Magellan™ Motion Processor

MC55000 Electrical Specifications

for Pulse and Direction Motion Processors

P M D

Performance Motion Devices, Inc.
55 Old Bedford Road
Lincoln, MA 01773

Revision 2.0, December 2006

NOTICE

This document contains proprietary and confidential information of Performance Motion Devices, Inc., and is protected by federal copyright law. The contents of this document may not be disclosed to third parties, translated, copied, or duplicated in any form, in whole or in part, without the express written permission of PMD.

The information contained in this document is subject to change without notice. No part of this document may be reproduced or transmitted in any form, by any means, electronic or mechanical, for any purpose, without the express written permission of PMD.

Warranty

PMD warrants performance of its products to the specifications applicable at the time of sale in accordance with PMD's standard warranty. Testing and other quality control techniques are utilized to the extent PMD deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

Performance Motion Devices, Inc. (PMD) reserves the right to make changes to its products or to discontinue any product or service without notice, and advises customers to obtain the latest version of relevant information to verify, before placing orders, that information being relied on is current and complete. All products are sold subject to the terms and conditions of sale supplied at the time of order acknowledgement, including those pertaining to warranty, patent infringement, and limitation of liability.

Safety Notice

Certain applications using semiconductor products may involve potential risks of death, personal injury, or severe property or environmental damage. Products are not designed, authorized, or warranted to be suitable for use in life support devices or systems or other critical applications. Inclusion of PMD products in such applications is understood to be fully at the customer's risk.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards must be provided by the customer to minimize inherent procedural hazards.

Disclaimer

PMD assumes no liability for applications assistance or customer product design. PMD does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of PMD covering or relating to any combination, machine, or process in which such products or services might be or are used. PMD's publication of information regarding any third party's products or services does not constitute PMD's approval, warranty or endorsement thereof.

Related Documents

Magellan Motion Processor User's Guide

Complete description of the Magellan Motion Processor features and functions with detailed theory of its operation.

Magellan Motion Processor Programmer's Command Reference

Descriptions of all Magellan Motion Processor commands, with coding syntax and examples, listed alphabetically for quick reference.

Pro-Motion User's Guide

User's guide to Pro-Motion, the easy-to-use motion system development tool and performance optimizer. Pro-Motion is a sophisticated, easy-to-use program which allows all motion parameters to be set and/or viewed, and allows all features to be exercised.

Magellan Motion Processor Developer's Kit Manual

How to install and configure the DK58000 series and DK55000 series developer's kit PC board.

Other Documents

Magellan Motion Processor Electrical Specifications

Contains physical and electrical characteristics, timing diagrams, pinouts, and pin descriptions for MC58000 Series, for DC brush, brushless DC, Microstepping, and Pulse & Direction motion processors.

ION Digital Drive User's Manual

How to install and configure ION Digital Drives.

Prodigy-PC/104 Motion Card User's Guide

How to install and configure the Prodigy-PC/104 motion board.

Prodigy-PCI Motion Card User's Guide

How to install and configure the Prodigy-PCI motion board.

Table of Contents

List of Figures vii

1. The MC50000 Family **9**
1.1 Introduction 10
1.2 Family Summary 10
1.3 How to Order 11

2. Functional Characteristics **13**
2.1 Configurations, Parameters, and Performance 13
2.2 Physical Characteristics and Mounting Dimensions 15
2.3 Absolute Maximum Environmental and Electrical Ratings 16
2.4 MC55110 System Configuration — Single Chip, 1 Axis Control 17
2.5 MC55x20 System Configuration — Two Chip, 1 to 4 Axis Control 18

3. Electrical Characteristics **19**
3.1 DC Characteristics for 55110, 55x20 CP 19
3.2 DC Characteristics for 55x20 IO 20
3.3 AC Characteristics 20

4. I/O Timing Diagrams **23**
4.1 Clock 23
4.2 Quadrature Encoder Input 23
4.3 Reset 24
4.4 Host Interface, 8/16 Mode 24
4.5 Host Interface, 16/16 Mode 26
4.6 External Memory Timing 28
4.7 Peripheral Device Timing 30

5. Pinouts and Pin Descriptions **33**
5.1 Pinouts for the MC55110 33
5.2 Pinouts for the MC55420 38
5.3 Pinouts for the MC55320 39
5.4 Pinouts for the MC55220 40
5.5 Pinouts for the MC55120 41

6. Application Notes — MC55110 and MC55x20 .... **49**
6.1 General Design Notes 49
6.2 Design Tips 50
6.3 Peripheral Device Address Map 51
6.4 Device Initialization 51
6.5 Power Supplies 52
6.6 Clock Generator, Grounding and Decoupling, and Device Reset 55
6.7 Serial Communication Interface (SCI) 60
6.8 CAN Communication Interface 64
6.9 External Memory 66
6.10 Asynchronous SRAM 66
6.11 Dual Port Synchronous SRAM (DPRAM) 70
6.12 Using the On-chip ADC 76
6.13 User I/O Space 82
6.14 Parallel Communication Interface 84
6.15 Overcurrent and Emergency Braking Circuits for Motor Drivers 88
6.16 Using the Allegro A3977 to Drive Microstepping Motors 90

This page intentionally left blank.

List of Figures

2-1 CP chip (all dimensions in millimeters) 15
2-2 IO Chip (all dimensions in millimeters) 16
2-3 MC55110 control and data paths 17
2-4 MC55x20 control and data paths 18
4-1 Clock timing 23
4-2 Quad encoder timing 23
4-3 Reset timing 24
4-4 Instruction write, 8/16 mode 24
4-5 Data write, 8/16 mode 25
4-6 Data read, 8/16 mode 25
4-7 Status read, 8/16 mode 26
4-8 Instruction write, 16/16 mode 26
4-9 Data write, 16/16 mode 27
4-10 Data read, 16/16 mode 27
4-11 Status read, 16/16 mode 28
4-12 External memory read 28
4-13 External memory write 29
4-14 Peripheral device read 30
4-15 Peripheral device write 31
5-1 MC55110 pinouts 33
5-2 MC55420 pinouts 38
5-3 MC55320 pinouts 39
5-4 MC55220 pinouts 40
5-5 MC55120 pinouts 41
6-1 Basics, power supplies, 55000 53
6-2 Basics, clock and bypass caps, 55110 56
6-3 Basics, clock and bypass caps, 55420 57
6-4 Oscillator filter circuit 58
6-5 Basics, reset 59
6-6 Host communication, SCI 61
6-7 Host communication, RS232 and RS485/422 63
6-8 Host communication, CAN bus 65
6-9 External memory, SRAM 512Kx16 67
6-10 External memory, Ready signal generator 69
6-11 External memory, DPRAM, fast flow-through 71
6-12 External memory, DPRAM, flow-through with clock signal control 74
6-13 External memory, DPRAM, RCLK timing 75
6-14 External memory, DPRAM, host management 77
6-15 ADC, single-ended temperature sensor 79
6-16 ADC, differential interface 81
6-17 User I/O space, red LED 83
6-18 Host communication, parallel interface, 16/16 mode 85
6-19 Host communication, parallel interface, 8/16 mode 87
6-20 Motor driver protection circuitry 89
6-21 Pulse and directon, A3977 91

This page intentionally left blank.

1. The MC50000 Family

In This Chapter

- Introduction
- Family Summary
- How to Order

	MC55x20 Series	**MC58x20 Series**	**MC55110**	**MC58110**
Number of axes	4,3,2 or 1	4,3,2 or 1	1	1
Number of chips	2 (CP and IO)	2 (CP and IO)	1 (CP)	1 (CP)
Motor type	Stepping	DC Brush servo Brushless DC servo Stepping	Stepping	DC Brush servo Brushless DC servo Stepping
Output format	Pulse and Direction	Brushed single phase Sinusoidal commutation Microstepping Pulse and direction	Pulse and Direction	Brushed single phase Sinusoidal commutation Microstepping Pulse and direction
Communication interface				
Parallel	✓	✓	✓	✓
Asynchronous serial	✓	✓	✓	✓
CAN 2.0B	✓	✓	✓	✓
Position input				
Incremental encoder input	✓	✓	✓	✓
Parallel word device input	✓	✓	✓	✓
Index & Home signals	✓	✓	✓	✓
Position capture	✓	✓	✓	✓
Directional limit switches	✓	✓	✓	✓
Motor command output				
PWM output	–	✓	–	✓
Parallel DAC output	–	✓	–	✓
SPI DAC output	–	✓	–	✓
Pulse & Direction output	✓	✓	✓	✓
Trajectory generation				
Trapezoidal profiling	✓	✓	✓	✓
S-curve profiling	✓	✓	✓	✓
Velocity profiling	✓	✓	✓	✓
Electronic gearing	✓	✓	✓	✓
On-the-fly changes	✓	✓	✓	✓
Servo filter				
PID position loop	–	✓	–	✓
Dual encoder loop	–	✓	–	–
Derivative sampling time	–	✓	–	✓
Feedforward (accel & vel)	–	✓	–	✓
Dual bi-quad filter	–	✓	–	✓

	MC55x20 Series	MC58x20 Series	MC55110	MC58110
Miscellaneous				
Data trace/diagnostics	✓	✓	✓	✓
Motion error detection	✓ (with encoder)	✓	✓ (with encoder)	✓
Axis settled indicator	✓ (with encoder)	✓	✓ (with encoder)	✓
Analog input	✓	✓	✓	✓
Programmable bit output	✓	✓	✓	✓
Software-invertible signals	✓	✓	✓	✓
User-defined I/O	✓	✓	✓	✓
External RAM support	✓	✓	✓	✓
Multi-chip synchronization	✓	✓	✓	✓
Chipset part numbers	MC55120 MC55220 MC55320 MC55420	MC58120 MC58220 MC58320 MC58420	MC55110	MC58110
Developer's Kit part numbers	DK55420	DK58420	DK55110	DK58110

1.1 Introduction

This manual describes the operational characteristics of the MC55000 Series Motion Processors from PMD. These devices are members of PMD's third-generation motion processor family.

Each of these devices is a complete chip-based motion processor. They provide trajectory generation and related motion control functions. Depending on the type of motor controlled, they provide servo-loop closure, on-board commutation for brushless motors, and high-speed pulse and direction outputs. Together, these products provide a software-compatible family of dedicated motion processors which can handle a large variety of system configurations.

Each of these chips utilizes a similar architecture, consisting of a high-speed computation unit, along with an ASIC (Application Specific Integrated Circuit). The computation unit contains special on-board hardware which makes it well suited for the task of motion control.

Having similar hardware architecture enables the MC55000 family of chips to share most software commands, so that software written for one series may be re-used with another; even though the type of motor may be different.

1.2 Family Summary

MC55000 Series – The MC55000 chipsets provide high-speed pulse and direction signals for step motor systems. For the MC55x20 series, two TQFP ICs are required: a 100-pin Input/Output (IO) chip, and a 144-pin Command Processor (CP) chip. The MC55110 has all functions integrated into one 144-pin Command Processor (CP) chip.

For the MC55x20 Series, two TQFP ICs are required: a 100-pin Input/Output (IO) chip, and a 144-pin Command Processor (CP) chip; while the MC55110 has all functions integrated into a single 144-pin CP chip.

MC58000 Series – This series outputs motor commands in Sign/Magnitude PWM or DAC-compatible format for use with DC-Brush motors or Brushless DC motors having external commutation; two-phase or three-phase sinusoidally commutated motor signals in PWM or DAC-compatible format for brushless servo motors; pulse and direction output for step motors; and two phase signals per axis in either PWM or DAC-compatible signals for microstepping motors.

1.3 How to Order

When ordering a single-chip configuration, only the CP part number is necessary. For two-IC and multi-axis configurations, both the CP and the IO part numbers are required.

CP (1 or 2 chip configurations)

MC5□□□0CP□.□.G

Motor Type
8 = Multi Motor
5 = Pulse & Direction

# Axes
1,2,3,4

# Chips
1 (CP only)
2 (CP & IO)

CP Version
(Call PMD)

IO (2 chip configurations only)

MC50000IOAD8.G

Developer's Kit

DK5□□□0CP□.□IOAD8.R

Motor Type
8 = Multi Motor
5 = Pulse & Direction

# Axes
1,2,3,4

# Chips
1 (CP only)
2 (CP & IO)

CP Version
(Call PMD)

This page intentionally left blank.

2. Functional Characteristics

In This Chapter

- Configurations, Parameters, and Performance
- Physical Characteristics and Mounting Dimensions
- Absolute Maximum Environmental and Electrical Ratings — CP 55110, 55x20
- Absolute Maximum Environmental and Electrical Ratings — IO 55x20
- System Configuration — Single Chip, 1 Axis Control
- MC55x20 System Configuration — Two Chip, 1 To 4 Axis Control

2.1 Configurations, Parameters, and Performance

Configuration	4 axes (MC55420) 3 axes (MC55320) 2 axes (MC55220) 1 axis (MC55120 or MC55110)	
Operating mode	Open loop (pulse generator is driven by trajectory generator output, encoder input used for stall detection)	
Communication modes	8/16 parallel	8-bit external parallel bus with 16-bit command word size
	16/16 parallel	16-bit external parallel bus with 16-bit command word size
	Point-to-point asynchronous serial	
	Multi-drop asynchronous serial	
	CAN bus 2.0B, protocol co-exists with CANOpen, 11-bit identifier	
Serial port baud rate range	1,200 baud to 460,800 baud	
CAN port transmission rate range	10,000 baud to 1,000,000 baud	
Profile modes	S-curve point-to-point	Position, velocity, acceleration, deceleration, and jerk parameters
	Trapezoidal point-to-point	Position, velocity, acceleration, and deceleration parameters
	Velocity-contouring	Velocity, acceleration, and deceleration parameters
	Electronic Gear	Encoder or trajectory position of one axis used to drive a second axis. Master and slave axes and gear ratio parameters.
Position range	-2,147,483,648 to +2,147,483,647 counts or steps	
Velocity range	-32,768 to +32,767 counts or steps per cycle with a resolution of 1/65,536 counts or steps per cycle	
Acceleration and deceleration ranges	0 to +32,767 counts or steps per cycle^2 with a resolution of 1/65,536 counts or steps per cycle^2	
Jerk range	0 to ½ counts or steps per cycle^3 with a resolution of 1/4,294,967,296 counts or steps per cycle^3	
Electronic gear ratio range	-32,768 to +32,767 with a resolution of 1/65,536 (negative and positive direction)	
Position error	32 bits	

Position error tracking	Motion error window	Allows axis to be stopped upon exceeding programmable window
	Tracking window	Allows flag to be set if axis exceeds a programmable position window
	Axis settled	Allows flag to be set if axis exceeds a programmable position window for a programmable amount of time after trajectory motion is complete
Motor output mode	Pulse and direction	MC55x20: 4.98 Mpulses/sec maximum MC55110: 97.6 kpulses/sec maximum
Maximum encoder rate	Incremental (up to 8 Mcounts/sec) Parallel-word (up to 160 Mcounts/sec)	
Parallel encoder word size	16 bits	
Parallel encoder read rate	20 kHz (reads all axes every 50 μsec)	
Cycle timing range	51.2 microseconds to 1.048576 seconds	
Minimum cycle time (MC55110 only)	51.2 microseconds	
Multi-chip synchronization	<1 μsec difference between master and slave servo cycle	
Limit switches	2 per axis: one for each direction of travel	
Position-capture triggers	2 per axis: index and home signals	
Other digital signals (per axis)	1 *AxisIn* signal per axis, 1 *AxisOut* signal per axis	
Software-invertible signals	*Encoder A, Encoder B, Index, Home, AxisIn, AxisOut, PositiveLimit, NegativeLimit, Pulse, Direction*	
Analog input	8 10-bit analog inputs	
User defined discrete I/O	256 16-bit width user defined I/O	
RAM/external memory support	65,536 blocks of 32,768 16-bit words per block. Total accessible memory is 2,147,483,648 16-bit words.	
Trace modes	one-time, continuous	
Maximum number of trace variables	4	
Number of traceable variables	31	

2.2 Physical Characteristics and Mounting Dimensions

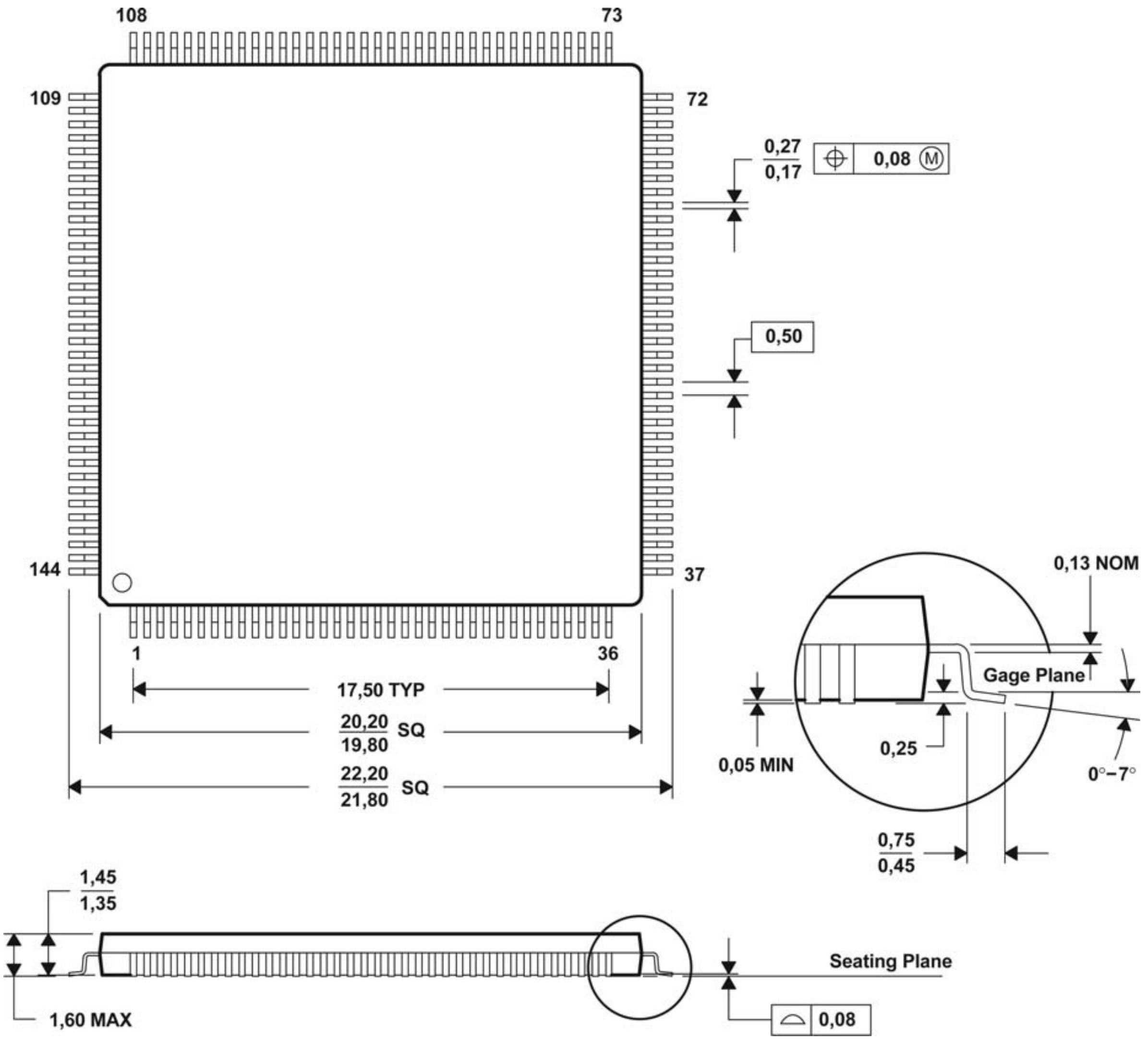
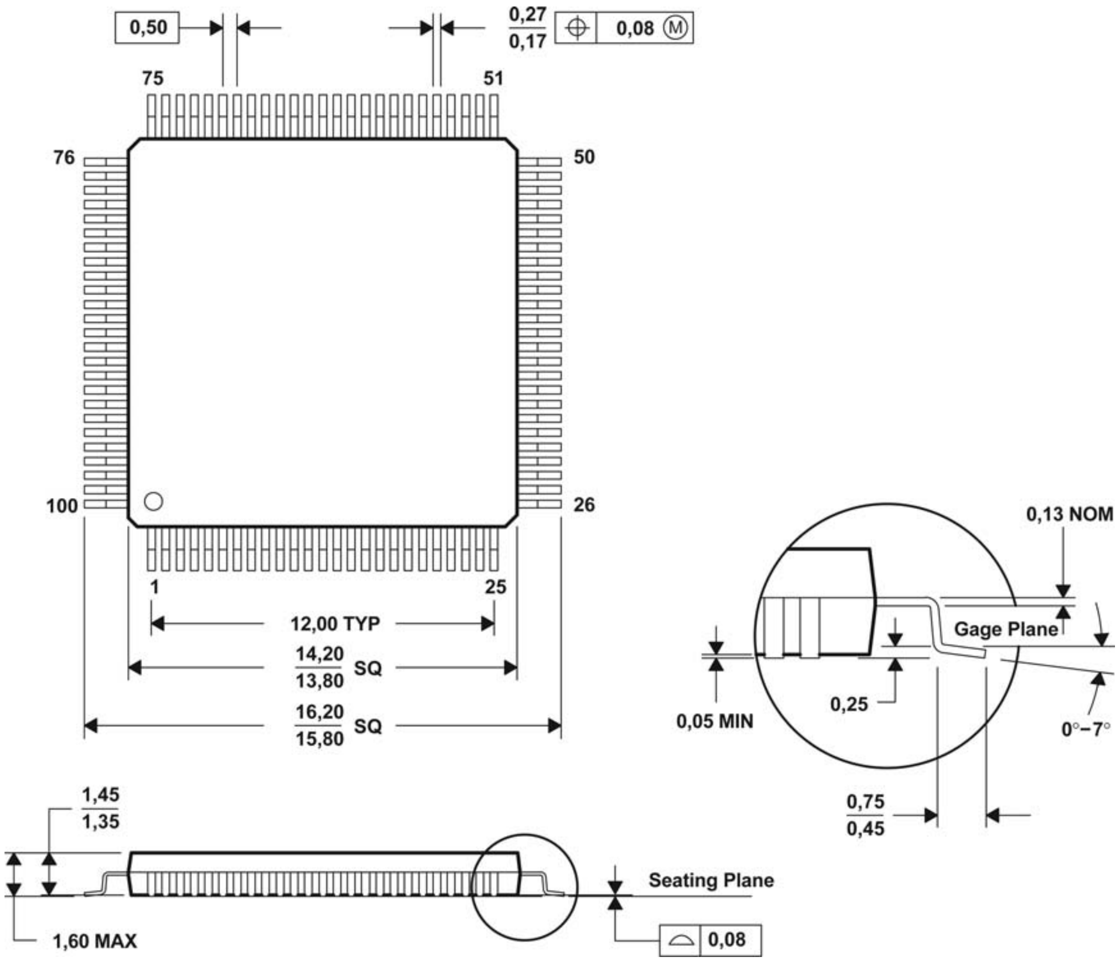


Figure 2-1: CP chip (all dimensions in millimeters)

Figure 2-2: IO Chip (all dimensions in millimeters)



2.3 Absolute Maximum Environmental and Electrical Ratings

2.3.1 CP 55110, 55x20

Supply Voltage (Vcc)	-0.3V to +4.6V
Vcc5 range	-0.3V to +5.5V
Input voltage (Vi)	-0.3V to +4.6V
Package thermal impedance (θ_{JA})	32°C/W
Junction temperature range (Tj)	-40°C to 150°C
Storage Temperature (Ts)	-65°C to 150°C
Nominal Clock Frequency (Fclk)	20.0 MHz

2.3.2 IO 55x20

Supply Voltage (Vcc)	-0.5V to +3.6V
Input voltage (Vi)	-0.5V to Vcc +0.5V
Package thermal impedance (θ_{JA})	39.7°C/W
Junction temperature range (Tj)	-40°C to 150°C
Storage Temperature (Ts)	-65°C to 150°C
Nominal Clock Frequency (Fclk)	40.0 MHz

2.4 MC55110 System Configuration — Single Chip, 1 Axis Control

The following figure shows the principal control and data paths in an MC55110 system.

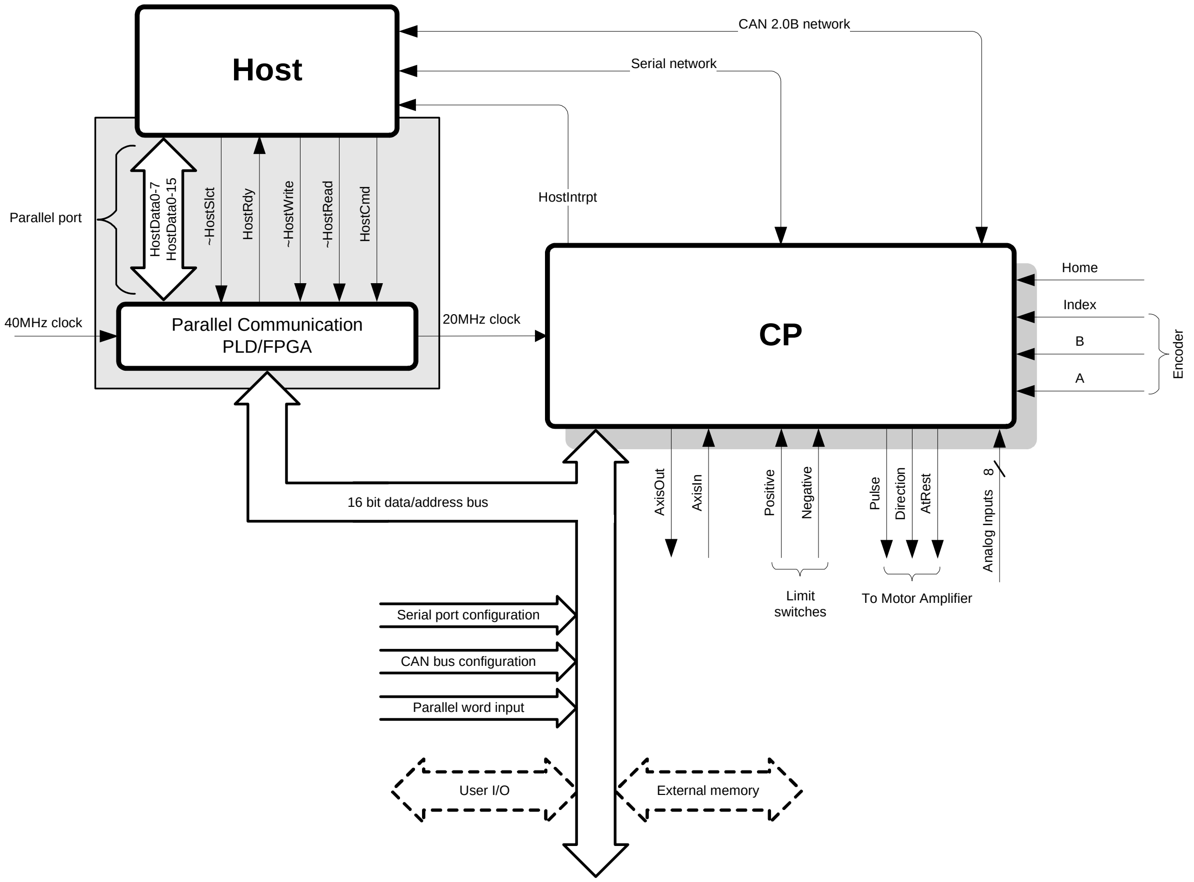


Figure 2-3: MC55110 control and data paths

The CP chip is a self-contained motion processor. In addition to handling all system functions, the CP chip contains the profile generator, which calculates position, velocity, acceleration, and values for a trajectory. Then, the CP chip generates pulse and direction signals. Optional axis position information returns to the motion processor in the form of encoder feedback using either the incremental encoder input signals, or via the bus as parallel word input.

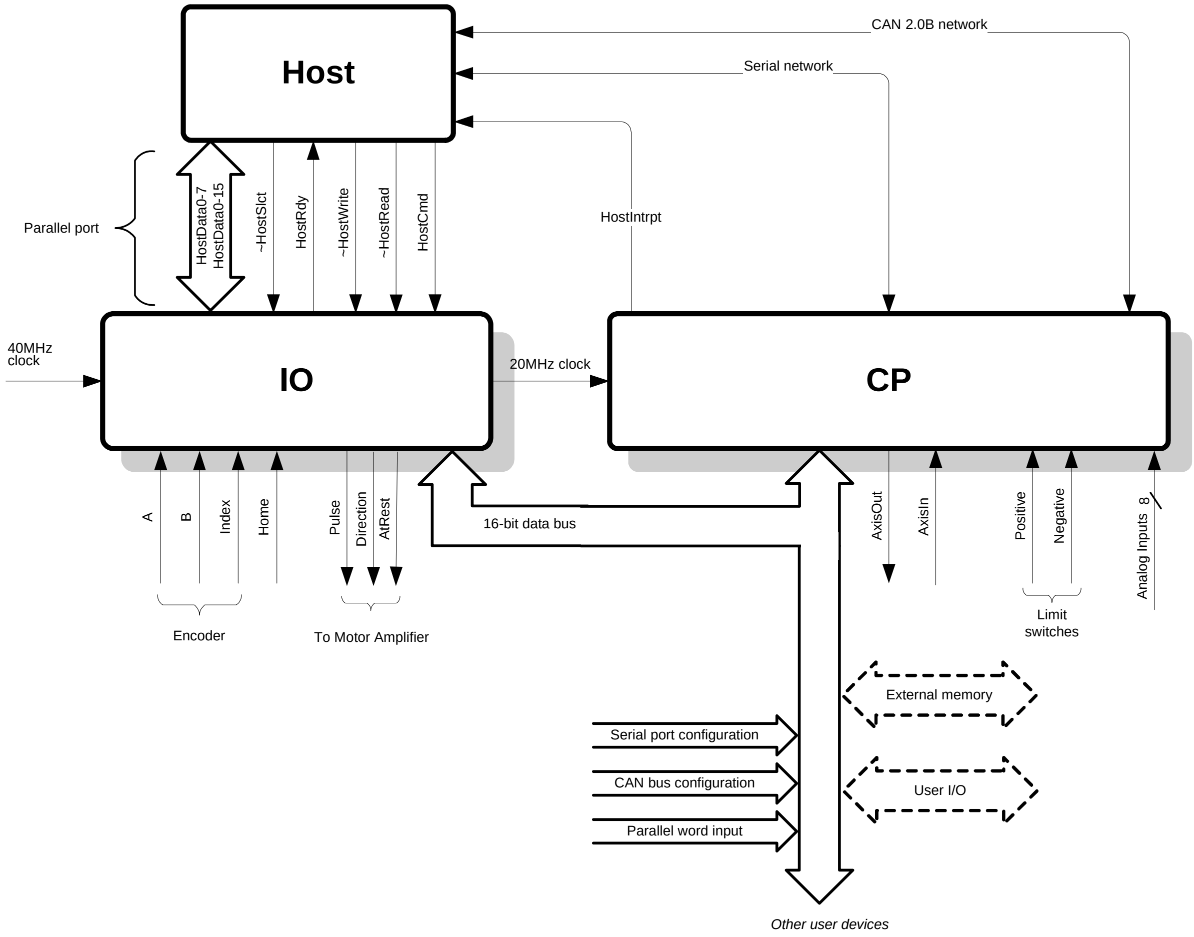
The MC55110 can co-exist in a CANOpen network as a slave device. It is CAN 2.0B compliant.

The shaded area shows the PLD/FPGA that must be provided by the designer if parallel communication is required. For a description and an example of the necessary logic (in schematic format) contact PMD.

2.5 MC55x20 System Configuration — Two Chip, 1 to 4 Axis Control

The following figure shows the principal control and data paths in an MC55x20 system.

