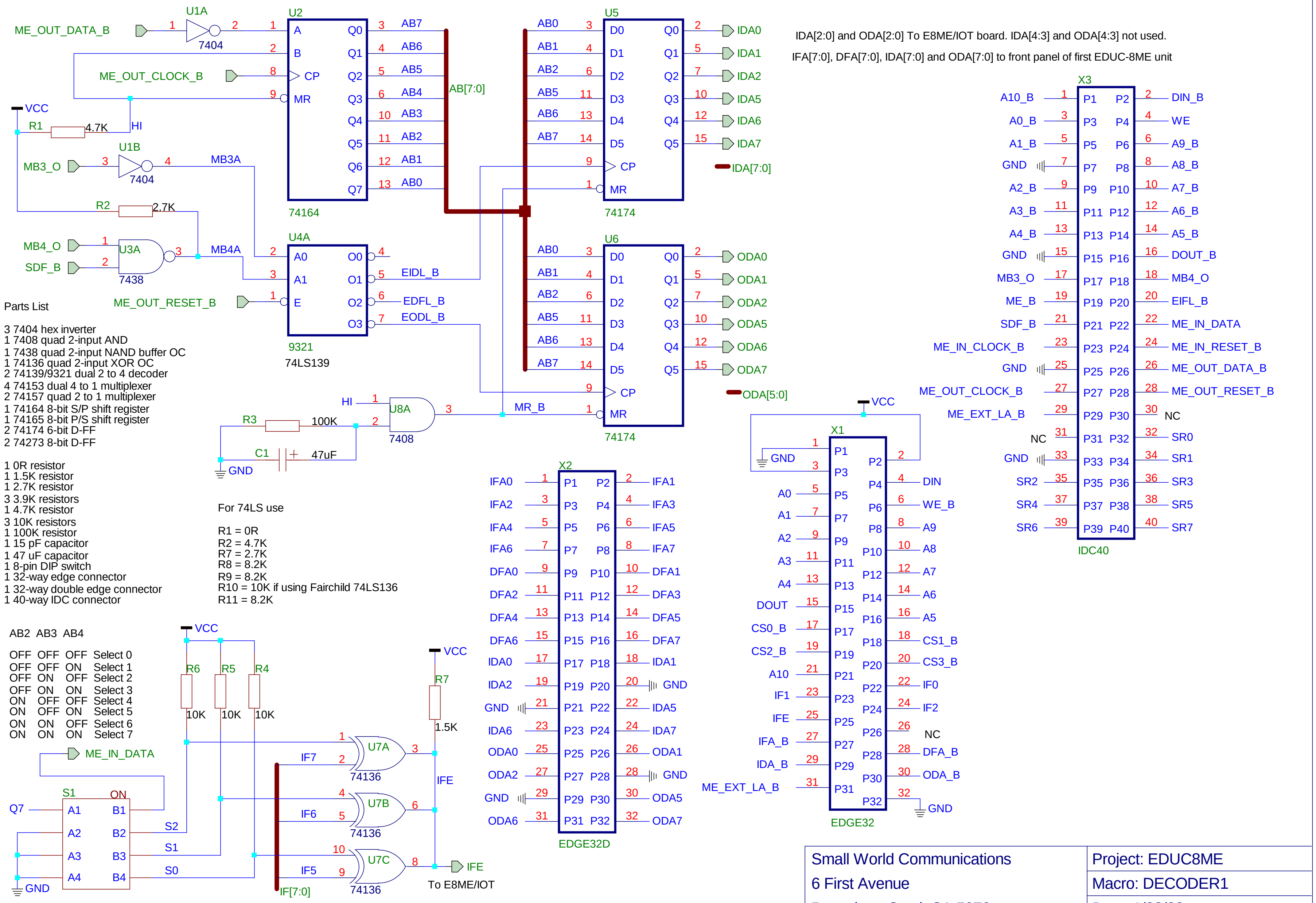
Connect cable from X3 to upper or lower memory of E8/M.
VCC of X2 must not be connected to board power supply.

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

For normal operation jumper pins 1 and 2 of X2.

