Application Note

Connecting the MCF5307 to 168-Pin Unbuffered SDRAM DIMMs

William R. Benner, Jr.
Pangolin Laser Systems.
William_Benner@msn.com

68K/ColdFire Applications Team

imaging@oakhill.sps.mot.com

A special note of appreciation goes to William Benner, Jr. of Pangolin Laser Systems, the primary author of this application note. Mr. Benner is gracious enough to allow Freescale to share his experience and knowledge with other customers.

This application note shows how to interchange various standard synchronous DRAM (SDRAM) dual-inline memory modules (DIMMs) in a ColdFire® MCF5307 design without re-routing the board. The MCF5307 integrates a Version 3 core with an 8-KByte unified cache, 4-KByte SRAM, an asynchronous/synchronous DRAM controller, and various other popular embedded peripherals. For further details on the MCF5307, refer to the MCF5307 Product Brief (MCF5307/D).

1.1 Intoduction

As the demand for lower-cost, higher-performance embedded products increase, designers are discovering new techniques for lowering the price and complexity of their embedded board platforms. By integrating inexpensive, fast SDRAM devices on board designs, many embedded board manufacturers are realizing lower board costs. Due to their large volume

use in standard PCs, SDRAM devices are currently the least expensive memory available, in addition to being one of the fastest memory types. These devices normally come packaged in handy, upgradable modules called DIMMs, which contain several SDRAM components on one or both sides of the memory card.

Using the SDRAM controller, the MCF5307 can seamlessly interface to standard SDRAM components and DIMMs. Although the number of row, column, and bank select lines can vary from module to module, the multiplexing scheme in the MCF5307 is designed to support a large variety of SDRAM configurations. To extend this MCF5307 feature further, a PLD can be designed to interface to the MCF5307 SDRAM controller that allows the connection of a wide variety of SDRAM DIMMs having different row and column configurations. Thus, rather than hardwiring one specific SDRAM DIMM to a MCF5307 board design, SDRAM DIMMs with varying row and column combinations can be swapped without re-design.

This document details a method of connecting the MCF5307 to single-sided DIMMs with 8, 9, 10, or 11 column address lines, 11, 12, or 13 row address lines, and up to 2 bank address lines using a PLD design. This design can support up to 512 MBytes of memory, which is compatible with the MCF5307's addressing capability.

1.1.1 Definitions

Before the PLD design is presented, it is helpful to review some of the terminology that will be used in this document:

- MCF5307 Memory Bank—This refers to any group of memories that are selected by one of the MCF5307 RAS[1:0] signals. Thus, the MCF5307 can support two SDRAM banks. Note that the RAS[1:0] signals interface to the chip-select signals ($\overline{CS}$) on SDRAMs.
- SDRAM Bank—This term is often used by SDRAM manufacturers to distinguish between the internal partitions, or banks, in a single SDRAM device. For example, one SDRAM component can have four internal SDRAM banks (that is, a 64-Mbit SDRAM is configured as 512K x 32 x 4 banks) Bank selection is controlled through the bank select pins on the SDRAM.
- SDRAM—Synchronous dynamic random access memory. These operate similar to asynchronous DRAMs (ADRAMs) with the advantage of a synchronous clock, a pipelined multibank architecture, and faster speed. These memories also maintain high memory density.
- DIMM—Dual inline memory module. DIMMs contain rows of SDRAM components on one or both sides of the memory card. This is not to be confused with the SDRAM row address lines, called $\overline{SRAS}$ signals. Note that in this application note all DIMMs mentioned in the design will be single-sided, since single-sided DIMMs contain two $\overline{CS}$ lines (in other words, two MCF5307 banks) that the MCF5307 can support. Double-sided DIMMs generally have four $\overline{CS}$ lines—two on one side of the memory card and two on the other. Because double–sided modules present greater load to the address and data lines of a processor, use of single-sided modules are preferred.

  Table 1 shows a fairly complete list of the extensive DIMM configurations, along with their associated parameters. Note that modules can have the same capacity, but have different number of bank select lines, row and column address lines, depending on their organization.