Figure 2-4: MC55x20 control and data paths



The IO chip contains the parallel host interface, the incremental encoder input along with the pulse and direction motor output signals.

The CP chip contains the profile generator, which calculates position, velocity, acceleration, and values for a trajectory and communicates the results to the IO chip for output. Optional axis position information returns to the motion processor in the form of encoder feedback using either the incremental encoder input signals, or via the bus as parallel word input.

The MC55x20 can co-exist in a CANOpen network as a slave device. It is CAN 2.0B compliant.

3. Electrical Characteristics

In This Chapter

- DC Characteristics for 55110, 55x20 CP
- DC Characteristics for 55x20 IO
- AC Characteristics

3.1 DC Characteristics for 55110, 55x20 CP

(V_{cc} and T_a per operating ratings, F_{clk} = 20.0 MHz)

Symbol	Parameter	Minimum	Maximum	Conditions
V_{cc}	Supply Voltage	3.0V	3.6V	
I_{dd}	Supply Current		120 mA	All I/O pins are floating
T_a	Operating free-air temperature	-40°C	85°C	
Input Voltages				
V_{ih}	Logic 1 input voltage	2.0V	V_{cc} + 0.3V	
V_{il}	Logic 0 input voltage	0	0.8V	
Output Voltages				
V_{oh}	Logic 1 Output Voltage	2.4V		Io = -2 mA
V_{ol}	Logic 0 Output Voltage		0.4V	Io = 2 mA
Other				
I_{out}	Tri-State output leakage current	-2 µA	2 µA	Vin = 0 or Vcc
I_{in}	Input current	-30 µA	30 µA	
C_{io}	Input/Output capacitance		2/3 pF	typical
Analog Input				
Z_{ai}	Analog input source impedance		1.4 kΩ	
I_a	Analog supply current		22 mA	
I_{refhi}	Vrefhi input current		1.5 mA	
C_{ai}	Analog input capacitance		30 pF	typical
E_{zo}	Zero-offset error		±2 LSB	typical
E_{dnl}	Differential nonlinearity error. Difference between the step width and the ideal value.		±2 LSB	
E_{inl}	Integral nonlinearity error. Maximum deviation from the best straight line through the ADC transfer characteristics, excluding the quantization error.		±2 LSB	

3.2 DC Characteristics for 55x20 IO

(Vcc and Ta per operating ratings, Fclk = 40.0 MHz)

Symbol	Parameter	Minimum	Maximum	Conditions
Vcc	Supply Voltage	3.0V	3.6V	
Idd	Supply Current		24 mA	All I/O pins are floating
Ta	Operating free-air temperature	-40°C	85°C	
Input Voltages				
Vih	Logic 1 input voltage	2.0V	Vcc	
Vil	Logic 0 input voltage	0	0.8V	
Output Voltages				
Voh	Logic 1 Output Voltage	2.4V		Io = -2 mA
Vol	Logic 0 Output Voltage		0.4V	Io = 6 mA
Other				
Iout	Tri-State output leakage current	-10 µA	10 µA	
Iin	Input current	-10 µA	10 µA	
Cio	Input/Output capacitance		7/7 pF	typical

3.3 AC Characteristics

See timing diagrams in Chapter 4, "I/O Timing Diagrams," for *Tn* numbers. The symbol "~" indicates active low signal.

Timing Interval	*Tn*	Minimum	Maximum
Clock			
IOClkIn Frequency (Fclk)[1]		8 MHz	40 MHz
IOClkIn pulse duration[3]	T1a	0.4 T2a	0.6 T2a
IOClkIn Period	T2a	25 nsec	125 nsec
CPClkIn Frequency (Fclk)[1]		4 MHz	20 MHz
CPClkIn pulse duration[3]	T1b	0.4 T2b	0.6 T2b
CPClkIn Period	T2b	50 nsec	250 nsec
CPClkIn rise/fall time	T58		5 nsec
Encoder			
Encoder Pulse Width	T3	200 nsec	
Dwell Time Per State	T4	100 nsec	
Index Setup and Hold (relative to Quad A and Quad B low)	T5	0 nsec	
Host IO			
~HostSlct Hold Time	T6	0 nsec	
~HostSlct Setup Time	T7	0 nsec	
HostCmd Setup Time	T8	0 nsec	
HostCmd Hold Time	T9	0 nsec	
Read Data Access Time	T10		25 nsec
Read Data Hold Time	T11		10 nsec
~HostRead High to HI-Z Time	T12		20 nsec
HostRdy Hold Time	T13	40 nsec	70 nsec
~HostWrite Pulse Width	T14	70 nsec	
Write Data Delay Time	T15		15 nsec
Write Data Hold Time	T16	0 nsec	
Read Recovery Time[2]	T17	60 nsec	
Write Recovery Time[2]	T18	60 nsec	
~HostRead Pulse Width	T19	70 nsec	

Timing Interval	*Tn*	Minimum	Maximum
External Memory Read			
ClockOut low to control valid	T20		4 nsec
ClockOut low to address valid	T21		8 nsec
Address valid to ~ReadEnable low	T22	5.5 nsec	
ClockOut high to ~ReadEnable low	T23		5 nsec
ClockOut low to ~ReadEnable high	T23a	-8 nsec	1 nsec
Data access time from Address valid	T24		40 nsec
Data access time from ~ReadEnable low	T25		31 nsec
Data setup time before ~ReadEnable high	T25a	8 nsec	
Data hold time after ~ReadEnable high	T26	0 nsec	
~ReadEnable high to Address invalid	T26a	0 nsec	
ClockOut low to control inactive	T27		5 nsec
Address hold time after ClockOut low	T28	2 nsec	
ClockOut low to Strobe low	T29		5 nsec
ClockOut low to Strobe high	T30		6 nsec
W/~R low to R/~W rising delay time	T31		5 nsec
External Memory Write			
ClockOut high to control valid	T32		4 nsec
ClockOut high to address valid	T33		10 nsec
Address valid to ~WriteEnable low	T34	3.5 nsec	
ClockOut low to ~WriteEnable low	T35		6 nsec
ClockOut low to ~WriteEnable high	T35a		6 nsec
Data setup time before ~WriteEnable high	T36	33 nsec	
Data bus driven from ClockOut low	T37	-3 nsec	
Data hold time after ~WriteEnable high	T38	2 nsec	
ClockOut high to control inactive	T39		5 nsec
Address hold time after ClockOut low	T40	-5 nsec	
ClockOut low to Strobe low	T41		6 nsec
ClockOut low to Strobe high	T42		6 nsec
R/~W low to W/~R rising delay time	T43		5 nsec
ClockOut high to control valid	T44		6 nsec
ClockOut high to R/~W high	T44a		6 nsec
Peripheral Device Read			
ClockOut high to ClockOut low[4]	T45	112.5 nsec	562.5 nsec
Data access time from Address valid	T46		65 nsec
Data access time from ~ReadEnable low	T47		56 nsec
Peripheral Device Write			
ClockOut low to ClockOut low[4]	T48	125 nsec	625 nsec
Data setup time before ~WriteEnable high	T49	58 nsec	
Device Reset			
Reset low pulse width	T50	400 nsec	
Device Ready/ Outputs Initialized	T57		1.5 msec

1. Performance figures and timing information valid at Fclk = 40.0 MHz for the dual chip configuration and Fclk = 20.0 MHz for the single chip configurations only. For timing information and performance parameters at lower Fclk, see Section 6.2.2, "Using a Non-standard System Clock Frequency."
2. For 8/16 interface modes only.
3. The clock low/high split has an allowable range of 40 - 60%.
4. The minimum and maximum values correspond to a 50 nsec and 250 nsec CPClkIn clock periods, or 25 nsec and 125 nsec IOClkIn clock periods, respectively.

This page intentionally left blank.

4. I/O Timing Diagrams

In This Chapter

- Clock
- Quadrature Encoder Input
- Reset
- Host Interface, 8/16 Mode
- Host Interface, 16/16 Mode
- External Memory Timing
- Peripheral Device Timing

For the values of *Tn*, please refer to the table in Section 3.3, "AC Characteristics."

4.1 Clock

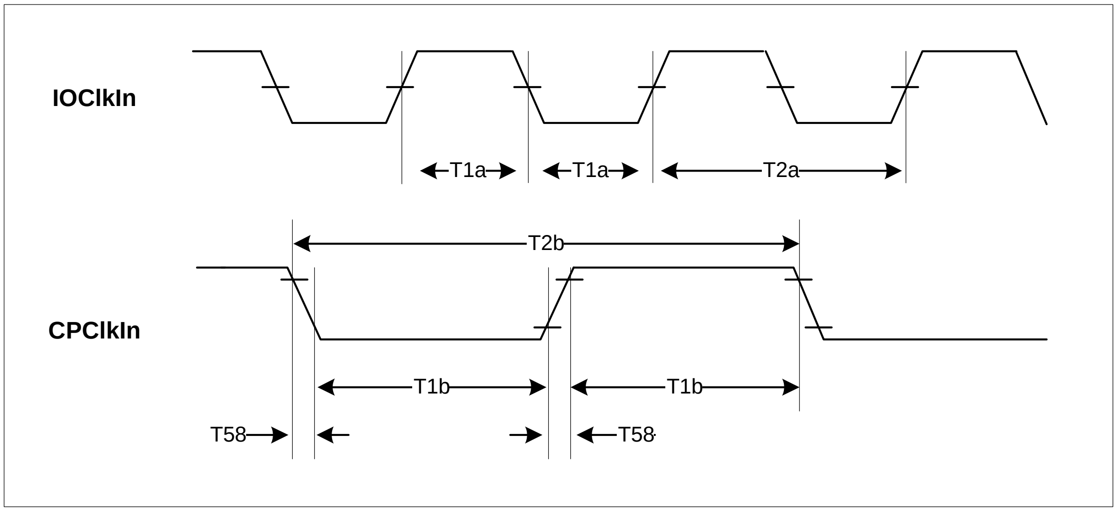


Figure 4-1: Clock timing

4.2 Quadrature Encoder Input

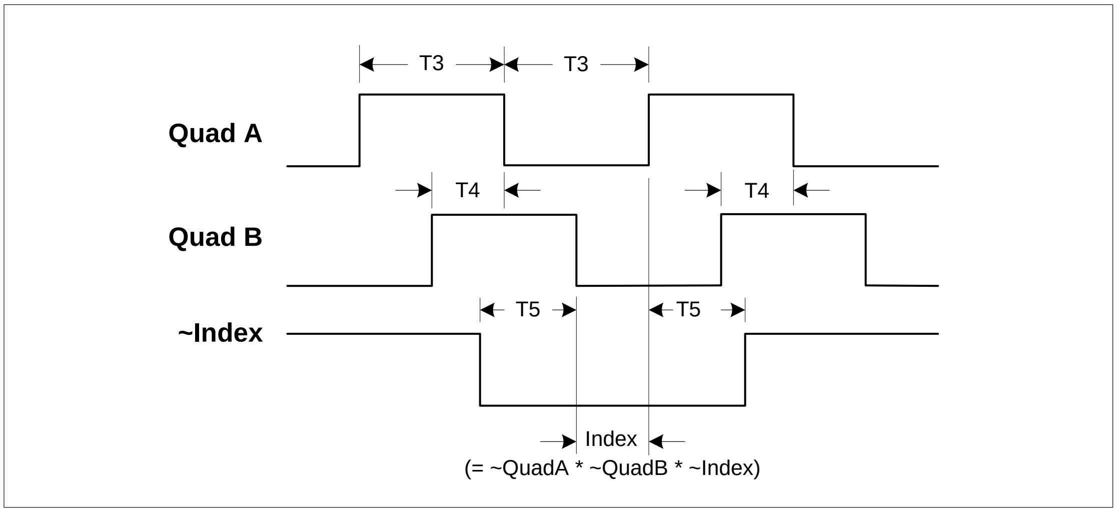
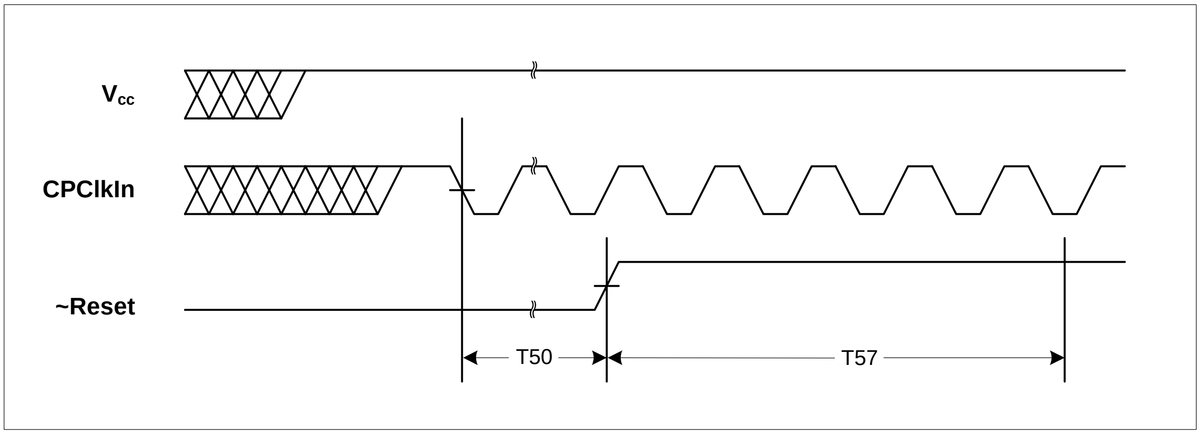


Figure 4-2: Quad encoder timing

4.3 Reset

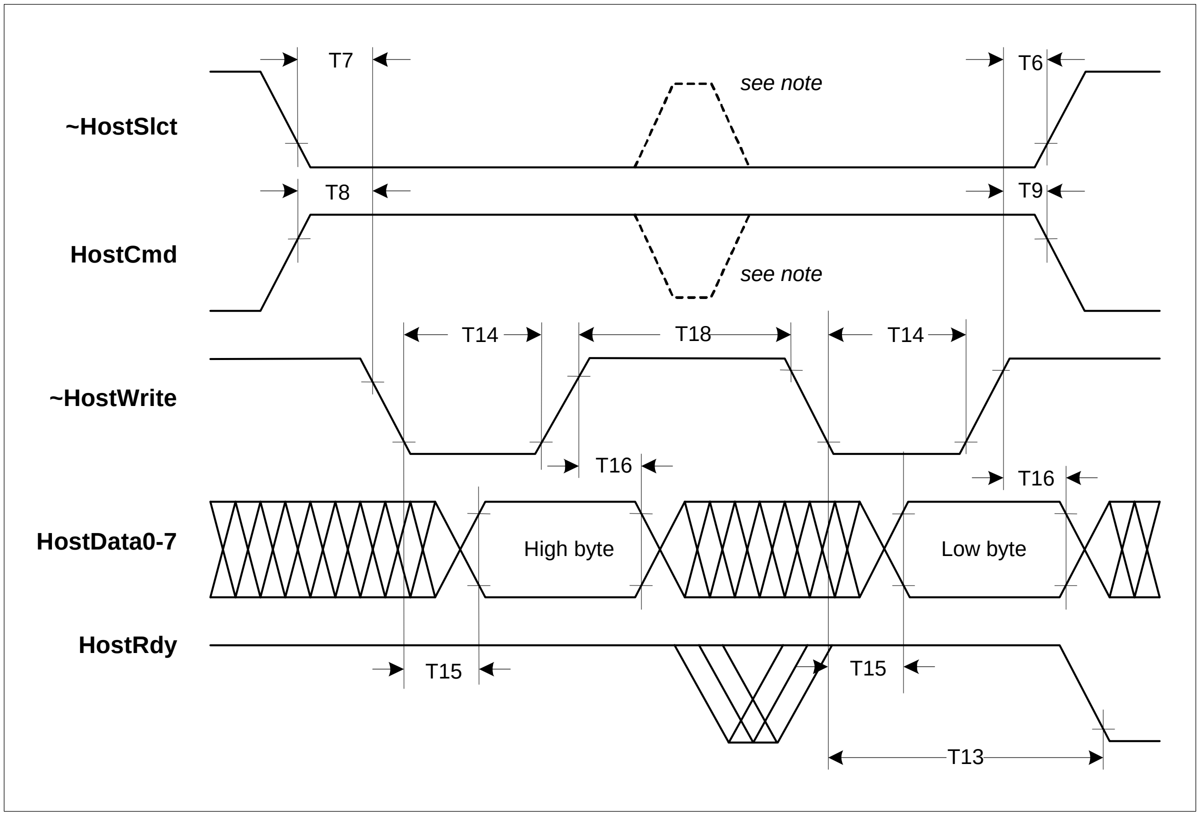
Figure 4-3: Reset timing



NOTE: The device must be reset after power on.

4.4 Host Interface, 8/16 Mode

Figure 4-4: Instruction write, 8/16 mode



NOTE: If setup and hold times are met, ~HostSlct and HostComd may be de-asserted at this point.

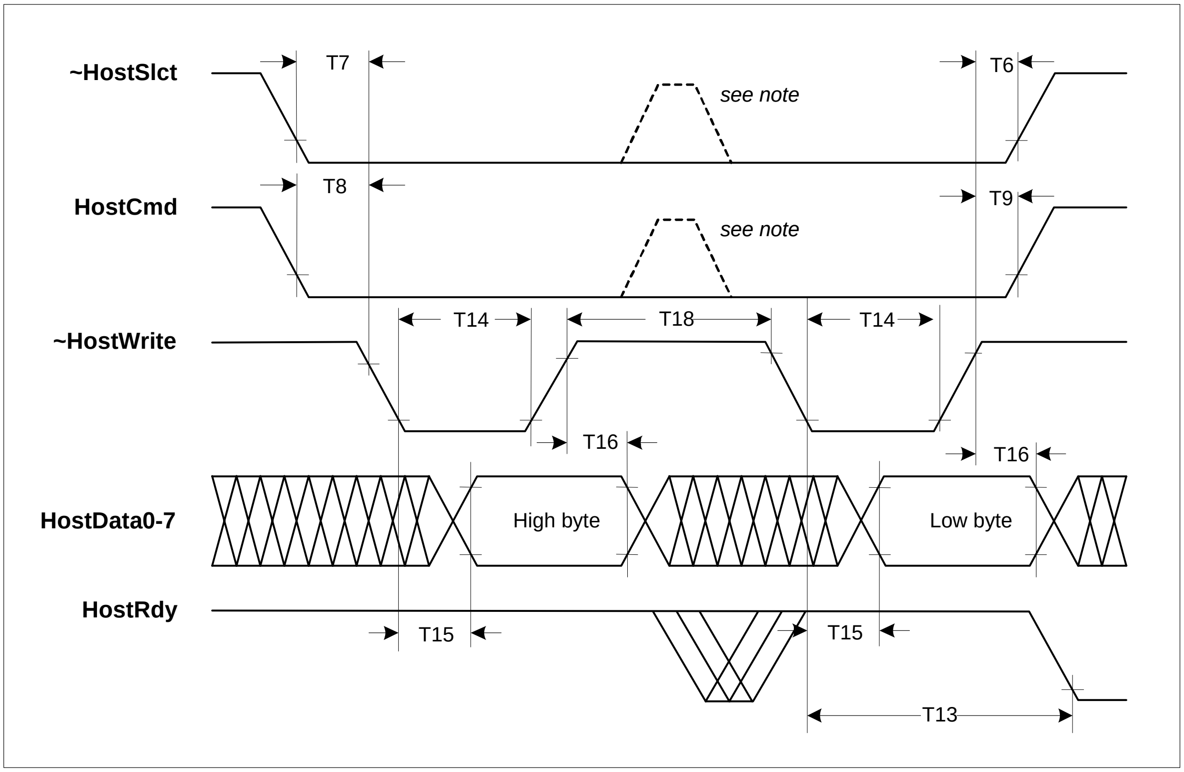


Figure 4-5: Data write, 8/16 mode

NOTE: If setup and hold times are met, ~HostSlct and HostComd may be de-asserted at this point.

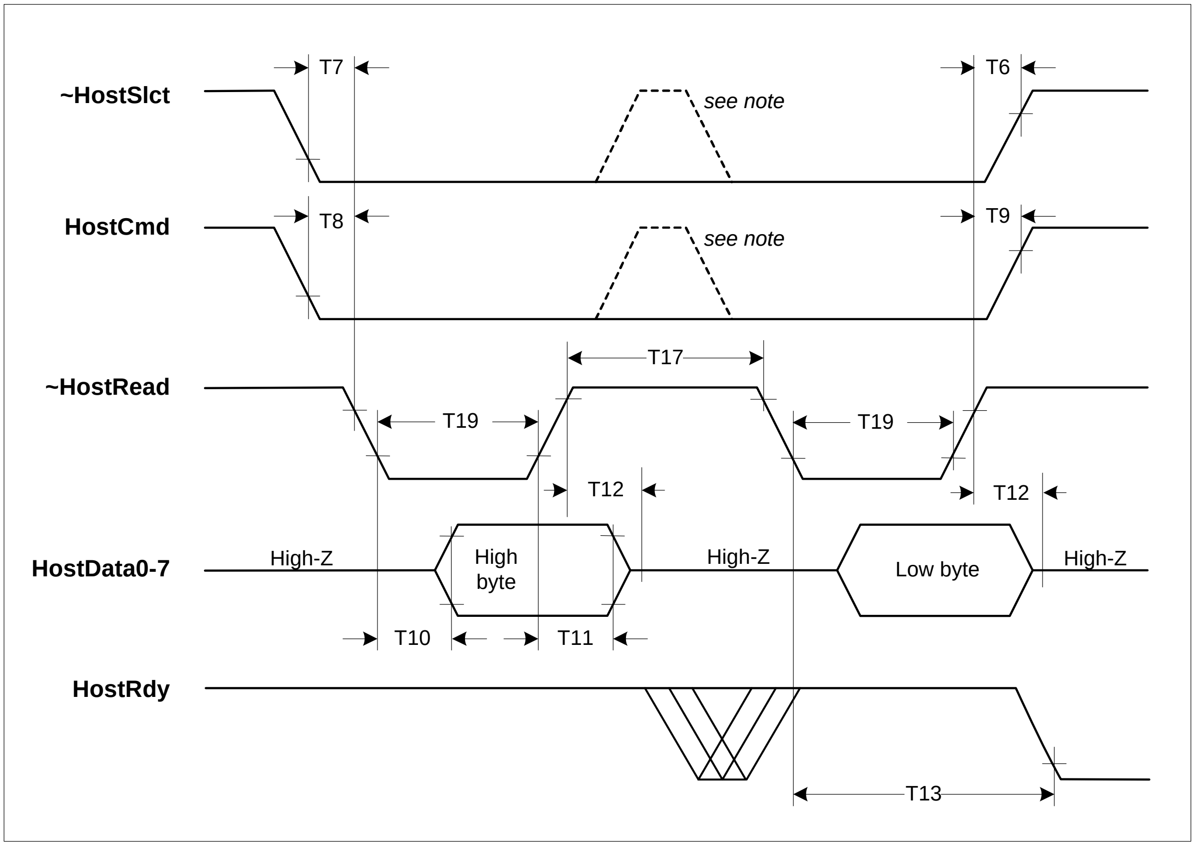
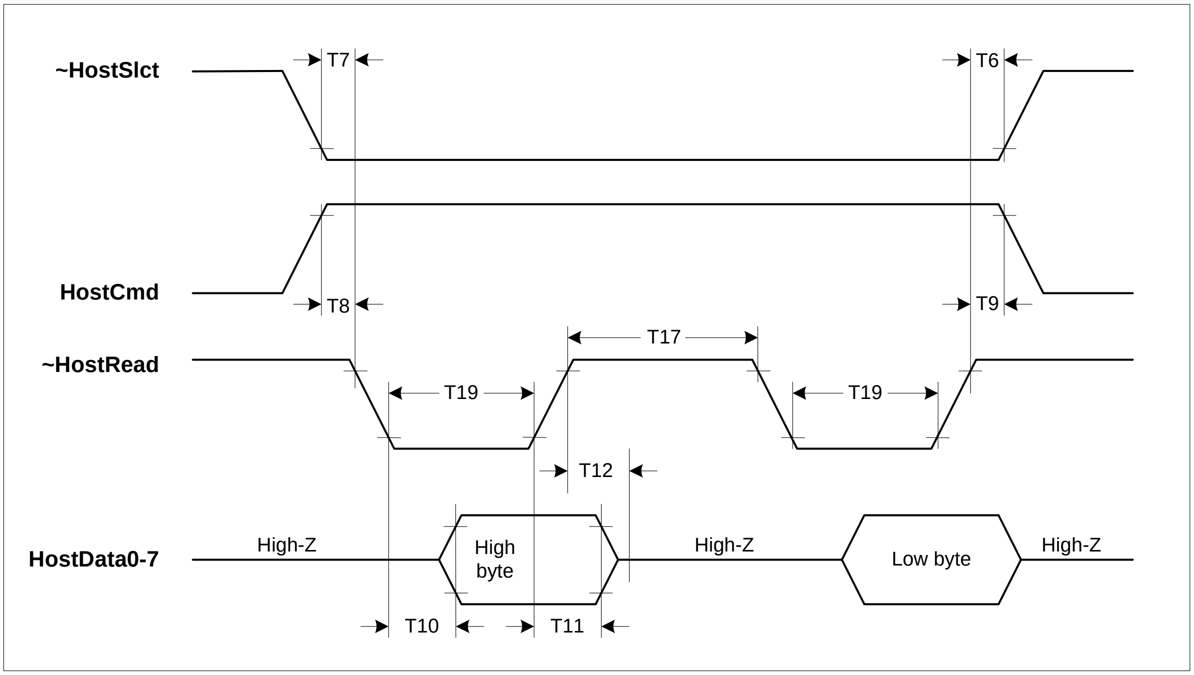


Figure 4-6: Data read, 8/16 mode

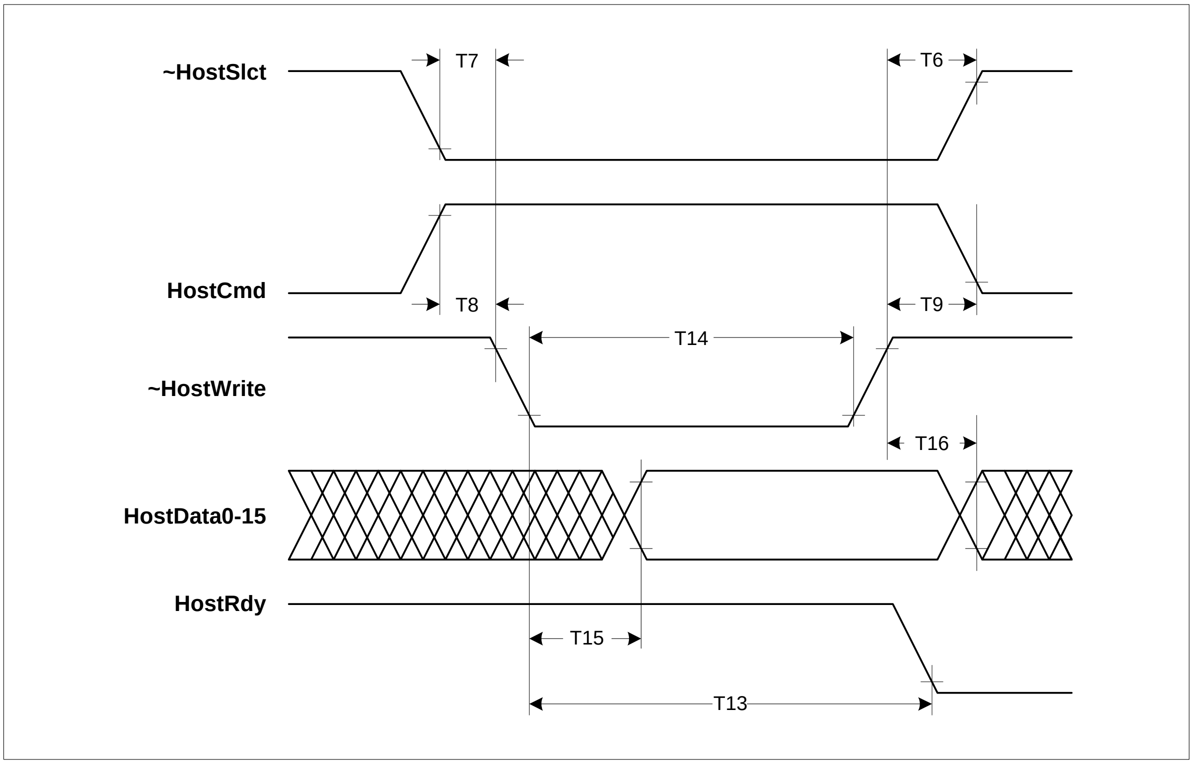
NOTE: If setup and hold times are met, ~HostSlct and HostComd may be de-asserted at this point.

