COMPREHENSIVE SERVICES

We offer competitive repair and calibration services, as well as easily accessible documentation and free downloadable resources.

SELL YOUR SURPLUS

We buy new, used, decommissioned, and surplus parts from every NI series. We work out the best solution to suit your individual needs.

Sell For Cash Get Credit Receive a Trade-In Deal

OBSOLETE NI HARDWARE IN STOCK & READY TO SHIP

We stock **New, New Surplus, Refurbished,** and **Reconditioned** NI Hardware.

APEX WAVES

Bridging the gap *between the manufacturer and your legacy test system.*

1-800-915-6216

www.apexwaves.com

sales@apexwaves.com

All trademarks, brands, and brand names are the property of their respective owners.

Request a Quote CLICK HERE *DAQCard-1200*

DAQCard™-1200
Register-Level Programmer Manual

Multifunction I/O Card for the PCMCIA Bus

October 1995 Edition

Part Number 340988A-01

© Copyright 1995 National Instruments Corporation.
All Rights Reserved.

National Instruments Corporate Headquarters
6504 Bridge Point Parkway
Austin, TX 78730-5039
(512) 794-0100
Technical support fax: (800) 328-2203
(512) 794-5678

Branch Offices:
Australia 03 9 879 9422, Austria 0662 45 79 90 0, Belgium 02 757 00 20, Canada (Ontario) 519 622 9310,
Canada (Québec) 514 694 8521, Denmark 45 76 26 00, Finland 90 527 2321, France 1 48 14 24 24,
Germany 089 741 31 30, Hong Kong 2645 3186, Italy 02 48301892, Japan 03 5472 2970, Korea 02 596 7456,
Mexico 95 800 010 0793, Netherlands 0348 433466, Norway 32 84 84 00, Singapore 2265886, Spain 91 640 0085,
Sweden 08 730 49 70, Switzerland 056 200 51 51, Taiwan 02 377 1200, U.K. 01635 523545

Limited Warranty

The DAQCard-1200 is warranted against defects in materials and workmanship for a period of one year from the date of shipment, as evidenced by receipts or other documentation. National Instruments will, at its option, repair or replace equipment that proves to be defective during the warranty period. This warranty includes parts and labor.

The media on which you receive National Instruments software are warranted not to fail to execute programming instructions, due to defects in materials and workmanship, for a period of 90 days from date of shipment, as evidenced by receipts or other documentation. National Instruments will, at its option, repair or replace software media that do not execute programming instructions if National Instruments receives notice of such defects during the warranty period. National Instruments does not warrant that the operation of the software shall be uninterrupted or error free.

A Return Material Authorization (RMA) number must be obtained from the factory and clearly marked on the outside of the package before any equipment will be accepted for warranty work. National Instruments will pay the shipping costs of returning to the owner parts which are covered by warranty.

National Instruments believes that the information in this manual is accurate. The document has been carefully reviewed for technical accuracy. In the event that technical or typographical errors exist, National Instruments reserves the right to make changes to subsequent editions of this document without prior notice to holders of this edition. The reader should consult National Instruments if errors are suspected. In no event shall National Instruments be liable for any damages arising out of or related to this document or the information contained in it.

EXCEPT AS SPECIFIED HEREIN, NATIONAL INSTRUMENTS MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AND SPECIFICALLY DISCLAIMS ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. CUSTOMER'S RIGHT TO RECOVER DAMAGES CAUSED BY FAULT OR NEGLIGENCE ON THE PART OF NATIONAL INSTRUMENTS SHALL BE LIMITED TO THE AMOUNT THERETOFORE PAID BY THE CUSTOMER. NATIONAL INSTRUMENTS WILL NOT BE LIABLE FOR DAMAGES RESULTING FROM LOSS OF DATA, PROFITS, USE OF PRODUCTS, OR INCIDENTAL OR CONSEQUENTIAL DAMAGES, EVEN IF ADVISED OF THE POSSIBILITY THEREOF. This limitation of the liability of National Instruments will apply regardless of the form of action, whether in contract or tort, including negligence. Any action against National Instruments must be brought within one year after the cause of action accrues. National Instruments shall not be liable for any delay in performance due to causes beyond its reasonable control. The warranty provided herein does not cover damages, defects, malfunctions, or service failures caused by owner's failure to follow the National Instruments installation, operation, or maintenance instructions; owner's modification of the product; owner's abuse, misuse, or negligent acts; and power failure or surges, fire, flood, accident, actions of third parties, or other events outside reasonable control.

Copyright

Under the copyright laws, this publication may not be reproduced or transmitted in any form, electronic or mechanical, including photocopying, recording, storing in an information retrieval system, or translating, in whole or in part, without the prior written consent of National Instruments Corporation.

Trademarks

LabVIEW®, NI-DAQ®, RTSI®, and DAQCard™ are trademarks of National Instruments Corporation.

Product and company names listed are trademarks or trade names of their respective companies.

WARNING REGARDING MEDICAL AND CLINICAL USE OF NATIONAL INSTRUMENTS PRODUCTS

National Instruments products are not designed with components and testing intended to ensure a level of reliability suitable for use in treatment and diagnosis of humans. Applications of National Instruments products involving medical or clinical treatment can create a potential for accidental injury caused by product failure, or by errors on the part of the user or application designer. Any use or application of National Instruments products for or involving medical or clinical treatment must be performed by properly trained and qualified medical personnel, and all traditional medical safeguards, equipment, and procedures that are appropriate in the particular situation to prevent serious injury or death should always continue to be used when National Instruments products are being used. National Instruments products are NOT intended to be a substitute for any form of established process, procedure, or equipment used to monitor or safeguard human health and safety in medical or clinical treatment.

Contents

About This Manual ix
Organization of This Manual ix
Conventions Used in This Manual x
National Instruments Documentation x
Related Documentation xi
Customer Communication xi

Chapter 1
General Description 1-1
General Characteristics 1-1
Board Configuration Overview 1-2
Analog Input Configuration 1-2
Analog Output Configuration 1-2
Digital I/O Configuration 1-3
Counter Configuration 1-3

Chapter 2
Register Map and Descriptions 2-1
Register Map 2-1
Register Description Overview 2-3
Configuration and Status Register Group 2-4
Command Register 1 2-5
Command Register 2 2-7
Command Register 3 2-9
Command Register 4 2-10
Command Register 5 2-12
Command Register 6 2-13
Status Register 1 2-15
Status Register 2 2-16
Analog Input Register Group 2-17
A/D FIFO Register 2-18
A/D Clear Register 2-20
Start Convert Register 2-20
Analog Output Register Group 2-21
DAC0 Low-Byte, DAC0 High-Byte, DAC1 Low-Byte, and DAC1 High-Byte Register 2-22
82C53 Counter/Timer Register Groups A and B 2-23
Counter A0 Data Register 2-24
Counter A1 Data Register 2-24
Counter A2 Data Register 2-25
Counter A Mode Register 2-25
Timer Interrupt Clear Register 2-26
Counter B0 Data Register 2-26
Counter B1 Data Register 2-27
Counter B2 Data Register 2-27

Counter B Mode Register 2-28
82C55A Digital I/O Register Group 2-29
Port A Register 2-30
Port B Register 2-30
Port C Register 2-31
Digital Control Register 2-31
Interval Counter Register Group 2-32
Interval Counter Data Register 2-33
Interval Counter Strobe Register 2-33

Chapter 3
Programming 3-1
PCMCIA Card Initialization 3-1
Register Programming Considerations 3-2
Initializing the DAQCard-1200 Circuitry 3-2
Programming the Analog Input Circuitry for Single A/D Conversions 3-4
Clearing the Analog Input Circuitry 3-4
Configuring the Analog Input Circuitry 3-5
Performing Single A/D Conversions 3-6
Programming a DAQ Operation Using Internal Timing 3-6
Programming Counter A0 and Counter B0 3-8
Programming Counter A1 3-9
Programming Counter B1 and the Interval Counter Register 3-9
Triggering the DAQ Operation 3-10
Servicing the DAQ Operation 3-10
Programming a DAQ Operation Using External Timing 3-11
Programming a DAQ Operation Using EXTCONV* 3-11
Programming a DAQ Operation Using EXTTRIG in Posttrigger Mode 3-11
Programming a DAQ Operation Using EXTTRIG in Pretrigger Mode 3-12
Programming a DAQ Operation Using OUTB1 3-12
DAQ Interrupt Programming 3-13
Programming the Analog Output Circuitry 3-14
Configuring the Analog Output Circuitry 3-14
Programming the Update Mode of the Analog Output Circuitry 3-15
DAC Interrupt Programming 3-16
Programming the Digital I/O Circuitry 3-17
Programming the General-Purpose Counter/Timers 3-17

Chapter 4
Calibration 4-1
Storing User-Defined Constants 4-1
Calibration DACs 4-2
Analog Input Calibration 4-3
Bipolar Input Calibration Procedure 4-4
Pregain Offset Coarse Calibration 4-4
Pregain Offset Fine Calibration 4-4
Gain Calibration 4-5
Postgain Offset Calibration 4-5

Calibration at Higher Gains 4-5
Unipolar Input Calibration Procedure....4-5
Pregain Offset Calibration 4-6
Gain Calibration....4-6
Postgain Offset Calibration....4-6
Analog Output Calibration....4-6
Bipolar Output Calibration Procedure 4-7
Gain Calibration....4-7
Offset Calibration....4-8
Unipolar Output Calibration Procedure 4-8
Gain Calibration....4-8
Offset Calibration....4-8
EEPROM Map 4-9

Appendix A
Card Information Structure (CIS) A-1

Appendix B
Fujitsu MB88341/MB88342 Data Sheet B-1

Appendix C
Xicor X25020 Data Sheet C-1

Appendix D
Oki MSM82C53 Data Sheet D-1

Appendix E
MSM82C55A Data Sheet E-1

Appendix F
Customer Communication F-1

Glossary G-1

Index Index-1

Tables

Table 1-1. Analog Input Settings....1-2
Table 1-2. Analog Output Settings....1-2

Table 2-1. DAQCard-1200 Register Map....2-2

Table 3-1. Analog Output Voltage Versus Digital Code (Unipolar Mode, Straight Binary Coding)....3-15
Table 3-2. Analog Output Voltage Versus Digital Code (Bipolar Mode, Two's Complement Coding)....3-15

Table 4-1. Calibration DAC Characteristics for Analog Input Circuitry....4-2
Table 4-2. Calibration DAC Characteristics for Analog Output Circuitry....4-3
Table 4-3. DAQCard-1200 EEPROM Map....4-9

About This Manual

This manual contains information about the internal operation and programming of the DAQCard-1200. The DAQCard-1200 is a low-cost, low-power analog input, analog output, digital, and timing PCMCIA card for computers equipped with a Type II PCMCIA slot.

This manual assumes you are familiar with the *DAQCard-1200 User Manual*. If you will be using National Instruments software with the DAQCard-1200, you do not need to read this manual. For information on the DAQCard-1200 installation, signal connections, and theory of operation, consult your user manual.

Organization of This Manual

The *DAQCard-1200 Register-Level Programmer Manual* is organized as follows:

- Chapter 1, *General Description*, describes the general characteristics and gives a configuration overview of the DAQCard-1200.
- Chapter 2, *Register Map and Descriptions*, describes in detail the address and function of each of the DAQCard-1200 registers.
- Chapter 3, *Programming*, contains programming instructions for operating the DAQCard-1200 circuitry.
- Chapter 4, *Calibration*, contains instructions for creating user-defined calibration constants for the DAQCard-1200 CALDACs.
- Appendix A, *Card Information Structure (CIS)*, lists the CIS for the DAQCard-1200.
- Appendix B, *Fujitsu MB88341/MB88342 Data Sheet*, contains a manufacturer data sheet for the MB88341 CALDACs (Fujitsu). These CALDACs are used on the DAQCard-1200.
- Appendix C, *Xicor X25020 Data Sheet*, contains a manufacturer data sheet for the X25020 SPI Serial EEPROM (Xicor). This EEPROM is used on the DAQCard-1200.
- Appendix D, *Oki MSM82C53 Data Sheet*, contains a manufacturer data sheet for the MSM82C53 counter/timer (OKI Semiconductor). This counter/timer is used on the DAQCard-1200.
- Appendix E, *Oki MSM82C55A Data Sheet*, contains a manufacturer data sheet for the MSM82C55A programmable peripheral interface (PPI) (OKI Semiconductor). This PPI is used on the DAQCard-1200.
- Appendix F, *Customer Communication*, has a form you can use to comment on the product documentation. This appendix also contains information on how to access technical assistance for your National Instruments product.

- The *Glossary* contains an alphabetical list and description of terms used in this manual, including abbreviations, acronyms, metric prefixes, mnemonics, and symbols.
- The *Index* contains an alphabetical list of key terms and topics covered in this manual, including the page where you can find each one.

Conventions Used in This Manual

The following conventions are used in this manual:

< >	Angle brackets containing numbers separated by an ellipses represent a range of values associated with a bit, signal, or port (for example, ACH<0..7> stands for ACH0 through ACH7).
bold	Bold text denotes menus, menu items, or dialog box buttons or options.
bold italic	Bold italic text denotes a note, caution, or warning.
italic	Italic text denotes emphasis, a cross reference, or an introduction to a key concept.
`monospace`	Text in this font denotes text or characters that are to be literally input from the keyboard, sections of code, programming examples, and syntax examples. This font is also used for the proper names of disk drives, paths, directories, programs, subprograms, subroutines, device names, functions, variables, filenames, and extensions, and for statements and comments taken from program code.
NI-DAQ	NI-DAQ refers to the NI-DAQ software for PC or Macintosh unless otherwise stated.

Abbreviations, acronyms, metric prefixes, mnemonics, and terms are listed in the *Glossary*.

National Instruments Documentation

The *DAQCard-1200 Register-Level Programmer Manual* is one piece of the documentation set for your data acquisition (DAQ) system. You could have any of several types of manuals, depending on the hardware and software in your system. Use the different types of manuals you have as follows:

- *Getting Started with SCXI*—If you are using SCXI, this is the first manual you should read. It gives an overview of the SCXI system and contains the most commonly needed information for the modules, chassis, and software.
- Your SCXI hardware user manuals—If you are using SCXI, read these manuals next for detailed information about signal connections and module configuration. They also explain in greater detail how the module works and contain application hints.
- Your DAQ hardware user manuals—These manuals have detailed information about the DAQ hardware that plugs into or is connected to your computer. Use these manuals for

hardware installation and configuration instructions, specification information about your DAQ hardware, and application hints.

- Software manuals—Examples of software manuals you may have are the LabVIEW and LabWindows®/CVI manual sets and the NI-DAQ manuals. After you set up your hardware system, use either the application software (LabVIEW or LabWindows/CVI) manuals or the NI-DAQ manuals to help you write your application. If you have a large and complicated system, it is worthwhile to look through the software manuals before you configure your hardware.
- Accessory installation guides or manuals—If you are using accessory products, read the terminal block and cable assembly installation guides or accessory board user manuals. They explain how to physically connect the relevant pieces of the system. Consult these guides when you are making your connections.
- SCXI chassis manuals—If you are using SCXI, read these manuals for maintenance information on the chassis and installation instructions.

Related Documentation

As you read this manual, you may find it helpful to refer to the following documents:

- Your *DAQCard-1200 User Manual*
- Your personal computer technical reference manual
- *PC Card Standard, Release 2.1* or later, Personal Computer Memory Card International Association (PCMCIA)
- *Card Services Specifications, Release 2.1* or later, Personal Computer Memory Card International Association (PCMCIA)
- *Socket Services Specifications*, *Release* 2.1 or later, Personal Computer Memory Card International Association (PCMCIA)

Customer Communication

National Instruments wants to receive your comments on our products and manuals. We are interested in the applications you develop with our products, and we want to help if you have problems with them. To make it easy for you to contact us, this manual contains a comment form for you to complete. This form is in Appendix F, *Customer Communication*, at the end of this manual.

Chapter 1
General Description

This chapter describes the general characteristics and gives a configuration overview of the DAQCard-1200.

General Characteristics

Thank you for your purchase of the National Instruments DAQCard-1200. The DAQCard-1200 is a low-cost, low-power analog input, output, digital, and timing I/O board for PCs equipped with a type II PCMCIA slot. The board contains a 12-bit, successive-approximation ADC with eight input channels, which you can configure as eight single-ended, eight pseudodifferential, or four differential. The DAQCard-1200 also has two 12-bit DACs with voltage outputs, 24 lines of TTL-compatible digital I/O, and three 16-bit counter/timers for timing I/O. The DAQCard-1200 is fully software configurable and calibrated so that you can easily install the card and begin your acquisition without having to spend time configuring or calibrating the board.

The small size and weight of the DAQCard-1200 coupled with its low-power consumption makes this board ideal for use in portable computers, making portable data acquisition practical. The board requires very little power when operating and has a power-down mode that uses even less power, thus conserving the life of your computer batteries.

In addition, the low cost of a system based on the DAQCard-1200 makes it ideal for laboratory work in industrial and academic environments. The multichannel analog input is useful in signal analysis and data logging. The 12-bit ADC is useful in high-resolution applications such as chromatography, temperature measurement, and DC voltage measurement. You can use the analog output channels to generate experimental stimuli, for machine and process control, and for analog function generation. You can use the TTL-compatible digital I/O line for switching external devices such as transistors and solid-state relays, for reading the status of external digital logic, and for generating interrupts. You can use the counter/timers to synchronize events, generate pulses, and measure frequency and time. The DAQCard-1200, used in conjunction with the PC, is a versatile, cost-effective platform for laboratory test, measurement, and control.

The DAQCard-1200 ships with NI-DAQ, the National Instruments complete DAQ driver that handles every function listed on the DAQ hardware data sheet. Using NI-DAQ you can quickly and easily start your application without having to program the card at the register level.

This manual is intended for programming at the register level. Even if you are an experienced register-level programmer, consider using NI-DAQ, LabVIEW, or LabWindows/CVI to program the DAQCard-1200. If NI-DAQ does not support your operating system, or you have other reasons to write your own register-level programs, continue reading this manual.

Board Configuration Overview

This section is a reference to the DAQCard-1200 configuration options. You should already have unpacked and installed your DAQCard-1200. Refer to your *DAQCard-1200 User Manual* if you have not already performed these tasks.

Analog Input Configuration

The DAQCard-1200 is completely software configurable, and at startup, defaults to the following configuration:

- Referenced single-ended input mode
- ±5 V analog input range

Table 1-1 lists the available analog I/O configurations for the DAQCard-1200 and shows the default settings.

Table 1-1. Analog Input Settings

Parameter	Configuration
Analog Input Range	Bipolar—±5 V (default setting) Unipolar—0–10 V
Analog Input Mode	Referenced single-ended (RSE) (default setting) Nonreferenced single-ended (NRSE) Differential (DIFF)

The analog input circuit is software configurable.

Analog Output Configuration

At startup, the two channels of analog output of the DAQCard-1200 default to the following configuration:

- ±5 V analog input range

Table 1-2 lists the available analog I/O configurations for the DAQCard-1200 and shows the default settings.

Table 1-2. Analog Output Settings

Parameter	Configuration
Analog Output Range	Bipolar—±5 V (default setting) Unipolar—0–10 V

The analog output circuit is software configurable.

Digital I/O Configuration

The DAQCard-1200 uses the MSM82C55 PPI, which provides 24 digital lines in the form of three ports—A, B, and C. On power up, all three ports reset to mode 0 input. Appendix E, *OKI MSM82C55 Data Sheet*, has the 82C55 data sheets that you need to program the digital I/O.

Counter Configuration

You can use the MSM82C53 counter/timers for general-purpose applications, such as pulse and square wave generation, event counting, and pulsewidth, time-lapse, and frequency measurement. Appendix D, *Oki MSM82C53 Data Sheet*, has the 82C53 data sheet that you need to program the counters/timers.

Chapter 2
Register Map and Descriptions

This chapter describes in detail the address and function of each of the DAQCard-1200 registers.

Register Map

Table 2-1 shows the register map for the DAQCard-1200 and lists the register name, address, type (read-only, write-only, or read-write), and size in bits.

Table 2-1 divides the DAQCard-1200 registers into six groups. The Configuration and Status Register Group controls the overall operation of the DAQCard-1200. The Analog Input Register Group reads output from the 12-bit successive-approximation ADC. The Analog Output Register Group accesses the two 12-bit DACs. The two Counter/Timer Register Groups (A and B) access each of the two onboard 82C53 counter/timer integrated circuits. The Digital I/O Register Group consists of the four registers of the onboard 82C55A PPI integrated circuit that are used for digital I/O. The Interval Counter registers are used in the single-channel interval-acquisition mode.

The DAQCard-1200 registers are 8-bit registers. To transfer 16-bit data, you must perform two consecutive I/O readings or writings. For example, to read the 16-bit A/D conversion result, you must make two consecutive 8-bit readings of the FIFO. The first reading returns the low byte of the 16-bit data, and the second returns the high byte of the data.

The register addresses are the offsets from the base address. For example, if the base address is 0x220, Command Register 1 is at address 0x220, Command Register 2 is at address 0x221, and so on. The base address and interrupt channel must be allocated prior to activating the card. For more information, refer to *PCMCIA Card Initialization* in Chapter 3, *Programming*.

Table 2-1. DAQCard-1200 Register Map

Register Name	Address Offset (Hex)	Type	Size
Configuration and Status Register Group			
Command Register 1	00	Write-only	8-bit
Command Register 2	01	Write-only	8-bit
Command Register 3	02	Write-only	8-bit
Command Register 4	0F	Write-only	8-bit
Command Register 5	1C	Write-only	8-bit
Command Register 6	0E	Write-only	8-bit
Status Register 1	00	Read-only	8-bit
Status Register 2	1D	Read-only	8-bit
Analog Input Register Group			
A/D FIFO Register	0A	Read-only	8-bit
A/D Clear Register	08	Write-only	8-bit
Start Convert Register	03	Write-only	8-bit
Analog Output Register Group			
DAC0 Low-Byte Register	04	Write-only	8-bit
DAC0 High-Byte Register	05	Write-only	8-bit
DAC1 Low-Byte Register	06	Write-only	8-bit
DAC1 High-Byte Register	07	Write-only	8-bit
82C53 Counter/Timer Register Group A			
Counter A0 Data Register	14	Read-Write	8-bit
Counter A1 Data Register	15	Read-Write	8-bit
Counter A2 Data Register	16	Read-Write	8-bit
Counter A Mode Register	17	Write-only	8-bit
Timer Interrupt Clear Register	0C	Write-only	8-bit
82C53 Counter/Timer Register Group B			
Counter B0 Data Register	18	Read-Write	8-bit
Counter B1 Data Register	19	Read-Write	8-bit
Counter B2 Data Register	1A	Read-Write	8-bit
Counter B Mode Register	1B	Write-only	8-bit
82C55A Digital I/O Register Group			
Port A Register	10	Read-Write	8-bit
Port B Register	11	Read-Write	8-bit
Port C Register	12	Read-Write	8-bit
Digital Control Register	13	Write-only	8-bit
Interval Counter Register Group			
Interval Counter Data Register	1E	Write-only	8-bit
Interval Counter Strobe Register	1F	Write-only	8-bit

Register Description Overview

The remainder of this chapter discusses each of the DAQCard-1200 registers in the order shown in Table 2-1. Each register group is introduced, followed by a detailed bit description of each register on the DAQCard-1200. For a detailed bit description of each register concerning the 82C53 (A or B) chip or the 82C55A chip on the DAQCard-1200, refer to Appendix D, *OKI MSM82C53 Data Sheet*, or Appendix E, *OKI MSM82C55A Data Sheet*. The individual register description gives the address, type, word size, and bit map of the register, followed by a description of each bit.

The register bit map shows a diagram of the register with the most significant bit (MSB) (bit 7 for an 8-bit register) shown on the left, and the least significant bit (LSB), bit 0, shown on the right. A rectangle labeled with the bit name inside its rectangle represents each bit. An asterisk (*) after the bit name indicates that the bit is inverted (negative logic).

In a few of the registers, several bits are labeled with an X, indicating *don't care bits*. When you read a register, these bits may appear set or cleared but should be ignored because they have no significance. When you write to a register, setting or clearing these bit locations has no effect on the DAQCard-1200 hardware.

The bit map field for some write-only registers states *not applicable, no bits used*. Writing to these registers causes some event to occur on the DAQCard-1200, such as clearing the analog input circuitry. The data is ignored when writing to these registers; therefore, any bit pattern will suffice.

Configuration and Status Register Group

The eight registers of the Configuration and Status Register Group allow general control and monitoring of the DAQCard-1200 A/D and D/A circuitry.

Command Register 1 and Command Register 2 contain bits that control the operation modes of the A/D and D/A circuitry. Command Register 3 enables or disables interrupt operations. Use Command Register 4 to select the analog input mode and to allow certain DAQ signals to be externally driven at the I/O connector. Use Command Register 5 for software calibration of the A/D circuitry. Use Command Register 6 to enable and disable interrupt operations and to configure the A/D and D/A circuitry.

Status Register 1 reports the status of a DAQ operation and the status of analog output during waveform generation. Status Register 2 reports the status of a DAQ operation and gives access to the output of the EEPROM.

On power up or through a software reset, all of the Command Registers are cleared.

Bit descriptions for the registers in the Configuration and Status Register Group are on the following pages.

Command Register 1

Use Command Register 1 to select the input channel you want to read, the gain for the analog input circuitry, the DAQ scanning mode, and the coding used for the output of the ADC.

Address: 00 (hex)

Type: Write-only

Word Size: 8-bit

Bit Map:

7	6	5	4	3	2	1	0
SCANEN	GAIN2	GAIN1	GAIN0	TWOSCMP	MA2	MA1	MA0

Bit	Name	Description
7	SCANEN	Scan Enable—This bit enables or disables multiple-channel scanning during data acquisition. Set this bit to scan the analog channels as specified by MA<2..0> and SE*/D (bit 3 of Command Register 4). Clear this bit to sample a single analog channel specified by MA<2..0> and SE*/D during the entire DAQ operation.
6–4	GAIN<2..0>	Gain—These three bits select the gain setting as follows:

GAIN<2..0>	Selected Gain
000	1
001	Invalid
010	2
011	5
100	10
101	20
110	50
111	100

3	TWOSCMP	Two's Complement—This bit selects the coding format of the ADC output. Set this bit to sign-extend the 12-bit data from the ADC to 16 bits (two's complement). Clear this bit to make bits 12 through 15 return 0 (straight binary).

Bit	Name	Description (continued)
2–0	MA<2..0>	Multiplexed Analog—These three bits select which of the eight input channels are read. The analog input multiplexers depend on these bits and also on SCANEN and SE*/D. Input channels are selected as follows:

MA<2..0>	Selected Analog Input Channels		
	Single-Ended Mode	Differential Mode	
	Scan Disabled/Enabled	Scan Disabled	Scan Enabled
000	0	0	0
001	1	0	2
010	2	2	4
011	3	2	6
100	4	4	0
101	5	4	2
110	6	6	4
111	7	6	6

In single-ended mode (SE*/D cleared), if you set SCANEN, analog channels MA<2..0> through 0 are sampled sequentially. If you clear SCANEN, a single analog channel specified by MA<2..0> is sampled during the entire DAQ operation.

In DIFF mode, the number of analog inputs reduces to four. The single-ended input channels 0 and 1 (pins 3 and 4) become differential input channel 0. The single-ended input channels 2 and 3 (pins 5 and 6) become differential input channel 2. There are no odd differential input channels.

Command Register 2

Command Register 2 contains eight bits that control the DAQCard-1200 analog input trigger modes, analog output update modes, and the coding scheme of the DACs.

Address: 01 (hex)

Type: Write-only

Word Size: 8-bit

Bit Map:

7	6	5	4	3	2	1	0
LDAC1	LDAC0	2SDAC1	2SDAC0	TBSEL	SWTRIG	HWTRIG	PRETRIG

Bit	Name	Description
7	LDAC1	LDAC1—This bit enables timer waveform generation from DAC1. If you set this bit, DAC1 updates its output at regular intervals as determined by counter A2 or the EXTUPDATE* signal at the I/O connector. If you clear this bit, the voltage output of DAC1 is immediately updated when data is loaded into the DAC1 High-Byte Register.
6	LDAC0	LDAC0—This bit enables timer waveform generation from DAC0. If you set this bit, DAC0 updates its output at regular intervals as determined by counter A2 or the EXTUPDATE* signal at the I/O connector. If you clear this bit, the voltage output of DAC0 is immediately updated when data is loaded into the DAC0 High-Byte Register.
5	2SDAC1	2SDAC1—This bit selects the binary coding scheme used for the DAC1 data. If you set this bit, a two's complement binary coding scheme is used for interpreting the 12-bit data. Two's complement is used with bipolar output mode. If you clear this bit, a straight binary coding scheme is used. Straight binary is used with unipolar output mode.
4	2SDAC0	2SDAC0—This bit selects the binary coding scheme used for the DAC0 data. If you set this bit, then a two's complement binary coding scheme is used for interpreting the 12-bit data. Two's complement is used with bipolar output mode. If you clear this bit, a straight binary coding scheme is used. Straight binary is used with unipolar output mode.

Bit	Name	Description (continued)
3	TBSEL	Time Base Select—This bit selects the clock source for counter A0, the sample interval timer. If you clear this bit, a 1 MHz clock drives counter A0, and the interval between samples is the value loaded into counter A0 multiplied by 1 µs. If you set this bit, the output of counter B0 is used as the clock source. The timebase for counter B0 is fixed at 2 MHz. The sample interval is the value loaded into counter A0 multiplied by the period of the output signal from counter B0.
2	SWTRIG	Software Trigger—This bit is a software trigger for a DAQ operation. You can trigger a DAQ operation by setting this bit. The terminal count signal of counter A1 or a cleared SWTRIG terminates a DAQ process.
1	HWTRIG	Hardware Trigger—This bit enables or disables the posttrigger mode using the EXTTRIG signal at the I/O connector. If you set this bit, you can use the EXTTRIG signal to trigger a DAQ operation in place of SWTRIG. A DAQ process is terminated by a terminal count signal of counter A1 or by writing to the A/D Clear Register. You must clear PRETRIG to use this mode.
0	PRETRIG	Pretrigger—This bit enables or disables the pretrigger mode using the EXTTRIG signal at the I/O connector. If you set this bit, you can use the EXTTRIG signal at the I/O connector to terminate a DAQ operation by using counter A1. Data acquisition is terminated by a terminal count on A1. You must clear the HWTRIG to use this mode.

Command Register 3

The Command Register 3 contains four bits that enable and disable interrupts for a DAQ operation and for digital I/O.

Address: 02 (hex)

Type: Write-only

Word Size: 8-bit

Bit Map:

7	6	5	4	3	2	1	0
0	0	FIFOINTEN	ERRINTEN	CNTINTEN	0	DIOINTEN	0

Bit	Name	Description
7, 6, 2, 0	0	Always leave these bits cleared.
5	FIFOINTEN	FIFO Interrupt Enable—This bit enables and disables the generation of an interrupt when an A/D conversion result is available to be read from the A/D FIFO. If you set FIFOINTEN, an interrupt is generated whenever the DAVAIL bit becomes set in Status Register 1. Service this interrupt by reading the data from the FIFO.
4	ERRINTEN	Error Interrupt Enable—This bit enables and disables the generation of an interrupt when an A/D error condition is detected. If you set ERRINTEN, an interrupt is generated whenever the OVERFLOW or OVERRUN bit becomes set in Status Register 1. Service the interrupt by writing to the A/D Clear Register.
3	CNTINTEN	Counter Interrupt Enable—This bit enables the counter A2 output or the EXTUPDATE* signal to generate an interrupt. If you set CNTINTEN, an interrupt occurs whenever the CNTINT bit becomes set in Status Register 1. Clear this interrupt by writing to the Timer Interrupt Clear Register. This interrupt allows waveform generation on the analog output because the same signal that sets the interrupt also updates the DAC output if the corresponding LDAC bit in Command Register 2 is set.
1	DIOINTEN	DIO Interrupt Enable—This bit enables or disables generation of an interrupt when either Port A or Port B is ready to transfer data, and an interrupt request is set via PC3 or PC0 of 82C55A. See Appendix D, *OKI MSM82C55A Data Sheet*, for details. Clear this interrupt by clearing PC3 or PC0. If you clear DIOINTEN, the interrupts from PC3 or PC0 are disabled.

Command Register 4

Use this register to select the analog input mode, to enable interval scanning, and to allow the I/O connector pins to externally drive certain DAQ signals.

Address: 0F (hex)

Type: Write-only

Word Size: 8-bit

Bit Map:

7	6	5	4	3	2	1	0
0	0	0	ECLKRCV	SE*/D	ECLKDRV	EOIRCV	INTSCAN

Bit	Name	Description
7–5	0	Always leave these bits cleared.
4	ECLKRCV	External Clock Receive—This bit disables or enables the external signal EXTCONV*. If you set this bit, transitions on EXTCONV* will not effect data acquisition. If you clear this bit, a falling edge on EXTCONV* initiates an A/D conversion.
3	SE*/D	Single-Ended/Differential—This bit, along with bit 0 of Command Register 6 (RSE*/NRSE), selects one of three analog input modes of the DAQCard-1200. You can select the single-ended mode by clearing this bit and you can select the differential mode by setting this bit. Refer to the *DAQCard-1200 User Manual* for an explanation of the different modes. The following table illustrates how to choose the various input modes by using SE*/D and RSE*/NRSE.

RSE*/NRSE	SE*/D	Input Mode
0	0	RSE (reset condition)
1	0	NRSE
X	1	DIFF

2	ECLKDRV	External Clock Drive—When you reset this bit (default power up), you can drive the EXTCONV* pin at the I/O connector to cause conversions (if ECLKRCV is also cleared). When you set this bit, you enable internally timed conversions and you drive conversion pulses onto the EXTCONV pin for synchronizing channels on SCXI modules (if used with SCXI).

Bit	Name	Description (continued)
1	EOIRCV	External Output Interval Clock Receive—This bit selects the clock source for interval scanning. If you clear this bit, counter B1 drives the interval scanning circuitry. This will also configure OUTB1 on the I/O Connector as an output signal. If you set this bit, OUTB1 on the I/O Connector is selected as an input signal and allows you to externally drive the interval scanning circuitry.
0	INTSCAN	Interval Scan—This bit distinguishes between the continuous and interval DAQ modes. When you set this bit, the DAQCard-1200 performs interval data acquisition. If you clear this bit, continuous data acquisition occurs. For an explanation of the two different modes, refer to Chapter 4, *Theory of Operation*, in the *DAQCard-1200 User Manual.* Also, this bit selects the clock source for counter B1 used in interval scanning. If interval scanning is disabled (INTSCAN = 0), then counter B1 is available for user applications. You can then drive CLKB1 externally at the I/O connector. If interval scanning is enabled (INTSCAN = 1), the clock source of counter A0 also drives CLKB1. This source can further be selected by using the TBSEL bit in Command Register 2.

Command Register 5

Use Command Register 5 for software calibration of the A/D and D/A circuitry, for interaction with the EEPROM, and for enabling dither.

Address: 1C (hex)

Type: Write-only

Word Size: 8-bit

Bit Map:

7	6	5	4	3	2	1	0
EEPROMCS	SDATA	SCLK	CALDACLD	DITHEREN	WRTPRT	0	0

Bit	Name	Description
7	EEPROMCS	EEPROM Chip Select—This bit enables and disables the EEPROM. You can enable the EEPROM for both read and write operations by setting this bit. You can disable the EEPROM by clearing this bit. Notice that this bit is inverted on the DAQCard-1200 to make EEPROMCS* as explained in the *Calibration* section of Chapter 4, *Theory of Operation*, in your *DAQCard-1200 User Manual*.
6	SDATA	Serial Data—This bit is a serial data input for both the calibration DACs and the EEPROM.
5	SCLK	Serial Clock—This bit is a serial clock for both the calibration DACs and the EEPROM. A low-to-high transition of this bit clocks data into the EEPROM (during a write operation) and the calibration DAC. A high-to-low transition of the bit clocks data out of the EEPROM (during a read operation).
4	CALDACLD	Calibration DAC Load—This bit updates the calibration DACs. After you load the calibration DAC address and data, set CALDACLD high to update the selected CALDAC output signal.
3	DITHEREN	Dither Enable—This bit enables or disables the dither circuitry. When you set this bit, 0.5 LSB of white Gaussian noise is added to the selected analog input signal. By enabling dither and using averaging, you can achieve greater input resolution.
2	WRTPRT	Write Protect—This bit controls the write protect input signal for the EEPROM. When you set this bit, you enable normal write operations. When you clear this bit, you disable write operations.
1, 0	0	Always leave these bits cleared.

Command Register 6

Use Command Register 6 to enable A/D interrupts and to configure the A/D and D/A circuitry.

Address: 0E (hex)

Type: Write-only

Word Size: 8-bit

Bit Map:

7	6	5	4	3	2	1	0
0	DQINTEN	HFINTEN	0	DAC1UNI/BI*	DAC0UNI/BI*	ADCUNI/BI*	RSE*/NRSE

Bit	Name	Description
7, 4	0	Always leave these bits cleared.
6	DQINTEN	DAQ Interrupt Enable—This bit enables and disables the end of a DAQ operation interrupt. Set this bit to generate an interrupt whenever the OUTA1 bit in Status Register 2 becomes set. Service this interrupt by resetting counter A1. Clear this bit to disable interrupt generation.
5	HFINTEN	Half-Full Interrupt Enable—This bit enables and disables the FIFO half-full interrupt. Set this bit to generate an interrupt whenever the FIFOHF* bit in Status Register 2 becomes cleared. Service this interrupt by reading data from the FIFO. Clear this bit to disable interrupt generation.
3	DAC1UNI/BI*	DAC1 Unipolar/Bipolar—This bit sets the analog voltage output range for DAC1. Set this bit to configure DAC1 for a unipolar (0 to +10 V) output voltage range. Clear this bit to configure DAC1 for a bipolar (-5 to +5 V) output voltage range.
2	DAC0UNI/BI*	DAC0 Unipolar/Bipolar—This bit sets the analog voltage output range for DAC0. Set this bit to configure DAC0 for a unipolar (0 to +10 V) output voltage range. Clear this bit to configure DAC0 for a bipolar (-5 to +5 V) output voltage range.