Table 1. Example SDRAM DIMMs

Capacity	# of Chips	Chip Organization	Chip Density	Row/Column/Bank Select Address Lines
8 MBytes	4	1 Mbits X 16	16 Mbits	11 rows/8 columns/1 bank select[1]
16 MBytes	8	2 Mbits X 8	16 Mbits	11 rows/9 columns/1 bank select[1]
32 MBytes[2]	16	4 Mbits X 4	16 Mbits	11 rows/10 columns/1 bank select[1]
16 MBytes	2	2 Mbits X 32	64 Mbits	11 rows/8 columns/2 bank select[3]
32 MBytes	4	4 Mbits X 16	64 Mbits	12 rows[2]/8 columns/2 bank select[3]
64 MBytes	8	8 Mbits X 8	64 Mbits	12 rows/9 columns/2 bank select[3]
128 MBytes[2]	16	16 Mbits X 4	64 Mbits	12 rows/10 columns/2 bank select[3]
32 MBytes	2	4 Mbits X 32	128 Mbits	12 rows/8 columns/2 bank select[3]
64 MBytes	4	8 Mbits X 16	128 Mbits	12 rows/9 columns/2 bank select[3]
128 MBytes	8	16 Mbits X 8	128 Mbits	12 rows/10 columns/2 bank select[3]
256 MBytes[2]	16	32 Mbits X 4	128 Mbits	12 rows/11 columns/2 bank select[3]
64 MBytes	2	8 Mbits X 32	256 Mbits	13 rows/8 columns/2 bank select[3]
128 MBytes	4	16 Mbits X 16	256 Mbits	13 rows/9 columns/2 bank select[3]
256 MBytes	8	32 Mbits X 8	256 Mbits	13 rows/10 columns/2 bank select[3]
512 MBytes[2]	16	64 Mbits X 4	256 Mbits	13 rows/11 columns/2 bank select[3]

1 One bank select line selects between two banks within the SDRAM component.

2 Denotes a double-sided module. The memory from only one side can be used due to the SDRAM controller only supporting two $\overline{CS}$ signals.

3 Two bank select lines select between four banks within the SDRAM component.

1.2 Hardware Configuration

Unlike ADRAM memory, SDRAM does not use a symmetrical multiplexed addressing scheme, one in which each address line on the DRAM device connects to two internal address lines—a row and column address. ADRAM memories interfacing to the MCF5307 can use a simple wiring scheme in which a single wire is added each time an ADRAM address bus grows by one bit, corresponding to one row address and one column address.With SDRAM, however, the lower 8 (or 9 or 10 or 11) address lines typically do connect internally to both row and column address lines, but higher address lines do not connect to column address lines. This is illustrated in Table 1, where the 8 MByte module has 8 column address lines, but 11 row address lines.

The MCF5307 SDRAM controller was designed to interface to these asymmetrical SDRAMs seamlessly. Standard SDRAM component can be directly connected to the MCF5307 by following the easy connection chart found in the Asynchronous/Synchronous Operation Section of the MCF5307 User's Manual. Because the MCF5307 SDRAM controller can be continually re-programmed to support various SDRAM configurations, this advantage can be leveraged to create a helper MUX that can support swapping of these various SDRAMs in hardware.

Because different density SDRAM devices have different asymmetries, a single direct connection scheme is not possible. An easy connection scheme for each specific module type can be derived by referring to Table 1 and the MCF5307 address multiplexing scheme in the asynchronous operation section of the MCF5307 User's Manual. Since there is some commonality with respect to the connection scheme of each module type, an in–between helper MUX can be conceived that interfaces between the MCF5307 and SDRAM DIMM.

Table 2 and Table 3 depict the MCF5307 SDRAM controller to SDRAM connections[1] necessary for different SDRAMs. Table 2 lists the address line connections for various SDRAMs while Table 3 lists the required bank select interface.

Table 2. SDRAM Address Line Connections

CF Address	Condition	SDRAM Address	Column Address	Row Address
A15	Always	A0	A2	A15
A14	Always	A1	A3	A14
A13	Always	A2	A4	A13
A12	Always	A3	A5	A12
A11	Always	A4	A6	A11
A10	Always	A5	A7	A10
A9	Always	A6	A8	A9
A17	Always	A7	A16	A17
A18	8 columns	A8	N/A[1]	A18
A19	9, 10 , or 11 columns		A18	A19
A19	8 columns	A9	N/A	A19
A20	9 columns		N/A	A20
A21	10 or 11 columns		A20	A21
A20	8 columns	A10	N/A	A20
A21	9 columns		N/A	A21
A22	10 columns		N/A	A22
A23	11 columns		A22	A23
A21	8 columns	A11	N/A	A21
A22	9 columns		N/A	A22
A23	10 columns		N/A	A23
A24	11 columns		N/A	A24

[1]Note that although the data bus and other control connections to SDRAM are not necessary to these discussions or detailed in these tables, information on these hardware hookups can be found in the LAB5307 or SBC5307 schematics and the MCF5307 User's Manual at http://www.mot.com/ColdFire.