Figure 4-7: Status read, 8/16 mode



4.5 Host Interface, 16/16 Mode

Figure 4-8: Instruction write, 16/16 mode



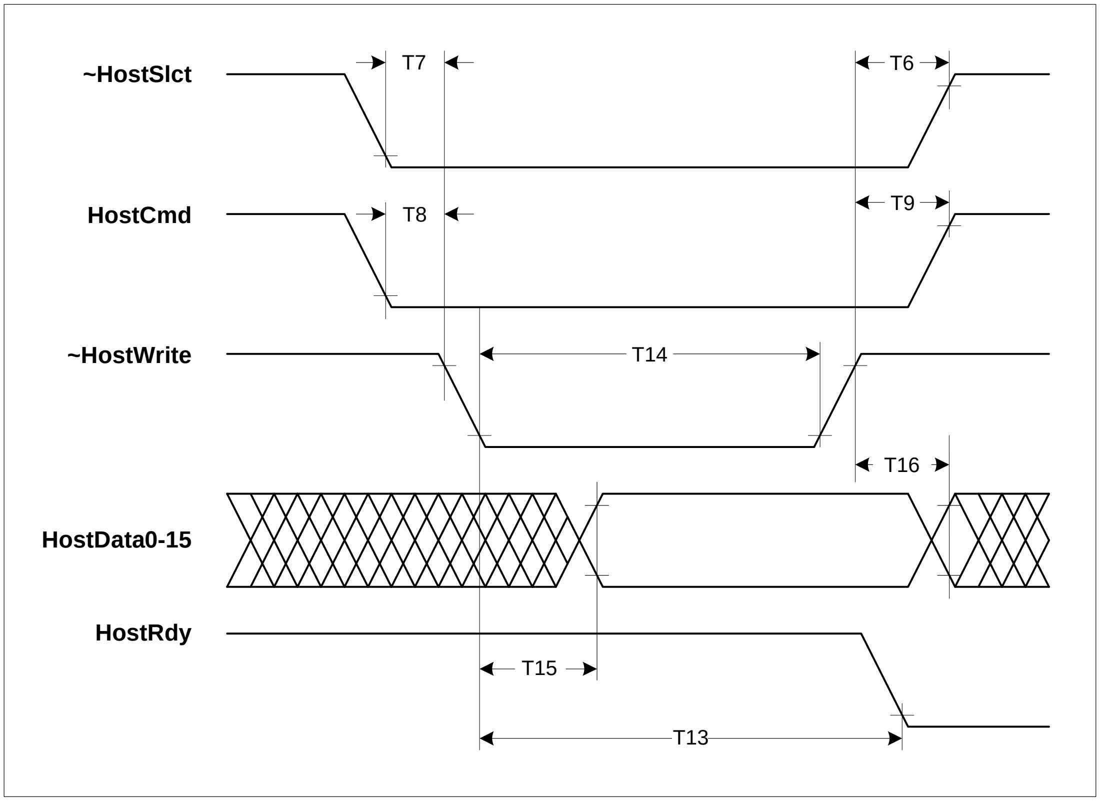


Figure 4-9: Data write, 16/16 mode

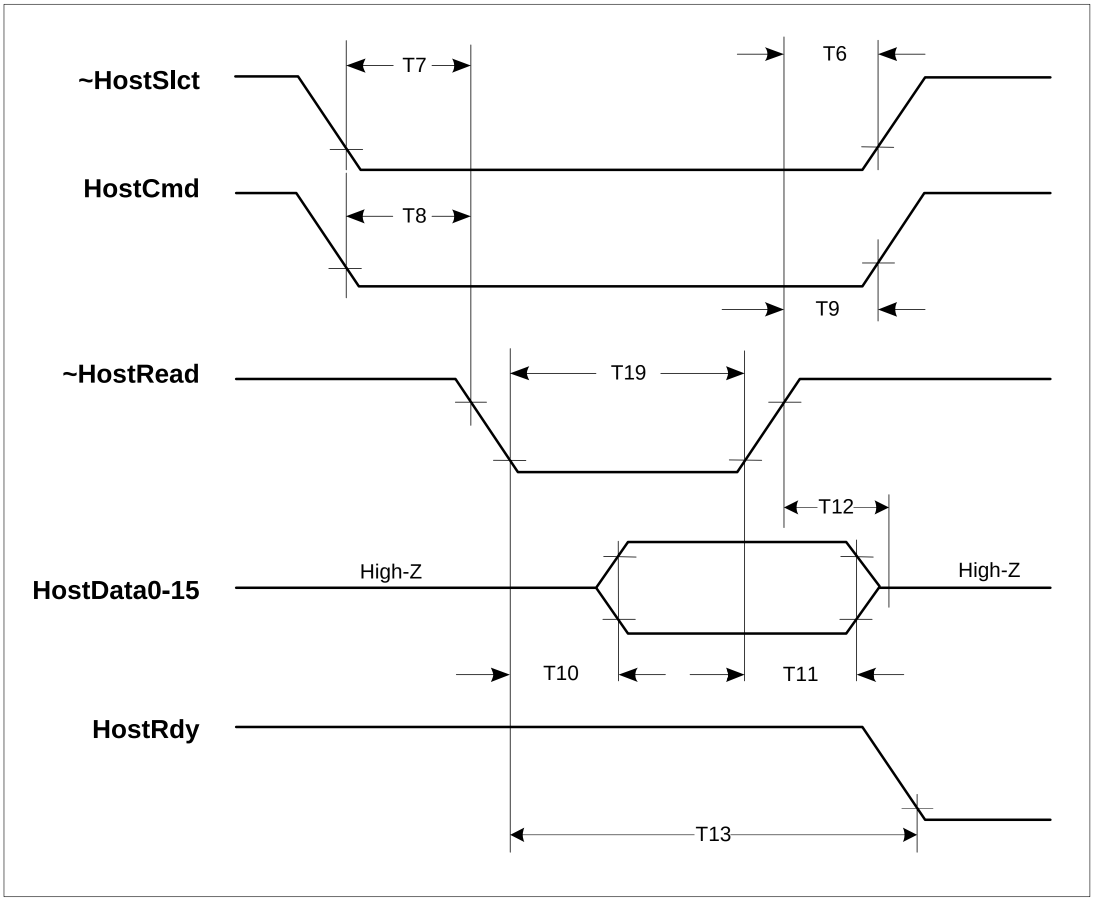
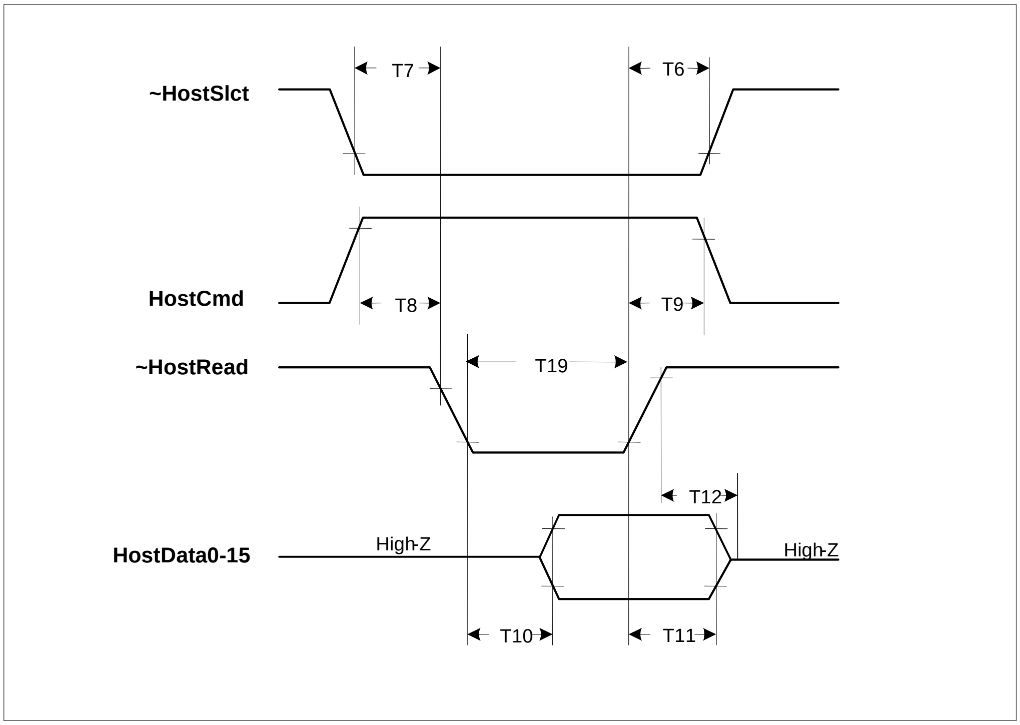


Figure 4-10: Data read, 16/16 mode

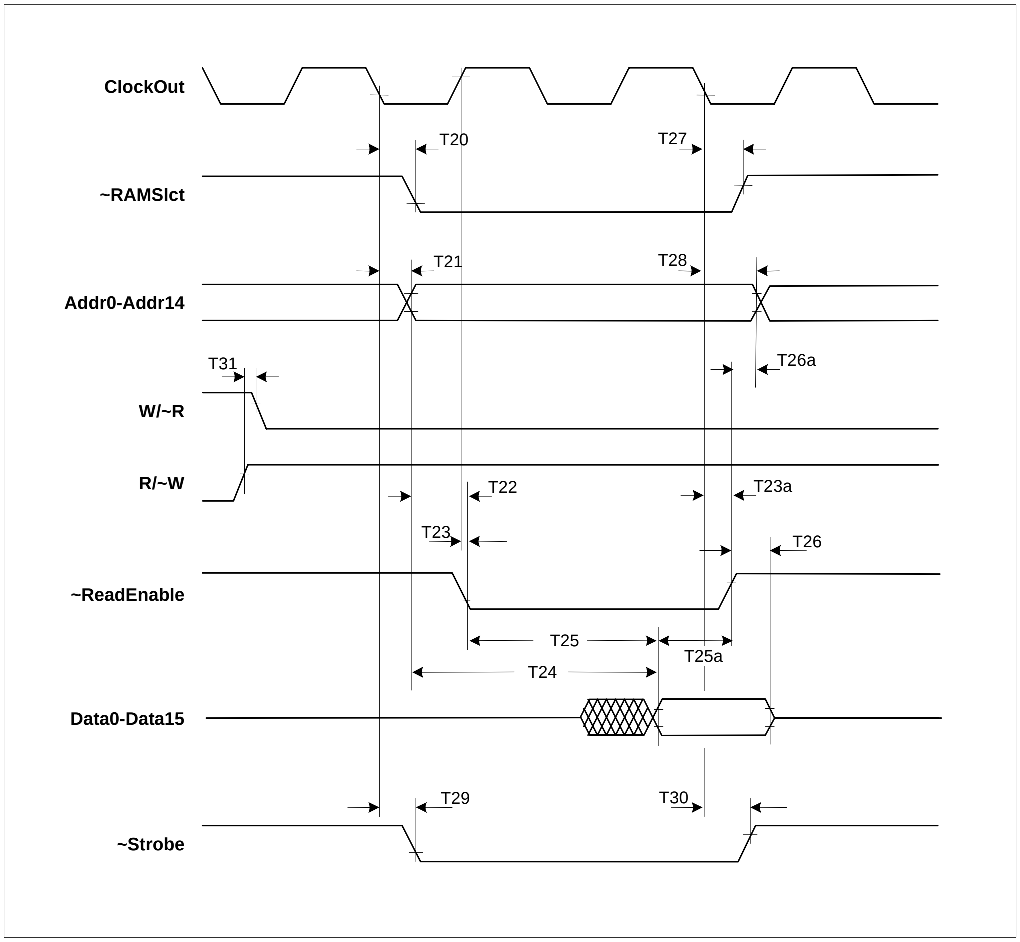
Figure 4-11: Status read, 16/16 mode



4.6 External Memory Timing

NOTE: PMD recommends using memory with an access time no greater than 15 nsec.

Figure 4-12: External memory read



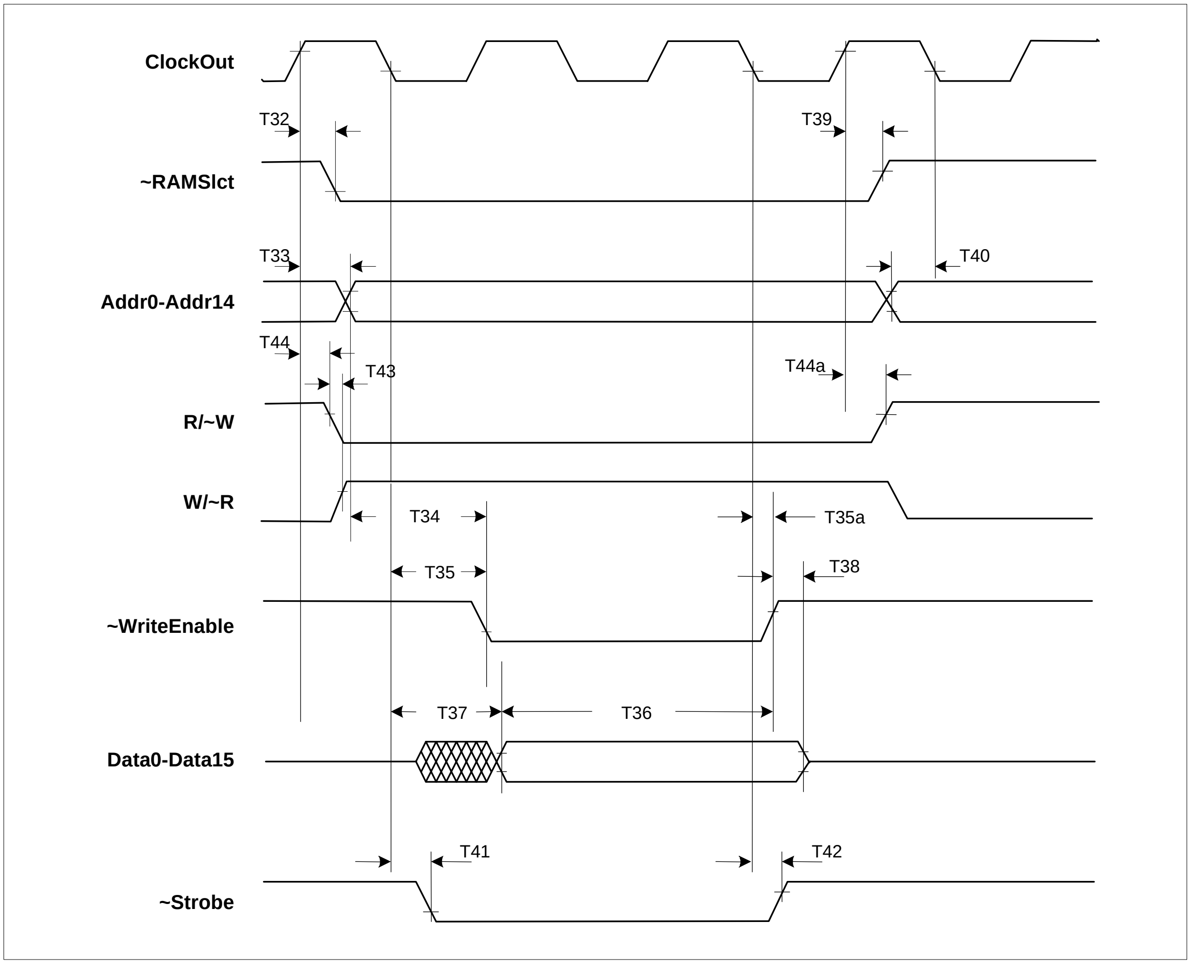
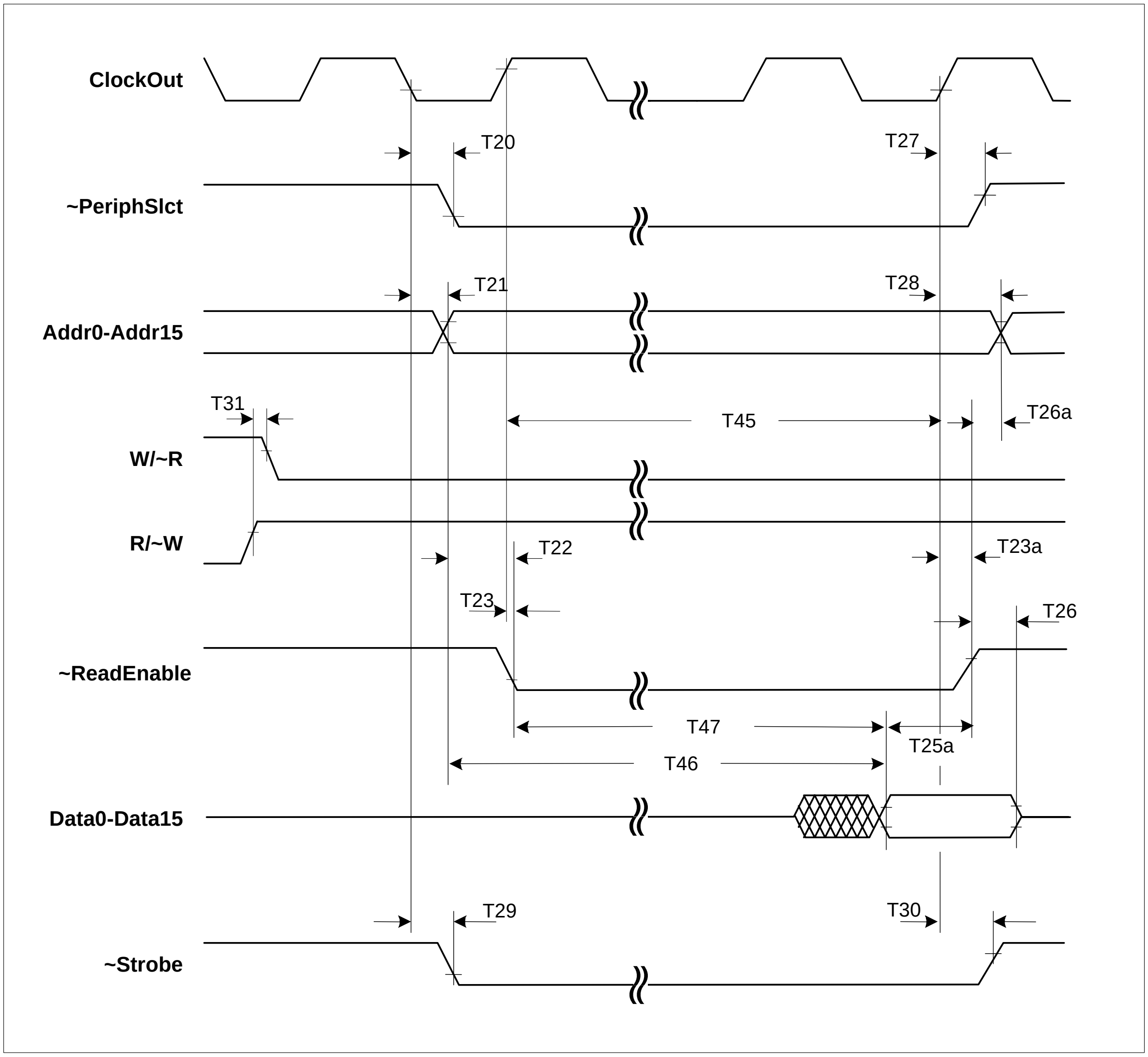


Figure 4-13: External memory write

4.7 Peripheral Device Timing

Figure 4-14: Peripheral device read



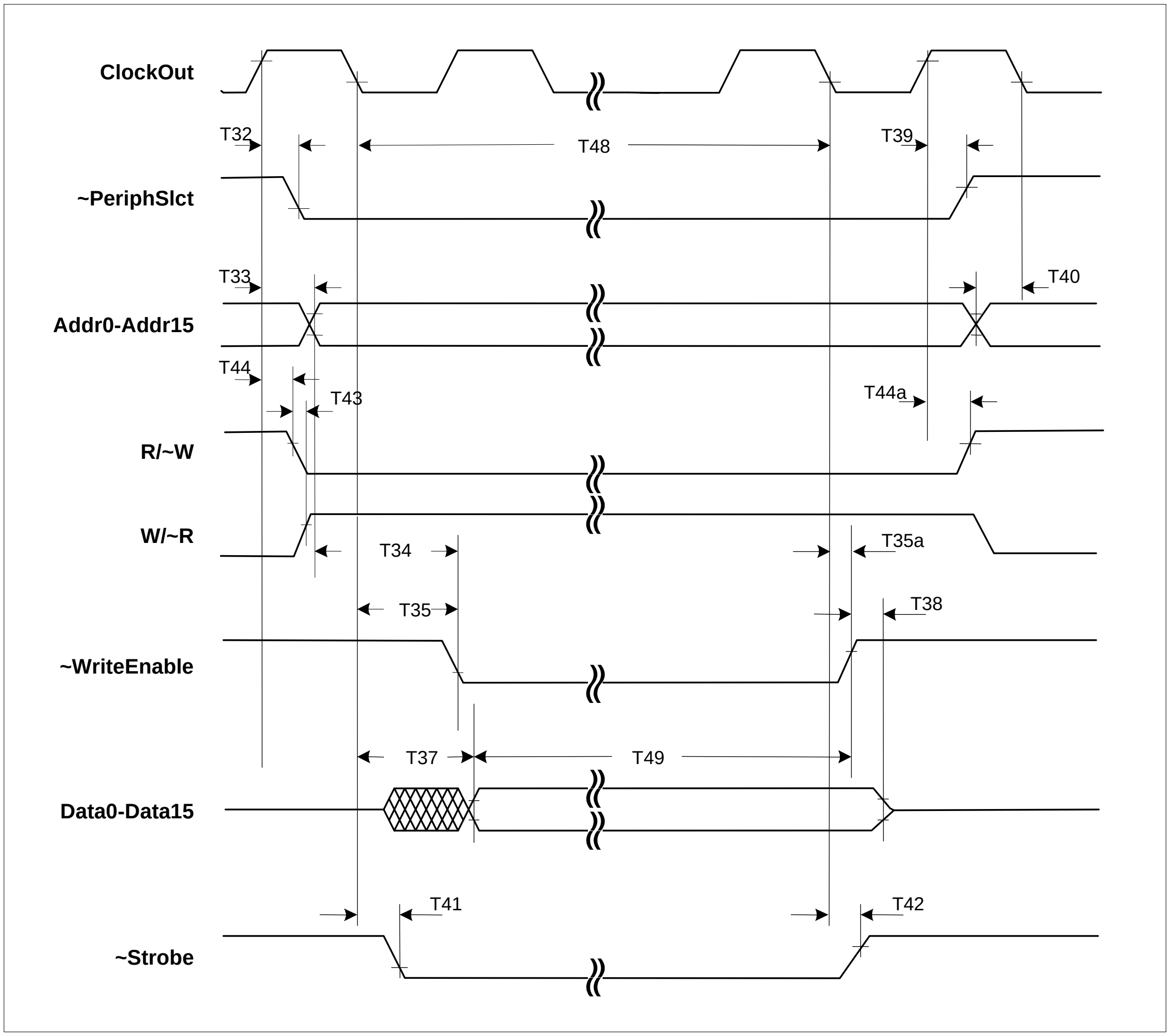


Figure 4-15: Peripheral device write

This page intentionally left blank.

5. Pinouts and Pin Descriptions

In This Chapter

- Pinouts for the MC55110
- Pinouts for the MC55420
- Pinouts for the MC55320
- Pinouts for the MC55220
- Pinouts for the MC55120

5.1 Pinouts for the MC55110

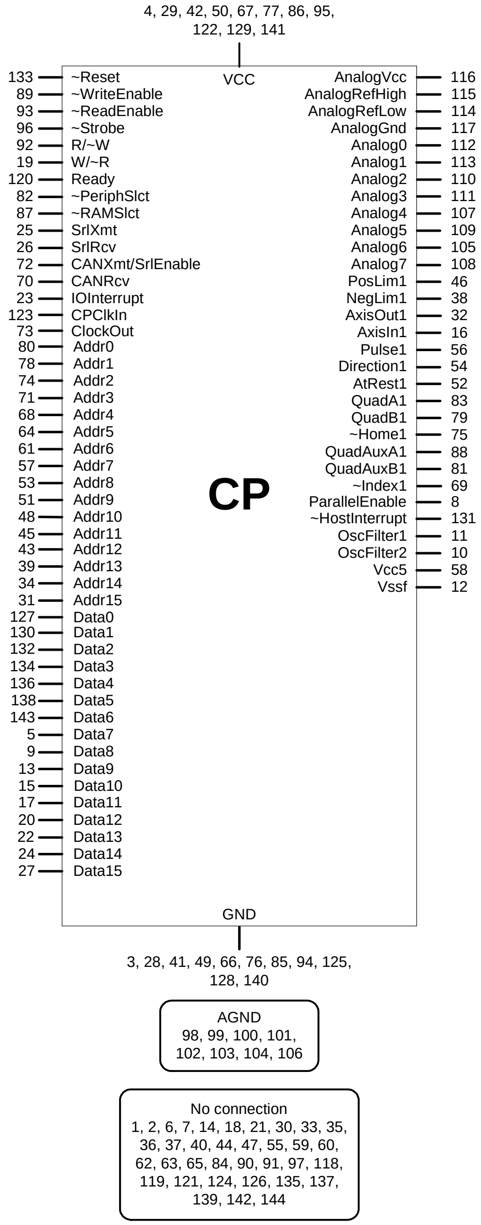


Figure 5-1: MC55110 pinouts

5.1.1 MC55110 CP Chip Pin Description

55110 CP			
Pin Name and Number		**Direction**	**Description**
~Reset	133	input/output	This is the master reset signal. This pin must be brought *low* to reset the chipset to its initial condition. **NOTE**: A software reset will momentarily drive this pin *low*.
~WriteEnable	89	output	This signal is the write-enable strobe. When *low*, this signal indicates that data is being written to the bus.
~ReadEnable	93	output	This signal is the read-enable strobe. When *low*, this signal indicates that data is being read from the bus.
~Strobe	96	output	This signal is *low* when the data and address are valid during CP communications.
R/~W	92	output	This signal is *high* when the CP chip is performing a read, and *low* when it is performing a write.
W/~R	19	output	This signal is the inverse of *R/~W*; it is *high* when *R/~W* is low, and vice versa. For some decode circuits and devices this is more convenient than *R/~W*.
Ready	120	input	*Ready* can be pulled low to add wait states for external accesses. *Ready* indicates that an external device is prepared for a bus transaction to be completed. If the device is not ready, it pulls the *Ready* pin *low*. The motion processor then waits one cycle and checks *Ready* again. This signal may remain unconnected if it is not used.
~PeriphSlct	82	output	This signal is *low* when peripheral devices on the data bus are being addressed.
~RAMSlct	87	output	This signal is *low* when external memory is being accessed.
SrlXmt	25	output	This pin outputs serial data from the asynchronous serial port.
SrlRcv	26	input	This pin inputs serial data to the asynchronous serial port.
CANXmt/ SrlEnable	72	output	When the CAN host interface is used, this pin transmits serial data to the CAN transceiver. When the multi-drop serial interface is used, this pin sets the serial port enable line and the *CANXmt* function is not available. *SrlEnable* is *high* during transmission for the multi-drop protocol and always *high* during point–point mode.
CANRcv	70	input	This pin receives serial data from the CAN transceiver.
IOInterrupt	23	input	This interrupt signal is used for IO to CP communication. This signal may remain unconnected if it is not used.
CPClkIn	123	input	This is the clock signal for the motion processor. It is driven at a nominal 20 MHz.
ClockOut	73	output	This signal is the reference output clock. Its frequency is twice the frequency of the input clock (which is normally 20 MHz), resulting in a nominal output frequency of 40 MHz. ClockOut will not be active when *~Reset* is active.
Addr0 Addr1 Addr2 Addr3 Addr4 Addr5 Addr6 Addr7 Addr8 Addr9 Addr10 Addr11 Addr12 Addr13 Addr14 Addr15	80 78 74 71 68 64 61 57 53 51 48 45 43 39 34 31	output	Multi-purpose address lines. These pins comprise the CP chip's external address bus, which is used to select devices for communication over the data bus. Other address pins may be used for DAC output, parallel word input, external memory, or user-defined I/O operations. See Section 6.3, "Peripheral Device Address Map," for a complete memory map.

55110 CP

Pin Name and Number		Direction	Description
Data0 Data1 Data2 Data3 Data4 Data5 Data6 Data7 Data8 Data9 Data10 Data11 Data12 Data13 Data14 Data15	127 130 132 134 136 138 143 5 9 13 15 17 20 22 24 27	bi-directional	Multi-purpose data lines. These pins comprise the CP chip's external data bus, which is used for all communications with peripheral devices such as external memory or DACs. They may also be used for parallel-word input and for user-defined I/O operations.
AnalogVcc	116	input	Analog input V_{cc}. This pin should be connected to the analog input supply voltage, which must be in the range of 3.0V to 3.6V. If the analog input circuitry is not used, this pin should be tied to V_{cc}.
AnalogRefHigh	115	input	Analog high voltage reference for A/D input. The allowed range is *AnalogRefLow* to *AnalogVcc*. If the analog input circuitry is not used, this pin should be tied to V_{cc}
AnalogRefLow	114	input	Analog low voltage reference for A/D input. The allowed range is *AnalogGND* to *AnalogRefHigh*. If the analog input circuitry is not used, this pin should be tied to *GND*.
AnalogGND	117	input	Analog input ground. This pin should be connected to the analog input power supply return. If the analog input circuitry is not used, this pin should be tied to *GND*.
Analog0 Analog1 Analog2 Analog3 Analog4 Analog5 Analog6 Analog7	112 113 110 111 107 109 105 108	input	These signals provide general-purpose analog voltage levels which are sampled by an internal A/D converter. The A/D resolution is 10 bits. The allowed signal input range is *AnalogRefLow* to *AnalogRefHigh*. Any unused pins should be tied to *AnalogGND*. If the analog input circuitry is not used, these pins should be tied to *GND*.
PosLim1	46	input	This signal provides input from the positive-side (forward) travel limit switch. On power-up or after reset, this signal defaults to active *low* interpretation, but the interpretation can be set to active *high* interpretation using the **SetSignalSense** instruction. If this pin is not used, it may remain unconnected.
NegLim1	38	input	This signal provides input from the negative-side (reverse) travel limit switch. On power-up or after reset, this signal defaults to active *low* interpretation, but the interpretation can be set to active *high* interpretation using the **SetSignalSense** instruction. If this pin is not used, it may remain unconnected.
AxisOut1	32	output	This pin can be programmed to track the state of any bit in the status registers. If this pin is not used, it may remain unconnected.
AxisIn1	16	input	This pin is a general-purpose input which can also be used as a breakpoint input. If this pin is not used, it may remain unconnected.

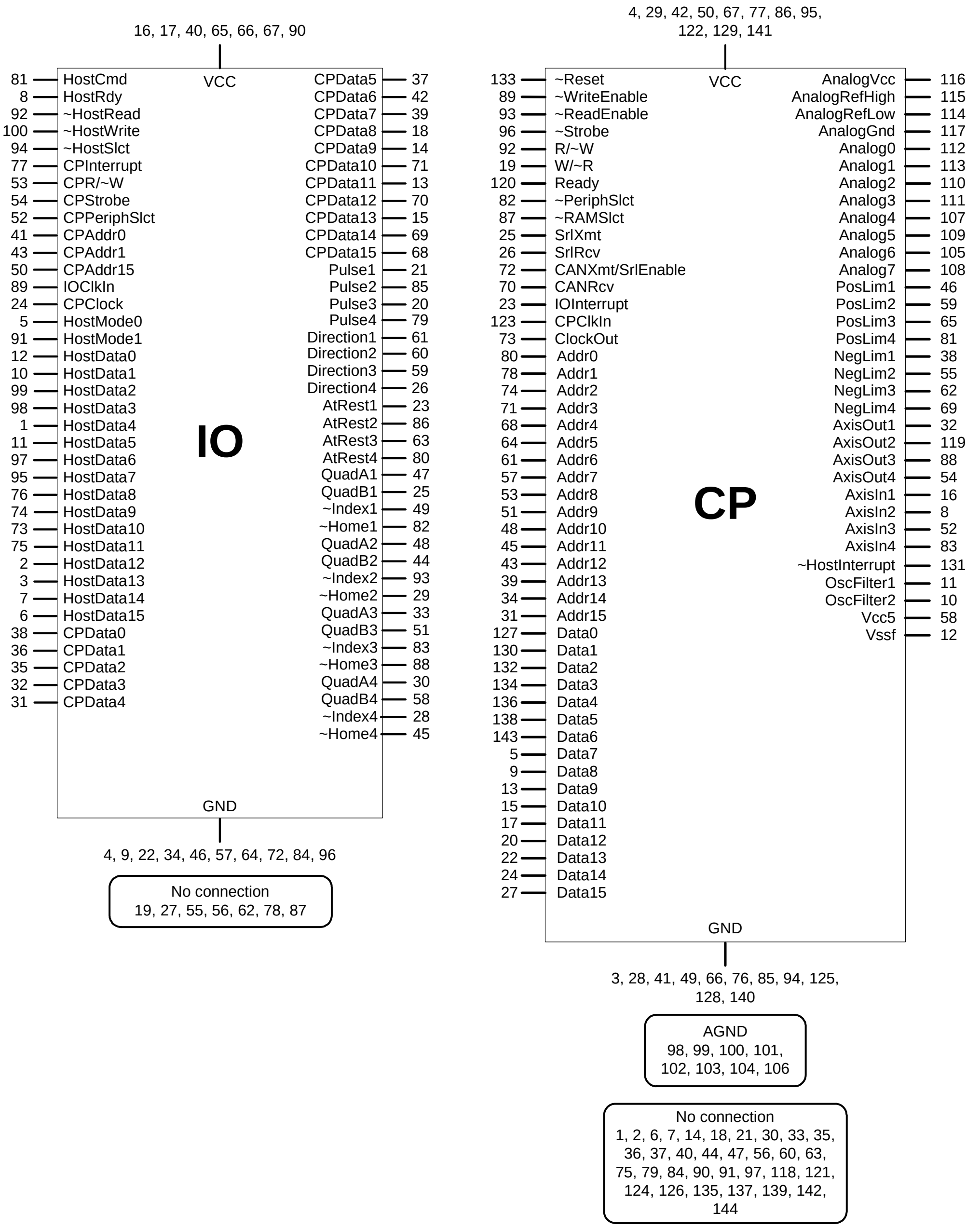
55110 CP

Pin Name and Number		Direction	Description
Pulse1	56	output	This pin provides the pulse (step) signal to the motor. A step occurs when the signal transitions from a *high* to a *low* state. This default behavior can be changed to a *low* to *high* state transition using the command **SetSignalSense.** If this pin is not used, it may remain unconnected.
Direction1	54	output	This pin indicates the direction of motion, and works in conjunction with the pulse signal. A *high* level on this signal indicates a positive direction move, and a *low* level indicates a negative direction move.
AtRest1	52	output	This signal indicates that the axis is at rest and the step motor can be switched to low power or standby mode. A *high* level on this signal indicates the axis is at rest while a *low* signal indicates the axis is in motion.
QuadA1 QuadB1	83 79	input	These pins should be connected to the A and B quadrature signals from the incremental encoder. When the axis is moving in the positive (forward) direction, signal A leads signal B by 90°. **NOTE**: Some encoders require a pull-up resistor to 3.3V on each signal to establish a proper high signal. Check your encoder's electrical specification. If these pins are not used, they may remain unconnected.
~Home1	75	input	This pin provides the home signal; a general-purpose input to the position capture mechanism. A valid home signal is recognized by the motion processor when *~Home* transitions from *high* to *low*. If this pin is not used, it may remain unconnected.
QuadAuxA1 QuadAuxB1 ~Index1	88 81 69	input	If index capture is required, the encoder A and B signals connected to *QuadA1* and *QuadB1* signals must also be connected to *QuadAuxA1* and *QuadAuxB1*. The index pin should be connected to the index signal from the incremental encoder. A valid index pulse is recognized by the motion processor when it meets the criteria shown in Figure 4-2. If these pins are not used, they may remain unconnected. **WARNING! There is no internal gating of the index signal with the encoder A and B inputs. This must be performed externally if desired.**
ParallelEnable	8	input	This signal enables/disables the parallel communication with the host. If this signal is tied *high*, the parallel interface is enabled. If this signal is tied *low,* the parallel interface is disabled. Contact PMD for more information on parallel communication. **WARNING! This signal should only be tied high if an external logic device which implements the parallel communication logic is included in the design.**
~HostInterrupt	131	output	When *low*, this signal causes an interrupt to be sent to the host processor.
OscFilter1 OscFilter2	11 10		These signals connect to the external oscillator filter circuitry. Section 6.6.5, "External Oscillator Filter," details the required filter circuitry.
V_{cc5}	58		This signal can be tied to a 5V supply if available. If 5V is not available this signal must be tied to GND. Being tied to GND will not adversely affect the device performance.
V_{ssf}	12		This signal must be tied to V_{cc}. It must also be tied to pin 28 via a bypass capacitor. A ceramic capacitor with a value between 0.1 µF and 0.01 µF should be used.
V_{cc}	4, 29, 42, 50, 67, 77, 86, 95, 122, 129, 141		CP digital supply voltage. All of these pins must be connected to the supply voltage. V_{cc} must be in the range of 3.0V to 3.6V.