To adjust WE pulse width, disconnect EDUC8ME 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: EDUC8ME
6 First Avenue	Macro: BUFFER2
Payneham South SA 5070	Date: 1/22/22
Date Last Modified: 3/13/23	



IDA[2:0] and ODA[2:0] To E8ME/IOT board. IDA[4:3] and ODA[4:3] not used.

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

Parts List

3 7404 hex inverter
1 7408 quad 2-input AND
1 7438 quad 2-input NAND buffer OC
1 74136 quad 2-input XOR OC
2 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

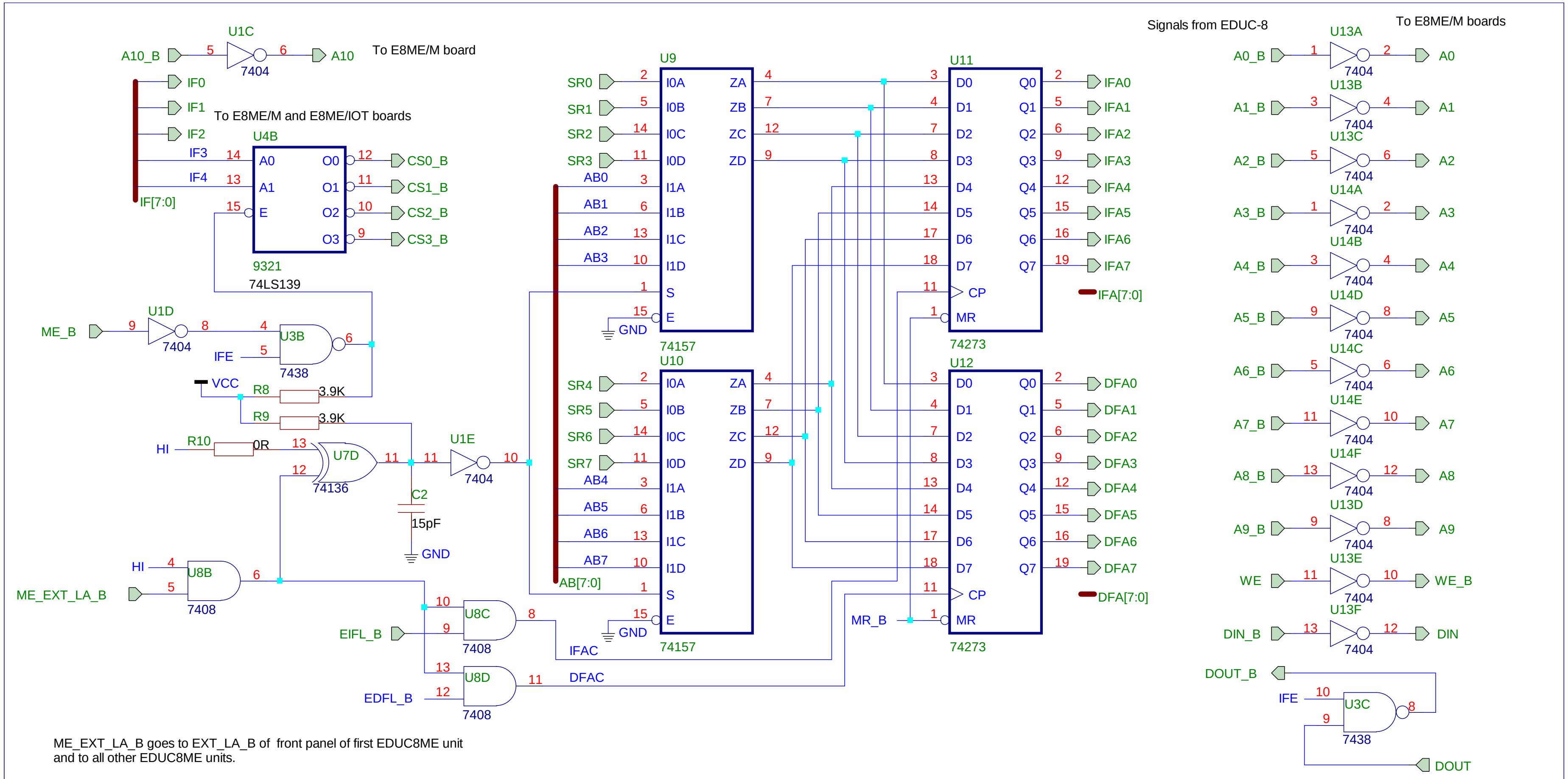
1 0R resistor
1 1.5K resistor
1 2.7K resistor
3 3.9K resistors
1 4.7K resistor
3 10K resistors
1 100K resistor
1 15 pF capacitor
1 47 uF capacitor
1 8-pin DIP switch
1 32-way edge connector
1 32-way double edge connector
1 40-way IDC connector