Bit	Name	Description
1	ADCUNI/BI*	ADC Unipolar/Bipolar—This bit sets the analog voltage input range for data acquisition. Set this bit to select a unipolar (0 to +10 V) voltage input range. Clear this bit to select a bipolar (-5 to +5 V) voltage input range.
0	RSE*/NRSE	Referenced Single-Ended/Nonreferenced Single-Ended—This bit, and bit 3 of Command Register 4 (SE*/D), selects one of three input modes of the DAQCard-1200. The status of RSE*/NRSE is only important with the single-ended analog-input modes. Set this bit to select the nonreferenced single-ended mode. Clear this bit to select the referenced single-ended mode. For an explanation of the three input modes, refer to the *DAQCard-1200 User Manual*.

Status Register 1

Status Register 1 indicates the status of the current DAQ operation and the status of analog output during waveform generation. These bits indicate if a DAQ operation is in progress or if data is available, whether any errors have been found, and the analog output interrupt status.

Address: 00 (hex)

Type: Read-only

Word Size: 8-bit

Bit Map:

7	6	5	4	3	2	1	0
X	EXTGATA0	GATA0	X	CNTINT	OVERFLOW	OVERRUN	DAVAIL

Bit	Name	Description
7, 4	X	Don't care bits.
6	EXTGATA0	External Gate A0—This bit indicates the status of the external trigger signal (EXTTRIG) during a DAQ operation in posttrigger mode. If this bit is set, EXTTRIG has triggered a DAQ operation. Clear this bit by writing to the A/D Clear Register.
5	GATA0	Gate A0—This bit indicates the status of the GATEA0 input for counter A0. Use this bit as a busy indicator for DAQ operations because conversions are enabled as long as GATEA0 is high and counter A0 is programmed appropriately. A DAQ operation is terminated when GATA0 is cleared.
3	CNTINT	Counter Interrupt—This bit reflects the status of the interrupt caused by counter A2 output or the EXTUPDATE* signal. A low-to-high transition on counter A2 output or on EXTUPDATE* sets this bit. You can generate an interrupt if you set CNTINTEN in Command Register 3. Clear this bit by writing to the Timer Interrupt Clear Register.
2	OVERFLOW	Overflow—This bit indicates if an overflow error has occurred. If this bit is cleared, no error was encountered. If this bit is set, the A/D FIFO has overflowed because the DAQ servicing operation could not keep up with the sampling rate.
1	OVERRUN	Overrun—This bit indicates if an overrun error has occurred. If this bit is cleared, no error occurred. This bit is set if a convert command is issued to the ADC while the last conversion is still in progress.
0	DAVAIL	Data Available—This bit indicates if conversion output is available. If this bit is set, the ADC is finished with the last conversion and the result can be read from the FIFO. This bit is cleared if the FIFO is empty. After writing to the A/D Clear Register, this bit is set. Take two 8-bit readings of FIFO to clear this bit.

Status Register 2

Status Register 2 reports the status of a DAQ operation and the FIFO half-full output as well as gives access to the output of the EEPROM.

Address: 1D (hex)

Type: Read-only

Word Size: 8-bit

Bit Map:

7	6	5	4	3	2	1	0
X	X	X	X	X	FIFOHF*	OUTA1	PROMOUT

Bit	Name	Description
7–3	X	Don't care bits.
2	FIFOHF*	FIFO Half-Full—This bit indicates the status of the FIFO half-full output. If the on-card FIFO sets this bit (negative logic, therefore FIFOHF* low), it means that the FIFO is half full. If the FIFO clears this bit (FIFOHF* high), the FIFO is less than half full.
1	OUTA1	Output Counter A1—This bit indicates the status of the output signal of counter A1. If the user sets the counter A1 mode for a terminal count, OUTA1 low indicates counter A1 has not started counting or that counting is in progress. OUTA1 high indicates that counter A1 has finished counting.
0	PROMOUT	EEPROM Output—This bit allows access to the serial output pin of the EEPROM. During calibration procedures, the software reads the calibration data from the EEPROM through PROMOUT and writes the data to the calibration DACs.

Analog Input Register Group

The five registers making up the Analog Input Register Group control the analog input circuitry and the FIFO. Reading the FIFO Register returns stored A/D conversion results. Writing to the Start Convert Register initiates a single A/D conversion. Writing to the A/D Clear Register clears the analog input circuitry.

Bit descriptions for the registers of the Analog Input Register Group are on the following pages.

A/D FIFO Register

The 12-bit A/D conversion results are sign-extended to 16-bit data in either two's complement or straight binary format and are stored into a 2 K word deep A/D FIFO buffer. Two consecutive 8-bit readings of the A/D FIFO Register return an A/D conversion value stored in the A/D FIFO. The first reading returns the low byte of the 16-bit value, and the second reading returns the high byte. The value read is removed from the A/D FIFO, thereby freeing space for another A/D conversion value to be stored.

The A/D FIFO is empty when all values it contains are read. The DAVAIL bit in Status Register 1 should be read before the A/D FIFO Register is read. If the A/D FIFO contains one or more A/D conversion values, the DAVAIL bit is set and the A/D FIFO Register can be read to retrieve a value. If the DAVAIL bit is cleared, the A/D FIFO is empty. Therefore, reading the A/D FIFO Register returns meaningless information.

The values returned by reading the A/D FIFO Register are available in two different binary formats: straight binary or two's complement binary. The binary format used is selected by the TWOSCMP bit in Command Register 1. The bit pattern returned for either format is as follows.

Address: 0A (hex)

Type: Read-only

Word Size: 8-bit

Bit Map: Straight binary mode

High Byte

7	6	5	4	3	2	1	0
0	0	0	0	D11	D10	D9	D8

Low Byte

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

Bit	Name	Description
High Byte		
7–0	D<15..8>	Data—These bits contain the high byte of the straight binary result of a 12-bit A/D conversion. Bits D<15..12> are always 0 in straight binary mode. Values made up of D<11..0>, therefore, range from 0 to +4,095 decimal (0000 to 0FFF hex). Straight binary mode is useful for unipolar analog input readings because all values read reflect a positive polarity input signal.
Low Byte		
7–0	D<7..0>	Data—These bits contain the low byte of the straight binary result of a 12-bit A/D conversion. The first of the two consecutive readings of the A/D FIFO Register returns this byte.

Bit Map: Two's complement binary mode

High Byte

7	6	5	4	3	2	1	0
D15	D14	D13	D12	D11	D10	D9	D8

Low Byte

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

Bit	Name	Description
High Byte		
7–0	D<15..8>	Data—These data bits contain the high byte of the 16-bit, sign-extended two's complement result of a 12-bit A/D conversion. Values made up of D<15..0>, therefore, range from -2,048 to +2,047 decimal (F800 to 07FF hex). Two's complement mode is useful for bipolar analog input readings because the values read reflect the polarity of the input signal.
Low Byte		
7–0	D<7..0>	Data—These data bits contain the low byte of the 16-bit, sign-extended two's complement result of a 12-bit A/D conversion. The first of the two consecutive readings of A/D FIFO Register returns this byte.

A/D Clear Register

Write to this register to reset the ADC. This operation clears the FIFO and loads the last conversion value into the FIFO. All error bits in Status Register 1 are cleared as well. Notice that the FIFO contains one data word after reset, so two consecutive FIFO readings are necessary after reset to empty the FIFO. The data that is read should be ignored.

Address: 08 (hex)

Type: Write-only

Word Size: 8-bit

Bit Map: Not applicable, no bits used

Start Convert Register

Write to the Start Convert Register location to initiate a single A/D conversion.

Address: 03 (hex)

Type: Write-only

Word Size: 8-bit

Bit Map: Not applicable, no bits used

Analog Output Register Group

Use the four registers of the Analog Output Register Group to load the two 12-bit DACs. DAC0 controls analog output channel 0. DAC1 controls analog output channel 1. Write to these DACs individually.

Bit descriptions of the registers making up the Analog Output Register Group are on the following pages.

Note: DAC*x* ***and*** LDAC*x* ***represent DAC0, DAC1, LDAC0 and LDAC1 registers.***

DAC0 Low-Byte, DAC0 High-Byte, DAC1 Low-Byte, and DAC1 High-Byte Registers

Write to DAC0 Low-Byte and then to DAC0 High-Byte to load DAC0. Write to DAC1 Low-Byte and then to DAC1 High-Byte to load DAC1. If you clear the LDAC*x* bit in Command Register 2, then the corresponding analog output channel is updated immediately after you write to the DAC*x* High-Byte register. If you set the LDAC*x* bit, then the corresponding analog output channel is updated when an active low pulse occurs on the output of counter A2 or on the EXTUPDATE* line on the I/O connector.

Address: 04 (hex) DAC0 Low Byte.
05 (hex) DAC0 High Byte.
06 (hex) DAC1 Low Byte.
07 (hex) DAC1 High Byte.

Type: Write-only (all)

Word Size: 8-bit (all)

Bit Map:

DAC*x*H

7	6	5	4	3	2	1	0
D15	D14	D13	D12	D11	D10	D9	D8

DAC*x*L

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

Bit	Name	Description
DAC*x*H		
7–4	D<15..12>	Data—Zero in straight binary mode, sign extension in two's complement mode.
3–0	D<11..8>	Data—These four bits are loaded into the specified DAC high byte.
DAC*x*L		
7–0	D<7..0>	Data—These eight bits are loaded into the specified DAC low byte. The low byte should be loaded first, followed by the corresponding high byte.

82C53 Counter/Timer Register Groups A and B

The nine registers of the two counter/timer register groups access the two onboard 82C53 counter/timers. Each 82C53 has three counters. For convenience, the two counter/timer groups and their respective 82C53 integrated circuits have been designated A and B. The three counters of group A control onboard DAQ timing and waveform generation. You can use the three counters of group B for general-purpose timing functions.

Each 82C53 has three independent 16-bit counters and one 8-bit Mode Register. The Mode Register sets the mode of operation for each of the three counters.

Writing to the Timer Interrupt Clear Register clears the interrupt request asserted when a low pulse is detected on the output of counter A2 or on the EXTUPDATE* line.

Bit descriptions for the registers in the Counter/Timer Register Groups are in the following pages.

Counter A0 Data Register

Use the Counter A0 Data Register to load and read back contents of 82C53(A) counter 0. Counter A0 is the sample interval counter for a DAQ operation.

Address: 14 (hex)

Type: Read-and-write

Word Size: 8-bit

Bit Map:

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

Bit	Name	Description
7–0	D<7..0>	Data—8-bit counter A0 contents.

Counter A1 Data Register

Use the Counter A1 Data Register to load and read back contents of 82C53(A) counter 1. Counter A1 is the sample counter for a DAQ operation.

Address: 15 (hex)

Type: Read-and-write

Word Size: 8-bit

Bit Map:

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

Bit	Name	Description
7–0	D<7..0>	Data—8-bit counter A1 contents.

Counter A2 Data Register

Use the Counter A2 Data Register to load and read back contents of 82C53(A) counter A2. Counter A2 is the DAC update timer for waveform generation.

Address: 16 (hex)

Type: Read-and-write

Word Size: 8-bit

Bit Map:

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

Bit	Name	Description
7–0	D<7..0>	Data—8-bit counter A2 contents.

Counter A Mode Register

The Counter A Mode Register determines the operation mode for each of the three counters on the 82C53(A) chip. The Counter A Mode Register selects the counter involved, its read/load mode, its operation mode (that is, any of the 82C53 six operation modes), and the counting mode (binary or BCD counting).

The Counter A Mode Register is an 8-bit register. Bit and programming descriptions for each of these bits are in Appendix D, *OKI MSM82C53 Data Sheet*.

Address: 17 (hex)

Type: Write-only

Word Size: 8-bit

Bit Map:

7	6	5	4	3	2	1	0
SC1	SC0	RL1	RL0	M2	M1	M0	BCD

Timer Interrupt Clear Register

Write to the Timer Interrupt Clear Register to clear the interrupt request asserted when a low pulse is detected on the counter A2 output or on EXTUPDATE* line.

Address: 0C (hex)

Type: Write-only

Word Size: 8-bit

Bit Map: Not applicable, no bits used.

Counter B0 Data Register

Use the Counter B0 Data Register to load and read back the contents of 82C53(B) counter 0. Counter B0 either supplies the time base clock for counter A0 or is reserved for external usage.

Address: 18 (hex)

Type: Read-and-write

Word Size: 8-bit

Bit Map:

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

Bit	Name	Description
7–0	D<7..0>	Data—8-bit counter B0 contents.

Counter B1 Data Register

Use the Counter B1 Data Register to load and read back the contents of 82C53(B) counter 1. Counter B1 is either the interval timer for a DAQ operation or is reserved for external usage.

Address: 19 (hex)

Type: Read-and-write

Word Size: 8-bit

Bit Map:

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

Bit	Name	Description
7–0	D<7..0>	Data—8-bit counter B1 contents.

Counter B2 Data Register

Use the Counter B2 Data Register to load and read back the contents of 82C53(B) counter 2. Counter B2 is reserved for external usage.

Address: 1A (hex)

Type: Read-and-write

Word Size: 8-bit

Bit Map:

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

Bit	Name	Description
7–0	D<7..0>	Data—8-bit counter B2 contents.

Counter B Mode Register

The Counter B Mode Register determines the operation mode for each of the three counters on the 82C53(B) chip. The Counter B Mode Register selects the counter involved, its read/load mode, its operation mode (that is, any of the 82C53 six operation modes), and the counting mode (binary or BCD counting).

The Counter Mode Register is an 8-bit register. Bit descriptions for each of these bits are in Appendix D, *OKI MSM82C53 Data Sheet*.

Address: 1B (hex)

Type: Write-only

Word Size: 8-bit

Bit Map:

7	6	5	4	3	2	1	0
SC1	SC0	RL1	RL0	M2	M1	M0	BCD

82C55A Digital I/O Register Group

Digital I/O on the DAQCard-1200 uses an 82C55A integrated circuit. The 82C55A is a general-purpose peripheral interface containing 24 programmable I/O pins. These pins represent the three 8-bit I/O ports (A, B, and C) of the 82C55A and the 24 digital I/O lines on the I/O connector. You can program these ports as two groups of 12 signals or as three individual 8-bit ports. You can also configure them as either input or output pins. Use the Digital Control Register to configure the three ports. For further information on the 82C55A, refer to Appendix E, *OKI MSM82C55A Data Sheet*.

Bit descriptions for the registers in the Digital I/O Register Group are on the following pages.

Port A Register

Use the Port A Register to load and to read the eight digital I/O lines constituting port A on the I/O connector (PA<0..7>). See Appendix E, *OKI MSM82C55A Data Sheet*, for information on how to configure port A for input or output.

Address: 10 (hex)

Type: Read-and-write

Word Size: 8-bit

Bit Map:

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

Bit	Name	Description
7–0	D<7..0>	Data—8-bit port A data.

Port B Register

Use the Port B Register to load and to read the eight digital I/O lines constituting port B, that is, PB<0..7>. See Appendix E, *OKI MSM82C55A Data Sheet*, for information on how to configure port B for input or output.

Address: 11 (hex)

Type: Read-and-write

Word Size: 8-bit

Bit Map:

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

Bit	Name	Description
7–0	D<7..0>	Data—8-bit port B data.

Port C Register

You can use port C as an 8-bit I/O port like port A and port B if neither port A nor port B is used in handshaking (latched) mode. If either port A or port B is configured for latched I/O, some of the bits in port C are used for handshaking signals. See Appendix E, *OKI MSM82C55A Data Sheet,* for a description of the individual bits in the Port C Register.

Address: 12 (hex)

Type: Read-and-write

Word Size: 8-bit

Bit Map:

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

Bit	Name	Description
7–0	D<7..0>	Data—8-bit port C data.

Digital Control Register

You can use the Digital Control Register to configure port A, port B, and port C as inputs or outputs, and you can select simple mode (basic I/O) or handshaking mode (strobed I/O) for transfers. See Appendix E, *OKI MSM82C55A Data Sheet*, for a description of the individual bits in the Digital Control Register.

Address: 13 (hex)

Type: Write-only

Word Size: 8-bit

Bit Map:

7	6	5	4	3	2	1	0
CW7	CW6	CW5	CW4	CW3	CW2	CW1	CW0

Bit	Name	Description
7–0	CW<7..0>	Control Word—8-bit control word.

Interval Counter Register Group

Use the 8-bit Interval Counter only in the single-channel interval mode (SCANEN = 0 and INTSCAN = 1). Refer to the *DAQ Timing* section in Chapter 4, *Theory of Operation*, of your *DAQCard-1200 User Manual* for an explanation of interval mode. The Interval Counter consists of two 8-bit registers—the Interval Counter Data Register and the Interval Counter Strobe Register. Load the Interval Counter Data Register with the count. Writing to the Interval Counter Strobe Register loads this count into the Interval Counter. The Interval Counter decrements with each conversion. When the count reaches 0, the Interval Counter autoinitializes, restoring the original count value.

Bit descriptions for the registers in the Interval Counter Register Group are on the following pages.

Interval Counter Data Register

Load the Interval Counter Data Register with the desired number of samples to be acquired within an interval cycle of a single channel DAQ operation.

Address: 1E (hex)

Type: Write-only

Word Size: 8-bit

Bit Map:

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

Bit	Name	Description
7–0	D<7..0>	Data—Interval counter count.

Interval Counter Strobe Register

Writing to Interval Counter Strobe Register strobes the contents of the Interval Counter Data Register into the Interval Counter. This action arms the Interval Counter, which decrements with each conversion pulse.

Address: 1F (hex)

Type: Write-only

Word Size: 8-bit

Bit Map:

7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	1

Bit	Name	Description
7–1	0	Each of these bits must be 0 for proper operation of the DAQCard-1200.
0	1	This bit must be 1 for proper operation of the DAQCard-1200.

Chapter 3
Programming

This chapter contains programming instructions for operating the DAQCard-1200 circuitry. Programming the DAQCard-1200 involves writing to and reading from the various registers described in Chapter 2, *Register Descriptions*. The programming instructions included here list the sequence of steps to take. The instructions are language independent; that is, they tell you to read from or write to a given register or to detect whether a given bit is set or cleared without presenting the actual code.

For information concerning the Fujitsu MB88341 CALDACs, refer to Appendix B, *Fujitsu MB88341/MB88342 Data Sheet.* For information about the Xicor X25020 SPI Serial EEPROM, see Appendix C, *Xicor X25020 Data Sheet*. For additional details concerning the 82C53 counter/timer, see Appendix D, *OKI MSM82C53 Data Sheet*. For information about the 82C55A programmable port interface (PPI), see Appendix E, *OKI MSM82C55A Data Sheet*.

Note: ***In this chapter, all numbers preceded by 0x are given in hexadecimal and X signifies bits that should remain cleared.***

PCMCIA Card Initialization

Before you can access the data acquisition circuitry on the DAQCard-1200, the card must be activated using Card Services. PCMCIA I/O cards are kept inactive until a program has requested that Card Services activate the card by assigning an interrupt level and an address space for the card's I/O registers. The DAQCard-1200 requires a 32-byte I/O address window and one interrupt level.

There are at least three different ways of activating the card:

- If you are using the DAQCard-1200 with National Instruments software such as NI-DAQ, LabVIEW, and LabWindows/CVI, the `DAQCONF` or `WDAQCONF` programs request the activation of the card. For more information about this procedure, see the *Board Installation and Configuration* section of Chapter 1 in the *NI-DAQ User Manual for PC Compatibles* in your DAQCard-1200 kit.

- If you are not using any of the programs listed above, you can use a third party card services software such as CardWare to activate the card. Specifically, the `PCCARD.EXE` and `PCENABLE.EXE` programs in CardWare allow a particular card to be activated after a definition has been created for it in the `CARDWARE.INI` file. For more information about these programs, see the CardWare user manual.

- If neither of these options is feasible for your application, you can develop your own program to activate the card. However, this is fairly complicated, and it requires significantly more programming. If you develop your own program, you should consult the documents, *Card Services Specifications* and *Socket Services Specifications* (PCMCIA) which explain how to activate a card using system-level calls. You will need to request an I/O window, an interrupt level, and a configuration. In the configuration, the configuration index should be

set to 01 hex for normal operation. For more information about these operations, see the PCMCIA documents, *Card Services Specifications* and *Socket Services Specifications*.

After you activate the card, you are ready to configure the DAQCard-1200 for the data acquisition setup. The following pages explain how to set the registers for different operations.

Register Programming Considerations

The DAQCard-1200 supports 8-bit I/O transfers; thus, all the I/O read-and-write operations are 8-bit operations. You must do 16-bit transfers in two consecutive 8-bit operations.

Several write-only registers on the DAQCard-1200 contain bits that control several independent pieces of the onboard circuitry. In the set or clear instructions, specific register bits should be set or cleared without changing the current state of the remaining bits in the register. However, writing to these registers affects all register bits simultaneously. Because you cannot read these registers to determine their current status, you should maintain a software copy of the write-only registers. To change the state of a single bit without disturbing the remaining bits, set or clear the bit in the software copy and then write the modified software copy to the register.

Initializing the DAQCard-1200 Circuitry

Initializing the DAQCard-1200 circuitry involves two steps—reset and calibration. The DAQCard-1200 circuitry is reset on power up. The DAQCard-1200 circuitry can also be reset through software. Calibration of the analog input and output circuitry involves reading the factory-defined calibration constants from the onboard EEPROM and loading them into the appropriate calibration DACs. For information on using user-defined calibration constants, refer to Chapter 4, *Calibration*.

When reset, the DAQCard-1200 is left in the following state:

1. All of the command registers are cleared.
2. The A/D FIFO has one invalid word of data.
3. All of the interrupts are disabled and cleared.
4. The 82C55A digital I/O is reset.
5. The analog output DACs are reset to 0.0 V.

Calibrating the DAQCard-1200 involves two steps—reading eight factory-defined calibration constants from the EEPROM and writing each value to the appropriate CALDAC. Choose the desired factory-defined calibration constants as explained in Chapter 2, *Theory of Operation*, of your *DAQCard-1200 User Manual*.

Use the following sequence of steps to read a single byte from the EEPROM.

1. Set the EEPROMCS and WRTPRT bits in Command Register 5.
2. Serially write the READ instruction, 0x03, to the EEPROM to start a read operation.
3. Serially write the 8-bit address.
4. Serially read one byte of data from the EEPROM.
5. Clear the EEPROMCS and WRTPRT bits in Command Register 5 to end the read operation.

Repeat the following two steps eight times to serially write one byte to the EEPROM.

1. Write a single bit of the 8-bit value, MSB first, by setting or clearing the SDATA bit in Command Register 5. The SCLK bit in Command Register 5 should be cleared during this step.
2. Set the SCLK bit in Command Register 5.

Repeat the following three steps eight times to serially read one byte from the EEPROM.

1. Set the SCLK bit in Command Register 5.
2. Clear the SCLK bit in Command Register 5.
3. Read a single bit of the 8-bit value, MSB first, by reading the PROMOUT bit of Status Register 2.

After reading a single calibration constant from the EEPROM, you must write this value to the appropriate CALDAC. Use the following steps to write a single byte to a CALDAC.

1. Write the 4-bit address of the desired CALDAC (address 0x3-0xA), LSB first. CALDACLD should be cleared during this step.
2. Write the 8-bit calibration constant, MSB first.
3. Set the CALDACLD bit in Command Register 5.
4. Clear the CALDACLD bit in Command Register 5.

Repeat the following steps the appropriate number of times to serially write required bits to the CALDAC.

1. Write a single bit of the value by setting or clearing the SDATA bit in Command Register 5. SCLK should be cleared during this step.
2. Set the SCLK bit in Command Register 5.

Repeat the calibration operation for each CALDAC to fully calibrate the DAQCard-1200.

Programming the Analog Input Circuitry for Single A/D Conversions

This section explains how to clear the analog input circuitry, how to configure the analog input circuitry, and how to perform single A/D conversions.

Clearing the Analog Input Circuitry

Before clearing the analog input circuitry, perform the following steps:

1. Clear SWTRIG in Command Register 2.
2. Ensure that EXTCONV* from the I/O connector does not cause any conversions.
3. Ensure that EXTTRIG from the I/O connector does not cause any conversions.

   Clearing the analog input circuitry does not stop a DAQ operation that was started by the SWTRIG bit in Command Register 2. Therefore, you should clear this bit before clearing the analog input circuitry.

   While clearing the analog input circuitry, do not externally drive the EXTCONV* pin on the I/O connector or, if you do, drive it high. Another option is to set the ECLKRCV bit in Command Register 4. This blocks any transitions on the EXTCONV* pin and no unwanted conversions will occur while you are clearing the analog input circuitry.

   While clearing the analog input circuitry, do not externally drive the EXTTRIG pin on the I/O connector or , if you do, drive it high. Another option is to make sure the HWTRIG bit in Command Register 2 is cleared. This blocks any transitions on the EXTTRIG pin and no unwanted conversions will occur while you are clearing the analog input circuitry.

   The analog input circuitry can be cleared by writing to the A/D Clear Register, which leaves the analog input circuitry in the following state:

1. Analog input status bits (OVERRUN, OVERFLOW, DAVAIL and FIFOHF*) are cleared.
2. Pending interrupt requests from the analog input circuitry are cleared.
3. A/D FIFO has one invalid word of data.

The command registers are not cleared, so you do not necessarily have to reconfigure the DAQCard-1200 before initiating another DAQ sequence. After clearing the analog input circuitry, empty the A/D FIFO before starting any A/D conversions by performing two 8-bit reads on the A/D FIFO Register and ignoring the data that is read.

Configuring the Analog Input Circuitry

Configure the analog input circuitry after initializing the DAQCard-1200 and anytime the characteristics of the analog input signals change. Program the appropriate register bits as follows (not necessarily in this order):

- Select the appropriate input mode (DIFF, NRSE, or RSE).
- Select the input polarity (bipolar or unipolar) and coding of the resulting conversions (straight binary or two's complement).
- Select the analog input channel and gain.

You determine the input mode by identifying the types of signal sources that you are using. For more information about determining the input mode, consult the *DAQCard-1200 User Manual.* Select the input mode by setting or clearing the RSE*/NRSE and SE*/D bits in Command Register 6 and Command Register 4, respectively.

You determine the input polarity according to the voltage range of the analog input signal. For more information about determining the input polarity, consult the *DAQCard-1200 User Manual*. The conversion coding is then selected according to the input polarity. If you select a bipolar range for the DAC, the two's complement coding is recommended. If you select a unipolar range for the DAC, the straight binary coding is recommended. Select the input polarity by setting or clearing the ADCUNI/BI* bit in Command Register 6. Select the respective coding by setting or clearing the TWOSCMP bit in Command Register 1.

If you will be performing single-channel data acquisition, select the desired analog channel and gain by setting or clearing the GAIN<2..0> and MA<2..0> bits in Command Register 1. You must also clear SCANEN in Command Register 1. If you will be doing multiple channel data acquisition, set MA<2..0> to specify the highest numbered channel in the scan sequence. For example, if you set MA<2..0> to 011 (binary), the following scan sequence is used:

channel 3, channel 2, channel 1, channel 0, channel 3, channel 2, and so on.

Select the analog input channel and gain for multiple channel data acquisition in the following order.

1. Set the gain and the highest channel number in the scan sequence in Command Register 1. Clear the SCANEN bit during this first write to Command Register 1.
2. Write to Command Register 1 again, but this time set the SCANEN bit. This latches the channel number into the scan counter.

Performing Single A/D Conversions

After configuring the analog input circuitry, you can perform single A/D conversions and then read the conversion result from the A/D FIFO. Perform the following steps:

1. Set counter A0 so that OUTA0 is high.

2. Initiate a conversion by writing to the Start Convert Register.

3. Read the conversion result from the A/D FIFO.

You must program counter A0 so that OUTA0 is high. You can do this by writing 0x14 (select counter A0, mode 2) to Counter A Mode Register. This enables the conversions initiated by writing to the Start Convert Register.

Initiate a conversion by writing to the Start Convert Register. When you initiate an A/D conversion, the ADC stores the result in the A/D FIFO at the end of its conversion cycle (approximately 8 µs later). You can acquire a specific number of samples by writing to the Start Convert Register that same number of times. In multiple-channel data acquisition, the hardware scans the channels as described before with each write to the Start Convert Register.

You obtain A/D conversion results by reading the A/D FIFO Register. First, you must read the status registers to determine the state of the A/D FIFO. The useful status bits are OVERRUN, OVERFLOW, and DAVAIL in Status Register 1 and FIFOHF* in Status Register 2. The DAVAIL bit will be set if there is at least one conversion result stored in the A/D FIFO. The DAVAIL bit should become set within a maximum of 12 µs after you initiate an A/D conversion. If DAVAIL is set, you can follow the Status Register read by reading the A/D FIFO. If the FIFOHF* bit is cleared, you can follow the Status Register read with 1,024 consecutive readings of the A/D FIFO. If either the OVERFLOW or the OVERRUN bit is set, an error has occurred. You have either lost at least one conversion by overflowing the A/D FIFO, or you have attempted to initiate a conversion before the previous one has completed.

You must read the A/D FIFO twice to obtain the result. The first reading returns the low byte of the 16-bit data, and the second reading returns the high byte. Reading the A/D FIFO removes the A/D conversion result from the A/D FIFO. If the DAVAIL bit is cleared, then the A/D FIFO is empty and further reading of the A/D FIFO returns meaningless data.

Programming a DAQ Operation Using Internal Timing

A sequence of timed A/D conversions is referred to in this manual as a DAQ operation. There are two parameters that are timed in a DAQ operation—the sample interval time and the total number of samples. There are four different internal counters that can help you time these two parameters—counter A0, counter A1, counter B0, and counter B1. There are four different DAQ operation modes—continuous, interval, controlled, and freerun.

A continuous DAQ operation differs from an interval DAQ operation in the timing of the sample intervals. A controlled DAQ operation differs from a freerun DAQ operation in the counting of the number of samples. You can combine these different modes and perform up to four different DAQ operations—controlled continuous, controlled interval, freerun continuous, and freerun interval.

In a continuous DAQ mode, only one counter is required to time the sample intervals. Use counter A0 to time the sample intervals without any delays. Each sample is taken with the same sample interval. Counter A0 is clocked by a 1 MHz clock. The period of counter A0, or the sample interval, is equal to the value programmed into counter A0 multiplied by 1 µs. The minimum period that can be selected for counter A0 is 2 µs. Choose your sample interval according to the specifications in Chapter 4, *Theory of Operation*, of your *DAQCard-1200 User Manual,* to maintain 12-bit accuracy.

Alternatively, a programmable timebase for counter A0 is available by using counter B0. Counter B0 has a fixed, unalterable 2 MHz clock as its own timebase. Therefore, its period is the value programmed into counter B0 multiplied by 500 ns. The minimum period that can be selected for counter B0 is 1 µs. The period of counter A0, or the sample interval, is then equal to the period of counter B0 multiplied by the value programmed into counter A0. The maximum sample interval is approximately 35 minutes. You can use counter B0 in any DAQ operation as an alternative timebase for the sample interval.

In an interval DAQ mode, you need two counters. Use counter B1 to time the interval cycle. Within the interval cycle, each conversion occurs at regular sample intervals timed by counter A0. In multiple channel data acquisition, the conversions stop after the sample counter A1 counts down to 0. In single channel data acquisition, the conversions stop after the programmed count in the Interval Counter Register has expired. An interval DAQ operation consists of back-to-back interval cycles. Counter B1 is clocked by the same timebase used for counter A0. If the 1 MHz clock is used for the timebase of counter A0, then the period of counter B1, or the interval cycle, is the value programmed into counter B1 times 1 µs.

In a controlled DAQ mode, use counter A1 as a sample counter. In this mode, you can perform a specified number of conversions, after which the hardware ends the DAQ operation. The number of conversions in a single DAQ operation in this case is limited to a 16-bit count (or 65,535 samples).

In a freerun DAQ mode, no onboard counters are used to count the total number of samples taken. The software can keep track of the number of conversions that have occurred and can turn off counter A0 after a certain number of conversions have been obtained or after some other user-defined criteria have been met. The number of conversions in a single DAQ operation in this case is unlimited.

Programming a DAQ operation requires setting up the four available counters, setting up the Interval Counter Register in single channel interval mode, and then triggering the DAQ operation. If you are using counter B0 or counter B1 internally for a DAQ operation, be sure that GATB0 or GATB1 are not being driven externally through the I/O connector or are being driven high. Perform the following steps to program a DAQ operation:

1. Clear the analog input circuitry.

2. Configure the analog input circuitry.

3. Set up the four available counters (and the Interval Counter Register if necessary).

4. Trigger the operation.

5. Service the operation.

Perform the configuration and clear the analog input circuitry as described in *Programming the Analog Input Circuitry for Single A/D Conversions*.

Programming Counter A0 and Counter B0

You must always program counter A0 for a DAQ operation using internal timing. A high-to-low transition on OUTA0 (counter A0 output) initiates a conversion. You can program counter A0 to generate a pulse once every *N* µs, where *N* is the value programmed into counter A0 and the clock input is a 1 MHz signal. The minimum number that you can use for *N* is 2. The sample interval can then be programmed to be between 2 µs and 65,535 µs. Use the following equation to determine the sample interval:

$$sample\ interval = N \bullet 1\ \mu s$$

Use the following sequence to program counter A0, the sample interval counter. All writes are 8-bit write operations.

1. Write 0x34 (select counter A0, mode 2) to the Counter A Mode Register.
2. Write the least significant byte of the sample interval (*N*) to the Counter A0 Data Register.
3. Write the most significant byte of the sample interval (*N*) to the Counter A0 Data Register.

You can achieve a larger option of sample intervals by using counter B0 along with counter A0. The resulting sample interval is then N_b multiplied by N_a multiplied by 500 ns, where N_b is the value programmed into counter B0 and N_a is the value programmed into counter A0. The sample interval can then be programmed to be between 2 µs and $[(65{,}535)^2/2]$µs. Use the following equation to determine the sample interval:

$$sample\ interval = N_a \bullet N_b \bullet 500\ ns$$

Use the following programming sequence to program counter B0 as an alternative timebase for counter A0. All writes are 8-bit operations.

1. Set the TBSEL bit in Command Register 2.
2. Write 0x36 (select counter B0, mode 3) to the Counter B Mode Register.
3. Write the least significant byte of the sample interval (N_b) to the Counter B0 Data Register.
4. Write the most significant byte of the sample interval (N_b) to the Counter B0 Data Register.

Programming Counter A1

You must program counter A1 even if you are planning on doing a freerun DAQ operation. In a freerun DAQ operation, set the output of counter A1 low to gate counter A0 on and thus allow conversions to occur. The number of samples you want to acquire in a controlled DAQ operation is equal to $N + 2$, where N is the value programmed into counter A1. Therefore the number of samples can range from 3 to 65,537. Use the following equation to determine the number of samples:

$$number\ of\ samples = N + 2.$$

Use the following sequence to program counter A1. All writes are 8-bit operations.

1. Write 0x70 (select counter A1, mode 0) to the Counter A Mode Register. This step sets the output of counter A1 (OUTA1) low.

   Note: ***Continue to steps 2 and 3 only for controlled DAQ operations.***

2. Write the least significant byte of the sample count (*N*) to the Counter A1 Data Register.

3. Write the most significant byte of the sample count (*N*) to the Counter A1 Data Register.

Programming Counter B1 and the Interval Counter Register

If you are doing interval data acquisition, you must program counter B1. CLKB1 (the clock input of counter B1) is the same as that used for counter A0 in order to synchronize the individual conversions and the interval cycle. The interval cycle is equal to *N* (the value programmed into counter B1) multiplied by the clock period used for counter A0. The interval cycle should be longer than the total conversion time. In a multiple-channel interval DAQ operation, the total conversion time is equal to the number of channels being scanned, multiplied by the sample interval. In a single-channel interval DAQ operation, the total conversion time is equal to the number loaded into the Interval Counter Register, multiplied by the sample interval.

For single-channel interval data acquisition, the number that you program into the Interval Counter Register should be smaller than the total number of desired samples. For example, if you want to acquire 2,000 samples in batches of 100, load the Interval Counter Register with 100 and the sample counter (counter A1) with 2,000. In this example, 20 interval cycles are required to convert 2,000 samples. The Interval Counter Register is an 8-bit register. Therefore, you can convert up to 255 samples in a single interval cycle. Use the following sequence to program the Interval Counter Register:

1. Write the desired number of samples of a single channel that will be acquired during an interval cycle to the Interval Counter Data Register.

2. Write 0x01 to the Interval Counter Strobe Register to latch the Interval Counter Register.

Use the following sequence to program counter B1 for interval data acquisition. All of the writes are 8-bit operations.

1. Set the INTSCAN bit in Command Register 4.

2. Clear the EOIRCV bit in Command Register 4.

3. Write 0x74 (select counter B1, mode 2) to the Counter B Mode Register.

4. Write the least significant byte of the interval cycle (*N*) to the Counter B1 Data Register.

5. Write the most significant byte of the interval cycle (*N*) to the Counter B1 Data Register.

Triggering the DAQ Operation

To start the DAQ operation, set the SWTRIG bit in Command Register 2, which enables counter A0 to start counting. In a freerun DAQ operation (or in any other type of DAQ operation in which you want to stop the operation prematurely), you can stop the DAQ operation by clearing the SWTRIG bit.