55110 CP		
Pin Name and Number	**Direction**	**Description**
GND	3, 28, 41, 49, 66, 76, 85, 94, 125, 128, 140	CP digital supply ground. All of these pins must be connected to the digital power supply return.
AGND	98, 99, 100, 101, 102, 103, 104, 106	These signals must be tied to *AnalogGND*. If the analog input circuitry is not used, these pins must be tied to *GND*.
No connection	1, 2, 6, 7, 14, 18, 21, 30, 33, 35, 36, 37, 40, 44, 47, 55, 59, 60, 62, 63, 65, 84, 90, 91, 97, 118, 119, 121, 124, 126, 135, 137, 139, 142, 144	These signals must remain unconnected.

5.2 Pinouts for the MC55420

Figure 5-2: MC55420 pinouts



5.3 Pinouts for the MC55320

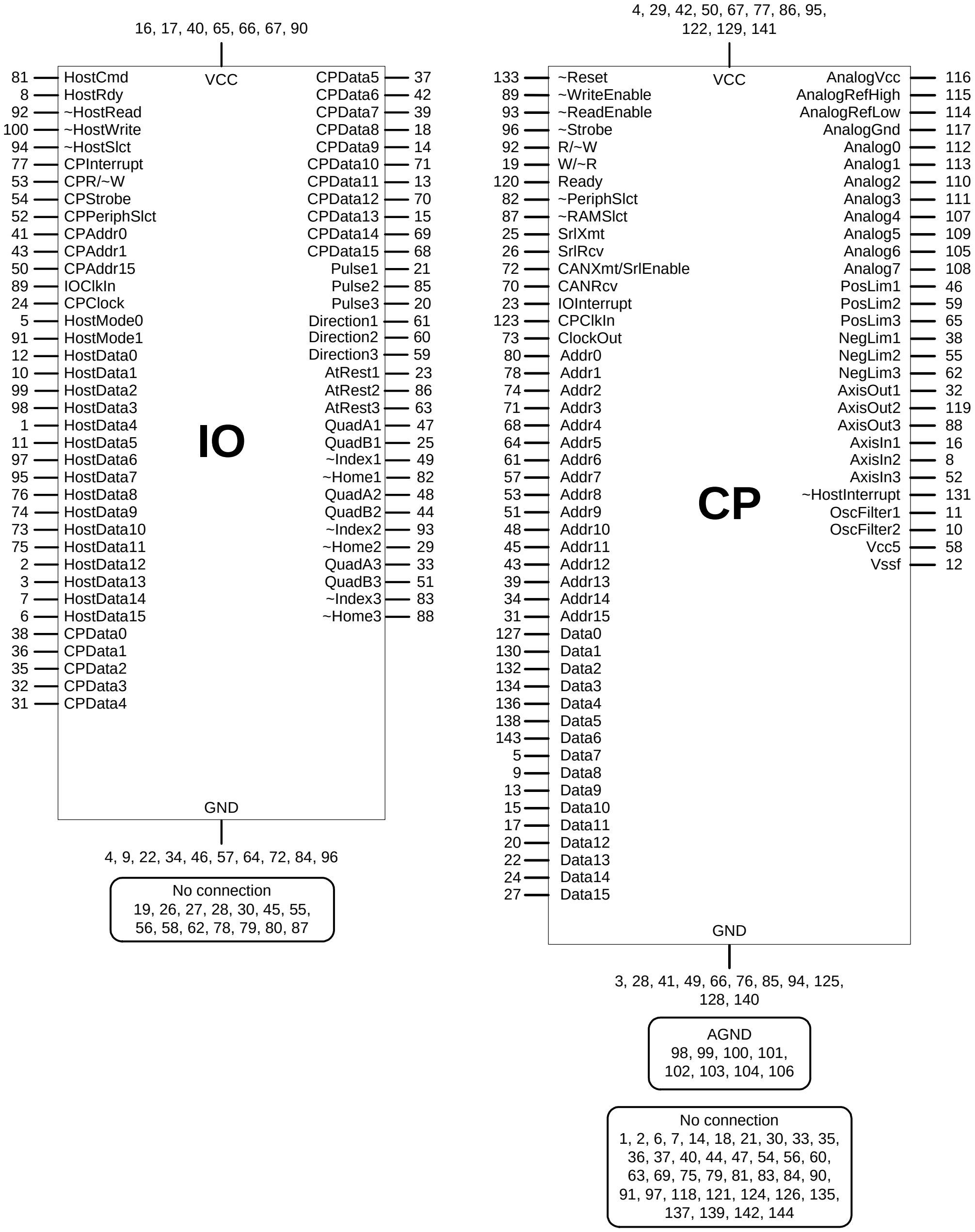
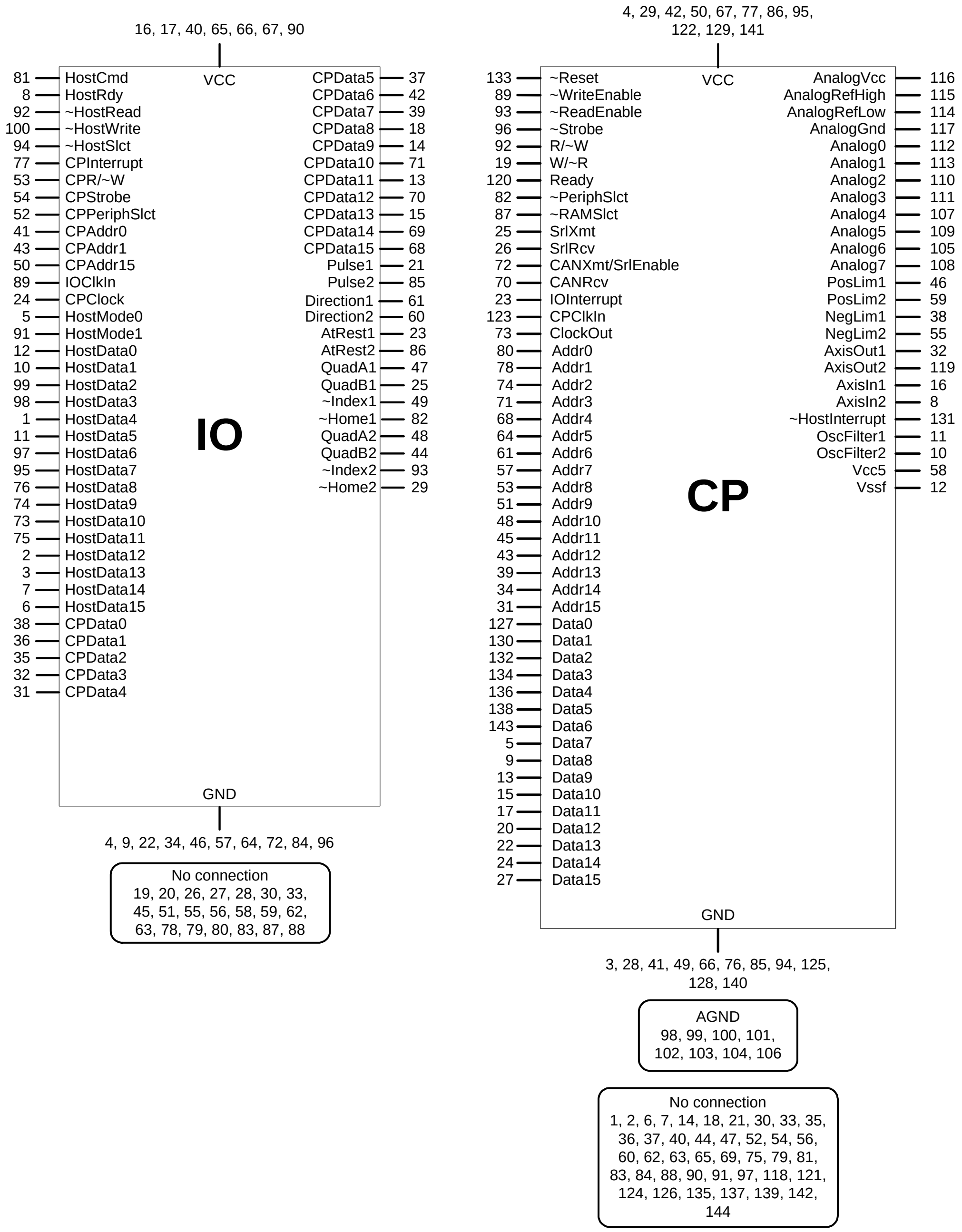


Figure 5-3: MC55320 pinouts

5.4 Pinouts for the MC55220

Figure 5-4: MC55220 pinouts



5.5 Pinouts for the MC55120

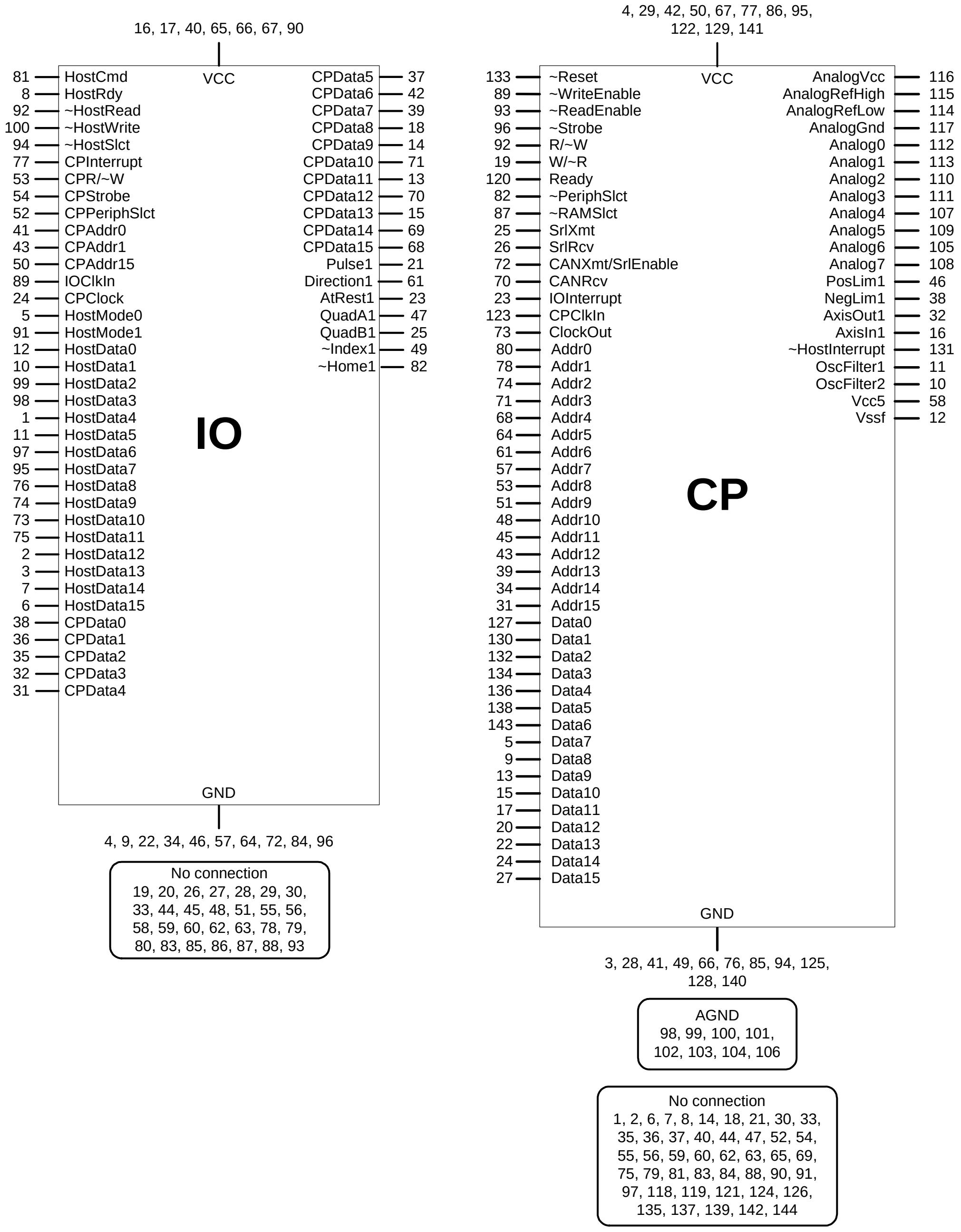


Figure 5-5: MC55120 pinouts

5.5.1 MC55x20 IO Chip Pin Description

MC55x20 IO

Pin Name and Number		**Direction**	**Description**
HostCmd	81	input	This signal is asserted *high* to write a host instruction to the motion processor, or to read the status of the *HostRdy* and *HostInterrupt* signals. It is asserted *low* to read or write a data word.
HostRdy	8	output	This signal is used to synchronize communication between the motion processor and the host. *HostRdy* (Host Ready) will go *low* indicating host port busy at the end of a read or write operation according to the interface mode in use, as follows: **Interface Mode** — ***HostRdy* goes *low*** 8/16 — after the second byte of the instruction word after the second byte of each data word is transferred 16/16 — after the 16-bit instruction word after each 16-bit data word *HostRdy* will go *high,* indicating that the host port is ready to transmit, when the last transmission has been processed. All host port communications must be made with *HostRdy high* (ready). A typical busy-to-ready cycle is 10 microseconds, but can be as long as 50 microseconds
~HostRead	92	input	When *~HostRead* is *low*, a data word is read from the motion processor.
~HostWrite	100	input	When *~HostWrite* is *low*, a data word is written to the motion processor.
~HostSlct	94	input	When *~HostSlct* is *low*, the host port is selected for reading or writing operations.
CPInterrupt	77	output	IO chip to CP chip interrupt. It should be connected to CP chip pin 23, *IOInterrupt*.
CPR/~W	53	input	This signal is *high* when the CP chip is reading data from the IO chip, and *low* when it is writing data. It should be connected to CP chip pin 92, *R/~W*.
CPStrobe	54	input	This signal goes *low* when the data and address become valid during motion processor communication with peripheral devices on the data bus, such as external memory or a DAC. It should be connected to CP chip pin 96, *~Strobe*.
CPPeriphSlct	52	input	This signal goes *low* when a peripheral device on the data bus is being addressed. It should be connected to CP chip pin 82, *~PeriphSlct.*
CPAddr0 CPAddr1	41 43	input	These signals can be *high* or *low*, and are used when the CP chip is communicating with the IO chip. They should be connected to CP chip pin 80 (*Addr0*), and pin 78 (*Addr1*).
CPAddr15	50	input	This signal is used by the CP chip when communicating with the IO chip. It should be connected to CP chip pin 31 (*Addr15*).
IOClkIn	89	input	This is the master clock signal for the chip set. It is driven at a nominal 40 MHz.
CPClock	24	output	This signal provides the clock pulse for the CP chip. Its frequency is half that of *IOClkIn* (pin 89), or 20 MHz nominal. It is connected directly to the CP chip *IOClock* signal (pin 123).
HostMode0 HostMode1	5 91	input	These two signals determine the host communications mode, as follows: *HostMode1* — *HostMode0* 0 — 0 — 16/16 parallel (16-bit bus, 16-bit instruction) 0 — 1 — *not used* 1 — 0 — 8/16 parallel (8-bit bus, 16-bit instruction) 1 — 1 — Parallel disabled

MC55x20 IO

Pin Name and Number		Direction	Description
HostData0	12	bi-directional, tri-state	These signals transmit data between the host and the motion processor through the parallel port. Transmission is mediated by the control signals *~HostSelect, ~HostWrite, ~HostRead and HostCmd*. In 16-bit mode, all 16 bits are used (*HostData0-15*). In 8-bit mode, only the low-order 8 bits of data are used (*HostData0-7*). The *HostMode0* and *HostMode1* signals select the communication mode in which this port operates.
HostData1	10		
HostData2	99		
HostData3	98		
HostData4	1		
HostData5	11		
HostData6	97		
HostData7	95		
HostData8	76		
HostData9	74		
HostData10	73		
HostData11	75		
HostData12	2		
HostData13	3		
HostData14	7		
HostData15	6		
CPData0	38	bi-directional	These signals transmit data between the IO chip and pins *Data0-15* of the CP chip.
CPData1	36		
CPData2	35		
CPData3	32		
CPData4	31		
CPData5	37		
CPData6	42		
CPData7	39		
CPData8	18		
CPData9	14		
CPData10	71		
CPData11	13		
CPData12	70		
CPData13	15		
CPData14	69		
CPData15	68		
Pulse1	21	output	These pins provide the pulse (step) signal to the motor. This signal is always a square wave, regardless of the pulse rate. A step occurs when the signal transitions from a *high* state to a *low* state. This default behavior can be changed to a *low* to *high* state transition using the command **SetSignalSense**. The number of available axes determines which of these signals are valid. Invalid axis pins may remain unconnected.
Pulse2	85		
Pulse3	20		
Pulse4	79		
Direction1	61	output	These pins indicate the direction of motion and work in conjunction with the pulse signal. A *high* level on this signal indicates a positive direction move and a *low* level indicates a negative direction move. The number of available axes determines which of these signals are valid. Invalid axis pins may remain unconnected.
Direction2	60		
Direction3	59		
Direction4	26		
AtRest1	23	output	The *AtRest* signal indicates the axis is at rest and the step motor can be switched to low power or standby. A *high* level on this signal indicates the axis is at rest. A *low* signal indicates the axis is in motion. The number of available axes determines which of these signals are valid. Invalid axis pins may remain unconnected.
AtRest2	86		
AtRest3	63		
AtRest4	80		
QuadA1	47	input	These pins provide the A and B quadrature signals for the incremental encoder for each axis. When the axis is moving in the positive (forward) direction, signal A leads signal B by 90°. **NOTE**: Some encoders require a pull-up resistor to 3.3V on each signal to establish a proper high signal. Check your encoder's electrical specification. The number of available axes determines which of these signals are valid. **WARNING! If a valid axis pin is not used, its signal should be tied high.** Invalid axis pins may remain unconnected, or may be connected to ground.
QuadB1	25		
QuadA2	48		
QuadB2	44		
QuadA3	33		
QuadB3	51		
QuadA4	30		
QuadB4	58		

MC55x20 IO

Pin Name and Number		Direction	Description
~Index1 ~Index2 ~Index3 ~Index4	49 93 83 28	input	These pins provide the Index quadrature signals for the incremental encoders. A valid index pulse is recognized by the chipset when *~Index*, *A*, and *B* are all *low*. The number of available axes determines which of these signals are valid. **WARNING! If a valid axis pin is not used, its signal should be tied high.** Invalid axis pins may remain unconnected, or may be connected to ground.
~Home1 ~Home2 ~Home3 ~Home4	82 29 88 45	input	These pins provide the home signals, which are the general-purpose inputs to the position-capture mechanism. A valid home signal is recognized by the chipset when *~Home*n goes *low*. These signals are similar to *~Index*, but are not gated by the A and B encoder channels. The number of available axes determines which of these signals are valid. **WARNING! If a valid axis pin is not used, its signal should be tied high.** Invalid axis pins may remain unconnected, or may be connected to ground.
Vcc	16, 17, 40, 65, 66, 67, 90		All of these pins must be connected to the IO chip digital supply voltage, which should be in the range of 3.0V to 3.6V.
GND	4, 9, 22, 34, 46, 57, 64, 72, 84, 96		IO chip ground. All of these pins must be connected to the digital power supply return.
Not connected	19, 27, 55, 56, 62, 78, 87		These pins must remain unconnected (floating).

5.5.2 MC55x20 CP Chip Pin Description

MC55x20 CP			
Pin Name and Number		**Direction**	**Description**
~Reset	133	input/output	This is the master reset signal. This pin must be brought *low* to reset the chipset to its initial conditions. **NOTE:** A software reset will momentarily drive this pin *low*.
~WriteEnable	89	output	This signal is the write-enable strobe. When *low*, this signal indicates that data is being written to the bus.
~ReadEnable	93	output	This signal is the read-enable strobe. When *low*, this signal indicates that data is being read from the bus.
~Strobe	96	output	This signal is *low* when the data and address are valid during CP communications. It should be connected to IO chip pin 54, *CPStrobe*.
R/~W	92	output	This signal is *high* when the CP chip is performing a read, and *low* when it is performing a write. It should be connected to IO chip pin 53, *CPR/~W*.
W/~R	19	output	This signal is the inverse of *R/~W*; it is *high* when *R/~W* is *low*, and vice-versa. For some decode circuits and devices, this is more convenient than *R/~W*.
Ready	120	input	*Ready* can be pulled *low* to add wait states for external accesses. *Ready* indicates that an external device is prepared for a bus transaction to be completed. If the device is not ready, it pulls the *Ready* pin *low*. The motion processor then waits one cycle and checks *Ready* again. This signal may remain unconnected if it is not used.
~PeriphSlct	82	output	This signal is *low* when peripheral devices on the data bus are being addressed. It should be connected to IO chip pin 52, *CPPeriphSlct*.
~RAMSlct	87	output	This signal is *low* when external memory is being accessed.
SrlXmt	25	output	This pin outputs serial data from the asynchronous serial port.
SrlRcv	26	input	This pin inputs serial data to the asynchronous serial port.
CANXmt/ SrlEnable	72	output	When the CAN host interface is used, this pin transmits serial data to the CAN transceiver. When the multi-drop serial interface is used, this pin sets the serial port enable line, and the *CANXmt* function is not available. *SrlEnable* is *high* during transmission for the multi-drop protocol, and always *high* during point–point mode.
CANRcv	70	input	This pin receives serial data from the CAN transceiver.
IOInterrupt	23	input	This interrupt signal is used for IO to CP communication. It should be connected to IO chip pin 77, *CPInterrupt*.
CPClkIn	123	input	This is the CP chip clock signal. It should be connected to IO chip pin 24, *CPClock*.
ClockOut	73	output	This signal is the reference output clock. Its frequency is the same as the *CPClkIn* signal to the IO chip, nominally 40 MHz. ClockOut will not be active when ~Reset is active.
Addr0 Addr1 Addr2 Addr3 Addr4 Addr5 Addr6 Addr7 Addr8 Addr9 Addr10 Addr11 Addr12 Addr13 Addr14 Addr15	80 78 74 71 68 64 61 57 53 51 48 45 43 39 34 31	output	Multi-purpose address lines. These pins comprise the CP chip's external address bus, and are used to select devices for communication over the data bus. *Addr0*, *Addr1*, and *Addr15* are connected to the corresponding *CPAddr* pins on the IO chip, and are used to communicate between the CP and IO chips. Other address pins may be used for DAC output, parallel word input, external memory, or user-defined I/O operations. See Section 6.3, "Peripheral Device Address Map," for a complete memory map.

MC55x20 CP

Pin Name and Number		Direction	Description
Data0 Data1 Data2 Data3 Data4 Data5 Data6 Data7 Data8 Data9 Data10 Data11 Data12 Data13 Data14 Data15	127 130 132 134 136 138 143 5 9 13 15 17 20 22 24 27	bi-directional	Multi-purpose data lines. These pins comprise the CP chip's external data bus, which is used for all communications with the IO chip and peripheral devices such as external memory or DACs. They may also be used for parallel-word input and for user-defined I/O operations.
AnalogVcc	116	input	Analog input Vcc. This pin should be connected to the analog input supply voltage, which must be in the range of 3.0V to 3.6V. If the analog input circuitry is not used, this pin should be tied to V_{cc}.
AnalogRefHigh	115	input	Analog high voltage reference for A/D input. The allowed range is 2V to *AnalogVcc*. Furthermore, the difference between *Vcc* and *AnalogVcc* should not be larger than 0.3V. If the analog input circuitry is not used, this pin should be tied to V_{cc}.
AnalogRefLow	114	input	Analog low voltage reference for A/D input. The allowed range is *AnalogGND* to *AnalogRefHigh*. If the analog input circuitry is not used, this pin should be tied to *GND*.
AnalogGND	117	input	Analog input ground. This pin should be connected to the analog input power supply return. If the analog input circuitry is not used, this pin should be tied to *GND*.
Analog0 Analog1 Analog2 Analog3 Analog4 Analog5 Analog6 Analog7	112 113 110 111 107 109 105 108	input	These signals provide general-purpose analog voltage levels which are sampled by an internal A/D converter. The A/D resolution is 10 bits. The allowed signal input range is *AnalogRefLow* to *AnalogRefHigh*. Any unused pins should be tied to *AnalogGND*. If the analog input circuitry is not used, these pins should be tied to *GND*.
PosLim1 PosLim2 PosLim3 PosLim4	46 59 65 81	input	These signals provide inputs from the positive-side (forward) travel limit switches. On power-up or after reset, these signals default to active *low* interpretation, but the interpretation can be set explicitly using the **SetSignalSense** instruction. The number of available axes determines which of these signals are valid. Invalid or unused pins may remain unconnected.
NegLim1 NegLim2 NegLim3 NegLim4	38 55 62 69	input	These signals provide inputs from the negative-side (reverse) travel limit switches. On power-up or after reset, these signals default to active *low* interpretation, but the interpretation can be set explicitly using the **SetSignalSense** instruction. The number of available axes determines which of these signals are valid. Invalid or unused pins may remain unconnected.
AxisOut1 AxisOut2 AxisOut3 AxisOut4	32 119 88 54	output	Each of these pins can be conditioned to track the state of any bit in the status registers associated with its axis. The number of available axes determines which of these signals are valid. Invalid or unused pins may remain unconnected.
AxisIn1 AxisIn2 AxisIn3 AxisIn4	16 8 52 83	input	These are general-purpose inputs which can also be used as a breakpoint input. The number of available axes determines which of these signals are valid. Invalid or unused pins may remain unconnected.
~HostInterrupt	131	output	When *low*, this signal causes an interrupt to be sent to the host processor.

MC55x20 CP

Pin Name and Number		Direction	Description
OscFilter1 OscFilter2	11 10		These signals connect to the external oscillator filter circuitry. Section 6.6.5, “External Oscillator Filter,” details the required filter circuitry.
V_{cc5}	58		This signal can be tied to a 5V supply if available. If 5V is not available this signal must be tied to GND. Being tied to GND will not adversely affect the device performance.
V_{ssf}	12		This signal must be tied to V_{cc}. It must also be tied to pin 28 via a bypass capacitor. A ceramic capacitor with a value between 0.1 μF and 0.01 μF should be used.
V_{cc}	4, 29, 42, 50, 67, 77, 86, 95, 122, 129, 141		CP digital supply voltage. All of these pins must be connected to the supply voltage. V_{cc} must be in the range of 3.0V to 3.6V.
GND	3, 28, 41, 49, 66, 76, 85, 94, 125, 128, 140		CP digital supply ground. All of these pins must be connected to the digital power supply return.
AGND	98, 99, 100, 101, 102, 103, 104, 106		These signals must be tied to *AnalogGND*. If the analog input circuitry is not used, these pins must be tied to *GND*.
No connection	1, 2, 6, 7, 14, 18, 21, 30, 33, 35, 36, 37, 40, 44, 47, 56, 60, 63, 75, 79, 84, 90, 91, 97, 118, 121, 124, 126, 135, 137, 139, 142, 144		These signals must remain unconnected.

This page intentionally left blank.

6. Application Notes — MC55110 and MC55x20

In This Chapter

- General Design Notes
- Design Tips
- Peripheral Device Address Mapping
- Device Initialization
- Power Supplies
- Clock Generator, Grounding and Decoupling, and Device Reset
- Serial Communication Interface (SCI)
- CAN Communication Interface
- External Memory
- Asynchronous SRAM
- Dual Port Synchronous SRAM (DPRAM)
- Using the On-chip ADC
- User I/O Space
- Parallel Communication Interface
- Overcurrent and Emergency Braking Circuits for Motor Drivers
- DC Brush Motor Control Using SPI Interfaced DACs
- Brushless DC Motor Control Using High-Precision Parallel DACs
- Using PWM for DC Brush, Brushless DC and Microstepping Motors
- Using the Allegro A3977 to Drive Microstepping Motors

6.1 General Design Notes

Logic functions presented in the example schematics are implemented by standard logic gates. In cases where specific parameters are of significance (propagation delay, voltage levels, etc…) a recommended part number is given.

One important point of note is that the single and dual chip configurations of Magellan share several signals with the same name that reside on physically different chips. For example, on the MC55x20, the pulse & direction signals are located on the IO chip, while on the MC55110, the pulse & direction signals are located on the CP chip. As such, care should be taken to ensure that during the initial schematic layout that the correct CP chip is selected.

> The schematic designs presented in this chapter are accurate to the best of PMD's knowledge. They are intended for reference only and have not all been tested in hardware implementations.

6.1.1 Interfacing to Other Logic Families

When integrating different logic families, consideration should be given to timing, logic level compatibility, and output drive capabilities. The Magellan CP and IO chips are 3.3V CMOS input/output compatible and cannot be directly interfaced to 5V CMOS components. In order to drive a 5V CMOS device, level shifters from the 5V CMOS AHCT (or the slower HCT) families can be used. When using a 5V CMOS component to drive the CP, a voltage divider may be used or a member from the CMOS 3.3V LVT family may serve as a level shifter.

6.2 Design Tips

The following are recommendations/requirements for the design of circuits which utilize a PMD Motion Processor.

6.2.1 Serial Interface

The serial interface is a convenient interface which can be used before host software has been written to communicate through the parallel interface. It is recommended that even if the serial interface is not utilized as a standard communication interface, that the serial receive and transmit signals are brought to test points so that they may be connected during initial board configuration/debugging. This is especially important during the prototype phase. The serial receive line should include a pull-up resistor to avoid spurious interrupts when it is not connected to a transceiver.

If the serial configuration decode logic is not implemented, and the serial interface is used for debugging as previously mentioned, the CP data bus should be tied high. This places the serial interface in a default configuration of 57,600, n, 8, 1 after power on or reset.