For 74LS use

R1 = 0R
R2 = 4.7K
R7 = 2.7K
R8 = 8.2K
R9 = 8.2K
R10 = 10K if using Fairchild 74LS136
R11 = 8.2K

AB2	AB3	AB4	
OFF	OFF	OFF	Select 0
OFF	OFF	ON	Select 1
OFF	ON	OFF	Select 2
OFF	ON	ON	Select 3
ON	OFF	OFF	Select 4
ON	OFF	ON	Select 5
ON	ON	OFF	Select 6
ON	ON	ON	Select 7

Small World Communications	Project: EDUC8ME
6 First Avenue	Macro: DECODER1
Payneham South SA 5070	Date: 1/22/22
Date Last Modified: 3/13/23	

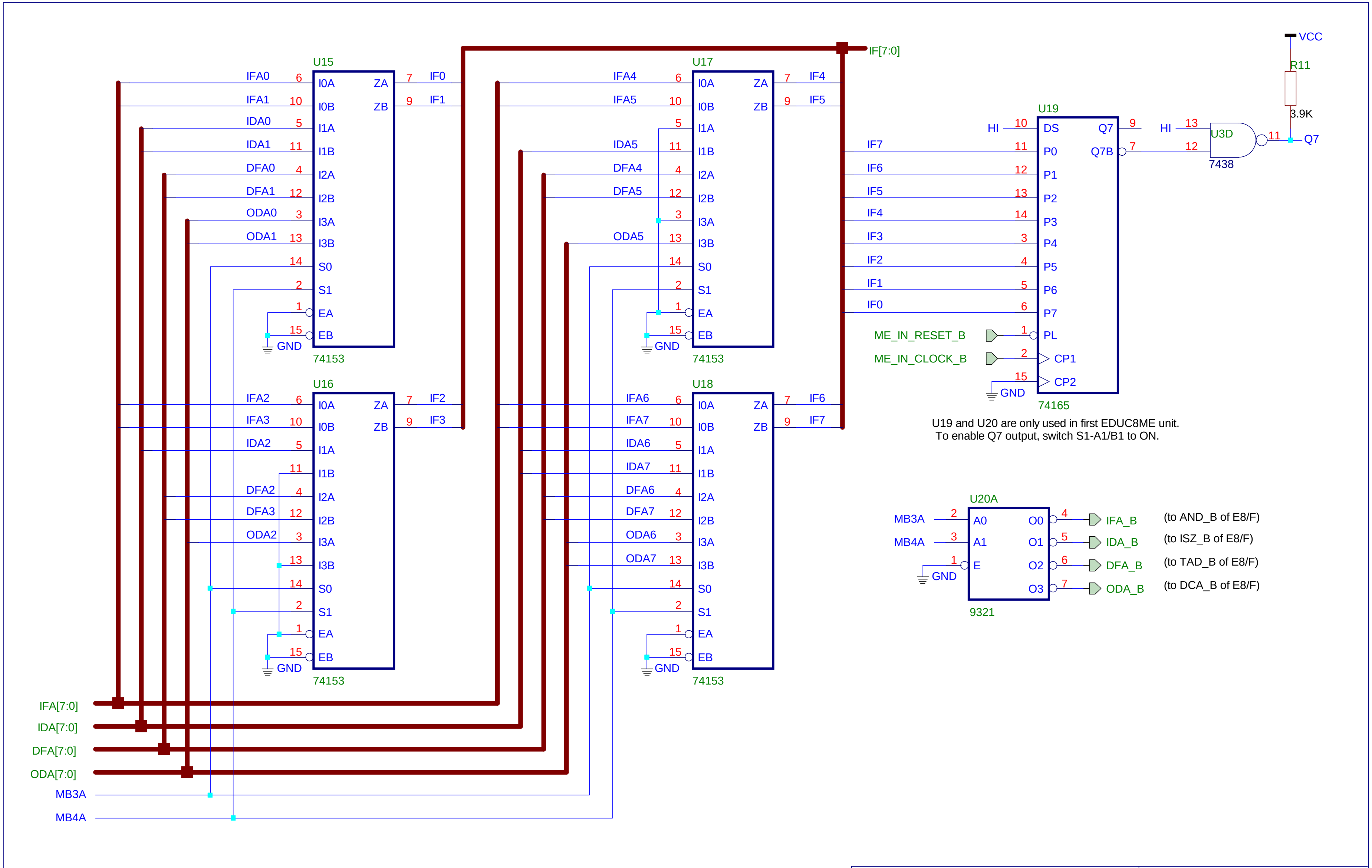


ME_EXT_LA_B goes to EXT_LA_B of front panel of first EDUC8ME unit