Note: ***In interval DAQ operations, the first interval cycle is not synchronized with counter B1. Therefore you may wish to discard the conversions acquired in the first interval cycle.***

Servicing the DAQ Operation

When you start a DAQ operation, you must service the operation by reading the A/D FIFO Register. You can either read the A/D FIFO every time a conversion is available or when the A/D FIFO is half full. You read the A/D FIFO as explained in the section *Programming Single A/D Conversions*. You can also use interrupts to service the DAQ operation. This is discussed in the *A/D Interrupt Programming* section later in this chapter.

Two error conditions, overflow or overrun, may occur during a DAQ operation. If these error conditions occur, the OVERFLOW and/or the OVERRUN bits are set in Status Register 1. Check these bits every time you read Status Register 1 to check the DAVAIL bit.

An overflow condition occurs if more than 1,024 A/D conversions have been stored in the A/D FIFO without the A/D FIFO being read; that is, the A/D FIFO is full and cannot accept any more data. This condition occurs if the software loop reading the A/D FIFO is not fast enough to keep up with the A/D conversion rate. When an overflow occurs, at least one A/D conversion result is lost.

An overrun condition occurs if a second A/D conversion is initiated before the previous conversion is finished. This condition may result in one or more missing A/D conversions. This condition occurs if the sample interval is smaller than the conversion time of the ADC.

Programming a DAQ Operation Using External Timing

You can use three external timing signals, EXTTRIG, EXTCONV* and OUTB1, to time a DAQ operation. Use EXTTRIG to initiate a DAQ operation (posttrigger mode) or to terminate an ongoing DAQ operation (pretrigger mode). In posttrigger mode, you use EXTTRIG in place of writing to the SWTRIG bit in Command Register 2. Use the EXTCONV* signal to time the individual A/D conversions in place of counter A0. If you are performing an interval DAQ operation and are timing the individual A/D conversions using EXTCONV*, you must time the interval cycle through OUTB1 in place of using counter B1. For signal specifications of these external timing signals, see Chapter 3, *Signal Connections,* in the *DAQCard-1200 User Manual*.

Programming a DAQ Operation Using EXTCONV*

If you want to use EXTCONV* instead of counter A0 for sample-interval timing, you can follow the same sequence of steps described in *Programming a DAQ Operation Using Internal Timing,* except for programming counters A0 and B0. Replace the steps for programming counters A0 and B0 with the following sequence.

1. Write 0x34 (select counter A0, mode 2) to the Counter A Mode Register.

2. Clear the ECLKRCV bit in Command Register 4.

You must set up counter A0 to force OUTA0 high and enable conversions initiated by EXTCONV*. You also need to ensure that the ECLKRCV bit in Command Register 4 is cleared to enable EXTCONV*.

Note: ***After you trigger the DAQ operation, using either SWTRIG or EXTTRIG in posttrigger mode, the first EXTCONV* pulse may not cause an A/D conversion. See Chapter 3,*** *Signal Connections,* ***in the*** *DAQCard-1200 User Manual* ***for specifications regarding EXTCONV*.***

Programming a DAQ Operation Using EXTTRIG in Posttrigger Mode

If you want to use EXTTRIG in posttrigger mode instead of SWTRIG to trigger a DAQ operation, you can follow the same sequence of steps described in *Programming a DAQ Operation Using Internal Timing* except for the section describing triggering of the DAQ operation. Use the following sequence to trigger a DAQ operation using EXTTRIG:

1. Set the HWTRIG bit in Command Register 2.

2. Trigger a DAQ operation with a rising edge on EXTTRIG.

When you set the HWTRIG bit in Command Register 2, the next rising edge on EXTTRIG will trigger a DAQ operation. Further transitions on EXTTRIG do not affect anything. In a freerun DAQ operation (or in a controlled DAQ operation triggered by EXTTRIG in which you want to stop the operation prematurely), you can stop conversions by first clearing the HWTRIG bit in Command Register 2 and then writing to the A/D Clear Register. After writing to the A/D Clear Register, any remaining data in the FIFO will have been cleared.

Programming a DAQ Operation Using EXTTRIG in Pretrigger Mode

If you want to use EXTTRIG in a pretrigger mode, you must trigger the DAQ operation using SWTRIG, and you must program counter A1 as described in *Programming a DAQ Operation Using Internal Timing*. However, the number of samples that you want to occur after the EXTTRIG trigger is equal to *N + 1*, where *N* is the programmed count in counter A1. The number of samples that can occur after the EXTTRIG trigger ranges from 3 to 65,536. Use the following sequence to use EXTTRIG in a pretrigger mode:

1. During configuration of the analog input circuitry, set the PRETRIG bit in Command Register 2.

2. Clear the analog input circuitry, program the appropriate counters, and trigger the DAQ operation as described in *Programming a DAQ Operation Using Internal Timing*.

3. Gate on counter A1 with a low-to-high transition on EXTTRIG.

The first rising edge on EXTTRIG should occur after you trigger the DAQ operation using SWTRIG. This rising edge gates on counter A1. The DAQ operation stops after the programmed count in counter A1 has expired.

Programming a DAQ Operation Using OUTB1

If you want to drive your own interval pulse on OUTB1 instead of using counter B1 to time the interval cycle, you can follow the same sequence of steps as described in *Programming a DAQ Operation Using Internal Timing,* except for the section describing the programming of counter B1. Use the following sequence of steps to use OUTB1 in place of counter B1:

1. Program the Interval Counter Register (if necessary).

2. Set the INTSCAN bit in Command Register 4.

3. Set the EOIRCV bit in Command Register 4.

If you want to externally time the interval cycle, you should also externally time the sample interval through EXTCONV* to synchronize the sample interval and the interval cycle. Refer to Chapter 3, *Signal Connections*, in the *DAQCard-1200 User Manual* for timing specifications.

DAQ Interrupt Programming

Four different interrupts are generated by the analog input circuitry.

1. An interrupt when a conversion is available to be read from the A/D FIFO.

2. An interrupt when the A/D FIFO is half full.

3. An interrupt when an error condition (overflow or overrun) is detected.

4. An interrupt when a DAQ operation is terminated by counter A1.

You can enable these four interrupts individually.

Set the FIFOINTEN bit in Command Register 3 to enable interrupt generation when a conversion is available to be read from the A/D FIFO. If FIFOINTEN is set, an interrupt is generated whenever the DAVAIL bit in Status Register 1 is set.

Set the HFINTEN bit in Command Register 6 to enable interrupt generation when the A/D FIFO becomes half full (512 samples). If HFINTEN is set, an interrupt is generated whenever the FIFOHF* bit is cleared in Status Register 2.

Set the ERRINTEN bit in Command Register 3 to enable interrupt generation when an error condition is detected. If ERRINTEN is set, an interrupt is generated whenever either the OVERFLOW or OVERRUN bits are set in Status Register 1.

Set the DQINTEN bit in Command Register 6 to enable interrupt generation when a DAQ operation is terminated by counter A1. If DQINTEN is set, an interrupt is generated whenever OUTA1 is set in Status Register 2.

Programming the Analog Output Circuitry

This section explains how to configure the analog output circuitry and how to update the analog output voltage.

Configuring the Analog Output Circuitry

You must configure the analog output circuitry after initializing the DAQCard-1200 and anytime the characteristics of the analog output signals change. Program the appropriate register bits as follows:

1. Select the output polarity (unipolar or bipolar) and the coding of the digital code (straight binary or two's complement).

2. Set or clear the DAC1UNI/BI* and DAC0UNI/BI* bits in Command Register 6 and the 2SDAC1 and 2SDAC0 bits in Command Register 2.

If you select a unipolar output polarity, the straight binary coding is recommended. If you select a bipolar output polarity, the two's complement coding is recommended.

Use the following formula to calculate the output voltage versus digital code for a unipolar analog output configuration and straight binary coding. The digital code is a decimal value ranging from 0 to +4,095.

$$V_{out} = 10.0 \bullet \frac{Code}{4,095}$$

The following formula calculates the output voltage versus digital code for a bipolar analog output configuration and two's complement coding. The digital code is a decimal value ranging from -2,048 to +2,047.

$$V_{out} = 5.0 \bullet \frac{Code}{2,048}$$

Tables 3-1 and 3-2 show the analog output voltage versus the input digital code in unipolar and bipolar mode, respectively.

Table 3-1. Analog Output Voltage Versus Digital Code
(Unipolar Mode, Straight Binary Coding)

Digital Code		Voltage Output
Decimal	**Hex**	
0	0000	0 V
1	0001	2.4414 mV
2,048	0800	5.0 V
4,095	0FFF	9.9976 V

Table 3-2. Analog Output Voltage Versus Digital Code
(Bipolar Mode, Two's Complement Coding)

Digital Code		Voltage Output
Decimal	**Hex**	
-2,048	F800	-5.0 V
-1,024	FC00	2.5 V
0	0000	0.0 V
1,024	0400	2.5 V
2,047	07FF	4.9976 V

Programming the Update Mode of the Analog Output Circuitry

The analog output circuitry on the DAQCard-1200 uses double-buffered DACs. Therefore the output voltages (DAC0OUT and DAC1OUT on the I/O connector) do not have to be updated immediately with each write to the DAC Data Registers. You can update the analog output in synchronization with counter A2 or the external update timing signal EXTUPDATE*. This is useful for waveform generation applications because the timed update pulses eliminate the timing jitter associated with software writes to the DAC Data Registers.

You can operate the analog output circuitry in three ways—immediate update, update on OUTA2, or update on EXTUPDATE*.

Use the following sequence to program for immediate update mode:

1. Clear the LDAC1 or LDAC0 bit in Command Register 2.

2. Write the low byte to the DAC0 or DAC1 Low-Byte Register.

3. Write the high byte to the DAC0 or DAC1 High-Byte Register.

The analog output voltage is updated immediately after you write the high byte to the DAC0 or DAC1 High-Byte Register.

Use the following sequence to program for update on OUTA2 or EXTUPDATE*.

1. Set the LDAC1 or LDAC0 bit in Command Register 2.
2. Write to the Interrupt Enable Register to enable interrupt generation.
3. Set the CNTINTEN bit in Command Register 3 to enable timer interrupts.
4. Program counter A2.
5. Service the interrupts for waveform generation.

You must program counter A2 even if you are using EXTUPDATE*. If you are using EXTUPDATE*, you must set OUTA2 high to enable updates caused by EXTUPDATE*. If you are using counter A2, do not drive EXTUPDATE* externally or, if you do, drive it high. You can not block EXTUPDATE* through software. The clock for counter A2 is the same as that used for counter A0, so you can use a 1 MHz clock or the output of counter B0 to clock counter A2. The update period is equal to the value programmed into counter A2 times the period of the clock. You should set the update period to be longer than the time it takes to write a new value to the DAC Registers. Use the following sequence to program counter A2.

1. Write a 0xB4 (select counter A2, mode 2) to the Counter A Mode Register.

   Note: ***Continue to steps 2 and 3 if you are using counter A2 to update the analog output circuitry.***

2. Write the least significant byte of the update period to the Counter A2 Data Register.
3. Write the most significant byte of the update period to the Counter A2 Data Register.

The update cycle starts immediately after you write the most significant byte to the Counter A2 Data Register. If you are using EXTUPDATE*, the update cycle starts after setting OUTA2 high and on the first falling edge of EXTUPDATE*.

DAC Interrupt Programming

Set the CNTINTEN bit in Command Register 3 to enable interrupt generation on the rising edge of either OUTA2 or EXTUPDATE*. If CNTINTEN is set, then an interrupt is generated whenever the CNTINT bit in Status Register 1 is set.

You service this interrupt by writing a new value to DAC0 or DAC1 Low-Byte and High-Byte Registers. To clear the interrupt, write to the Timer Interrupt Clear Register. This allows continuous waveform generation. The DAC output voltage is then updated by a high-to-low transition on either OUTA2 or EXTUPDATE*.

Programming the Digital I/O Circuitry

Digital I/O on the DAQCard-1200 uses the 82C55A integrated circuit. Programming the digital I/O circuitry involves setting the mode of the 82C55A by writing to the Digital Control Register and then writing and reading from the three port registers (port A, port B, and port C). The various modes of the 82C55A are illustrated in Appendix E, *OKI MSM82C55A Data Sheet*. Examples for using the digital I/O circuitry are in the *DAQCard-1200 User Manual*.

You can generate interrupts through PC0 and PC3 on the I/O connector. Enable digital I/O interrupts by setting the DIOINTEN bit in Command Register 3. There are no status bits associated with the digital I/O interrupts. You clear this interrupt by clearing PC0 and PC3.

Programming the General-Purpose Counter/Timers

You can use Counter/Timer Group B of the 82C53 timing circuitry as general-purpose counters when they are not being used for internal timing. To program the general-purpose counters, set the mode of the 82C53 by writing to the Counter B Mode Register, then write and read from the three data registers (counter B0, counter B1, and counter B2). See Appendix D, *OKI MSM82C53 Data Sheet* for information on the various modes of the 82C53. Examples for using the general-purpose counters are given in the *DAQCard-1200 User Manual*. You cannot generate interrupts with the general-purpose counter/timers.

Chapter 4
Calibration

This chapter contains instructions for creating user-defined calibration constants for the DAQCard-1200 CALDACs. This information is important if you do not want to use NI-DAQ to create user-defined calibration constants to be stored in the user areas of the DAQCard-1200 EEPROM. Since the calibration process is quite complicated, the user is advised to use NI-DAQ whenever possible. If NI-DAQ does not support your operating system, only then should you try to write register level code to perform calibration. Also, if you accidentally overwrite the factory area, you will permanently loose factory calibration information, and may have to send your unit back to National Instruments for recalibration.

Note: ***National Instruments is not liable for accidental overwriting of the calibration EEPROM in the field.***

For information concerning writing register-level programs to write to the CALDACs, refer to Chapter 3, *Programming*. For information on calibration equipment requirements, refer to the *DAQCard-1200 User Manual*.

Storing User-Defined Constants

You should store only one set of user-defined calibration constants in one user area. One set of calibration constants consists of twenty constants, six for CALDACs 3, 4 and 7–10, seven for CALDAC6, and seven for CALDAC5 (one at each gain setting). Therefore, one set of calibration constants can calibrate the DAQCard-1200 analog input and analog output circuitry in either bipolar or unipolar polarity and at all gains.

Store your user-defined calibration constants in the same format as that shown for the factory-defined calibration constants in Table 4-1. For example, if you use user area 1, store the calibration constants for CALDACs 3–10 in EEPROM addresses 117–110, then store the seven gain constants for CALDAC6 in EEPROM locations 109–102, followed by the seven post-gain offset values for each gain in locations 101–94. Table 4-3 shows the DAQCard-1200 EEPROM map.

Note that the location for gain 1.25 is provided; however, the corresponding value is ignored. Also note that within each user area, the AI CALDAC 3 value is the analog input gain calibration constant for the gain at which the calibration is performed. This value must also be duplicated in the corresponding location in the GAIN *X* value in that user area. For example, if the calibration is performed at gain = 5, then the calibration constant must be written in both locations 114 and 106 for user area 1. Similarly, the post-gain offset value for that gain must be duplicated in the corresponding Gain *X* value-offset location.

When the DAQCard-1200 is shipped, the contents of factory area for bipolar mode are copied in all the user areas. Consequently, there will be constants in the user areas that will be very accurate when the DAQCard-1200 is used in the bipolar mode for both analog input and output.

To save your user-defined calibration constants in the EEPROM, you need programming instructions for writing to the EEPROM. Use the following sequence of steps to store a calibration constant to the EEPROM. For information on user areas in the EEPROM, refer to the EEPROM memory map at the end of this chapter.

1. Set the EEPROMCS and WRTPRT bits in Command Register 5.
2. Serially write the WRITE instruction, 0x06, to the EEPROM to enable a write operation.
3. Clear the EEPROMCS bit in Command Register 5.
4. Set the EEPROMCS bit in Command Register 5.
5. Serially write the WRITE instruction, 0x02, to the EEPROM to start a write operation.
6. Serially write the 8-bit user area address to the EEPROM.
7. Serially write the 8-bit calibration constant to the EEPROM.
8. Clear the EEPROMCS and WRTPRT bits in Command Register 5.

Repeat the following two steps eight times to serially write one byte to the EEPROM.

1. Clear the SCLK bit in Command Register 5. Write a single bit of the 8-bit value, MSB first, by setting or clearing the SDATA bit in Command Register 5.
2. Set the SCLK bit in Command Register 5.

After you store a single calibration constant, you must wait a minimum of 10 ms before accessing the EEPROM again. Alternatively, you may use the RDSR (Read status register) instruction of the X25020 and poll the WIP (write-in-process) bit. If this bit is 1, then the previous write is still in progress. You must wait till this bit reads 0. Then you may proceed with the next write.

Calibration DACs

There are eight 8-bit calibration DACs (CALDACs) on the DAQCard-1200 that are used for calibration. These DACs are described in Tables 4-1 and 4-2 for analog input and output calibration respectively. The tolerance in both tables is simply the adjustment range divided by 255 (8-bit). In Table 4-1, V_{ref} refers to the reference voltage value that you provide during the gain calibration procedure. In Table 4-2, V_{ref} refers to the voltage that you write to either DAC0 or DAC1 during the gain calibration procedure.

Table 4-1. Calibration DAC Characteristics for Analog Input Circuitry

DAC Name	Function	Adjustment Range	Tolerance
CALDAC3	Pregain offset, coarse	820 LSBs (gain = 100)	3.2 LSBs
CALDAC4	Pregain offset, fine	8 LSBs (gain =100)	0.04 LSBs
CALDAC5	Postgain offset	82 LSBs	0.32 LSBs
CALDAC6	Gain	2% of V_{ref}	0.008% of V_{ref}

Table 4-2. Calibration DAC Characteristics for Analog Output Circuitry

DAC Name	Function	Adjustment Range	Tolerance
CALDAC7	Offset, DAC0	31 LSBs	0.12 LSBs
CALDAC8	Gain, DAC0	2% of V_{ref}	0.008% of V_{ref}
CALDAC9	Offset, DAC1	31 LSBs	0.12 LSBs
CALDAC10	Gain, DAC1	2% of V_{ref}	0.008% of V_{ref}

Analog Input Calibration

To null out error sources, you must calibrate the analog input circuitry by adjusting the following potential sources of error (not necessarily in this order):

- Offset error at the input of the instrumentation amplifier.
- Offset error at the input of the ADC.
- Gain error of the analog input circuitry.

Offsets at the input to the instrumentation amplifier contribute gain-dependent error to the analog input system. This offset is multiplied by the gain of the instrumentation amplifier. To calibrate this offset, you must ground the inputs of the instrumentation amplifier, measure the input at two different gains, and adjust CALDAC3 and CALDAC4 until the measured offset in LSBs is independent of the gain setting. Calibration of this pregain offset is done in bipolar mode for both bipolar and unipolar analog input configurations.

Offset error at the input of the ADC is the total of the voltage offsets contributed by the circuitry from the output of the instrumentation amplifier to the ADC input (including the ADC's own offsets). Offset errors appear as a voltage added to the input voltage being measured. To calibrate this offset, you must connect either AGND in bipolar mode or an external voltage source in unipolar mode to the inputs of the ADC and adjust CALDAC5 until the measured voltage is equal to either AGND or (external voltage source + gain error) respectively.

If the three analog input offset DACs are adjusted in this way, there is no significant residual offset error, and reading a grounded channel returns, on average, less than 0.5 LSB regardless of the gain setting.

All the stages up to and including the input of the ADC contribute to the gain error of the analog input circuitry. With the instrumentation amplifier set to a gain of 1, the gain of the analog input circuitry is ideally 1. The gain error is the deviation of the gain from 1 and appears as a multiplication of the input voltage being measured. To eliminate this error source, you must measure the input first with the inputs grounded and then with the inputs connected to the external voltage source. Then adjust CALDAC6 until the measured difference between the two voltages is equal to the value of the external voltage source. After the DAQCard-1200 is calibrated at a gain of 1, there is only a small residual gain error (±0.5% max) at the other gains. To reduce this error, calibrate the card at all other gains and for corresponding values stored in the EEPROM User gain area.

In both bipolar and unipolar modes, calibration of post-gain offset does not affect the gain characteristics. However, gain calibration does affect post-gain offset. Therefore, you must perform gain calibration before post-gain calibration.

Perform the calibration procedure for both bipolar and unipolar mode with dither enabled (by setting the DITHEREN bit in Command Register 5) and in referenced single-ended mode (clear

the RSE*/NRSE bit in Command Register 6 and the SE*/D bit in Command Register 4). Refer to Table 4-1 for calibration tolerances.

Bipolar Input Calibration Procedure

If your board is configured for bipolar input, which provides the -5 to +5 V range, complete the following procedures. This procedure assumes that ADC readings are in the -2,048 to +2,047 range; that is, you have selected the two's complement coding scheme.

Because adjusting the gain affects the postgain offset adjustment, you must calibrate gain before calibrating postgain offset. Also, initialize all of the CALDACs for analog input (3, 4, 5, and 6) to 128 before starting the calibration procedure. This sets each CALDAC at midscale.

Pregain Offset Coarse Calibration

1. Connect ACH0 (pin 1 on the rear panel 50-pin I/O connector) to AGND (pin 11).
2. Take 1,024 readings from channel 0 at a gain of 1. Take the mean of these readings and call it *mean1*.
3. Take 1,024 readings from channel 0 at a gain of 100. Take the mean of these readings and call it *mean100*.
4. Adjust CALDAC3 so that:

   $|mean1 - mean100| \leq$ *pregain offset coarse calibration tolerance*.

Pregain Offset Fine Calibration

1. Repeat steps 1–3 of the pregain offset coarse procedure.
2. Adjust CALDAC4 so that:

   $|mean1 - mean100| \leq$ *pregain offset fine calibration tolerance.*

At this point, the pregain offset is nulled out. However, there is a residual postgain offset remaining.

Gain Calibration

1. Take 1,024 samples from channel 0 (still connected to AGND) at a gain of 1. Take the mean and call it *postgain_offset*.
2. Connect the voltage reference (V_{ref}) between ACH1 (pin 2) and AGND (pin 11). Choose a voltage reference between 3.0 and 4.5 V.
3. Take 1,024 samples from channel 1 at a gain of 1. Take the mean and call it *mean1*.
4. Adjust CALDAC6 so that:

$$\left|(mean1 - postgain\_offset) - \frac{Vref}{5\ V} \bullet 2{,}047\right| \leq gain\ calibration\ tolerance.$$

Postgain Offset Calibration

1. Take 1,024 samples from channel 0 (still connected to AGND) at a gain of 1. Take the mean and call this *postgain_offset*.
2. Adjust CALDAC5 so that:

$$|postgain\_offset| \leq postgain\ calibration\ tolerance.$$

Calibration at Higher Gains

If you have performed gain calibration at a gain of 1 and the gain is changed to a value not equal to 1 (i.e., 2, 5, 10, 20, 50, or 100), you will get a maximum gain error of 0.5%. In addition, the post-gain offset will change, so you must recalibrate both the post-gain offset (CALDAC5) as well as gain (CALDAC6) at that gain. If you perform gain and post-gain offset calibrations at all other gains and store these values in the EEPROM, the maximum gain error will be 0.02% at all gains. Follow the same steps as given in the *Gain Calibration* section, but use a gain not equal to 1.

Note: ***When you use a gain not equal to 1, remember that the voltage reference (V_{ref}) multiplied by the gain should be less than 4.5 V.***

Unipolar Input Calibration Procedure

If your board is configured for unipolar input, which has an input range of 0 to +10V, complete the following steps. This procedure assumes that your ADC readings are in the 0 to 4,095 range; that is, you have selected the straight binary coding scheme.

In unipolar mode, the offset can be negative and while doing offset adjustment, all of the acquired samples can be 0 V. This results in incorrect offset calibration. To correct this problem, initialize CALDAC5 to 0 to create a maximum positive offset. Initialize the other CALDACs (3, 4, and 6) to 128 as before.

Pregain Offset Calibration

Follow the same steps as in the *Bipolar Input Configuration Procedure* section for pregain offset coarse and pregain offset fine calibration. Remember to configure the analog input for bipolar mode when performing pregain offset calibration. Reconfigure the analog input for unipolar mode for gain and postgain offset calibration.

After the pregain offset calibration, there will be a residual positive post-gain offset remaining because the post-gain CALDAC is biased to one extreme.

Gain Calibration

For gain calibration, use the following procedure:

1. Take 1,024 samples from channel 0 (still connected to AGND) at a gain of 1. Take the mean and call it *postgain_offset.*
2. Connect the voltage reference (V_{ref}) between ACH1 (pin 2) and AGND (pin 11). Choose a voltage reference between 8.0 and 9.5 V.
3. Take 1,024 samples from channel 1 at a gain of 1. Take the mean and call it *mean1*.
4. Adjust CALDAC6 so that :

$$\left|(mean1 - postgain\_offset) - \frac{Vref}{10\ V} \bullet 4{,}095\right| \leq gain\ calibration\ tolerance.$$

Perform the calibration for gains not equal to 1 as you did for bipolar input. Remember that the voltage reference (V_{ref}) multiplied by the gain used should be less than 9.5 V.

Postgain Offset Calibration

For postgain offset calibration, use the following procedure:

1. Take 1,024 samples from channel 1 (still connected to V_{ref}) at a gain of 1. Take the mean and call this *mean1*.
2. Adjust CALDAC5 so that:

$$|mean1| \leq postgain\ calibration\ tolerance.$$

Analog Output Calibration

To null out error sources that affect the accuracy of the output voltages generated, you must calibrate the analog output circuitry by adjusting the following potential sources of error.

- Analog output offset error
- Analog output gain error

Offset error in the analog output circuitry equals the total of the voltage offsets contributed by each component in the circuitry. This error appears as a voltage difference between the desired voltage and the actual output voltage generated and is independent of the D/A voltage setting. To correct this offset error, the routine sets the D/A to 0 V and adjusts CALDAC7 or CALDAC9 (for DAC0 or DAC1 respectively) until the output voltage is 0 V.

Gain error in the analog output circuitry is the product of the gains contributed by each component in the circuitry. This error appears as a voltage difference between the desired voltage and the actual output voltage generated and is dependent on the D/A voltage setting. To correct this gain error, the routine sets the D/A to a positive voltage (V_{ref}) and adjusts CALDAC8 or CALDAC10 (for DAC0 or DAC1 respectively) until the output voltage corresponds to V_{ref}.

You must calibrate the analog input circuitry before calibrating the analog output circuitry because the output calibration procedure depends on the analog input circuitry. Also, for analog output calibration, set the analog input circuitry calibration to referenced single-ended (RSE), bipolar mode. Refer to Tables 4-1 and 4-2 for calibration tolerances.

Bipolar Output Calibration Procedure

If your board is configured for bipolar output, which provides the -5 to +5 V range, complete the following procedure. This procedure assumes that DAC coding is in the -2,048 to +2,047 range; that is, you have selected the two's complement coding scheme.

Initialize all of the CALDACs for analog output (7, 8, 9, and 10) to 128 before you start the calibration procedure. This sets each CALDAC at midscale. Perform gain calibration before offset calibration.

Gain Calibration

1. Write a value ($-V_{ref}$) to DAC0 or DAC1. Choose a value of *Vref* between 3 and 4.5 V.
2. Take 1,024 readings from analog input channel 2 (still connected to DAC0OUT or DAC1OUT) at a gain of 1. Take the mean and call it *mean_gain_neg_vref*.
3. Write the value *Vref* to DAC0 or DAC1.
4. Take 1024 readings from analog input channel 2 at a gain of 1. Take the mean and call it *mean_gain_vref*.
5. Adjust CALDAC8 (or CALDAC10) so that:

$$\left|(mean\_gain\_ref - mean\_gain\_neg\_ref) - \frac{Vref}{5\ V} \bullet 4095\right| \leq gain\ calibration\ tolerance.$$

Offset Calibration

1. Connect DAC0OUT, pin 10, (or DAC1OUT, pin 12) to analog input channel 2 (pin3).
2. Write a 0 to DAC0 (or DAC1).
3. Take 1,024 readings from analog input channel 2 at a gain of 1. Take the mean and call it *offset*.
4. Adjust CALDAC7 (or CALDAC9) so that:

   $|offset| \leq offset\ calibration\ tolerance$.

Unipolar Output Calibration Procedure

If your board is configured for unipolar output, which provides the 0 to +10 V range, complete the following procedure. This procedure assumes that DAC coding is in the 0 to +4,095 range; that is, you have selected the straight binary coding scheme.

Set analog input to bipolar mode for analog output unipolar offset calibration. For gain calibration, set the analog input mode to unipolar. Initialize all of the CALDACs for analog output (7–10) to 128 before starting the gain calibration procedure. This sets each CALDAC to mid-scale.

Gain Calibration

1. Write 0 to DAC0 (or DAC1). Choose a value between 8.0 and 9.5 V.
2. Take 1,024 readings from analog input channel 2 (still connected to DAC0OUT or DAC1OUT) at a gain of 1. Take the mean and call it *mean_gain_zero*.
3. Write the value v_{ref} to DAC0 or DAC1. Choose a value between 8.0 and 9.5 V.
4. Take 1,024 readings from analog input channel 2 at a gain of 1. Take the mean and call it $mean\_gain\_v_{ref}$.
3. Adjust CALDAC8 (or CALDAC10) so that:

$$\left|(mean\_gain\_v_{ref} - mean\_gain\_zero) - \frac{Vref}{10\ V} \bullet 4{,}095\right| \leq gain\ calibration\ tolerance.$$

Offset Calibration

1. Connect DAC0OUT, pin 10, (or DAC1OUT, pin 12) to analog input channel 2 (pin3).
2. Write a 0 to DAC0 (or DAC1).
3. Take 1,024 readings from analog input channel 2 at a gain of 1. Take the mean and call it *offset*.
4. Adjust CALDAC7 (or CALDAC9) so that:

   $|offset| \leq offset\ calibration\ tolerance$.

EEPROM Map

Table 4-3 shows part of the EEPROM map for the DAQCard-1200. Locations 180–255 contain information about the DAQCard-1200 that NI-DAQ uses. These locations are not shown and you should not access them. The factory bipolar area contains locations 156–179 and the factory unipolar area contains 132–155. The user areas are in the lower half of the EEPROM. The pointers from 120–127 are initialized to point to factory locations. To use user area calibration data, you must set these pointers to point to the appropriate user area. Notice that if you point the AI bipolar frame to user area 1, you must also point the corresponding gain and offset pointers to the user area 1 gain and offset frames respectively.

Note: ***NI-DAQ uses these pointers to load a set of calibration constants into the CALDAC each time you run a function pertaining to the DAQCard-1200. Hence, if you use NI-DAQ along with your register-level code, you must assign these pointers correctly or NI-DAQ will load incorrect values into the CALDACs.***

Table 4-3. DAQCard-1200 EEPROM Map

Location	Hex	Decimal	Description
179*	00	0	Factory AI CALDAC0 bipolar value
178*	00	0	Factory AI CALDAC1 bipolar value
177*	00	0	Factory AI CALDAC2 bipolar value
176*	00	0	Factory AI CALDAC3 bipolar value
175*	00	0	Factory AO CALDAC4 bipolar value
174*	00	0	Factory AO CALDAC5 bipolar value
173*	00	0	Factory AO CALDAC6 bipolar value
172*	00	0	Factory AO CALDAC7 bipolar value
171*	00	0	Factory Gain 1 bipolar value-gain
170*	00	0	Factory Gain 1.25 bipolar value-gain
169*	00	0	Factory Gain 2 bipolar value-gain
168*	00	0	Factory Gain 5 bipolar value-gain
167*	00	0	Factory Gain 10 bipolar value-gain
166*	00	0	Factory Gain 20 bipolar value-gain
165*	00	0	Factory Gain 50 bipolar value-gain
164*	00	0	Factory Gain 100 bipolar value-gain
163	00	0	Factory Gain 1 bipolar value-offset
162	00	0	Factory Gain 1.25 bipolar value-offset
161	00	0	Factory Gain 2 bipolar value-offset
160	00	0	Factory Gain 5 bipolar value-offset
159	00	0	Factory Gain 10 bipolar value-offset
158	00	0	Factory Gain 20 bipolar value-offset
157	00	0	Factory Gain 50 bipolar value-offset
156	00	0	Factory Gain 100 bipolar value-offset

(continues)

Table 4-3. DAQCard-1200 EEPROM Map (Continued)

Location	Hex	Decimal	Description
155	00	0	Factory AI CALDAC0 unipolar value
154	00	0	Factory AI CALDAC1 unipolar value
153	00	0	Factory AI CALDAC2 unipolar value
152	00	0	Factory AI CALDAC3 unipolar value
151	00	0	Factory AO CALDAC4 unipolar value
150	00	0	Factory AO CALDAC5 unipolar value
149	00	0	Factory AO CALDAC6 unipolar value
148	00	0	Factory AO CALDAC7 unipolar value
147	00	0	Factory Gain 1 unipolar value-gain
146	00	0	Factory Gain 1.25 unipolar value-gain
145	00	0	Factory Gain 2 unipolar value-gain
144	00	0	Factory Gain 5 unipolar value-gain
143	00	0	Factory Gain 10 unipolar value-gain
142	00	0	Factory Gain 20 unipolar value-gain
141	00	0	Factory Gain 50 unipolar value-gain
140	00	0	Factory Gain 100 unipolar value-gain
139	00	0	Factory Gain 1 unipolar value-offset
138	00	0	Factory Gain 1.25 unipolar value-offset
137	00	0	Factory Gain 2 unipolar value-offset
136	00	0	Factory Gain 5 unipolar value-offset
135	00	0	Factory Gain 10 unipolar value-offset
134	00	0	Factory Gain 20 unipolar value-offset
133	00	0	Factory Gain 50 unipolar value-offset
132	00	0	Factory Gain 100 unipolar value-offset
131	00	0	Not used
130	00	0	Not used
129	00	0	Not used
128	00	0	Not used
127	B3	179	Point to AI bipolar frame
126	A3	155	Point to AI unipolar frame
125	AF	175	Point to AO bipolar frame
124	9F	151	Point to AO unipolar frame
123	AB	171	Point to bipolar Gain frame
122	9B	147	Point to unipolar Gain frame
121	00	163	Point to bipolar offset frame
120	00	139	Point to unipolar offset frame
119	00	0	Not used
118	00	0	Not used
117	00	0	User 1 AI CALDAC0 value
116	00	0	User 1 AI CALDAC1 value
115	00	0	User 1 AI CALDAC2 value
114	00	0	User 1 AI CALDAC3 value
113	00	0	User 1 AO CALDAC4 value
112	00	0	User 1 AO CALDAC5 value
111	00	0	User 1 AO CALDAC6 value
110	00	0	User 1 AO CALDAC7 value
109	00	0	User 1 Gain 1 value-gain

(continues)

Table 4-3. DAQCard-1200 EEPROM Map (Continued)

Location	Hex	Decimal	Description
108	00	0	User 1 Gain 1.25 value-gain
107	00	0	User 1 Gain 2 value-gain
106	00	0	User 1 Gain 5 value-gain
105	00	0	User 1 Gain 10 value-gain
104	00	0	User 1 Gain 20 value-gain
103	00	0	User 1 Gain 50 value-gain
102	00	0	User 1 Gain 100 value-gain
101	00	0	User 1 Gain 1 value-offset
100	00	0	User 1 Gain 1.25 value-offset
99	00	0	User 1 Gain 2 value-offset
98	00	0	User 1 Gain 5 value-offset
97	00	0	User 1 Gain 10 value-offset
96	00	0	User 1 Gain 20 value-offset
95	00	0	User 1 Gain 50 value-offset
94	00	0	User 1 Gain 100 value-offset
93	00	0	Not used
92	00	0	Not used
91	00	0	User 2 AI CALDAC0 value
90	00	0	User 2 AI CALDAC1 value
89	00	0	User 2 AI CALDAC2 value
88	00	0	User 2 AI CALDAC3 value
87	00	0	User 2 AO CALDAC4 value
86	00	0	User 2 AO CALDAC5 value
85	00	0	User 2 AO CALDAC6 value
84	00	0	User 2 AO CALDAC7 value
83	00	0	User 2 Gain 1 value-gain
82	00	0	User 2 Gain 1.25 value-gain
81	00	0	User 2 Gain 2 value-gain
80	00	0	User 2 Gain 5 value-gain
79	00	0	User 2 Gain 10 value-gain
78	00	0	User 2 Gain 20 value-gain
77	00	0	User 2 Gain 50 value-gain
76	00	0	User 2 Gain 100 value-gain
75	00	0	User 2 Gain 1 value-offset
74	00	0	User 2 Gain 1.25 value-offset
73	00	0	User 2 Gain 2 value-offset
72	00	0	User 2 Gain 5 value-offset
71	00	0	User 2 Gain 10 value-offset
70	00	0	User 2 Gain 20 value-offset
69	00	0	User 2 Gain 50 value-offset
68	00	0	User 2 Gain 100 value-offset
67	00	0	Not used
66	00	0	Not used
65	00	0	User 3 AI CALDAC0 value
64	00	0	User 3 AI CALDAC1 value
63	00	0	User 3 AI CALDAC2 value
62	00	0	User 3 AI CALDAC3 value

(continues)

Table 4-3. DAQCard-1200 EEPROM Map (Continued)

Location	Hex	Decimal	Description
61	00	0	User 3 AO CALDAC4 value
60	00	0	User 3 AO CALDAC5 value
59	00	0	User 3 AO CALDAC6 value
58	00	0	User 3 AO CALDAC7 value
57	00	0	User 3 Gain 1 value-gain
56	00	0	User 3 Gain 1.25 value-gain
55	00	0	User 3 Gain 2 value-gain
54	00	0	User 3 Gain 5 value-gain
53	00	0	User 3 Gain 10 value-gain
52	00	0	User 3 Gain 20 value-gain
51	00	0	User 3 Gain 50 value-gain
50	00	0	User 3 Gain 100 value-gain
49	00	0	User 3 Gain 1 value-offset
48	00	0	User 3 Gain 1.25 value-offset
47	00	0	User 3 Gain 2 value-offset
46	00	0	User 3 Gain 5 value-offset
45	00	0	User 3 Gain 10 value-offset
44	00	0	User 3 Gain 20 value-offset
43	00	0	User 3 Gain 50 value-offset
42	00	0	User 3 Gain 100 value-offset
41	00	0	Not used
40	00	0	Not used
39	00	0	User 4 AI CALDAC0 value
38	00	0	User 4 AI CALDAC1 value
37	00	0	User 4 AI CALDAC2 value
36	00	0	User 4 AI CALDAC3 value
35	00	0	User 4 AO CALDAC4 value
34	00	0	User 4 AO CALDAC5 value
33	00	0	User 4 AO CALDAC6 value
32	00	0	User 4 AO CALDAC7 value
31	00	0	User 4 Gain 1 value-gain
30	00	0	User 4 Gain 1.25 value-gain
29	00	0	User 4 Gain 2 value-gain
28	00	0	User 4 Gain 5 value-gain
27	00	0	User 4 Gain 10 value-gain
26	00	0	User 4 Gain 20 value-gain
25	00	0	User 4 Gain 50 value-gain
24	00	0	User 4 Gain 100 value-gain
23	00	0	User 4 Gain 1 value-offset
22	00	0	User 4 Gain 1.25 value-offset
21	00	0	User 4 Gain 2 value-offset
20	00	0	User 4 Gain 5 value-offset
19	00	0	User 4 Gain 10 value-offset
18	00	0	User 4 Gain 20 value-offset
17	00	0	User 4 Gain 50 value-offset
16	00	0	User 4 Gain 100 value-offset
15	00	0	Not used

(continues)

Table 4-3. DAQCard-1200 EEPROM Map (Continued)

Location	Hex	Decimal	Description
14	00	0	Not used
13	00	0	Not used
12	00	0	Not used
11	00	0	Not used
10	00	0	Not used
9	00	0	Not used
8	00	0	Not used
7	00	0	Not used
6	00	0	Not used
5	00	0	Not used
4	00	0	Not used
3	00	0	Not used
2	00	0	Not used
1	00	0	Not used
0	00	0	Not used

Appendix A
Card Information Structure (CIS)

This appendix lists the CIS for the DAQCard-1200. The semicolons indicate comments, the *X* indicates hexadecimal, and *A* indicates ASCII. The bytes are arranged contiguously starting from location 0X00 upwards and are accessible at even addresses only.

```
; DAQCard-1200 CIS source file
; National Instruments Inc.
;
; Compile this file to an absolute binary file
;
; For more information on the Card Information Structure,
; see the Metaformat section of the PCMCIA Standard 2.01
;
; The TPL_LINK field should be the number of bytes in the
; current tuple which follow that one

; Device information tuple

X 01 ; CISTPL_DEVICE
X 03 ; TPL_LINK
X 00 ; Device ID=(No device)
X 00 ; DSPEED_NULL (Device speed = unspecified)
X FF ; End marker for this tuple

; Level 1 version/product information tuple

X 15 ; CISTPL_VERS_1
X 29 ; TPL_LINK
X 04 ; TPLLV1_MAJOR (Version number)
X 01 ; TPLLV1_MINOR (Release 2.01)
A National Instruments
X 00
A DAQCard-1200
X 00 ; Revision B
A B
X 00 ; This section is unused instead of a zero, could put any string
A 0
X 00
X FF ; End marker for this tuple

; Configuration tuple
```

```
X 1A ; CISTPL_CONFIG
X 05 ; TPL_LINK
X 01 ; TPCC_SZ: TPCC_RFSZ=0 (Reserved); TPCC_RMSZ=0 (1 reg mask field);
;           TPCC_RASZ=1 (2 bytes for base address)
X 01 ; TPCC_LAST=1 (Index of last entry in card configuration table)
X 00 ; TPCC_RADR=$0100 (Configuration registers base address)
X 01
X 03 ; TPCC_RMSK (Only registers 0 and 1 are implemented)

; Card configuration table entry tuple

X 1B ; CISTPL_CFTABLE_ENTRY
X 0A ; TPL_LINK
X C1 ; TPCE_INDX: ConfigEntry=1; Default=1=True; Interface=1=(TPCE_IF follows)
X 01 ; TPCE_IF: Interface=1=I/O; BVDs,WP,RdyBsy=0=inactive; Wait=0=NotRequired
X 99 ; TPCE_FS: Power=1=(Vcc specified); Timing=0=default; I/O Space=1=required
;           IRQ=1=required; MemSpace=0=none; Misc=1=present
X 01 ; TPCE_PD: Only nominal Vcc value is specified (may provide more later)
X 55 ;        Vcc= 5 * 1V = 5V
X 45 ; TPCE_IO: I/OAddressLines=5; Bus8=1; Bus16=0; Range=0
X 50 ; TPCE_IR: Share=0=No; Pulse=1=T; Level=1=F; Mask=1=present; OtherInts=0
X FF ;        IRQ7-0 enabled
X FF ;        IRQ15-8 enabled
X 20 ; TPCE_MI: MaxTwins=0; Audio=0; ReadOnly=0; PowerDown=1=T; Ext=0

; Manufacturer Identification Tuple

X 20 ; CISTPL_MANFID
X 04 ; TPL_LINK
X 0B ; TPLMID_MANF=$010B (PCMCIA PC Card manufacturer code)
X 01
X 03 ; TPLMID_CARD=$0103 (NI ID number)
X 01

; End of list tuple

X FF ; CSTPL_END
```

Appendix B
Fujitsu MB88341/MB88342 Data Sheet*

This appendix contains the manufacturer data sheet for the MB88341/MB88342 R-2R type 8-bit D/A converter manufactured by Fujitsu Microelectronics, Inc. The MB88341 D/A converter is used on the DAQCard-1200.

