INSTRUCTION MANUAL

Sensoray Model 425
ISAbus Analog/Digital I/O Card

01/10/2000

Sensoray Co., Inc.
7313 SW Tech Center Dr., Tigard, Oregon 97223
voice: 503.684.8005, fax: 503.684.8164, e-mail: sales@sensoray.com
www.sensoray.com

M A N U A L - M O D E L 4 2 5

Table of Contents

Limited Warranty

Sensoray Company, Incorporated (Sensoray) warrants the Model 425 hardware to be free from defects in material and workmanship and perform to applicable published Sensoray specifications for one year from the date of shipment to purchaser. Sensoray will, at its option, repair or replace equipment that proves to be defective during the warranty period. This warranty includes parts and labor.

The warranty provided herein does not cover equipment subjected to abuse, misuse, accident, alteration, neglect, or unauthorized repair or installation. Sensoray shall have the right of final determination as to the existence and cause of defect.

As for items repaired or replaced under warranty, the warranty shall continue in effect for the remainder of the original warranty period, or for ninety days following date of shipment by Sensoray of the repaired or replaced part, whichever period is longer.

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. Sensoray will pay the shipping costs of returning to the owner parts which are covered by warranty.

Sensoray 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, Sensoray reserves the right to make changes to subsequent editions of this document without prior notice to holders of this edition. The reader should consult Sensoray if errors are suspected. In no event shall Sensoray be liable for any damages arising out of or related to this document or the information contained in it.

EXCEPT AS SPECIFIED HEREIN, SENSORAY 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 SENSORAY SHALL BE LIMITED TO THE AMOUNT THERETOFORE PAID BY THE CUSTOMER. SENSORAY 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.

Special Handling Instructions

The Model 425 circuit board contains CMOS circuitry that is sensitive to Electrostatic Discharge (ESD).

Special care should be taken in handling, transporting, and installing the 425 to prevent ESD damage to the board. In particular:

- Do not remove the circuit board from its protective antistatic bag until you are ready to configure the board for installation.
- Handle the circuit board only at grounded, ESD protected stations.
- Remove power from the ISAbus before installing or removing the circuit board.

Introduction

The Sensoray model 425 is an ISAbus I/O card that performs several functions commonly required in embedded applications. On-board peripherals include:

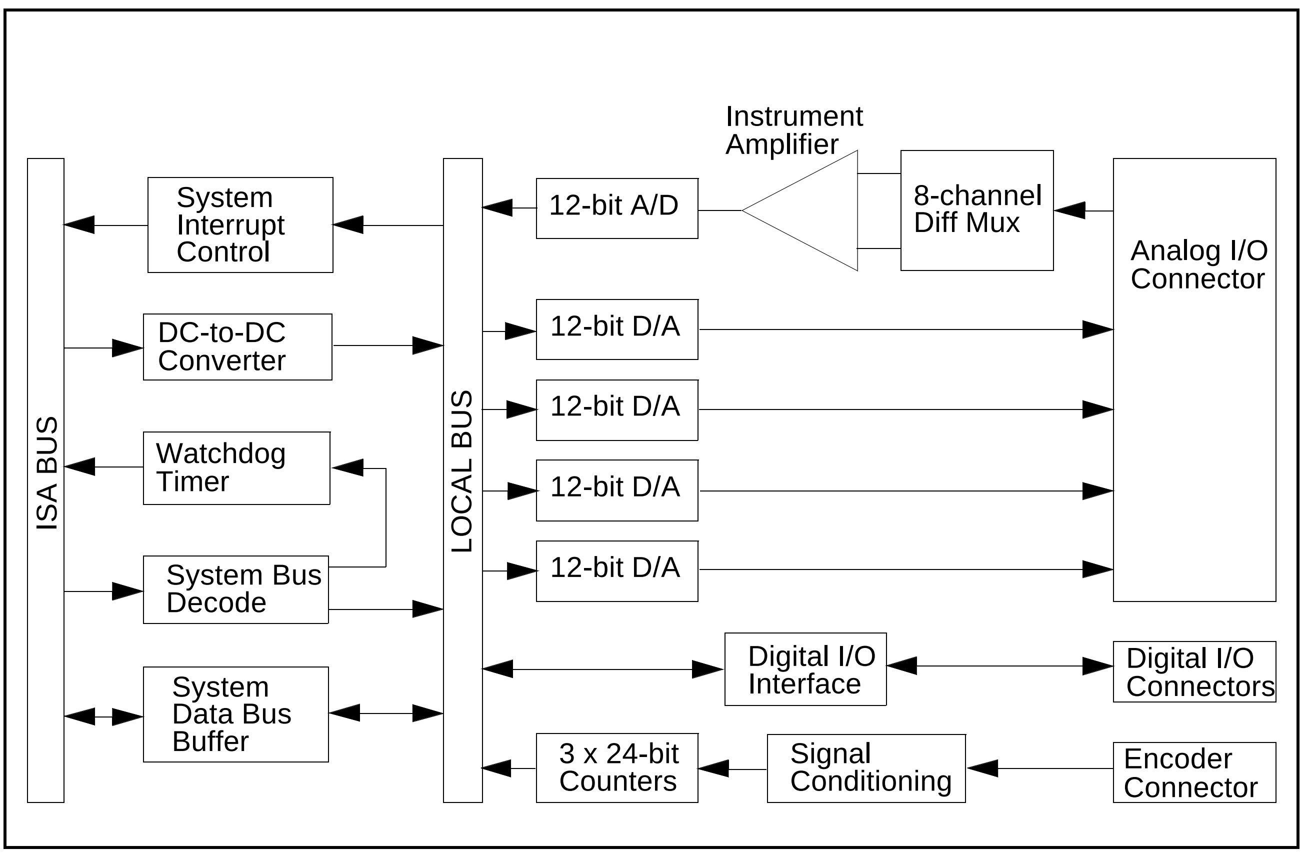
- Programmable watchdog timer.
- 48 digital I/O channels. 16 channels have edge detection and interrupt capability.
- 12-bit analog-to-digital converter with eight differential input channels and resistor-programmed gain.
- Four 12-bit analog output channels.
- Three 24-bit counters optimized for interface to incremental encoders.

The 425 is powered from the ISAbus 5-volt supply. No auxiliary 12- or 15-volt supplies are required. Power for linear circuitry is derived from an on-board DC-to-DC converter. Low power CMOS circuitry is used extensively to minimize power consumption and enhance reliability.

The board occupies a 64-byte block of I/O addresses in the ISAbus 16-bit I/O space. Programming shunts may be installed to map the board to any 64-byte address boundary.

Four headers are provided for connecting on-board peripherals to external circuitry. Low profile vertical headers are used to satisfy the tight clearances mandated by modern high-density system designs.

Block Diagram



Selecting a Base Address

The 425 board occupies a block of 64 consecutive addresses in the ISAbus 16-bit I/O space. This block must begin on a 64-byte address boundary. The first address in the block — the “base address” — is assigned by option shunts A6-A15.

Care must be exercised to ensure that no other devices use addresses in the block assigned to the 425. Use the tables shown below to determine the proper settings for your application.

First Address Nibble (MSB)

Adrs	A15	A14	A13	A12
0	**On**	**On**	**On**	**On**
1	On	On	On	Off
2	On	On	Off	On
3	On	On	Off	Off
4	On	Off	On	On
5	On	Off	On	Off
6	On	Off	Off	On
7	On	Off	Off	Off
8	Off	On	On	On
9	Off	On	On	Off
A	Off	On	Off	On
B	Off	On	Off	Off
C	Off	Off	On	On
D	Off	Off	On	Off
E	Off	Off	Off	On
F	Off	Off	Off	Off

Second Address Nibble

Adrs	A11	A10	A9	A8
0	On	On	On	On
1	On	On	On	Off
2	On	On	Off	On
3	On	On	Off	Off
4	On	Off	On	On
5	On	Off	On	Off
6	On	Off	Off	On
7	On	Off	Off	Off
8	Off	On	On	On
9	Off	On	On	Off
A	Off	On	Off	On
B	Off	On	Off	Off
C	**Off**	**Off**	**On**	**On**
D	Off	Off	On	Off
E	Off	Off	Off	On
F	Off	Off	Off	Off

Third Address Nibble

Adrs	A7	A6
0	On	On
4	On	Off
8	**Off**	**On**
C	Off	Off

For example, to program the board base address to 0300 hex, install shunts A15-A10 and A7-A6, and remove shunts A9-A8.

Note: default settings (address 0C80) in **bold**

Assigning an Interrupt Level

Several onboard subsystems are capable of generating interrupt service requests to the host processor. All three encoder counters as well as relay channels 0-15 may source such requests.

To compensate for the limited interrupt resources in ISAbus systems, the 425 board is restricted to a single physical interrupt request signal for all onboard interrupt sources.

For maximum flexibility, the board’s interrupt request may be assigned to any available IRQ line on the ISAbus. When choosing an interrupt level for the 425 board, verify that no other boards are using the same level.

If you will be using the 425 board to generate interrupts, you must install an option shunt to assign an interrupt level to the board. Install a shunt at the position corresponding to the desired interrupt service level per the table to the right. Make sure that only one interrupt level is selected.

IRQ	Position
3	IRQ3
4	IRQ4
5	IRQ5
6	IRQ6
7	IRQ7
9	IRQ9
10	IRQ10
11	IRQ11
12	IRQ12
14	IRQ14
15	IRQ15

If the 425 board will not be sourcing interrupt requests, be sure that no option shunts are installed in any of the IRQ positions.

Fault Indicator

A red light-emitting diode located near the top edge of the circuit board indicates reset and fault conditions. In normal operation, the indicator emits only during system or local (board) reset. The indicator should be off at all other times.

When the indicator is emitting, the board is said to be in a “fault” condition, regardless of whether a true fault condition exists or the board is simply being reset.

Functional Limitations During Fault

While the fault indicator is active, only the Status and Reset ports are available to the ISAbus master. Consequently, the host should limit itself to accessing these two ports during a fault condition.

The state of the fault indicator may be read by the host processor through the status port.

Persistent Fault

If the fault indicator remains on for more than one-half second following a reset, a genuine fault condition may exist on the board that requires servicing.

Before servicing the unit, make sure that your ISAbus master is not invoking repeated soft resets. Also, verify that there are no addressing conflicts between 425 and system board or other expansion boards in your system.

Reset

The 425 may be reset by either “hard reset” from the ISAbus system reset signal or “soft reset” under software control of the ISAbus master. A reset may be performed at any time.

Soft Reset

A “soft reset” causes a local reset on only the 425 board and will not reset any other devices on the ISAbus. A soft reset is invoked by writing to the Reset port at the board base address + 62.

Reset port (base + 62, write only):

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The value written to the Reset port should be all zeros to ensure compatibility with future product enhancements.

Board State During Reset

The board’s fault indicator will activate upon receipt of a soft or hard reset command. The indicator will remain active until the reset command is released and the board completes an initialization sequence. During this initialization sequence, the board exhibits the following behavior:

- Only the Status and Reset ports are accessible.
- The Status port FLT flag is valid; all other Status port flags are meaningless.

Default State Following a Reset

When the initialization sequence is completed, the fault indicator will turn off, all board I/O ports become accessible and the board defaults to the following condition:

- Watchdog Timer
  Disabled
  Fast mode (630 milliseconds)
- D/A Converters
  Outputs are disabled
  Setpoints are indeterminate
- Encoder Counters
  Reset to zero counts
  Forced to Mode 0
  Event captures disabled
  Index capture type selected
- Relay Channels
  Switched to off (open-collector, pulled up) state
  Event captures disabled
  Negative edge capture is selected
- A/D Converter
  Forced to unipolar mode
  Input channel 0 selected
- Interrupts
  Board master interrupt enabled
  Encoder counter interrupts disabled
  Relay interrupts disabled

Status Register

A status port, located at board base address + 22, makes available status bits to the ISAbus host. Unless otherwise specified, all status bits are active high.

Status port (base + 22, read only):

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FLT	CP2	CP1	CP0	RQR	RQ2	RQ1	RQ0	x	IE#	WDI	WDR	BSY	AE	WDE	BIP

FLT — reflects state of the Fault LED. *
CP2 — counter 2 event captured
CP1 — counter 1 event captured
CP0 — counter 0 event captured
RQR — relay interrupt service request.
RQ2 — counter 2 interrupt service request.
RQ1 — counter 1 interrupt service request.
RQ0 — counter 0 interrupt service request.
IE# — board master interrupt enabled (active low).
WDI — watchdog slow timeout interval.
WDR — watchdog timer timed out.
BSY — ADC is busy digitizing.
AE — DAC outputs are enabled.
WDE — watchdog timer is enabled.
BIP — ADC is programmed for unipolar operation.

