EM66xx IN-CIRCUIT EMULATOR

HARDWARE DESCRIPTION

Version 2.8

October 2005

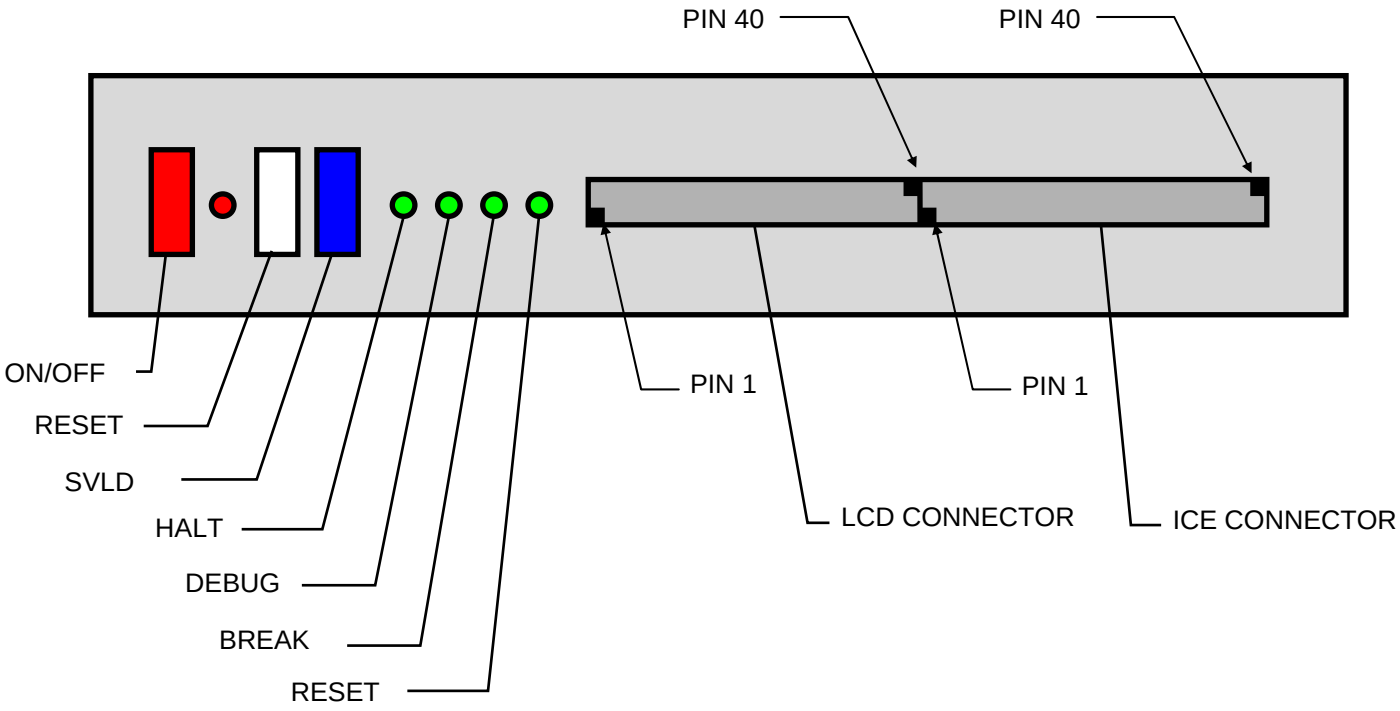
1. EM66xx Emulator hardware description

1.1. Warning

When connecting or disconnecting the EM66xx emulator to the target system or when connecting headers or LCD's to the connectors on the front panel it is essential that the emulator is powered off. Failure to do this could result in damage to the emulator hardware, which would require the system being returned to EM Microelectronic for repair.

Please also be aware that when Port pins are set to input, any voltage greater that the V I/O level that has been set (See Below) can also damage the emulator. This includes transitory switching spikes as well as sustained over voltage conditions. It is recommended that any high levels driven onto the emulator port pins for sustained periods should be set to approximately 10 percent below the set V I/O level. So for example, if V I/O is set to 3.0V then to drive a high on to PortB the recommended level would be around 2.7V. Alternatively V I/O can be set above the switching level applied to the emulator to say 3.3V in the above example.

1.2. Front panel



The RESET switch will perform a hardware reset to the entire emulator, however it will not reinitialise peripheral RAM or trace and program memory. It will set peripheral registers to their reset values as specified in the respective microcontroller data sheets.

The SVLD (Supply Voltage Level Detect) switch will return a failing result during software control of the SVLD peripheral, i.e. simulating a battery or supply low condition.

The four green LED's indicate the following conditions. HALT LED will light when a HALT instruction is executed. DEBUG LED will light when connection to the host is made and the core is in a BREAK state. When the core is running this LED will go out. BREAK LED will light when the connection to the host has been made. If this LED in not illuminated then there is no communication with the monitor. RESET LED will light when a hardware reset is performed.

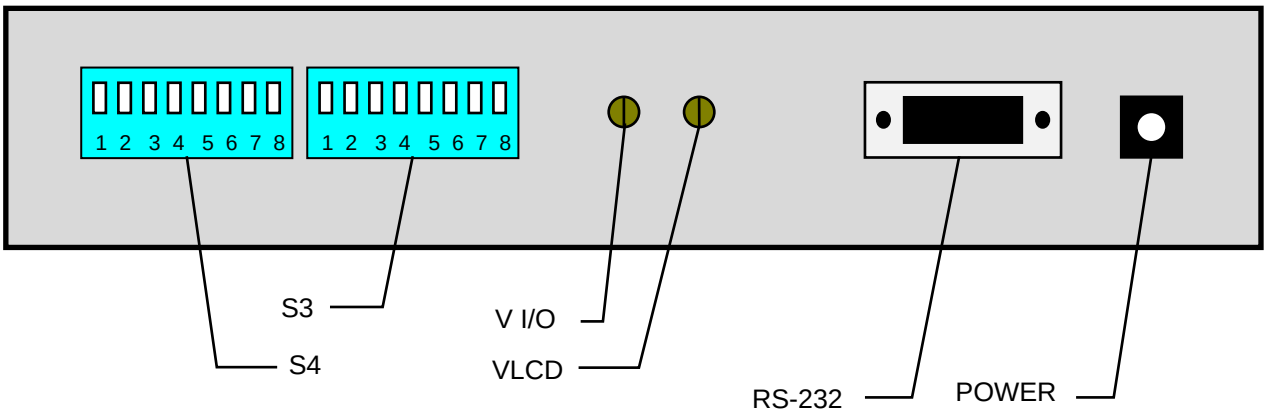
1.2.1. ICE connector pin-out

PIN	SIGNAL	PIN	SIGNAL
1	PORTA 0	21	BUZZER
2	PORTA 1	22	ADC VGND
3	PORTA 2	23	CLOCK OUT
4	PORTA 3	24	ADC VREF
5	PORTB 0	25	PORTS 3
6	PORTB 1	26	ADC Cin
7	PORTB 2	27	PORTS 0
8	PORTB 3	28	ADC Bin
9	PORTC 0	29	
10	PORTC 1	30	ADC Ain
11	PORTC 2	31	
12	PORTC 3	32	GROUND
13	PORTD 0	33	OSC IN
14	PORTD 1	34	PORTS 1 / SWB CLK
15	PORTD 2	35	VIOSET
16	PORTD 3	36	PORTS 2 / SWB DATA
17	RESET IN	37	PORTE 0
18	GROUND	38	PORTE 1
19	STROBE / RESET OUT	39	PORTE 2
20	ADC Din	40	PORTE 3