Table 2. SDRAM Address Line Connections (Continued)

CF Address	Condition	SDRAM Address	Column Address	Row Address
A22	8 columns	A12	N/A	A22
A23	9 columns		N/A	A23
A24	10 columns		N/A	A24
A25	11 columns		N/A	A25
A23	8 columns	A13	N/A	A23
A24	9 columns		N/A	A24
A25	10 columns		N/A	A25
A26	11 columns		N/A	A26

[1] Note: N/A indicates that although a ColdFire address will be multiplexed during the column phase, this does not matter because the number of column lines on the device is satis ed b y lower address lines. For example, if an 8 column SDRAM is used, the 8 column lines are satis ed b y ColdFire address lines A9–A15 and A17. See the table above.

Table 3. SDRAM Bank-Select Line Connections

CF Address	Condition	SDRAM Bank Select
A21	8 columns, 11 rows (19 address lines)	BA0
A22	9 columns, 11 rows (20 address lines) 8 columns, 12 rows	
A23	10 columns, 11 rows 9 columns, 12 rows (21 address lines) 8 columns, 13 rows	
A24	10 columns, 12 rows (22 address lines) 9 columns, 13 rows	
A25	11 columns, 12 rows (23 address lines) 10 columns, 13 rows	
A26	11 columns, 13 rows (24 address lines)	
A22	8 columns, 11 rows (19 address lines)	BA1
A23	9 columns, 11 rows (20 address lines) 8 columns, 12 rows	
A24	10 columns, 11 rows 9 columns, 12 rows (21 address lines) 8 columns, 13 rows	
A25	10 columns, 12 rows (22 address lines) 9 columns, 13 rows	
A26	11 columns, 12 rows (23 address lines) 10 columns, 13 rows	
A27	11 columns, 13 rows (24 address lines)	

An example of a 2 Mbit x 32-bit x 4 bank (8 MByte) SDRAM using Table 2 and Table 3 is shown in Table 4 .

Table 4. 2-Mbit x 32-bit x 4 bank SDRAM Connection to MCF5307

MCF5307 Pins	SDRAM Pins
A15	A0
A14	A1
A13	A2
A12	A3
A11	A4
A10	A5
A9	A6
A17	A7
A18	A8
A19	A9
A20	A10
A21	BA0
A22	BA1

1.2.1 Helper MUX Design

By organizing the MCF5307 SDRAM controller hardware connection information in Table 2 and Table 3, the configuration of the helper MUX inputs/outputs and multiplex selects can be devised. One possible pin configuration for interfacing to a standard 168-pin unbuffered SDRAM DIMM is shown in Figure 1, however this same concept can be carried over to other PLD and pin configurations. This MUX configuration is implemented in a single PLD device that has a low cost and profile, minimal propagation delay, and matches the drive capability of the MCF5307. This example uses a 3.3-volt Lattice ispGAL22LV10K, which has almost identical output characteristics to the MCF5307 and only presents a 5 nS max. propagation delay. This PLD also is available in an SSOP package.

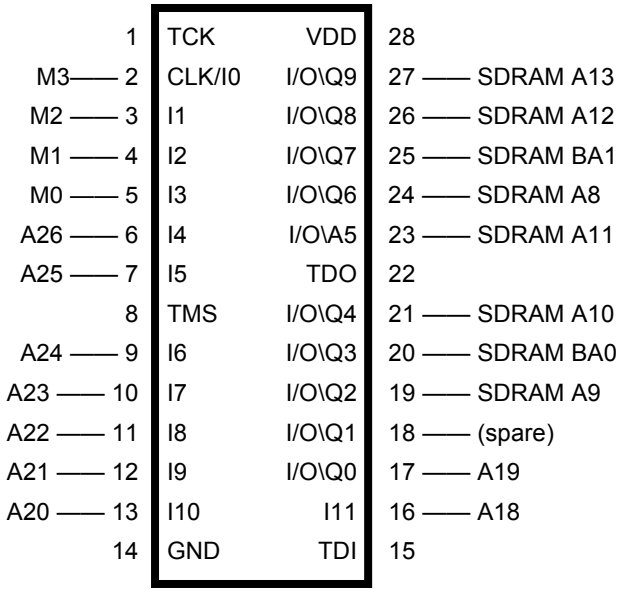

Figure 1. Pin Configuration for ispGAL22LV10K Interface to Standard 168-pin SDRAM DIMM

The M[3:0] lines represent the MUX select configuration mapped from Table 2 and Table 3. A recommended encoding of M[3:0] is shown in Table 5. These signals can be driven by spare parallel port lines on the MCF5307.