and to all other EDUC8ME units.

SR[7:0] is from EDUC8 and goes to all EDUC8ME units.
SR pullups of 10K on E8/F suitable for eight EDUC8ME circuits using 74LS157
or four EDUC8ME circuits using 74157.
Recommend replace EDUC-8 E8/F 10K SR pullups with 3.9K resistors.

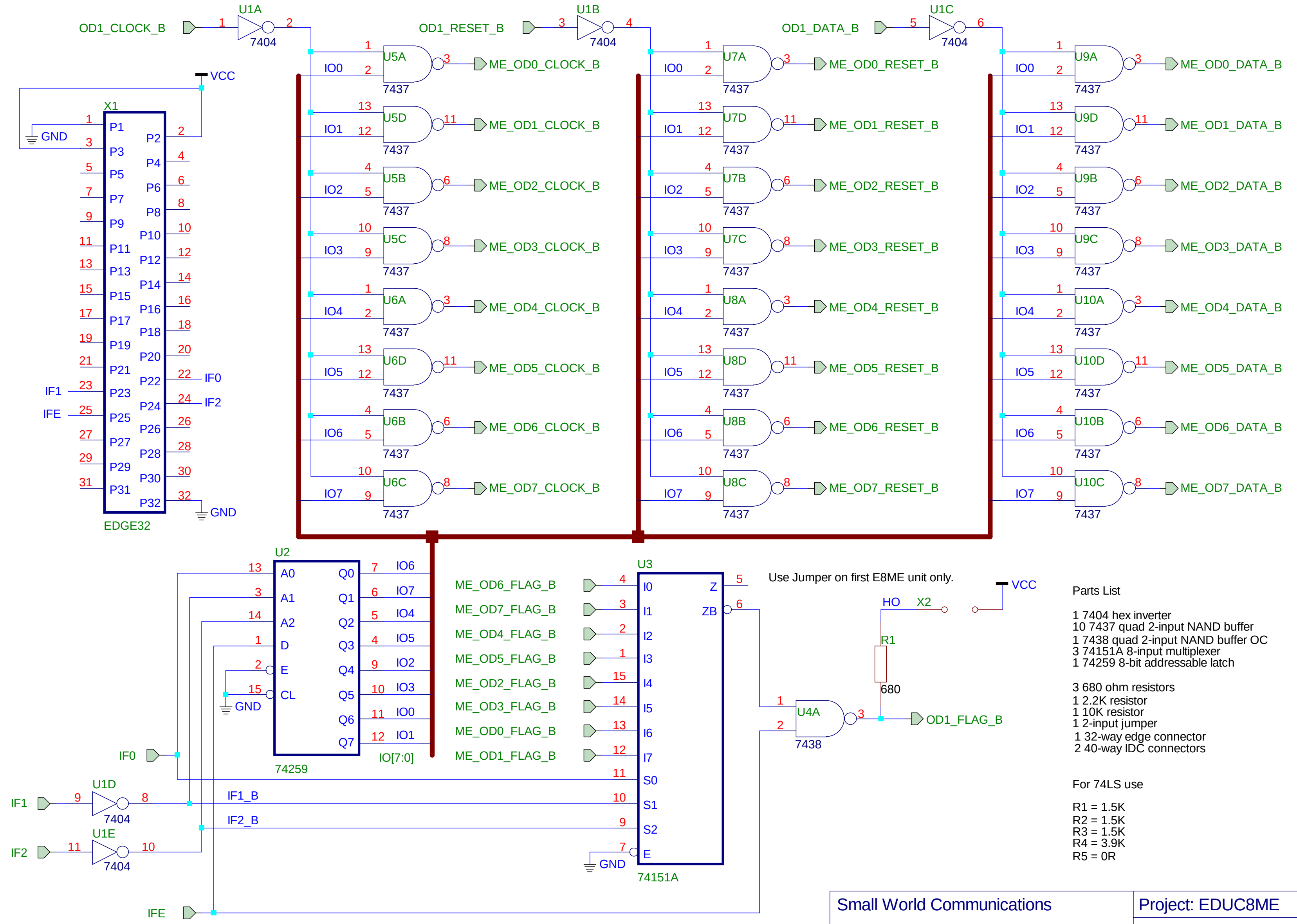
U16 and U17 must be 7404 to drive 64 RAM chips
2102 input current is 10 uA. Total = 640 uA.
7404 output high current is 800 uA.
74LS04 output high current is 400 uA.