* After a reset, verify that FLT is false before attempting to access other registers on the board. Only the Status and Reset ports are functional while FLT is active.

Interrupts

Individual board interrupt sources that are associated with specific devices, such as encoder counters and relay channels, are provided with dedicated interrupt masking hardware. These interrupt masking features are described in detail in other sections of this manual.

In addition to the individual device interrupt masks, a board interrupt mask is provided. This mask serves as a master interrupt enable for the entire 425 board.

When enabled, the board interrupt mask permits individual device interrupt requests to pass through to the system IRQ, thereby interrupting the host. When disabled, the board interrupt mask blocks all individual device interrupt requests and negates the system IRQ, regardless of the states of any of the individual device interrupt requests.

The board interrupt is automatically enabled in response to a board reset.

Enabling/Disabling the Board Interrupt

The board interrupt is enabled and disabled by writing to the Control port with bit 14 set. Note: the “V” bit is active low.

Control Port (base + 40, write only): Enab/Disab Board Int

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
V	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0

V — enables (0) or disables (1) the board interrupt.

Watchdog Timer

Embedded systems often include a watchdog timer to gain control following an unplanned loss of control by the ISAbus master. In such systems, the CPU periodically refreshes the watchdog. Should the CPU lose control, the watchdog won’t be refreshed and will eventually timeout. The resulting timeout will interrupt or restart the CPU.

Watchdog Timing

The watchdog timer has a guaranteed minimum timeout interval (MTI) of 630 milliseconds in the Fast mode or 10 seconds in the Slow mode. The interval between timer refreshes must not exceed the MTI.

CPUs may not be able to refresh the watchdog within the MTI during system boot or critical I/O operations. To accommodate these situations, the 425 provides a control register for enabling and disabling the watchdog under program control. The watchdog is automatically disabled by a soft reset or system-wide hard reset.

The first watchdog refresh must occur no later than the MTI after enabling the timer.

Selecting the Timeout Interval

The watchdog timeout interval (WTI) defaults to the Fast mode. The interval may be changed by writing to the Control port with bit 13 set.

Control Port (base + 40, write only): Select Wdog Interval

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
V	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0

V — Slow interval (1) or Fast interval (0).

Enabling/Disabling the Watchdog

The watchdog is enabled and disabled by writing to the Control port with bit 9 set.

Control Port (base + 40, write only): Enab/Disab Watchdog

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
V	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0

V — enables (1) or disables (0) watchdog timer.

Refreshing the Watchdog

The watchdog is refreshed by writing to the Control port with bits 8 and 15 set.

Control Port (base + 40, write only): Refresh Watchdog

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

Routing the Watchdog to the ISAbus

A watchdog timeout event may be configured to cause either of two possible ISAbus actions: a system-wide hard reset, or a non-maskable interrupt (NMI).

To generate a hard reset, connect P4 to your ISAbus active-low external reset input. This requires a user-supplied two-conductor cable.

To generate a NMI, install an option shunt at the posts labelled “NMI.” For proper operation, you must install appropriate host interrupt handler software. In addition, you may need to configure various control ports on your system board to enable pass-through of the interrupt to the host (by means of the ISAbus CHCHK signal). Refer to your system board documentation for further information.

If you will not be using the watchdog, you may leave connector P4 disconnected and the NMI shunt uninstalled.

Encoder Interface

Overview

Three counter channels reside on the 425. Each channel is optimized for incremental encoders by providing the following elements and features:

- Input buffers interface directly to TTL, CMOS, or differential RS422 signals.
- Decoder logic detects and converts encoder edges into clock and direction signals.
- 24-bit up/down counter tracks encoder position without resorting to multiple counter channels.
- Mode register selects internal or external clock source, clock multiplier and count direction.
- Index input auto-resets counter when armed.
- Programmable interrupt on index or rollover.
- 5 Volt encoder power available at encoder connector.

Additional elements are shared by all counter channels:

- Holding register for reading counter contents.
- Preload register for initializing counter contents.

Index Inputs

Each counter channel has a dedicated "Index" input signal and programmable arming register. A counter is reset to zero when its index is asserted and armed.

Use of the Index signal is optional. If you are not using the Index input of an encoder channel, simply leave the corresponding input signals disconnected.

Phase Inputs

Each counter channel has two clock input phases named "A" and "B." Depending on the application, one or both of these signals may be connected to an encoder.

If both phases are used, they are assumed to be quadrature encoded, meaning that they are 90 degrees out of phase. In this case, counters will count both up and down based on the timing relationship of the two phase inputs.

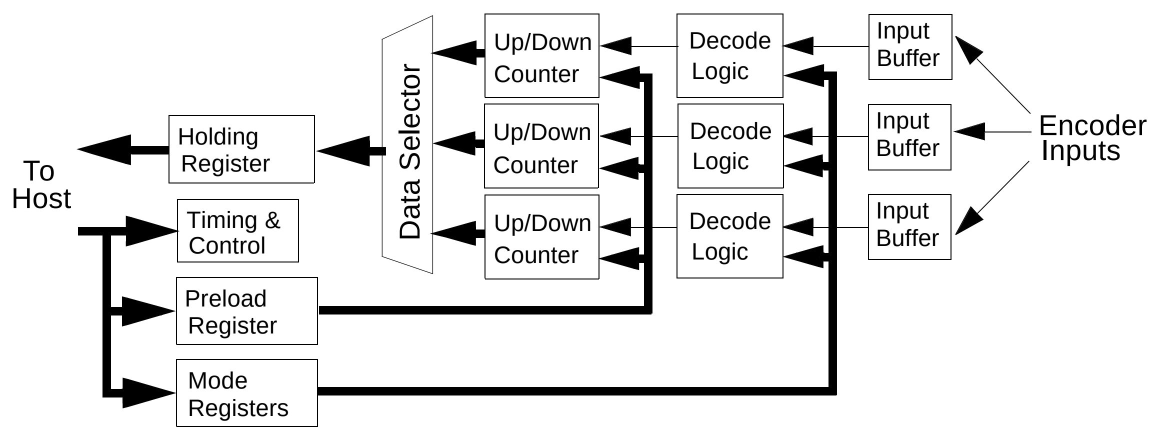
If only one phase is used, the input is said to be single-phase. In this case, counter channels will count either up or down, but not both. This configuration is typically used to count pulses from devices that produce a single clock output.

Quadrature encoders have advantages over single-phase encoders. Counters will not accumulate errors when an encoder changes direction or dithers about a state transition boundary. Also, it is possible to increase encoder resolution by clocking the counters at a multiple of the single-phase clock rate.

Reset

Hardware or software reset of the 425 will zero all counters and the Mode, Holding and Preload registers. Counting is inhibited during the reset pulse and subsequent fault condition.

Encoder Interface Block Diagram



Mode Selection

Each counter channel may be configured for any of ten valid operating modes (eight counter modes and two timer modes) through the Counter control port:

Counter Control Port (base + 16, write only) : Set Mode

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	1	1	A1	A0	M3	M2	M1	M0

A_1 and A_0 specify which counter is to have its mode programmed:

A_1A_0	Encoder
00	Channel 0
01	Channel 1
10	Channel 2
11	Not Valid

Mode			Counter Function		
		M<3:0>	Clk	Phases	Direction
Counter	**0**	0000	x2	2	Normal
Counter	**1**	0001	x4	2	Normal
Counter	**2**	0010	x1	2	Normal
Counter	**3**	0011	x1	1	Normal
Counter	**4**	0100	x2	2	Reverse
Counter	**5**	0101	x4	2	Reverse
Counter	**6**	0110	x1	2	Reverse
Counter	**7**	0111	x1	1	Reverse
Timer	**8-10**	1000-1010	Invalid modes		
Timer	**11**	1011	3MHz		Down
Timer	**12-14**	1100-1110	Invalid modes		
Timer	**15**	1111	3MHz		Up

Four functional attributes are specified by the mode: clock source (internal or external), clock multiplier (x1, x2 or x4), counter input type (quadrature or single-phase), and count direction.

The *clock source*, controlled by bit M3, determines whether the channel will act as a counter or a timer. When configured as a timer, a channel will count either up or down at a three megahertz rate. When configured as a counter, a channel counts pulses from the external clock source.

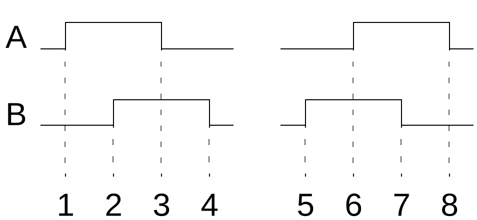
The *clock multiplier* -- relevent only in modes 0 through 7 -- determines the number of count events per cycle on the “A” phase input. A count event is a phase state transition that causes a counter increment or decrement. For example, a “x2” multiplier means that two count events will occur for each full cycle of the “A” phase input. See the counter timing diagram for details.

The *counter input type* -- relevent only in modes 0 through 7 -- may be configured for either one or two active phases. A two-phase mode must be specified when using Quadrature inputs. A one-phase mode must be specified for single-phase sources.

The *Count direction*, controlled by bit M2, is specified as either normal or reverse. Refer to the timing diagram for details.

Counter Timing Diagram

Encoder counters change state (count up or down) upon detection of a valid input transition. All possible transitions are shown in the timing diagram below. For example, transition number 1 specifies a rising edge on the “A” phase while “B” is held low.



Valid transitions depend on the operating mode. The table to the right lists valid transitions for each mode. In mode 0, for example, a counter will count up only at transitions 6 and 8, and will count down only at transitions 1 and 3.

Valid count transitions vs. counter operating mode

Counter Configuration		Valid Count Transitions	
Mode	Function	Up	Down
0	Quadrature x2	6, 8	1, 3
1	Quadrature x4	5, 6, 7, 8	1, 2, 3, 4
2	Quadrature x1	8	1
3	Single Phase x1	---	1
4	Quadrature x2	1, 3	6, 8
5	Quadrature x4	1, 2, 3, 4	5, 6, 7, 8
6	Quadrature x1	1	8
7	Single Phase x1	1	---

Counter Data Format

Although the counters are physically organized as 24-bits wide, counter data are represented in a 32-bit two's-complement format.

When reading a counter, the most-significant byte of the returned 32-bit value is generated by sign-extending the most-significant bit of the physical 24-bit counter.

When writing to a counter, the most-significant byte of the specified 32-bit value is ignored and the least-significant 24-bits are loaded into the physical counter. Note that the most-significant bit of the 24-bit value is treated as a sign bit.

Reading Accumulated Counts

Because of the asynchronous relationship between counters and the ISAbus, counters may not be directly read by the host. Instead, counters must first be synchronously copied to the Holding register, which in turn may be asynchronously read by the host.

To copy a counter to the Holding register, issue a Latch command by writing the following value to the Counter control port:

Counter Control Port (base + 16, write only) : Latch Counts

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	1	0	A1	A0	0	0	0	0

A_1 and A_0 specify which counter is to be transferred to the holding register:

A_1A_0	Encoder
00	Channel 0
01	Channel 1
10	Channel 2
11	Not Valid

After executing the Latch command, counter data may be read from the two-word Holding register:

Holding Register Low Word (base + 18, read only)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0

Holding Register High Word (base + 20, read only)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
D23	D23	D23	D23	D23	D23	D23	D23	D23	D22	D21	D20	D19	D18	D17	D16

The low and high words may be read from the Holding register in any order. Data are held in the Holding register until the next Latch command.

Note: it is not required for the host to read either or both words from the Holding Register after issuing a Latch command. Consequently, in applications that require only 16-bit data, it is permissible to read only the low word from the Holding Register.

Writing to Counters

Counters may be preloaded with any 24-bit value. In order to preload a counter, the desired data value must first be written to the Preload Register, then synchronously transferred to the target counter.

To preload a counter, first write the desired data value to the Preload Register:

Preload Register Low Word (base +18, write only)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0

Preload Register High Word (base + 20, write only)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
x	x	x	x	x	x	x	x	D23	D22	D21	D20	D19	D18	D17	D16

Next, synchronously transfer the Preload register to the target counter by writing a Load Counts command to the Counter control port:

Counter Control Port (base + 16, write only) : Load Counts

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	1	A1	A0	0	0	0	0

A_1 and A_0 specify which counter will receive a copy of the Preload Register:

A_1A_0	Encoder
00	Channel 0
01	Channel 1
10	Channel 2
11	Not Valid

After executing the Load Counts command, the designated counter will begin counting from the value contained in the Preload Register.

Note: the Preload Register changes only when written to by the host. If your application uses a constant preload value, the Preload Register need be loaded only one time.