* Copyright © Fujitsu Microelectronics, Inc. Reprinted with permission of copyright owner.
All rights reserved.
Fujitsu Microelectronics, Inc. 1990 *Linear Products 1990 Data Book*.

October 1989
Edition 1.0

FUJITSU

DATA SHEET

MB88341/MB88342

R-2R TYPE 8-BIT D/A CONVERTER

DESCRIPTION

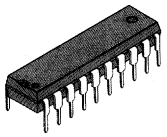
The Fujitsu MB88341 and MB88342 are R–2R type 8–bit resolution digital–to–analog converters (DAC), designed for interface with a wide range of general 4–bit and 8–bit microcomputers including Fujitsu's MB8840/50 series and MB88400/500 series 4–bit single–chip microcomputers.

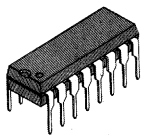
The MB88341 has an 8–bit x 12–channel D/A converter and the MB88342 has an 8–bit x 8–channel D/A converter. Digital data are input serially by individual channel units. The loaded digital data are convertered into analog DC voltages by the D/A converter in 60 µs settling time. The MB88341 and MB88342 are suitable for electronic volumes and replacement for potentiometers for adjustment, in addition to normal D/A converter applications.

FEATURES

- Conversion method : R–2R resistor ladder
- MB88341 : 8–bit x 12–channel D/A converter
- MB88342 : 8–bit x 8–channel D/A converter
- Serial data input
- Serial data output for cascade connection
- 60 µs DAC output settling time
- Two separate power supply/ground lines for digital and analog blocks.
- Single +5V power supply
- Wide operating temperature range: –20°C to +85°C
- Silicon–gate CMOS process
- Three package options :
  - –MB88341: 20–pin plastic DIP (Suffix : –P), 20–pin plastic SOP (Suffix : –PF), 20–pin plastic SSOP (Suffix : –PFV)
  - –MB88342: 16–pin plastic DIP (Suffix : –P), 16–pin plastic SOP (Suffix : –PF), 20–pin plastic SSOP (Suffix : –PFV)

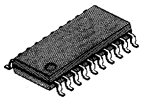
MB88341–P **MB88342–P**

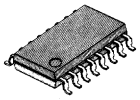




PLASTIC DIP (DIP–20P–M02) **PLASTIC DIP (DIP–16P–M04)**

MB88341–PF **MB88342–PF**





PLASTIC SOP (FPT–20P–M01) **PLASTIC SOP (FPT–16P–M02)**

MB88341–PFV/MB88342–PFV

PLASTIC SSOP (FPT–20P–M03)

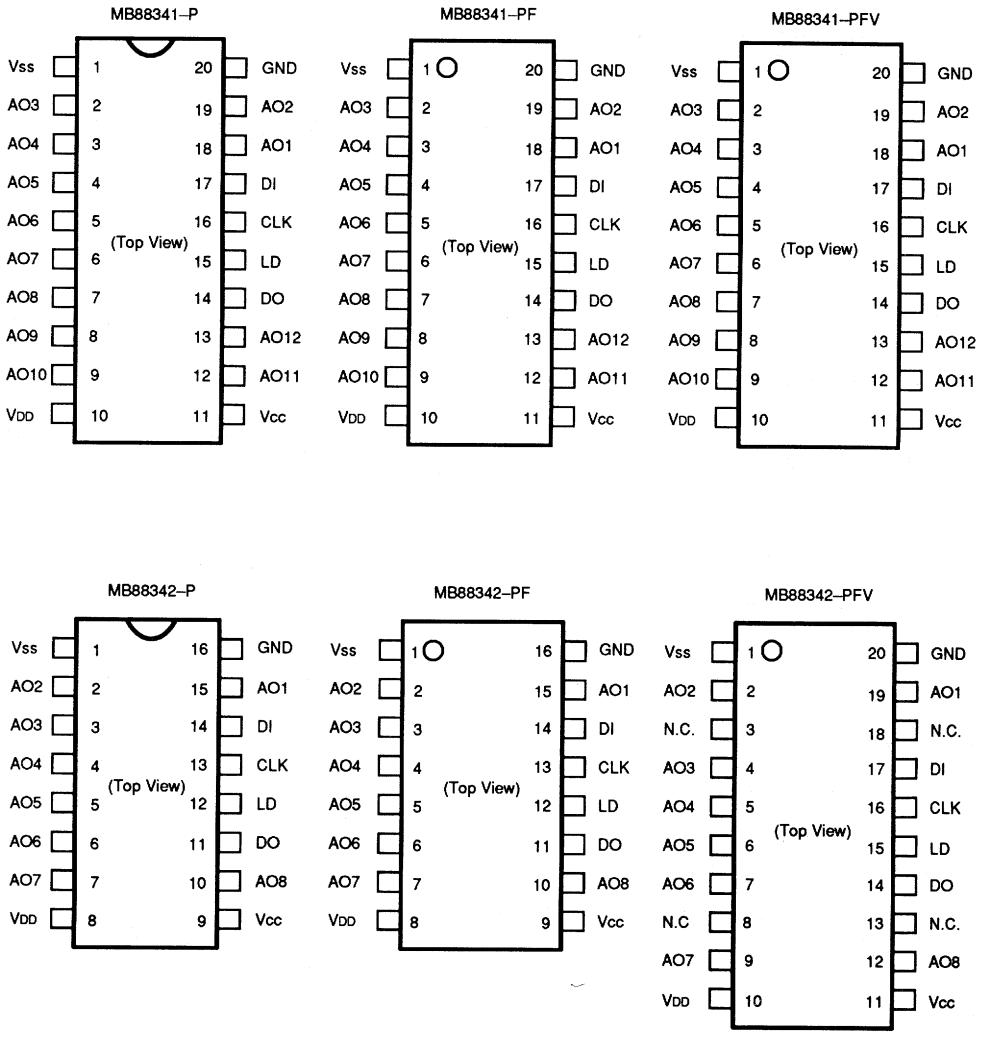
This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields. However, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high impedance circuit.

7

Copyright© 1989 FUJITSU LIMITED

MB88341
MB88342

Figure 1 Pin Assignment



7

 © National Instruments Corporation

Figure 2 Logic Symbol

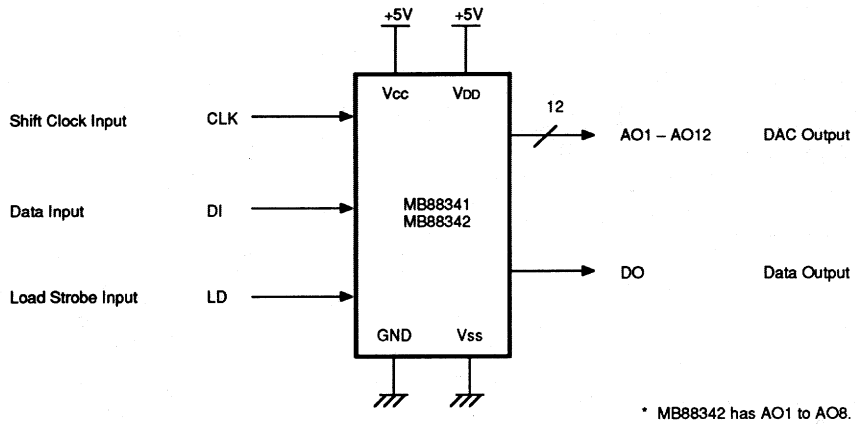
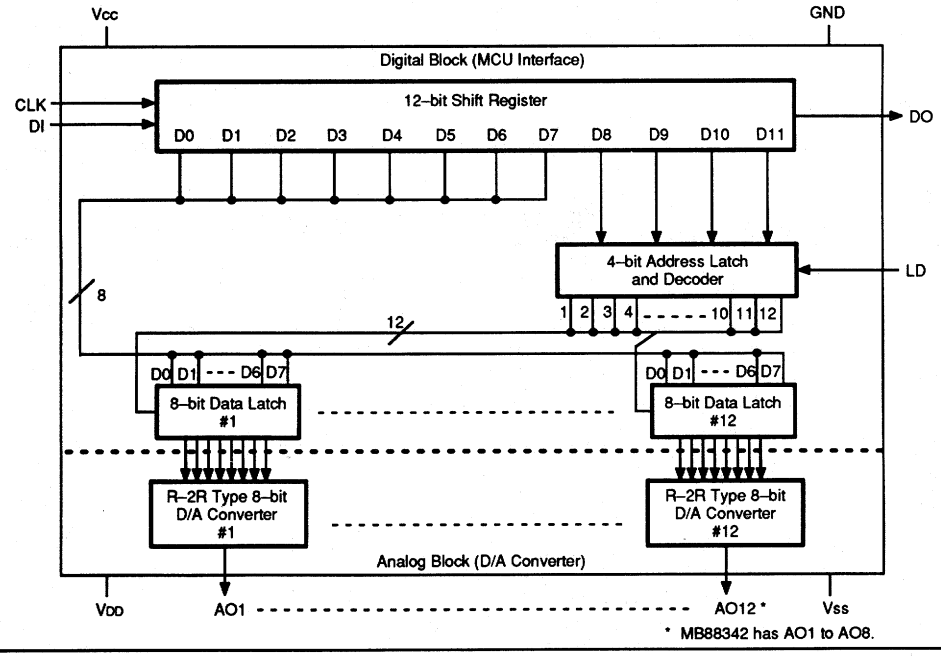


Figure 3 Block Diagram



7

MB88341
MB88342

PIN DESCRIPTION

Figure 1 and Table 1 show the pin assignment and pin description of the MB88341 and MB88342.

Table 1 Pin Description

Symbol	Pin No. MB88341	Pin No. MB88342	Type	Name & Function
Power Supply				
Vcc	11	9 (11)	–	+5V DC power supply pin for the digital block (MCU interface).
GND	20	16 (20)	–	Ground pin for the digital block (MCU interface).
VDD	10	8 (10)	–	+5V DC power supply pin for the analog block (D/A converter).
Vss	1	1 (1)	–	Ground pin for the analog block (D/A converter).
Control Input				
CLK	16	13 (16)	I	Shift clock input to the internal 12–bit shift register: At the rising edge of CLK data on the DI pin is shifted into the LSB of the shift register and contents of the shift register are shifted right (to the MSB).
LD	15	12 (15)	I	Load strobe input for a 12–bit address/data : A high level on the LD pin latches a 4–bit address (upper 4 bits: D11 to D8) of the internal 12–bit shift register into the internal address latch/decoder, and writes 8–bit data (lower 8 bits: D7 to D0) of the shift register into an internal data latch selected by the latched address.
Data Input/Output				
DI	17	14 (17)	I	Serial address/data input to the internal 12–bit shift register: The address/data format is that upper 4 bits (D11 to D8) indicate an address and lower 8 bits (D7 to D0) indicate data. The D11 (MSB) is the first–in bit and D0 (LSB) is the last–in bit.
DO	14	11 (14)	O	Serial address/data output from the internal 12–bit shift register: This is an output pin of the MSB bit data of the 12–bit shift register. This pin allows a cascade connection of the device.
DAC Output				
AO1	18	15 (19)	O	8–bit resolution D/A converter outputs : MB88341: 12 channels (AO1 to AO12) MB88342: 8 channels (AO1 to AO8)
AO2	19	2 (2)		
AO3	2	3 (4)		
AO4	3	4 (5)		
AO5	4	5 (6)		
AO6	5	6 (7)		
AO7	6	7 (9)		
AO8	7	10 (12)		
AO9	8	– (–)		
AO10	9	– (–)		
AO11	12	– (–)		
AO12	13	– (–)		

Note : Pin numbers in parentheses are applied to MB88342–PFV.

7

 © National Instruments Corporation

FUNCTIONAL DESCRIPTION

OVERVIEW

The MB88341 and MB88342 are R–2R resistor ladder type, 8–bit resolution digital–to–analog converter (DAC) devices. The MB88341 has 12 channels, and MB88342 has 8 channels of D/A converters. 8–bit digital data are loaded into internal data latches by individual DAC channel units. The loaded digital data are converted into analog DC voltages through the internal D/A converter in 60 µs settling time. For cascade connection, a serial data output is provided.

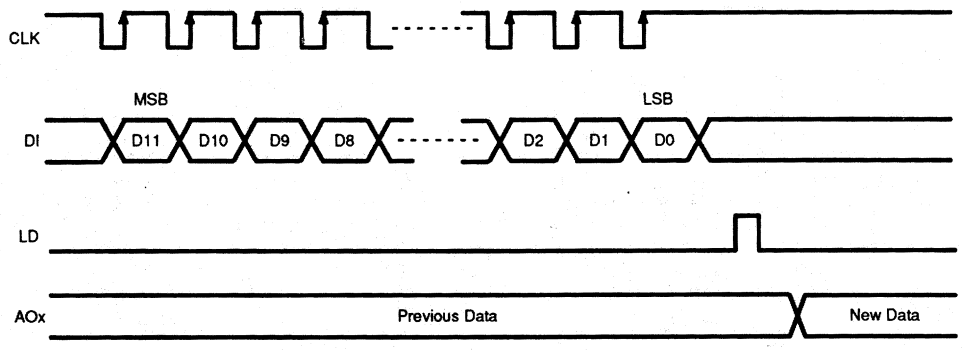
DEVICE CONFIGURATION

As illustrated in Figure 3 block diagram, the MB88341 (MB88342) device is composed by the digital block (MCU interface) and analog block (D/A converter). The digital block consists of a 12–bit shift register, a 4–bit address latch/decoder, and 12 (8) 8–bit data latches. The analog block includes 12 (8) 8–bit D/A converters connecting to the data latches. For electrically stable operation the power supply and ground lines are separate between the digital block (for MCU interface) and analog block (for D/A converter).

DEVICE OPERATION

Figure 4 shows the input/output timing. A 12–bit address/data is serially input into the shift register through the DI pin synchronously with the rising edge of CLK. The format of the shift register is shown in Figure 5. The lower 8 bits (D7 to D0) are data bits to be converted, and the upper 4 bits are address bits (D11 to D8) to select a data latch to be written. A high level on the LD pin loads the address-latch/decoder with the 4–bit address to select a data latch, and writes the 8–bit data into a selected data latch. Figure 6 shows the data latch address map, and Table 2, address decoding. 8–bit data written into individual data latches are converted into analog DC voltages, dividing the supply voltage |VDD–Vss| through R–2R resistor ladders of D/A converters. Figure 7 shows the R–2R resistor ladder D/A converter configuration, and Table 3 analog DC voltages corresponding to each digital data.

Figure 4 Input/Output Timing



7

MB88341
MB88342

Figure 5 Shift Register Format

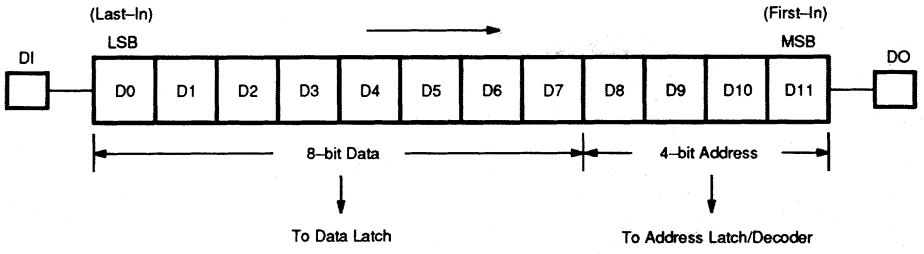
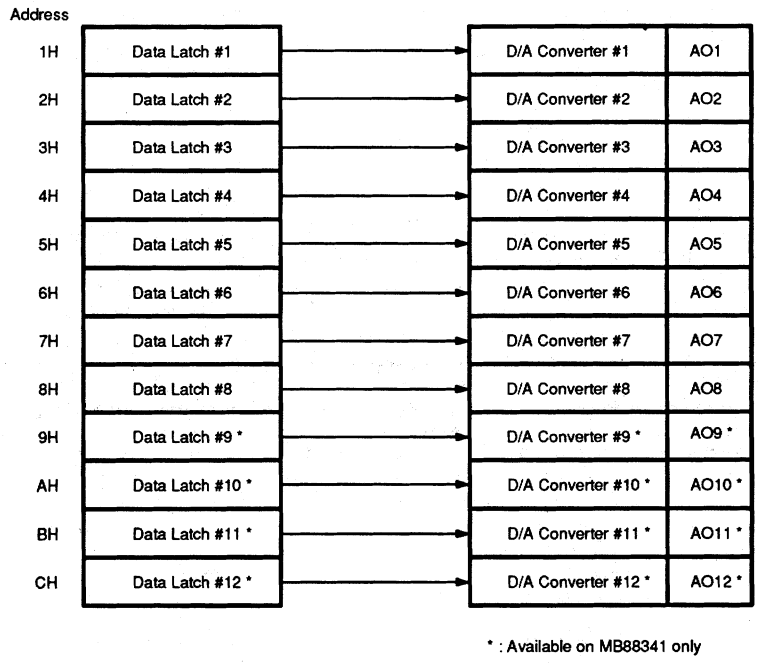


Figure 6 Data Latch Address Map

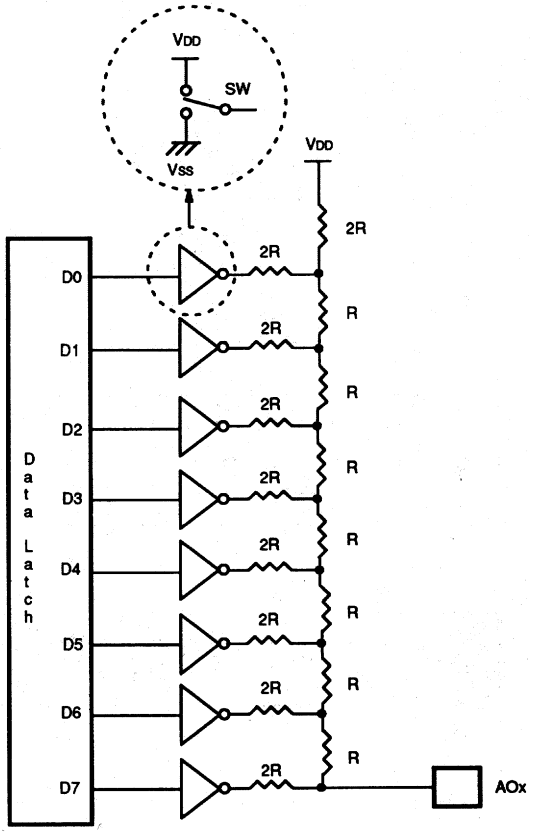


* : Available on MB88341 only

7

 © National Instruments Corporation

Figure 7 R–2R Resistor Ladder D/A Converter Configuration



7

MB88341
MB88342

Table 2 Address Decoding

Address				Data Latch Selected	
D8	D9	D10	D11	MB88341	MB88342
0	0	0	0	Deselected	
0	0	0	1	Data Latch #1	
0	0	1	0	Data Latch #2	
0	0	1	1	Data Latch #3	
0	1	0	0	Data Latch #4	
0	1	0	1	Data Latch #5	
0	1	1	0	Data Latch #6	
0	1	1	1	Data Latch #7	
1	0	0	0	Data Latch #8	
1	0	0	1	Data Latch #9	Deselected
1	0	1	0	Data Latch #10	Deselected
1	0	1	1	Data Latch #11	Deselected
1	1	0	0	Data Latch #12	Deselected
1	1	0	1	Deselected	
1	1	1	0	Deselected	
1	1	1	1	Deselected	

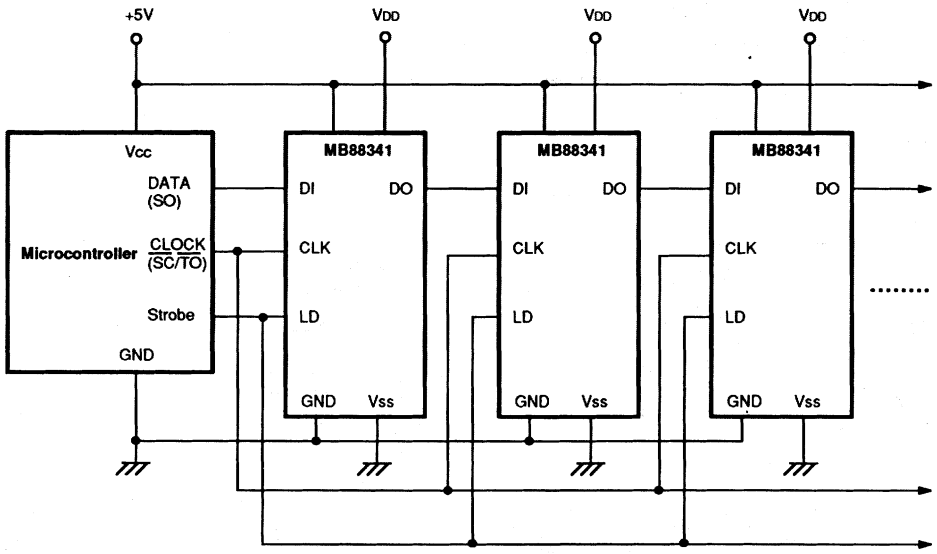
7

Table 3 Data Conversion

Data								DAC Output Level
D7	D6	D5	D4	D3	D2	D1	D0	AOx
0	0	0	0	0	0	0	0	≈ |VDD – Vss| x 1/256
0	0	0	0	0	0	0	1	≈ |VDD – Vss| x 2/256
0	0	0	0	0	0	1	0	≈ |VDD – Vss| x 3/256
0	0	0	0	0	0	1	1	≈ |VDD – Vss| x 4/256
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	1	1	1	1	1	0	≈ |VDD – Vss| x 255/256
1	1	1	1	1	1	1	1	≈ |VDD – Vss|

 © National Instruments Corporation

Figure 8 Cascade Connection Example



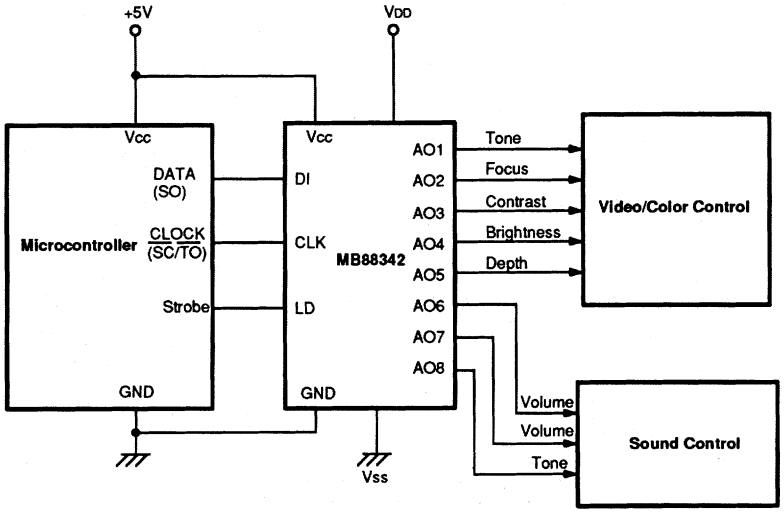
7

MB88341
MB88342

APPLICATION DESCRIPTION

The MB88341 and MB88342 are suitable for electronic volumes and replacement for adjustment potentiometers, in addition to normal D/A converter applications. Figure 8 illustrates an application example for TV set.

Figure 9 Application Example (for TV Set)



7

 © National Instruments Corporation

ELECTRICAL CHARACTERISTICS

ABSOLUTE MAXIMUM RATINGS †

Parameter	Symbol	Rating			Unit	Condition
		Min	Typ	Max		
Supply Voltage	Vcc	–0.3		7.0	V	Ta = 25°C GND = 0 V
	VDD	–0.3		7.0	V	
Input Voltage	VIN	–0.3		7.0	V	Ta = 25°C GND = 0 V Should not exceed Vcc + 0.3V
Output Voltage	VOUT	–0.3		7.0	V	
Power Dissipation	PD			250	mW	
Operating Ambient Temperature	TA	–20		+85	°C	
Storage Temperature	TSTG	–55		+150	°C	

† Permanent device damage may occur if the above ABSOLUTE MAXIMUM RATINGS are exceeded. Functional operation should be restricted to the conditions as detailed in the operational sections of this data sheet. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

RECOMMENDED OPERATING CONDITIONS

Parameter	Symbol	Value			Unit	Condition
		Min	Typ	Max		
Supply Voltage (for Digital Block)	Vcc	4.5	5.0	5.5	V	
	GND		0		V	
Supply Voltage (for Analog Block)	VDD	3.0		Vcc	V	VDD ≤ Vcc, Monotonicity, No load
	Vss	0		1.0	V	
Operating Ambient Temperature	TA	–20		+85	°C	

7

MB88341
MB88342

DC CHARACTERISTICS

(Recommended operating conditions unless otherwise noted.)

Digital Block (MCU Interface)

Parameter	Symbol	Value			Unit	Condition
		Min	Typ.	Max		
Active Supply Current	Icc			1.0	mA	CLK = 1MHz
Standby Supply Current	Iccs			10	µA	All inputs (including CLK) fixed at Vcc or GND. All outputs open.
Input Leakage Current	I_{ILK}	–10		10	µA	V_{IN} = 0 to Vcc
Input Low Voltage	V_{IL}			0.2•Vcc	V	
Input High Voltage	V_{IH}	0.8•Vcc			V	
Output Low Voltage	V_{OL}			0.4	V	I_{OL} = 2.5 mA
Output High Voltage	V_{OH}	Vcc–0.4			V	I_{OH} = –400 µA

Analog Block (D/A Converter)

Parameter	Symbol	Value			Unit	Condition	
		Min	Typ	Max			
Supply Current	I_{DD}		1.5	3.0	mA	MB88341	No load
			1.2	2.5	mA	MB88342	
Resolution		8			bit	Monotonicity, I_{OUT} = –0.01 µA	
Variation of Linearity among Channels				±3	LSB	Monotonicity, No load	

7

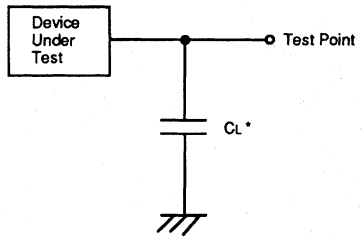
 © National Instruments Corporation

AC CHARACTERISTICS

(Recommended operating conditions unless otherwise noted.)

Parameter	Symbol	Value		Unit	Condition
		Min	Max		
Clock Low Time	t_{CKL}	200		ns	
Clock High Time	t_{CKH}	200		ns	
Clock Rise Time	t_{Cr}		200	ns	
Clock Fall Time	t_{Cf}		200	ns	
Data Setup Time	t_{DCH}	30		ns	
Data Hold Time	t_{CHD}	60		ns	
Load Strobe High Time	t_{LDH}	100		ns	
Load Strobe Setup Time	t_{CHL}	200		ns	
Load Strobe Hold Time	t_{LDC}	100		ns	
DAC Output Settling Time	t_{LDD}		60	µs	No load
Data Output Delay Time	t_{DO}	70	350	ns	*C_L = 20 pF (Min.), 100 pF (Max.)

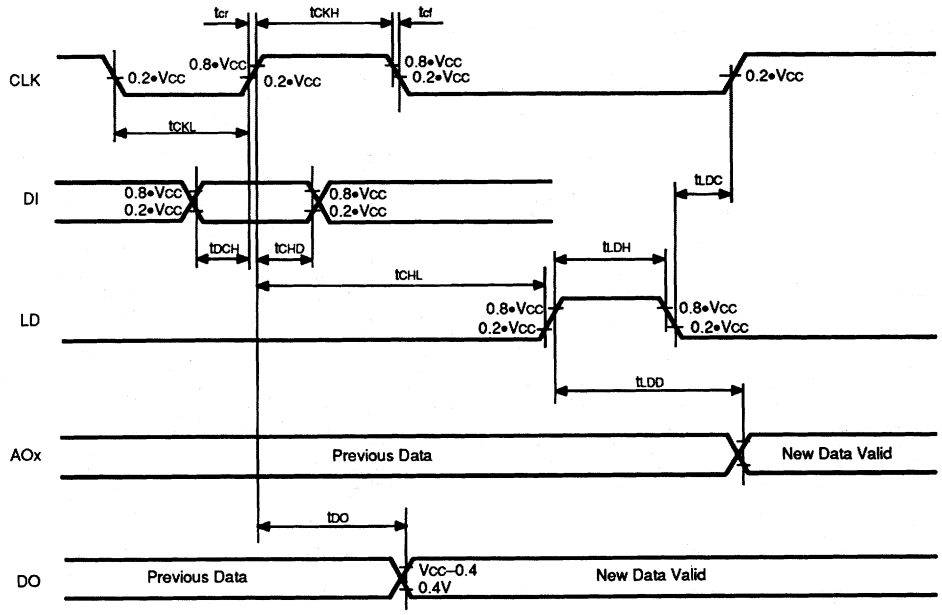
Figure 10 AC Test Conditions



7

MB88341
MB88342

Figure 11 Input/Output Timing



7

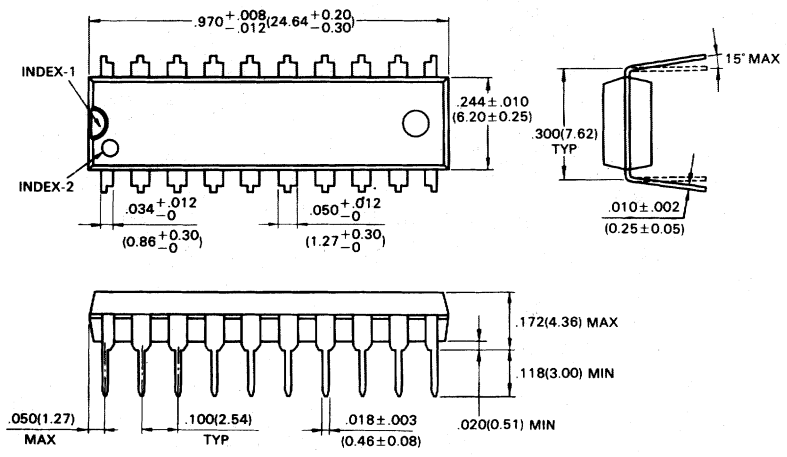
 © National Instruments Corporation

MB88341
MB88342

PACKAGE DIMENSIONS

MB88341–P

20-LEAD PLASTIC DUAL IN-LINE PACKAGE
(Case No. : DIP-20P-M02)



©1988 FUJITSU LIMITED D20003S-3C

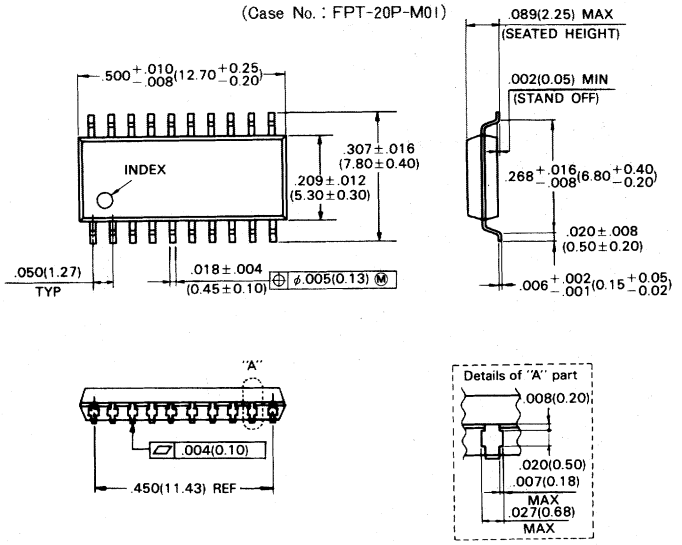
Dimensions in
inches (millimeters)

7

MB88341
MB88342

MB88341–PF

20-LEAD PLASTIC FLAT PACKAGE
(Case No. : FPT-20P-M01)



Dimensions in inches (millimeters)

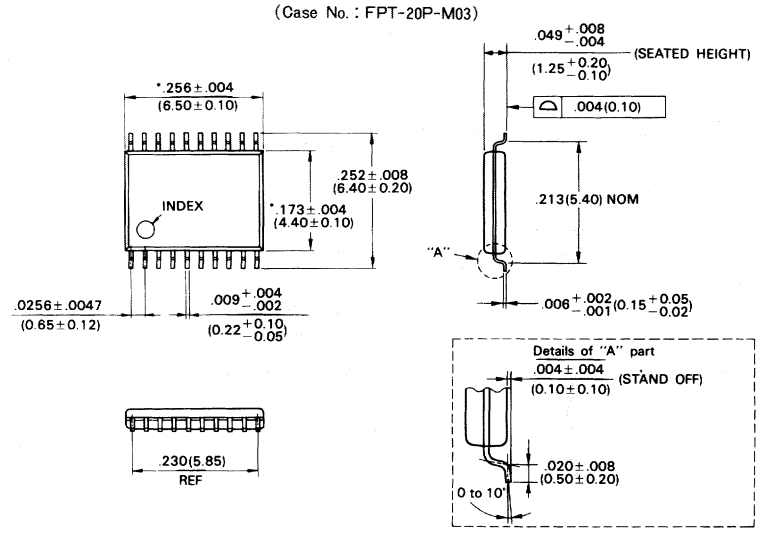
©1988 FUJITSU LIMITED F20003S-4C

7

MB88341
MB88342

MB88341–PFV

20-LEAD PLASTIC FLAT PACKAGE
(Case No. : FPT-20P-M03)



©1989 FUJITSU LIMITED F20012S-2C

* : This dimension does not include resin protrusion.