Table 5. MUX Select M[3:0] Encoding

M[3:0] MUX Inputs	Corresponding Configuration		
0000	8 columns, 11 rows	==	8 MBytes (16 Mbit) or 16 MBytes (64 Mbits)
0001	9 columns, 11 rows	==	16 MBytes (16 Mbits or 64 Mbits)
0010	10 columns, 11 rows	==	32 MBytes (16 Mbits or 64 Mbits)
0011	8 columns, 12 rows	==	16 MBytes (64 Mbits) or 32 MBytes (64 Mbits or 128 Mbits)
0100	9 columns, 12 rows	==	64 MBytes (64 Mbits or 128 Mbits)
0101	10 columns, 12 rows	==	128 MBytes (64 Mbits or 128 Mbits)
0110	11 columns, 12 rows	==	256 MBytes (128 Mbits double sided)
0111	8 columns, 13 rows	==	32 MBytes (64 Mbits) or 64 MBytes (256 Mbits)
1000	9 columns, 13 rows	==	64 MBytes (64 Mbits) or 128 MBytes (256 Mbits)
1001	10 columns, 13 rows	==	128 MBytes (64 Mbits) or 256 MBytes (256 Mbits)
1010	11 columns, 13 rows	==	512 MBytes (256 Mbits double sided)
1011–1111	Reserved		

1.2.2 Helper MUX Implementation

An in-system programmable device was chosen for the helper MUX implementation because it can easily be reconfigured while on the board. The ispGAL22v10 has a 500 gate density, which easily fits the required logic for the helper MUX. Using Lattice Semiconductor's freeware package "ispEXPERT™ System Starter Kit," the ispGAL was programmed in ABEL-HDL. For more information on obtaining the starter kit please refer to http://www.lattice.com/ftp/ispstarter.html.

Both the ABEL-HDL and PLD equation files for the helper MUX pictured in Figure 1 can be found at the end of this application note in section 1.6 on page 11 and section 1.7 on page 16.

1.2.3 Helper MUX Initialization

Once implemented in a system, the helper MUX in Figure 1 can be used to interface to various 168-pin SDRAM DIMMs by initializing the MUX select pins M[3:0] to the proper SDRAM configuration. SDRAM configuration information can be read at boot time through a serial presence detect (SPD) EEPROM on the SDRAM DIMM. This EEPROM contains data about the number of rows, columns, banks, access times, etc. of the DIMM. The SPD portion of this module is accessed on pins 82 and 83 of a 168-pin DIMM and can be connected to the SDA and SCL pins of the MCF5307, respectively. Information can be read from the SPD by using these I^2C pins on the MCF5307 to determine the configuration of the memory. This information should be read at boot time and it should be used to initialize the M[3:0] lines on the helper PLD, as well as the internal MCF5307 SDRAM configuration registers. Information on the MCF5307 SDRAM Controller initialization sequence can be found in the Freescale application note AN1766/D. The specification for the Intel® PC100 can be found at http://developer.intel.com/design/chipsets/memory/sdram.htm.

1.3 System Design

Besides interfacing the helper MUX to the MCF5307, other board design requirements must be met to allow for the swapping of various DIMMs.

One consideration is the number of clock inputs. Some 168-pin SDRAM DIMMs only require a single clock input on CLK0 (pin 42). Other DIMMs require two clocks on either CLK0 (pin 42) and CLK1 (pin 125) or on CLK0 (pin 42) and CLK2 (pin 79). Yet others require four clocks; CLK0 (pin 42), CLK1 (pin 125), CLK2 (pin 79), and CLK3 (pin 163). Thus, a clock driver with at least four outputs is recommended in the board design to satisfy the requirements of a four-clock input DIMM. DIMMs have on-board termination for unused clock inputs. Use of zero-delay PLL-type clock driver, such as the Cypress Semiconductor CY2305, CY2308, or CY2309 is highly recommended.

The next consideration is the connection to the DIMM chip-select lines that control the module. For single-sided or double-sided 168-pin DIMMs, the MCF5307 $\overline{RAS0}$ should be connected to $\overline{CS0}$ (pin 30) of the module and $\overline{RAS1}$ should be connected to $\overline{CS2}$ (pin 45) of the module. The remaining module chip-select lines—$\overline{CS1}$ (pin 114) and $\overline{CS3}$ (pin 129) should be connected to the 3.3-volt DIMM power-supply through pull-up resistors. This ensures that chips on the back side of double-sided modules remain deselected (i.e., inactive) and prevents any possible contention on the data bus.