Rollovers

Encoder counters physically increment from FFFFFF to 000000 when counting up, and decrement from 000000 to FFFFFF when counting down. These state transitions are called "rollovers."

Event Detection

Each counter may be programmed to capture the occurance of, and take special action in response to, a channel event.

The act of enabling captures is called “arming.” Before arming a channel, the event types to be captured must be selected. Two types of events may be captured: *index* and *rollover*. Index events occur when the channel’s index signal transitions to the active state, while rollover events occur as described earlier under *Rollovers*.

Each counter is allocated an “Arm” control register for enabling and disabling event capture, and two status flags, “Capture” and “IndexCapture,” to log captured events. These status flags are automatically reset when the channel is disarmed.

When armed, a selected event will set the Capture flag. In addition, captured index events will set the IndexCapture flag, and may be programmed to automatically reset the counter contents to zero.

New event captures are inhibited when the Capture flag is set. To capture new events after a previous capture has been processed, disarm the channel to clear the capture flags and then re-arm the channel to enable new captures.

Selecting Capture Type for Counters

When operating as a counter, a channel may be configured to capture an index, a rollover, or both. When operating as a timer, a channel may be configured to capture a rollover or both rollover and index, but not index only.

The event types to be captured depend on the channel operating mode and the CM1 and CM0 Capture Mode bits in the counter control port:

Operating Mode	CM<1:0>	Captured Events		Reset on Index
		Index	Rollover	
Counter	00	●		●
	01		●	
	10	●	●	
Timer	00	●	●	●
	01		●	
	10	●	●	

The CM1 and CM0 bits, which default to 00 upon soft or hard board reset, are individually programmed by separate write operations to the counter control port.

Counter Control Port (base + 16, write only) : CM1 Bit

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
V	0	0	1	0	0	0	0	0	0	A1	A0	0	0	0	0

Counter Control Port (base + 16, write only) : CM0 Bit

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
V	0	0	0	1	0	0	0	0	0	A1	A0	0	0	0	0

V=1 to set CMx bit to 1,
V=0 to reset CMx bit to 0.
A_1 and A_0 specify which counter channel is to be addressed:

A_1A_0	Encoder
00	Channel 0
01	Channel 1
10	Channel 2
11	Not Valid

Arming/Disarming Event Capture

The Counter control port is used to arm and disarm event capture:

Counter Control Port (base + 16, write only) : Arm/Disarm

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
V	0	0	0	0	1	0	0	0	0	A1	A0	0	0	0	0

V=1 to arm event capture,
V=0 to disarm event capture.
A_1 and A_0 specify which counter channel is to be armed/disarmed:

A_1A_0	Encoder
00	Channel 0
01	Channel 1
10	Channel 2
11	Not Valid

Monitoring Captured Events

All three counter Capture flags may be simultaneously read from the status port:

Status Port (base + 22, read only) : Capture Flags

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
x	CP2	CP1	CP0	x	x	x	x	x	x	x	x	x	x	x	x

Bit	Function
CP2	Counter Channel 2 Capture Flag
CP1	Counter Channel 1 Capture Flag
CP0	Counter Channel 0 Capture Flag

When set, each of these three status bits indicate that the corresponding channel captured an event. The captured event type is either index or rollover.

If a counter channel has been programmed to capture both index and rollover events, it may be necessary to determine whether a captured event was an index or a rollover. All three counter IndexCapture flags may be simultaneously read from the extended status port:

StatusX Port (base + 54, read only) : IndexCapture Flags

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
x	CX2	CX1	CX0	x	x	x	x	x	x	x	x	x	x	x	x

Bit	Function
CX2	Counter Channel 2 IndexCapture Flag
CX1	Counter Channel 1 IndexCapture Flag
CX0	Counter Channel 0 IndexCapture Flag

When set, each of these three status bits indicate that the corresponding channel captured an index event.

Counter Interrupts

Counter channels may be programmed to generate interrupt service requests in response to captured events. A channel asserts it's interrupt request when two conditions are simultaneously met: the Capture flag is set, and the channel interrupt is enabled. The Control port is used to enable and disable counter interrupts:

Control Port (base + 16, write only) : Set Interrupt Enables

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
V	0	0	0	0	0	1	0	0	0	A1	A0	0	0	0	0

V=1 to enable interrupt,
V=0 to disable interrupt.
A_1 and A_0 specify which counter channel is to be enabled/disabled:

A_1A_0	Encoder
00	Channel 0
01	Channel 1
10	Channel 2
11	Not Valid

When servicing an interrupt request from the 425 board, examine the Status port to determine which device (or devices) requires service:

Status Port (base + 22, read only) : Determine IRQ Source

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
x	x	x	x	RQR	RQ2	RQ1	RQ0	x	x	x	x	x	x	x	x

Bit	Service Requestor
RQR	Relay Channel(s)
RQ2	Counter Channel 2
RQ1	Counter Channel 1
RQ0	Counter Channel 0

Timer Applications

A timer is often used to generate an interrupt request upon expiration of a time interval. Such applications will typically follow this sequence of events:

- Preload an initial value into the counter. The initial value equals the number of 3MHz cycles that will occur during the desired time interval. For example, the value 3,000,000 will produce a 1 second interval.
- Enable the counter interrupt and arm the event capture.
- Timer rollover occurs, setting the Capture flag and generating an interrupt request.
- Host responds to the interrupt request by executing an interrupt handler. By reading the Status register, the host determines that the counter is requesting service. The host disables the interrupt and disarms event capture, then performs the appropriate application functions.

Encoder Applications

Encoders are frequently used in motion control systems to track absolute position. In a typical system, each control axis employs a limit switch in conjunction with encoder index to recognize the axis reference position. The motor seeks the reference position by “backing up” until the limit switch activates.

A typical encoder application, with interrupts, follows this sequence of events to seek a reference position:

- “Reverse” motor until the limit switch activates, then stop the motor. The limit switch could be easily connected to a relay channel for interrupt-driven edge detection.
- Arm the counter channel event capture and enable the counter channel interrupt.
- Start the motor in the “forward” direction.
- Encoder index activates, simultaneously zeroing the counter, setting the Capture flag, and generating an interrupt request.
- The host responds to the interrupt request by executing an interrupt handler. By reading the Status register, the host determines that the counter is requesting service. The host disables the interrupt and disarms event capture, then performs the appropriate application functions.

Encoder Connections

Connector J1 is used to make all electrical connections to external encoders or other pulse sources:

Pin	Name	Function
1	$-A_0$	Channel 0 phase A input (-)
2	$+A_0$	Channel 0 phase A input (+)
3	GND	GND
4	$-B_0$	Channel 0 phase B input (-)
5	$+B_0$	Channel 0 phase B input (+)
6	+5V	+5V
7	$-X_0$	Channel 0 index input (-)
8	$+X_0$	Channel 0 index input (+)
9	GND	GND
10	$-A_1$	Channel 1 phase A input (-)
11	$+A_1$	Channel 1 phase A input (+)
12	+5V	+5V
13	$-B_1$	Channel 1 phase B input (-)
14	$+B_1$	Channel 1 phase B input (+)
15	GND	GND
16	$-X_1$	Channel 1 index input (-)
17	$+X_1$	Channel 1 index input (+)
18	+5v	+5v
19	$-A_2$	Channel 2 phase A input (-)
20	$+A_2$	Channel 2 phase A input (+)
21	GND	GND
22	$-B_2$	Channel 2 phase B input (-)
23	$+B_2$	Channel 2 phase B input (+)
24	+5V	+5V
25	$-X_2$	Channel 2 index input (-)
26	$+X_2$	Channel 2 index input (+)

Signal Pairs

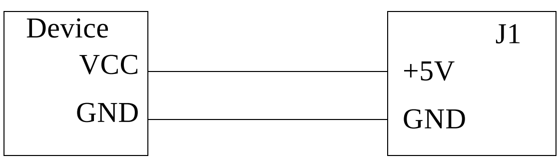
Each counter channel is assigned three input signal pairs. One pair connects to the A phase, one to the B phase, and one to the index. All input pairs are implemented as differential RS422 receivers.

Termination resistors are not supplied with the 425 board. If your encoder (or other signal source) employs RS422 drivers, you may need to supply external terminations near connector J1. Refer to your encoder documentation for the recommended practice.

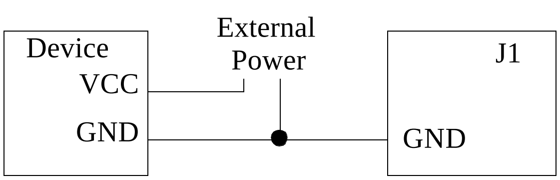
Encoder Power

Any connector J1 pins designated as +5V and GND may be used to power encoders. Regardless of whether or not an external power source is used to supply power to an encoder, you *must* connect one of the J1 GND pins to the encoder's power supply return. <u>Failure to connect these returns may damage the 425 board.</u>

If you are using an encoder or other pulse-generating device that will be powered by the 425 board, connect power signals as follows:

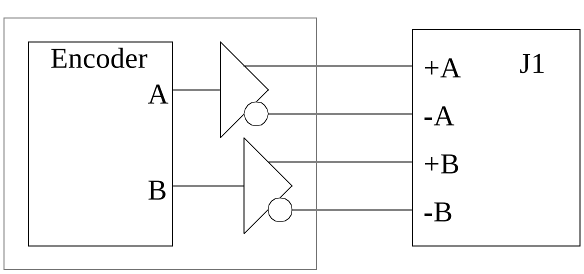


If you are supplying power from an external source, make sure the device ground is connected to one of the GND pins on J1:

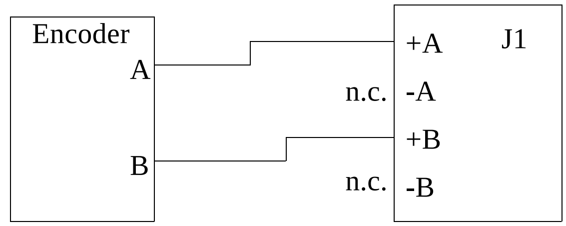


Phase Connections

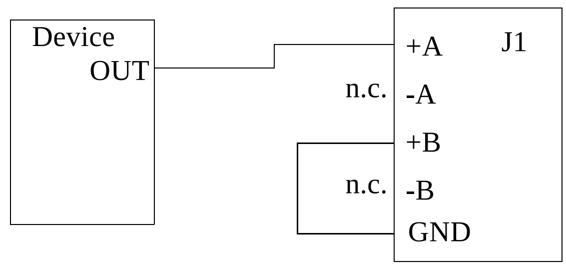
Some encoders have built-in RS422 drivers. If you are using this type of encoder, connect it as follows:



Other encoders have TTL or CMOS outputs. If you are using this type of encoder, connect it as follows:

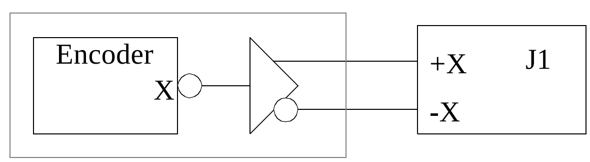


Counter channels may also be interfaced to single-phase devices. In such cases, the counter mode is set to 3 or 7, and the device is connected as follows:

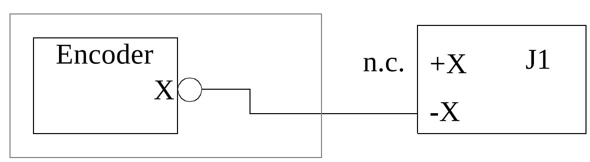


Index Connections

Index signals, which are treated as active-low sources, are connected in a fashion similar to that of the phase inputs. In the case of a differential RS422 index pair, connect the two signals to the +X and -X pins:



For a single-ended TTL or CMOS index, connect to the -X pin and leave the +X pin disconnected:



Leave both +X and -X disconnected if you are not using the index input.

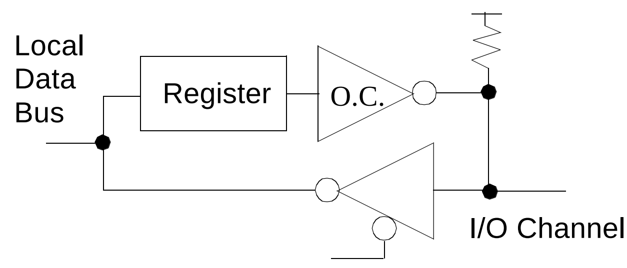
Relay Interface

The 425 board provides 48 high-current digital I/O channels designed for direct interface to industry standard relay racks.