Small World Communications	Project: EDUC8ME
6 First Avenue	Macro: DECODER2
Payneham South SA 5070	Date: 1/22/22
Date Last Modified: 3/10/23	



U19 and U20 are only used in first EDUC8ME unit.
To enable Q7 output, switch S1-A1/B1 to ON.

Small World Communications	Project: EDUC8ME
6 First Avenue	Macro: DECODER3
Payneham South SA 5070	Date: 1/22/22
Date Last Modified: 3/6/23	



Use Jumper on first E8ME unit only.

Parts List

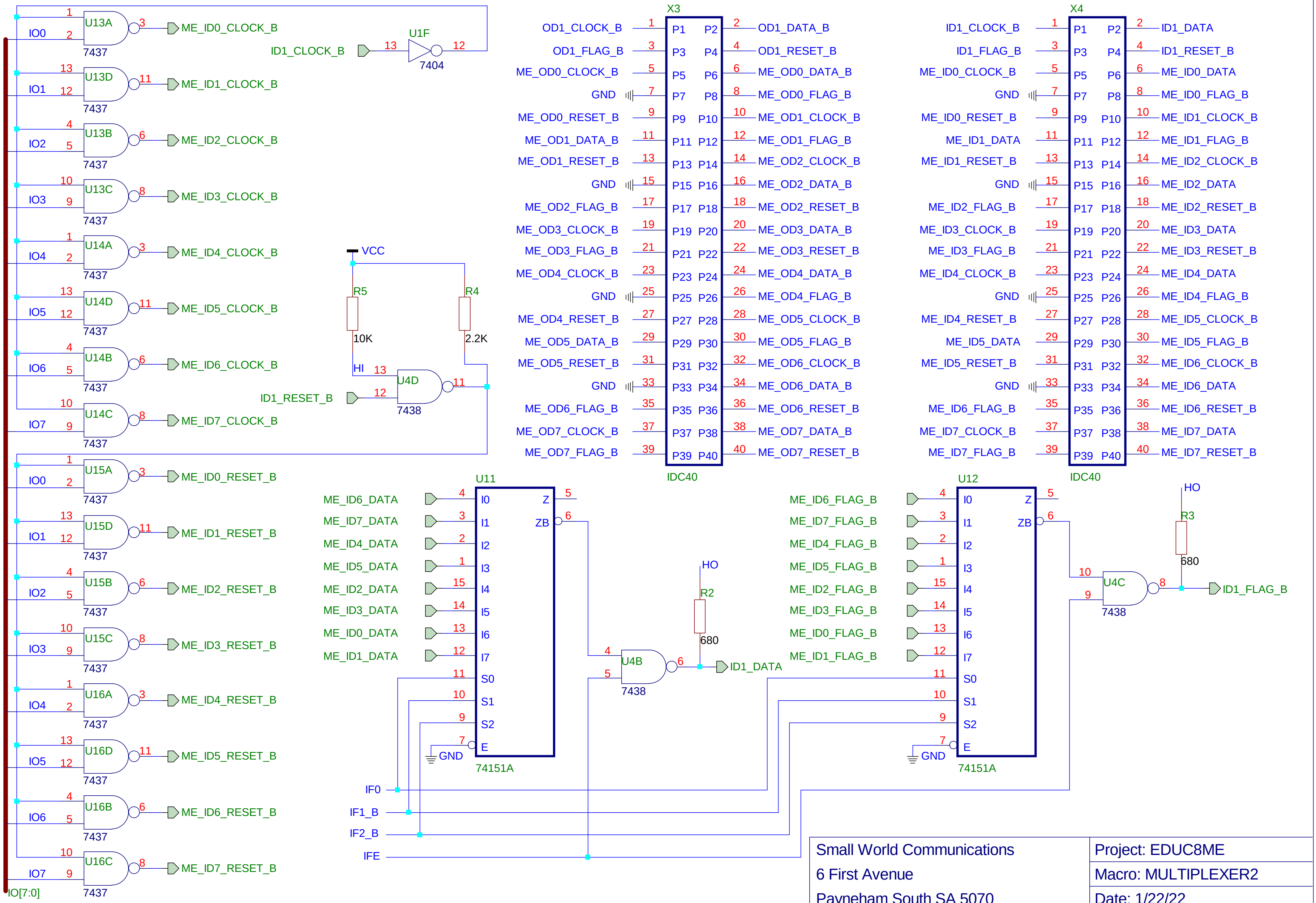
1 7404 hex inverter
10 7437 quad 2-input NAND buffer
1 7438 quad 2-input NAND buffer OC
3 74151A 8-input multiplexer
1 74259 8-bit addressable latch