Because the back side of the DIMM is not being used in this design, its clock select line should also be disabled. A 168-pin DIMM has two clock enable lines for each side of the DIMM, CKE0 (pin 128) and CKE1 (pin 63), which activate a low-power/self-refresh mode of an SDRAM. It is recommended that the CKE1 control line be left floating, while the CKE0 pin is connected to the SCKE pin of the MCF5307. The CKE1 line controls the clock select line on the back side of DIMMs, and has a 10 K Ohm pull-up resistor on the module itself.

1.4 Timing Analysis

To ensure the helper MUX did not interfere with the timing requirements of a standard SDRAM, a timing analysis was done based on the following design assumptions:

- BCLKO frequency—45 MHz (22 nS period) BCLKO/ 90 MHz core clock
- Clock Driver Delay—0 nS
- EDGESEL Connection—Tied high through a pull–up
- Data Bus Connection—Connected directly to the SDRAM DIMM with no buffer in between
- Address Bus Connection—Connected directly through the MUX to the SDRAM DIMM with no buffer in between
- SDRAM Control Signal Connections—Connected to SDRAM DIMM through a 22 Ω resistor
- BCLKO Rising to Valid Output—The MCF5307 output signals are guaranteed to be valid a maximum of 11 nS after BCLKO is clocked high (parameter B10[1])
- SDRAM Input Setup Time—3 nS for PC66 memory and 2 nS for PC100 memory per Intel® PC SDRAM specification
- SDRAM Input Hold Time—1.5 nS for PC66 memory and 1.0 nS for PC100 memory per Intel® PC SDRAM specification

Thus the MCF5307 to SDRAM Setup Time can be found by the calculation below:

```
 22nS (Bus Frequency)
-11nS (BCLKO to Valid Output time)
- 3nS (PCDRAM setup time)
= 8nS timing margin
- 5nS MUX PLD max. propagation delay
= 3nS worst case timing margin
```

This indicates that even at the worst case, there is enough margin for the MCF5307 signals to reach a PC66 SDRAM while using the helper MUX.

1.4.1 Write Bus Cycle

A write bus cycle was also evaluated as valid since the published hold time to BCLKO for the MCF5307 is 2.0 nS (Parameter B11) and the input hold time for a PC66 memory is 1.5 nS, leaving 0.5 nS as timing margin.

1.4.2 Read Bus Cycle

Read cycles also meet timing margins for setup and hold times to the MCF5307. For SDRAM to MCF5307 setup time, the following calculation was used:

```
 22 nS (BCLKO period)
-10 nS (Clock to valid data for PC66 memory) // This is 7 nS for PC100 memory
- 5.5 nS (Valid input to BCLKO falling - setup time (parameter B1))
= 6.5nS timing margin
```

[1]This corresponds to the parameter value found in the Electrical Speci cation Section of the MCF5307 User's Manual

For SDRAM to MCF5307 hold time the following calculation was used:

```
 3 nS (Output hold time for PC66 and PC100 memory)
- 2 nS (Input hold time for MCF5307) (Parameter B4)
= 1 nS timing margin
```

Thus, this analysis indicates all timing has adequate margin, even for PC66 memory, as long as a zero-delay clock driver is used. Although this timing analysis example is for a PC66 memory, there are no timing violations when using PC100 memory either.

1.5 Timing Considerations for Older OH55J Mask

The timing analysis reviewed in the previous section applies to the most recent MCF5307 mask, the 00J20C. For those using the 0H55J mask of the MCF5307, the output hold time (parameter B11, and parameter B11a), have different values from the 0H55J mask. Specifically, the output hold time for address, data and normal bus control signals is 0.0 nS, and for DRAM control lines such as $\overline{RAS}$ and $\overline{CAS}$, the hold time is -1.0 nS.

Thus, for write cycles, the subtracting a 1.5 nS PC66 input hold time from a -1.5 nS MCF5307 output hold time results in a -3 nS timing margin for SDRAM signals (-2.5 nS for PC100 memory).

In other words, there is insufficient hold time. The proposed solution is to use a clock driver that can provide a 1.0 nS negative propagation delay. This is actually possible with the zero-delay buffers from Cypress. The reference output can be loaded with a 20 pF capacitor, yielding a positive setup time in advance of the clock output. The following is the resultant analysis for all important times.