As shown in the diagram to the right, each relay channel is physically structured as a write-only register followed by an open-collector buffer (with 10Kohm pullup), which in turn is looped back to the local data bus by a 3-state buffer.

Channel data is active-high on the local bus side, and active-low on the I/O channel side.

One of 48 Digital I/O Channels



Relay Access Ports

The 48 relay channels are accessed through three bidirectional I/O ports. Each port accesses 16 channels simultaneously:

Relay 0-15 port (base + 24, read/write)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R15	R14	R13	R12	R11	R10	R9	R8	R7	R6	R5	R4	R3	R2	R1	R0

Relay 31-16 port (base + 26, read/write)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R31	R30	R29	R28	R27	R26	R25	R24	R23	R22	R21	R20	R19	R18	R17	R16

Relay 47-32 port (base + 28, read/write)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R47	R46	R45	R44	R43	R42	R41	R40	R39	R38	R37	R36	R35	R34	R33	R32

Reading from a Relay port returns the physical state of the channels associated with that port. Writing to a Relay port programs the associated relay channel output registers.

Note: the value read from a relay channel may not be the same as the value stored in that channels output register. Specifically, the values will differ if an output register is programmed to 0 and an external sink pulls the corresponding channel down to 0V.

Configuring I/O Type

Each channel may be independently configured as either an input or output. To configure a channel as an input, zero the associated channel register.

Connections

All electrical connections are made through 50-pin connectors P3 and P1. On each of these connectors, one pin is assigned to each active-low I/O channel.

All even pins are connected to ISAbus five volt return. In addition, pin 49 may be used to supply limited five volt power (<100mA) to an external relay rack.

Pin	P1	P3
1	CH23	CH47
3	CH22	CH46
5	CH21	CH45
7	CH20	CH44
9	CH19	CH43
11	CH18	CH42
13	CH17	CH41
15	CH16	CH40
17	CH15	CH39
19	CH14	CH38
21	CH13	CH37
23	CH12	CH36
25	CH11	CH35
27	CH10	CH34
29	CH9	CH33
31	CH8	CH32
33	CH7	CH31
35	CH6	CH30
37	CH5	CH29
39	CH4	CH28
41	CH3	CH27
43	CH2	CH26
45	CH1	CH25
47	CH0	CH24
49	+5V	+5V
Even	GND	GND

Enhanced Relay Channels

Relay channels 0-15 have enhanced functionality beyond the basic capabilities of channels 16-47. In addition to basic relay I/O, these enhanced channels have mechanisms for detecting and capturing (recording) state transitions. Captured transitions may be read by the host and may be programmed to request ISAbus interrupt service.

Selecting the Edge Type

Enhanced channels may be configured to detect either rising or falling edge transitions. The Edge Select register simultaneously programs the desired edge types for all enhanced relay channels:

Relay 0-15 Edge Select register (base + 34, write only)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R15	R14	R13	R12	R11	R10	R9	R8	R7	R6	R5	R4	R3	R2	R1	R0

Set a bit to detect rising edges on the associated relay channel, or zero the bit to detect falling edges. Note that “rising” or “falling” edges are viewed from the perspective of data polarity on the local data bus.

Arming Capture

A relay channel must be “armed” before it will capture a detected edge. Any arbitrary block of enhanced relay channels may be simultaneously armed by writing to the Capture Arm port:

Relay 0-15 Capture Arm port (base + 38, write only)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R15	R14	R13	R12	R11	R10	R9	R8	R7	R6	R5	R4	R3	R2	R1	R0

Set a bit to enable edge detection for the associated relay channel, or zero the bit to leave the capture enable unchanged.

Edge Capture Register

Each enhanced channel is associated with a one-bit Capture register. The Capture register is responsible for logging the occurance of a detected edge. When a selected edge occurs while capture is enabled, the associated Capture register is set. All Capture registers may be read simultaneously through the Capture port:

Relay 0-15 Capture port (base + 30, read only)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R15	R14	R13	R12	R11	R10	R9	R8	R7	R6	R5	R4	R3	R2	R1	R0

Note that before the Capture register can log a detected edge, channel capture must be armed. After capturing an edge, the capture register remains set until explicitly reset by the host.

Clearing Captured Edges

Sometime after a captured edge is detected, it is necessary to reset the associated capture register. This is accomplished by writing to the Capture Disarm port:

Relay 0-15 Capture Disarm port (base + 36, write only)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R15	R14	R13	R12	R11	R10	R9	R8	R7	R6	R5	R4	R3	R2	R1	R0

Set a bit to disable edge detection and reset the capture register for the associated relay channel, or zero the bit to leave the associated arm and capture registers unchanged. Note that a group of relay channels may be simultaneously disarmed by setting multiple bits when writing to this port.

Interrupts

Enhanced relay channels may be individually programmed to request interrupt service upon capture of a detected edge. Enable or disable relay channel interrupts by writing to the Interrupt Enable port:

Relay 0-15 Interrupt Enable port (base + 30, write only)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R15	R14	R13	R12	R11	R10	R9	R8	R7	R6	R5	R4	R3	R2	R1	R0

Set a bit to enable the associated relay interrupt, or zero the bit to disable the interrupt. Note that clearing the interrupt enable has no effect on the capture register.

Programming Sequence

In order for a relay to generate an interrupt request, the channel must be armed, an edge must be captured, and the interrupt must be enabled.

A typical application, with interrupts, goes like this:

- Edge Type is selected.
- Channel is armed to enable capture.
- Interrupt is enabled.
- An edge is detected by hardware, setting the capture register and generating an interrupt request.
- ISR disarms channel, thereby clearing capture register and negating interrupt request.
- Channel is armed to enable next capture.

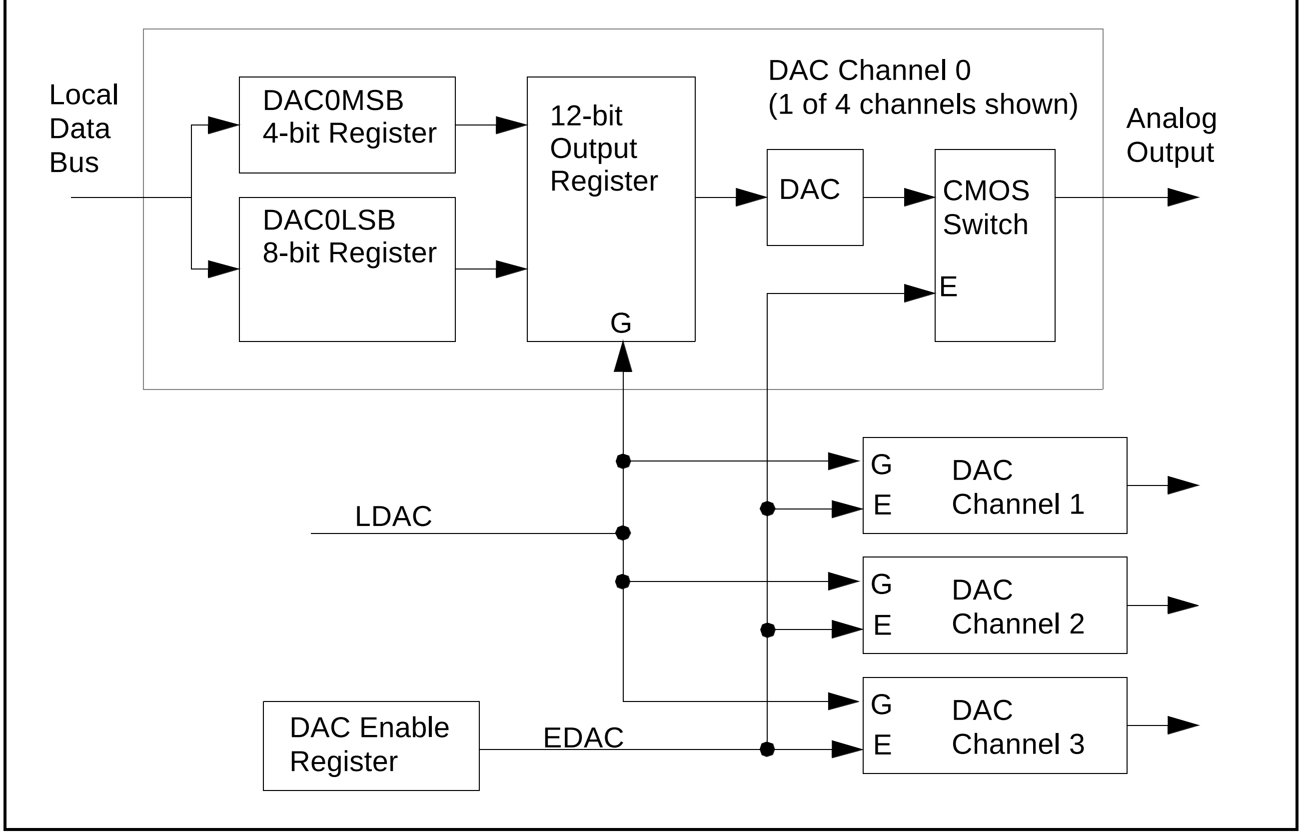
D/A Interface

Each DAC channel consists of a low-byte/high-nibble bus register pair, 12-bit output register, 12-bit D/A converter, and CMOS switch.

All four CMOS switches are enabled by the one-bit DAC Enable Register. When enabled, the CMOS switches connect all four DAC outputs to 40-pin header P2. When disabled, all DAC output signals at P2 are pulled down to zero volts.

A control strobe — LDAC — simultaneously transfers data from the four bus register pairs to their corresponding 12-bit output registers. DAC output ranges are fixed at 0 to +10 volts.

D/A Subsystem Block Diagram



DAC Enable Register

Following a reset, all four 12-bit output registers contain indeterminate values. To ensure orderly startup, the DAC Enable Register turns off all CMOS switches to prevent random DAC voltages from reaching the analog I/O connector.

The DAC Enable Register — which is automatically cleared by a reset — is programmed by writing to the General Control port with bit 11 set:

General Control Port (base + 40, write only) :

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
V	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0

V — enables (1) or disables (0) all DAC outputs.

DAC Data Ports

Each DAC is allocated two bus register ports, designated DACxLSB and DACxMSB (where x is the DAC identifier 0, 1, 2, or 3). All DAC setpoint data are written into these registers. Offsets from the board base address are shown in the table to the right:

Address Offset	Function
0	DAC0LSB
2	DAC0MSB
4	DAC1LSB
6	DAC1MSB
8	DAC2LSB
10	DAC2MSB
12	DAC3LSB
14	DAC3MSB

For each analog output channel, DACxLSB receives the least-significant data byte, and DACxMSB receives the most-significant data nibble.

In the case of the DACxMSB registers, the data nibble is right-justified and the high nibble is ignored. Data may be written to DACxLSB and DACxMSB registers in any order.

DAC x LSB Register (write only)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
x	x	x	x	x	x	x	x	D7	D6	D5	D4	D3	D2	D1	D0

DAC x MSB Register (write only)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
x	x	x	x	x	x	x	x	x	x	x	x	D11	D10	D9	D8

Note: DAC outputs do not change when the DACxLSB and DACxMSB registers are written to. Outputs change only when the LDAC port is read (see next section).

LDAC Port

DAC outputs change when data is transferred from the bus registers to the DAC output registers. The LDAC port is used to invoke such transfers.

Reading from the LDAC port transfers data to all four DAC output registers simultaneously.

LDAC port (base + 0, read only) : Update DAC outputs

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

The value read from the LDAC port is indeterminate.

Transfer Function

Input coding for DAC channels is 12-bit unsigned binary. The relationship between input code and analog output voltage is illustrated in the table shown to the right:

Code (hex)	Output (Volts)
000	0.0000
001	0.0024
...	...
7FF	4.9976
800	5.0000
801	5.0024
...	...
FFF	9.9976

DAC Initialization

Before setting the DAC Enable Register to enable the analog outputs, the ISAbus master should first zero all DAC output registers.

A typical application should follow this sequence to achieve orderly startup following a reset:

- Write zeros to all DACxLSB and DACxMSB registers.
- Read from the LDAC port to zero all DAC outputs.
- Enable DAC outputs via the Control port.

Channel Differences

DAC channel 0 is identical to the other channels except for the addition of a remote sense function.

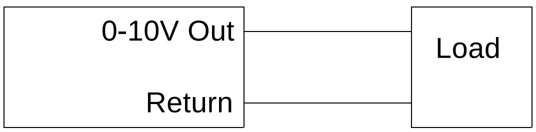
Each channel has an output source impedance specified at 85 ohms, maximum. Because channel 0 senses the DAC output after its CMOS switch, however, its effective source impedance is zero for output currents up to the specified maximum.