3 680 ohm resistors
1 2.2K resistor
1 10K resistor
1 2-input jumper
1 32-way edge connector
2 40-way IDC connectors

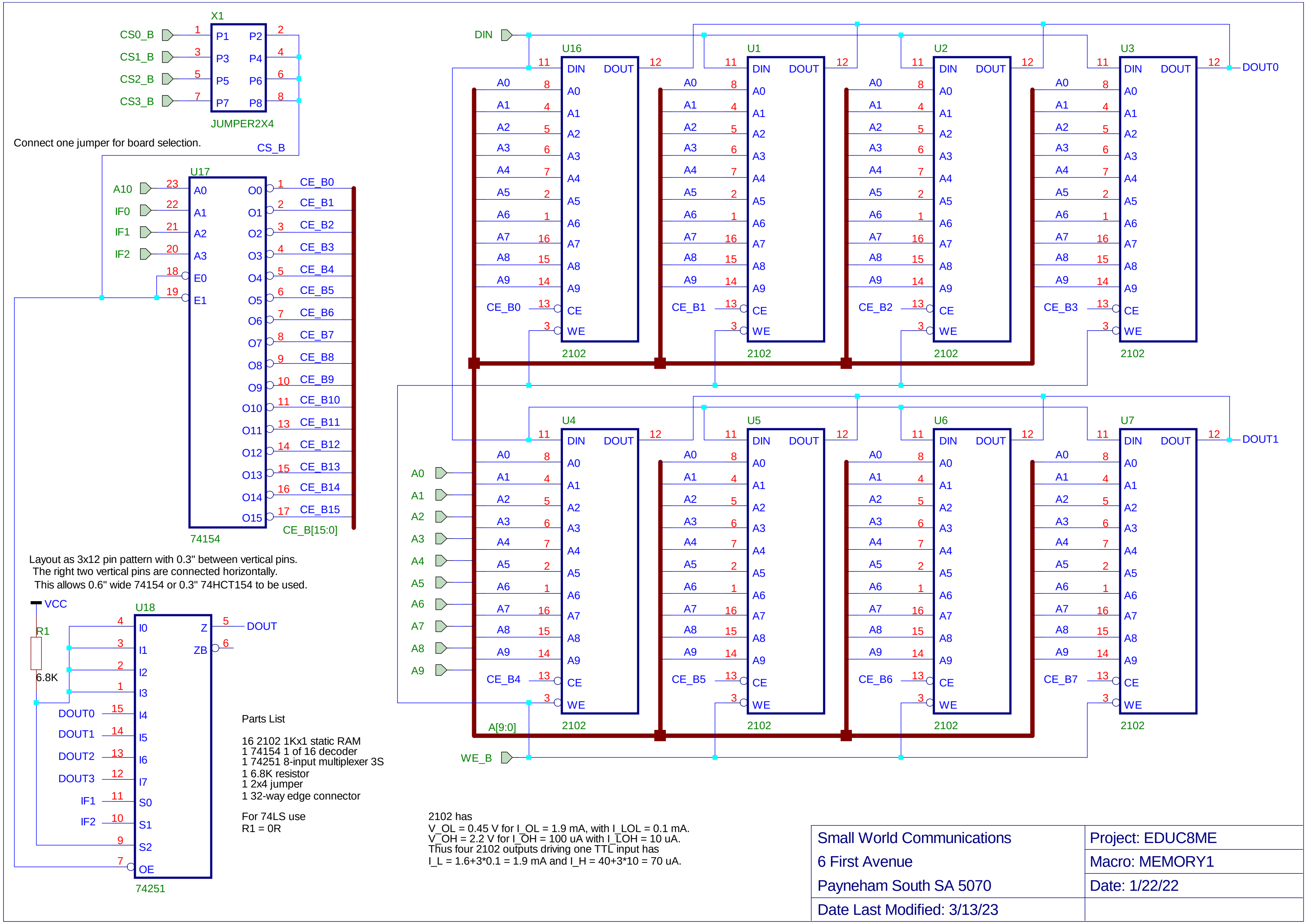
For 74LS use

R1 = 1.5K
R2 = 1.5K
R3 = 1.5K
R4 = 3.9K
R5 = 0R

Small World Communications	Project: EDUC8ME
6 First Avenue	Macro: MULTIPLEXER1
Payneham South SA 5070	Date: 1/22/22
Date Last Modified: 3/10/23	



Small World Communications	Project: EDUC8ME
6 First Avenue	Macro: MULTIPLEXER2
Payneham South SA 5070	Date: 1/22/22
Date Last Modified: 3/10/23	


X1
CS0_B
CS1_B
CS2_B
CS3_B
P1 P2 P3 P4 P5 P6 P7 P8
JUMPER2X4
CS_B
U17
A10
IF0
IF1
IF2
A0 A1 A2 A3 E0 E1
O0 O1 O2 O3 O4 O5 O6 O7 O8 O9 O10 O11 O12 O13 O14 O15
CE_B0 CE_B1 CE_B2 CE_B3 CE_B4 CE_B5 CE_B6 CE_B7 CE_B8 CE_B9 CE_B10 CE_B11 CE_B12 CE_B13 CE_B14 CE_B15
74154
CE_B[15:0]
DIN
U16 U1 U2 U3
U4 U5 U6 U7
DIN DOUT A0 A1 A2 A3 A4 A5 A6 A7 A8 A9 CE WE
2102
DOUT0
DOUT1
A[9:0]
WE_B
VCC
R1
6.8K
U18
I0 I1 I2 I3 I4 I5 I6 I7 S0 S1 S2 OE
Z ZB
DOUT
DOUT0 DOUT1 DOUT2 DOUT3
IF1 IF2
74251

Connect one jumper for board selection.

Layout as 3x12 pin pattern with 0.3" between vertical pins.
The right two vertical pins are connected horizontally.
This allows 0.6" wide 74154 or 0.3" 74HCT154 to be used.