- MCF5307-to-SDRAM setup time:

```
 22 nS (Bus Frequency)
-11 nS (BCLKO to Valid Output time)
- 2 nS (PC100 Memory Setup time)
- 1 nS (Clock Driver Advance)
= 8 nS timing margin
- 5 nS MUX PLD max. propagation delay
=  3 nS worst case timing margin
```

- MCF5307-to-SDRAM hold time for writes:

```
- 1 nS (worst case Output hold time MCF5307 for SDRAM controls)
+ 1 nS (advance from clock driver)
+ 1 nS (1 nS skew from termination resistors
- 1 nS (input hold time for PC100 Memory)
=  0 nS timing margin
```

  The OH55J mask set MCF5307 Errata published hold time from clock rising edge is 0.0 nS (Parameter B11) for normal signals and –1.0 nS for SDRAM control signals. Given the additional 1 nS advance on the clock with the loaded reference signal, this leaves a 1.0 nS hold time for normal

signals and 0.0 nS for control signals. However, SDRAM control signals are routed through 22 Ohm series termination resistors, before hitting their nominal 50 pF SDRAM input load. This would skew these signals by at least 1.0 nS. Because 1.0 nS is required for PC100 memory, this leaves 0.0 nS timing margin, an adequate outcome.

- SDRAM-to-MCF5307 setup time for reads:

```
 22 nS (BCLKO period)
- 7 nS (Clock to valid data for PC100 memory)  // This is 10 nS for PC66 memory
+ 1 nS (advance from clock driver)
- 5.5 nS (Valid input to BCLKO falling - setup time (parameter B1))
= 10.5 nS timing margin
```

- For the SDRAM-to-MCF5307 hold time:

```
 3 nS (Output hold time for PC66 and PC100 memory)
-2 nS (Input hold time for MCF5307) (parameter B4)
-1 nS (advance from clock)
= 0 nS timing margin
```

Taking into account the 1 nS clock advance with 2 nS required for the MCF5307, the 3 nS hold time of the SDRAM leaves 0 nS timing margin.

Hold times are probably less critical because trace impedances and capacitances on the board will tend to extend these hold times.

In conclusion, using a zero-delay buffer in the negative one nanosecond model works with PC100 memory, assuming that the MCF5307 from the 0H55J mask set is used.

