Freescale Semiconductor

3/21/91

Application Note

DRAM Control with the MC68302

This application note discusses an implementation of DRAM control logic to be used in conjunction with the MC68302 controller. The system uses a block of 512 Kbytes of RAM which is refreshed using the communications processor. This design could also be generalized to any MC68000 system.

Overview

The MC68302 is an integrated communications controller which contains a CMOS M68000 core. Additional peripherals include:

1) 3 Powerful serial communication controllers (SCCs).
2) 2 General Purpose Timers
3) A Watchdog Timer
4) Chip select logic
5) A serial control port (SCP)
6) 1152 bytes of dual port RAM
7) 7 channels of DMA control

The serial communication controllers are serviced by a dedicated processor which has been programmed in revision B silicon to operate as a dynamic RAM refresh controller. This allows the system designer to significantly reduce logic and/or processor overhead required to implement the DRAM interface.

What is a DRAM Interface?

Dynamic RAM is the densist writable memory technology in general use in today's microprocessor systems. The small size of the DRAM cell or bit, allows a maximum amount of memory bits to be stored in on a single piece of silicon. This leads to a minimization of the silicon area of the memory device, and hence it's cost of manufacture.

The key to this small size is a somewhat unusual use of the transistor. In a DRAM cell, it is not used as a switch as in most electronic applications, but as a capacitor. Thus, the value of an individual bit in the DRAM is dependent on the charge stored by an individual transistor. Since silicon is an imperfect capacitor, the charge leaks from the device and therefore needs to be recharged at regular intervals. This can be accomplished by accessing the individual bit with either a read or a write cycle, or by using a special mode of the dynamic RAM called refresh mode. In this mode a complete row of data bits can be refreshed in a single access to the memory device. Therefore a basic function of any DRAM system is to have the capability of accessing each row of each device in the system, before the charge stored by an individual bit has become indeterminate.

The functions required to do the refresh operation are as follows:

1) A timer to supply periodic stimulus to the DRAM system.
2) Logic to provide the bus cycle necessary for the refresh operation.
3) Counters to generate addresses
4) The creation of start and stop address and reset when complete.
5) Logic to interface these functions to the RAM device.

© Freescale Semiconductor, Inc., 2004. All rights reserved.

For More Information On This Product,
Go to: www.freescale.com

freescale™ semiconductor

Freescale Semiconductor, Inc.

Additionally, as a superset of the above, is a requirement to be able to read and write the DRAM cell!

In this design, the MC68302 provides items 2, 3, and 4 above with what is called its "communications processor". (The communications processor consists of a RISC processor, 6 Serial DMA channels, and three flexible serial channels.) One of the onboard MC68302 timers or baud rate generators can supply number 1. Additionally, the logic required for the refresh (item 5) can be the same as that which is used for the read and write cycles, as the refresh cycle is a subset of a read cycle to the device.

Functions of the Communications Processor

The function of the communications processor in the DRAM design is to provide the regular bus cycles or refresh cycles. This is done by using an external timer to provide regular stimulus. Then the communications processor provides an external read-byte cycle to the 68000 bus. The processor keeps the start, stop address and the page size of the RAM in its dual port RAM where buffer descriptors for SCC2 would otherwise reside. This allows unattended operation of the refresh control section of the communications processor. The processor does not generate the control strobes for the RAM. These are the row and column address strobes for strobing in row and column address to the RAM for nibble selection. These have very exacting timing requirements and are the heart of the design of a DRAM subsystem.

Design Criteria

This DRAM refresh design had the following goals:

1) INTERFACES TO 4 MCM514256P80 DRAMs
2) Two RAMs for the high byte and two for the low byte for any bus cycle.
3) Uses chip select 1 for the required chip selects.
4) LOGIC ABLE TO FIT INTO AN ALTERA EP630 PAL.
5) One wait state for the DRAM access.
6) Uses timer one for refresh timing.
7) USES TWO MC74F240s FOR THE ROW AND COLUMN ADDRESS MUXES.
8) Uses no faster than 80ns DRAMs

Thus, the circuitry required for the DRAMs consists of a PAL, and two TTL buffer devices in addition to the DRAMs.