Because of their 85 ohm source impedances, channels 1, 2, and 3 will tend to exhibit “gain error” as a function of load current. This is generally not a problem in control applications as long as the load current is constant, and hence, DAC output voltage is monotonic.

Use channel 0 if load impedance varies significantly or absolute accuracy is important for your application. Alternately, any of the channel 1, 2, or 3 outputs may be externally buffered.

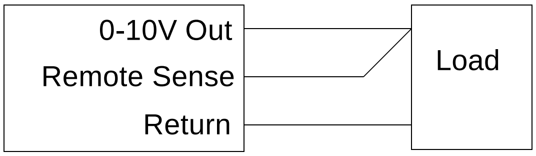
Connections: Channel 1, 2, and 3

Each DAC channel makes two connections to the 425: 0 to +10V output, and output return.



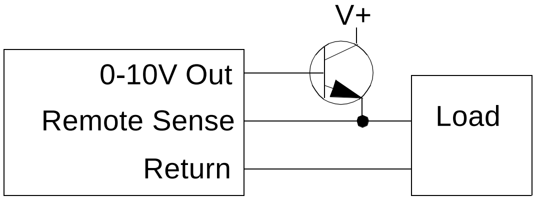
Connections: Channel 0

In addition to output and return, channel 0 has a connection for remote voltage sensing. The remote sense signal must be connected to the output signal. Connect the output and sense signals near to the load for best accuracy.



Connections: Channel 0 with External Buffer

In applications that use an external buffer for DAC channel 0, it is best to sense the voltage at the buffer output instead of the DAC output.



Connector Pinout

All external circuitry connects to the DAC channels through 40-pin header P2. Connector P2 is shared by the D/A and A/D converter circuits.

This table shows the portion of connector P2 used by DAC channels:

Connector P2 Pinout

Channel	Pin	Function
0	1	Output
	19	Sense
	3	Return
1	5	Output
	7	Return
2	9	Output
	11	Return
3	13	Output
	15	Return

A/D Interface

The digitizer consists of an eight-channel differential analog multiplexer, instrumentation amplifier, and a 12-bit analog-to-digital converter. A/D conversions are accomplished as follows:

- Select the conversion mode: unipolar or bipolar
- Select the analog input channel to be digitized.
- Start an A/D conversion.
- Wait for the conversion to finish.
- Read the conversion result.

Mode Selection

The digitizer input mode may be configured for either unipolar or bipolar operation. The input mode is programmed by writing to the General Control port with bit 10 set:

General Control Port (base + 40, write only) : Set A/D Mode

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
V	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0

V specifies input mode according to table shown at right. Unipolar is the default mode after a reset.

V	Mode
0	Unipolar
1	Bipolar

Channel Selection

A channel is selected for conversion by writing the channel number to the A/D Channel Register:

A/D Channel Register (base + 22, write only)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	D2	D1	D0

Valid channel numbers range from 0 to 7, inclusive.

Settling time must be allowed between writing to the Channel Register and starting a conversion. 14µs is adequate for low impedance analog sources. If you are digitizing high impedance sources or employing high gain, longer settling time may be required to meet specified performance.

Note: it is not necessary to repeatedly write to the A/D Channel Register if your application dwells on one channel. No settling time is required in such cases.

Starting a Conversion

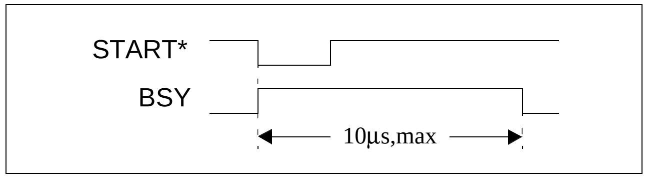
Conversions are started by writing to the General Control port with bits 15 and 12 set:

General Control port (base + 40, write only) : Start Conversion

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0

This starts a conversion and sets the status register "busy" flag. When the conversion ends — typically within 9µs after starting — the busy flag is cleared and the result may be read.

Timing Diagram



Name	Signal Description
START*	A/D start strobe. Coincident with host CPU write strobe to Start A/D Conversion port
BSY	"A/D busy" flag as viewed by host through Status port

Reading the Conversion Result

Digitized data may be read from the A/D Data Register any time no conversion is in progress:

A/D Data Register (base + 16, read only)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0

The 16-bit output data format is a function of the digitizer input mode:

Unipolar Mode

Output data is formatted as unsigned integer with leading zeros in the most-significant nibble. Assuming unity gain, the table to the right illustrates the relationship between input voltage and output codes.

Input	Data
0.000V	0000
2.441mV	0001
...	...
4.998V	07FF
5.000V	0800
5.002V	0801
...	...
+9.997V	0FFF

Bipolar Mode

Output data is formatted as signed integer with sign extension through the most-significant nibble. Assuming unity gain, the table to the right illustrates the relationship between input voltage and output codes.

Input	Data
-5.000V	F800
-4.997V	F801
...	...
-2.441mV	FFFF
0.000V	0000
+2.441mV	0001
...	...
+4.997V	07FF

Gain Programming

Digitizer gain may be set to unity or higher by means of a programming resistor. All analog input channels have the same gain.

Gain is programmed by installing a resistor at position R8. The value of resistor R8 can be calculated for any arbitrary gain using this formula:

$$R = \frac{49400}{Gain - 1}$$

To prevent thermally-induced gain errors, R8 should have a low temperature coefficient — preferably 50 ppm/C or less.

The table below shows required R8 values for various gains using standard 1% resistor values. The listed gain values are approximate and may be trimmed to exact values as described in the next section.

R8 (ohms)	Gain	Input Range	
		Unipolar	Bipolar
Open	1	0 to +10V	-5 to +5V
49.9K	2	0 to +5V	-2.5 to +2.5V
12.4K	5	0 to +2V	-1V to +1V
5.49K	10	0 to +1V	-500 to +500mV
2.61K	20	0 to +500mV	-250 to +250mV
1.00K	50	0 to +200mV	-100 to +100mV
499	100	0 to +100mV	-50 to +50mV
249	200	0 to +50mV	-25 to +25mV
100	500	0 to +20mV	-10 to +10mV
49.9	1000	0 to +10mV	-5 to +5mV

As shipped from the factory, digitizer gain is set to unity (R8 not installed).

Gain Trim

Digitizer gain may be trimmed by adjusting potentiometer R1. This trimmer provides an adjustment range of approximately ± 7 percent.

Offset Trim

Digitizer offset may be trimmed by adjusting potentiometer R2. This trimmer provides an adjustment range of approximately $\pm 6\text{mV}$. Since this adjustment may alter circuit gain, offset should be trimmed before adjusting gain.

Input Connections

Each analog input channel makes two connections to the board as a differential pair. The digitizer measures the voltage across any such pair.

Input common-mode voltage — the voltage at either input relative to the ISAbus five volt return (GND) — should not exceed ± 10 volts. Minor excursions beyond this limit may cause measurement errors. Significant excursions may cause damage to digitizer circuitry. Refer to the specifications for further details.

All analog input sources should be referenced to ISAbus GND. If you are connecting an isolated source (a source not referenced to GND), you should tie one side of the source to GND. Since analog input channels are differential pairs, it doesn't matter which side is connected to GND.

Connector Pinout

All analog inputs connect to connector P2. This connector is shared by analog input and analog output functions. The table below describes that portion of P2 related to analog input functions. Note that the GND signals are related to both analog input and output functions.

Connector P2: Analog I/O

Chan	Pin	Name	Function
0	2	ADC0+	+ input
	4	ADC0-	- input
1	6	ADC1+	+ input
	8	ADC1-	- input
2	10	ADC2+	+ input
	12	ADC2-	- input
3	14	ADC3+	+ input
	16	ADC3-	- input
4	18	ADC4+	+ input
	20	ADC4-	- input
5	22	ADC5+	+ input
	24	ADC5-	- input
6	26	ADC6+	+ input
	28	ADC6-	- input
7	30	ADC7+	+ input
	32	ADC7-	- input
	33,34, 38,39, 40	GND	Analog common

Appendix A: Specifications

General

Interface	Parameter	Description
Bus	Type	ISA, I/O slave, 16-bit address, 16-bit data
	Address requirements	64-address block, starting on any 64-address boundary
	Interrupt level	Jumper selectable to any one IRQ level: 3-7, 9-12, 14, 15
Encoder	Type	Quadrature encoded, single-ended
	Input characteristics	RS422 differential, single-ended TTL/CMOS compatible
Relay	Input characteristics	TTL/CMOS compatible, 10Kohm pullup to +5V
A/D	A/D type	12-bits, successive approximation, resistor programmed gain
D/A	D/A type	12 bits, voltage output, output range fixed at 0 to +10V

Limits

	Parameter	Typical	Min	Max	Units
Watchdog	Timeout (Fast mode)	700	630	770	msec
	Timeout (Slow mode)	11.2	10.0	12.4	sec
Encoders	Pulse rate		0	375	KHz
	Supply current, per chan.			100	mA
Relays	Output sink current		100		mA
	Captured pulse width		200		ns
A/D	DNL		-1	+1	LSB
	INL		-1	+1	LSB
	T_{SETTLE} to 12 bits (G = 1)	5		9	us
	$T_{CONVERT}$			10	us
	CMRR (G = 1)	90	73		dB
	CMV for spec'd accuracy		-10	+10	V
	Input limits (abs. max.)		-25	+25	V
	Offset error		-1	+1	LSB
	Input Z		10		Mohm
D/A	Gain error		-3	+3	LSB
	Output offset		-7.5	+7.5	mV
	Output Z (chan 1, 2 & 3)			85	ohms
Power	Operating range		+4.75	+5.25	V
	Quiescent current	200			mA
Temperature	Operating range		0	70	ºC

Appendix B: I/O Map

Addr	Write Function	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Read Function	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
00	**DAC 0 LSB**	x	x	x	x	x	x	x	x	D7	D6	D5	D4	D3	D2	D1	D0	**Transfer DAC data registers to outputs**	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
02	**DAC 0 MSB**	x	x	x	x	x	x	x	x	x	x	x	x	D11	D10	D9	D8																	
04	**DAC 1 LSB**	x	x	x	x	x	x	x	x	D7	D6	D5	D4	D3	D2	D1	D0																	
06	**DAC 1 MSB**	x	x	x	x	x	x	x	x	x	x	x	x	D11	D10	D9	D8																	
08	**DAC 2 LSB**	x	x	x	x	x	x	x	x	D7	D6	D5	D4	D3	D2	D1	D0																	
0A	**DAC 2 MSB**	x	x	x	x	x	x	x	x	x	x	x	x	D11	D10	D9	D8																	
0C	**DAC 3 LSB**	x	x	x	x	x	x	x	x	D7	D6	D5	D4	D3	D2	D1	D0																	
0E	**DAC 3 MSB**	x	x	x	x	x	x	x	x	x	x	x	x	D11	D10	D9	D8																	
10	**Counter Control Port**	V	x	x	CM1	CM0	ARM	IEN	x	C1	C0	A1	A0	SRC	DIR	M1	M0	**A/D Data Register**	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
12	**Counter Preload Register Low Word**	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	**Counter Holding Register Low Word**	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
14	**Counter Preload Register High Word**	x	x	x	x	x	x	x	x	D23	D22	D21	D20	D19	D18	D17	D16	**Counter Holding Register High Word**	D23	D23	D23	D23	D23	D23	D23	D23	D23	D22	D21	D20	D19	D18	D17	D16
16	**A/D Channel Register**	0	0	0	0	0	0	0	0	0	0	0	0	0	D2	D1	D0	**Status Register**	FLT	CP2	CP1	CP0	RQR	RQ2	RQ1	RQ0	x	IE#	WDI	WDR	BSY	AE	WDE	BIP
18	**Relay Channels 0-15**	R15	R14	R13	R12	R11	R10	R9	R8	R7	R6	R5	R4	R3	R2	R1	R0	**Relay Channels 0-15**	R15	R14	R13	R12	R11	R10	R9	R8	R7	R6	R5	R4	R3	R2	R1	R0
1A	**Relay Channels 16-31**	R31	R30	R29	R28	R27	R26	R25	R24	R23	R22	R21	R20	R19	R18	R17	R16	**Relay Channels 16-31**	R31	R30	R29	R28	R27	R26	R25	R24	R23	R22	R21	R20	R19	R18	R17	R16
1C	**Relay Channels 32-47**	R47	R46	R45	R44	R43	R42	R41	R40	R39	R38	R37	R36	R35	R34	R33	R32	**Relay Channels 32-47**	R47	R46	R45	R44	R43	R42	R41	R40	R39	R38	R37	R36	R35	R34	R33	R32
1E	**Relay Channel 0-15 Interrupt Enables**	R15	R14	R13	R12	R11	R10	R9	R8	R7	R6	R5	R4	R3	R2	R1	R0	**Relay Channel 0-15 Edge Capture Flags**	R15	R14	R13	R12	R11	R10	R9	R8	R7	R6	R5	R4	R3	R2	R1	R0

IMPORTANT NOTE: shaded areas in I/O map designate addresses that are reserved for future use. In the current implementation of Model 425 architecture, these locations should not be accessed. Unpredictable behavior may result when some reserved locations are accessed by the ISAbus master.