1.2.2. LCD Connector pin-out **

PIN	SIGNAL	PIN	SIGNAL
1	NO CONNECTION	21	COL 13
2	NO CONNECTION	22	COL 14
3	ROW 4	23	COL 15
4	ROW 3	24	COL 16
5	ROW 2	25	COL 17
6	ROW 1	26	COL 18
7	GROUND	27	COL 19
8	NO CONNECTION	28	COL 20
9	COL 1	29	COL 21
10	COL 2	30	COL 22
11	COL 3	31	COL 23
12	COL 4	32	COL 24
13	COL 5	33	COL 25
14	COL 6	34	COL 26
15	COL 7	35	COL 27
16	COL 8	36	COL 28
17	COL 9	37	COL 29
18	COL 10	38	COL 30
19	COL 11	39	COL 31
20	COL 12	40	COL 32

1.3. Rear Panel

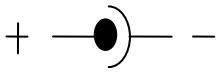


The V I/O adjustment allows the user to change the output drive level and input switching level of the V66xx emulator. Thus, if it is required that the emulator operate in a system where the supply voltage is 0 to 3 volts, then the inputs and outputs can be easily configured for this. The input switching level is CMOS compatible and is around 0.5Vdd. So if V I/O is set to 3v then when input a Port will change from a logic 0 to a logic 1 at about 1.5V. In output mode a logic 0 will be approximately 0V and a logic 1 will be at approximately 3.0V.

In order to set V I/O it is necessary to do the following. Set an output line to drive a logic 1, e.g. PortB-1. Then connect a voltmeter between PortB-1 and Ground (emulator side), turning the V I/O with a pot trimmer will cause the output on the port pin to change, until the required level is obtained. It is possible to vary the output drive between 1.5V and 5.5V.

Adjustment of the LCD drive levels are also possible using the VLCD trimmer pot. The output levels can be set using a similar method to that outlined above, i.e. by setting a particular LCD column line high and measuring with a voltmeter. The VLCD level can also be changed dynamically such that when display data is being updated and the emulator is in RUN mode, it is possible to vary VLCD and observe what is happening to the contrast on the display.

The POWER supply of the emulator is 12V DC/1A



1.4. Switch settings

When switch settings are changed an emulator hardware reset should be performed or the emulator should be powered off then on again.

SWITCH	FUNCTION	DEFAULT	POSITION
S4-1	RESERVED	RESERVED	UP
S4-2	RESERVED	RESERVED	UP
S4-3	RS-232 BAUD RATE (19.2K / 9.6K)	19.2K	UP
S4-4	WATCHDOG ENABLE	ENABLED	UP
S4-5	BREAK PERIPHERY	DISABLED	UP
S4-6	RESERVED	RESERVED	UP
S4-7	RESERVED	RESERVED	UP
S4-8	RESERVED	RESERVED	UP
S3-1	CORE CLOCK	DISABLED	UP
S3-2	OSCILLATOR 0	INTERNAL	UP
S3-3	OSCILLATOR 1	OSCILLATOR	UP
S3-4	CLOCK DIVIDER	DIVIDED BY 1	UP
S3-5	CLOCK DIVIDER	DIVIDED BY 1	UP
S3-6	RESERVED	RESERVED	UP
S3-7	RESERVED	RESERVED	UP
S3-8	INTERNAL BUZZER	DISABLED	UP

Note the following:
During BREAK, if switch S4-5 is in the up position the core will stop execution of instructions but the peripheral components such as timers and serial interfaces will continue to run. If this switch is set down then all peripheral functions will stop when the Core BREAKs.

Switch S4-4 enables or disable the internal watchdog timer. This will also be disabled on BREAK if S4-5 is in the down position.

Switch S3-1 will output the internal core clock out to the target system if required. However, this switch should not be put into the down position on a no load situation.

Switch S3-8 will route the buzzer tone to an internal buzzer.

Switches S3-4 and S3-5 can be set to give a core clock frequency which is a division of the fundamental oscillator fitted in the emulator. The following table shows the frequencies available for the EM66xx microcontrollers:

			EM6605	EM6640	EM66xx
S3-4	**S3-5**	**DIVISOR**	1000kHz	600kHz	32.768kHz
UP	UP	1	1000.00	600.00	32.768
DOWN	UP	2	500.00		
UP	DOWN	4	250.00		
DOWN	DOWN	8	125.00		

On the EM6625 / EM6626, Peripheral Clock is automatically divided to 32kHz:

					EM6625 / EM6626 base clock	
S3-2	**S3-3**	**S3-4**	**S3-5**	**DIVISOR**	U23 / 32,768KHz	U53 / 4,096 MHz
UP	UP	UP	UP	1	32,768kHz	-
DOWN	DOWN	UP	DOWN	2	-	32 kHz
DOWN	DOWN	DOWN	UP	1	-	64 kHz
DOWN	DOWN	DOWN	DOWN	4	-	128 kHz

It's possible to emulate the metal option "MDeb" by using the switch S3-7. CK[11] is used as debouncer clock when the switch is up, whereas CK14 is used when it's down.

Switches S3-2 and S3-3 can be set to select the clock mode. The following table shows the three available modes. The default mode is the internal oscillator.

S3-2	S3-3	SELECTION	EM6640	EM66xx
UP	UP	INTERNAL OSCILLATOR		✓
DOWN	UP	EXTERNAL CLOCK	✓	✓
UP	DOWN	NOT AVAILABLE		
DOWN	DOWN	SYNTHESISER	✓	

For the EM6607, Use the following table to set the input port A reset Combination option.