Timing

The most important aspect of any digital design is the analysis of the timing requirements of the individual components used in the system. The timing to focus on in this design is:

1) The read cycle of the MC68302
2) The read cycle of the MCM514256 DRAM
3) The PAL speed
4) ENABLE AND DISABLE TIMING OF THE MC74F240 BUFFERS

The DRAM Timing

In order to access the DRAM, the following events must be created.

1) A row access strobe or RAS must be asserted

For More Information On This Product,
Go to: www.freescale.com

Freescale Semiconductor, Inc.

2) At the falling edge of RAS, the row address must be valid at the address pins of the DRAMs being accessed.
3) A column access strobe or CAS must be asserted
4) At the falling edge of CAS the column address must be valid at the address pins of the DRAMs being accessed.
5) The W and G pins of the DRAM must be set depending on the type of access that the processor is making to the DRAM.

There are of course some unusual requirements of the DRAM that also need to be addressed:

1) Row address must be present at the DRAM for a minimum of 10 ns after the assertion of RAS.
2) Column address must be present at the DRAM for a minimum of 15 ns after the assertion of CAS.
3) RAS must remain unasserted for a minimum of 60 ns between accesses.

When you have met all of these requirements, you will get valid data from the DRAM, a maximum of 20 ns after the CAS strobe is valid. A comparison of the 3 types of DRAM cycles covered in this note: read, write and refresh shows that when the timing for the read cycle is met, the timing for the write and refresh cycles will be met. The write cycle is met by the fact that the data from the MC68302 processor is valid during nearly all of the bus cycle of the device, and the refresh cycle because a read or write of the DRAM will also refresh the accessed row. Note that this design does not use the specialized refresh operation of the MCM514256 DRAM device.

A Brief Discussion of MC68302 Bus Timing

The MC68302 communications processor has an external bus which is identical in timing to the MC68000 -- that is a minimum bus cycle requires 4 clock cycles minimum, with an integer number of cycles inserted as wait states. The half cycles of the clock are numbered S0 through S7 with the wait state half cycles are called SW1 and SW2. This wait cycle is inserted between S4 and S5.

The address strobe (AS) is active at the start of the bus cycle when the address bus is stable. This is during or slightly after S2. The read/write signal R/W* is activated depending on the type of bus cycle slightly after address strobe. For a read cycle data, needs to be available to the MC68302 slightly before the falling edge of S6. It is with this combination of signals that the interface to the DRAMs is derived.

The chip select signal is unique to the MC68302, and has timing which is similar to address strobe. It is programmed to respond to the address range of the DRAM bank, so that only accesses meant for the DRAM will be decoded by the DRAM interface circuitry.

DRAM Timing Generation from the MC68302 Bus Cycle

The signals which need to be generated by the system to interface to the DRAM are:

1) RAS*
2) CAS*
3) R/W*
4) CS

Additionally the signals which select either the address lines 1 through 9, or 10 through 18 also need to be generated. They are called:

For More Information On This Product,
Go to: www.freescale.com

Freescale Semiconductor, Inc.

1) MUXR - The lower addresses which are asserted during RAS
2) MUXC - The higher addresses which are asserted during CAS

We will first discuss the logic required to generate the R/W and the CS signals, as they are less complicated than the rest. Then in the following paragraphs, we will discuss the timing of the other signals in order given above.

R/W and CS

The read/write signal is generated directly by the R/W line from the MC68302. The CS signal is gated with UDS and LDS to select either the low or the high byte of the DRAM bank.

RAS*

The RAS signal is required early in the cycle when the address lines are stable. Since address strobe is not guaranteed to be active until sometime during S2, RAS* is the output of a latch which is clocked by the system clock, and whose data input is address strobe. This asserts RAS* at sometime during S4.