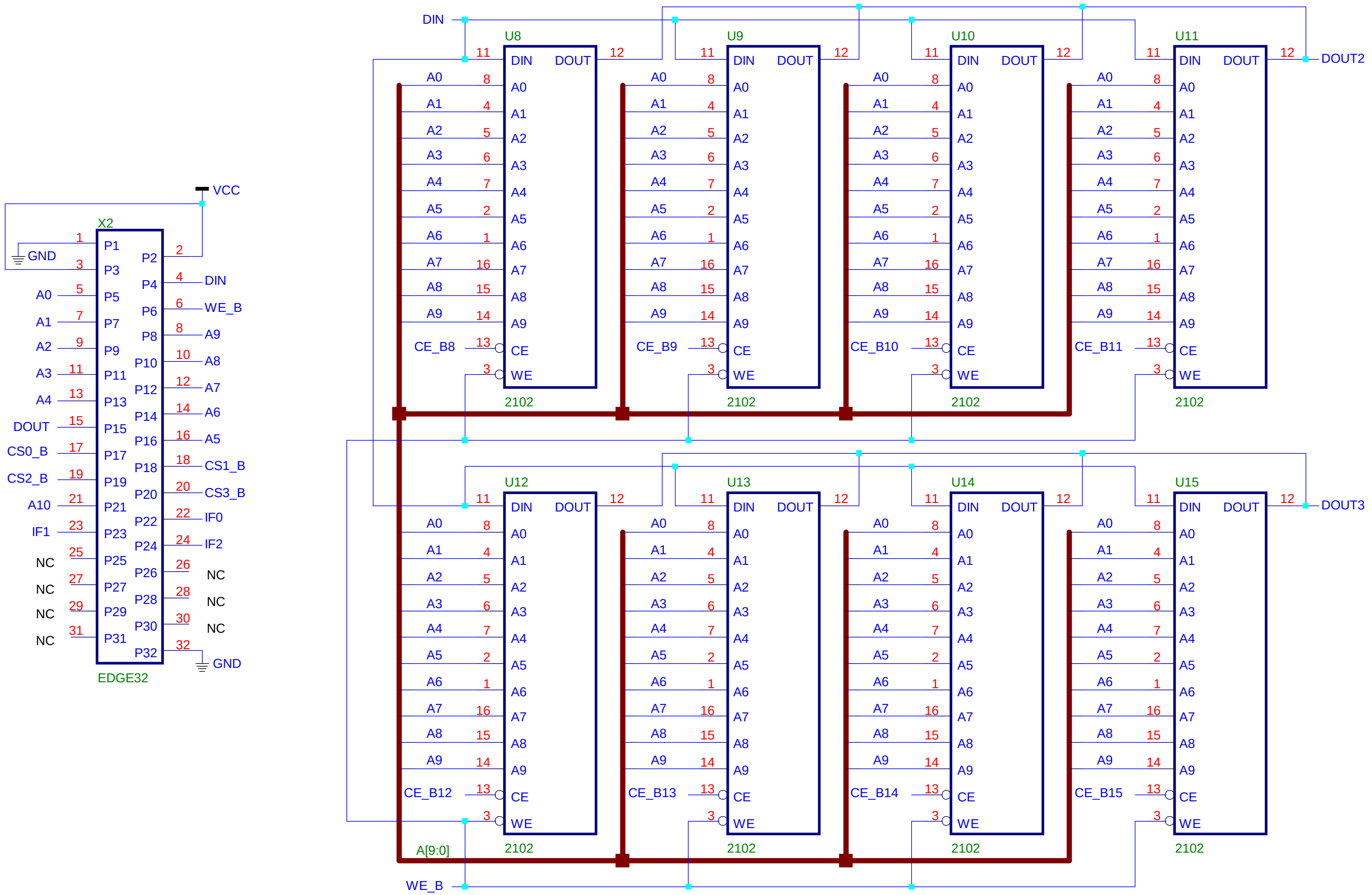
Parts List

16 2102 1Kx1 static RAM
1 74154 1 of 16 decoder
1 74251 8-input multiplexer 3S
1 6.8K resistor
1 2x4 jumper
1 32-way edge connector

For 74LS use
R1 = 0R

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.

Small World Communications	Project: EDUC8ME
6 First Avenue	Macro: MEMORY1
Payneham South SA 5070	Date: 1/22/22
Date Last Modified: 3/13/23	



Small World Communications
6 First Avenue
Payneham South SA 5070
Date Last Modified: 3/13/23

Project: EDUC8ME
Macro: MEMORY2
Date: 1/22/22