1.6 ABEL Code

The following is the ABEL code file for the helper MUX for interfacing between a MCF5307 and standard 168-pin unbuffered SDRAM:

```
module SDRAMmux
title 'SDRAM Mux Controller for the MCF5307EVM'
"5307mux device 'ispLSI22LV10';
;"*************************************************"
;"This abel file contains the code to mux the address lines"
;"allowing the MCF5307 to support all 168-pin 1 Bank x 64 bit PC compliant DIMMS"
;"It was targeted to Lattice ispLSI 22LV10 PAL  "
;"All logic with this PAL is com
;"CS: XXX "
;"*************************************************"
;"*************************************************"
;"Declaration Section                                "
```

```
;"***************************************************"
;" constants"
   C,P,X,Z,H,L = .C.,.P.,.X.,.Z.,1,0;
;"***************************************************"


M0      PIN     3;              "Mux Input (0)
M1      PIN     4;              "Mux Input (1)
M2      PIN     5;              "Mux Input (2)
M3      PIN     6;              "Mux Input (3)
CA18    PIN     2;              "Input - ColdFire driven address (18)
CA19    PIN     7;              "Input - ColdFire driven address (19)
CA20    PIN     9;              "Input - ColdFire driven address (20)
CA21    PIN     10;             "Input - ColdFire driven address (21)
CA22    PIN     11;             "Input - ColdFire driven address (22)
CA23    PIN     12;             "Input - ColdFire driven address (23)
CA24    PIN     13;             "Input - ColdFire driven address (24)
CA25    PIN     16;             "Input - ColdFire driven address (25)
CA26    PIN     23;             "Input - ColdFire driven address (26)
CA27    PIN     21;             "Input - ColdFire driven address (27)
SA8     PIN     24;             "Output - SDRAM input address (A8)
SA9     PIN     19;             "Output - SDRAM input address (A9)
SA10    PIN     25;             "Output - SDRAM input address (A10)
SA11    PIN     17;             "Output - SDRAM input address (A11)
SA12    PIN     27;             "Output - SDRAM input address (A12)
SA13    PIN     20;             "Output - SDRAM input address (A13)
BA0     PIN     18;             "Output - SDRAM input address (BA0)
BA1     PIN     26;             "Output - SDRAM input address (BA1)
select  = [M3,M2,M1,M0];

; "*******************************"
; " Lattice attributes             "
; "*******************************"
"pLSI property 'CLK XCLK0 CLK0 ';
"pLSI property 'CLK CLK8MHZ SLOWCLK ';
pLSI property 'ISP ON';
pLSI property 'PULLUP ON';
```

```
pLSI property 'Y1_AS_RESET OFF';


equations

;"##########################################"
;"COMBINATORIAL Logic Only"
;"##########################################"


when (select == 0) then {SA8=CA18;
                         SA9=CA19;
                         SA10=CA20;
                         BA0=CA21;
                         BA1=CA22;
                        }

when (select == 1) then {SA8=CA19;
                         SA9=CA20;
                         SA10=CA21;
                         BA0=CA22;
                         BA1=CA23;
                        }

when (select == 2) then {SA8=CA19;
                         SA9=CA21;
                         SA10=CA22;
                         BA0=CA23;
                         BA1=CA24;
                        }

when (select == 3) then {SA8=CA18;
                         SA9=CA19;
                         SA10=CA20;
                         SA11=CA21;
                         BA0=CA22;
                         BA1=CA23;
```

```
                        }

when (select == 4) then {
                                SA8=CA19;
                                SA9=CA20;
                                SA10=CA21;
                                SA11=CA22;
                                BA0=CA23;
                                BA1=CA24;
                        }

when (select == 5) then {
                                SA8=CA19;
                                SA9=CA21;
                                SA10=CA22;
                                SA11=CA23;
                                BA0=CA24;
                                BA1=CA25;
                        }

when (select == 6) then {
                                SA8=CA19;
                                SA9=CA21;
                                SA10=CA23;
                                SA11=CA24;
                                BA0=CA25;
                                BA1=CA26;
                        }

when (select == 7) then {SA8=CA18;
                                SA9=CA19;
                                SA10=CA20;
                                SA11=CA21;
                                SA12=CA22;
                                BA0=CA23;
                                BA1=CA24;
                        }
```

```
when (select == 8) then {SA8=CA19;
                                 SA9=CA20;
                                 SA10=CA21;
                                 SA11=CA22;
                                 SA12=CA23;
                                 BA0=CA24;
                                 BA1=CA25;
                         }

when (select == 9) then {SA8=CA19;
                                 SA9=CA21;
                                 SA10=CA22;
                                 SA11=CA23;
                                 SA12=CA24;
                                 BA0=CA25;
                                 BA1=CA26;
                         }

when (select == ^h0A) then {SA8=CA19;
                                 SA9=CA21;
                                 SA10=CA23;
                                 SA11=CA24;
                                 SA12=CA25;
                                 BA0=CA26;
                                 BA1=CA27;
                         }


"*************************************************"
" Test Vector Section"
"*************************************************"
test_vectors 'M0, M1, M2, M3  Test Vector'
([M3, M2, M1, M0, CA18, CA19, CA20, CA21, CA22, CA23, CA24, CA25, CA26, CA27]-
>[SA8, SA9, SA10, SA11, SA12, BA0, BA1])
 [0,0,0,0,1,0,1,0,1,0,1,0,1,0]->[X,X,X,X,X,X,X];
 [0,0,0,1,1,0,1,0,1,0,1,0,1,0]->[X,X,X,X,X,X,X];
```

```
 [0,0,1,0,1,0,1,0,1,0,1,0,1,0]->[X,X,X,X,X,X,X];
 [0,0,1,1,1,0,1,0,1,0,1,0,1,0]->[X,X,X,X,X,X,X];
 [0,1,0,0,1,0,1,0,1,0,1,0,1,0]->[X,X,X,X,X,X,X];
 [0,1,0,1,1,0,1,0,1,0,1,0,1,0]->[X,X,X,X,X,X,X];
 [0,1,1,0,1,0,1,0,1,0,1,0,1,0]->[X,X,X,X,X,X,X];
 [0,1,1,1,1,0,1,0,1,0,1,0,1,0]->[X,X,X,X,X,X,X];
 [1,0,0,0,1,0,1,0,1,0,1,0,1,0]->[X,X,X,X,X,X,X];
 [1,0,0,1,1,0,1,0,1,0,1,0,1,0]->[X,X,X,X,X,X,X];

end
```