6.2.2 Using a Non-standard System Clock Frequency

It is often desirable to share a common clock among several components in a design. In the case of the PMD Motion Processors, it is possible to use a clock below the standard value of 40 MHz. In this case, all system frequencies will be reduced as a fraction of the input clock versus the standard 40 MHz clock. The following list details the affected system parameters.

- Serial baud rate
- Cycle time

For example, if an input clock of 34 MHz is used with a serial baud rate of 9600, the following timing changes will result.

- Serial baud rate decreases to 9600 bps *34/40 = 8160 bps
- Total cycle time increases by a factor of 40/34

6.3 Peripheral Device Address Map

Device addresses on the CP chip's external bus are memory-mapped to the following locations.

Address	Device	Description
0100h	Motor type configuration	Contains the configuration data for the per axis motor type selection
0200h	Serial port configuration	Contains the configuration data (transmission rate, parity, stop bits, etc.) for the asynchronous serial port
0400h	CAN port configuration	Contains the configuration data (baud rate and node ID) for the CAN controller
0800h	Parallel-word encoder	Base address for parallel-word feedback devices
1000h	User-defined	Base address for user-defined I/O devices
2000h	RAM page pointer	Page pointer to external memory
4000h	Motor-output DACs	Base address for motor-output D/A converters
8000h	*reserved*	

6.4 Device Initialization

Following a hardware or software reset, the motion processor reads from three external configuration registers to determine the desired settings for the motor type, serial communication, and CAN communication. These reads take place sequentially using the peripheral address read. The timing for this read is shown in Figure 4-14.

6.4.1 Serial Port Configuration

When the serial configuration is read, the 16-bit word is interpreted according to the following table.

Bit Number	Name	Instance	Encodin g
0-3	transmission rate	1200 baud	0
		2400	1
		9600	2
		19200	3
		57600	4
		115200	5
		230400	6
		460800	7
4-5	parity	none	0
		odd	1
		even	2
6	stop bits	1	0
		2	1
7-8	protocol	Point-to-point	0
		Multi-drop using idle-line detection	1
		reserved	2
		reserved	3
11-15	multi-drop address	Address 0	0
		Address 1	1
		…	…
		Address 31	31

6.4.2 CAN Port Configuration

When the CAN configuration is read, the 16-bit word is interpreted according to the following table.

Bit Number	Name	Instance	Encoding
0-6	nodeID	Address 0	0
		Address 1	1
		…	…
		Address 127	127
7-12	*reserved*	*reserved*	*reserved*
13-15	transmission rate	1,000,000 baud	0
		800,000	1
		500,000	2
		250,000	3
		125,000	4
		50,000	5
		20,000	6
		10,000	7

As an alternative to decoding each configuration address, a special condition occurs when the device powers up and the external bus is pulled high. In this case, the device will read the contents of each configuration register as containing the value 0xffff. When this occurs, the device will configure the serial port as 57,600, n, 8, 1; and the CAN port as 20,000 bps with a NodeID of zero.

If the serial or CAN port is not required, the circuitry for decoding the relevant addresses can be omitted, and the *CANRcv* and *SrlRcv* signals can remain disconnected to prevent the chip from responding to either of these communication inputs.

6.5 Power Supplies

In the schematic shown in Figure 6-1, the design is powered by an external 5V DC power source. The MC55000 device requires a 3.3V supply and an optional 5V input. The 5V input to the motion processor can be omitted if 5V is not required elsewhere in the design. The 3.3V digital supply, VCC, is generated by an LT1086-3.3, a 1.5Amp fixed 3.3V low-dropout voltage regulator. Components with a larger power capacity are also available, such as the LT1085-3.3.

If the CP's analog-to-digital converter (ADC) is used, it should be supplied with a filtered 3.3V supply. The +3.3Vs supply is a filtered version of the VCC supply, which is used to supply the ADC and its related conditioning circuitry. The extra filtering is used to provide additional decoupling of the analog elements from the digital elements in the circuitry.

The following is the list of supplies which are referenced in the example schematics:

- +5Vs: a filtered version of 5VCC. This is used for analog components requiring +5V supplies. The extra filtering is used to reduce the voltage ripple, and to generate additional decoupling of the analog elements from the digital elements in the circuitry.
- ±15Vs: ±15V supplies, used for analog components dealing with large input or output voltage swings; usually when interfacing to motors or sensors. The Calex 5D15.033 is a 1W ±15V DC/DC converter which delivers ±33 mA. Depending on the current load of the final design, a larger power capacity DC/DC converter may be required. An LC filter is used to reduce the voltage ripple.

Notes:

- The schematic in Figure 6-1 should be used for reference only. The actual supplies used should be designed according to the stability and precision requirements of the application. The power supplies presented here are only designed to meet the requirements of the example schematics.
- Power supplies for the motor drivers are not shown. Care should be taken when designing these power supplies, as they should be capable of sinking high switching currents.

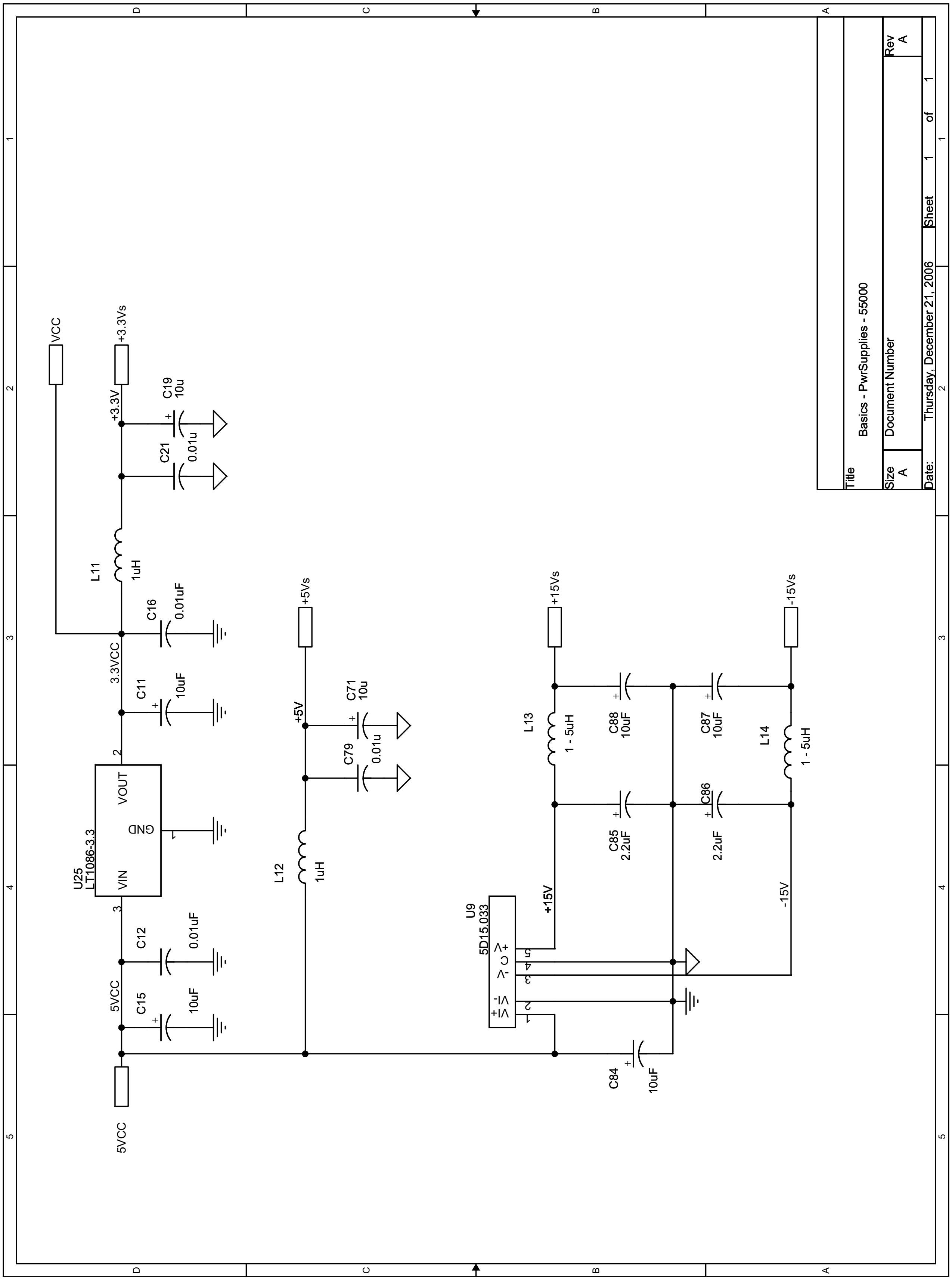


Figure 6-1: Basics, power supplies, 55000

This page intentionally left blank.

6.6 Clock Generator, Grounding and Decoupling, and Device Reset

6.6.1 Clock Generator — MC55110

The nominal clock frequency of the MC55110 CP is 20 MHz. A separate 20 MHz clock may be generated for the device and peripherals or a pre-existing derivative of a clock generated on the board may be used. If an existing clock is to be used, ensure that the input voltages and timing requirements of the MC55110 are met and that the correct frequency is generated.

For any frequency in the range, the bypass capacitor (labeled C3 in figures 6-2 and 6-3) should be between 0.1 and 0.01 µF, ceramic, and it should have private and as short as possible traces to the pins. This method will reduce noise and jitter, and increase isolation.

6.6.2 Clock Generator — MC55x20

The nominal clock frequency of the MC55x20 IO is 40 MHz. The IO chip generates a nominal 20 MHz clock signal for the CP chip by dividing the input frequency by two. When applying a lower clock frequency to the IO chip, the CP external oscillator filter circuit must adhere to the values listed in Section 6.6.5, "External Oscillator Filter."

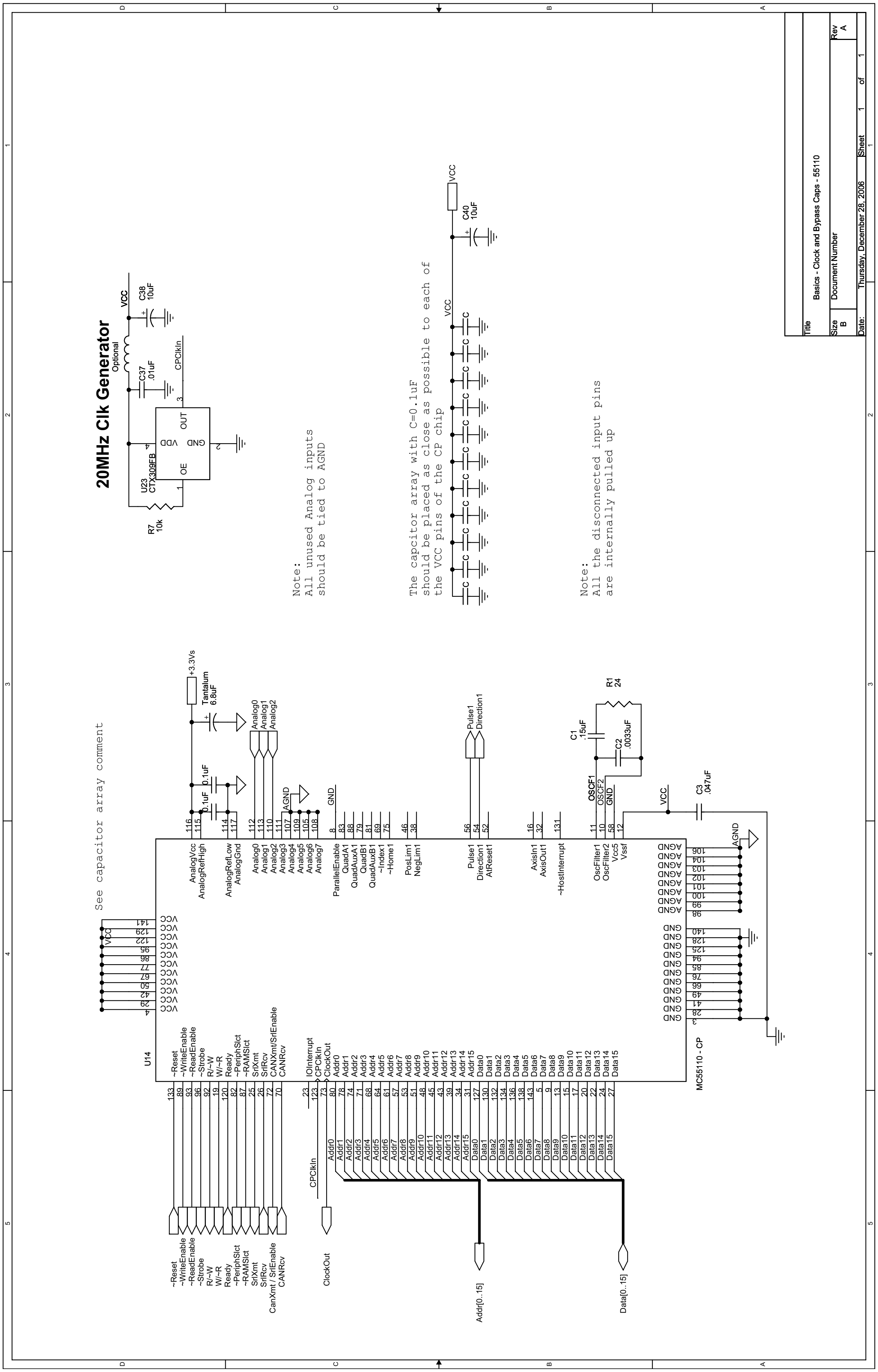
6.6.3 Grounding and Decoupling

Each component should be decoupled with the use of large capacitors, usually tantalum 6.7-10 µF in parallel, with a set of 10-100 nF ceramic capacitors placed as close as possible between each one of the power supply pins and ground. This general rule applies to all analog and digital components, although in some of the schematics that follow these capacitors are not shown for reasons of brevity. In some cases, especially in analog parts, it may be beneficial to run a separate power line from the power supply to the component in order to prevent power supply fluctuations from impacting low-level signal components.

The same points should be considered when designing the ground. The schematics in figures 6-2 and 6-3 show a star connection at one point in the power supply. Care should be taken to ensure that voltage differences do not accumulate between the grounds, especially in mixed signal components such DACs and ADCs.

Additional isolation, for example ferrite beads, may be inserted between the analog and digital grounds to suppress high frequency ground noise. Some components, such as motor drivers, require special grounding. The system designer should refer to the component data sheets of selected components in order to ensure correct usage of the grounding methods.

Figure 6-2: Basics, clock and bypass caps, 55110



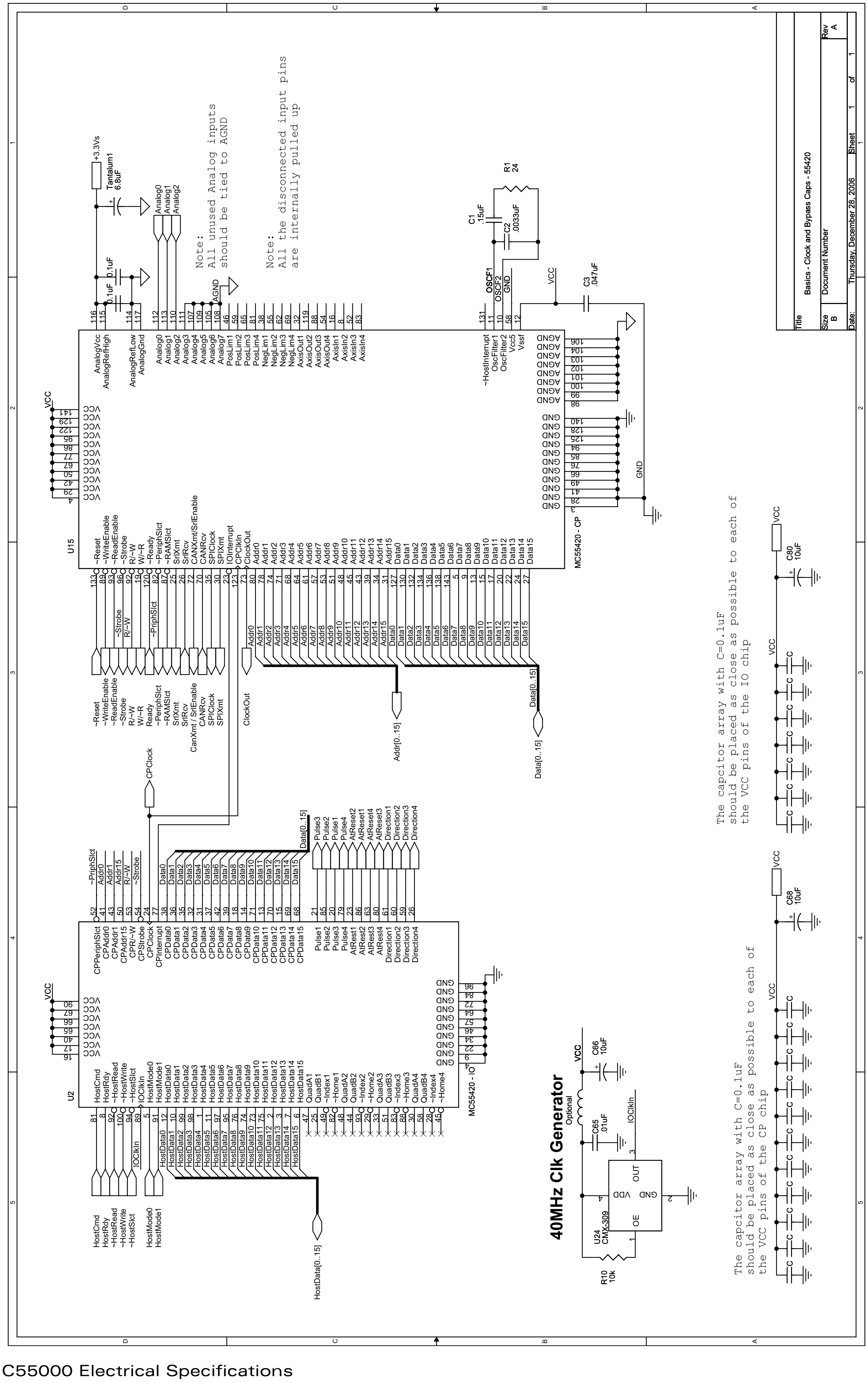


Figure 6-3: Basics, clock and bypass caps, 55420

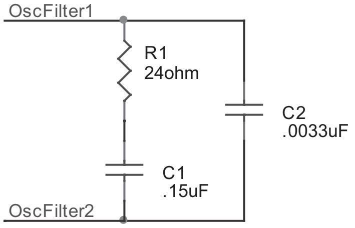
6.6.4 Decoupling of the On-chip ADC

The voltage supply to the ADC should be decoupled with the use of a 2.2-6.8 μF tantalum capacitor in parallel with a 0.01-0.1 μF ceramic capacitor placed as close as possible to the power supply and ground pins. For additional isolation purposes, an additional 0.01-0.1 μF ceramic capacitor should be placed across AnalogRefLow and AnalogRefHigh.

6.6.5 External Oscillator Filter

The circuit in Figure 6-4 shows the recommended configuration and suggested values for the filter which must be connected to the OscFilter1 and OscFilter2 pins of the CP chip. The resistor tolerance is ±5%, and the capacitor tolerance is ±20%.

Figure 6-4: Oscillator filter circuit



When applying a different clock frequency, the PLL's external loop filter circuit (capacitors C1 and C2, and resistor R1) should be varied as a function of the clock frequency. These reference values are detailed in the following table. C1 and C2 capacitors must be non-polarized.

CPClkIn [MHz]	R1 [Ω] (±5%)	C1 [μF] (±20%)	C2 [μF] (±20%)
5	5.6	2.7	0.056
10	11	0.68	0.015
15	16	0.33	0.0068
20	24	0.15	0.0033

6.6.6 Reset Signal

The CP accepts a reset signal, *~Reset*, which should go low after power-up or when an external reset event occurs. From the rising edge of this signal, the CP begins an initialization procedure, which is concluded within 1.5 milliseconds. During this period, the outputs of the CP and IO chips will be in an unknown state. In order to prevent signals in an arbitrary state from driving the motors, a disabling signal, *~ResetHold*, is generated. The *~ResetHold* signal is a ~2.2 msec extension to the active low period of the *~Reset* signal, and is used to disable the motor drivers during the initialization period. Figure 6-5 shows a circuit for generating the *~ResetHold* signal.

During the initialization period, the CP reads the two external configuration registers to determine the configuration for serial and CAN communication. Refer to sections 6.7.2 ("Interfacing to Off-board Hosts Through Asynchronous Serial Communications") and 6.8 ("CAN Communication Interface") for more information.

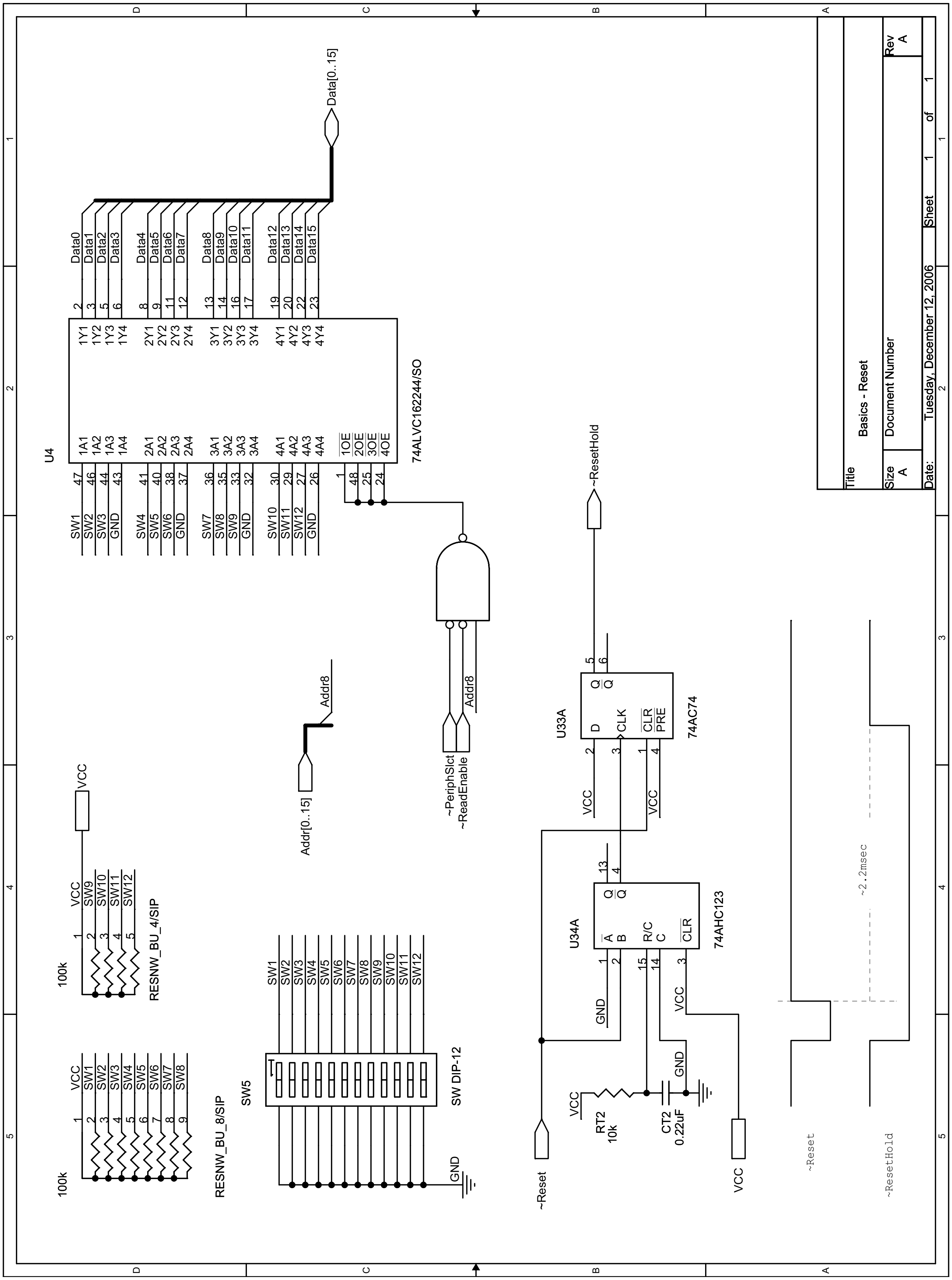


Figure 6-5: Basics, reset

6.7 Serial Communication Interface (SCI)

In this section, the serial communication interface to the host is described. Figure 6-6 shows circuitry used to configure the SCI port on power-up. This circuitry may be omitted if the default configuration values are suitable.

Subsequent sections demonstrate the use of RS-232, RS-422 and RS-485 line-drivers for interfacing to a remote host.

6.7.1 SCI Configuration During Power-up or Reset

On power-up or after a reset, the CP configures the SCI according to the 16-bit value residing at the peripheral address 200h. If a value of FFFFh is read, then the SCI is configured to its default configuration: 57,600 baud, no parity, one stop bit, point-to-point mode. Since the data bus inputs are internally pulled-up, the pins may remain disconnected and the default configuration will take effect at power-up or reset. If a non-default SCI configuration is required the circuitry presented in the reference schematic may be used.

Note that after communication has been established, the SCI configuration may be altered via the SetSerialPortMode command.

The following should be observed when designing the power-up/reset SCI circuitry:

1 The MC55000 peripheral address map is arranged so that Addr9 is dedicated for SCI configuration.

2 The following logic condition for presenting the setup word on the data bus should be used:

~SCISetupDataEnable = ~ReadEnable + ~PeriphSlct + ~Addr9

Where:

~SCISetupDataEnable	When high the tri-state buffer outputs are placed in a high-impedance state.
~ReadEnable	When low the bus is in a read cycle.
~PeriphSlct	When low the peripheral address space is being addressed.

The logic may be implemented in a PLD, and its propagation delay should not exceed 20 nsec; assuming an enable time for the tri-state buffer of less than 10 nsec.

3 The DIPswitch resistors of ~100K ensure sufficient VIH level. In case of a zero input, a current of ~33 µA will be flowing between VCC and GND. This may result in a worst-case scenario of ~0.55 mA when the all-zero word is encoded.

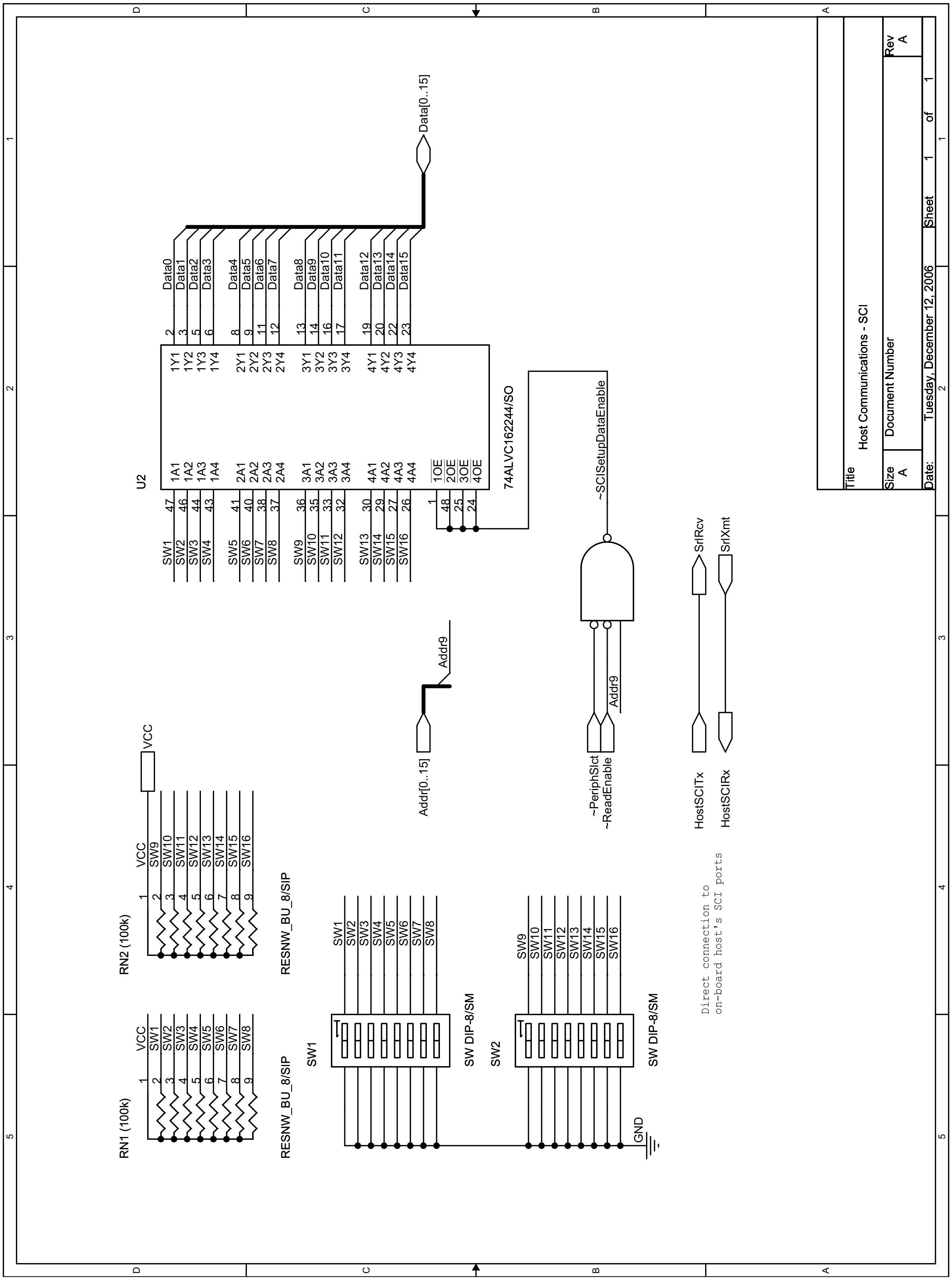


Figure 6-6: Host communication, SCI

6.7.2 Interfacing to Off-board Hosts Through Asynchronous Serial Communications

When the host and motion processor are located on the same physical board it is most likely that simply wiring the transmit and receive lines directly between the host and CP chip is all that is required (assuming they are both 3.3V CMOS devices). When the host is remote and the interface requires longer communication lines, achieving reliable communication is more involved.

TIA/EIA standards provide reliable communication over varying cable lengths and communication rates. The most commonly used standards are RS-232, RS-422 and RS-485. These standards are separated into two categories: single-ended and differential. RS-232 is a single-ended standard allowing for moderate communication rates over relatively short cables. RS-422 and RS-485 are differential, offering higher data rates and longer cable runs.

Line drivers and receivers (transceivers) are commonly used in order to mediate between the cable interface and the digital circuitry signal levels. Although RS-485 transceivers also support the RS-422 electrical specification (the reverse is not true), there are several design considerations that should be taken into account when deciding which of these two communication methods is the best fit for an application.

- Full-duplex vs. half-duplex

  The terms full-duplex and half–duplex are used to distinguish between a system having two separate physical communications lines from one having one common line for transmission and reception.

- Line contention

  This problem can occur in half-duplex systems. Most line-drivers supply physical protection against such conditions but there is no automatic recovery of lost data in these levels. When interfacing the Magellan to a half-duplex communication system the designer should note that the turn-around time for command processing and response is at least 1 byte at the current baud rate. As a result the host should release the communication line before this time elapses so that contention can be avoided.

- Termination impedance

  Long cables and/or high data rates require termination resistors if the transceiver is located at the end of the transmission lines. One way to determine if termination is required is if the propagation delay across the cable is larger than ten times the signaling transition time. If this condition is satisfied, then termination is required. The RS485 standard specifies the signaling transition time to be less than 0.3 times the signaling period, thus imposing an upper limit on the maximum cable length for a specified baud rate.