Option	S3-6	S3-7	Function
A	Up	Up	No Inputs Reset
B	Down	Up	Reset = PA0 * PA1
C	Up	Down	Reset = PA0 * PA1 * PA2
D	Down	Down	Reset = PA0 * PA1 * PA2 * PA*3

For the EM6x80, Two ways is available on the emulator to select the desired RC clock frequency.
The first solution is by hardware switches which represent the metal option present on the EM6680.
S3-6 is used to enable the switches metal option when it's down.
Use the following table to select the core clock value:

Pos.	S3-6	S3-5	S3-4	S3-3	S3-2	S3-7	RC Frequency
1	Down	Down	Down	Up	Up	X	512kHz
2	Down	Down	Up	Up	Up	X	256kHz
3	Down	Up	Down	Up	Up	X	128kHz
4	Down	Up	Up	Up	Up	UP	32kHz
5	Down	Up	Up	Up	Up	Down	64kHz
6	Down	Down	Down	Down	Down	X	800kHz
7	Down	Down	Up	Down	Down	X	400kHz
8	Down	Up	Down	Down	Down	X	200kHz
9	Down	Up	Up	Down	Down	Up	50kHz
10	Down	Up	Up	Down	Down	Down	100kHz

System clock is automatically set to base frequency like 32 or 50kHz.

The second solution is to keep each S3-6, S3-3, S3-4 in up position and to work only with the register RegMFP1 to adjust the desired RC frequency. If one of the switches is down it force a condition which cannot modify by the register.

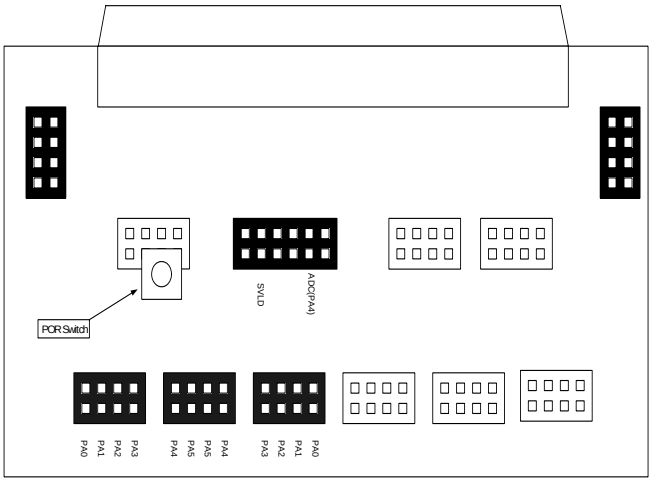
The following table is showing the possible selection:

RegMFP1[3]	RegMFP1[2]	RegMFP1[1]	RegMFP1[0]	Frequency of RCclk	Unit
Opt[7]	Opt[6]	Opt[5]	Opt[4]		
0	0	0	0	**32.00**	**kHz**
0	0	0	1	**64.00**	**kHz**
0	0	1	0	**128.00**	**kHz**
0	0	1	1	**256.00**	**kHz**
0	1	0	0	**512.00**	**kHz**
1	0	0	0	**50.00**	**kHz**
1	0	0	1	**100.00**	**kHz**
1	0	1	0	**200.00**	**kHz**
1	0	1	1	**400.00**	**kHz**
1	1	0	0	**800.00**	**kHz**

RegMFP1[3] bit selects the base frequency, 512kHz or 800kHz.
RegMFP1[2:0] bit select the frequency divider rate, 1, 2, 4, 8, 16.
When the RC frequency is changed the system clock, sysClk, will be automatically close as possible to 32/50 kHz thanks to frequency dividers. It's possible to modify the RC CPU frequency on the fly from 32 to 800kHz.

A special Power on Reset switch is implemented on the EME6680 emulator to be able to generate a POR of the device without disconnect the EME6680 from the PC. The Controller is fully reset like a POR, but you don't need to reconnect the emulator and reload the software on it.

Have a look on the following drawing to see where is situated the Switch for POR.



The following switches are used to select different metal options on EM6682.

Switches	Options	Up pos.	Down pos.
S4-6	Counter update option	RC	RC/2
S4-7	SVLD level selection	EM6681	EM6680
S4-8	Power check selection	After POR	After Reset
S3-6	RC frequency selection mode	Register option as EM6580	Metal option as EM6680
S3-7	Metal option for frequency selection	32/50kHz	64/100kHz
S3-8	Power check level selection	Level 5	Level 9

The ADC/SVLD levels are the following when Vbat is 3V.

LEVEL	EM6681	EM6680
0	0.36	0.5
1	0.49	0.65
2	0.49	0.8
3	0.58	0.95
4	0.68	1.1
5	0.77	1.25
6	0.829	1.4
7	0.922	1.55
8	1.02	1.7
9	1.11	1.85
10	1.2	2
11	1.3	2.15
12	1.39	2.30
13	1.48	2.45
14	1.57	2.60
15	1.66	2.75

The two following tables show the metal mask option of the EM6680 which are implemented in register like the EM6580.

Register **RegMFP0** @ addr. 121 decimal = 79 hex.

Bit	Name	Reset	R/W	Description
3	Opt[3]	0	R/W	Debouncer clock select “0” = ck[11], “1”=ck[14]
2	Opt[2]	0	R/W	Must be kept to “0”
1	Opt[1]	0	R/W	Counter Clock source 7 selection "0" PA3/PA4, "1" RCclk/2
0	Opt[0]	0	R/W	No effect

Note: PA4 Pull resistor is located on the EME66xx PCB at location R24 and R8. (Detail on page 22)

The ADC/SVLD Voltage level option is not implemented on the emulator, But to used Bit of the following register are use for emulate some specific metal options.

Bit3 of RegMFP2 is used to select the SCR timing value, IF ‘0’ is set the EM6680 timing value are used, if ‘1’ the EM6580 timing value are set.

Bit2 of RegMFP2 is used to select the length of the counter, If ‘0’ value is set the 10 bits counter is used, if ‘1’ the 9 bits counter is used.

Register **RegMFP2** @ addr. 123 decimal = 7D hex.

Bit	Name	Reset	R/W	Description
3	Opt[11]	3	R/W	SCR Timing selection (not implemented on EM6580 silicon)
2	Opt[10]	2	R/W	Counter length selection ‘0’ is 10bits ‘1’ is 9bits (not implemented on EM6580 silicon)
1	Opt[9]	?	R/W	not used
0	Opt[8]	0	R/W	ADC/SVLD voltage level#15: "0" 2.75v, "1" 3v (not implemented on the emulator)

Keep Guard: The RegMFPX Registers cannot be used in final ROM version software for EM6680, All three registers are metal option on the EM6680. Please contact EMMicroeletronic for more details.

2. General operational points.

2.1. Using external clock

An external clock can be used to control the EM66xx core (see previous table for switches selection). The clock signal must be between 0 to 5Volts and a 50% duty cycle.