	Write Function	Read Function
20		
22	**Relay Channel 0-15 Edge Select Register** 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 R15 R14 R13 R12 R11 R10 R9 R8 R7 R6 R5 R4 R3 R2 R1 R0	
24	**Relay Channel 0-15 Disarm Edge Detection** 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 R15 R14 R13 R12 R11 R10 R9 R8 R7 R6 R5 R4 R3 R2 R1 R0	
26	**Relay Channel 0-15 Arm Edge Detection** 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 R15 R14 R13 R12 R11 R10 R9 R8 R7 R6 R5 R4 R3 R2 R1 R0	
28	**General Control Port** 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 V IE# WDI DIG DE BIP WDE HIT x x x x x x x x	
2A		
2C		
2E		
30		
32		
34		
36		**Extended Status Register** 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 x CX2 CX1 CX0 x x x x x x x x x x x x
38		
3A		
3C		
3E	**Board Reset** 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	

Appendix C: Connector Pinouts

Connector P3 and P1: Digital I/O

Pin	P1	P3
1	CH23	CH47
3	CH22	CH46
5	CH21	CH45
7	CH20	CH44
9	CH19	CH43
11	CH18	CH42
13	CH17	CH41
15	CH16	CH40
17	CH15	CH39
19	CH14	CH38
21	CH13	CH37
23	CH12	CH36
25	CH11	CH35
27	CH10	CH34
29	CH9	CH33
31	CH8	CH32
33	CH7	CH31
35	CH6	CH30
37	CH5	CH29
39	CH4	CH28
41	CH3	CH27
43	CH2	CH26
45	CH1	CH25
47	CH0	CH24
49	+5V	+5V
Even	GND	GND

Connector J1: Encoder Inputs

Chan	Pin	Function
0	1	Phase A input (-)
	2	Phase A input (+)
	3	**GND**
	4	Phase B input (-)
	5	Phase B input (+)
	6	**+5V**
	7	Index input (-)
	8	Index input (+)
	9	**GND**
1	10	Phase A input (-)
	11	Phase A input (+)
	12	**+5V**
	13	Phase B input (-)
	14	Phase B input (+)
	15	**GND**
	16	Index input (-)
	17	Index input (+)
	18	**+5V**
2	19	Phase A input (-)
	20	Phase A input (+)
	21	**GND**
	22	Phase B input (-)
	23	Phase B input (+)
	24	**+5V**
	25	Index input (-)
	26	Index input (+)

Connector P2: Analog I/O

D/A Chan	A/D Chan	Pin	Name	Function
0		1	DAC0+	+ Out
	0	2	ADC0+	+ In
0		3	DAC0-	- Out
	0	4	ADC0-	- In
1		5	DAC1+	+ Out
	1	6	ADC1+	+ In
1		7	DAC1-	- Out
	1	8	ADC1-	- In
2		9	DAC2+	+ Out
	2	10	ADC2+	+ In
2		11	DAC2-	- Out
	2	12	ADC2-	- In
3		13	DAC3+	+ Out
	3	14	ADC3+	+ In
3		15	DAC3-	- Out
	3	16	ADC3-	- In
		17	—	no connect
	4	18	ADC4+	+ In
0		19	DAS0+	+ Sense
	4	20	ADC4-	- In
		21	—	no connect
	5	22	ADC5+	+ In
		23	—	no connect
	5	24	ADC5-	- In
		25	—	no connect
	6	26	ADC6+	+ In
		27	—	no connect
	6	28	ADC6-	- In
		29	—	no connect
	7	30	ADC7+	+ In
		31	—	no connect
	7	32	ADC7-	- In
		33	GND	Common
		34	GND	Common
		35	—	no connect
		36	+15V	+15 Volts
		37	—	no connect
		38	GND	Common
		39	GND	Common
		40	GND	Common

Connector P4: Watchdog Timer Output

Pin	Name	Function
1	-Reset	Output to ISAbus reset connector
2	GND	ISAbus 5V return

Appendix D: Programming

Sample Visual Basic and C/C++ code segments are included on the 425 distribution diskette. If you are coding your application in a language other than Visual Basic or C/C++, you may find these listings useful as a framework for writing your own drivers.

Although intended as programming examples, these listings may be used as is (or with minor modification) as a basic set of interface drivers for the 425. In particular, the C/C++ listing may be easily converted to a dynamic link library for use with a Windows-based application.

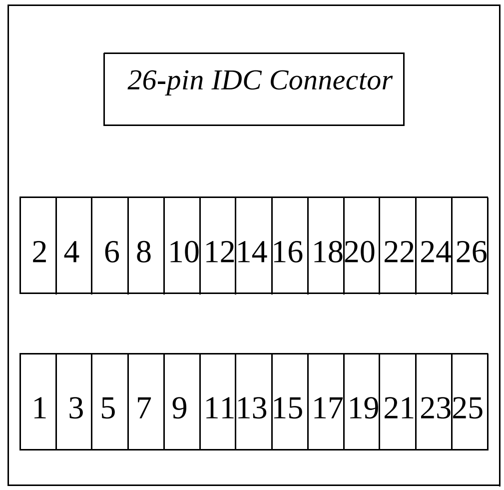
Feel free to modify, plagiarize and adapt these listings to suit your application requirements. Although these sample listings are believed to be complete and functionally correct, no responsibility is assumed by Sensoray for typographic or other errors contained herein. Note that these listings, and other related information, may be available in electronic form at Sensoray's internet FTP site.

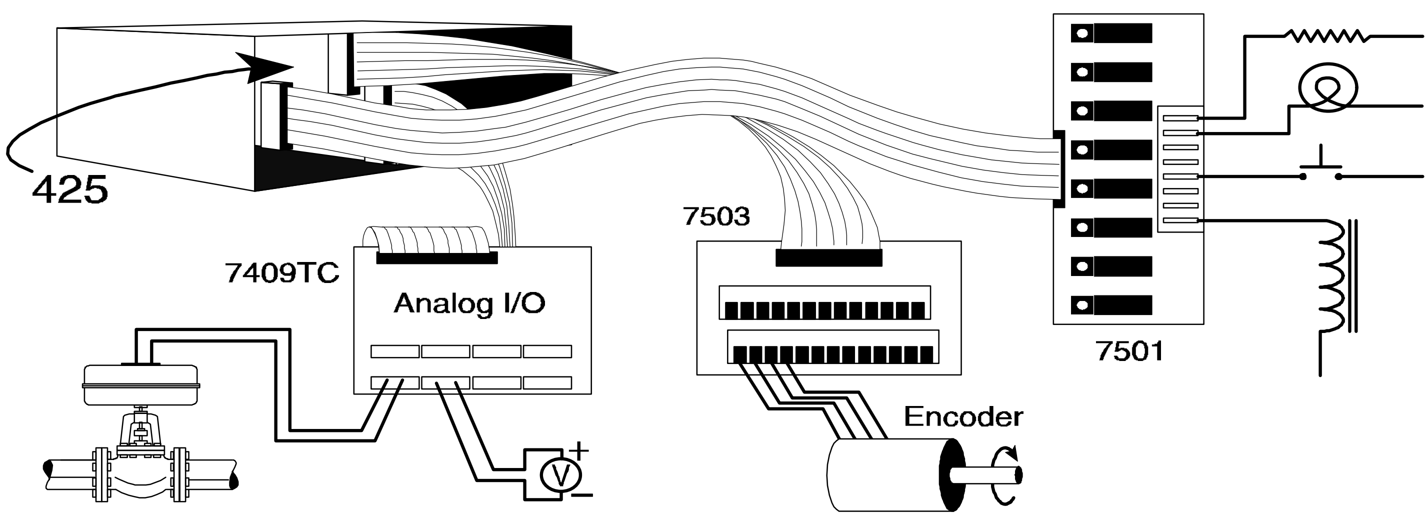
Appendix E: Model 7503TDIN Termination Board

Model 425 connector J1 is used to make all electrical connections to external encoders or other pulse sources. Such sources may be connected to an optional model 7503TDIN screw termination board, which in turn connects to 425 connector J1 by means of a 26-conductor flat cable. Sources may be electrically connected and disconnected from the 425 or 7503TDIN without removing the 425 circuit board from the backplane.

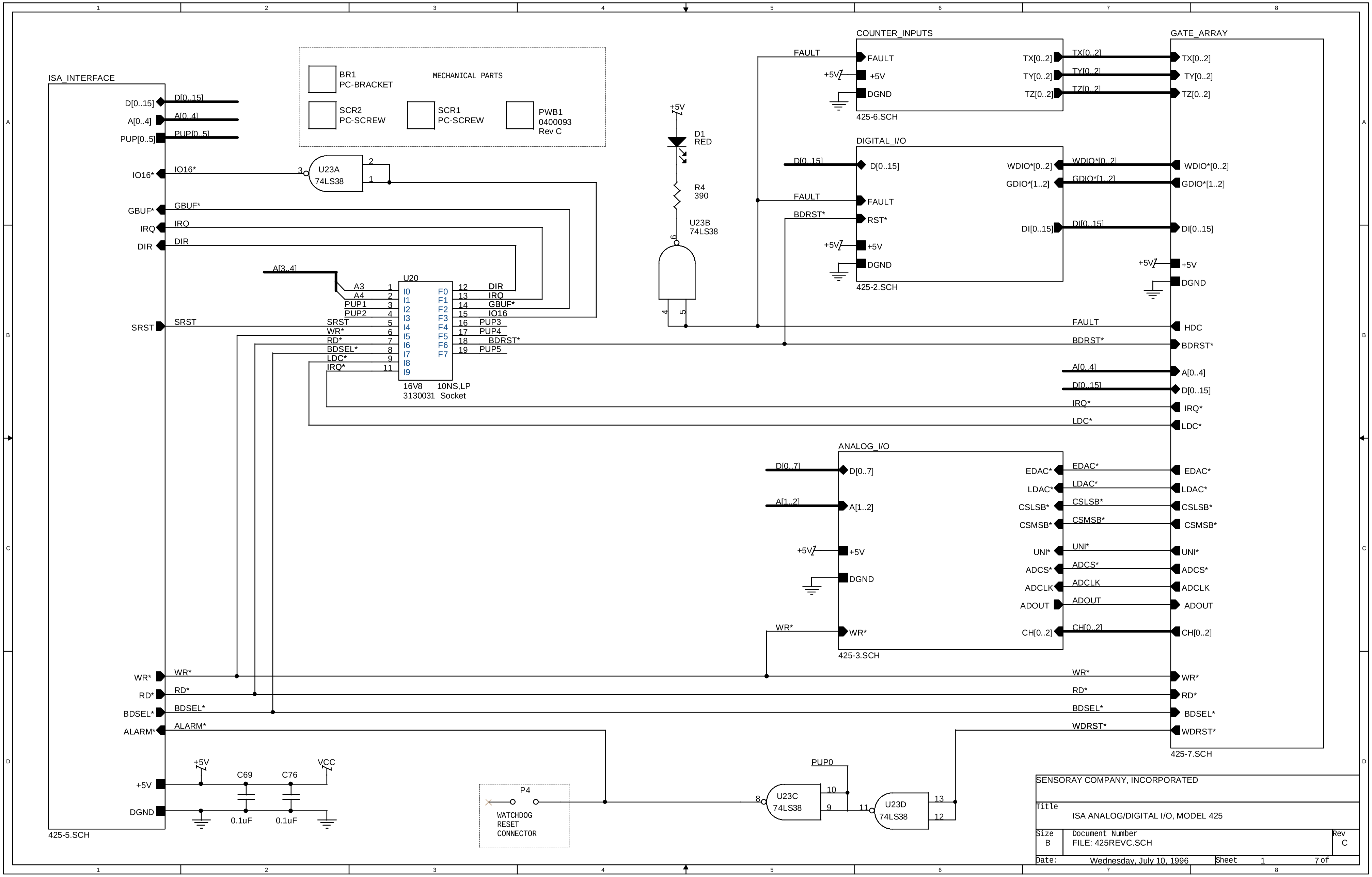
Pin	Name	Function
1	$-A_0$	Channel 0 phase A input (-)
2	$+A_0$	Channel 0 phase A input (+)
3	GND	GND
4	$-B_0$	Channel 0 phase B input (-)
5	$+B_0$	Channel 0 phase B input (+)
6	+5V	+5V
7	$-X_0$	Channel 0 index input (-)
8	$+X_0$	Channel 0 index input (+)
9	GND	GND
10	$-A_1$	Channel 1 phase A input (-)
11	$+A_1$	Channel 1 phase A input (+)
12	+5V	+5V
13	$-B_1$	Channel 1 phase B input (-)
14	$+B_1$	Channel 1 phase B input (+)
15	GND	GND
16	$-X_1$	Channel 1 index input (-)
17	$+X_1$	Channel 1 index input (+)
18	+5v	+5v
19	$-A_2$	Channel 2 phase A input (-)
20	$+A_2$	Channel 2 phase A input (+)
21	GND	GND
22	$-B_2$	Channel 2 phase B input (-)
23	$+B_2$	Channel 2 phase B input (+)
24	+5V	+5V
25	$-X_2$	Channel 2 index input (-)
26	$+X_2$	Channel 2 index input (+)