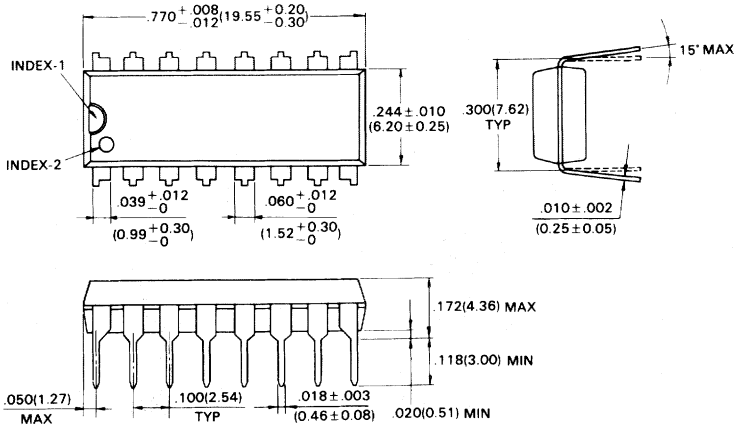
Dimensions in inches (millimeters)

7

MB88341
MB88342

MB88342–P

16-LEAD PLASTIC DUAL IN-LINE PACKAGE
(Case No. : DIP-16P-M04)



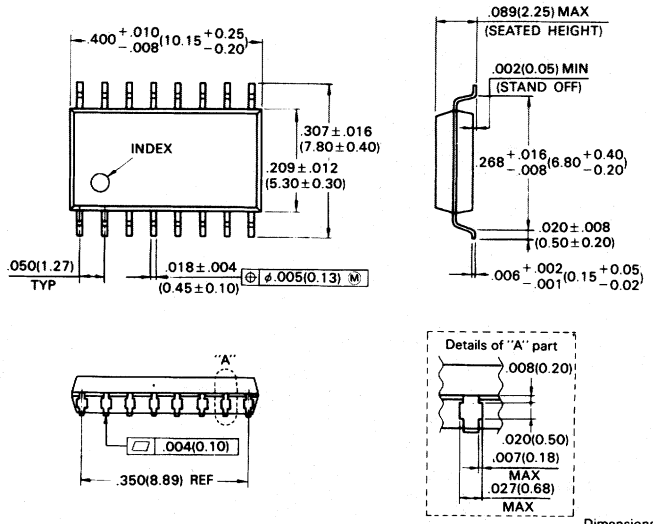
© 1988 FUJITSU LIMITED D16033S-2C

Dimensions in
inches (millimeters)

7

MB88342–PF

16-LEAD PLASTIC FLAT PACKAGE
(Case No. : FPT-16P-M02)



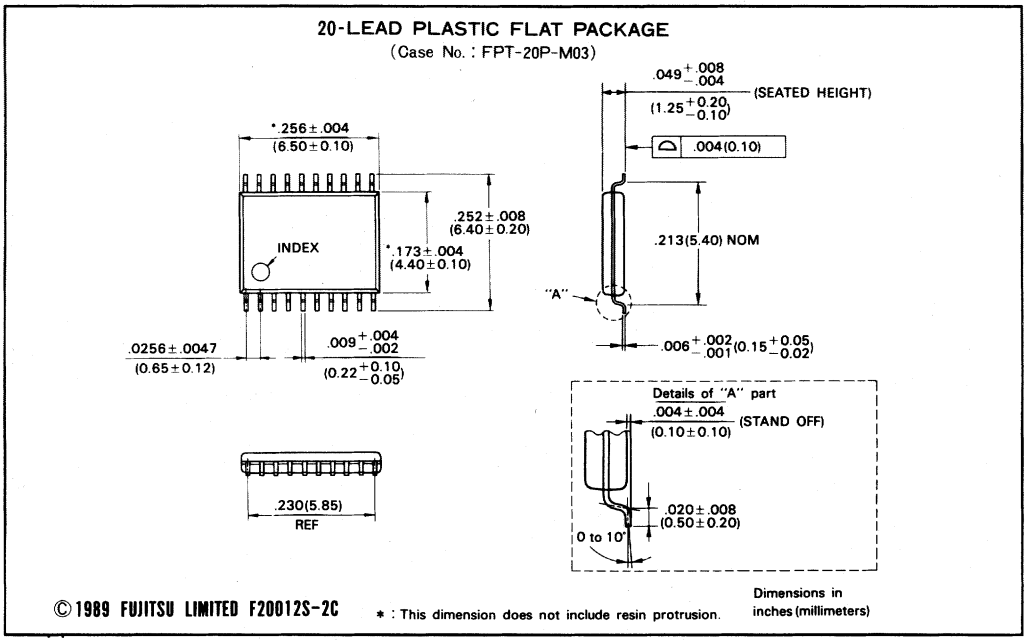
©1988 FUJITSU LIMITED F16005S-4C

Dimensions in inches (millimeters)

7

MB88341
MB88342

MB88342–PFV



7

Appendix C
Xicor X25020 Data Sheet*

This appendix contains the manufacturer data sheet for the X25020 SPI serial EEPROM manufactured by Xicor, Inc. This EEPROM is used on the DAQCard-1200.

* Copyright © Xicor, Inc. Reprinted with permission of copyright owner.
All rights reserved.
Xicor Inc. 1995 *Nonvolatile Solutions Data Book*.

APPLICATION NOTES
AVAILABLE
AN9 • AN18 • AN31 • AN37 • AN40

2K X25020 256 x 8 Bit

SPI Serial E²PROM With BLOCK LOCK™ PROTECTION

FEATURES

- 1MHz Clock Rate
- SPI Modes (0,0 & 1,1)
- 256 X 8 Bits
  - —4 Byte Page Mode
- Low Power CMOS
  - —150µA Standby Current
  - —3mA Active Write Current
- 2.7V To 5.5V Power Supply
- Block Lock Protection
  - —Protect 1/4, 1/2 or all of E²PROM Array
- Built-in Inadvertent Write Protection
  - —Power-Up/Power-Down protection circuitry
  - —Write Latch
  - —Write Protect Pin
- Self-Timed Write Cycle
  - —5ms Write Cycle Time (Typical)
- High Reliability
  - —Endurance: 100,000 cycles per byte
  - —Data Retention: 100 Years
  - —ESD protection: 2000V on all pins
- 8-Pin Mini-DIP Package
- 8-Lead SOIC Package
- 8-Lead TSSOP

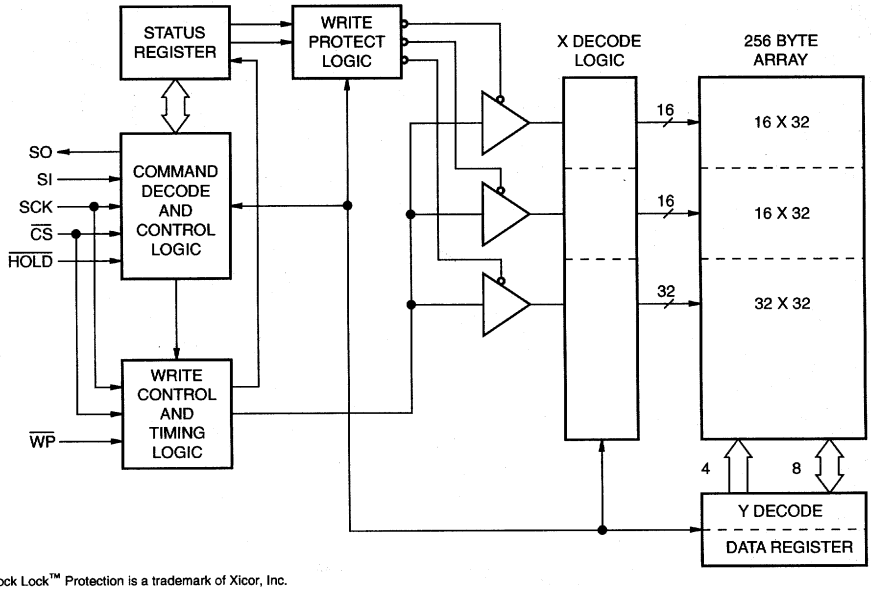
DESCRIPTION

The X25020 is a CMOS 2048-bit serial E²PROM, internally organized as 256 x 8. The X25020 features a serial interface and software protocol allowing operation on a simple three-wire bus. The bus signals are a clock input (SCK) plus separate data in (SI) and data out (SO) lines. Access to the device is controlled through a chip select ($\overline{CS}$) input, allowing any number of devices to share the same bus.

The X25020 also features two additional inputs that provide the end user with added flexibility. By asserting the $\overline{HOLD}$ input, the X25020 will ignore transitions on its inputs, thus allowing the host to service higher priority interrupts. The $\overline{WP}$ input can be used as a hardwire input to the X25020 disabling all write attempts, thus providing a mechanism for limiting end user capability of altering the memory.

The X25020 utilizes Xicor's proprietary Direct Write™ cell, providing a minimum endurance of 100,000 cycles per byte and a minimum data retention of 100 years.

FUNCTIONAL DIAGRAM



3834 FHD F01

Direct Write™ and Block Lock™ Protection is a trademark of Xicor, Inc.

©Xicor, Inc. 1994, 1995 Patents Pending
3834-1.4 5/26/95 T17/C6/D9 TD

Characteristics subject to change without notice

X25020

PIN DESCRIPTIONS

Serial Output (SO)

SO is a push/pull serial data output pin. During a read cycle, data is shifted out on this pin. Data is clocked out by the falling edge of the serial clock.

Serial Input (SI)

SI is the serial data input pin. All opcodes, byte addresses, and data to be written to the memory are input on this pin. Data is latched by the rising edge of the serial clock.

Serial Clock (SCK)

The Serial Clock controls the serial bus timing for data input and output. Opcodes, addresses, or data present on the SI pin are latched on the rising edge of the clock input, while data on the SO pin change after the falling edge of the clock input.

Chip Select ($\overline{CS}$)

When $\overline{CS}$ is HIGH, the X25020 is deselected and the SO output pin is at high impedance and unless an internal write operation is underway, the X25020 will be in the standby power mode. $\overline{CS}$ LOW enables the X25020, placing it in the active power mode. It should be noted that after power-up, a HIGH to LOW transition on $\overline{CS}$ is required prior to the start of any operation.

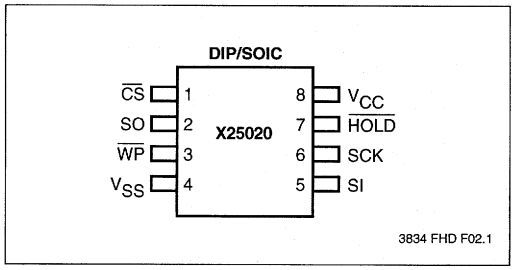
Write Protect ($\overline{WP}$)

When $\overline{WP}$ is LOW, nonvolatile writes to the X25020 are disabled, but the part otherwise functions normally. When $\overline{WP}$ is held HIGH, all functions, including nonvolatile writes operate normally. $\overline{WP}$ going LOW while $\overline{CS}$ is still LOW will interrupt a write to the X25020. If the internal write cycle has already been initiated, $\overline{WP}$ going LOW will have no affect on a write.

Hold ($\overline{HOLD}$)

$\overline{HOLD}$ is used in conjunction with the $\overline{CS}$ pin to select the device. Once the part is selected and a serial sequence is underway, $\overline{HOLD}$ may be used to pause the serial communication with the controller without resetting the serial sequence. To pause, $\overline{HOLD}$ must be brought LOW while SCK is LOW. To resume communication, $\overline{HOLD}$ is brought HIGH, again while SCK is LOW. If the pause feature is not used, $\overline{HOLD}$ should be held HIGH at all times.

PIN CONFIGURATION



PIN NAMES

Symbol	Description
$\overline{CS}$	Chip Select Input
SO	Serial Output
SI	Serial Input
SCK	Serial Clock Input
$\overline{WP}$	Write Protect Input
V_{SS}	Ground
V_{CC}	Supply Voltage
$\overline{HOLD}$	Hold Input

3834 PGM T01.1

X25020

PRINCIPLES OF OPERATION

The X25020 is a 256 x 8 E^2PROM designed to interface directly with the synchronous serial peripheral interface (SPI) of many popular microcontroller families.

The X25020 contains an 8-bit instruction register. It is accessed via the SI input, with data being clocked in on the rising SCK. $\overline{CS}$ must be LOW and the $\overline{HOLD}$ and $\overline{WP}$ inputs must be HIGH during the entire operation.

Table 1 contains a list of the instructions and their opcodes. All instructions, addresses and data are transferred MSB first.

Data input is sampled on the first rising edge of SCK after $\overline{CS}$ goes LOW. SCK is static, allowing the user to stop the clock and then resume operations. If the clock line is shared with other peripheral devices on the SPI bus, the user can assert the $\overline{HOLD}$ input to place the X25020 into a "PAUSE" condition. After releasing $\overline{HOLD}$, the X25020 will resume operation from the point when $\overline{HOLD}$ was first asserted.

Write Enable Latch

The X25020 contains a "write enable" latch. This latch must be SET before a write operation will be completed internally. The WREN instruction will set the latch and the WRDI instruction will reset the latch. This latch is automatically reset upon a power-up condition and after the completion of a byte, page, or status register write cycle.

Status Register

The RDSR instruction provides access to the status register. The status register may be read at any time, even during a write cycle. The status register is formatted as follows:

7	6	5	4	3	2	1	0
X	X	X	X	BP1	BP0	WEL	WIP

3834 PGM T02

BP0 and BP1 are set by the WRSR instruction. WEL and WIP are read-only and automatically set by other operations.

The Write-In-Process (WIP) bit indicates whether the X25020 is busy with a write operation. When set to a "1", a write is in progress, when set to a "0", no write is in progress. During a write, all other bits are set to "1".

The Write Enable Latch (WEL) bit indicates the status of the "write enable" latch. When set to a "1", the latch is set, when set to a "0", the latch is reset.

The Block Protect (BP0 and BP1) bits are nonvolatile and allow the user to select one of four levels of protection. The X25020 is divided into four 512-bit segments. One, two, or all four of the segments may be protected. That is, the user may read the segments but will be unable to alter (write) data within the selected segments. The partitioning is controlled as illustrated below.

Status Register Bits		Array Addresses Protected
BP1	BP0	
0	0	None
0	1	$C0–$FF
1	0	$80–$FF
1	1	$00–$FF

3834 PGM T03

Table 1. Instruction Set

Instruction Name	Instruction Format*	Operation
WREN	0000 0110	Set the Write Enable Latch (Enable Write Operations)
WRDI	0000 0100	Reset the Write Enable Latch (Disable Write Operations)
RDSR	0000 0101	Read Status Register
WRSR	0000 0001	Write Status Register
READ	0000 0011	Read Data from Memory Array beginning at selected address
WRITE	0000 0010	Write Data to Memory Array beginning at Selected Address (1 to 32 Bytes)

3834 PGM T04

*Instructions are shown MSB in leftmost position. Instructions are transferred MSB first.

 © National Instruments Corporation

X25020

Clock and Data Timing

Data input on the SI line is latched on the rising edge of SCK. Data is output on the SO line by the falling edge of SCK.

Read Sequence

When reading from the E²PROM memory array, $\overline{CS}$ is first pulled LOW to select the device. The 8-bit READ instruction is transmitted to the X25020, followed by the 8-bit address. After the READ opcode and address are sent, the data stored in the memory at the selected address is shifted out on the SO line. The data stored in memory at the next address can be read sequentially by continuing to provide clock pulses. The address is automatically incremented to the next higher address after each byte of data is shifted out. When the highest address is reached ($FF) the address counter rolls over to address $00 allowing the read cycle to be continued indefinitely. The read operation is terminated by taking $\overline{CS}$ HIGH. Refer to the read E²PROM array operation sequence illustrated in Figure 1.

To read the status register $\overline{CS}$ line is first pulled LOW to select the device followed by the 8-bit RDSR instruction. After the read status register opcode is sent, the contents of the status register are shifted out on the SO line. Figure 2 illustrates the read status register sequence.

Write Sequence

Prior to any attempt to write data into the X25020 the "write enable" latch must first be set by issuing the WREN instruction (See Figure 3). $\overline{CS}$ is first taken LOW, then the WREN instruction is clocked into the X25020. After all eight bits of the instruction are transmitted, $\overline{CS}$ must then be taken HIGH. If the user continues the write operation without taking $\overline{CS}$ HIGH after issuing the WREN instruction, the write operation will be ignored.

To write data to the E²PROM memory array, the user issues the WRITE instruction, followed by the address and then the data to be written. This is minimally a thirty-two clock operation. $\overline{CS}$ must go LOW and remain LOW for the duration of the operation. The host may continue to write up to 4 bytes of data to the X25020. The only restriction is that the 4 bytes must reside on the same page. If the address counter reaches the end of the page and the clock continues, the counter will "roll over" to the first address of the page and overwrite any data that may have been written.

For the write operation (byte or page write) to be completed, $\overline{CS}$ can only be brought HIGH after bit 0 of data byte N is clocked in. If it is brought HIGH at any other time the write operation will not be completed. Refer to Figures 4 and 5 below for a detailed illustration of the write sequences and time frames in which $\overline{CS}$ going HIGH are valid.

To write to the status register, the WRSR instruction is followed by the data to be written. Data bits 0, 1, 4, 5, 6 and 7 must be "0". Figure 6 illustrates this sequence.

While the write is in progress following a status register or E²PROM write sequence, the status register may be read to check the WIP bit. During this time the WIP bit will be HIGH.

Hold Operation

The $\overline{HOLD}$ input should be HIGH (at V_{IH}) under normal operation. If a data transfer is to be interrupted $\overline{HOLD}$ can be pulled LOW to suspend the transfer until it can be resumed. The only restriction is the SCK input must be LOW when $\overline{HOLD}$ is first pulled LOW and SCK must also be LOW when $\overline{HOLD}$ is released.

The HOLD input may be tied HIGH either directly to V_{CC} or tied to V_{CC} through a resistor.

X25020

Operational Notes

The X25020 powers-up in the following state:

- The device is in the low power standby state.
- A HIGH to LOW transition on $\overline{CS}$ is required to enter an active state and receive an instruction.
- SO pin is high impedance.
- The "write enable" latch is reset.

Data Protection

The following circuitry has been included to prevent inadvertent writes:

- The "write enable" latch is reset upon power-up.
- A WREN instruction must be issued to set the "write enable" latch.
- $\overline{CS}$ must come HIGH at the proper clock count in order to start a write cycle.

2

Figure 1. Read E²PROM Array Operation Sequence

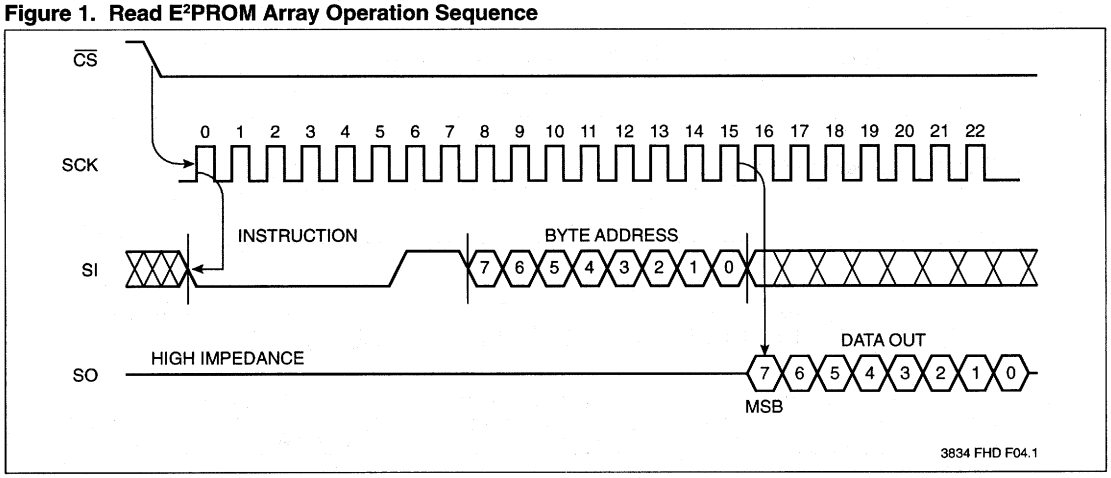
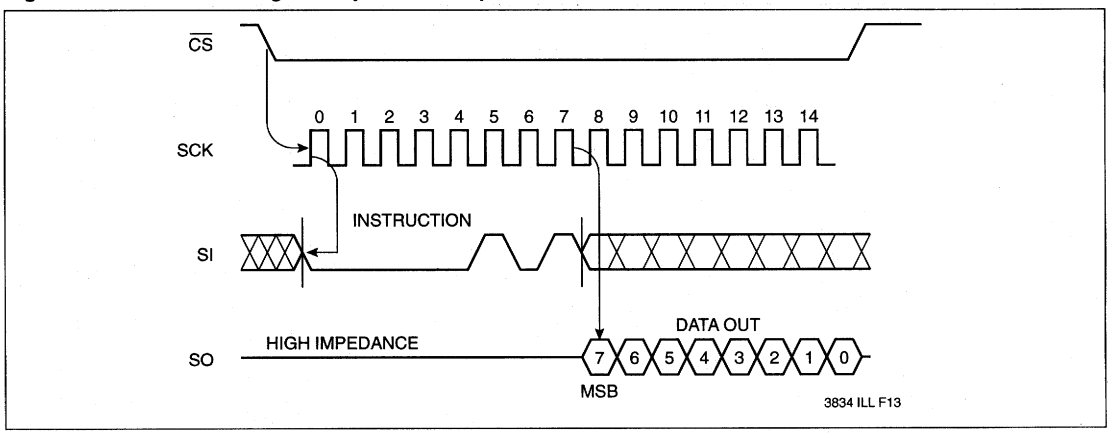


Figure 2. Read Status Register Operation Sequence



 © National Instruments Corporation

X25020

Figure 3. Write Enable Latch Sequence

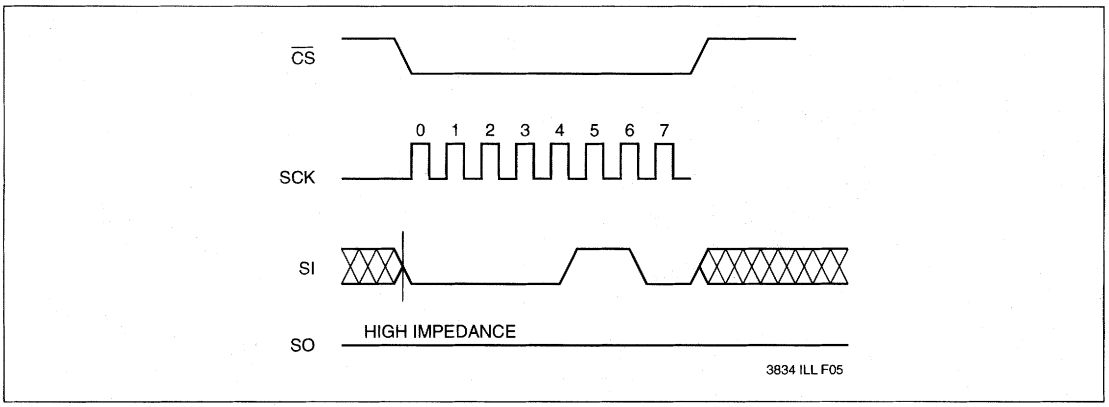
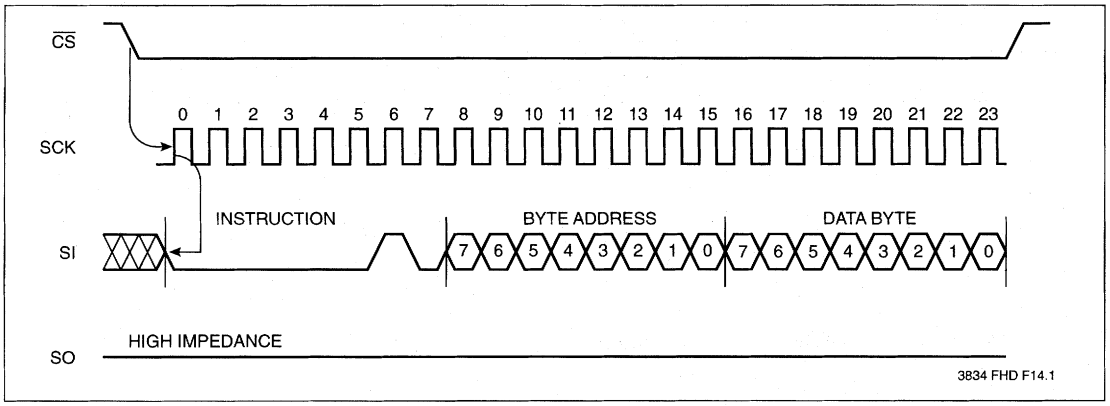


Figure 4. Byte Write Operation Sequence



X25020

Figure 5. Page Write Operation Sequence

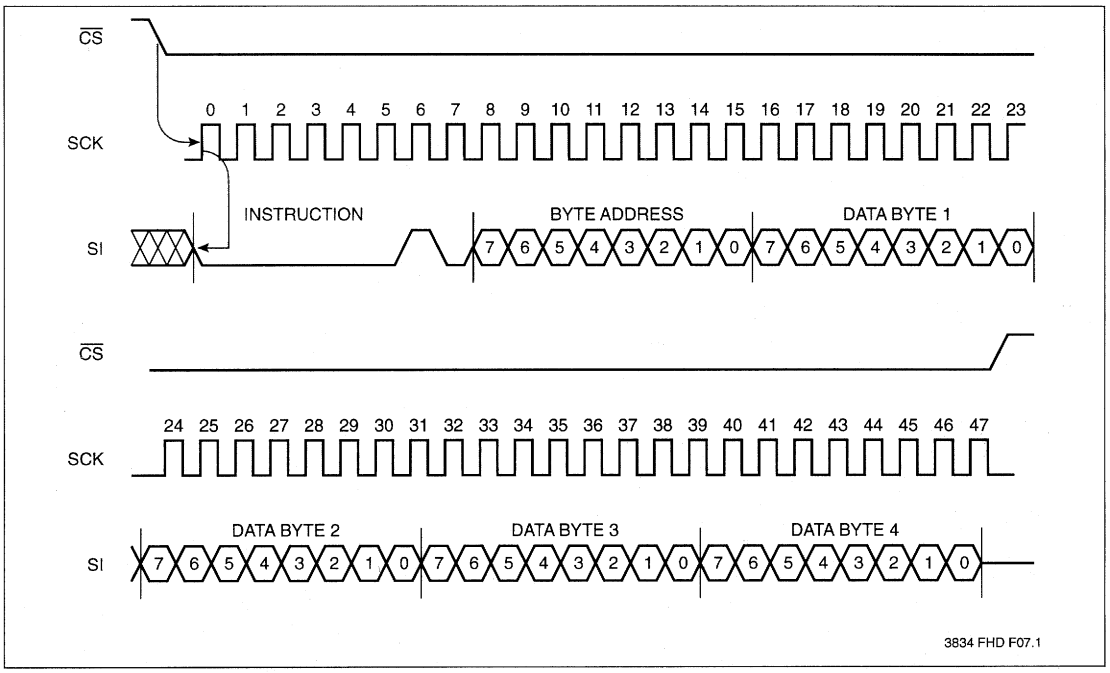
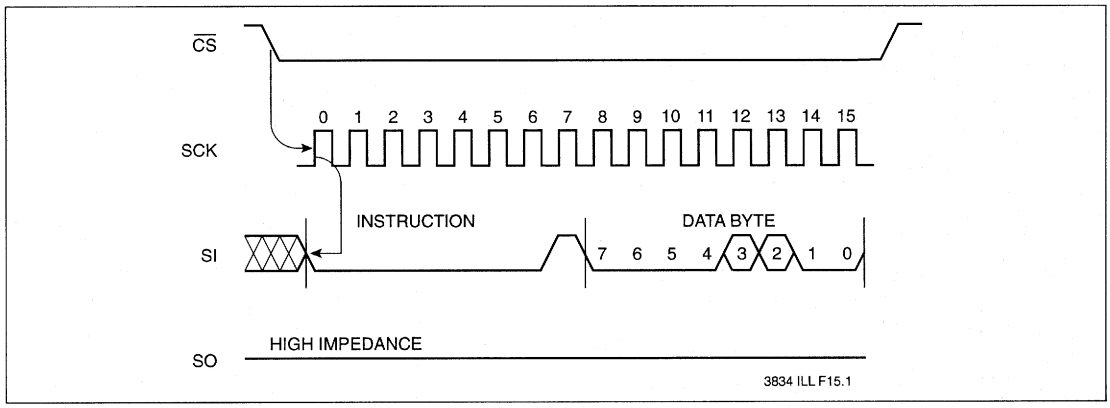


Figure 6. Write Status Register Operation Sequence



X25020

ABSOLUTE MAXIMUM RATINGS*

Temperature under Bias ..... –65°C to +135°C
Storage Temperature ..... –65°C to +150°C
Voltage on any Pin with Respect to V_{SS} ..... –1V to +7V
D.C. Output Current 5mA
Lead Temperature
(Soldering, 10 seconds) ..... 300°C

***COMMENT**

Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and the functional operation of the device at these or any other conditions above those listed in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

RECOMMENDED OPERATING CONDITIONS

Temp	Min.	Max.
Commercial	0°C	+70°C
Industrial	–40°C	+85°C
Military	–55°C	+125°C

3834 PGM T05.1

Supply Voltage	Limits
X25020	5V ±10%
X25020–3	3V to 5.5V
X25020–2.7	2.7V to 5.5V

3834 PGM T06.1

D.C. OPERATING CHARACTERISTICS (Over the recommended operating conditions unless otherwise specified.)

Symbol	Parameter	Limits		Units	Test Conditions
		Min.	Max.		
I_{CC}	V_{CC} Supply Current (Active)		3	mA	SCK = V_{CC} x 0.1/V_{CC} x 0.9 @ 1MHz, SO = Open
I_{SB}	V_{CC} Supply Current (Standby)		150	µA	$\overline{CS}$ = V_{CC}, V_{IN} = V_{SS} or V_{CC} – 0.3V
I_{LI}	Input Leakage Current		10	µA	V_{IN} = V_{SS} to V_{CC}
I_{LO}	Output Leakage Current		10	µA	V_{OUT} = V_{SS} to V_{CC}
V_{IL}(1)	Input LOW Voltage	–1	V_{CC} x 0.3	V	
V_{IH}(1)	Input HIGH Voltage	V_{CC} x 0.7	V_{CC} + 0.5	V	
V_{OL}	Output LOW Voltage		0.4	V	I_{OL} = 2mA
V_{OH}	Output HIGH Voltage	V_{CC}–0.8		V	I_{OH} = –1mA

3834 PGM T07.3

POWER-UP TIMING

Symbol	Parameter	Min.	Max.	Units
t_{PUR}(2)	Power-up to Read Operation		1	ms
t_{PUW}(2)	Power-up to Write Operation		5	ms

3834 PGM T08

CAPACITANCE T_A = +25°C, f = 1MHz, V_{CC} = 5V.

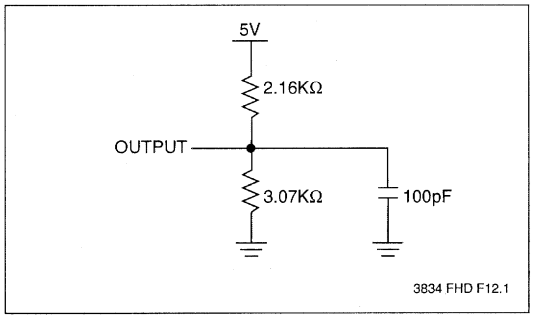
Symbol	Test	Max.	Units	Conditions
C_{OUT}(2)	Output Capacitance (SO)	8	pF	V_{OUT} = 0V
C_{IN}(2)	Input Capacitance (SCK, SI, $\overline{CS}$, $\overline{WP}$, $\overline{HOLD}$)	6	pF	V_{IN} = 0V

3834 PGM T09.1

Notes: (1) V_{IL} min. and V_{IH} max. are for reference only and are not tested.
(2) This parameter is periodically sampled and not 100% tested.

X25020

EQUIVALENT A.C. LOAD CIRCUIT AT 5V V_{CC}



A.C. TEST CONDITIONS

Input Pulse Levels	V_{CC} x 0.1 to V_{CC} x 0.9
Input Rise and Fall Times	10ns
Input and Output Timing Level	V_{CC} x 0.5

3834 PGM T10

2

A.C. CHARACTERISTICS (Over recommended operating conditions, unless otherwise specified)

Data Input Timing

Symbol	Parameter	Min.	Max.	Units
f_{SCK}	Clock Frequency	0	1	MHz
t_{CYC}	Cycle Time	1000		ns
t_{LEAD}	$\overline{CS}$ Lead Time	500		ns
t_{LAG}	$\overline{CS}$ Lag Time	500		ns
t_{WH}	Clock HIGH Time	400		ns
t_{WL}	Clock LOW Time	400		ns
t_{SU}	Data Setup Time	100		ns
t_H	Data Hold Time	100		ns
t_{RI}	Data In Rise Time		2	µs
t_{FI}	Data In Fall Time		2	µs
t_{HD}	HOLD Setup Time	200		ns
t_{CD}	HOLD Hold Time	200		ns
t_{CS}	$\overline{CS}$ Deselect Time	500		ns
t_{WC}(3)	Write Cycle Time		10	ms

3834 PGM T11.2

Data Output Timing

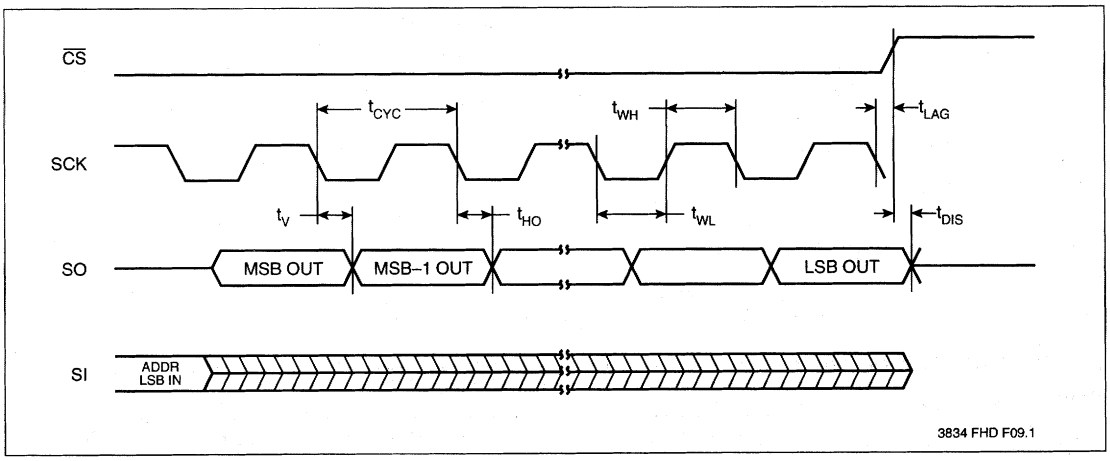
Symbol	Parameter	Min.	Max.	Units
f_{SCK}	Clock Frequency	0	1	MHz
t_{DIS}	Output Disable Time		500	ns
t_V	Output Valid from Clock LOW		360	ns
t_{HO}	Output Hold Time	0		ns
t_{RO}	Output Rise Time		300	ns
t_{FO}	Output Fall Time		300	ns
t_{LZ}	HOLD HIGH to Output in Low Z	100		ns
t_{HZ}	HOLD LOW to Output in High Z	100		ns

3834 PGM T12.1

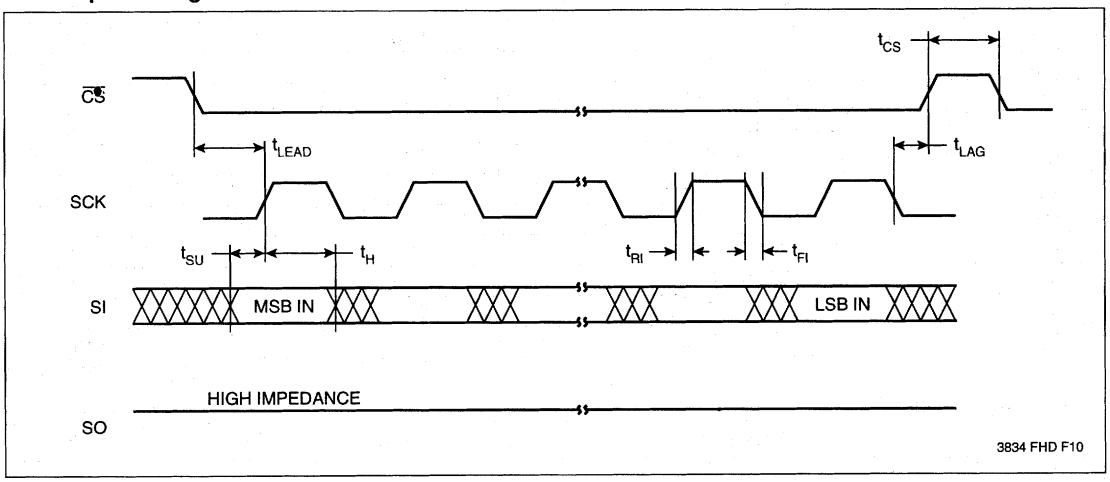
Notes: (3) t_{WC} is the time from the rising edge of $\overline{CS}$ after a valid write sequence has been sent to the end of the self-timed internal nonvolatile write cycle.