2.2. Option Registers

The EM66xx range of microcontrollers has two methods of customising functions; mask options and option registers. In the emulator only some of these functions are directly implemented. For example the selection of pull-up and pulldown resistors at port pins is implemented directly on the emulator system board as outlined in the following section. The pull resistors can be disable by software (if the register exist in the memory map of the EM66xx microcontrollers).
However those functions that are internal (i.e. those not associated with I/O) such as interrupt edge selection, debouncer selection, etc, are implemented by manipulation of the corresponding option register as they would be in the normal operation of a microcontroller. Those option registers that change I/O characteristics still exist within the emulator and can be written and read as normal, but they will perform no function otherwise. The exception to this is the EM6622 emulation system, which due to system constraints, does not implement those option registers that change pull-up and pulldown selects. Reading these registers will always produce a value of '1111' on the data bus no matter what the written value was.

2.3. Watchdog timers

Only the logic watchdog is implemented in the EM66xx emulation system. This watchdog will generate a reset signal to the core every 2.5 - 3.5 seconds at a system clock of 32.768khz. If the watchdog is enabled on the emulator and the system is running in asynchronous mode then the core will be reset periodically. This will also be the case when the emulator is in BREAK mode, resulting in all internal registers being set to their default values as defined in the specification for each microcontroller.

If the watchdog switch is enabled and the emulator is in synchronous mode then during a BREAK the watchdog will not generate a reset since all peripheral timing functions are statically timed with the core. If the watchdog is disabled and the core is running asynchronously then no reset will be generated however all other timers will be active as normal during a BREAK.

The other watchdog in the EM range of microcontrollers is the analogue watchdog, which supervises the oscillator. If oscillation stops then a reset will be generated (see the appropriate microcontroller specification). This function is not implemented in the emulator.

2.4. Synchronous and asynchronous modes

As outlined above, if the switch enabling synchronous mode is activated then all peripheral functions will stop when the Core BREAKs. This is a useful feature if one wishes to examine the contents of counter registers for example. The exception to this is if a LCD controller is being emulated. Since a frame the emulator for the display generates clock then if the periphery is running synchronously with the core then the frame clock will also stop and the display will fade. It is recommended that when using the LCD controller that the periphery is run asynchronously from the core with the logic watchdog disabled.

3. Configuration resistors.

3.1. Port Configuration

The table below gives the reference designators on the emulator PCB of the pull-up and pulldown resistors for the port configurations on the EM66xx emulator. Any value can be fitted to give the desired configuration, but since these are 1206 SMD resistors it recommended that only EM Microelectronic personnel or someone qualified in SMD fabrication should attempt configuration.

Configuration		Pull-up	Pull Down
PortA	PA0	R2	R18
	PA1	R1	R17
	PA2	R3	R19
	PA3	R4	R20
PortB	PB0	R6	R22
	PB1	R5	R21
	PB2	R7	R23
	PB3	R8	R24
PortC	PC0	R10	R26
	PC1	R9	R25
	PC2	R11	R27
	PC3	R12	R28
PortD	PD0	R14	R30
	PD1	R13	R29
	PD2	R15	R31
	PD3	R16	R32
PortE	PE0	R149	R153
	PE1	R150	R154
	PE2	R151	R155
	PE3	R152	R156
Serial Port	PS0	R35	R39
	PS1	R37	R41
	PS2	R36	R40
	PS3	R34	R38

3.2. Pull-up and Pull-down Resistor placement.

The following representation shows the positions of the configuration resistors given in the table above. Reference designators are also shown on the silk screen of the PCB.

R 13	R 29

R 30	R 14	R 150	R 154

R 153	R 149	R 35	R 39

R 34	R 38

R 15	R 31

R 32	R 16	R 151	R 155

R 156	R 152	R 36	R 40

R 37	R 41

R 1	R 17

R 18	R 2	R 5	R 21

R 22	R 6	R 9	R 25

R 10	R 26

R 19	R 3

R 20	R 4	R 7	R 23

R 24	R 8	R 11	R 27

R 12	R 28

4. Default configurations

Outlined below are each of the resistors and placements for the default configurations of each of the EM microcontrollers supported.

4.1. EM66xx

PortA	Pull-Down
PortB	Pull-Down
PortC	Pull-Down
PortD	Pull-Down
PortE	Pull-Down

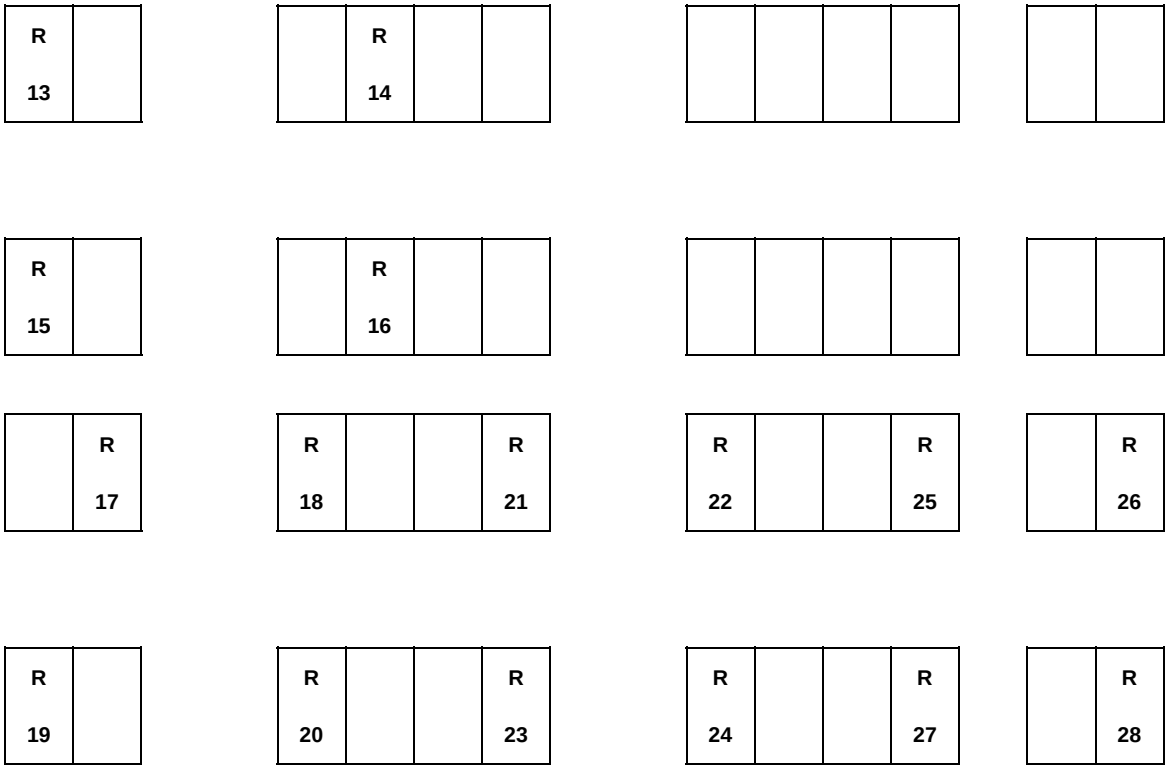
	R 29	R 30			R 154	R 153					
	R 31	R 32			R 155	R 156					
	R 17	R 18			R 21	R 22			R 25		R 26
R 19		R 20			R 23	R 24			R 27		R 28