Model 7503TDIN Connector Layout





Appendix F: Schematics

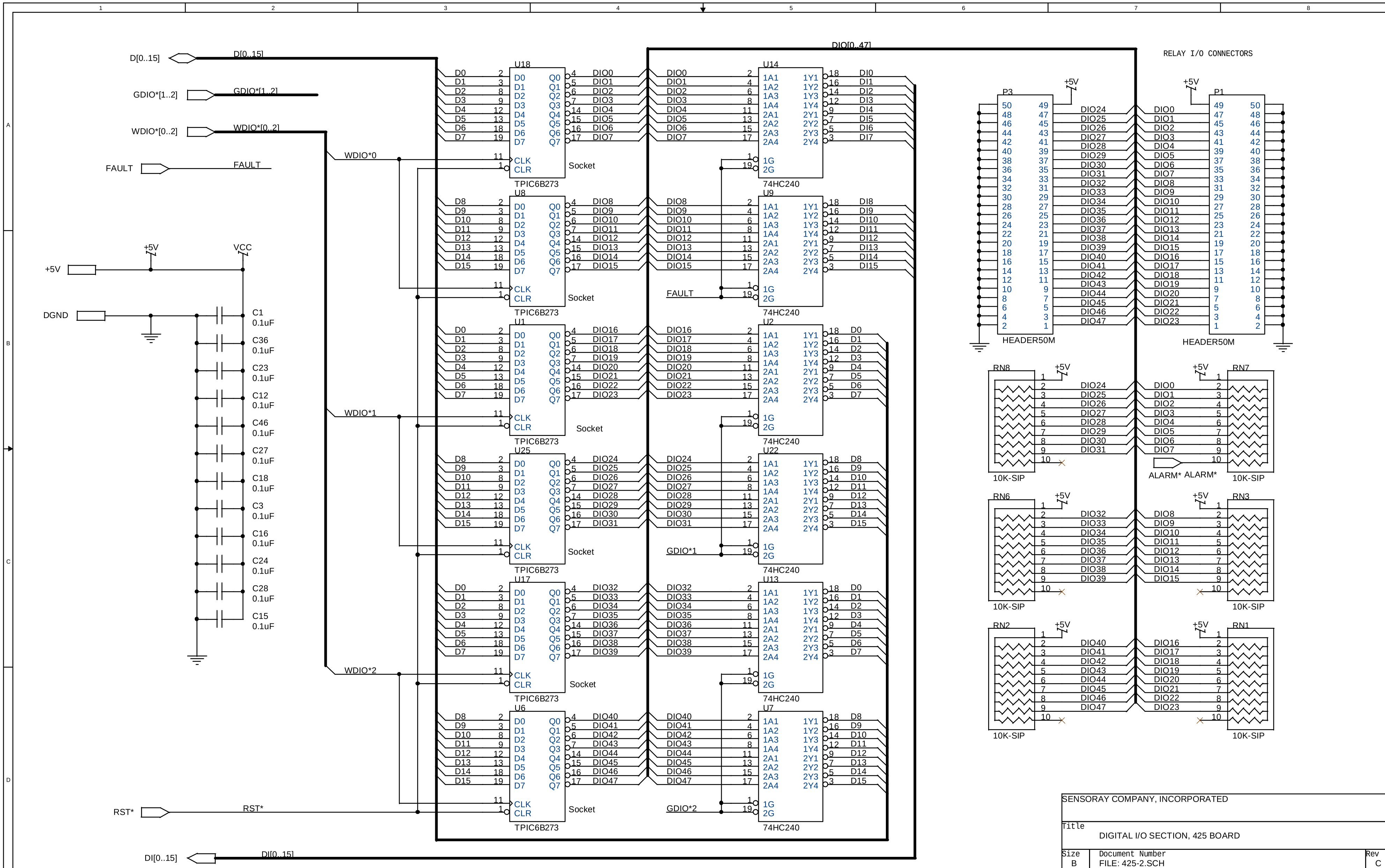


SENSORAY COMPANY, INCORPORATED

Title: ISA ANALOG/DIGITAL I/O, MODEL 425

Size: B | Document Number: FILE: 425REVC.SCH | Rev: C

Date: Wednesday, July 10, 1996 | Sheet 1 of 7

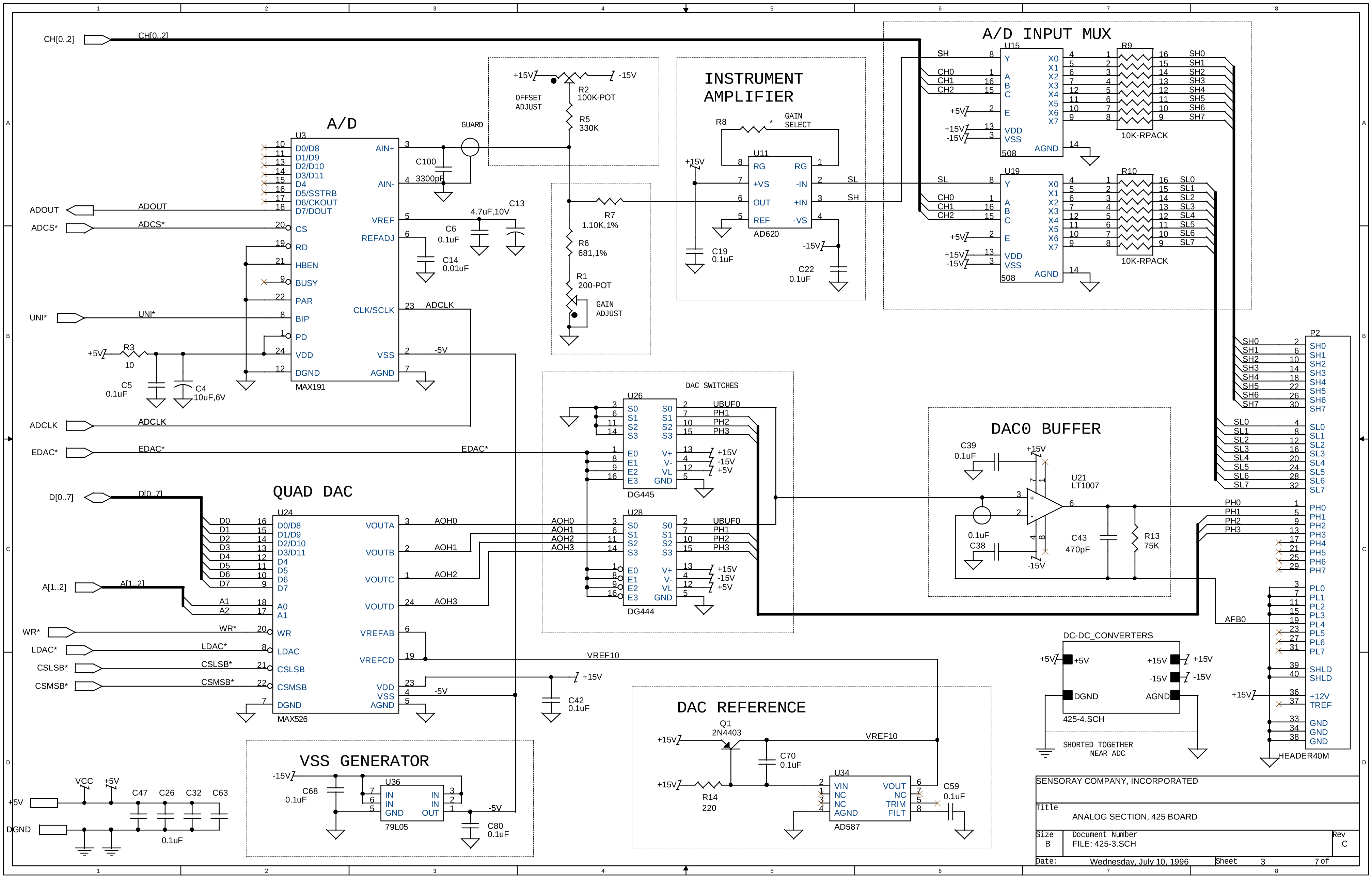


SENSORAY COMPANY, INCORPORATED

Title DIGITAL I/O SECTION, 425 BOARD

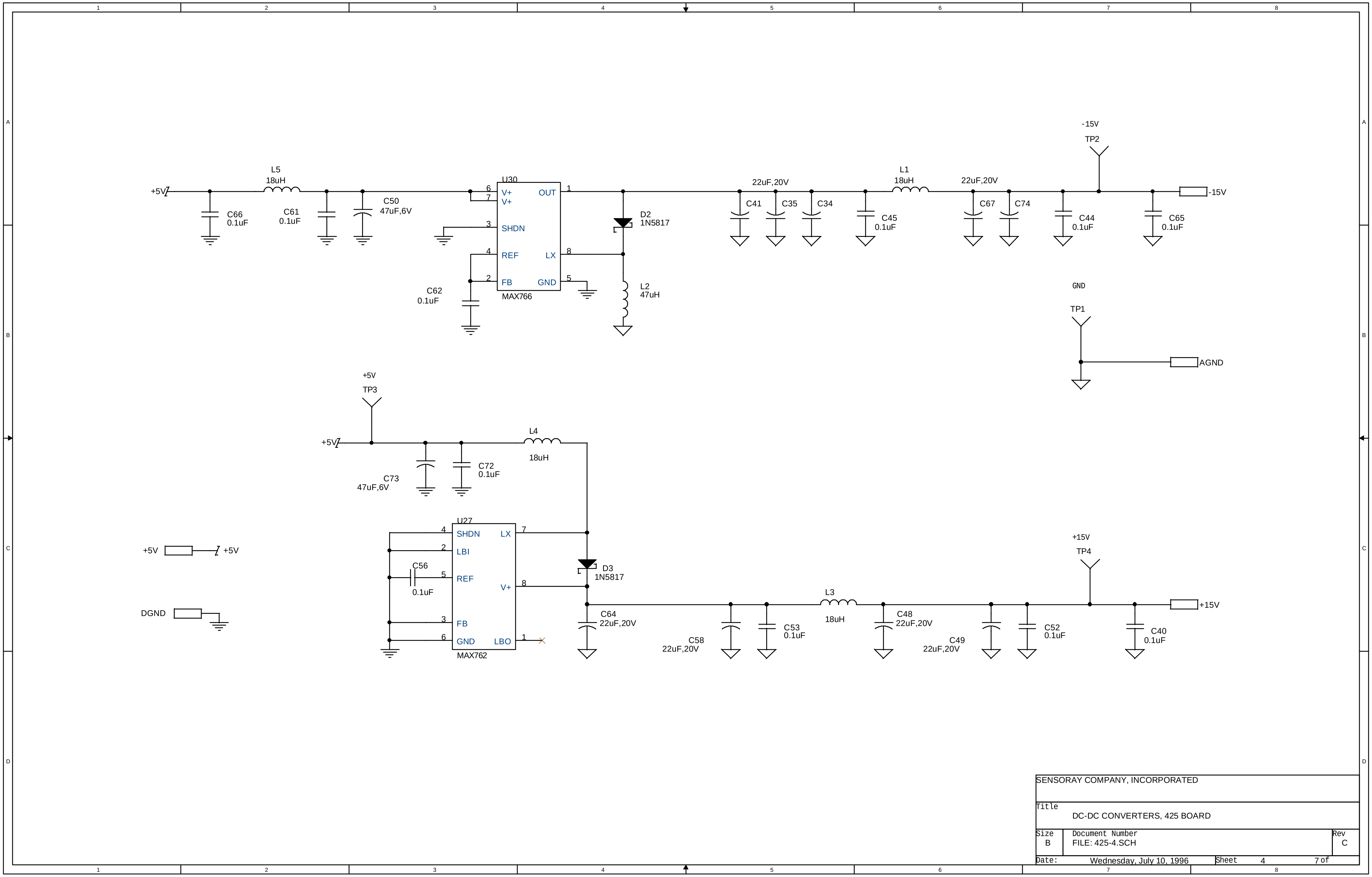
Size	Document Number	Rev
B	FILE: 425-2.SCH	C