1.7 PAL Equations

The following code represents the PAL equations generated by compiling the PAL ABEL-HDL files using Lattice Semiconductor's freeware package "ispEXPERT(TM) System Starter Kit":

```
ISP Synario 5.01  -  Device Utilization Chart
SDRAM Mux Controller for the MCF5307EVM
------------------------------------------------------------------------------
--
Module                   : 'mux'
------------------------------------------------------------------------------
--

Input files:

    ABEL PLA file        : mux.tt3
    Device library       : P22V10C.dev

Output files:

    Report file          : mux.rep
    Programmer load file : mux.jed
------------------------------------------------------------------------------
--
SDRAM Mux Controller for the MCF5307EVM
P22V10C Programmed Logic:
------------------------------------------------------------------------------
--
```

```
SA8      = (  CA18 & M0 & M1 & !M3
           #   CA18 & !M0 & !M1 & !M2 & !M3
           #   M0 & !M1 & !M2 & CA19
           #   !M0 & !M2 & M3 & CA19
           #   !M0 & M1 & !M3 & CA19
           #   !M1 & M2 & !M3 & CA19 );

SA9      = (  M0 & M1 & !M3 & CA19
           #   !M0 & !M1 & !M2 & !M3 & CA19
           #   !M0 & !M1 & !M2 & M3 & CA20
           #   !M0 & !M1 & M2 & !M3 & CA20
           #   M0 & !M1 & !M2 & !M3 & CA20
           #   !M0 & M1 & !M2 & CA21
           #   M0 & !M1 & !M2 & M3 & CA21
           #   !M0 & M1 & !M3 & CA21
           #   M0 & !M1 & M2 & !M3 & CA21 );

SA10      = (  M0 & M1 & !M3 & CA20
            #   !M0 & !M1 & !M2 & !M3 & CA20
            #   !M0 & !M1 & !M2 & M3 & CA21
            #   !M0 & !M1 & M2 & !M3 & CA21
            #   M0 & !M1 & !M2 & !M3 & CA21
            #   M0 & !M1 & !M2 & M3 & CA22
            #   M0 & !M1 & M2 & !M3 & CA22
            #   !M0 & M1 & !M2 & !M3 & CA22
            #   !M0 & M1 & !M2 & M3 & CA23
            #   !M0 & M1 & M2 & !M3 & CA23 );

SA11      = (  M0 & M1 & !M3 & CA21
            #   !M0 & !M1 & !M2 & M3 & CA22
            #   !M0 & !M1 & M2 & !M3 & CA22
            #   M0 & !M1 & !M2 & M3 & CA23
            #   M0 & !M1 & M2 & !M3 & CA23
            #   !M0 & M1 & !M2 & M3 & CA24
            #   !M0 & M1 & M2 & !M3 & CA24 );
```

```
SA12     = (  M0 & M1 & M2 & !M3 & CA22
           #   !M0 & !M1 & !M2 & M3 & CA23
           #   M0 & !M1 & !M2 & M3 & CA24
           #   !M0 & M1 & !M2 & M3 & CA25 );

BA0      = (  !M0 & !M1 & !M2 & !M3 & CA21
           #   M0 & !M2 & !M3 & CA22
           #   M0 & M1 & M2 & !M3 & CA23
           #   !M0 & !M1 & M2 & !M3 & CA23
           #   !M0 & M1 & !M2 & !M3 & CA23
           #   !M0 & !M1 & !M2 & M3 & CA24
           #   M0 & !M1 & M2 & !M3 & CA24
           #   M0 & !M1 & !M2 & M3 & CA25
           #   !M0 & M1 & M2 & !M3 & CA25
           #   !M0 & M1 & !M2 & M3 & CA26 );

BA1      = (  !M0 & !M1 & !M2 & !M3 & CA22
           #   M0 & !M2 & !M3 & CA23
           #   M0 & M1 & M2 & !M3 & CA24
           #   !M0 & !M1 & M2 & !M3 & CA24
           #   !M0 & M1 & !M2 & !M3 & CA24
           #   !M0 & !M1 & !M2 & M3 & CA25
           #   M0 & !M1 & M2 & !M3 & CA25
           #   M0 & !M1 & !M2 & M3 & CA26
           #   !M0 & M1 & M2 & !M3 & CA26
           #   !M0 & M1 & !M2 & M3 & CA27 );
```

How to Reach Us:

Home Page:
www.freescale.com

E-mail:
support@freescale.com

USA/Europe or Locations Not Listed:
Freescale Semiconductor
Technical Information Center, CH370
1300 N. Alma School Road
Chandler, Arizona 85224
+1-800-521-6274 or +1-480-768-2130
support@freescale.com

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
support@freescale.com

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064
Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor Hong Kong Ltd.
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate
Tai Po, N.T., Hong Kong
+800 2666 8080
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

freescale™
semiconductor

Freescale Semiconductor, Inc.