Level shifters

PortA	U60, U61, U62, U63
PortB	U8, U9, U11, U12
PortC	U14, U15
PortD	U16, U17
PortE	U55, U56, U57, U58
STRB	U18
RESET	U19

4.2. EM6603 / EM6605

PortA	Pull-Down
PortB	Pull-Down
PortC	Pull-Down
PortD	Pull-Up



Level shifters

PortA	U60, U61, U62, U63
PortB	U8, U9, U11, U12
PortC	U14, U15
PortD	U16, U17
STRB	U18
RESET	U19

Jumpers

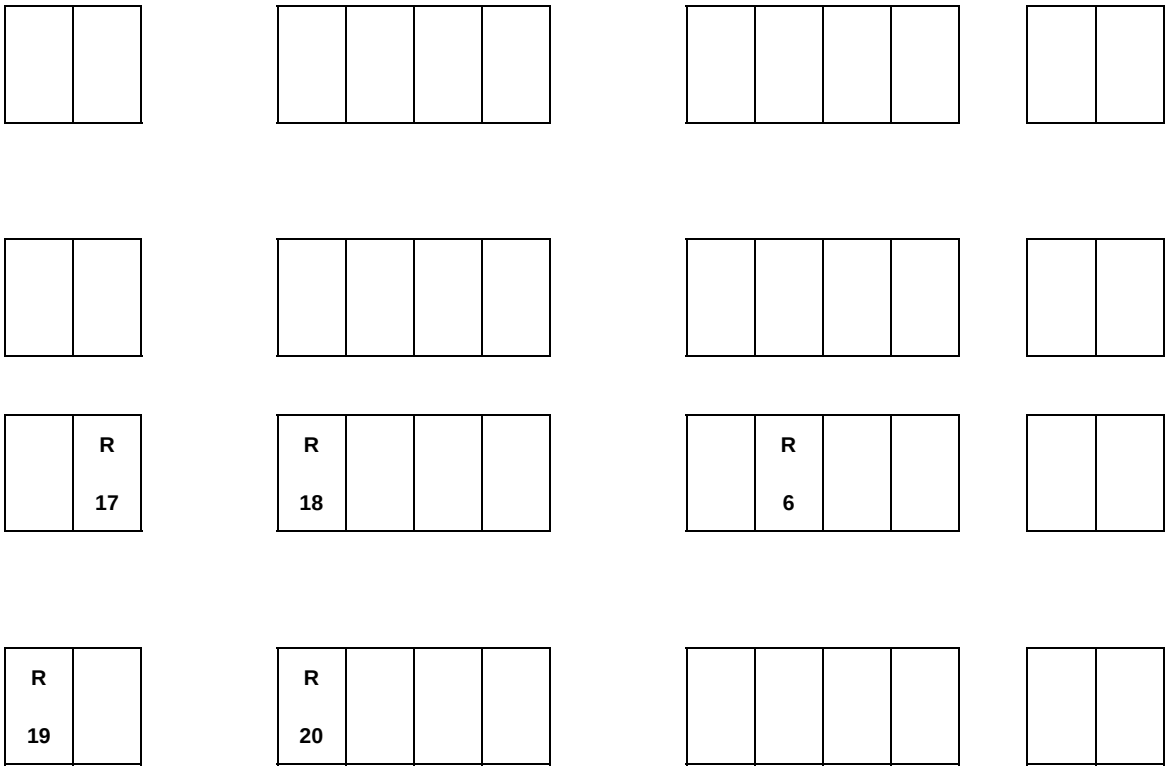
JP19 & JP20 on Pos. 1-2
JP30–JP35 on Pos. 1-2

Oscillator

U23 3,2768MHz

4.3. EM6604

PortA	Pull-Down
PortB = PortC	CMOS Output
PortC = PortB	PC0 : Pull-up
	PC1 : CMOS Output
	PC2 : CMOS Output
	PC3 : CMOS Output