  The termination resistor should match the characteristic impedance of the cable with 20% tolerance. Resistors with a value of 80-120Ω are typically used. For RS-422, only the receiver end should use a termination resistor, due to the communication line being unidirectional (full duplex). Note that if the transceiver is not placed at the ends of the cable, no termination resistors are required. However, the stubs should be kept as short as possible to prevent reflections.

The schematic in Figure 6-7 employs the ADM3202 and ADM3491 transceivers as an example of RS232 and RS485/422 interfaces respectively. Other RS232 transceivers may be used, such as Maxim's MAX3321E. The ADM3491 circuitry can be configured for both full-duplex and half-duplex communications, and may include termination resistors. As an alternative, transceivers from the MAX307xE family may be used.

The following table shows configuration options for the RS-485/422 circuitry of Figure 6-7**.**

Configuration	Jumper Position	Application
Half Duplex[1]	JMP1/2/3 in 2-3	RS-485 in multipoint system
Full Duplex	JMP1/2/3 in 1-2	RS-422 or RS-485 in point to point system
Termination on[2]	JMP4/5 in 1-2	Both RS-485 and RS-422. For high transmission rates and/or long cable. Only when placed at the end of the cable.
Termination off	JMP4/5 in 2-3	Both RS-485 and RS-422. For low transmission rates and short cable. Or when placed at the middle of the cable.

[1] JMP3 should only be placed be in the half duplex state (2-3) if multi-point communication is being used.

[2] Note that the reference circuitry does not support resistance termination on the transmitting side when operated in full duplex because it is assumed that RS 485 will only be used in the half-duplex configuration.

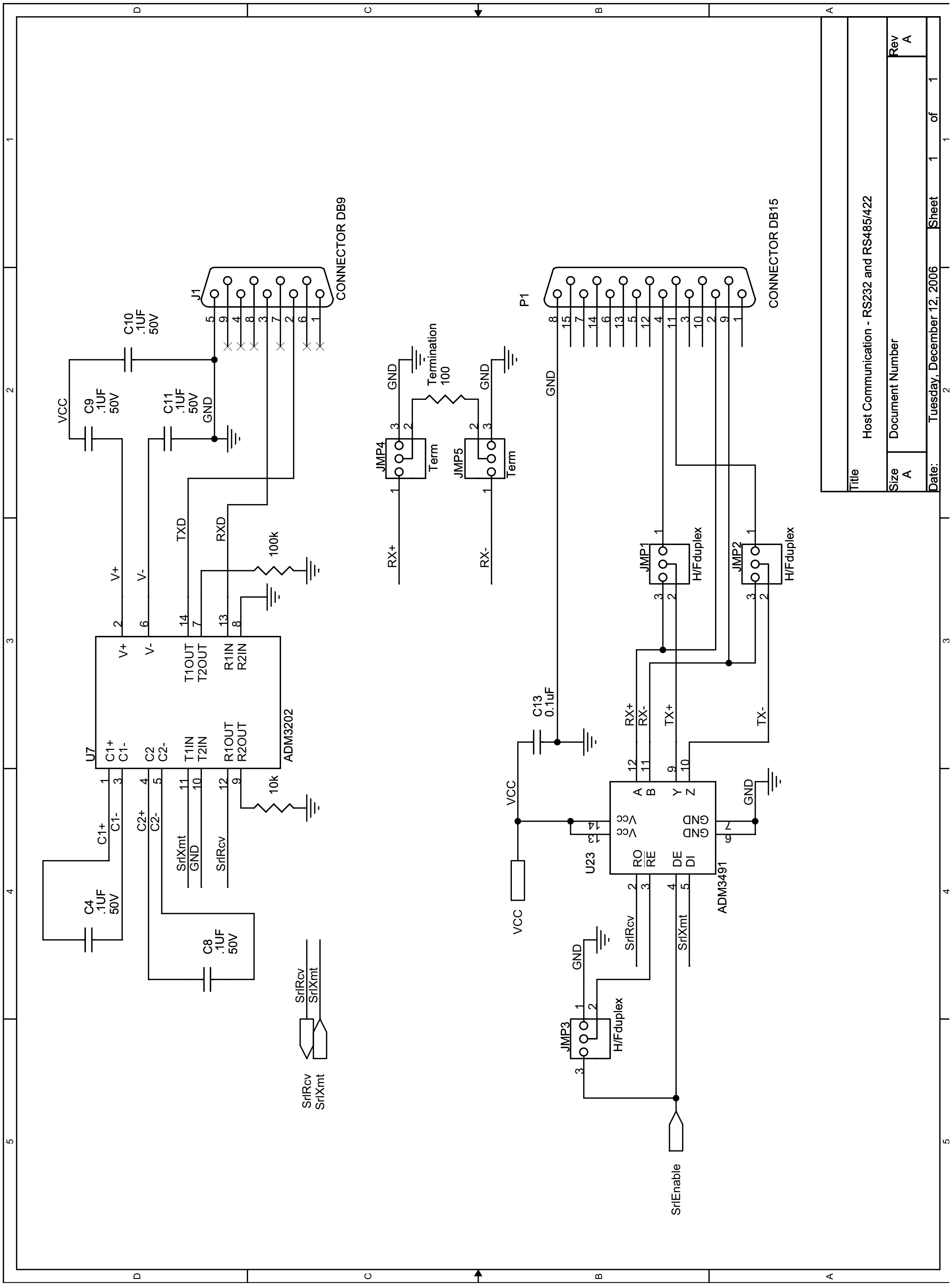


Figure 6-7: Host communication, RS232 and RS485/422

6.8 CAN Communication Interface

The following example illustrates an interface to a CAN backbone using of TI's SN65HVD232 transceiver, which supports ISO 11898 standard. Generally the CAN high-speed standard ISO 11898 provides a single line structure as network topology. The bus line is terminated at both ends with a single termination resistor of ~120ohms. However in practice some deviation from that topology may be needed to accommodate appropriate drop cable lengths and particular applications. Consult CAN ISO 11898 standard for more information on termination schemes and EMC considerations.

6.8.1 CAN Configuration During Power-up or Reset

On power-up or after reset, the CP configures the CAN controller according to the 16-bit value residing at the peripheral address 400h. If a value of FFFFh is read, then the CAN controller is configured to its default configuration: 20 kbps with a NodeID of 0. Since the data bus inputs are internally pulled-up, the pins may remain disconnected and the default configuration will take effect at power-up or reset. If a non-default CAN configuration is required the circuitry presented in the reference schematic (Figure 6-8) may be used.

Note that after communication has been established, the CAN configuration may be altered via the SetCANMode command.

More advanced CAN bus drivers such as the SN65HVD230 supply a programmable input pin which may be used to adjust the rise and fall times of the transmitter. This may be important in unshielded, low-cost systems in order to reduce electromagnetic interference. The pin may be hard-wired through a resistor to ground (refer to the component data sheet for calculating the resistor's value), or the host may control this pin by introducing additional circuitry attached to the I/O user space of the CP for a more flexible and tunable design.

The following should be observed when designing the power-up/reset CAN circuitry.

1 The MC55000 peripheral address map is arranged so that Addr10 is dedicated for CAN configuration.

2 The following logic condition for presenting the setup word on the data bus should be used.

~CANSetupDataEnable = ~ReadEnable + ~PeriphSlct + ~Addr10

Where:

~CANSetupDataEnable	When high the tri-state buffer outputs are placed in a high-impedance state.
~ReadEnable	When low, the bus is in a read cycle.
~PeriphSlct	When low, the peripheral address space is being addressed.

The logic may be implemented in a PLD, and its propagation delay should not exceed 20 nsec; assuming an enable time for the tri-state buffer of less than 10 nsec.

3 The DIPswitch resistors of ~100K ensure sufficient VIH level. In case of a zero input, a current of ~33 µA will be flowing between VCC and GND. This may result in a worst-case scenario of ~0.55 mA when the all-zero word is encoded.

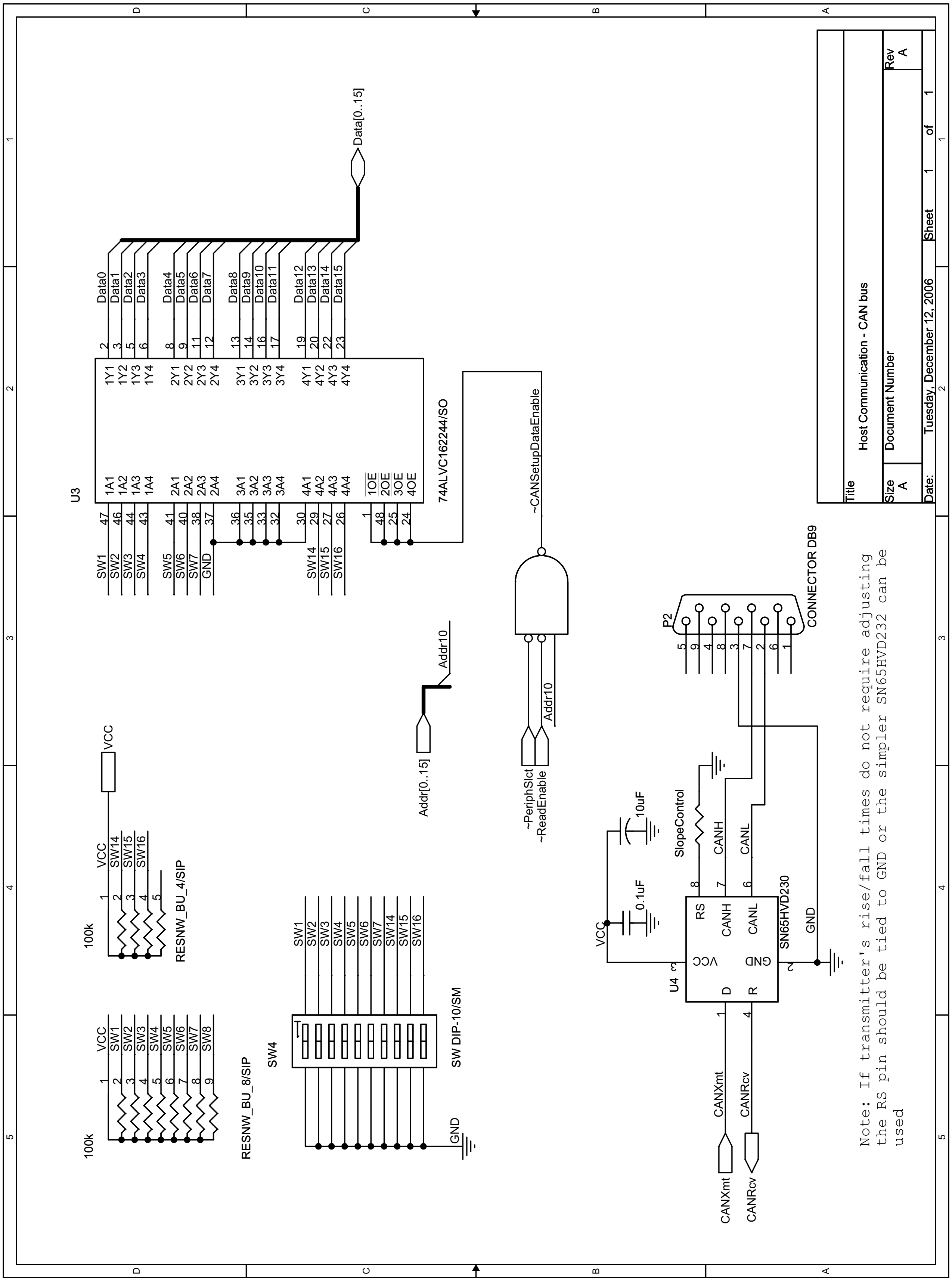

Host Communication - CAN bus
Title
Document Number
Size A
Rev A
Date: Tuesday, December 12, 2006
Sheet 1 of 1
U3
74ALVC162244/SO
1A1 1A2 1A3 1A4 2A1 2A2 2A3 2A4 3A1 3A2 3A3 3A4 4A1 4A2 4A3 4A4
1Y1 1Y2 1Y3 1Y4 2Y1 2Y2 2Y3 2Y4 3Y1 3Y2 3Y3 3Y4 4Y1 4Y2 4Y3 4Y4
1OE 2OE 3OE 4OE
Data0 Data1 Data2 Data3 Data4 Data5 Data6 Data7 Data8 Data9 Data10 Data11 Data12 Data13 Data14 Data15
Data[0..15]
SW1 SW2 SW3 SW4 SW5 SW6 SW7 GND SW14 SW15 SW16
~CANSetupDataEnable
Addr10
Addr[0..15]
~PeriphSlct
~ReadEnable
P2
CONNECTOR DB9
VCC
100k
RESNW_BU_4/SIP
RESNW_BU_8/SIP
SW4
SW DIP-10/SM
GND
U4
SN65HVD230
RS CANH CANL GND VCC D R
SlopeControl
CANH
CANL
10uF
0.1uF
CANXmt
CANRcv
Note: If transmitter's rise/fall times do not require adjusting
the RS pin should be tied to GND or the simpler SN65HVD232 can be
used

Figure 6-8: Host communication, CAN bus

6.9 External Memory

Utilizing its external bus, the Magellan Motion Processor can interface with two types of external memory: asynchronous SRAM and synchronous dual port RAM (DPRAM). External memory is used for trace data storage and is optional. SRAM is typically used in designs that do not require real-time access to the data. DPRAM permits high speed downloading of trace data and is most applicable in applications where the data is being downloaded and analyzed on a real-time basis.

6.9.1 CP External Memory Interface

The MC55000 external bus is comprised of the ***Addr[0..15]*** and ***Data[0..15]*** signals. The signals ***~WriteEnable***, ***~ReadEnable***, ***~RAMSlct***, ***W/~R***, ***R/~W*** and ***~Strobe*** are used in conjunction with the address bus signals for controlling access to the attached memory device. The ***Ready*** input signal may also be used to insert wait-states for accessing slower memory devices. Signal timing information is given in chapters 3 and 4 of this manual.

All signals are time referenced to the ***ClockOut*** signal, which has a nominal 25 nsec period. The MC55000 can directly access 32Kx16 bits of external memory and uses the 15 least significant bits of the address bus. ***Addr15*** is not used. Larger external memories may be used by adding a page register, as detailed in the following section.

6.10 Asynchronous SRAM

The following schematic (Figure 6-9) illustrates a pair of IDT71V424SA15 512Kx8 SRAMs with a 15 ns access time interfaced to the MC55000, resulting in a total of 16 pages of storage. Expansion to 32 pages is easily achieved with four IDT71V428 1024Kx8 SRAMs. Memory blocks are accessed with the use of a page register. The IDT71V424SA15 is an asynchronous SRAM which is controlled by three input signals: chip select (~CS), output enable (~OE), and write enable (~WE). The SRAM is interfaced with the MC55000 output signals as shown in the following table.

Device	Signal Name		
MC55000	*~RAMSlct*	*~ReadEnable*	*~WriteEnable*
IDT71V424SA15	*~CS*	*~OE*	*~WE*

Note that the selected SRAM device should meet the timing requirements of the MC55000 output signals. Usually, asynchronous SRAMs with cycle times of less than 1.5 ***ClockOut*** cycles will meet this requirement. One example is the CY7C1020CV33-15 32Kx16 SRAM which has a 15 nsec access time. This device may be used to provide one page of storage.

If larger external memory is required, several pages may be addressed (up to 64K pages) by using a page register. The page register may be accessed at the peripheral address 2000h and the ***Addr13*** signal can be used as a chip-select. The page register operates the extra address lines required to interface with larger external memory devices. Before the external memory write or read cycle, the CP performs a write to the page register to select the required page.

The use of larger SRAM chips is recommended. If the capacity of one SRAM is not adequate, multiple chips may be cascaded. The two common methods for cascading SRAM chips are:

- Using high capacity, lower organization (x8 or x4) chips. In this configuration SRAMs share the same address bus and each SRAM is wired to a different portion of the data bus.
- Using the SRAM's chip-select input(s) as an additional address line. This option requires a decoder to map the address to the appropriate chip select signal. The propagation delay of the decoder must be below 0.5 of the ***ClockOut*** period. In addition the total access time should not exceed 1.5 ***ClockOut*** cycles including the SRAM access time.

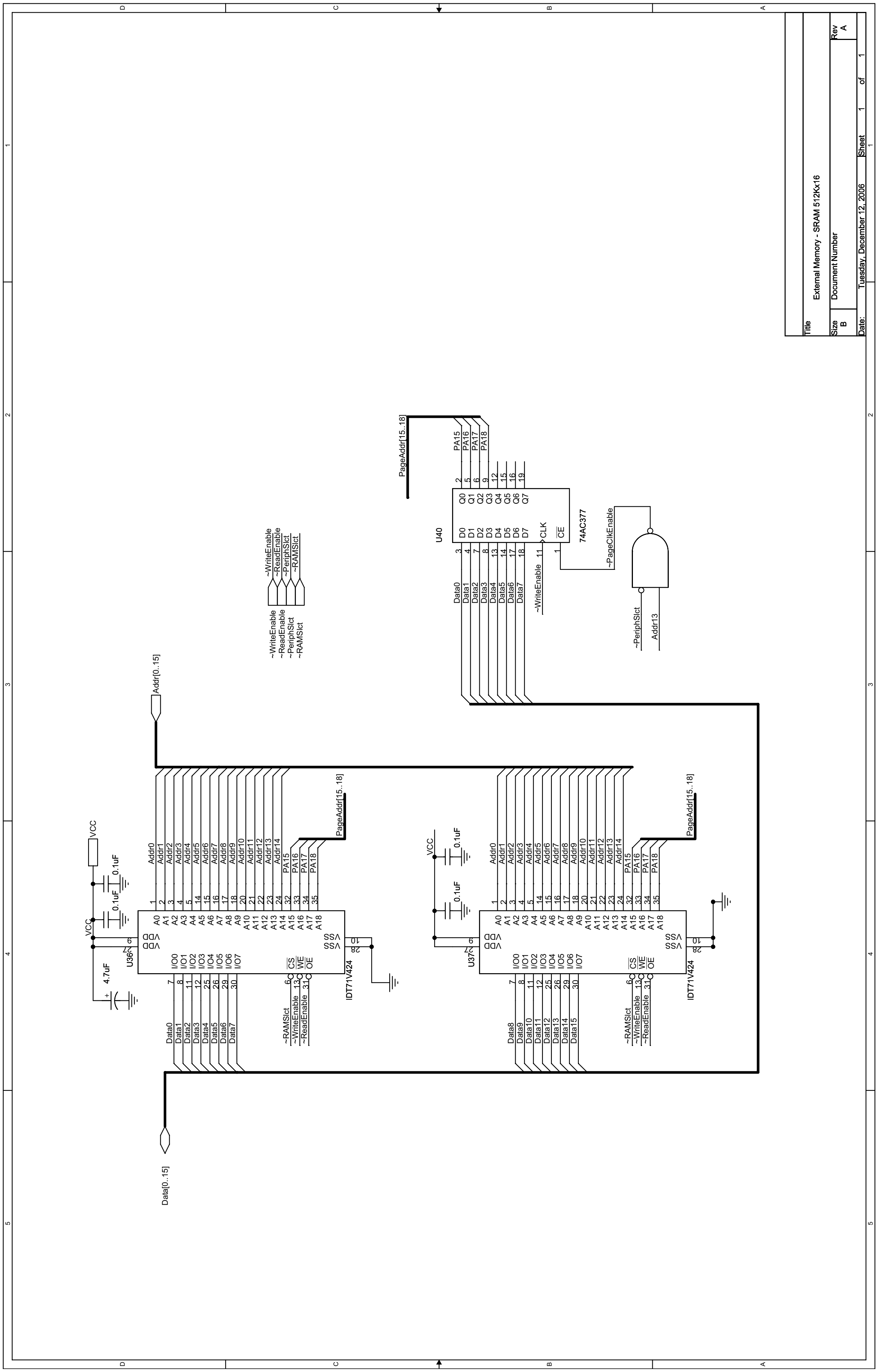

Title
External Memory - SRAM 512Kx16
Size B
Document Number
Rev A
Date: Tuesday, December 12, 2006
Sheet 1 of 1
U36
U37
U40
IDT71V424
74AC377
Addr[0..15]
Data[0..15]
PageAddr[15..18]
~WriteEnable
~ReadEnable
~PeriphSlct
~RAMSlct
~PageClkEnable
Addr13
VCC
VDD
VSS
0.1uF
4.7uF
CS
WE
OE
CLK
CE
D0 D1 D2 D3 D4 D5 D6 D7
Q0 Q1 Q2 Q3 Q4 Q5 Q6 Q7
PA15 PA16 PA17 PA18
A0 A1 A2 A3 A4 A5 A6 A7 A8 A9 A10 A11 A12 A13 A14 A15 A16 A17 A18
I/O0 I/O1 I/O2 I/O3 I/O4 I/O5 I/O6 I/O7
Data0 Data1 Data2 Data3 Data4 Data5 Data6 Data7
Data8 Data9 Data10 Data11 Data12 Data13 Data14 Data15
Addr0 Addr1 Addr2 Addr3 Addr4 Addr5 Addr6 Addr7 Addr8 Addr9 Addr10 Addr11 Addr12 Addr13 Addr14
A B C D
1 2 3 4 5

Figure 6-9: External memory, SRAM 512Kx16

6.10.1 Slow Asynchronous SRAM

If the SRAM does not meet the timing requirements, it may still be connected to the MC55000 by adding wait states. In order to generate the wait-states, the ***Ready*** signal must be activated during read/write memory accesses. As long as the ***Ready*** signal is kept low during the rising edge of the ***ClockOut*** signal, the end of the current read/write cycle will be deferred to the next ***ClockOut*** rising edge.

The following schematic (Figure 6-10) contains a circuit and timing diagram for generating one wait state. Expanding to two or more wait states is similar. Note that adding wait states slows the operation of the MC55000 and therefore the number of wait states should be kept as low as possible if the device is expected to maintain normal operation. Contact PMD to determine if the use of wait states will affect device operation in your application.

The following timing restrictions apply when the device is operating with the standard 40 MHz ***ClockOut*** frequency:-

1 tp4 + tp2 < 22 ns

2 tp2+ ts1 < 19 ns

3 tp1 + tp4 + tp3 < 9.5 ns

Where tp1 and ts1 are the propagation delay and setup time of the D flip-flop. TPx is the propagation delay of logic gate Ux.

Notes:

1 In order to meet the above timing requirements high-speed gates may be used or the logic may be implemented in a fast PLD. Timing restriction 3 may be relaxed by the use of fast negative edge JK-FF, such as the 74LCX112, resulting in a very tight constraint on tp4.

2 If read and write cycles do not require the same number of wait states, then either ***~WriteEnable*** or ***~ReadEnable*** may replace the ***~Strobe*** signal. If ***~ReadEnable*** is used, then restriction 2 becomes more stringent: tp2 + ts1 < 7.5 ns.

3 If there is no need to add wait-states the ***Ready*** input pin may be left disconnected as it is internally pulled up.

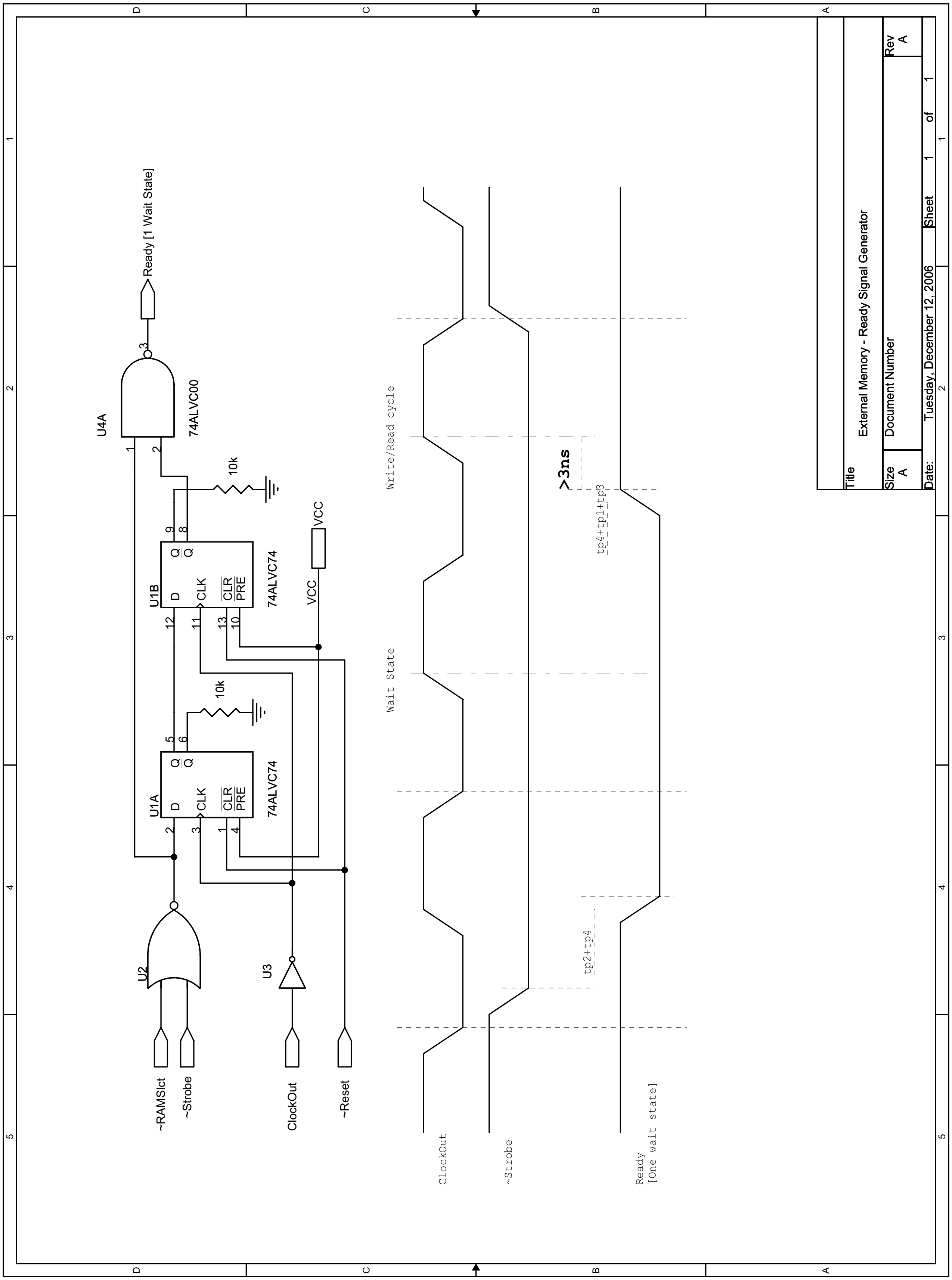


Figure 6-10: External memory, Ready signal generator

6.11 Dual Port Synchronous SRAM (DPRAM)

DPRAM is used in applications that require real-time access to the trace data, or applications where constant tracing is required but a large external memory block is not desirable because of cost or space limitations. The DPRAM is generally treated as a circular buffer with data being downloaded by the host on the fly. As a result, the size of the DPRAM can be minimized so long as the host is retrieving the trace data in a timely fashion.

Two options for interfacing the MC55000 to a DPRAM are presented here. In the first example, a Cypress CY7C09269-6 is used, which is a fast DPRAM. In the second example, the slower IDT70V9269S9 is used. Each access port of these devices may be independently programmed to be either in a pipe-lined or a flow-through mode. The interface to the MC55000 shown in both cases requires the DPRAM's port to be in the flow-through mode while the port mode used to interface to the host is determined according to its own requirements.

6.11.1 Option 1: Fast Flow-through DPRAM

In the following schematic (Figure 6-11), an interface to a fast, flow-though CY7C09269-6 DPRAM is shown. The interface follows the table below.

In order to meet the timing requirements, the *ClockOut* inverter should have a propagation delay of no more than 3 nsec. Note that using this scheme will result in multiple read and write cycles but this has no affect on functionality.

Device	Signal Name			
MC55000	*~RAMSlct*	*~WriteEnable*[1]	*~ClockOut*	*~ReadEnable*
CY7C09269-6	*~CE0*	*R/~W*	*Clk*	*~OE*

[1] In order to satisfy the setup and hold times of the CY7C09269, the WriteEnable is buffered on the rising edge of ClockOut before interfacing it to the ~CE0 input.

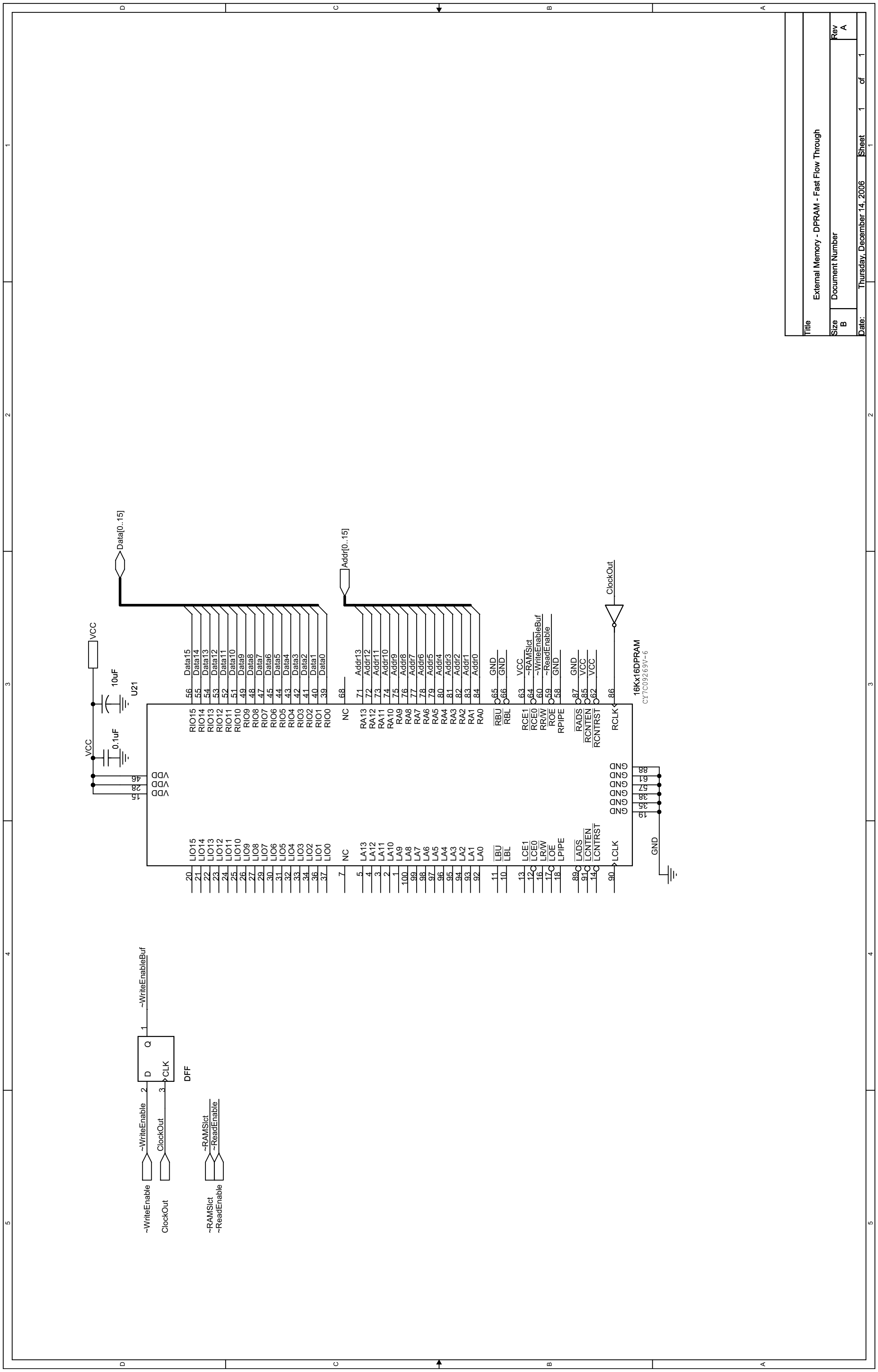

Title
External Memory - DPRAM - Fast Flow Through
Size
B
Document Number
Rev
A
Date:
Thursday, December 14, 2006
Sheet
1
of
1
Data[0..15]
Addr[0..15]
ClockOut
VCC
10uF
0.1uF
U21
16Kx16DPRAM
CY7C09269V-6
Data15 Data14 Data13 Data12 Data11 Data10 Data9 Data8 Data7 Data6 Data5 Data4 Data3 Data2 Data1 Data0
RIO15 RIO14 RIO13 RIO12 RIO11 RIO10 RIO9 RIO8 RIO7 RIO6 RIO5 RIO4 RIO3 RIO2 RIO1 RIO0
LIO15 LIO14 LIO13 LIO12 LIO11 LIO10 LIO9 LIO8 LIO7 LIO6 LIO5 LIO4 LIO3 LIO2 LIO1 LIO0
NC
Addr13 Addr12 Addr11 Addr10 Addr9 Addr8 Addr7 Addr6 Addr5 Addr4 Addr3 Addr2 Addr1 Addr0
RA13 RA12 RA11 RA10 RA9 RA8 RA7 RA6 RA5 RA4 RA3 RA2 RA1 RA0
LA13 LA12 LA11 LA10 LA9 LA8 LA7 LA6 LA5 LA4 LA3 LA2 LA1 LA0
RBU RBL
LBU LBL
RCE1 RCE0 RR/W ROE RPIPE
LCE1 LCE0 LR/W LOE LPIPE
RADS RCNTEN RCNTRST RCLK
LADS LCNTEN LCNTRST LCLK
VDD VDD VDD
GND GND GND GND GND GND
GND
VCC
~RAMSlct
~WriteEnableBuf
~ReadEnable
DFF
D
Q
CLK
~WriteEnable
ClockOut
~RAMSlct
~ReadEnable

Figure 6-11: External memory, DPRAM, fast flow-through

This page intentionally left blank.