Let's take a look at some of the alternatives in the design at this point. First there is the selection of the address strobe as the RAS* signal. This is limited by the precharge time (the time that the line must be high between successive accesses) of the RAS* signal. This is particularly critical when being used with a system clock frequency of greater than 16Mhz. In that case, the guaranteed deasserted time of address strobe becomes less. This means that this design philosophy will require progressively higher speed DRAMs for faster system clock speeds. This can add significantly to the system price as fast DRAMs are expensive, however, this is a possible tradeoff in the design.

MUXR

This signal enables the address buffer which drives the address lines on the DRAM during the RAS address cycle. It needs to be asserted at a time before RAS* such that the address lines of the DRAM are at the proper logic level coincident with the assertion of RAS*. This line IS TO BE HELD FOR A TIME OF 80 NS. WHICH WILL ALLOW THE ROW ADDRESS TO be strobed into the DRAM. In addition during the time that MUXR is asserted plus the time that it takes to deassert the column address lines, it is necessary for MUXC to be deasserted so that two drivers are not driving the same address lines to the DRAM. This is because the technique used to select the address lines for CAS* and RAS* is a three-statable buffer and not a multiplexer. This was done to reduce chip count, and system cost.

MUXC

After the deassertion of MUXR the next signal to assert is MUXC. It needs to be asserted at a time before CAS* such that the address lines of the DRAM are at the proper logic level coincident with the assertion of CAS*. The this line is to be held for a TIME OF 15 NS. WHICH WILL ALLOW THE COLUMN ADDRESS TO BE STROBED into the DRAM.

CAS*

The CAS* line is the last line to be asserted to the DRAM. The timing OF CAS* NEEDS TO BE A MINIMUM OF 20 NS BEFORE THE PROCESSOR REQUIRES data. The falling edge of S5 is the latest clock edge which can be used to strobe the CAS* signal and still guarantee data to the processor. Note that this allows one clock edge for each of the signals that need to be asserted to generate timing to the DRAM. The complete list is as follows:

For More Information On This Product,
Go to: www.freescale.com

Freescale Semiconductor, Inc.

1) MUXR Falling edge of S2
2) CAS* Rising edge of S4
3) MUXR Deasserted on falling edge of S4
4) MUXC Rising edge of SW1
5) CAS* Falling edge of SW2
6) MUXC Deasserted on rising edge of S6
7) CAS* and RAS* Deasserted on rising edge of S0

Implementation of this Logic in a PAL

The following shows the timing generation behind the design of a PAL for the MC68302 for DRAM access and refresh timing generation.

WRITE TIMING

The most difficult timing to meet for the interface to the DRAM is the read timing as the data window to the processor is small. However during a write cycle, the data from the processor is available during address strobe, and the DRAM only requires data to be presented to its data pins for a period of 20 ns after CAS* strobe is asserted. Therefore when read timing is met it is assured that write timing will be correct.

DESIGN IMPLEMENTATION

The next step is to implement the design. The basis for this design was a board design which used a minimum of board space. The space allowed was for three devices:

1) A 20 pin PAL device or equivalent
2) Two bus buffers

The bus buffers are used for buffering the address lines and also as a multiplexer by the fact that the chosen bus buffers have 3 state capability. The PAL is used for the complex logic functions that we have described earlier. The details of the implementation of the design of the PAL follows.

PAL DESIGN

THE PAL CHOSEN IN THIS DESIGN WAS AN ALTERA EP630. IT WAS CHOSEN AS THERE A BROAD range of speeds available, and the functionality of the device was appropriate.

RAS*

Chip select 1, which is the chip select chosen for the RAM address space, is ORed with address strobe to generate the input the D flip flop which is clocked by the system clock. Thus, on the next rising edge of the system clock after both chip select and address strobe have gone low, the output of the D flip flop goes low, and RAS* is asserted.