Level shifters

PortA	U60, U61, U62, U63
PortB	U14
PortC	U8, U9, U11, U12
STRB	U18
RESET	U19

Jumpers

JP19 & JP20 on Pos. 1-2
JP30–JP35 on Pos. 1-2

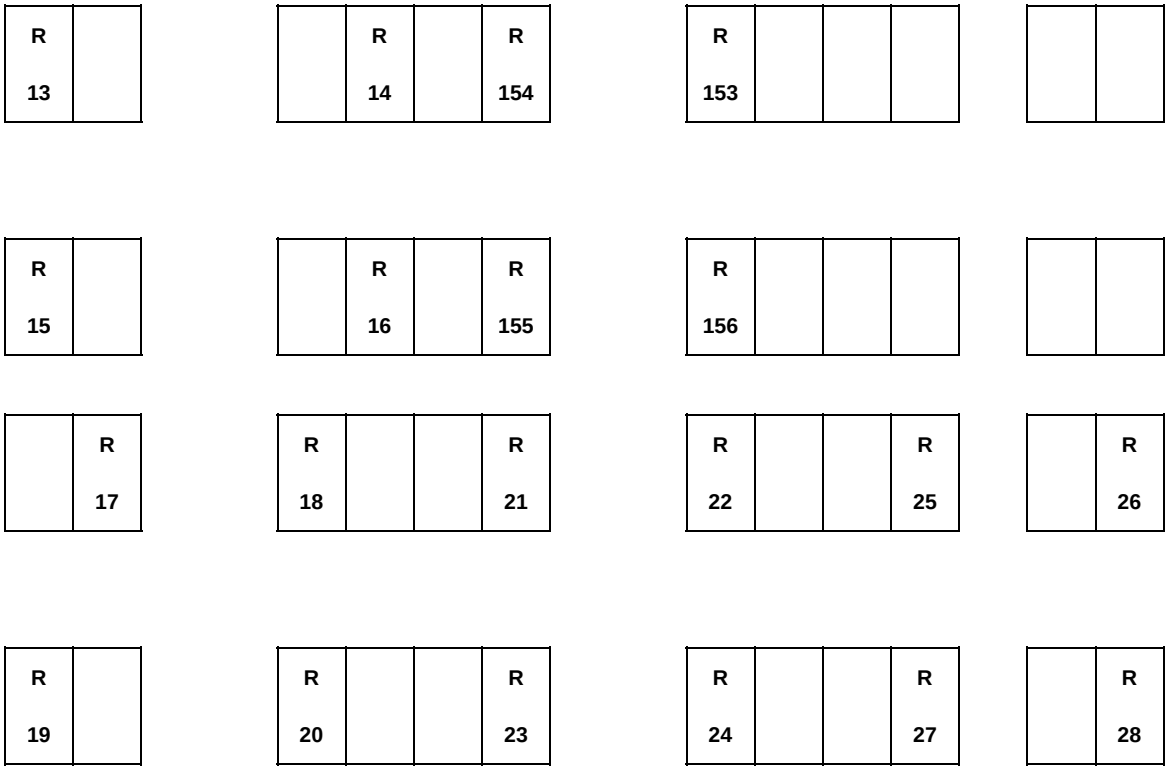
Oscillator

U23 3,2768MHz

Caution: Keep guard that the Port B and the Port C are inverted on the EME6604 due to a hardware constraint.

4.4. EM6607

PortA	Pull-Down
PortB	Pull-Down
PortC	Pull-Down
PortD	Pull-Up
PortE	Pull-Down



Level shifters

PortA	U60, U61, U62, U63
PortB	U8, U9, U11, U12
PortC	U14, U15
PortD	U16, U17
PortE	U55, U56, U57, U58
STRB	U18
RESET	U19

Jumpers

JP19 & JP20 on Pos. 1-2
JP30–JP35 on Pos. 1-2

ADC

PAD8 on Pos. U54

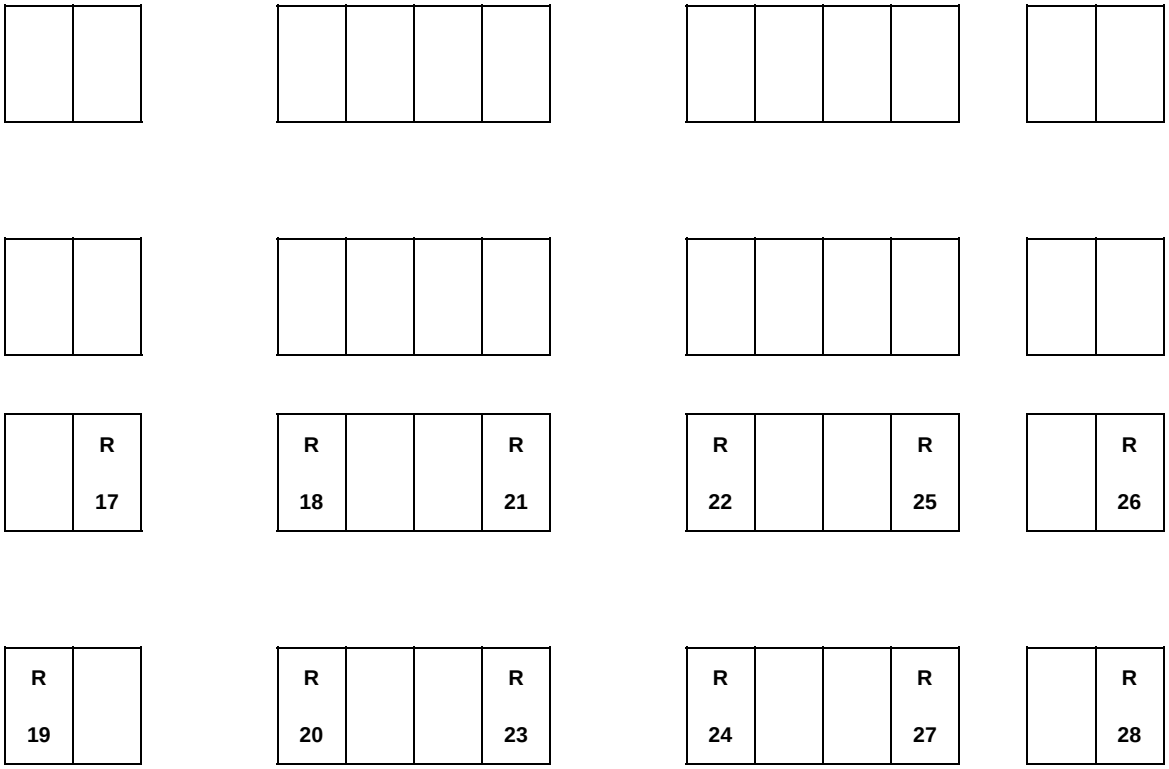
Oscillator

U23 32,768KHz

SVLD Level implemented on the emulator are (01=1.3V), (10=2V),(11=2.3V).
Use the input called SVLD on the ICE board to modify the SVLD level.

4.5. EM6617

PortA	Pull-Down
PortB	Pull-Down
PortC	Pull-Down
SWB	CMOS Output



Level shifters

PortA	U60, U61, U62, U63
PortB	U8, U9, U11, U12
PortC	U14, U15
SWB	U6
STRB	U18
RESET	U19