6.11.2 Option 2: Flow-Through with Clock Signal Control

In this example, a clock input signal is generated to the DPRAM for the *~ReadEnable* and *~WriteEnable* output signals. The IDT70V9269S9 is used in a flow-through mode using the connection scheme shown in the table below.

Device	Signal Name			
MC55000	External Logic	*~RAMSlct*	*~WriteEnable*	*~ReadEnable*
IDT70V9269S9	*RCLK*	*~CE0*	*R/~W*	*~OE*

As shown in figures 6-12 and 6-13, each MC55000 read and write cycle will generate one clock pulse, *RCLK*, triggering the DPRAM read/write cycle. The *DslctClkIn* signal is generated in order to clock the DPRAM in the event of *~RAMSlct* going inactive high. Otherwise, the DPRAM will remain selected until the next read or write cycle. Other than power consumption, there is no functional effect. If power consumption is not an issue, or the bus is frequently written or read, this circuitry may be omitted.

Generating the clock for the DPRAM enables the use of lower speed DPRAM, such as the IDT70V9269S9, or the IDT70V9269S12 (as long as high-speed logic, tp < 5 ns, is used).

6.11.3 Host Interface to the DPRAM

This section provides details on the interface of a Motorola ColdFire MCF5282 microprocessor to a flow-through DPRAM. The MCF5282 utilizes several output signals for accessing external memory, including *~CSx*, *~OE*, *R/~W*, and *~TS*. The MCF5282 supports a 32-bit data bus but must be configured for 16-bit words in this case. The 16 active data lines are MSBs. The address bus is designed to access bytes; thus for 16-bit word organization the LSB of the address should not be used.

The table below shows the interface to a flow-through IDT70V9269 DPRAM and requires that the following control parameters have been selected via the chip select CSCR register assigned to the DPRAM.

- Auto-acknowledge enabled (AA=1)
- Port size set to 16 bits

Device	Signal					
MCF5282	*IA[1:14]*	*ID[16:31]*	*~CS1*	*R/~W*	*~TS+R/~W*	*~OE*
IDT70V9269	*A[0..13]*	*D[0:15]*	*~CE0*	*R/~W*	*CE1*	*~OE*

Note: The MCF5282's ~TA and ~TAE inputs are internally pulled up, thus can remain disconnected.

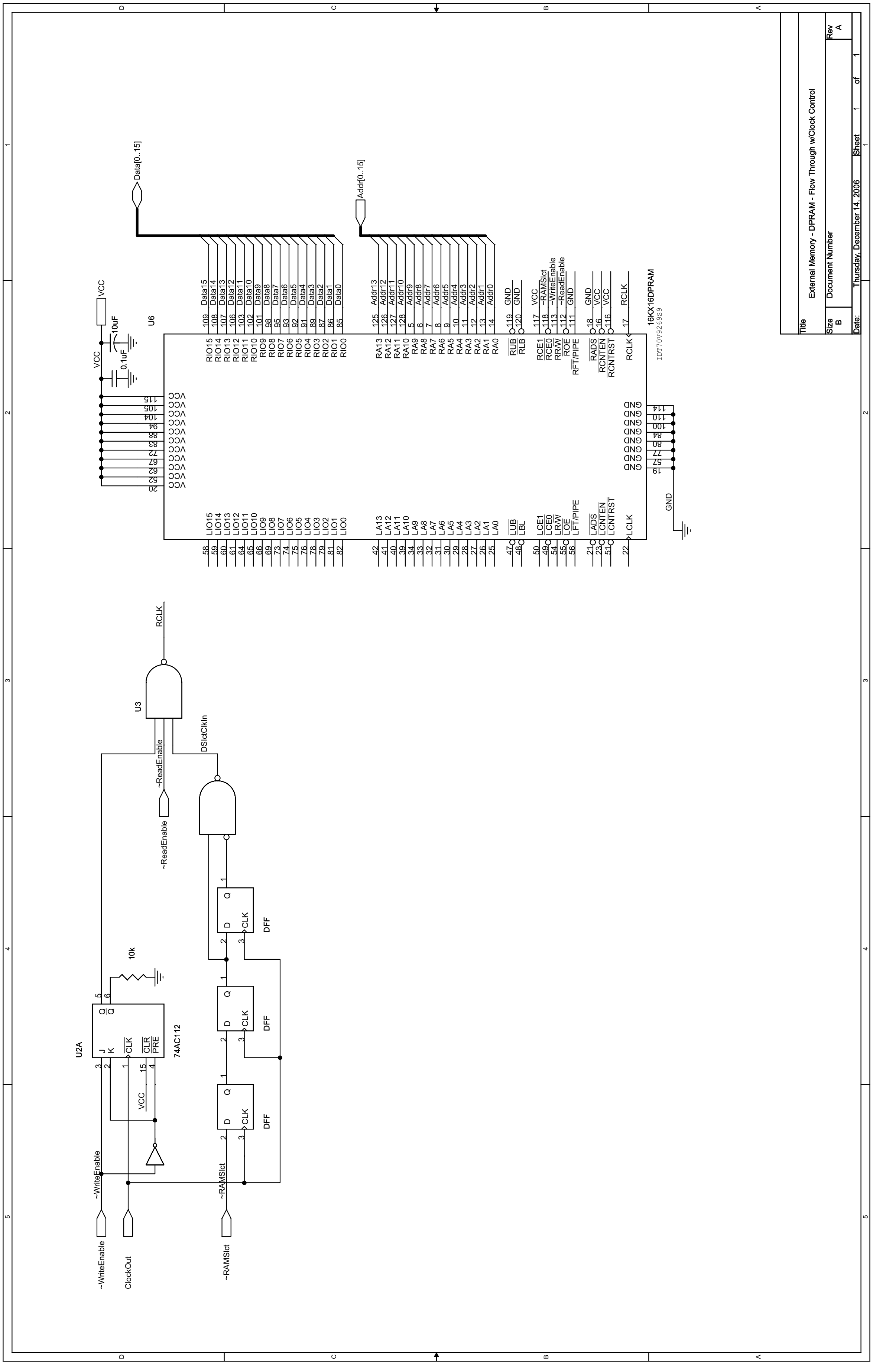
In the write cycle, the latching of the data into the DPRAM must be deferred because the MCF5282 presents the data onto the bus half a clock cycle later. To generate this delay, The MCF5282 *~TS* (Transfer Start) output signal is used. This signal marks the first bus clock cycle in the read/write operation and is used to keep the DPRAM deselected until valid data is presented by the MCF5282.

The MCF5282 supports a burst read/write (line transfer) mode which may be useful when downloading large trace data or when feeding an external profile.

The interface will support burst read/write cycles as long as the access cycles are internally terminated by the MCF5282 (AA=1). Note that when writing in burst mode, the first word will be written twice. This has no functional significance and is due to the 2-1-1-1 cycle pattern used by the MCF5282 in a line transfer mode.

Of importance is that the microprocessor's BUS clock should not exceed the speed of the DPRAM. For example, for the IDT70V9269S9, the MCF5282's BUS clock should not exceed 40 MHz.

Figure 6-12: External memory, DPRAM, flow-through with clock signal control



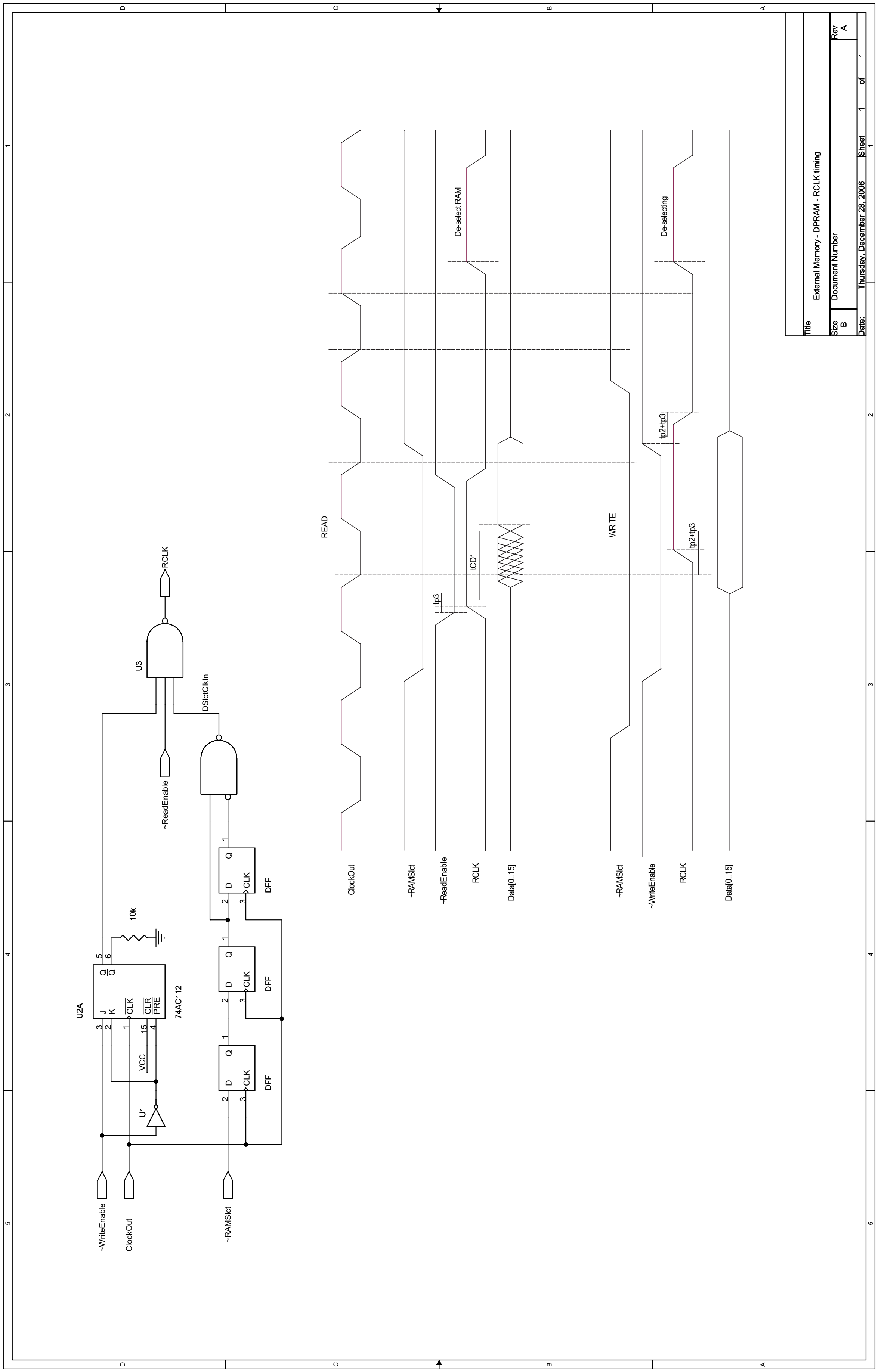


Figure 6-13: External memory, DPRAM, RCLK timing

6.11.4 Host DPRAM Management

Whenever data trace is in operation, the host must monitor the RAM in order to prevent overwrites. This is generally accomplished using timers to poll the MC55000 address pointer at regular intervals. It may also be assisted with the use of an external interrupt that indicates the condition of the RAM. The schematic in Figure 6-14 includes an example for generating a signal based upon Addr12 from the MC55000. It will generate a rising edge periodic interrupt whenever the DPRAM is being written to in the 1000h and 3000h address blocks, allowing the DPRAM to be half filled in between interrupts.

6.12 Using the On-chip ADC

In this section two types of conditioning circuits which interface to the on-chip analog-to-digital converter are demonstrated. The first circuit interfaces to a single-ended voltage signal, and the second circuit to a differential voltage signal. The conditioning circuits should be adjusted appropriately in order to meet the system's requirements.

The MC55000 is equipped with an eight-channel 10-bit ADC. The sampling rate of each channel is 57.8K samples per second, and the sample-and-hold time per channel is 1.6 µsec. The sampling capacitor is 30 pF, and the sampling resistor is in the range of 100 - 200Ω. In order to meet the timing requirements, the output impedance of the conditioning circuitry, as seen by the ADC's inputs, should not exceed 6.1 kΩ.

The digital value derived from the input analog voltage is determined using the following formula.

$$\text{Digital value} = 1023 \text{ x } (\text{input voltage} - V_{REFLO}) / (V_{REFHI} - V_{REFLO}) \qquad (1)$$

Where V_{REFLO} and V_{REFHI} are the voltages applied at ***AnalogRefLow*** and ***AnalogRefHigh*** pins, respectively.

The ADC's performance is guaranteed when V_{REFLO} = AGND and V_{REFHI} = AVCC. Not adhering to these values may result in performance degradation.

The ADC power supply should be decoupled with the use of a 2.2-6.8 µF tantalum capacitor in parallel with a 0.01-0.1 µF ceramic capacitor placed as closely as possible to the power supply and ground pins. An additional 0.01-0.1 µF ceramic capacitor should be placed across ***AnalogRefLow*** and ***AnalogRefHigh***.

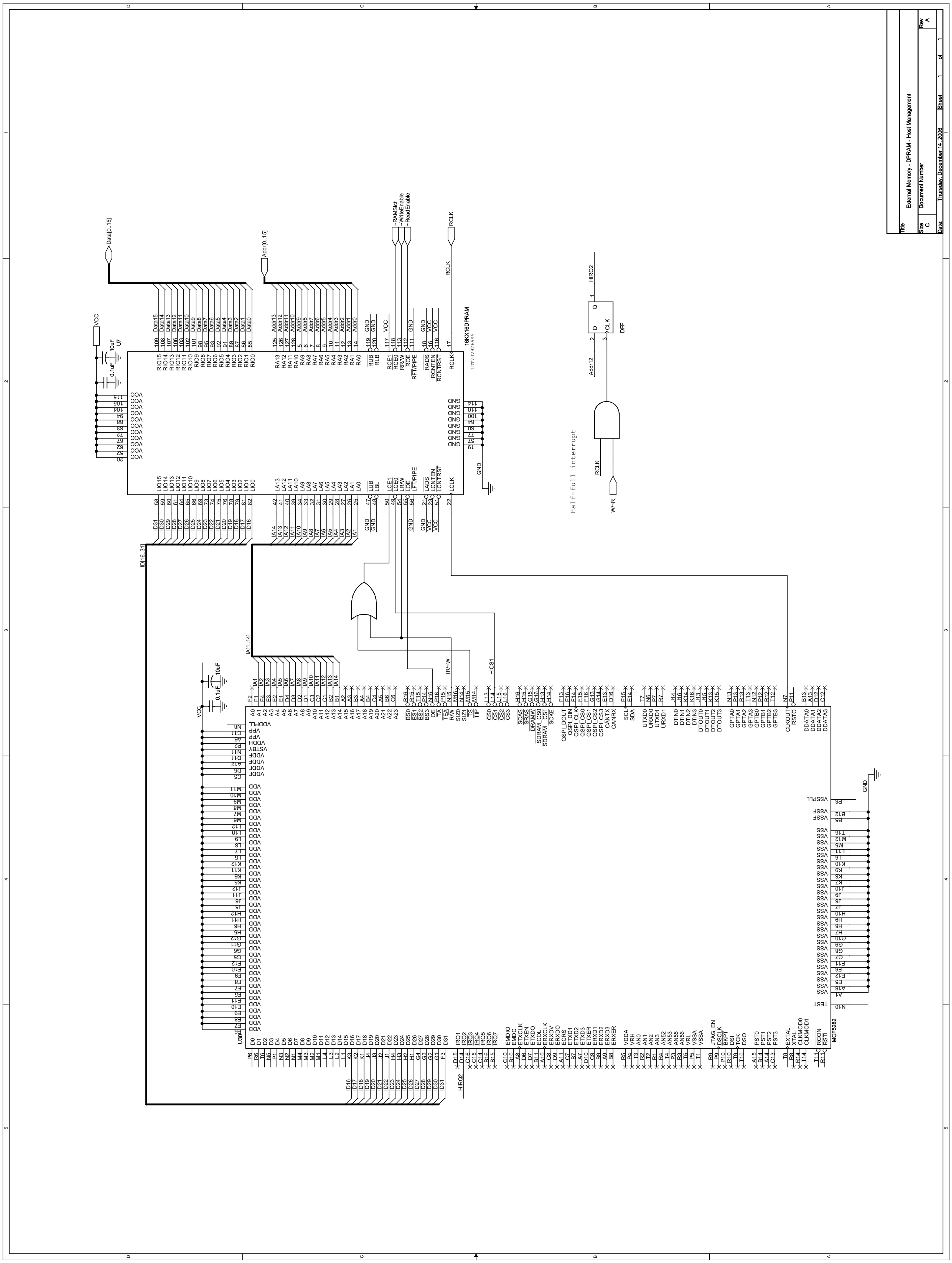


Figure 6-14: External memory, DPRAM, host management

6.12.1 Single-ended Interface

The following schematic (Figure 6-15) is a single-ended conditioning circuit that may be used for interfacing an on-board temperature sensor, the RTI ACW-027(refer to application notes). The input signal, V_T, is a single-ended voltage signal with a range of 0.45 - 2.9V and it is assumed to be varying slowly, at no greater than 100 Hz.

The goal of the conditioning circuitry is to match the analog signal to the ADC's voltage range and supply it with the required power. The conditioning circuit should be kept as simple as possible and make use of a single +3.3V supply.

6.12.1.1 Conditioning Circuitry and Op-amp Selection

Because the input is a voltage signal, an inverting amplifier is used to ensure a large input impedance. The operational amplifier should have rail-to-rail inputs/outputs with a unipolar supply. The TLV2471 is recommended as it can swing to within 180 mV of each supply rail while driving a 10 mA load.

The functionality of the circuitry at DC is depicted in equation (2).

$$Vout = V_T\left(1+\frac{Rf}{Rg+R_3\|R_2}\right)-Vs\cdot\frac{R_2}{R_2+R_3}\cdot\frac{Rf}{Rg+R_3\|R_2} \tag{2}$$

The gain of the circuitry is calculated in this manner so as to accommodate the full output swing of the op-amp, and to match it to the input swing of V_T. This is shown in the following equation.

$$\left(1+\frac{Rf}{Rg+R_3\|R_2}\right)=\frac{3.3-2\cdot 0.18}{2.9-0.45}=1.2 \tag{3}$$

Additionally, the circuitry should bias the output so that when V_T reaches the lowest value of interest, the op-amp also reaches its lowest output voltage. Applying (3) and calculating equation (2) at Vout = 0.18, and V_T = 0.45 results in the following.

$$\frac{R_2}{R_2+R_3}=0.5455 \tag{4}$$

Selecting R_2 = 12k (1%), R_3 = 10.0k (1%), Rg = 19.6k (1%), Rf = 4.99k (1%) satisfies equations (3) and (4), while maintaining load currents in the working range of the op-amp and ADC.

Note that if the input voltage V_T is linear within the supply voltage Vs, then the variation and sensitivity of the circuitry in Vs is relatively small since the same variations will affect the ADC. This will cancel out the variation's effects on the conditioning circuitry.

6.12.1.2 Rlp and Clp values

A low-pass RC filter is used to eliminate noise and prevent aliasing. Additionally, it is used to limit the load on the op-amp, which enables it to swing as close as possible to its rails.

Using Rlp = 1 kΩ and a ceramic Clp = 0.05 μF will result in a low-pass filter with a 3 dB point at f_0~3 kHz (which is assumed to be at least one order larger than the signal's bandwidth). The capacitor should be placed as close as possible to the ADC input pin, as it partially drives the sample capacitor of the ADC.

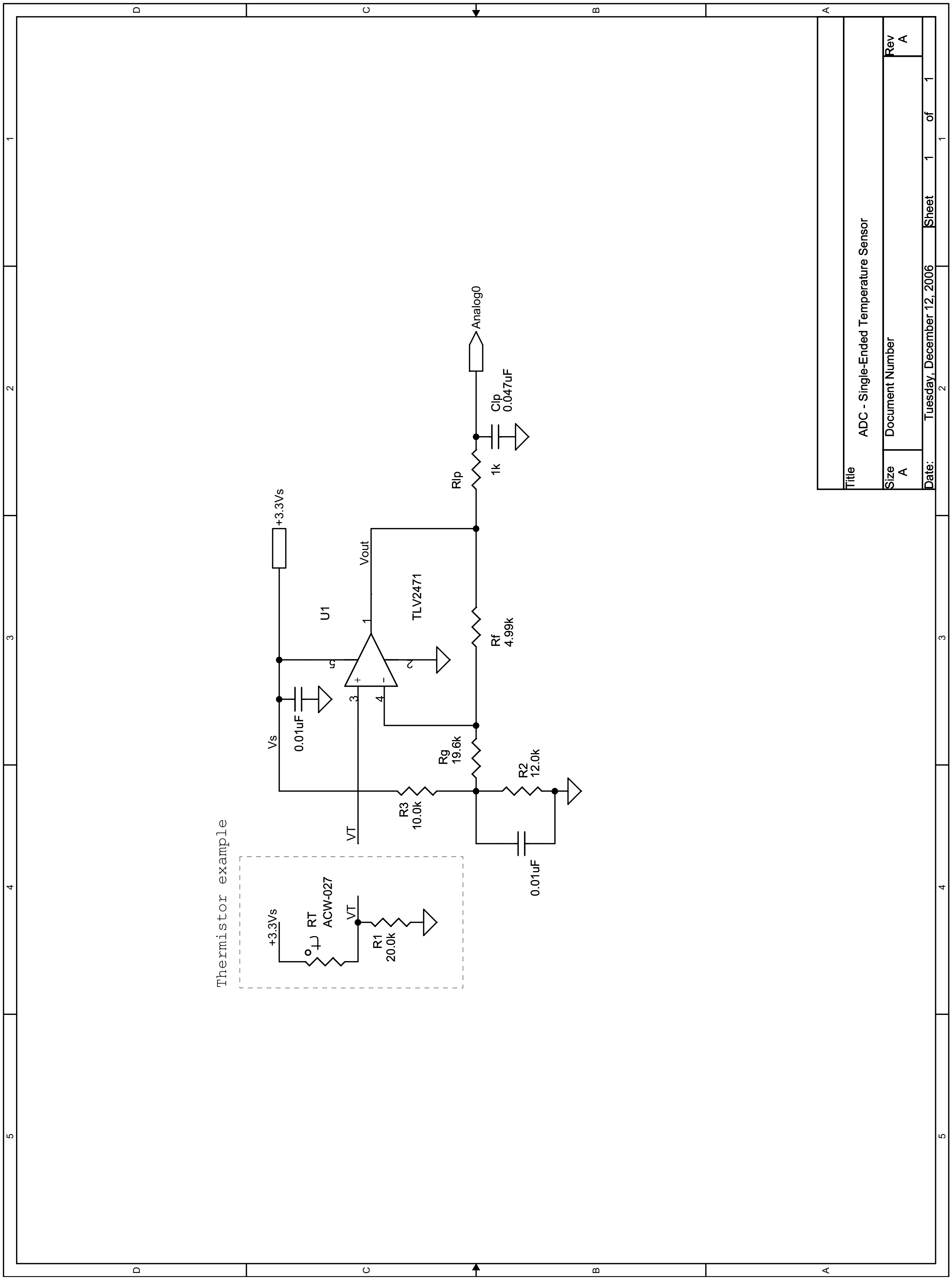


Figure 6-15: ADC, single-ended temperature sensor

6.12.2 Differential Interface

The input signal is assumed to be differential, Vin+ and Vin-. The voltage signal is in the range of (Vin+ - Vin-) = [-3V,+3V], and slowly varying (not greater than 100 Hz).

The goal is to condition the differential input signal to fit the ADC's voltage range, and supply it with the required drive. For example, the circuitry may be used to interface to a resonator sensor rate such as the RRS75 from Inertial Science, Inc. Additional ADC channels may be used simultaneously in order to expand the dynamic range of the ADC.

6.12.2.1 Conditioning Circuitry

The purpose of this interface is to generate a signal with the following format.

Analog0 = G(Vin+ - Vin-) +Vref, with nominal G = 0.55 and Vref = 1.65.

The interface shown in Figure 6-16 forms an instrumentation amplifier. The high input impedance of the instrumentation amplifier is highly desirable to eliminate voltage drops and CMRR concerns due to the signal source output impedance.

The following equation describes the functionality of the conditional circuitry at DC.

$$RO = (Vin+) \cdot G - (Vin-) \cdot m + Vref \cdot [(R_2+R_1) / R_2] \cdot [R_4 / (R_3+R_4)] \qquad (7)$$

where:

$$G = [R_1 / R_2] \cdot [(R_5+R_g) / R_g] + [R_6 / R_g] \cdot [R_3 / (R_3+R_4)] \cdot [(R_2+R_1) / R_2]$$

$$m = [(R_6+R_g)/ R_g] \cdot [R_3 / (R_3+R_4)] \cdot [(R_2+R_1) / R_2] + [R_1 / R_2] \cdot [R_5 / R_g]$$

Selecting $R_3 = R_1$ and $R_4 = R_2$ will result in the following simplified version:

$$RO = (Vin+ - Vin-) \cdot [R_1 / R_2] \cdot [1+ (R_5+R_6) / R_g] + Vref \qquad (7a)$$

Specifying resistors $R_3 = R_1 = 10.0$ kΩ (%1), $R_4 = R_2 = 100.0$ kΩ (1%), $R_5 = R_6 = 20.0$ kΩ (1%), and nominal $R_g = 8.9$ kΩ, will result in the desired G = 0.55. The importance of having matching pairs of resistors should be evident from equation (7). If matching is not done common mode voltage will be introduced.

The OPA2345 is an input/output rail-to-rail operational amplifier with low voltage bias, and high CMRR. It tolerates input common voltages of ± 0.3V from its rails. As a protection measure, the addition of Shottky diodes with 0.3V forward voltage, such as 20L15T, is recommended (but not shown). The output swing of the op-amp is closely related to the load current. In order to make this current as low as possible, a resistor is added at the output of the op-amp. Adding a capacitor forms a LPF, with a 3 dB cut-off at ~3 kHz. The capacitor should be placed as close as possible to the ADC input pins, and is partially used to drive the sampling capacitor.

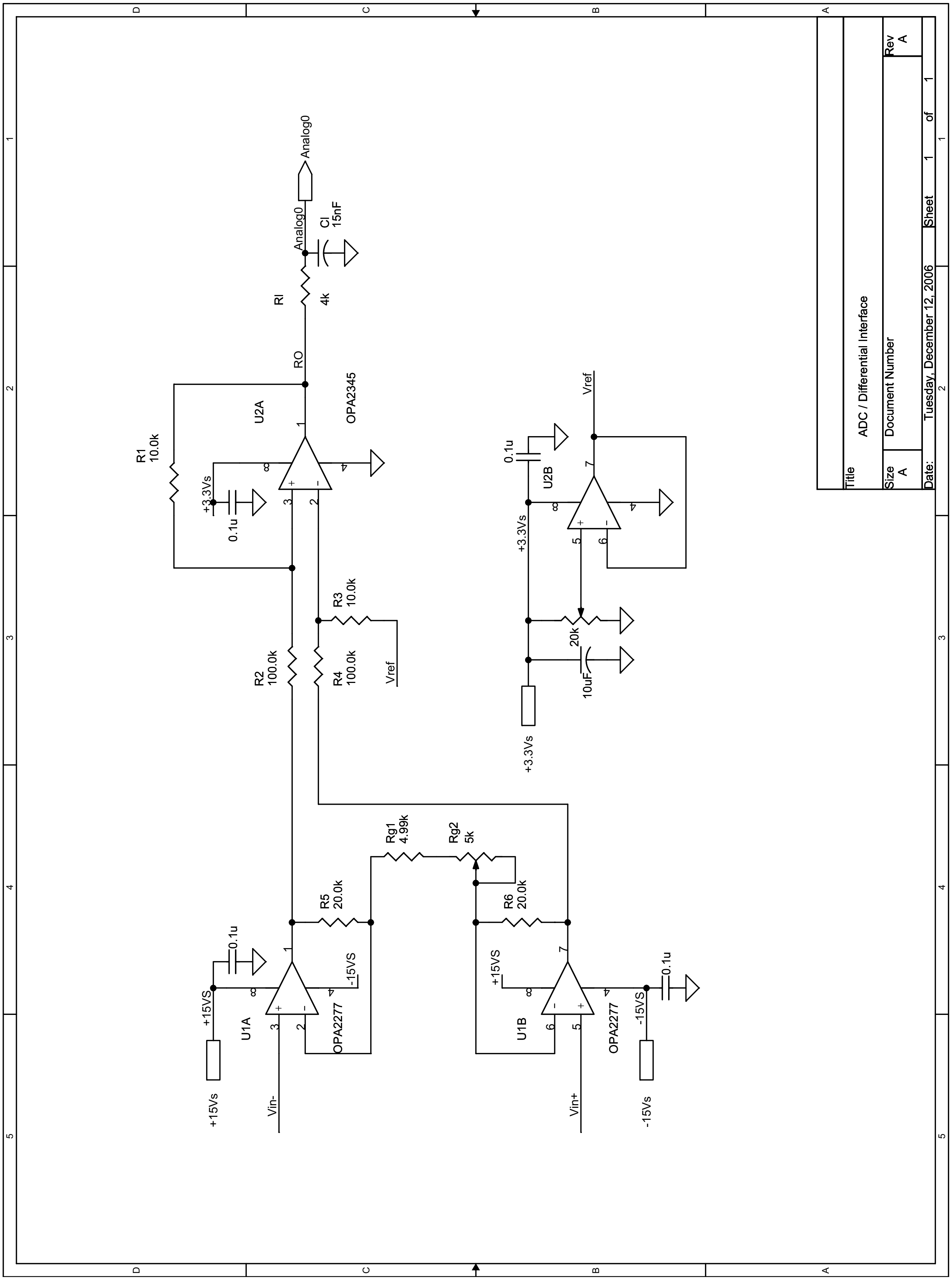


Figure 6-16: ADC, differential interface

6.13 User I/O Space

In the following schematic (Figure 6-17), the User I/O space is used to control four LEDs. The MC55000 User I/O accessible address space located from address 1000h to 10FFh on the external bus. The Addr12 bit is reserved in order to serve as a chip select, while the least significant eight bits serve as the active address. In this example, it is assumed that the active I/O space consists of eight registers; using only the three least significant bits of the address word. Additionally, the register to control the LEDs is assumed to reside at address offset 0x7h.