The output for RAS*, MUXR*, and MUXC* is enabled by chip select and address strobe going low. This causes both of the external buffer enables to be in the disable state when the DRAM bank is not being selected. Additionally it allows the maximum precharge time for the RAS* signal between successive accesses as the address strobe is deasserted before the end of the cycle, and allows the RAS* control signal to deassert before S0.

For More Information On This Product,
Go to: www.freescale.com

Freescale Semiconductor, Inc.

MUXR* and MUXC*

These signals are generated by MUXC* being the inverse of the MUXR* signal. Thus when the signals are initially enabled by the assertion of chip select and address strobe, MUXR* which is high as it is the output of a D flip flop whose input is RAS* and is clocked by the system clock. The change in MUXR* comes at the rising edge of SW1 when it is asserted. At the same time MUXC* is deasserted by the fact that it is the inverse of the MUXR* signal. This has the effect of generating appropriate timing for the bus buffers as the address changes one full clock after the RAS* signal has changed. Also MUXC* is asserted on half of a clock cycle before the CAS* signal which meets the required DRAM timing.

It is important to note here that the MUXR* signal does not become deasserted on the falling edge of S4. Although this is the ideal case, it would require a flip flop which could be clocked on the falling edge of the clock. This is not possible with this PAL device.

CAS*

The logic equation for the CAS* signal is:

(((AS* + CS1*)* CAS) + (CLKO* MUXR RAS))*

The first part of the equation serves to keep the CAS* signal asserted until the deassertion of either AS* or CS1* after CAS* has been asserted. The second part of the equation is what asserts CAS* at the proper time. THIS SIGNAL CAN THEN BE GATED WITH THE UPPER (UDS) AND LOWER (LDS) BE ADDRESS STROBES SUCH THAT ONLY THE UPPER OR LOWER BYTE MEMORY IS ACCESSED ON 8 BIT ACCESSES.

When both RAS* and MUXR* are asserted on the next falling edge of the clock, the CAS* signal will be asserted. Then by the first part of the equation, CAS* will be held low until the end of the bus cycle.

It is here that the timing parameters of the PAL device used are the most critical. The maximum propagation delay for the device must be such that the following equation is met:

PF < CLKO PERIOD

Where

PF - The forward propagation delay of the PAL
CLKO PERIOD - The inverse of the frequency of the MC68302 clock signal times .48 which is the minimium duty cycle for the clock.

The CLKO PERIOD is simply the minimum time that the clock will be in either state, or the minimum duty cycle times the total clock period. The minimum duty cycle is given in the electrical specifications section OF THE MC68302 USERS MANUAL FOR A 16.67 MHZ SYSTEM CLOCK THIS IS 28.8 NS.

This equation allows the CAS* signal to be logically generated, and the feedback signal to propogate through the device without the generation of a race condition with the clock signal. The clock is gated with the CAS* signal to guarantee that it goes low on the falling edge of the clock and therefore allows enough set up time for the column address.

Other Hardware Considerations

For More Information On This Product,
Go to: www.freescale.com

Freescale Semiconductor, Inc.

The only additional wiring that is requried for the DRAM refresh to operate is the connection of the Timer 1 output to the input of the RISC controllers request input. Thus when the counter is properly programmed, will generate the periodic stimulus required by the RISC communication controller to generate an appropriate read bus cycle. Since in this application we are using counter 1 for the interrrupt timer, the output pin is PB4. The input connection for the RISC controller is PB8. Thus, connection between pin PB4 and PB8 is required for the design to be effective.

CLK0 and CLK1

There are two clock inputs to the PAL . These are labeled CLK0 and CLK1. They represent the same electrical signal from the processor which is CLKO. CLK0 is used as the input to the registers in the PAL, and CLK1 is used as a logical input. CLK1 is inverted and its result is used to generate and hold the CAS* signal to the proper timing.