Jumpers

JP19 & JP20 on Pos. 1-2
JP30–JP33 on Pos. 1-2
JP34 & JP35 on Pos. 2-3

ADC

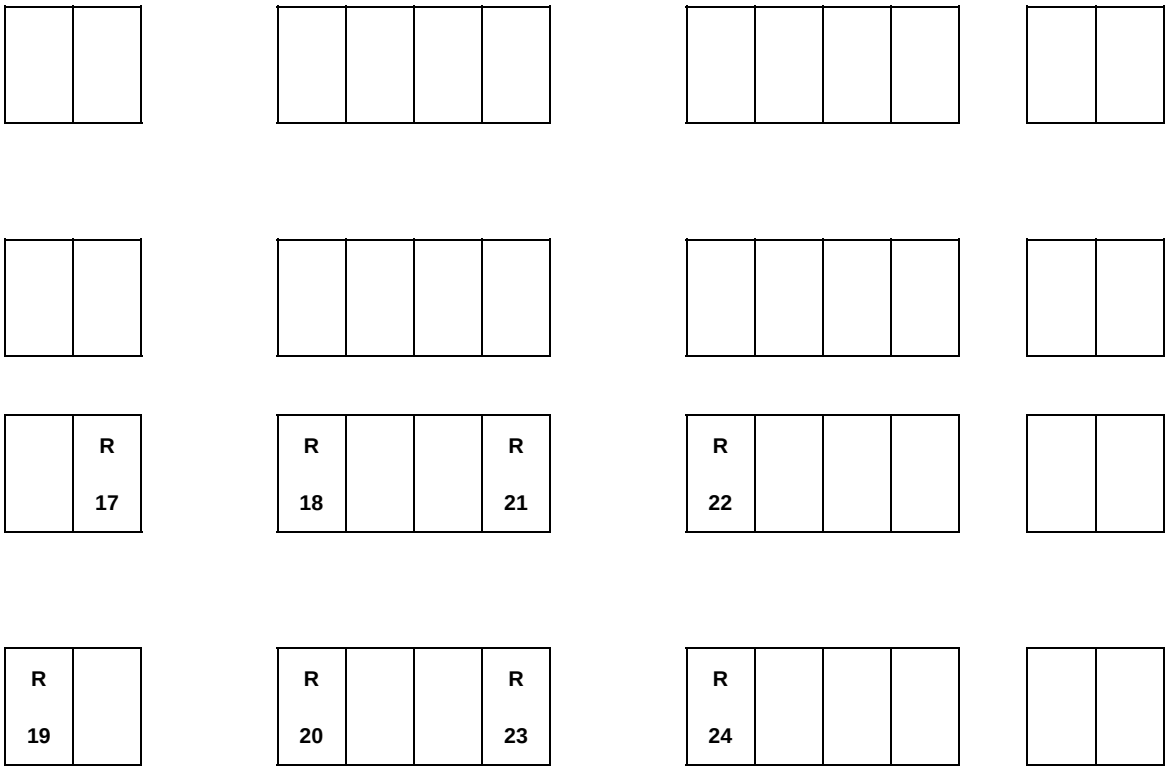
PAD8 on Pos. U54

Oscillator

U23 3,2768MHz

4.6. EM6620

PortA Pull-Down
PortB Pull-Down



Level shifters

PortA U60, U61, U62, U63
PortB U8, U9, U11, U12
STRB U18

Jumpers

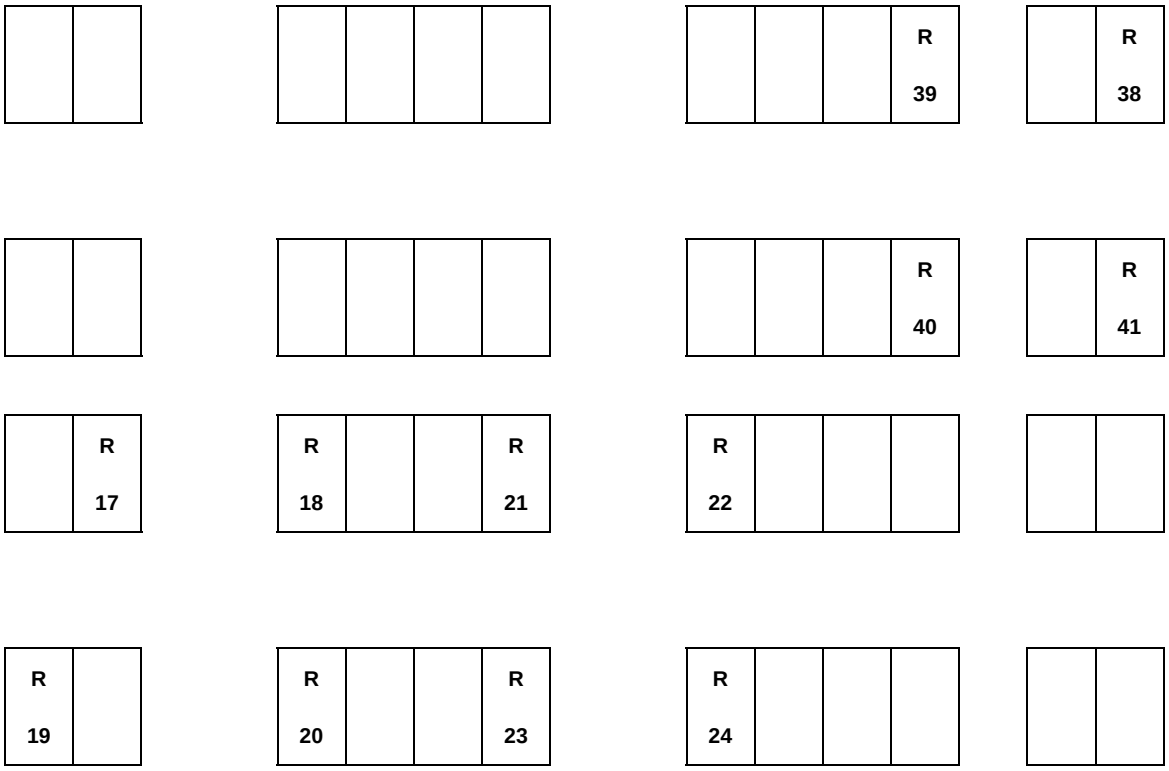
JP19 & JP20 on Pos. 1-2
JP13 on Pos. 1-2

Oscillator

U23 3,2768MHz

4.7. EM6621, EM6622, EM6625, EM6626

PortA	Pull-Down
PortB	Pull-Down
Serial Port	Pull-Down



Level shifters

PortA	U60, U61, U62, U63
PortB	U8, U9, U11, U12
Serial Port	U6, U7, U50
STRB	U18
RESET	U19