X25020

Serial Output Timing

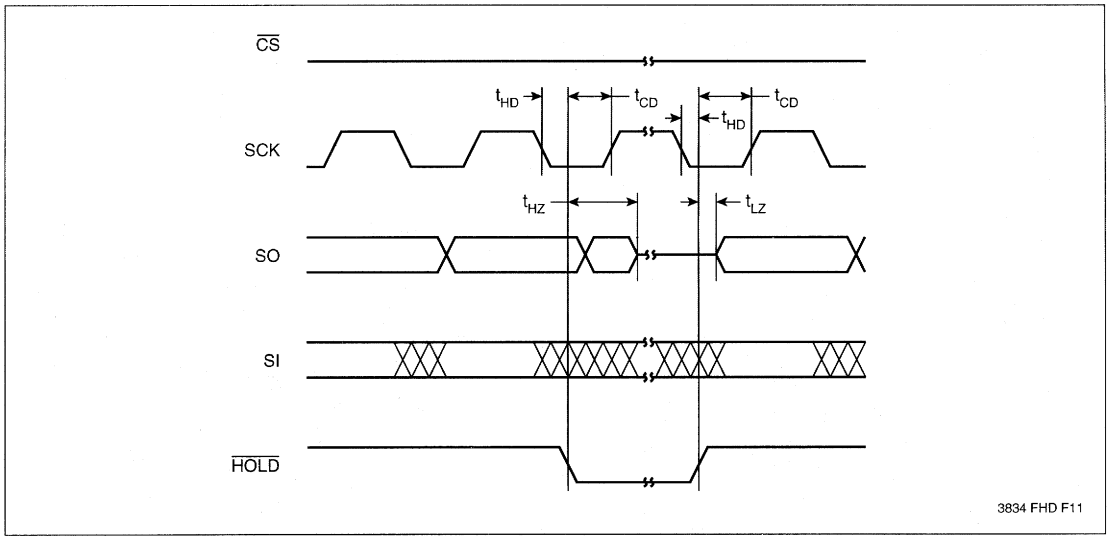


Serial Input Timing



X25020

Hold Timing



2

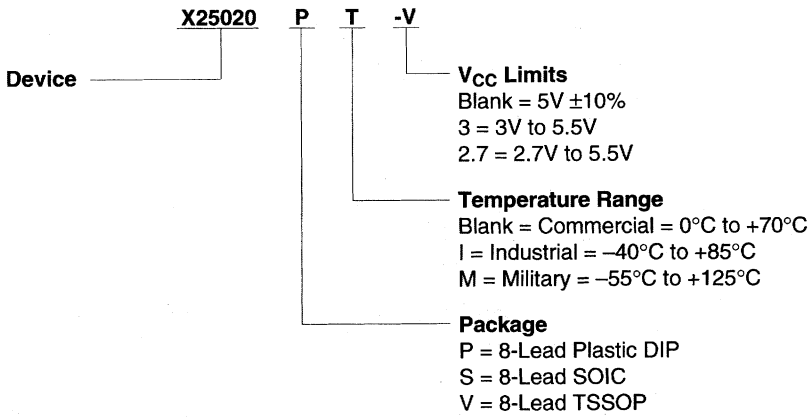
SYMBOL TABLE

WAVEFORM	INPUTS	OUTPUTS
	Must be steady	Will be steady
	May change from LOW to HIGH	Will change from LOW to HIGH
	May change from HIGH to LOW	Will change from HIGH to LOW
	Don't Care: Changes Allowed	Changing: State Not Known
	N/A	Center Line is High Impedance

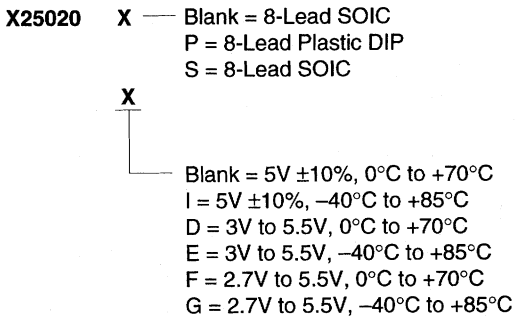
 © National Instruments Corporation

X25020

ORDERING INFORMATION



Part Mark Convention



LIMITED WARRANTY

Devices sold by Xicor, Inc. are covered by the warranty and patent indemnification provisions appearing in its Terms of Sale only. Xicor, Inc. makes no warranty, express, statutory, implied, or by description regarding the information set forth herein or regarding the freedom of the described devices from patent infringement. Xicor, Inc. makes no warranty of merchantability or fitness for any purpose. Xicor, Inc. reserves the right to discontinue production and change specifications and prices at any time and without notice.

Xicor, Inc. assumes no responsibility for the use of any circuitry other than circuitry embodied in a Xicor, Inc. product. No other circuits, patents, licenses are implied.

U.S. PATENTS

Xicor products are covered by one or more of the following U.S. Patents: 4,263,664; 4,274,012; 4,300,212; 4,314,265; 4,326,134; 4,393,481; 4,404,475; 4,450,402; 4,486,769; 4,488,060; 4,520,461; 4,533,846; 4,599,706; 4,617,652; 4,668,932; 4,752,912; 4,829, 482; 4,874, 967; 4,883, 976. Foreign patents and additional patents pending.

LIFE RELATED POLICY

In situations where semiconductor component failure may endanger life, system designers using this product should design the system with appropriate error detection and correction, redundancy and back-up features to prevent such an occurence.

Xicor's products are not authorized for use in critical components in life support devices or systems.

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

Appendix D
OKI MSM82C53 Data Sheet*

This appendix contains the manufacturer data sheet for the MSM82C53 CMOS programmable interval timer manufactured by OKI Semiconductor, Inc. This counter/timer is used on the DAQCard-1200.

* Copyright © OKI Semiconductor, Inc. Reprinted with permission of copyright owner.
All rights reserved.
OKI Semiconductor. *Microprocessor Data Book 1990/1991.*

OKI semiconductor

MSM82C53-2RS/GS/JS

CMOS PROGRAMMABLE INTERVAL TIMER

GENERAL DESCRIPTION

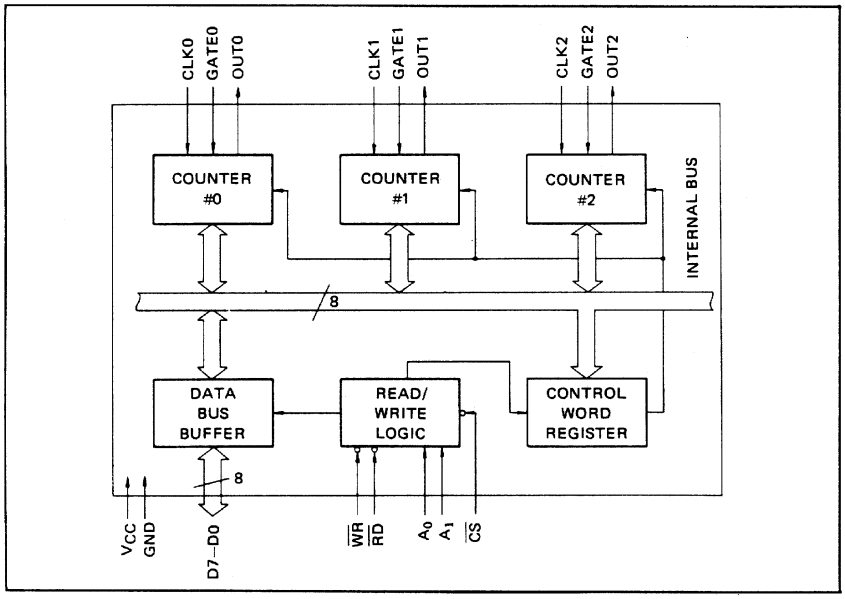
The MSM82C53-2RS/GS/JS is programmable universal timers designed for use in microcomputer systems. Based on silicon gate CMOS technology, it requires a standby current of only 100 μA (max.) when the chip is in the nonselected state. During timer operation, power consumption is still very low with only 8 mA (max.) at 8 MHz of current required.

The device consists of three independent counters, and can count up to a maximum of 8 MHz (MSM82C53-2) The timer features six different counter modes, and binary count/BCD count functions. Count values can be set in byte or word units, and all functions are freely programmable.

FEATURES

- Maximum operating frequency of 8 MHz (MSM82C53-2)
- High speed and low power consumption achieved through silicon gate CMOS technology
- Completely static operation
- Three independent 16-bit down-counters
- 3V to 6V single power supply
- Six counter modes abailable for each counter
- Binary and decimal counting possible
- 24 pin Plastic DIP (DIP24-P-600): MSM82C53-2RS
- 28 pin Plastic QFJ (QFJ28-P-S450): MSM82C53-2JS
- 32 pin-V Plastic SOP (SSOP32-P-430-VK): MSM82C53-2GS-VK

FUNCTIONAL BLOCK DIAGRAM

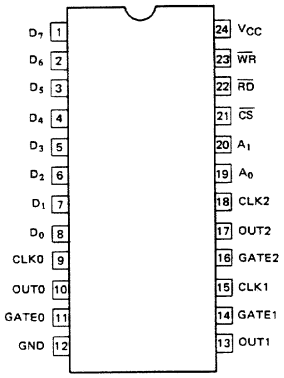


 © National Instruments Corporation

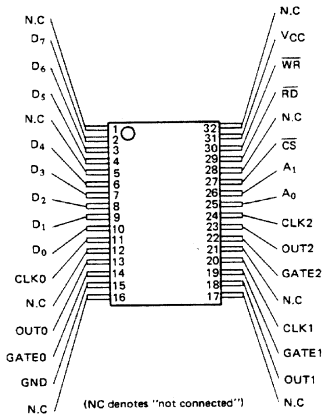
■ I/O-MSM82C53-2RS/GS/JS ■

PIN CONFIGURATION (Top View)

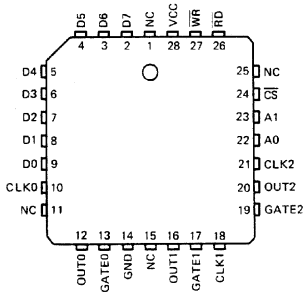
24 pin Plastic DIP



32 pin Plastic SOP



28 pin Plastic QFJ



ABSOLUTE MAXIMUM RATINGS

Parameter	Symbol	Conditions	Limits			Unit
			MSM82C53-2RS	MSM82C53-2GS	MSM82C53-2JS	
Ssupply Voltage	V_{CC}	Respect to GND	−0.5 to +7			V
Input Voltage	V_{IN}		−0.5 to V_{CC} + 0.5			V
Output Voltage	V_{OUT}		−0.5 to V_{CC} + 0.5			V
Storage Temperature	T_{stg}		−55 to +150			°C
Power Dissipation	P_D	Ta = 25°C	0.9	0.7	0.9	W

OPERATING RANGES

Parameter	Symbol	Limits	Conditions	Unit
Supply Voltage	V_{CC}	3 to 6	V_{IL} = 0.2V, V_{IH} = V_{CC} − 0.2V, operating frequency 2.6 MHz	V
Operating Temperature	T_{OP}	−40 to +85		°C

RECOMMENDED OPERATING CONDITIONS

Parameter	Symbol	Min.	Typ.	Max.	Unit
Supply Voltage	V_{CC}	4.5	5	5.5	V
Operating Temperature	T_{OP}	−40	+25	+85	°C
"L" Input Voltage	V_{IL}	−0.3		+0.8	V
"H" Input Voltage	V_{IH}	2.2		V_{CC} + 0.3	V

DC CHARACTERISTICS

Parameter	Symbol	Conditions		Min.	Typ.	Max.	Unit
"L" Output Voltage	V_{OL}	I_{OL} = 4mA				0.45	V
"H" Output Voltage	V_{OH}	I_{OH} = −1mA		3.7			V
Input Leak Current	I_{LI}	$0 \leq V_{IN} \leq V_{CC}$	V_{CC}=4.5V to 5.5V	−10		10	μA
Output Leak Current	I_{LO}	$0 \leq V_{OUT} \leq V_{CC}$	Ta=−40°C to +85°C	−10		10	μA
Standby Supply Current	I_{CCS}	$\overline{CS} \geqq V_{CC}$ −0.2V $V_{IH} \geqq V_{CC}$ − 0.2V $V_{IL} \leqq$0.2V				100	μA
Operating Supply Current	I_{CC}	t_{CLK} = 125 ns C_L = 0pF				8	mA

■ I/O-MSM82C53-2RS/GS/JS ■

AC CHARACTERISTICS

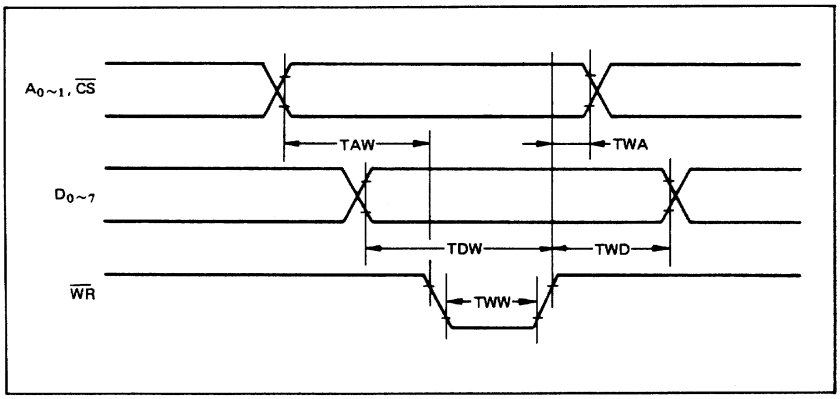
(V_{CC} = 4.5V ~ 5.5V, Ta = –40 ~ +85°C)

Parameter	Symbol	MSM82C53-2 Min.	MSM82C53-2 Max.	Unit	Conditions	
Address Set-up Time before reading	TAR	30		ns	Read cycle	C_L = 150pF
Address Hold Time after reading	TRA	0		ns		
Read Pulse Width	TRR	150		ns		
Read Recovery Time	TRVR	200		ns		
Address Set-up Time before writing	TAW	0		ns	Write cycle	
Address Hold Time after writing	TWA	20		ns		
Write Pulse Width	TWW	150		ns		
Data Input Set-up Time before writing	TDW	100		ns		
Data Input Hold Time after writing	TWD	20		ns		
Write Recovery time	TRVW	200		ns		
Clock Cycle Time	TCLK	125	D.C.	ns	Clock and gate timing	
Clock "H" Pulse Width	TPWH	60		ns		
Clock "L" Pulse Width	TPWL	60		ns		
"H" Gate Pulse Width	TGW	50		ns		
"L" Gate Pulse Width	TGL	50		ns		
Gate Input Set-up Time before clock	TGS	50		ns		
Gate Input Hold Time after clock	TGH	50		ns		
Output Delay Time after reading	TRD		120	ns	Delay time	
Output Floating Delay Time after reading	TDF	5	90	ns		
Output Delay Time after gate	TODG		120	ns		
Output Delay Time after clock	TOD		150	ns		
Output Delay Time after address	TAD		180	ns		

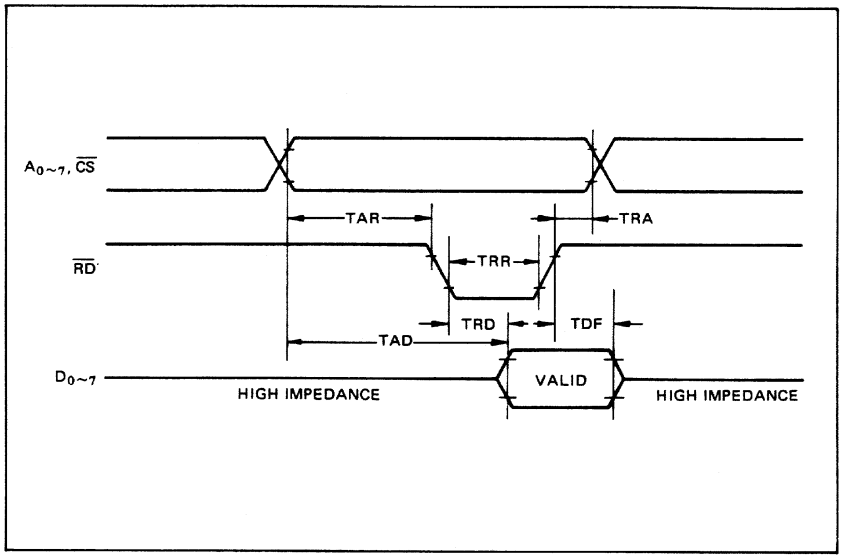
Note: Timing measured at V_L = 0.8V and V_H = 2.2V for both inputs and outputs.

TIME CHART

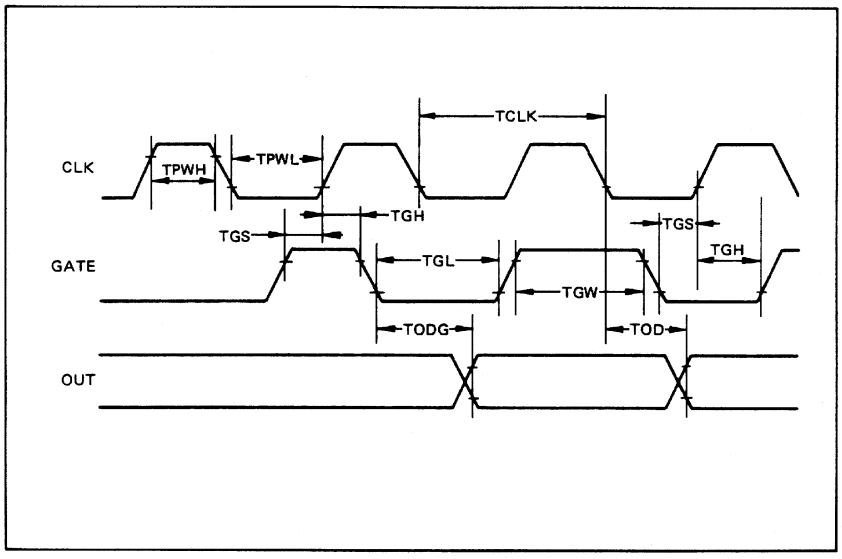
Write Timing



Read Timing



Clock & Gate Timing



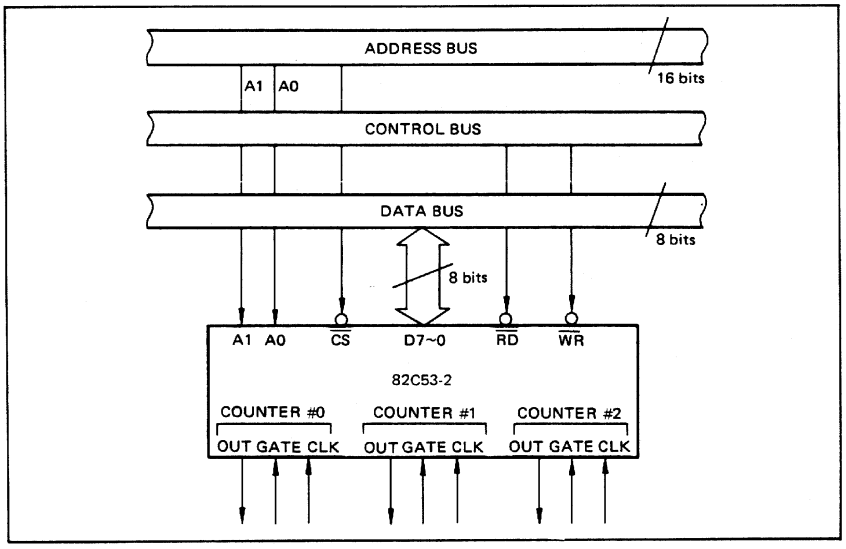
 © National Instruments Corporation

■ I/O-MSM82C53-2RS/GS/JS ■

DESCRIPTION OF PIN FUNCTIONS

Pin Symbol	Name	Input/output	Function
D7 ~ D0	Bidirectional data bus	Input/output	Three-state 8-bit bidirectional data bus used when writing control words and count values, and reading count values upon reception of $\overline{WR}$ and $\overline{RD}$ signals from CPU.
$\overline{CS}$	Chip select input	Input	Data transfer with the CPU is enabled when this pin is at low level. When at high level, the data bus (D_0 thru D_7) is switched to high impedance state where neither writing nor reading can be executed. Internal registers, however, remain unchanged.
$\overline{RD}$	Read input	Input	Data can be transferred from MSM82C53 to CPU when this pin is at low level.
$\overline{WR}$	Write input	Input	Data can be transferred from CPU to MSM82C53 when this pin is at low level.
A0, A1	Address input	Input	One of the three internal counters or the control word register is selected by A0/A1 combination. These two pins are normally connected to the two lower order bits of the address bus.
CLK0~2	Clock input	Input	Supply of three clock signals to the three counters incorporated in MSM82C53.
GATE0~2	Gate input	Input	Control of starting, interruption, and restarting of counting in the three respective counters in accordance with the set control word contents.
OUT0~2	Counter output	Output	Output of counter output waveform in accordance with the set mode and count value.

SYSTEM INTERFACING



DESCRIPTION OF BASIC OPERATIONS

Data transfers between the internal registers and the external data bus is outlined in the following table.

$\overline{CS}$	$\overline{RD}$	$\overline{WR}$	A1	A0	Function
0	1	0	0	0	Data bus to counter # 0 Writing
0	1	0	0	1	Data bus to counter # 1 Writing
0	1	0	1	0	Data bus to counter # 2 Writing
0	1	0	1	1	Data bus to control word register Writing
0	0	1	0	0	Data bus from counter # 0 Reading
0	0	1	0	1	Data bus from counter # 1 Reading
0	0	1	1	0	Data bus from counter # 2 Reading
0	0	1	1	1	Data bus in high impedance status
1	x	x	x	x	
0	1	1	x	x	

x denotes "not specified".

DESCRIPTION OF OPERATION

82C53 functions are selected by a control word from the CPU. In the required program sequence, the control word setting is followed by the count value setting and execution of the desired timer operation.

Control Word and Count Value Program

Each counter operation mode is set by control word programming. The control word format is outlined below.

D7	D6	D5	D4	D3	D2	D1	D0
SC1	SC0	RL1	RL0	M2	M1	M0	BCD
Select Counter		Read/Load		Mode			BCD

($\overline{CS}$ = 0, A0, A1 = 1,1, $\overline{RD}$ = 1, $\overline{WR}$ = 0)

- **Select Counter (SC0, SC1):** Selection of set counter

SC1	SC0	Set Contents
0	0	Counter # 0 selection
0	1	Counter # 1 selection
1	0	Counter # 2 selection
1	1	Illegal combination

- **Read/Load (RL1, RL0):** Count value Reading/ Loading format setting

RL1	RL0	Set Contents
0	0	Counter Latch operation
0	1	Reading/Loading of Least Significant byte (LSB)
1	0	Reading/Loading of Most Significant byte (MSB)
1	1	Reading/Loading of LSB followed by MSB

- **Mode (M2, M1, M0):** Operation waveform mode setting

M2	M1	M0	Set Contents
0	0	0	Mode 0 (Interrupt on Terminal Count)
0	0	1	Mode 1 (Programmable One-Shot)
x	1	0	Mode 2 (Rate Generator)
x	1	1	Mode 3 (Square Wave Generator)
1	0	0	Mode 4 (Software Triggered Strobe)
1	0	1	Mode 5 (Hardware Triggered Strobe)

x denotes "not specified".

- **BCD:** Operation count mode setting

BCD	Set Contents
0	Binary Count (16-bits Binary)
1	BCD Count (4-decades Binary Coded Decimal)

After setting Read/Load, Mode, and BCD in each counter as outlined above, next set the desired count value. (In some Modes, counting is started immediately after the count value has been written). This count value setting must conform with the Read/Load format set in advance. Note that the internal counters are reset to OOOOH during control word setting. The counter value (OOOOH) can.t be read.

If the two bytes (LSB and MSB) are written at this stage (RL0 and RL1 = 1,1), take note of the following precaution.

Although the count values may be set in the three counters in any sequence after the control word has been set in each counter, count values must be set consecutively in the LSB – MSB order in any one counter.

 © National Instruments Corporation

■ I/O-MSM82C53-2RS/GS/JS ■

- **Example of control word and count value setting**

```
Counter #0: Read/Load LSB only, Mode 3, Binary count, count value 3H
Counter #1: Read/Load MSB only, Mode 5, Binary count, count value AA00H
Counter #2: Read/Load LSB and MSB, Mode 0, BCD count, count value 1234

MVI A, 1EH
OUT n3          Counter #0 control word setting
MVI A, 6AH
OUT n3          Counter #1 control word setting
MVI A, B1H
OUT n3          Counter #2 control word setting
MVI A, 03H
OUT n0          Counter #0 count value setting
MVI A, AAH
OUT n1          Counter #1 count value setting
MVI A, 34H
OUT n2          Counter #2 count value setting
MVI A, 12H      (LSB then MSB)
OUT n2
```

Notes: n0: Counter #0 address
n1: Counter #1 address
n2: Counter #2 address
n3: Control word register address

- **The minimum and maximum count values which can be counted in each mode are listed below.**

Mode	Min.	Max.	Remarks
0	1	0	0 executes 10000H count (ditto in other modes)
1	1	0	
2	2	0	1 cannot be counted
3	2	1	1 executes 10001H count
4	1	0	
5	1	0	

Mode Definition

- **Mode 0 (terminal count)**
  The counter output is set to "L" level by the mode setting. If the count value is then written in the counter with the gate input at "H" level (that is, upon completion of writing the MSB when there are two bytes), the clock input counting is started. When the terminal count is reached, the output is switched to "H" level and is maintained in this status until the control word and count value are set again.
  Counting is interrupted if the gate input is switched to "L" level, and restarted when switched back to "H" level.
  When Count Values are written during counting, the operation is as follows:

  1-byte Read/Load.... When the new count value is written, counting is stopped immediately, and then restarted at the new count value by the next clock.
  2-byte Read/Load.... When byte 1 (LSB) of the new count value is written, counting is stopped immediately. Counting is restarted at the new count value when byte 2 (MSB) is written.

- **Mode 1 (programmable one-shot)**
  The counter output is switched to "H" level by the mode setting. Note that in this mode, counting is not started if only the count value is written. Since counting has to be started in this mode by using the leading edge of the gate input as a trigger, the counter output is switched to "L" level by the next clock after the gate input trigger. This "L" level status is maintained during the set count value, and is switched back to "H" level when the terminal count is reached.
  Once counting has been started, there is no interruption until the terminal count is reached, even if the gate input is switched to "L" level in the meantime. And although counting continues even if a new count value is written during the counting, counting is started at the new count value if another trigger is applied by the gate input.

- **Mode 2 (rate generator)**
  The counter output is switched to "H" level by the mode setting. When the gate input is at "H" level, counting is started by the next clock after the count value has been written. And if the gate input is at "L" level, counting is started by using the rising edge of the gate input as a trigger after the count value has been set.
  An "L" level output pulse appears at the counter output during a single clock duration once every n clock inputs where n is the set count value. If a new count value is written during while counting is in progress, counting is started at the new count value following output of the pulse currently being counted. And if the gate input is switched to "L" level during counting, the counter output is forced to switch to "H" level, the counting being restarted by the rising edge of the gate input.

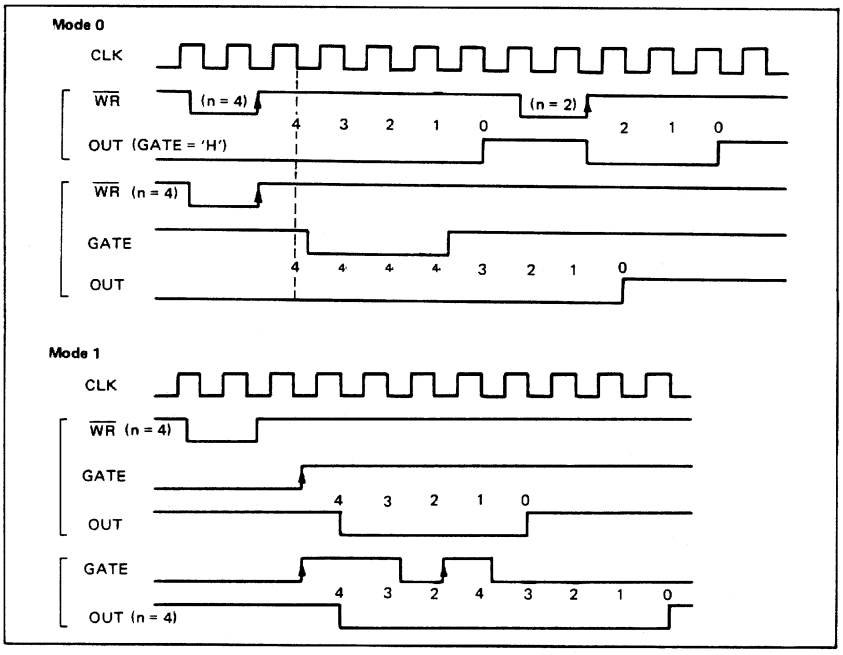
- **Mode 3 (square waveform rate generator)**
  The counter output is switched to "H" level by the mode setting. Counting is started in the same way as described for mode 2 above.
  The repeated square wave output appearing at the counter output contains half the number of counts as the set count value. If the set count value (n) is an odd number, the repeated square wave output consists of only (n + 1)/2 clock inputs at "H" level and (n − 1)/2 clock inputs at "L" level.
  If a new count value is written during counting, the new count value is reflected immediately after the

change ("H" to "L" or "L" to "H") in the next counter output to be executed. The counting operation at the gate input is done the same as in mode 2.

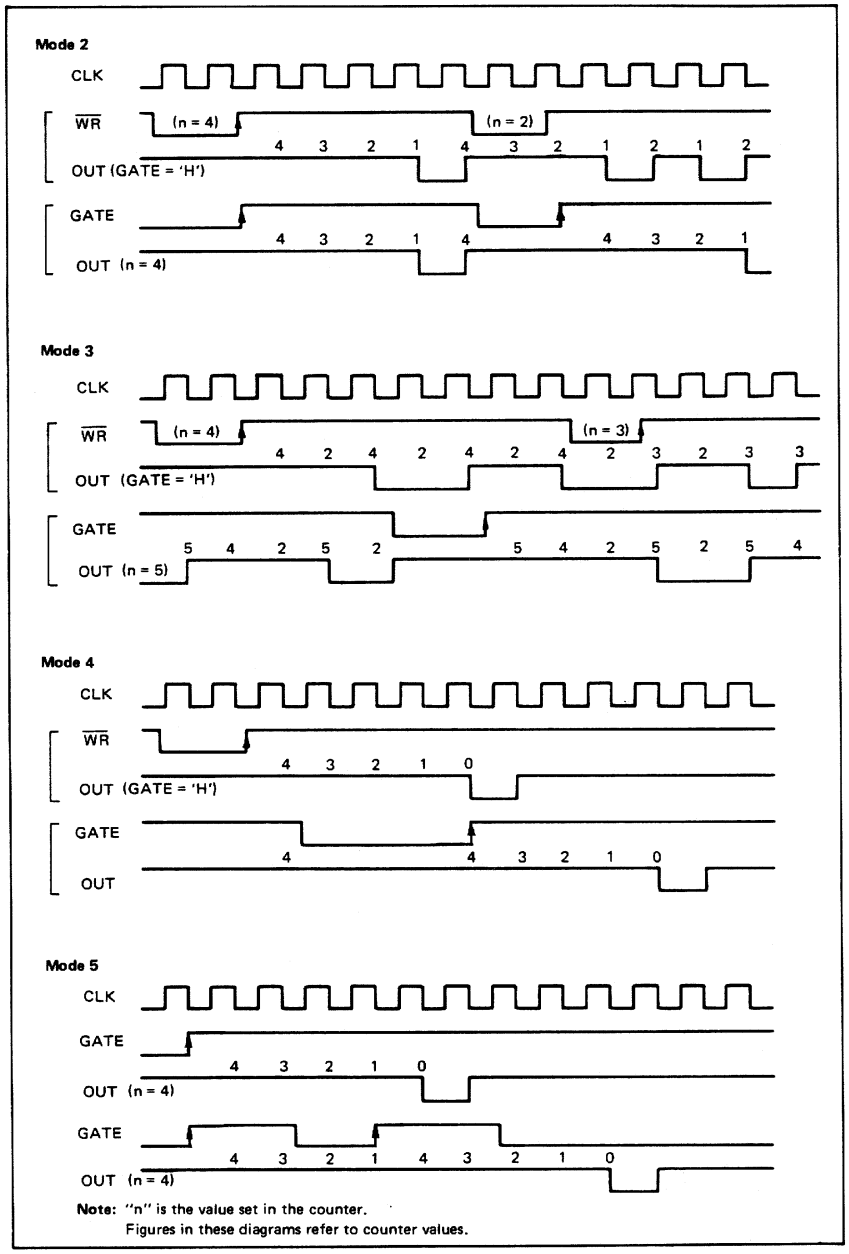
- **Mode 4 (software trigger strobe)**
  The counter output is switched to "H" level by the mode setting. Counting is started in the same way as described for mode 0. A single "L" pulse equivalent to one clock width is generated at the counter output when the terminal count is reached.
  This mode differs from 2 in that the "L" level output appears one clock earlier in mode 2, and that pulses are not repeated in mode 4. Counting is stopped when the gate input is switched to "L" level, and restarted from the set count value when switched back to "H" level.

- **Mode 5 (hardware trigger strobe)**
  The counter output is switched to "H" level by the mode setting. Counting is started, and the gate input used, in the same way as in mode 1.
  The counter output is identical to the mode 4 output.
  The various roles of the gate input signals in the above modes are summarized in the following table.

Mode \ Gate	"L" Level Falling Edge	Rising Edge	"H" Level
0	Counting not possible		Counting possible
1		(1) Start of counting (2) Retriggering	
2	(1) Counting not possible (2) Counter output forced to "H" level	Start of counting	Counting possible
3	(1) Counting not possible (2) Counter output forced to "H" level	Start of counting	Counting possible
4	Counting not possible		Counting possible
5		(1) Start of counting (2) Retriggering	



■ I/O-MSM82C53-2RS/GS/JS ■



Note: "n" is the value set in the counter.
Figures in these diagrams refer to counter values.

Reading of Counter Values

All 82C53 counting is down-counting, the counting being in steps of 2 in mode 3. Counter values can be read during counting by (1) direct reading, and (2) counter latching ("read on the fly").

- **Direct reading**
  Counter values can be read by direct reading operations.
  Since the counter value read according to the timing of the $\overline{RD}$ and CLK signals is not guaranteed, it is necessary to stop the counting by a gate input signal, or to interrupt the clock input temporarily by an external circuit to ensure that the counter value is correctly read.

- **Counter latching**
  In this method, the counter value is latched by writing a counter latch command, thereby enabling a stable value to be read without effecting the counting in any way at all. An example of a counter latching program is given below.
  Counter latching executed for counter #1 (Read/Load 2-byte setting)

MVI A 0 1 0 0 x x x x
- Denotes counter latching

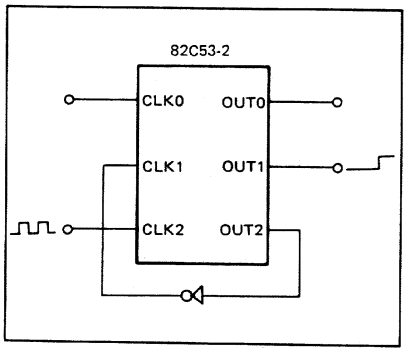
OUT n3
- Write in control word address (n3)
- The counter value at this point is latched

IN n1
- Reading of the LSB of the counter value latched from counter #1.
- n1: Counter #1 address

MOV B, A
IN n1
MOV C, A
- Reading of MSB from counter #1.

Example of Practical Application

- **82C53 used as a 32-bit counter.**



Use counter #1 and counter #2
Counter #1: mode 0, upper order 16-bit counter value
Counter #2: mode 2, lower order 16-bit counter value
This setting enables counting up to a maximum of 2^{32}

175

Appendix E
OKI MSM82C55A Data Sheet*

This appendix contains the manufacturer data sheet for the MSM82C55A CMOS programmable peripheral interface manufactured by OKI Semiconductor, Inc. This interface is used on the DAQCard-1200.