UDS and LDS

In the implementation of this design, it is important to consider the action of the UDS and LDS strobes in both the refresh and memory access modes. By definition, the strobes are asserted during a bus cycle dependent on the state of the lowest bit of the requested address. Thus the correct data byte can be accessed in 8-bit data requests. It is important to note that although the strobes are used to specify which of the RAMs are accessed during a memory access, during a refresh only operation, they are not used. This is because UDS and LDS are used to qualify the CAS* signal and not the RAS* signal which is used for refresh. Thus on any access a row of the DRAM is refreshed, but data is only made ready on the device which is actually being addresse.

Pull-up Resistors

For this specific design implementation of the PAL, it is necessary to add 4.7K ohm pull up resistors to the outputs for RAS*, CAS*, and MUXC. This is because the mechanism for disabling these signals is to three-state the output driver.

General Software Considerations

In addition to the hardware connections and the PAL design described above, it is vitally important that the internal parameters be properly loaded into the proper section of the dual port ram. The sections of the chip that must be programmed and the order in which they are to be programmed are the following:

1) Set the wrap bit in SCC2's transmit buffer descriptor #5.

Since the DRAM microcode in the RISC processor uses transmit buffer descriptors 6 and 7, it is important that these ram locations are not accessed as buffers.

2) DRAM refresh parameters.

These parameters are the memory start and stop locations as well as the configuration of the DRAM memory bank.

3) Programming the counter.

The counter output must generate an output pulse every 15.625 microseconds in order to properly refresh the DRAM.

For More Information On This Product,
Go to: www.freescale.com

Freescale Semiconductor, Inc.

4) Setting the ERRE bit in the System Control Register (SCR).

This allows the RISC processor to receive external stimulus (from timer 1) to run the DRAM refresh microcode.

Software Detailed Programming

This section gives more detail on the individual sections to be programmed and the assembler statements which can be used to setup these values. The code makes the following assumptions:

a) The BAR address is stored in address register A0.

b) The memory accesses are made to supervisor data space, which means that the function code equals 5.

1. SCC2 WRAP BIT IN BUFFER DESCRIPTOR NUMBER 5

This bit is located in address base + 568 where the base address is the value loaded into the BAR register on powerup which determines the starting location of the dual port RAM. The upper half of this onchip dual port RAM is used as registers for the RISC communications processor. The instruction used to program this bit IS:

```
ORI    #$2000,(568,A0)
```

2. DRAM REFRESH PARAMETERS

The final parameters that we are concerned with are the actual register values that the RISC controller uses when actually performing the DRAM refresh. These are as follows:

1) DRAM high address and function code
2) DRAM low address
3) Increment count
4) Number of rows
5) Clearing Temporary Count

The description for these paramters are found in the MC68302 data book in the DRAM refresh controller description section. Each of these are word values which can be programmed using the following program sequence:

```
MOVE  #$1000,(570,A0)  # DRAM_high
MOVE  #$0200,(572,A0)  # DRAM_low
MOVE  #$2,(574,A0)     # INCREMENT
MOVE  #$200,(576,A0)   # COUNT
MOVE  #$0000,(57C,A0) # T_Count
```

Note that the function code presented during a DRAM refresh cycle is 1, to distinguish it from the supervisor data function code of 5. The starting address was chosen to be $200 (rather than $0) to keep the refresh address from overlapping the 68302 registers (i.e. BAR and SCR) at locations $0F0 to $0FF.

3. PROGRAMMING THE COUNTER

For More Information On This Product,
Go to: www.freescale.com

Freescale Semiconductor, Inc.

In this design example, we have used timer 1 (TMR1). The timer is set for the following conditions:

1) Timer reference register is initialized to the desired timeout.
2) Enable timer
3) Input clock source is the master clock
4) Restart the counter after the reference count is reached
5) Disable interrupts
6) Active low pulse for one CLKO period on reaching reference count.
7) Disable interrupt on capture event
8) Prescaller set to zero
9) Timer reference register is initialized to the desired timeout.