Jumpers

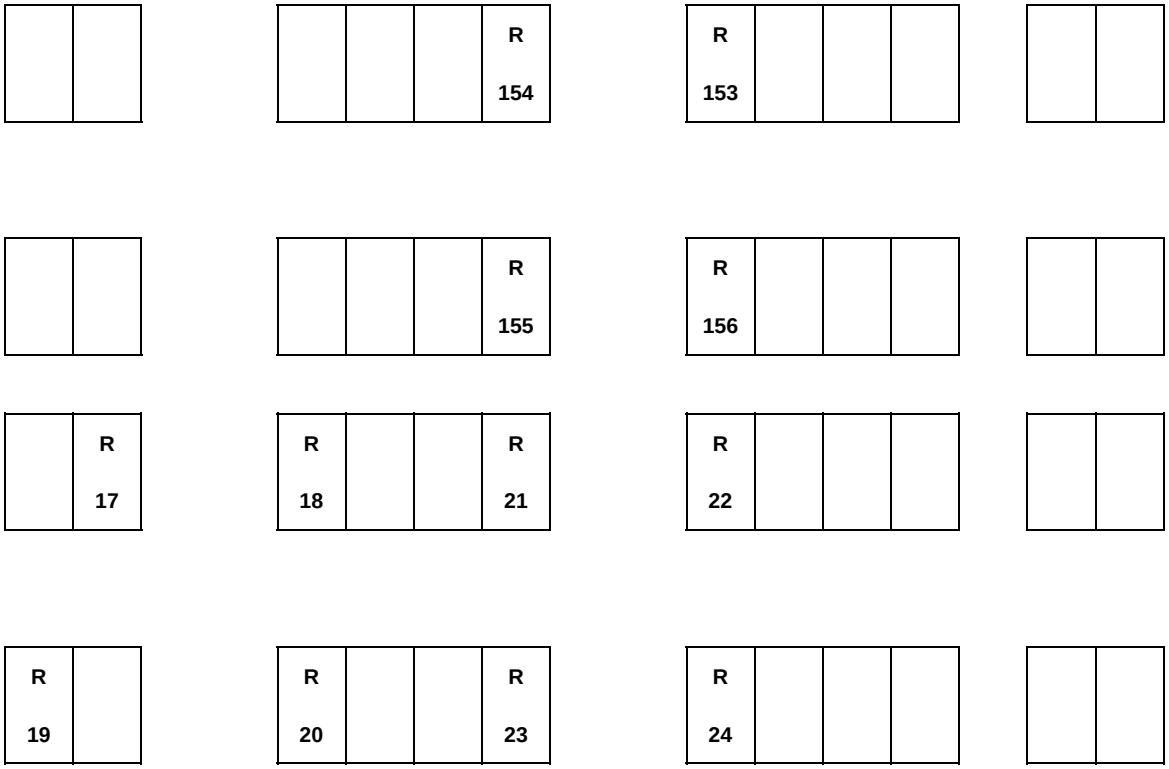
JP19 & JP20 on Pos. 1-2
JP13 on Pos. 1-2

Oscillator

U23	3,2768MHz	6621/22/25/26
U53	4,096 MHz	6625/26

4.8. EM6640

PortA	Pull-Down
PortB	Pull-Down
PortC = PortE	Pull-Down



Level shifters

PortA	U60, U61, U62, U63
PortB	U8, U9, U11, U12
PortC	U55, U56, U57, U58
STRB	U18
RESET	U19

Jumpers

JP19 & JP20 on Pos. 1-2
JP13 on Pos. 1-2

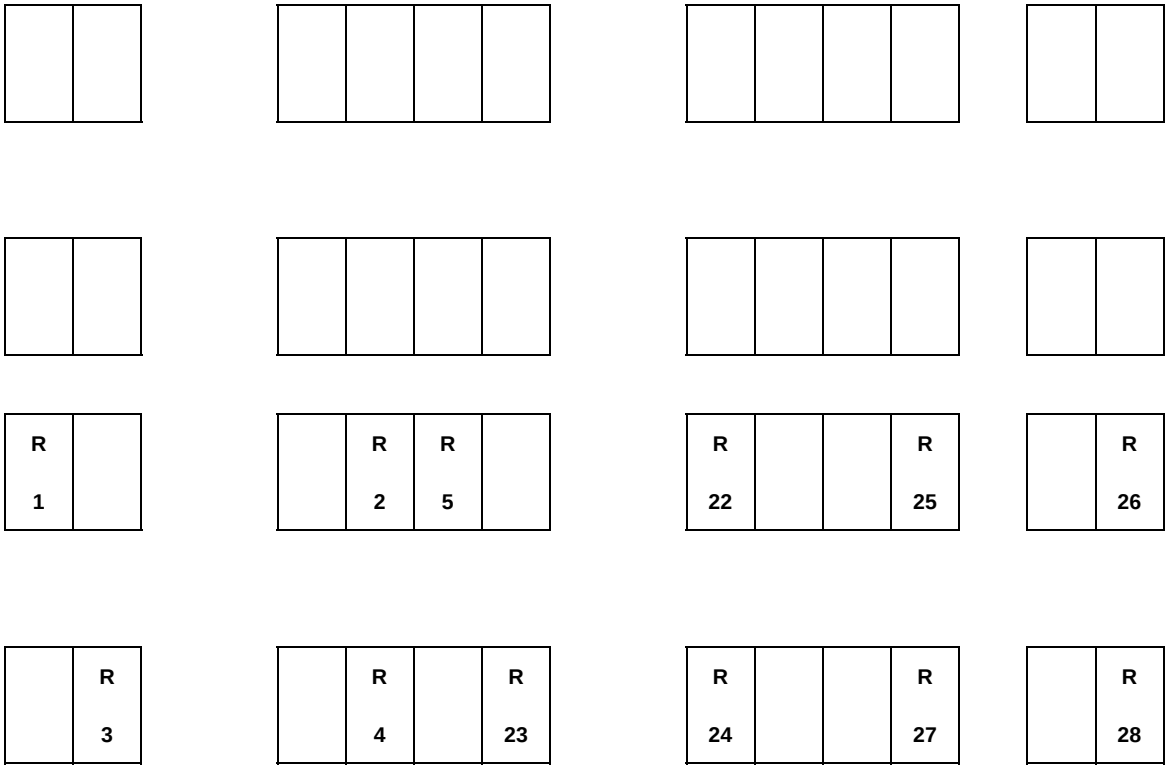
Synthesiser

U53 SPG8640AN

4.9. EM6580, EM6680, EM6682

	Pull-Down	Pull-Up
PortA bit 0	R28	R2
PortA bit 1	R27	R1
PortA bit 2	R25	R3
PortA bit 3	R26	R4
PortA bit 4	R24 (default)	R8
PortA bit 5	R23	R5

Note: R24 or R8 not both at the same time.



Level shifters

PortA	U60, U61, U62, U63
PortB	U8, U9, U11, U12
PortC	U14,U15
STRB	U18
RESET	U19

Jumpers

JP19 & JP20 on Pos. 1-2
JP30-JP35 on Pos. 1-2

ADC

PAD8 on Pos. U54

Oscillator

U23 4.096MHz
U53 8MHz

ADC PA4 ExtVcheck on Din
SVLD on Ain (SVLD value level 4 1.10V in the emulator instead 1.20V)

5. EM66xx emulator upgrade procedure

When changing from one emulator system to another or upgrading the emulator the following procedure should be performed.

1. Switch off power and remove power cable and RS-232 cable at the rear of the system.
2. Remove the four retaining screws on the front panel.
3. Remove front panel.
4. Slide emulator system board forward and out of the housing.
5. At the rear of the board is located an 8 pin DIP module at U42.
6. Carefully remove this device from its socket and replace it with the new ROM.
7. Slide board back into housing and replace front panel with retaining screws.
8. Reconnect power and RS-232 cables.
9. Switch on system and ensure that when the monitor connects to the emulator that both the DEBUG and BREAK LED's are illuminated.
10. The emulator should now be upgraded.