* Copyright © OKI Semiconductor, Inc. Reprinted with permission of copyright owner.
All rights reserved.
OKI Semiconductor. *Microprocessor Data Book 1990/1991.*

OKI semiconductor

MSM82C55A-2RS/GS/VJS

CMOS PROGRAMMABLE PERIPHERAL INTERFACE

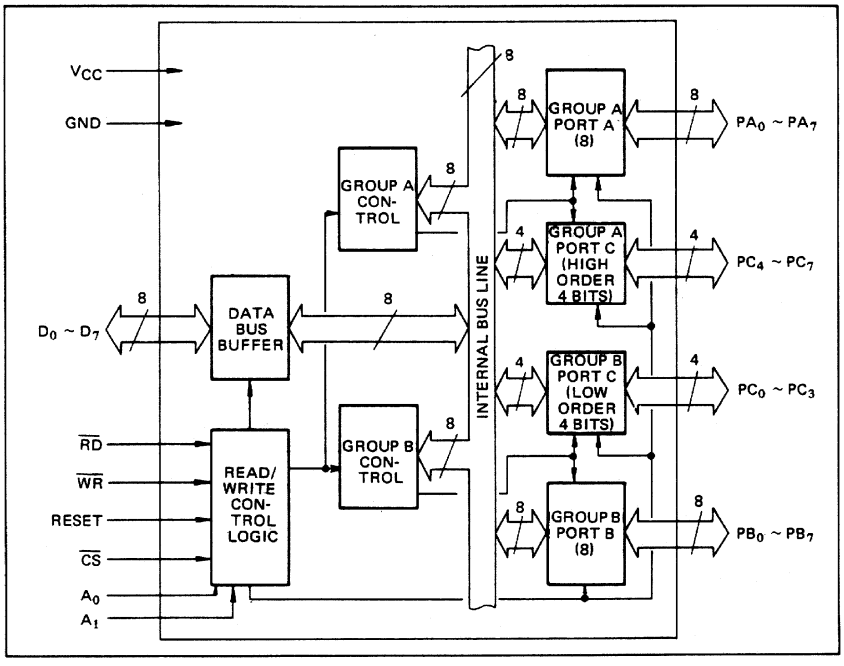
GENERAL DESCRIPTION

The MSM82C55A is a programmable universal I/O interface device which operates as high speed and on low power consumption due to 3 μ silicon gate CMOS technology. It is the best fit as an I/O port in a system which employs the 8-bit parallel processing MSM80C85A CPU. This device has 24-bit I/O pins equivalent to three 8-bit I/O ports and all inputs/outputs are TTL interface compatible.

FEATURES

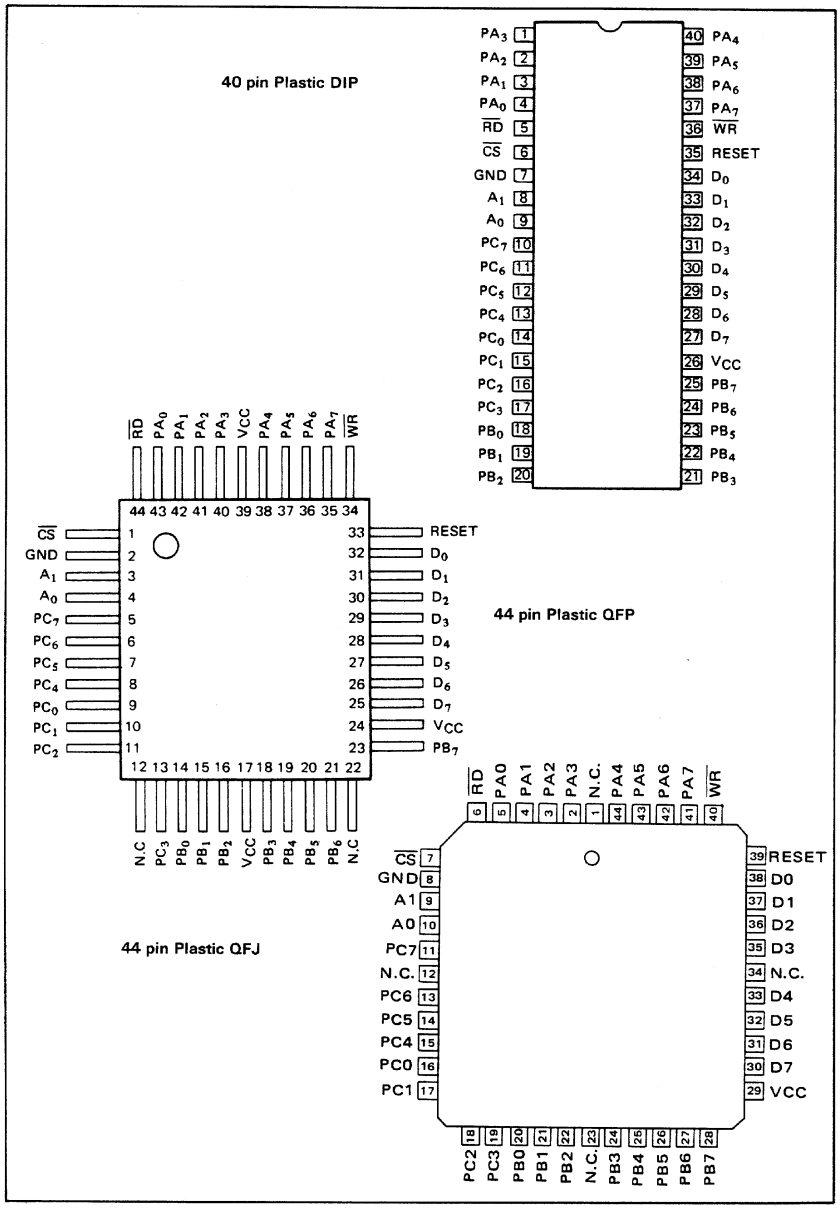
- High speed and low power consumption due to 3μ silicon gate CMOS technology
- 3V to 6V single power supply
- Full static operation
- Programmable 24-bit I/O ports
- Bidirectional bus operation (Port A)
- Bit set/reset function (Port C)
- TTL compatible
- Compatible with 8255A-5
- 40 pin Plastic DIP (DIP40-P-600): MSM82C55A-2RS
- 44 pin Plastic QFJ (QFJ44-P-S650): MSM82C55A-2JS
- 44 pin Plastic QFP (QFP44-P-910-2K): MSM82C55A-2GS-2K

CIRCUIT CONFIGURATION



 © National Instruments Corporation

PIN CONFIGURATION (Top View)



■ I/O-MSM82C55A-2RS/GS/VJS ■

ABSOLUTE MAXIMUM RATINGS

Parameter	Symbol	Conditions	Limits			Unit
			MSM82C55A-2RS	MSM82C55A-2GS	MSM82C55A-2VJS	
Ssupply Voltage	V_{CC}	Ta = 25°C with respect to GND	−0.5 to +7			V
Input Voltage	V_{IN}		−0.5 to V_{CC} + 0.5			V
Output Voltage	V_{OUT}		−0.5 to V_{CC} + 0.5			V
Storage Temperature	T_{stg}	—	−55 to +150			°C
Power Dissipation	P_D	Ta = 25°C	1.0	0.7	1.0	W

OPERATING RANGE

Parameter	Symbol	Limits	Unit
Supply Voltage	V_{CC}	3 to 6	V
Operating Temperature	T_{OP}	−40 to 85	°C

RECOMMENDED OPERATING RANGE

Parameter	Symbol	Min.	Typ.	Max.	Unit
Supply Voltage	V_{CC}	4.5	5	5.5	V
Operating Temperature	T_{OP}	−40	+25	+85	°C
"L" Input Voltage	V_{IL}	−0.3		+0.8	V
"H" Input Voltage	V_{IH}	2.2		V_{CC} +0.3	V

DC CHARACTERISTICS

Parameter	Symbol	Conditions		MSM82C55A-2			Unit
				Min.	Typ.	Max.	
"L" Output Voltage	V_{OL}	I_{OL} = 2.5 mA	V_{CC} = 4.5V to 5.5V Ta = −40°C to +85°C (C_L = 0pF)			0.4	V
"H" Output Voltage	V_{OH}	I_{OH} = −40 μA		4.2			V
		I_{OH} = −2.5 mA		3.7			V
Input Leak Current	I_{LI}	$0 \leq V_{IN} \leq V_{CC}$		−1		1	μA
Output Leak Current	I_{LO}	$0 \leq V_{OUT} \leq V_{CC}$		−10		10	μA
Supply Current (standby)	I_{CCS}	$\overline{CS} \geqq V_{CC}$ −0.2V $V_{IH} \geqq V_{CC}$ −0.2V $V_{IL} \leqq$ 0.2V			0.1	10	μA
Average Supply Current (active)	I_{CC}	I/O wire cycle 82C55A-2 ... 8MHzCPU timing				8	mA

 © National Instruments Corporation

AC CHARACTERISTICS

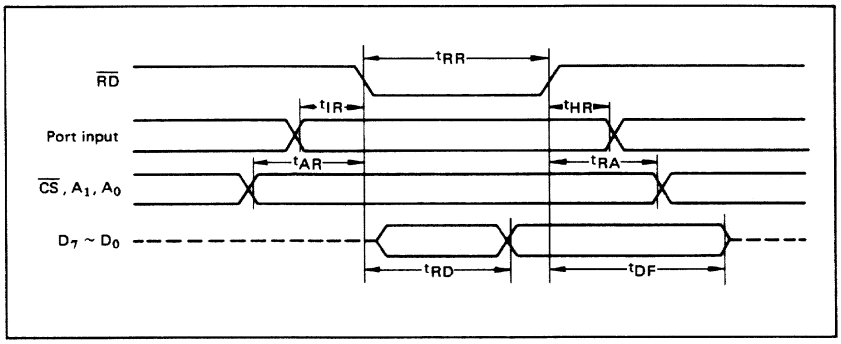
(V_{CC} = 4.5 to 5.5V, Ta = -40 to +80°C)

Parameter	Symbol	MSM82C55A-2		Unit	Remarks
		Min.	Max.		
Setup Time of address to the falling edge of $\overline{RD}$	t_{AR}	20		ns	Load 150 pF
Hold Time of address to the rising edge of $\overline{RD}$	t_{RA}	0		ns	
$\overline{RD}$ Pulse Width	t_{RR}	100		ns	
Delay Time from the falling edge of $\overline{RD}$ to the output of defined data	t_{RD}		120	ns	
Delay Time from the rising edge of $\overline{RD}$ to the floating of data bus	t_{DF}	10	75	ns	
Time from the rising edge of $\overline{RD}$ or $\overline{WR}$ to the next falling edge of $\overline{RD}$ or $\overline{WR}$	t_{RV}	200		ns	
Setup Time of address before the falling edge of $\overline{WR}$	t_{AW}	0		ns	
Hold Time of address after the rising edge or $\overline{WR}$	t_{WA}	20		ns	
$\overline{WR}$ Pulse Width	t_{WW}	150		ns	
Setup Time of bus data before the rising edge of $\overline{WR}$	t_{DW}	50		ns	
Holt Time of bus data after the rising edge of $\overline{WR}$	t_{WD}	30		ns	
Delay Time from the rising edge of $\overline{WR}$ to the output of defined data	t_{WB}		200	ns	
Setup Time of port data before the falling edge of $\overline{RD}$	t_{IR}	20		ns	
Hold Time of port data after the rising edge of $\overline{RD}$	t_{HR}	10		ns	
$\overline{ACK}$ Pulse Width	t_{AK}	100		ns	
$\overline{STB}$ Pulse Width	t_{ST}	100		ns	
Setup Time of port data before the rising edge of $\overline{STB}$	t_{PS}	20		ns	
Hold Time of port data after the rising edge of $\overline{STB}$	t_{PH}	50		ns	
Delay Time from the falling edge of $\overline{ACK}$ to the output of defined data	t_{AD}		150	ns	
Delay Time from the rising edge of $\overline{ACK}$ to the floating of port (Port A in mode 2)	t_{KD}	20	250	ns	
Delay Time from the rising edge of WR to the falling edge of $\overline{OBF}$	t_{WOB}		150	ns	
Delay Time from the falling edge of ACK to the rising edge of $\overline{OBF}$	t_{AOB}		150	ns	
Delay Time from the falling edge of STB to the rising edge of IBF	t_{SIB}		150	ns	
Delay Time from the rising edge of $\overline{RD}$ to the falling edge of IBF	t_{RIB}		150	ns	
Delay Time from the falling edge of $\overline{RD}$ to the falling edge of INTR	t_{RIT}		200	ns	
Delay Time from the rising edge of $\overline{STB}$ to the rising edge of INTR	t_{SIT}		150	ns	
Delay Time from the rising edge of $\overline{ACK}$ to the rising edge of INTR	t_{AIT}		150	ns	
Delay Time from the falling edge of $\overline{WR}$ to the falling edge of INTR	t_{WIT}		250	ns	

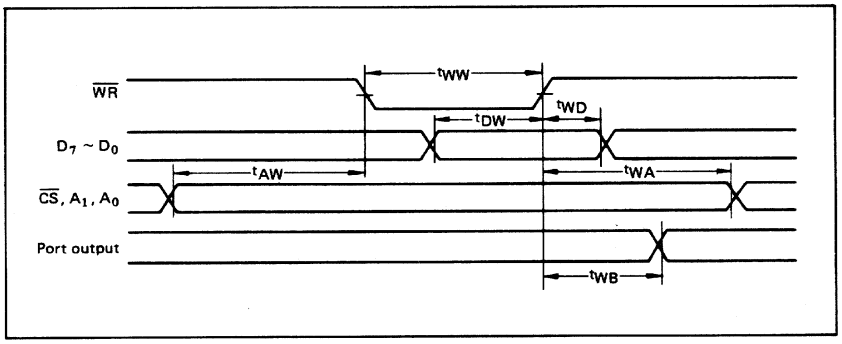
Note: Timing is measured at V_L = 0.8 V and V_H = 2.2 V for both input and outputs.

■ I/O-MSM82C55A-2RS/GS/VJS ■

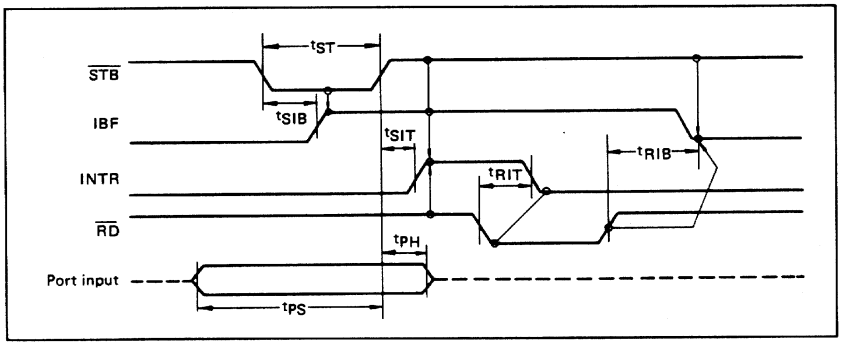
Basic Input Operation (Mode 0)



Basic Output Operation (Mode 0)

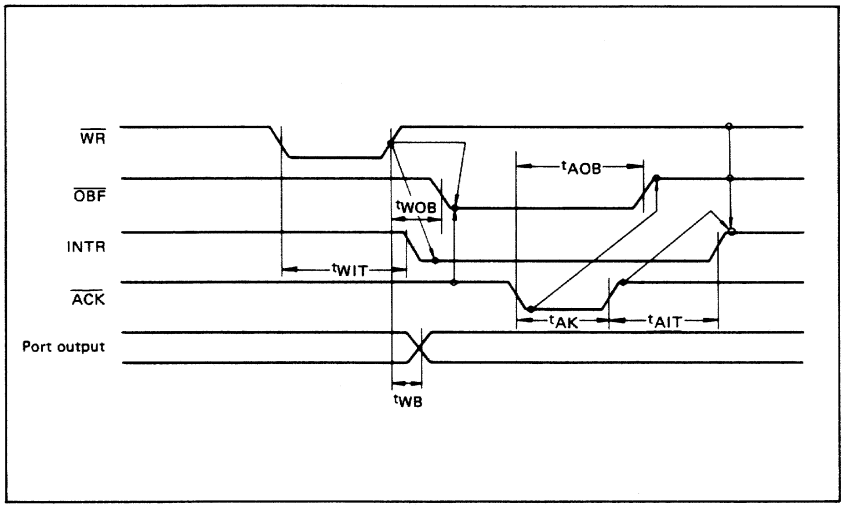


Strobe Input Operation (Mode 1)

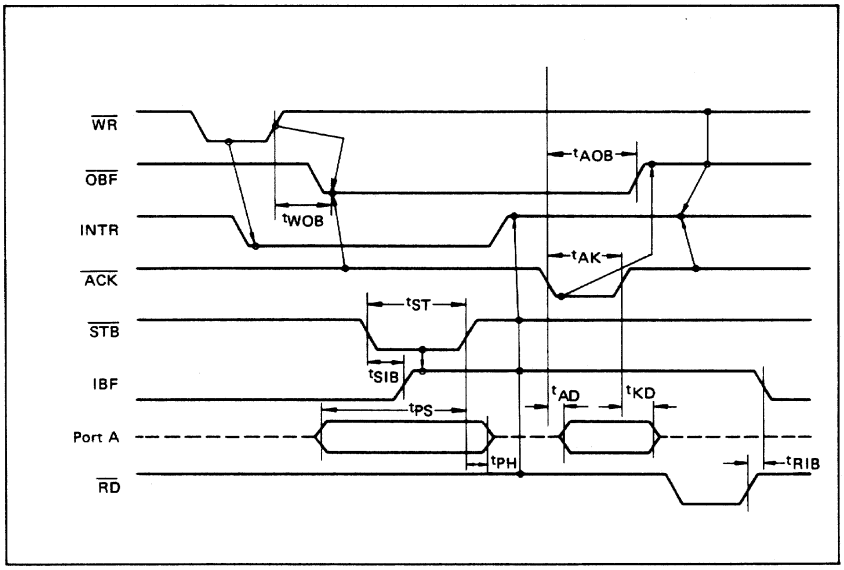


 © National Instruments Corporation

Strobe Output Operation (Mode 1)



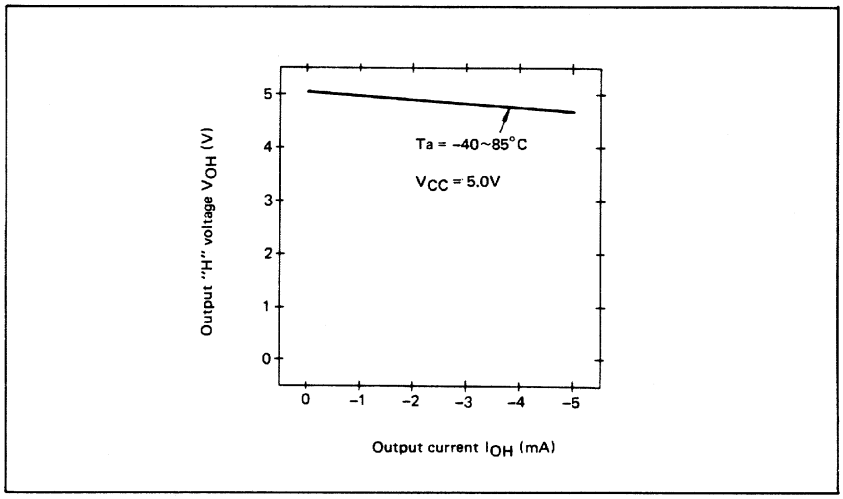
Bidirectional Bus Operation (Mode 2)



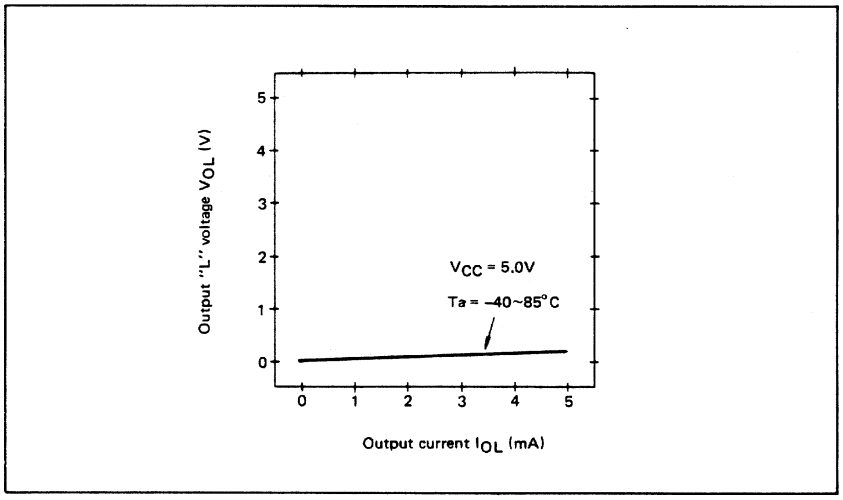
■ I/O-MSM82C55A-2RS/GS/VJS ■

OUTPUT CHARACTERISTICS (REFERENCE VALUE)

1 Output "H" Voltage (V_{OH}) vs. Output Current (I_{OH})



2 Output "L" Voltage (V_{OL}) vs. Output Current (I_{OL})



Note: The direction of flowing into the device is taken as positive for the output current.

 © National Instruments Corporation

FUNCTIONAL DESCRIPTION OF PIN

Pin No.	Item	Input/Output	Function
D7 ~ D0	Bidirectional data bus	Input and output	These are three-state 8-bit bidirectional buses used to write and read data upon receipt of the $\overline{WR}$ and $\overline{RD}$ signals from CPU and also used when control words and bit set/reset data are transferred from CPU to MSM82C55A.
RESET	Reset input	Input	This signal is used to reset the control register and all internal registers when it is in high level. At this time, ports are all made into the input mode (high impedance status). all port latches are cleared to O. and all ports groups are set to mode O.
$\overline{CS}$	Chip select input	Input	When the $\overline{CS}$ is in low level, data transmission is enabled with CPU. When it is in high level, the data bus is made into the high impedance status where no write nor read operation is performed. Internal registers hold their previous status, however.
$\overline{RD}$	Read input	Input	When RD is in low level, data is transferred from MSM82C55A to CPU.
$\overline{WR}$	Write input	Input	When WR is in low level, data or control words are transferred from CPU to MSM82C55A.
A0, A1	Port select input (address)	Input	By combination of A0 and A1, either one is selected from among port A, port B, port C, and control register. These pins are usually connected to low order 2 bits of the address bus.
PA7 ~ PA0	Port A	Input and output	These are universal 8-bit I/O ports. The direction of inputs/outputs can be determined by writing a control word. Especially, port A can be used as a bidirectional port when it is set to mode 2.
PB7 ~ PB0	Port B	Input and output	These are universal 8-bit I/O ports. The direction of inputs/outputs can be determined by writing a control word.
PC7 ~ PC0	Port C	Input and output	These are universal 8-bit I/O ports. The direction of inputs/outputs can be determined by writing a control word as 2 ports with 4 bits each. When port A or port B is used in mode 1 or mode 2 (port A only), they become control pins. Especially when port C is used as an output port, each bit can be set/reset independently.
V_{CC}			+5 V power supply.
GND			GND

BASIC FUNCTIONAL DESCRIPTION

Group A and Group B

When setting a mode to a port having 24 bits, set it by dividing it into two groups of 12 bits each.

- Group A: Port A (8 bits) and high order 4 bits of port C (PC7 ~ PC4)
- Group B: Port B (8 bits) and low order 4 bits of port C (PC3 ~ PC0)

Mode 0, 1, 2

There are 3 types of modes to be set by grouping as follows:

- Mode 0: Basic input operation/output operation (Available for both groups A and B)
- Mode 1: Strobe input operation/output operation (Available for both groups A and B)
- Mode 2: Bidirectional bus operation (Available for group A only)

When used in mode 1 or mode 2, however, port C has bits to be defined as ports for control signal for operation ports (port A for group A and port B for group B) of their respective groups.

Port A, B, C

The internal structure of 3 ports is as follows:

- Port A: One 8-bit data output latch/buffer and one 8-bit data input latch
- Port B: One 8-bit data input/output latch/buffer and one 8-bit data input buffer
- Port C: One 8-bit data output latch/buffer and one 8-bit data input buffer (no latch for input)

Single bit set/reset function for port C

When port C is defined as an output port, it is possible to set (to turn to high level) or reset (to turn to low level) any one of 8 bits individually without affecting other bits.

■ I/O-MSM82C55A-2RS/GS/VJS ■

OPERATIONAL DESCRIPTION

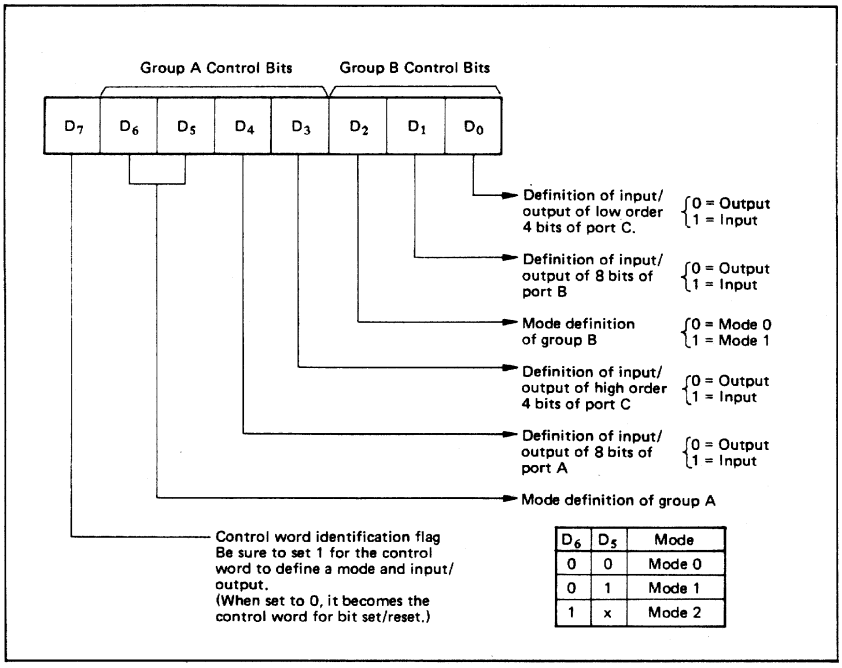
Control Logic

Operations by addresses and control signals, e.g., read and write, etc. are as shown in the table below:

Operation	A1	A0	$\overline{CS}$	$\overline{WR}$	$\overline{RD}$	Operation
Input	0	0	0	1	0	Port A → Data Bus
	0	1	0	1	0	Port B → Data Bus
	1	0	0	1	0	Port C → Data Bus
Output	0	0	0	0	1	Data Bus → Port A
	0	1	0	0	1	Data Bus → Port B
	1	0	0	0	1	Data Bus → Port C
Control	1	1	0	0	1	Data Bus → Control Register
Others	1	1	0	1	0	Illegal Condition
	x	x	1	x	x	Data bus is in the high impedance status.

Setting of Control Word

The control register is composed of 7-bit latch circuit and 1-bit flag as shown below.



D_6	D_5	Mode
0	0	Mode 0
0	1	Mode 1
1	x	Mode 2

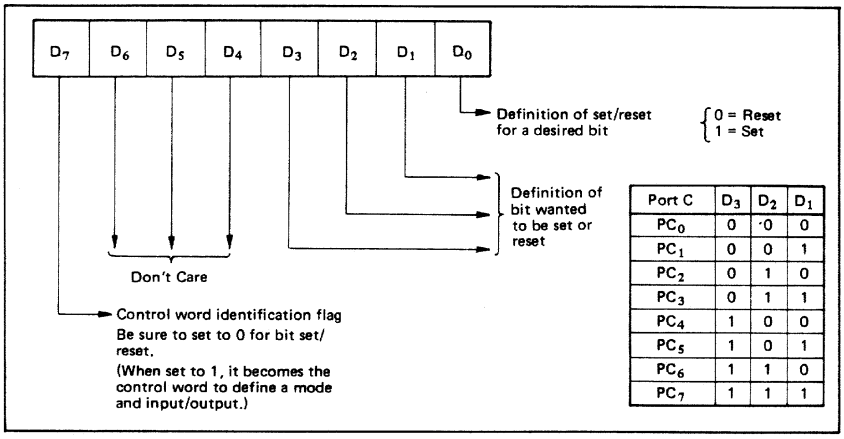
Precaution for mode selection

The output registers for ports A and C are cleared to ϕ each time data is written in the command register and the mode is changed, but the port B state is undefined.

Bit Set/Reset Function

When port C is defined as output port, it is possible to set (set output to 1) or reset (set output to 0) any one of 8 bits without affecting other bits as shown next page.

 © National Instruments Corporation



Port C	D_3	D_2	D_1
PC_0	0	0	0
PC_1	0	0	1
PC_2	0	1	0
PC_3	0	1	1
PC_4	1	0	0
PC_5	1	0	1
PC_6	1	1	0
PC_7	1	1	1

Interrupt Control Function

When the MSM82C55A is used in mode 1 or mode 2, the interrupt signal for the CPU is provided. The interrupt request signal is output from port C. When the internal flip-flop INTE is set beforehand at this time, the desired interrupt request signal is output. When it is reset beforehand, however, the interrupt request signal is not output. The set/reset of the internal flip-flop is made by the bit set/reset operation for port C virtually.

Bit set → INTE is set → Interrupt allowed
Bit reset → INTE is reset → Interrupt inhibited

Operational Description by Mode

1. **Mode 0 (Basic input/output operation)**
   Mode 0 makes the MSM82C55A operate as a basic input port or output port. No control signals such as interrupt request, etc. are required in this mode. All 24 bits can be used as two-8-bit ports and two 4-bit ports. Sixteen combinations are then possible for inputs/outputs. The inputs are not latched, but the outputs are.

Type	Control Word								Group A		Group B	
	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Port A	High Order 4 Bits of Port C	Port B	Low Order 4 Bits of Port C
1	1	0	0	0	0	0	0	0	Output	Output	Output	Output
2	1	0	0	0	0	0	0	1	Output	Output	Output	Input
3	1	0	0	0	0	0	1	0	Output	Output	Input	Output
4	1	0	0	0	0	0	1	1	Output	Output	Input	Input
5	1	0	0	0	1	0	0	0	Output	Input	Output	Output
6	1	0	0	0	1	0	0	1	Output	Input	Output	Input
7	1	0	0	0	1	0	1	0	Output	Input	Input	Output
8	1	0	0	0	1	0	1	1	Output	Input	Input	Input
9	1	0	0	1	0	0	0	0	Input	Output	Output	Output
10	1	0	0	1	0	0	0	1	Input	Output	Output	Input
11	1	0	0	1	0	0	1	0	Input	Output	Input	Output
12	1	0	0	1	0	0	1	1	Input	Output	Input	Input
13	1	0	0	1	1	0	0	0	Input	Input	Output	Output
14	1	0	0	1	1	0	0	1	Input	Input	Output	Input
15	1	0	0	1	1	0	1	0	Input	Input	Input	Output
16	1	0	0	1	1	0	1	1	Input	Input	Input	Input

Note: When used in mode 0 for both groups A and B

■ I/O-MSM82C55A-2RS/GS/VJS ■

2. Mode 1 (Strobe input/output operation)

In mode 1, the strobe, interrupt and other control signals are used when input/output operations are made from a specified port. This mode is available for both groups A and B. In group A at this time, port A is used as the data line and port C as the control signal.

Following is a descrption of the input operation in mode 1.

$\overline{STB}$ (Strobe input)

- When this signal is low level, the data output from terminal to port is fetched into the internal latch of the port. This can be made independent from the CPU, and the data is not output to the data bus until the RD signal arrives from the CPU.

IBF (Input buffer full flag output)

- This is the response signal for the $\overline{STB}$. This signal when turned to high level indicates that data is fetched into the input latch. This signal turns to high level at the falling edge of $\overline{STB}$ and to low level at the rising edge of $\overline{RD}$.

INTR (Interrupt request output)

- This is the interrupt request signal for the CPU of the data fetched into the input latch. It is indicated by high level only when the internal INTE flip-flop is set. This signal turns to high level at the rising edge of the $\overline{STB}$ (IBF = 1 at this time) and low level at the falling edge of the $\overline{RD}$ when the INTE is set.

$INTE_A$ of group A is set when the bit for PC_4 is set, while $INTE_B$ of group B is set when the bit for PC_2 is set.

Following is a description of the output operation of mode 1.

$\overline{OBF}$ (Output buffer full flag output)

- This signal when turned to low level indicates that data is written to the specified port upon receipt of the $\overline{WR}$ signal from the CPU. This signal turns to low level at the rising edge of the WR and high level at the falling edge of the ACK.

$\overline{ACK}$ (Acknowledge input)

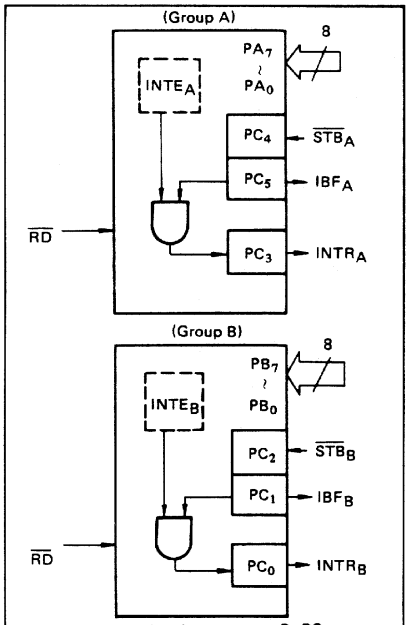
- This signal when turned to low level indicates that the terminal has received data.

INTR (Interrupt request output)

- This is the signal used to interrupt the CPU when a terminal receives data from the CPU via the MSM82C55A-5. It indicates the occurrence of the interrupt in high level only when the internal INTE flip-flop is set. This signal turns to high level at the rising edge of the $\overline{ACK}$ (OBF = 1 at this time) and low level at the falling edge of $\overline{WR}$ when the $INTE_B$ is set.

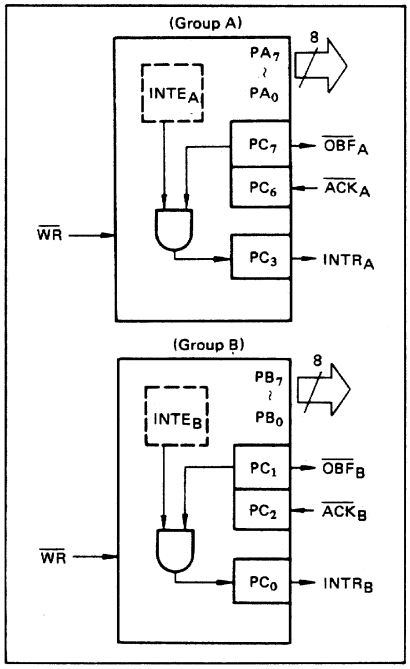
$INTE_A$ of group A is set when the bit for PC_6 is set, while $INTE_B$ of group B is set when the bit for PC_2 is set.

Mode 1 Input



Note: Although belonging to group B, PC_3 operates as the control signal of group A functionally.

Mode 1 output



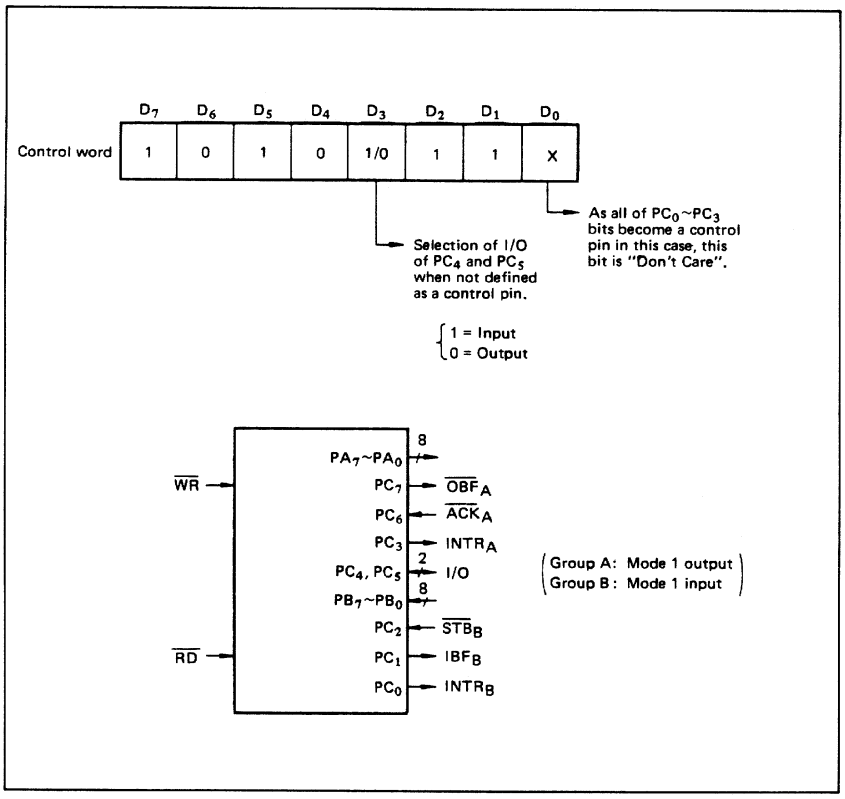
 © National Instruments Corporation

Port C Function Allocation in Mode 1

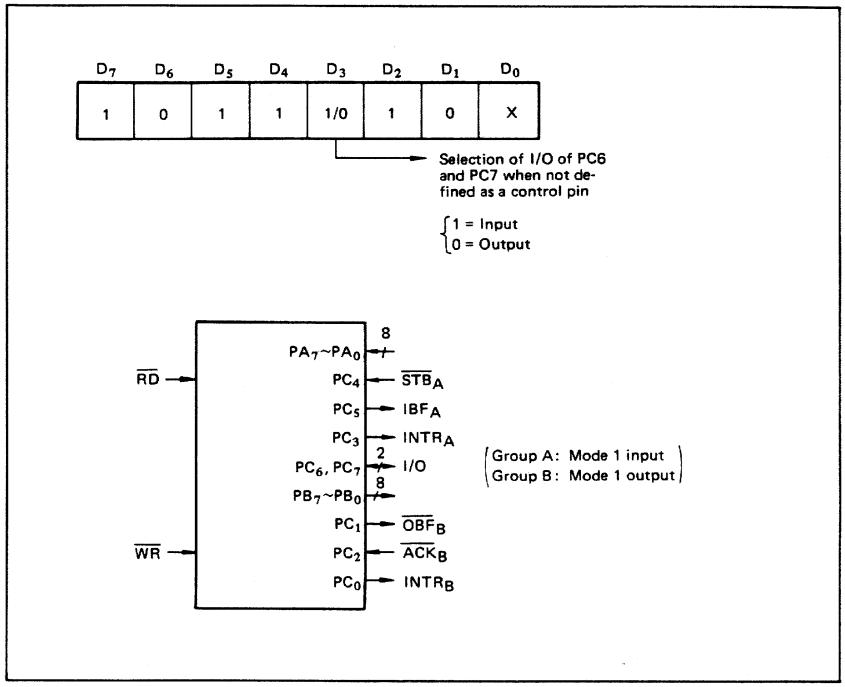
Combination of Input/Output / Port C	Group A: Input Group B: Input	Group A: Input Group B: Output	Group A: Output Group B: Input	Group A: Output Group B: Output
PC_0	$INTR_B$	$INTR_B$	$INTR_B$	$INTR_B$
PC_1	IBF_B	$\overline{OBF}_B$	IBF_B	$\overline{OBF}_B$
PC_2	$\overline{STB}_B$	$\overline{ACK}_B$	$\overline{STB}_B$	$\overline{ACK}_B$
PC_3	$INTR_A$	$INTR_A$	$INTR_A$	$INTR_A$
PC_4	$\overline{STB}_A$	$\overline{STB}_A$	I/O	I/O
PC_5	IBF_A	IBF_A	I/O	I/O
PC_6	I/O	I/O	$\overline{ACK}_A$	$\overline{ACK}_A$
PC_7	I/O	I/O	$\overline{OBF}_A$	$\overline{OBF}_A$

Note: I/O is a bit not used as the control signal, but it is available as a port of mode 0.