The timeout is written to the reference register TRR1.

```
MOVE  #$104,(842,A0)
```

The rest of the parameters are programmed by writing to the TMR1 mode register. The instruction for implementing this is:

```
MOVE  #$B,(840,A0)
```

Note that the Timer 1 interrupt and PB8 interrupt have not been enabled in the interrupt mask register (IMR), to keep these unneccessary interrupts from loading down the M68000 core.

4. ERRE BIT

The System Control Register (SCR) is a 32 bit register which consists of system status and control bits. The address of this register is fixed at address $F5. To set the ERRE bit for DRAM control use you must mask bit 6 of this address to a 1 in supervisor mode. The instruction used for this is as follows:

```
ORI   #$40,$F5
```

NOTE: The PAL schematics are not available in electronic form. Your Motorola representative can obtain a hard copy for you by calling the Datacommunications Operation in Austin.

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

Block Diagram

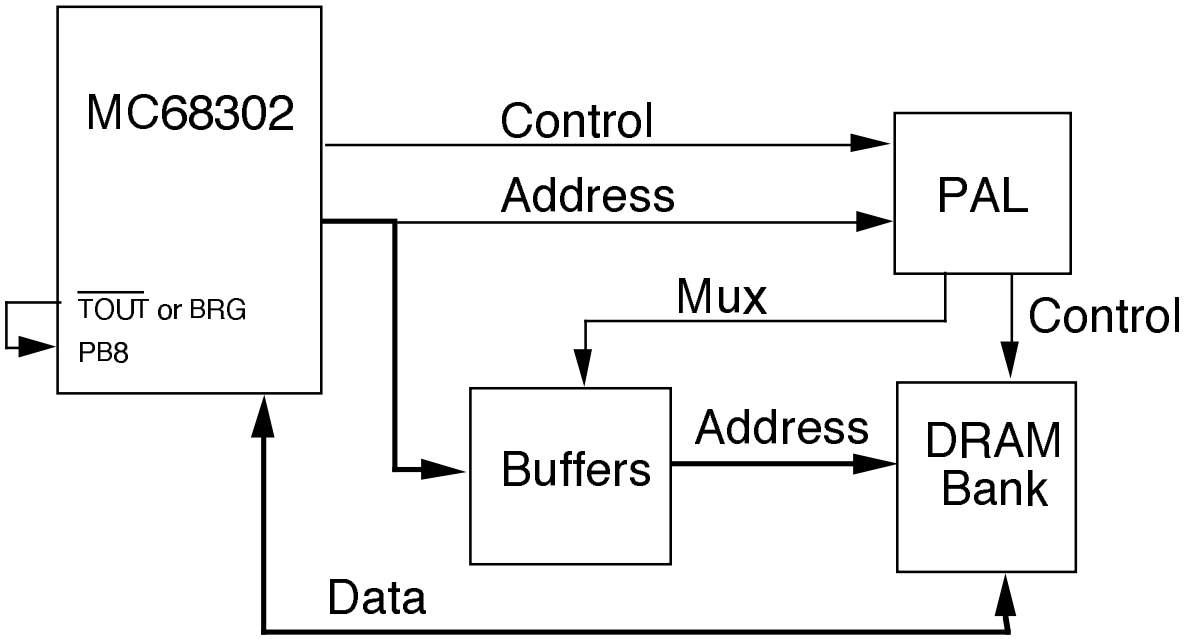
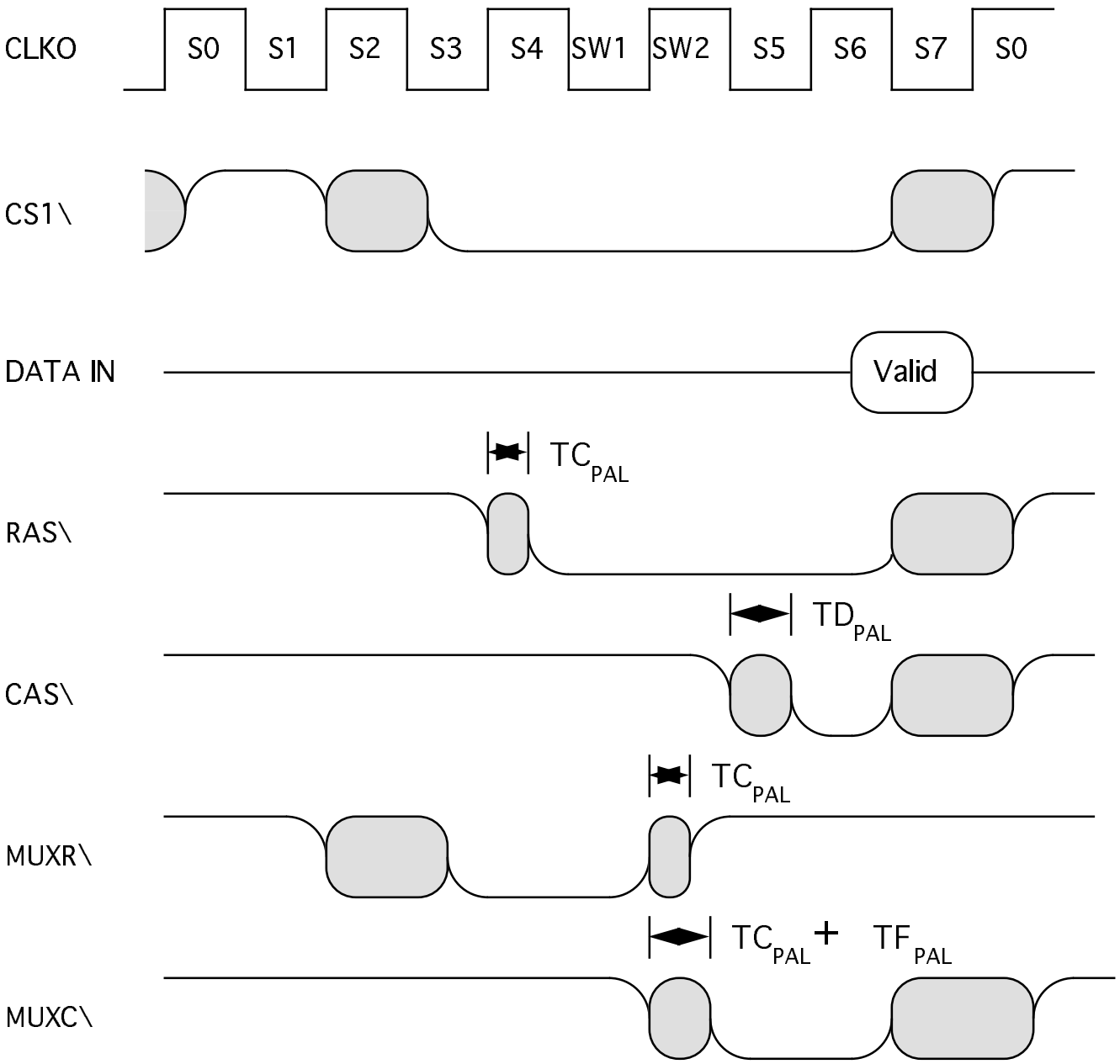


Figure 1

Freescale Semiconductor, Inc.

For More Information On This Product,
Go to: www.freescale.com

PAL Timing



TD_{PAL} = PAL Delay from clock to output

TC_{PAL} = PAL Delay asynchronous input to output

TF_{PAL} = PAL Delay to feedback output

Figure 2

For More Information On This Product,
Go to: www.freescale.com

Freescale Semiconductor, Inc.