The simplest way to add an LED is to connect it to a high sink/source current port. In this example, a 74AC377 is used to buffer the data and to drive the LEDs. The MV8141 super bright red LEDs have been used, with 1.5V/1.7V/2.4V minimum/typical/maximum forward voltages, respectively, at 20 mA. R = 150Ω ensures that the output current doesn't exceed 20 mA; with a typical current of 8 mA (assuming a supply of V_{CC} = 3.3V±5%). It also ensures that a minimum of 2 mA current will flow through the LED. Note that the variation in the actual current is relatively large and will result in large variations in the luminance.

The 74AC377 was chosen due to the simplicity with which it interfaces to the MC55000. The 74AC377 is not equipped with a master reset and therefore on power-up the LEDs may be in an arbitrary state until the first valid write is received. The master reset version, the 74AC273, may be used providing that logic for generating the clock signal to it is added. The 74ACTQ823 may also be used with 5V supplies and features both clock enable and master reset.

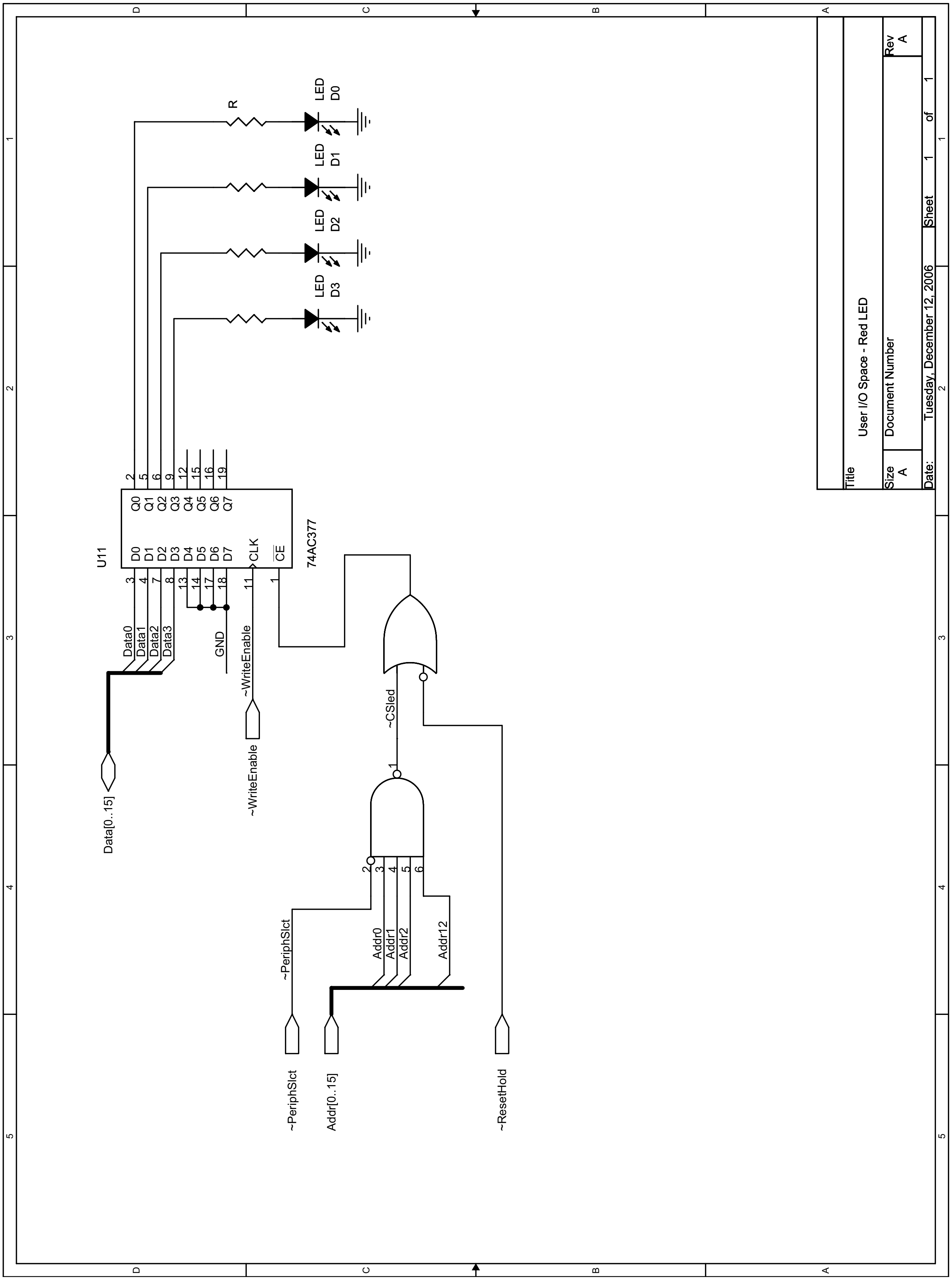


Figure 6-17: User I/O space, red LED

6.14 Parallel Communication Interface

Parallel communication supports the highest throughput of any of the communication interfaces. It is most often used when the host processor and MC55000 reside on the same circuit board. In the dual chip configuration (MC55x20) the IO chip provides the parallel interface to the host and care should be taken to ensure that the timing requirements specified in chapters 3 and 4 are observed. In the single chip configuration (MC55x10) if no external logic is providing the parallel interface the *ParallelEnable* input pin should be tied low to indicate that parallel communication is disabled.

In sections 6.14.1 and 6.14.2, the 16/16 (16-bit bus, 16-bit data) and 8/16 (8-bit bus, 16-bit data) host interfaces are demonstrated.

6.14.1 16/16 Host Interface

In this example, the IO chip is interfaced to the Motorola MCF5282 ColdFire microprocessor, as shown in Figure 6-18. The MCF5282's external interface module is used for the interface, which includes data and address buses as well as additional control signals such as *R/~W*, *~TS*, *~OE* and *~CS*. For a detailed description of this device's functionality and timing specifications, refer to the MCF5282 data sheet. The following design notes focus on the interface between the MCF5282 and the IO chip.

6.14.1.1 Write Cycle

The host should be able to generate valid data at less than T_{15} (refer to Section 3.3, "AC Characteristics") from the latest falling edge of either *R/~W* or *~CS* signals. Since both *R/~W* and *~CS* become active as much as half of the MCF5282's bus clock cycle prior to the data, the *~TS* (Transfer Start) signal is ANDed with the inverted *R/~W* signal. For the write cycle this achieves a one bus clock cycle delay in the incoming *IR/~W* signal to the IO chip. In this manner, a maximum of 10 nsec is guaranteed between the falling edge of *IR/~W* and valid data.

6.14.1.2 Read Cycle and Wait States

The host should add wait states in order to meet the timing requirements of the IO chip. The following formula may be used in order to calculate the number of required wait states as a function of the host's bus clock period, t_{CYC}.

$$N_{WS} = \lceil 70/t_{CYC} \rceil - 1$$

The operator $\lceil \cdot \rceil$ indicates rounding towards the largest integer. N_{WS} is the number of wait states that are required. The selected MC5282 has a 66 MHz input clock resulting in a t_{CYC} of ~15 nsec and thus four wait states are required.

6.14.1.3 Other Control Signals

In the example, address bit 3 is used for signaling to the IO chip whether a command or data word is being written. For a read cycle, this bit may be used for requesting either data or the status word.

HostRdy is used to interrupt the MCF5282. This can be a low priority interrupt used to invoke a communication ISR in the MCF5282. According to conditions that are programmable by the host the MC55000 can also activate the *~HostInterrupt* signal.

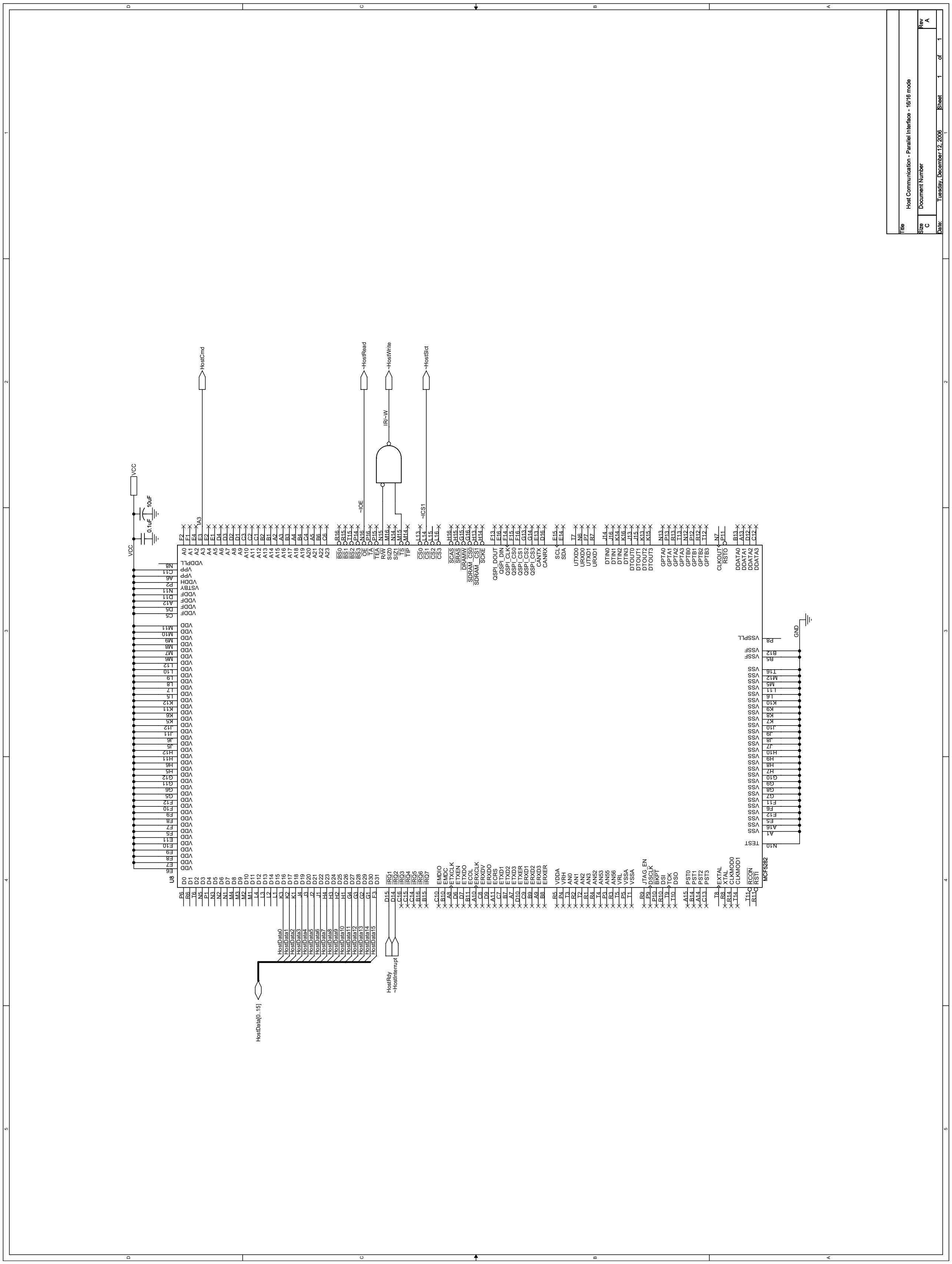


Figure 6-18: Host communication, parallel interface, 16/16 mode

6.14.2 8/16 Host Interface

In this example, a PIC microcontroller with limited I/O pins is interfaced with the IO chip. The minimum number of I/O pins required for the parallel 8/16 communications mode is eight bi-directional pins for the data, and five additional pins for the control signals (four outputs and one input). These signals are shown in Figure 6-19.

The PIC16F648 features:

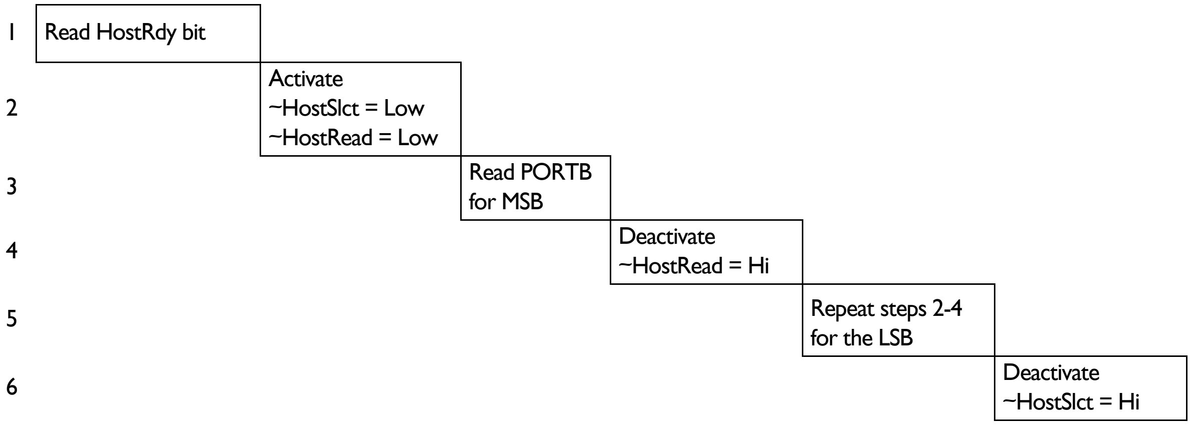
- An 8-bit microcontroller core
- Up to 10 MHz clock at 3.3V voltage supply
- 16 I/O pins divided into two ports, with a high impedance state and weak internal pull-up.

The eight I/O pins of the PORTB are used to connect to the IO chip's ***HostData*** bus (pins 0 through 7). The five pins of the PORTA are reserved for the control signals, as shown in the following table.

IO Chip Signal	PIC16F648 Signal	Dir	Comments
HostData0-7	PORTB RB0-RB7	I/O	Should be pulled up (Bit 7 of the OPTION register). Should be kept in high impedance state unless written to.
~HostSlct	PORTA RA0	O	
HostCmd	PORTA RA1	O	
~HostWrite	PORTA RA2	O	
~HostRead	PORTA RA3	O	
HostRdy	PORTA RA6	I	

Shown below is a typical sequence of events for performing the host read and write cycle.

6.14.2.1 Typical Data / Status Read Sequence



1 Read HostRdy bit

2 Activate ~HostSlct = Low ~HostRead = Low

3 Read PORTB for MSB

4 Deactivate ~HostRead = Hi

5 Repeat steps 2-4 for the LSB

6 Deactivate ~HostSlct = Hi

Where:

Step 1: May be performed using the BTFSS instruction. This step must loop until ***HostRdy*** is high.

Step 2: Write a byte into PORTA with the lowest four bits set to either 0x4 (data) or 0x6 (status) using the MOVLW and MOVWF instructions.

Step 3: Read Port B and store the result.

Step 4: Write a byte into PORTA with the lowest four bits set to either 0xC or 0xE to deactivate ~HostRead.

Step 5: Repeat steps 2 - 4 for the LSB.

Step 6: Write a byte into PORTA with the lowest four bits set to 0xF to deactivate ~HostSlct.

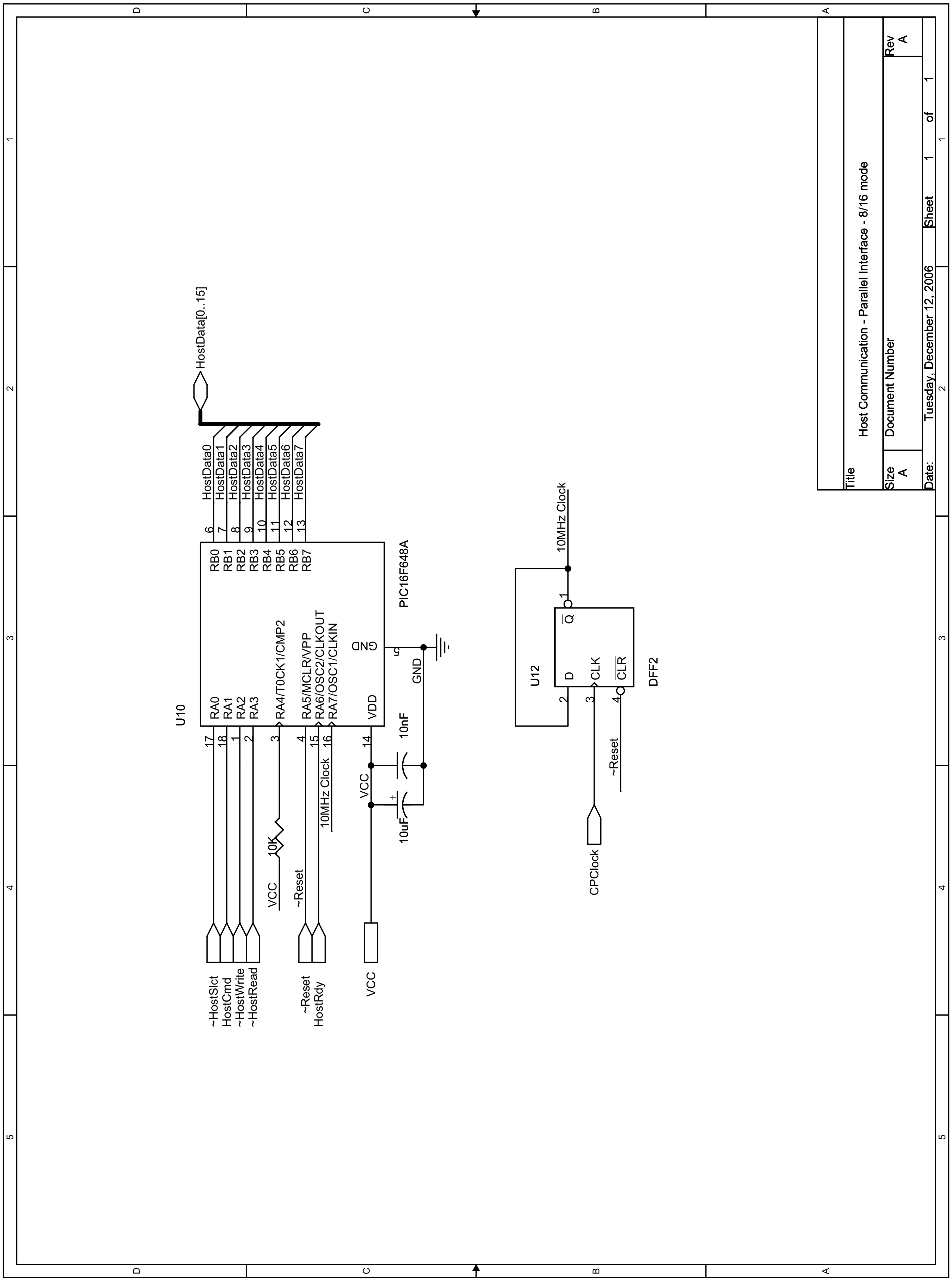


Figure 6-19: Host communication, parallel interface, 8/16 mode

6.14.2.2 Typical Data/Instruction Write Sequence

1	Read HostRdy bit
2	Enable PORTB outputs
3	Write MSB to PORTB
4	Activate ~HostSlct = Low ~HostWrite = Low
5	Deactivate ~HostWrite = Hi
6	Repeat steps 3-5 for the LSB
7	Deactivate ~HostSlct = Hi
8	Disable Port B outputs

Where:

Step 1: May be performed using the BTFSS instruction. This step must loop until *HostRdy* is high.

Step 2: Enable PORTB outputs by writing the all-zero word to the TRISB register.

Step 3: Write the MSB into PORTB.

Step 4: Write a byte into PORTA, with the lowest 4 bits set to either 0x8(data) or 0xA(instruction) using the MOVLW and MOVWF instructions.

Step 5: Write a byte into PORTA, with the lowest 4 bits set to either 0xC or 0xE to deactivate ~HostWrite.

Step 6: Repeat steps 3 - 5 for the LSB.

Step 7: Write a byte into PORTA with the lowest four bits set to 0xF to deactivate ~HostSlct.

Step 8: Disable PORT B outputs by writing 0xFF to the TRISB register.

6.14.2.3 **Note:** Since each step takes one or multiple execution cycles (400 nsec) there are no practical timing constraints. For the write cycle, the data should be present on the data bus prior to enabling the write operation by setting *~HostWrite* to active low.

6.15 Overcurrent and Emergency Braking Circuits for Motor Drivers

Most of the drivers demonstrated in this manual, either full or half bridges, are used with a sense power resistor through which the winding current flows. This results in a voltage drop, which is then sampled by the overcurrent circuitry. Due to switching transients in the driver, the current through the sense resistor is prone to spikes. The following schematic (Figure 6-20) shows protection circuitry based on the voltage developed over *Rsense*, *Vsense1* and *Vsense2*. This circuitry should serve as a protection device and as such in normal operation the circuitry should not reach its threshold voltage. As a protective device, the response time should be determined according to the application requirements. The response time for this circuitry is set to 2 µsec.

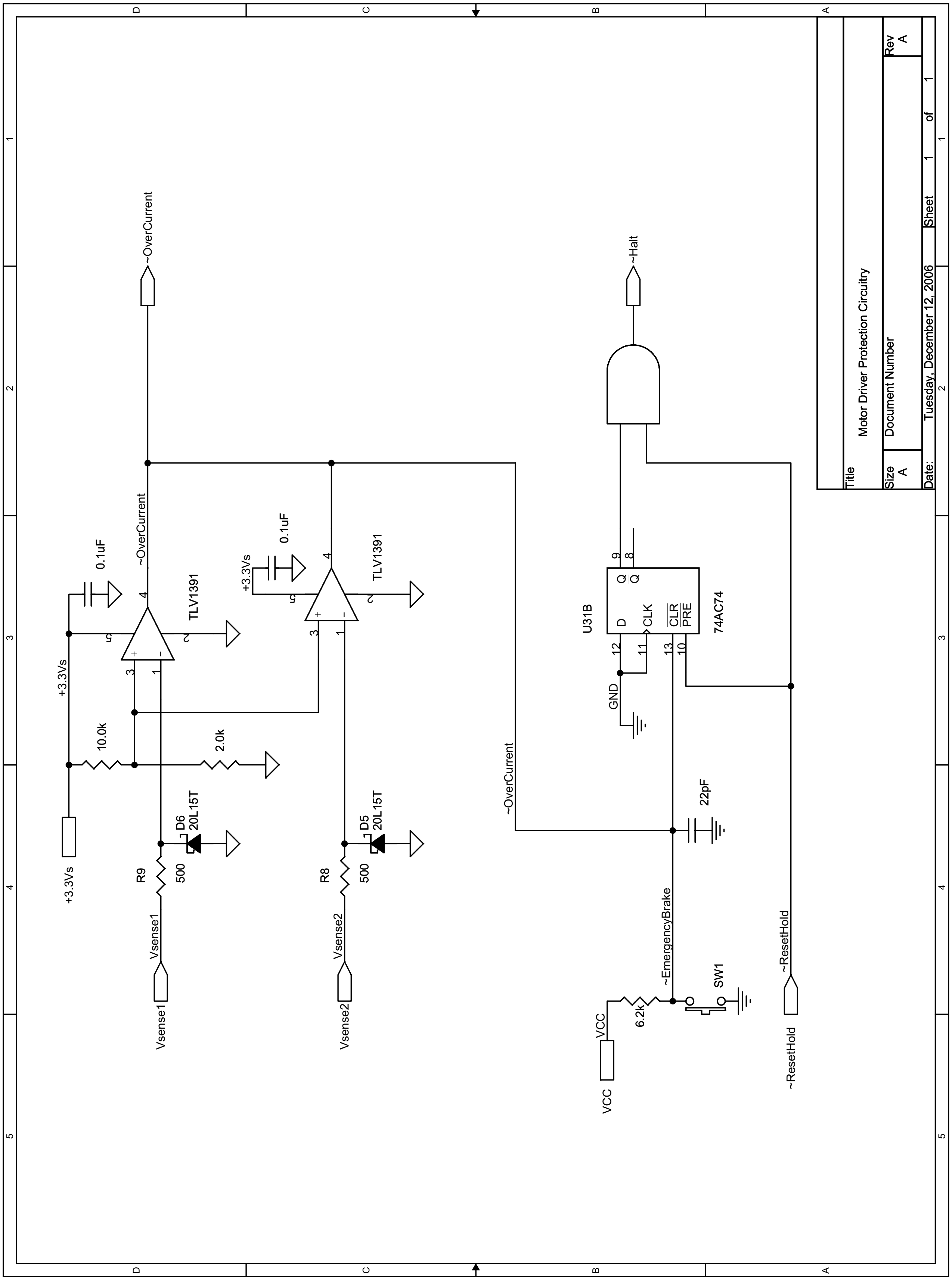


Figure 6-20: Motor driver protection circuitry

The comparator is a TLV1391, which is a single supply, fast response, open collector output. The TLV1391 tolerates an input range as low as –0.3V. Because the sense voltage may fall below this range, a protection diode is added, with a V_F = 0.25V. In order to avoid false over-current detection, the sense signal is low pass filtered with a cut off frequency in the MHz region. This is achieved with the use of a 500Ω resistor and the capacitance of the Schottky diode. The nominal reference voltage of the comparator is set to 0.55V, which is selected to be 80% higher than the nominal voltage drop over *Rsense* at the rated current of the motor windings (0.3V). The typical response time of the TLV1391 is 800 nsec; leaving ~1 µsec response time for the driver itself. Additional input branches may be added to the *~OverCurrent* circuitry. In this case, the value of the pull-up resistor should be re-calculated. The resultant *~OverCurrent* signal may either be used for the momentary disabling of the motor's driver, or to halt it completely until the next *~Reset* signal is generated.

An external switch, ~EmergencyBrake, is used as an additional method for halting the motor. Note that the U31B D-FF may momentarily have both Clear and Preset inputs active low.

6.16 Using the Allegro A3977 to Drive Microstepping Motors

The A3977 is a complete microstepping motor driver with a built-in translator. The translator is capable of driving bi-polar stepper motors in full-, half-, quad-, and eighth-step modes. When the step input transitions from low to high, the A3977 will advance the motor one full-, half-, quad-, or eighth-step according to the configuration of the MS1 and MS2 pins. In the example the driver is configured for eighth-step resolution.

The A3977 operation can be tuned with the use of external components. *CT* is used to determine the blanking period of the current sense comparator circuitry. The product of *RT* and *CT* is used to determine the PWM constant off period. *R1* and *R2*, along with *RT* and *CT*, determine the percentage of the fast decay in mixed decay mode. The sense resistors, *Rsense*, should be selected according to the maximum current and voltage restrictions of the driver. Refer to the device data sheet for further information.

For a direct interface of the pulse signal to the step input, the polarity of the pulse signal must be inverted using the **SetSignalSense** command. This is required because the A3977 recognizes a step during a low-to-high transition of the step input signal, whereas the non-inverted behavior of the MC55000 is to generate a step on a high-to-low transition. The pulse and direction outputs will satisfy the A3977 timing requirements if operated at a step rate of 155.625 kHz or less. This can be set using the **SetStepRange** command. The MC55110 has a maximum step rate of 97.6 kHz.

The schematic in Figure 6-21 uses the sense outputs to detect a malfunction by sensing the current through the motor windings. To generate the *~Halt* signal, the over-current circuitry should be configured with an *Rsense* = 0.15Ω power resistor for a rated 2A motor.

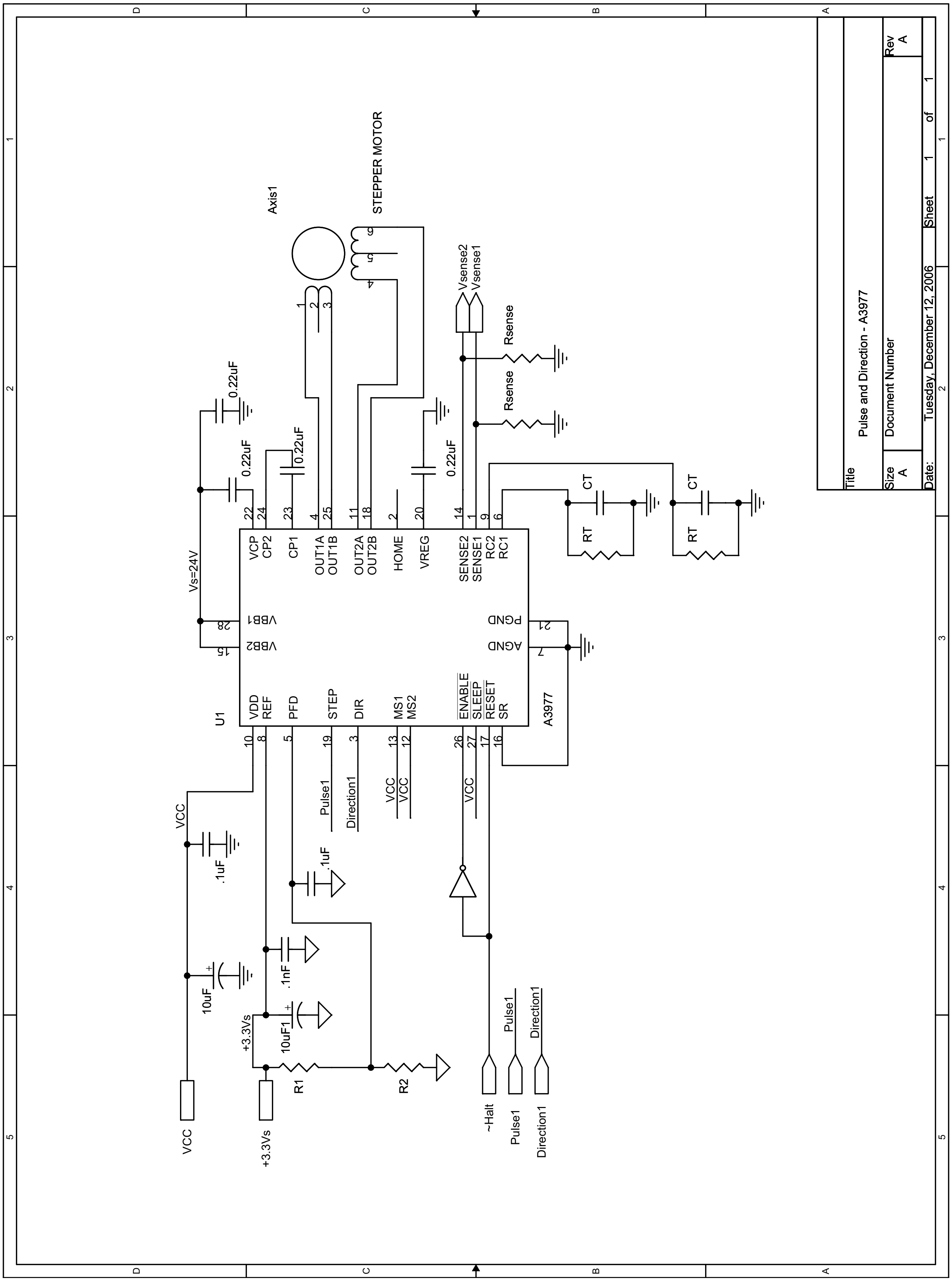


Figure 6-21: Pulse and directon, A3977

This page intentionally left blank.

For additional information, or for technical assistance, please contact PMD at (781) 674-9860.

You may also e-mail your request to support@pmdcorp.com

Visit our website at http://www.pmdcorp.com

Performance Motion Devices
55 Old Bedford Rd.
Lincoln, MA 01773