Examples of the relation between the control words and pins when used in mode 1 is shown below:
(a) When group A is mode 1 output and group B is mode 1 input.



(b) When group A is mode 1 input and group B is mode 1 output.



3. Mode 2 (Strobe bidirectional bus I/O operation)

In mode 2, it is possible to transfer data in 2 directions through a single 8-bit port. This operation is akin to a combination between input and output operations. Port C waits for the control signal in this case, too. Mode 2 is available only for group A, however.

Next, a description is made on mode 2.

$\overline{OBF}$ (Output buffer full flag output)

- This signal when turned to low level indicates that data has been written to the internal output latch upon receipt of the $\overline{WR}$ signal from the CPU. At this time, port A is still in the high impedance status and the data is not yet output to the outside. This signal turns to low level at the rising edge of the $\overline{WR}$ and high level at the falling edge of the $\overline{ACK}$.

$\overline{ACK}$ (Acknowledge input)

- When a low level signal is input to this pin, the high impedance status of port A is cleared, the buffer is enabled, and the data written to the internal output latch is output to port A. When the input returns to high level, port A is made into the high impedance status.

$\overline{STB}$ (Strobe input)

- When this signal turns to low level, the data output to the port from the pin is fetched into the internal input latch. The data is output to the data bus upon receipt of the RD signal from the CPU, but it remains in the high impedance status until then.

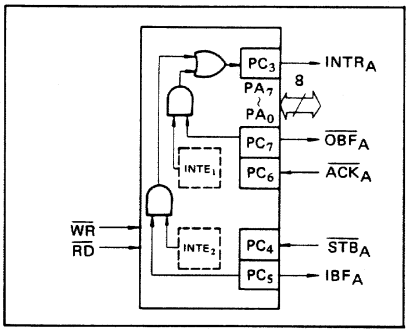
IBF (Input buffer full flag output)

- This signal when turned to high level indicates that data from the pin has been fetched into the input latch. This signal turns to high level at the falling edge of the $\overline{STB}$ and low level at the rising edge of the $\overline{RD}$.

INTR (Interrupt request output)

- This signal is used to interrupt the CPU and its operation in the same as in mode 1. There are two INTE flip-flops internally available for input and output to select either interrupt of input or output operation. The INTE1 is used to control the interrupt request for output operation and it can be reset by the bit set for PC6. INTE2 is used to control the interrupt request for the input operation and it can be set by the bit set for PC4.

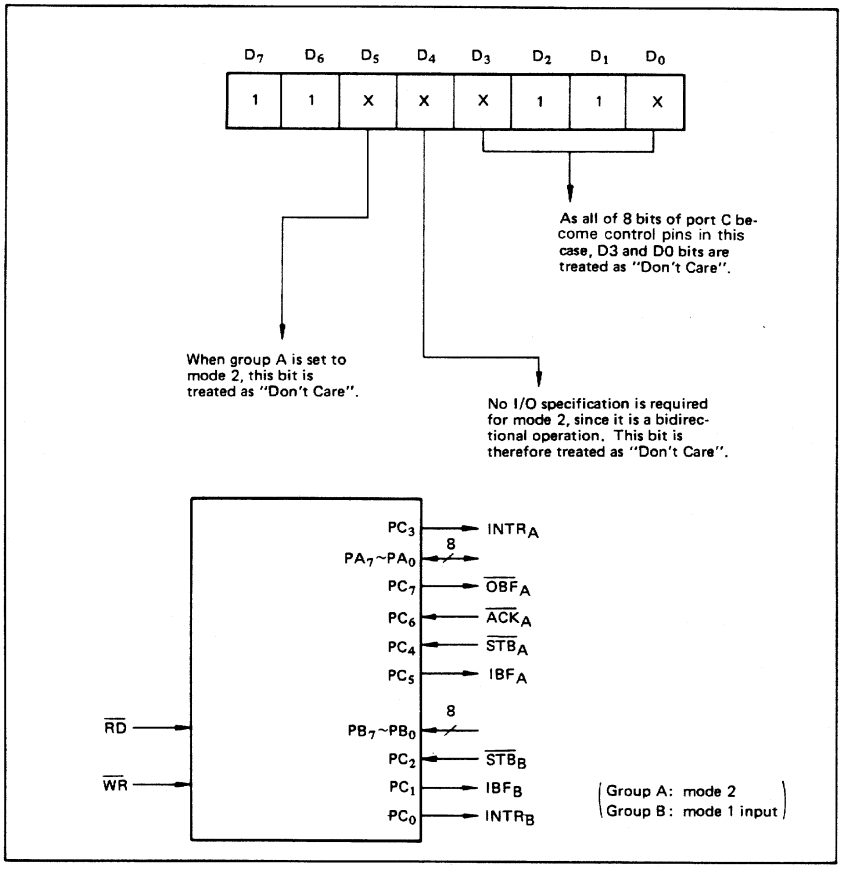
Mode 2 I/O Operation



Port C Function Allocation in Mode 2

Port C	Function
PC_0	Confirmed to the group B mode
PC_1	
PC_2	
PC_3	$INTR_A$
PC_4	$\overline{STB}_A$
PC_5	IBF_A
PC_6	$\overline{ACK}_A$
PC_7	$\overline{OBF}_A$

Following is an example of the relation between the control word and the pin when used in mode 2. When input in mode 2 for group A and in mode 1 for group B.



■ I/O-MSM82C55A-2RS/GS/VJS ■

4. When Group A is Different in Mode from Group B

Group A and group B can be used by setting them in different modes each other at the same time. When either group is set to mode1 or mode 2, it is possible to set the one not defined as a control pin in port C to both input and output as a port which operates in mode 0 at the 3rd and 0th bits of the control word.

(Mode combinations that define no control bit at port C)

	Group A	Group B	Port C							
			PC_7	PC_6	PC_5	PC_4	PC_3	PC_2	PC_1	PC_0
1	Mode 1 input	Mode 0	I/O	I/O	IBF_A	$\overline{STB}_A$	$INTR_A$	I/O	I/O	I/O
2	Mode 0 output	Mode 0	$\overline{OBF}_A$	$\overline{ACK}_A$	I/O	I/O	$INTR_A$	I/O	I/O	I/O
3	Mode 0	Mode 1 input	I/O	I/O	I/O	I/O	I/O	$\overline{STB}_B$	IBF_B	$INTR_B$
4	Mode 0	Mode 1 output	I/O	I/O	I/O	I/O	I/O	$\overline{ACK}_B$	$\overline{OBF}_B$	$INTR_B$
5	Mode 1 input	Mode 1 input	I/O	I/O	IBF_A	$\overline{STB}_A$	$INTR_A$	$\overline{STB}_B$	IBF_B	$INTR_B$
6	Mode 1 input	Mode 1 output	I/O	I/O	IBF_A	$\overline{STB}_A$	$INTR_A$	$\overline{ACK}_B$	$\overline{OBF}_B$	$INTR_B$
7	Mode 1 output	Mode 1 input	$\overline{OBF}_A$	$\overline{ACK}_A$	I/O	I/O	$INTR_A$	$\overline{STB}_B$	IBF_B	$INTR_B$
8	Mode 1 output	Mode 1 output	$\overline{OBF}_A$	$\overline{ACK}_A$	I/O	I/O	$INTR_A$	$\overline{ACK}_B$	$\overline{OBF}_B$	$INTR_B$
9	Mode 2	Mode 0	$\overline{OBF}_A$	$\overline{ACK}_A$	IBF_A	$\overline{STB}_A$	$INTR_A$	I/O	I/O	I/O
			Controlled at the 3rd bit (D3) of the control word				Controlled at the 0th bit (D0) of the control word			

When the I/O bit is set to input in this case, it is possible to access data by the normal port C read operation.
When set to output, PC7 ~ PC4 bits can be accessed by the bit set/reset function only. Meanwhile, 3 bits from PC2 to PC0 can be accessed by normal write operation.

The bit set/reset function can be used for all of PC3 ~ PC0 bits. Note that the status of port C varies according to the combination of modes like this.

5. Port C Status Read

When port C is used for the control signal, that is, in either mode 1 or mode 2, each control signal and bus status signal can be read out by reading the content of port C.
The status read out is as follows:

	Group A	Group B	Status read on the data bus							
			D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	Mode 1 input	Mode 0	I/O	I/O	IBF_A	$INTE_A$	$INTR_A$	I/O	I/O	I/O
2	Mode 1 output	Mode 0	$\overline{OBF}_A$	$INTE_A$	I/O	I/O	$INTR_A$	I/O	I/O	I/O
3	Mode 0	Mode 1 input	I/O	I/O	I/O	I/O	I/O	$INTE_B$	IBF_B	$INTR_B$
4	Mode 0	Mode 1 output	I/O	I/O	I/O	I/O	I/O	$INTE_B$	$\overline{OBF}_B$	$INTR_B$
5	Mode 1 input	Mode 1 input	I/O	I/O	IBF_A	$INTE_A$	$INTR_A$	$INTE_B$	IBF_B	$INTR_B$
6	Mode 1 input	Mode 1 output	I/O	I/O	IBF_A	$INTE_A$	$INTR_A$	$INTE_B$	$\overline{OBF}_B$	$INTR_B$
7	Mode 1 output	Mode 1 input	$\overline{OBF}_A$	$INTE_A$	I/O	I/O	$INTR_A$	$INTE_B$	IBF_B	$INTR_B$
8	Mode 1 output	Mode 1 output	$\overline{OBF}_A$	$INTE_A$	I/O	I/O	$INTR_A$	$INTE_B$	$\overline{OBF}_B$	$INTR_B$
9	Mode 2	Mode 0	$\overline{OBF}_A$	$INTE_1$	IBF_A	$INTE_2$	$INTR_A$	I/O	I/O	I/O
10	Mode 2	Mode 1 input	$\overline{OBF}_A$	$INTE_1$	IBF_A	$INTE_2$	$INTR_A$	$INTE_B$	IBF_B	$INTR_B$
11	Mode 2	Mode 1 output	$\overline{OBF}_A$	$INTE_1$	IBF_A	$INTE_2$	$INTR_A$	$INTE_B$	$\overline{OBF}_B$	$INTR_B$

6. Reset of MSM82C55A

Be sure to keep the RESET signal at power ON in the high level at least for 50 μs. Subsequently, it becomes the input mode at a high level pulse above 500 ns.

Note: Comparison of MSM82C55A-5 and MSM82C55A-2

MSM82C55A-5

After a write command is executed to the command register, the internal latch is cleared in PORTA PORTC. For instance, 00H is output at the beginning of a write command when the output port is assigned. However, if PORTB is not cleared at this time, PORTB is unstable. In other words, PORTB only outputs ineffective data (unstable value according to the device) during the period from after a write command is executed till the first data is written to PORTB.

MSM82C55A-2

After a write command is executed to the command register, the internal latch is cleared in All Ports(PORTA,PORTB,PORTC). 00H is ontput at the beginning of a write command when the output port is assigned.

Appendix F
Customer Communication

For your convenience, this appendix contains forms to help you gather the information necessary to help us solve technical problems you might have as well as a form you can use to comment on the product documentation. Filling out a copy of the *Technical Support Form* before contacting National Instruments helps us help you better and faster.

National Instruments provides comprehensive technical assistance around the world. In the U.S. and Canada, applications engineers are available Monday through Friday from 8:00 a.m. to 6:00 p.m. (central time). In other countries, contact the nearest branch office. You may fax questions to us at any time.

Corporate Headquarters
(512) 795-8248
Technical support fax: (800) 328-2203
(512) 794-5678

Branch Offices	**Phone Number**	**Fax Number**
Australia	03 9 879 9422	03 9 879 9179
Austria	0662 45 79 90 0	0662 45 79 90 19
Belgium	02 757 00 20	02 757 03 11
Canada Ontario)	519 622 9310	519 622 9311
Canada (Quebec)	514 694 8521	514 694 4399
Denmark	45 76 26 00	45 76 71 11
Finland	90 527 2321	90 502 2930
France	1 48 14 24 24	1 48 14 24 14
Germany	089 741 31 30	089 714 60 35
Hong Kong	2645 3186	2686 8505
Italy	02 48301892	02 48301915
Japan	03 5472 2970	03 5472 2977
Korea	02 596 7456	02 596 7455
Mexico	95 800 010 0793	05 404 0890
Netherlands	03480 33466	03480 30673
Norway	32 84 84 00	32 84 86 00
Singapore	2265886	2265887
Spain	91 640 0085	91 640 0533
Sweden	08 730 49 70	08 730 43 70
Switzerland	056 20 51 51	056 20 51 55
Taiwan	02 377 1200	02 737 4644
U.K.	01635 523545	01635 523154

Documentation Comment Form

National Instruments encourages you to comment on the documentation supplied with our products. This information helps us provide quality products to meet your needs.

Title: **DAQCard™-1200 Register-Level Programmer Manual**

Edition Date: **October 1995**

Part Number: **340988A-01**

Please comment on the completeness, clarity, and organization of the manual.

If you find errors in the manual, please record the page numbers and describe the errors.

Thank you for your help.

Name _____

Title _____

Company _____

Address _____

_____

Phone (_____) _____

Mail to: Technical Publications
National Instruments Corporation
6504 Bridge Point Parkway, MS 53-02
Austin, TX 78730-5039

Fax to: Technical Publications
National Instruments Corporation
MS 53-02
(512) 794-5678

Glossary

The following metric system prefixes are used with abbreviations for units of measure:

Prefix	Meaning	Value
p-	pico-	10^{-12}
n-	nano-	10^{-9}
µ-	micro-	10^{-6}
m-	milli-	10^{-3}
k-	kilo-	10^{3}
M-	mega-	10^{6}
G-	giga-	10^{12}

2SDAC	two's complement DAC bit
A/D	analog-to-digital
ADC	analog-to-digital converter
ADCUNI/BI*	ADC unipolar/bipolar bit
BCD	binary-coded decimal/select bit
CALDACLD	calibration DAC load bit
CIS	Card Information Structure
CMOS	complementary metallic oxide semiconductor
CNTINT	counter interrupt status bit
CNTINTEN	counter interrupt enable bit
CONVPROG	conversion in progress bit
CW	control word bit
D	data bit
DAC0UNI/BI*	DAC0 unipolar/bipolar bit
DAC1UNI/BI*	DAC1 unipolar/bipolar bit
DAQ	data acquisition
DATAERR	data error bit
DAVAIL	data available bit
DC	direct current
DIN	digital input bit/signal
DIOINTEN	DIO interrupt enable bit
DISABDAQ	disable data acquisition bit
DITHEREN	dither enable bit
DOUT	digital output bit/signal
DQINTEN	DAQ interrupt enable
ECLKRCV	external clock receive bit
ECLKDRV	external clock drive bit
EEPROMCS	EEPROM chip select bit
EOIRCV	external output interval clock receive bit
ERRINTEN	error interrupt enable bit
EXTCONV*	external conversion signal
EXTGATA0	external GateA0

FIFO	first-in-first-out
FIFOHF	FIFO half-full bit
FIFOINT	FIFO interrupt bit
FIFOINTEN	FIFO interrupt enable bit
GAIN	gain bit
GATA0	Gate A0 bit
hex	hexadecimal
HFINTEN	half-full interrupt enable bit
HWTRIG	hardware trigger bit
Hz	hertz
INTSCAN	interval scan bit
I/O	input/output
LDAC	Loaded DAC bit
LSB	least significant bit
M	counter mode select bit
MA	multiplexed analog bit
MB	megabytes of memory
MSB	most significant bit
OUT	counter output
OUTA1	Output A1 bit
OVERFLOW	overflow error status bit
OVERRUN	overrun error status bit
PCMCIA	Personal Computer Memory Card International Association
PRETRIG	pretrigger bit
PROMOUT	EEPROM output bit
RL	read/load select bit
RAM	random-access memory
rms	root mean square
RSE*/NRSE	referenced single-ended/nonreferenced single-ended bit
SC	counter select bit
SCANEN*	scan enable bit
SCLK	serial clock bit
SDATA	serial data bit
SE*/D	single-ended/differential bit
SWTRIG	software trigger bit
TBSEL	time base select bit
TTL	transistor-transistor logic
TWOSCOMP	two’s complement bit
V	volts
WRTPRT	write protect bit
X	don’t care bit

Index

Numbers/Symbols

2SDAC0 bit, 2-7
2SDAC1 bit, 2-7
82C53 Counter/Timer Register Groups A and B
  configuration, 1-3
  Counter A Mode Register, 2-25, 3-6
  Counter A0 Data Register, 2-24, 3-6, 3-7, 3-8
  Counter A1 Data Register, 2-24, 3-7, 3-9
  Counter A2 Data Register, 2-25, 3-16
  Counter B Mode Register, 2-28
  Counter B0 Data Register, 2-26, 3-7, 3-8
  Counter B1 Data Register, 2-27, 3-7, 3-9
  Counter B2 Data Register, 2-27
  OKI MSM82C53 data sheet, D-1
  overview, 2-23
  programming general-purpose counter/timers, 3-17
  register map, 2-2
  Timer Interrupt Clear Register, 2-26
82C55A Digital I/O Register Group
  Digital Control Register, 2-31
  OKI MSM82C55A data sheet, E-1
  overview, 2-29
  Port A Register, 2-30
  Port B Register, 2-30
  Port C Register, 2-31
  register map, 2-2

A

A/D Clear Register, 2-20
A/D conversions, single. *See under* analog input circuitry.
A/D FIFO Register
  description, 2-18 to 2-19
  obtaining A/D conversion results, 3-6
  servicing DAQ operation, 3-10
ADCUNI/BI* bit
  configuring analog output circuitry, 3-5
  description, 2-14
analog input circuitry
  calibration, 4-3 to 4-6
    bipolar input calibration procedure, 4-4 to 4-5
    unipolar input calibration procedure, 4-5 to 4-6
  configuration, 1-2
  programming for single A/D conversions, 3-4 to 3-6
    clearing analog input circuitry, 3-4 to 3-5
    configuring analog input circuitry, 3-5
    performing single A/D conversions, 3-6
Analog Input Register Group
  A/D Clear Register, 2-20
  A/D FIFO Register, 2-18 to 2-19, 3-6, 3-10
  overview, 2-17
  register map, 2-2
analog output circuitry
  calibration, 4-6 to 4-8
    bipolar output calibration procedure, 4-7 to 4-8
    unipolar output calibration procedure, 4-8
  configuration, 1-2
  programming, 3-14 to 3-16
    analog output voltage versus digital code
      bipolar mode, two's complement coding (table), 3-15
      unipolar mode, straight binary coding (table), 3-15
    configuring analog output circuitry, 3-14 to 3-15
    DAC interrupt programming, 3-16
    update mode of analog output circuitry, 3-15 to 3-16
Analog Output Register Group
  DAC0 Low-Byte, DAC0 High-Byte, DAC1 Low-Byte, and DAC1 High-Byte Registers, 2-22, 3-15, 3-16
  overview, 2-21
  register map, 2-2
analog output voltage versus digital code
  bipolar mode, two's complement coding (table), 3-15
  unipolar mode, straight binary coding (table), 3-15

B

bipolar input calibration, 4-4 to 4-5
  gain calibration, 4-5
  higher gains, 4-5
  postgain offset calibration, 4-5
  pregain offset coarse calibration, 4-4
  pregain offset fine calibration, 4-4
bipolar input polarity, selecting, 3-5
bipolar mode, two's complement coding (table), 3-15
bipolar output calibration, 4-7 to 4-8
  gain calibration, 4-7
  offset calibration, 4-8
bits
  2SDAC0, 2-7
  2SDAC1, 2-7
  ADCUNI/BI*, 2-14, 3-5
  CALDACLD, 2-12, 3-3
  CNTINT, 2-15, 3-16
  CNTINTEN, 2-9, 3-16
  CW<7..0>, 2-31
  D<7..0>
    A/D FIFO Register, 2-19
    Counter A0 Data Register, 2-24
    Counter A1 Data Register, 2-24
    Counter A2 Data Register, 2-25
    Counter B0 Data Register, 2-26
    Counter B1 Data Register, 2-27
    Counter B2 Data Register, 2-27
    DAC0 Low-Byte, DAC0 High_Byte, DAC1 Low-Byte, and DAC1 High-Byte Registers, 2-22
    Interval Counter Data Register, 2-33
    Port A Register, 2-30
    Port B Register, 2-30
    Port C Register, 2-31
  D<11..8>, 2-22
  D<15..8>, 2-19
  D<15..12>, 2-22
  DAC0UNI/BI*, 2-13, 3-14
  DAC1UNI/BI*, 2-13, 3-14
  DAVAIL, 2-15, 3-6
  DIOINTEN, 2-9
  DITHEREN, 2-12, 4-4
  DQINTEN, 2-13, 3-13
  ECLKDRV, 2-10
  ECLKRCV, 2-10, 3-4, 3-11
  EEPROMCS, 2-12, 3-3, 4-2
  EOIRCV, 2-11, 3-9, 3-12
  ERRINTEN, 2-9, 3-13
  EXTGATA0, 2-15
  FIFOHF*, 2-16, 3-6
  FIFOINTEN, 2-9, 3-13
  GAIN<2..0>, 2-5, 3-5
  GATA0, 2-15
  HFINTEN, 2-13, 3-13
  HWTRIG, 2-8, 3-11
  INTSCAN, 2-11, 3-9, 3-12
  LDAC0, 2-7, 3-15, 3-16
  LDAC1, 2-7, 3-15, 3-16
  MA<2..0>, 2-6, 3-5
  OUTA1, 2-16
  OVERFLOW, 2-15, 3-6
  OVERRUN, 2-15, 3-6
  PRETRIG, 2-8, 3-12
  PROMOUT, 2-16
  RSE*/NRSE, 2-14, 4-4
  SCANEN, 2-5, 3-5
  SCLK, 2-12, 3-3, 3-4, 4-2
  SDATA, 2-12, 3-4, 4-2
  SE*/D, 2-10, 4-4
  SWTRIG, 2-8, 3-4, 3-10, 3-12
  TBSEL, 2-8, 3-8
  TWOSCMP, 2-5, 3-5
  WRTPRT, 2-12, 3-3, 4-2
board configuration, 1-2 to 1-3

C

CALDACLD bit
  description, 2-12
  writing calibration constant to CALDAC, 3-3
calibration
  analog input calibration, 4-3 to 4-6
    bipolar input calibration procedure, 4-4 to 4-5
    unipolar input calibration procedure, 4-5 to 4-6
  analog output calibration, 4-6 to 4-8
    bipolar output calibration procedure, 4-7 to 4-8
    unipolar output calibration procedure, 4-8
  calibration DACs (CALDACs), 4-2 to 4-3
    analog input circuitry characteristics (table), 4-2
    analog output circuitry characteristics (table), 4-3
    writing calibration constant to CALDAC, 3-3 to 3-4
  EEPROM map, 4-9 to 4-13

initializing DAQCard-1200 circuitry, 3-3 to 3-4
storing user-defined constants, 4-1 to 4-2
card information structure (CIS), A-1 to A-2
Cardware software, 3-1
CLKB1 signal, 3-9
CNTINT bit
DAC interrupt programming, 3-16
description, 2-15
CNTINTEN bit
DAC interrupt programming, 3-16
description, 2-9
programming update mode of analog output circuitry, 3-16
Command Register 1, 2-5 to 2-6
Command Register 2, 2-7 to 2-8
Command Register 3, 2-9
Command Register 4, 2-10 to 2-11
Command Register 5, 2-12
Command Register 6, 2-13 to 2-14
configuration, 1-2 to 1-3
analog input configuration, 1-2
analog output configuration, 1-2
counter configuration, 1-3
digital I/O configuration, 1-3
Configuration and Status Register Group
Command Register 1, 2-5 to 2-6
Command Register 2, 2-7 to 2-8
Command Register 3, 2-9
Command Register 4, 2-10 to 2-11
Command Register 5, 2-12
Command Register 6, 2-13 to 2-14
overview, 2-4
register map, 2-2
Status Register 1, 2-15
Status Register 2, 2-16
continuous DAQ mode, 3-7
controlled DAQ mode, 3-7
Counter A Mode Register
description, 2-25
performing single A/D conversion, 3-6
Counter A0 Data Register
DAQ operation using internal timing, 3-7
description, 2-24
performing single A/D conversion, 3-6
programming, 3-8
Counter A1 Data Register
DAQ operations using internal timing, 3-7
description, 2-24
programming, 3-9
Counter A2 Data Register
description, 2-25
programming update mode of analog output circuitry, 3-16
Counter B Mode Register, 2-28
Counter B0 Data Register
DAQ operations using internal timing, 3-7
description, 2-26
programming, 3-8
Counter B1 Data Register
DAQ operations using internal timing, 3-7
description, 2-27
programming, 3-9
Counter B2 Data Register, 2-27
counter/timers. *See* 82C53 Counter/Timer Register Groups A and B; 82C55A Digital I/O Register Group.
customer communication, *xi,* F-1
CW<7..0> bits, 2-31

D

D<7..0> bits
A/D FIFO Register, 2-19
Counter A0 Data Register, 2-24
Counter A1 Data Register, 2-24
Counter A2 Data Register, 2-25
Counter B0 Data Register, 2-26
Counter B1 Data Register, 2-27
Counter B2 Data Register, 2-27
DAC0 Low-Byte, DAC0 High-Byte, DAC1 Low-Byte, and DAC1 High-Byte Registers, 2-22
Interval Counter Data Register, 2-33
Port A Register, 2-30
Port B Register, 2-30
Port C Register, 2-31
D<11..8> bits, 2-22
D<15..8> bits, 2-19
D<15..12> bits, 2-22
DAC interrupt programming, 3-16
DAC0 High-Byte Register
DAC interrupt programming, 3-16
description, 2-22
programming update mode of analog output circuitry, 3-15
DAC0 Low-Byte Register
DAC interrupt programming, 3-16
description, 2-22
programming update mode of analog output circuitry, 3-15
DAC0UNI/BI* bit

configuring analog output circuitry, 3-14
description, 2-13
DAC1 High-Byte Register
DAC interrupt programming, 3-16
description, 2-22
programming update mode of analog output circuitry, 3-15
DAC1 Low-Byte Register
DAC interrupt programming, 3-16
description, 2-22
programming update mode of analog output circuitry, 3-15
DAC1UNI/BI* bit
configuring analog output circuitry, 3-14
description, 2-13
DAQ interrupt programming, 3-13
DAQ operation, programming
continuous DAQ mode, 3-7
controlled DAQ mode, 3-7
freerun DAQ mode, 3-7
interval DAQ mode, 3-7
using external timing, 3-11 to 3-12
EXTCONV* instead of Counter A0, 3-11
EXTTRIG in posttrigger mode, 3-11
EXTTRIG in pretrigger mode, 3-12
OUTB1 instead of Counter B1, 3-12
using internal timing, 3-6 to 3-10
Counter A0 and Counter B0, 3-8
Counter A1, 3-9
Counter B1 and Interval Counter Register, 3-9 to 3-10
servicing DAQ operation, 3-10
triggering DAQ operation, 3-10
DAQCard-1200
card information structure (CIS), A-1 to A-2
general characteristics, 1-1
initializing circuitry, 3-2 to 3-4
DAQCONF program, 3-1
DAVAIL bit
description, 2-15
single A/D conversions, 3-6
Digital Control Register, 2-31
digital I/O circuitry
configuration, 1-3
programming, 3-17
DIOINTEN bit, 2-9
DITHEREN bit
analog input calibration, 4-4
description, 2-12
documentation
conventions used in manual, *x*
National Instruments documentation, *x-xi*
organization of manual, *ix-x*
related documentation, *xi*
DQINTEN bit
DAQ interrupt programming, 3-13
description, 2-13

E

ECLKDRV bit, 2-10
ECLKRCV bit
clearing analog input circuitry, 3-4
description, 2-10
programming DAQ operation, 3-11
EEPROM
accidental overwriting (note), 4-1
calibration map, 4-9 to 4-13
reading single byte from EEPROM, 3-3
storing user-defined constants, 4-2
writing single byte to EEPROM, 3-3
EEPROMCS bit
description, 2-12
reading single byte from EEPROM, 3-3
storing user-defined constants, 4-2
EOIRCV bit
DAQ operations using internal timing, 3-9
DAQ operations using OUTB1, 3-12
description, 2-11
ERRINTEN bit
DAQ interrupt programming, 3-13
description, 2-9
EXTCONV* signal
clearing analog input circuitry, 3-4
programming DAQ operation, 3-11
EXTGATA0 bit, 2-15
EXTTRIG signal
clearing analog input circuitry, 3-4
programming DAQ operation
posttrigger mode, 3-11
pretrigger mode, 3-12
EXTUPDATE* signal
DAC interrupt programming, 3-16
programming update mode of analog output circuitry, 3-15 to 3-16

F

fax and phone technical support, F-1
FIFOHF* bit

description, 2-16
single A/D conversions, 3-6
FIFOINTEN bit
DAQ interrupt programming, 3-13
description, 2-9
freerun DAQ mode, 3-7
Fujitsu MB88341/MB88342 data sheet, B-1

G

GAIN<2..0> bits
configuring analog input circuitry, 3-5
description, 2-5
GATA0 bit, 2-15
GATB0 signal, 3-7
GATB1 signal, 3-7
general-purpose counter/timers, programming, 3-17

H

HFINTEN bit
DAQ interrupt programming, 3-13
description, 2-13
HWTRIG bit
description, 2-8
programming DAQ operation, 3-11

I

initializing DAQCard-1200 circuitry, 3-2 to 3-4
input mode, selecting, 3-5
input polarity, selecting, 3-5
interrupts
DAC interrupt programming, 3-16
DAQ interrupt programming, 3-13
Interval Counter Register Group
Interval Counter Data Register, 2-33
Interval Counter Strobe Register, 2-33
overview, 2-32
programming, 3-9
register map, 2-2
interval DAQ mode, 3-7
INTSCAN bit
DAQ operations using internal timing, 3-9
DAQ operations using OUTB1, 3-12
description, 2-11

L

LDAC0 bit
description, 2-7
programming update mode of analog output circuitry, 3-15, 3-16
LDAC1 bit
description, 2-7
programming update mode of analog output circuitry, 3-15, 3-16

M

MA<2..0> bits
configuring analog input circuitry, 3-5
description, 2-6
manual. *See* documentation.

O

OKI MSM82C53 data sheet, D-1
OKI MSM82C55A data sheet, E-1
OUTA0 signal
performing single A/D conversions, 3-6
programming Counter A0 and Counter B0, 3-8
OUTA1 bit, 2-16
OUTA2 signal
DAC interrupt programming, 3-16
programming update mode of analog output circuitry, 3-16
OUTB1 for programming DAQ operation, 3-12
OVERFLOW bit
description, 2-15
single A/D conversions, 3-6
overflow condition, DAQ operations, 3-10
OVERRUN bit
description, 2-15
single A/D conversions, 3-6
overrun condition, DAQ operations, 3-10

P

PCCARD.EXE, 3-1
PCENABLE.EXE, 3-1

PCMCIA card initialization, 3-1 to 3-2
polarity, selecting, 3-5
Port A Register, 2-30
Port B Register, 2-30
Port C Register, 2-31
PRETRIG bit
  description, 2-8
  programming DAQ operation in pretrigger mode, 3-12
programming. *See also* registers.
  analog input circuitry for single A/D conversions, 3-4 to 3-6
    clearing analog input circuitry, 3-4 to 3-5
    configuring analog input circuitry, 3-5
    performing single A/D conversions, 3-6
  analog output circuitry, 3-14 to 3-16
    configuring analog output circuitry, 3-14 to 3-15
    DAC interrupt programming, 3-16
    update mode of analog output circuitry, 3-15 to 3-16
  DAQ interrupt programming, 3-13
  DAQ operation using external timing, 3-11 to 3-12
    EXTCONV* instead of Counter A0, 3-11
    EXTTRIG in posttrigger mode, 3-11
    EXTTRIG in pretrigger mode, 3-12
    OUTB1 instead of Counter B1, 3-12
  DAQ operation using internal timing, 3-6 to 3-10
    Counter A0 and Counter B0, 3-8
    Counter A1, 3-9
    Counter B1 and Interval Counter Register, 3-9 to 3-10
    servicing DAQ operation, 3-10
    triggering DAQ operation, 3-10
  digital I/O circuitry, 3-17
  general-purpose counter/timers, 3-17
  initializing DAQCard-1200 circuitry, 3-2 to 3-4
  PCMCIA card initialization, 3-1 to 3-2
  register programming considerations, 3-2
PROMOUT bit, 2-16

R

registers
  82C53 Counter/Timer Register Groups A and B
    configuration, 1-3
    Counter A Mode Register, 2-25, 3-6
    Counter A0 Data Register, 2-24, 3-6, 3-7, 3-8
    Counter A1 Data Register, 2-24, 3-7, 3-9
    Counter A2 Data Register, 2-25, 3-16
    Counter B Mode Register, 2-28
    Counter B0 Data Register, 2-26, 3-7, 3-8
    Counter B1 Data Register, 2-27, 3-7, 3-9
    Counter B2 Data Register, 2-27
    overview, 2-23
    Timer Interrupt Clear Register, 2-26
  82C55A Digital I/O Register Group
    Digital Control Register, 2-31
    OKI MSM82C55A data sheet, E-1
    overview, 2-29
    Port A Register, 2-30
    Port B Register, 2-30
    Port C Register, 2-31
  Analog Input Register Group
    A/D Clear Register, 2-20
    A/D FIFO Register, 2-18 to 2-19, 3-6, 3-10
    overview, 2-17
  Analog Output Register Group
    DAC0 Low-Byte, DAC0 High-Byte, DAC1 Low-Byte, and DAC1 High-Byte Registers, 2-22
    overview, 2-21
  Configuration and Status Register Group
    Command Register 1, 2-5 to 2-6
    Command Register 2, 2-7 to 2-8
    Command Register 3, 2-9
    Command Register 4, 2-10 to 2-11
    Command Register 5, 2-12
    Command Register 6, 2-13 to 2-14
    overview, 2-4
    Status Register 1, 2-15
    Status Register 2, 2-16
  Interval Counter Register Group
    Interval Counter Data Register, 2-33
    Interval Counter Strobe Register, 2-33
    overview, 2-32
    programming, 3-9
  overview, 2-3
  programming considerations, 3-2. *See also* programming.

register map, 2-2
Start Convert, 3-6
resetting DAQCard-1200 circuitry, 3-2
RSE*/NRSE bit
analog input calibration, 4-4
description, 2-14

S

SCANEN bit
configuring analog input circuitry, 3-5
description, 2-5
SCLK bit
description, 2-12
storing user-defined constants, 4-2
writing single byte to EEPROM, 3-3, 3-4
SDATA bit
description, 2-12
storing user-defined constants, 4-2
writing calibration constant to CALDAC, 3-4
SE*/D bit
analog input calibration, 4-4
description, 2-10
single A/D conversions. *See under* analog input circuitry.
Start Convert Register, 3-6
Status Register 1, 2-15
Status Register 2, 2-16
SWTRIG bit
clearing analog input circuitry, 3-4
description, 2-8
programming DAQ operation in pretrigger mode, 3-12
triggering DAQ operation, 3-10

T

TBSEL bit
DAQ operations using internal timing, 3-8
description, 2-8
technical support, F-1
Timer Interrupt Clear Register, 2-26
TWOSCMP bit
configuring analog input circuitry, 3-5
description, 2-5

U

unipolar input calibration, 4-5 to 4-6
gain calibration, 4-6
postgain offset calibration, 4-6
pregain offset calibration, 4-6
unipolar input polarity, selecting, 3-5
unipolar mode, straight binary coding (table), 3-15
unipolar output calibration, 4-8
gain calibration, 4-8
offset calibration, 4-8
update mode of analog output circuitry, programming, 3-15 to 3-16
user-defined calibration constants, 4-1 to 4-2

W

WDAQCONF program, 3-1
WRTPRT bit
description, 2-12
reading single byte from EEPROM, 3-3
storing user-defined constants, 4-2

X

Xicor X25020 data sheet, C-1