Design Diagram

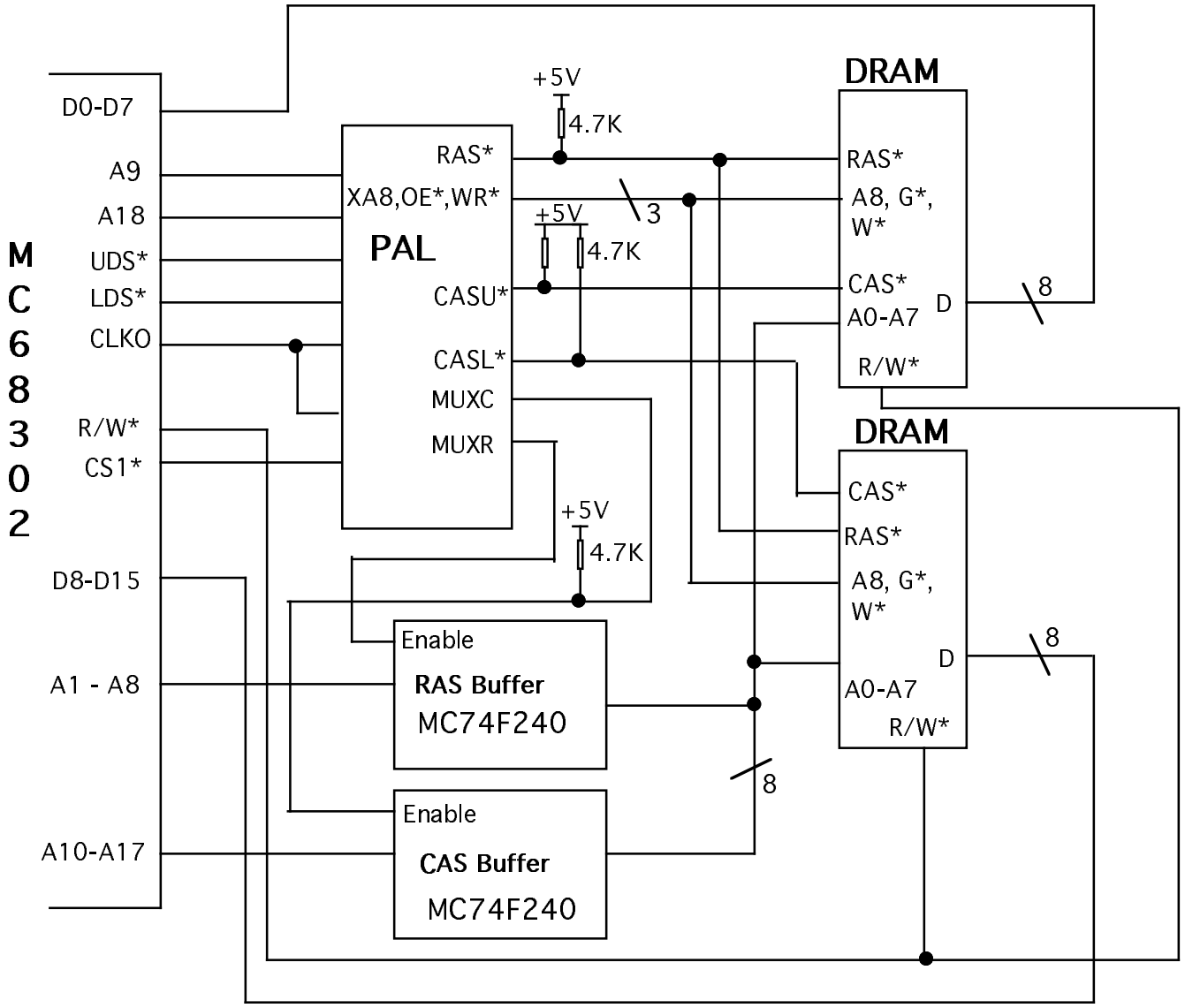


Figure 3

For More Information On This Product,
Go to: www.freescale.com

Freescale Semiconductor, Inc.