Date: Wednesday, July 10, 1996 Sheet 2 7 of

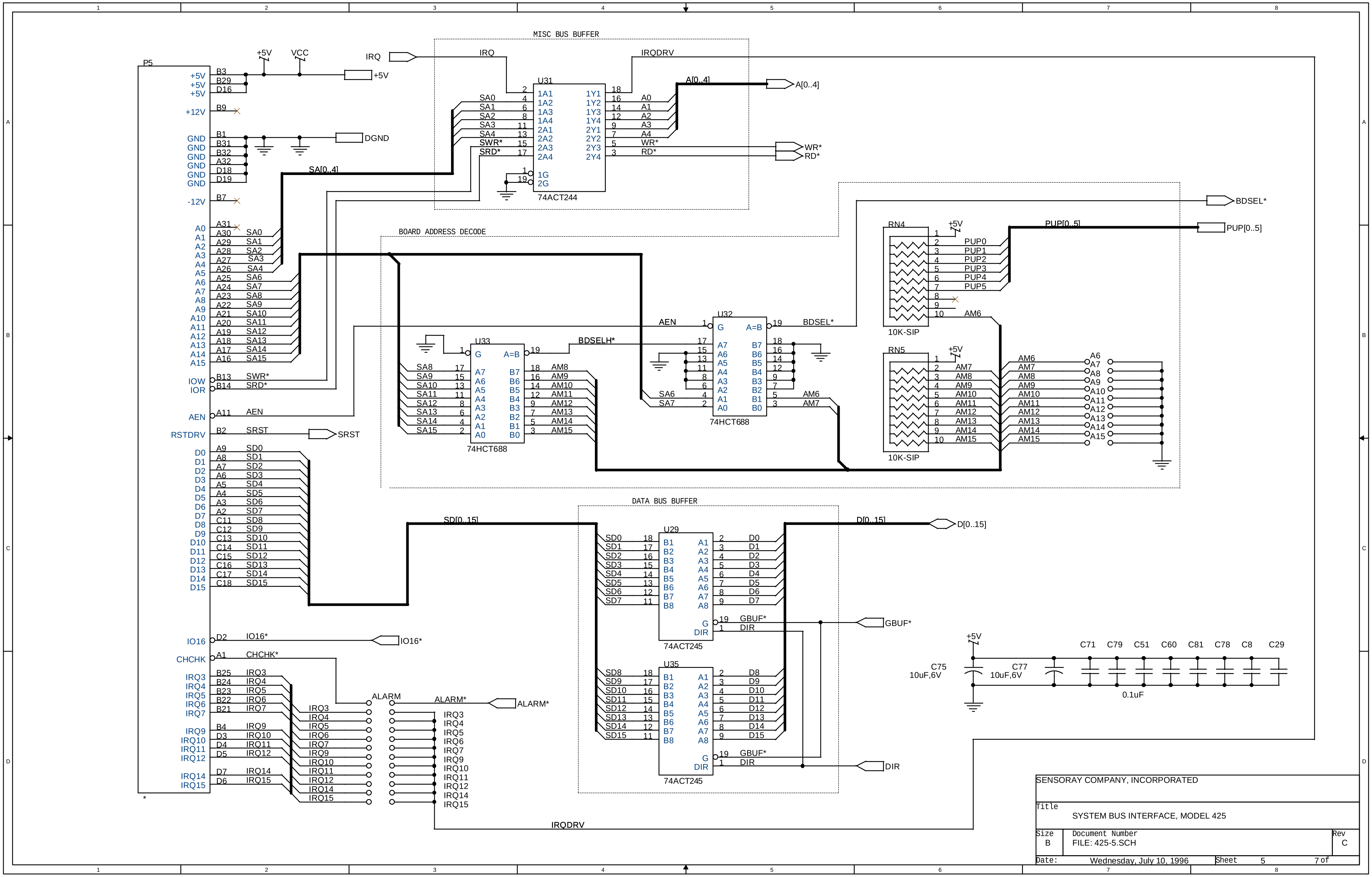


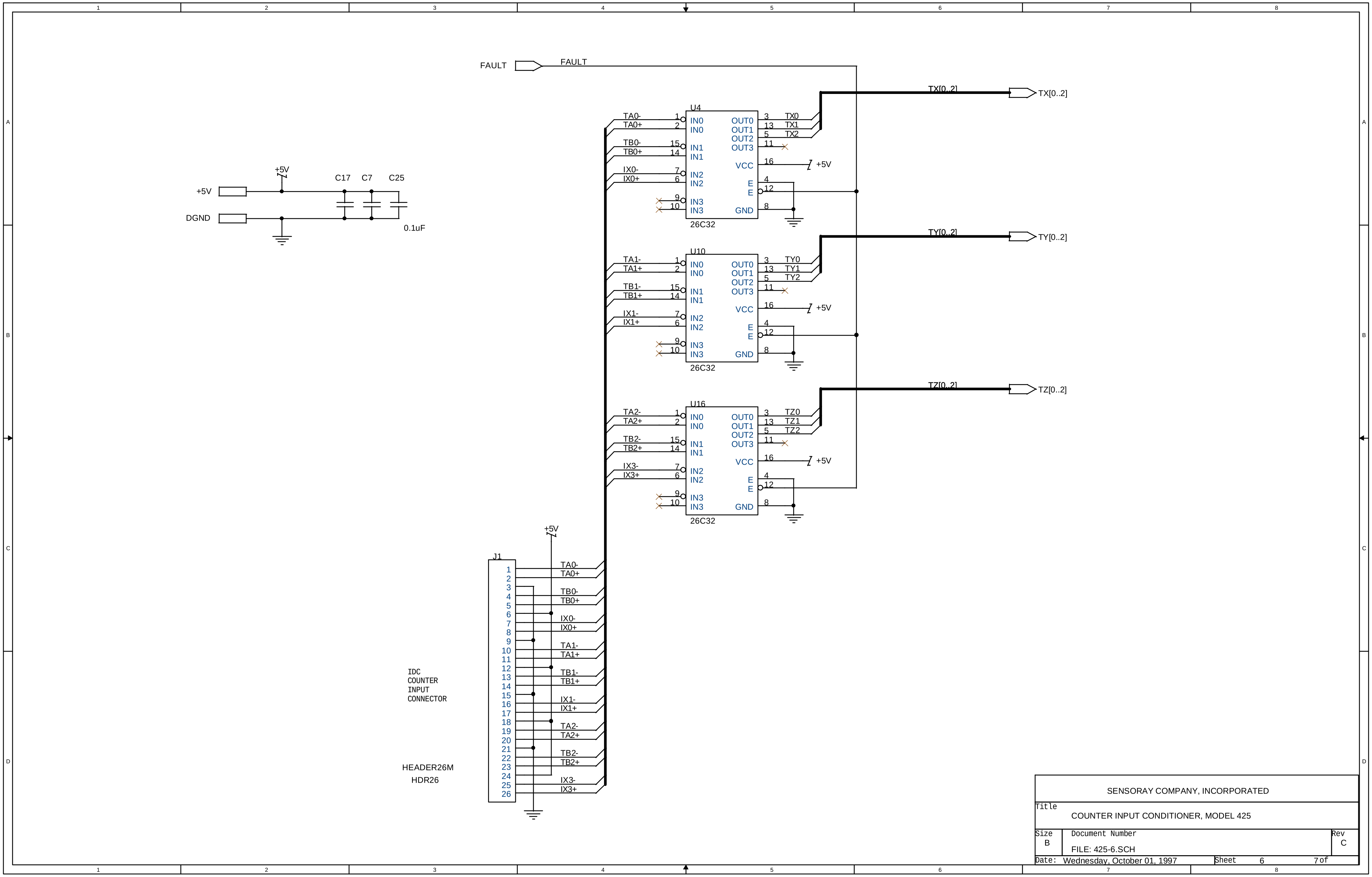
SENSORAY COMPANY, INCORPORATED

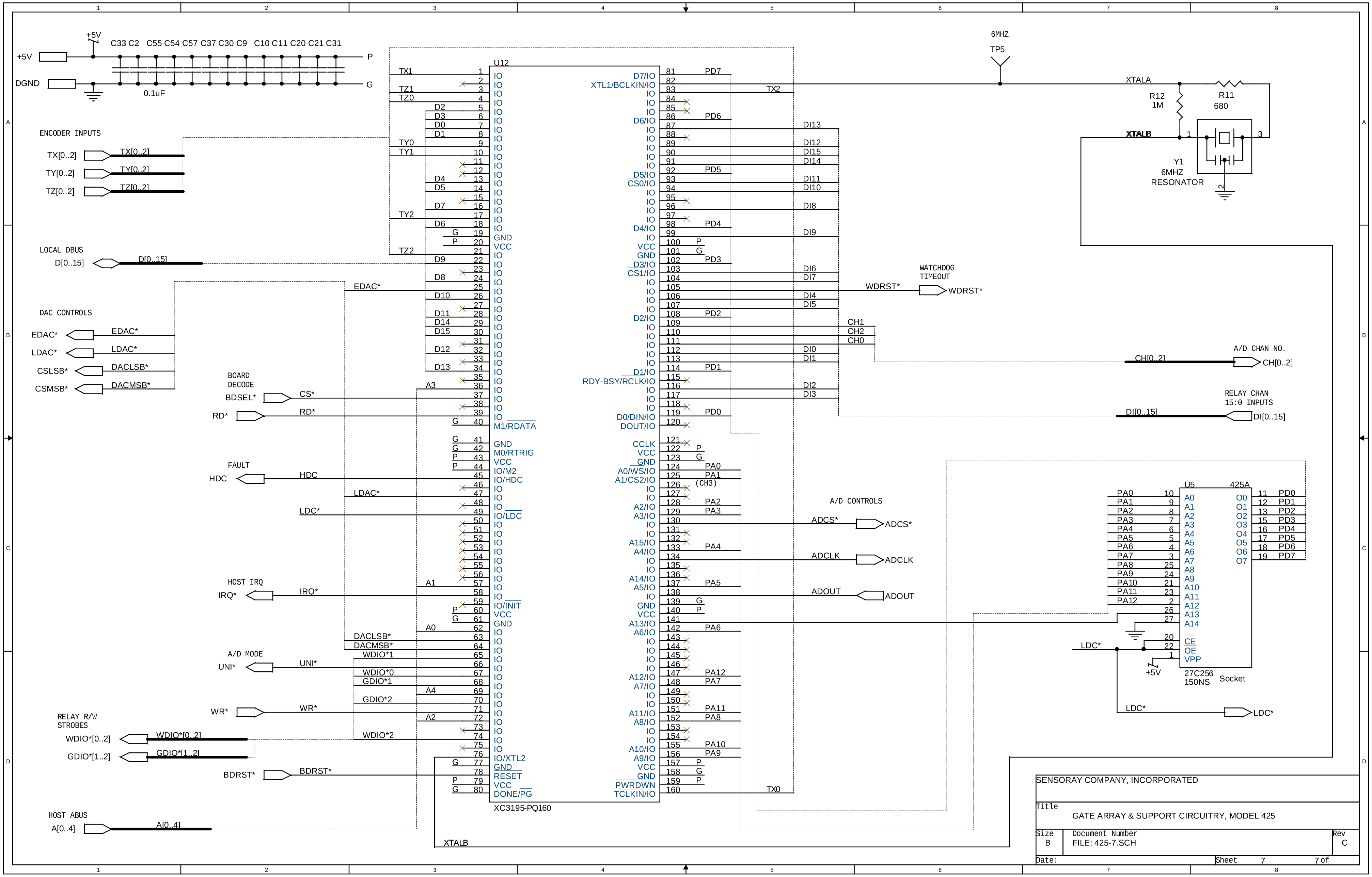
Title	ANALOG SECTION, 425 BOARD
Size B	Document Number FILE: 425-3.SCH — Rev C
Date: Wednesday, July 10, 1996	Sheet 3 of 7



U30
MAX766
V+
V+
SHDN
REF
FB
OUT
LX
GND
L5
18uH
C66
0.1uF
C61
0.1uF
C50
47uF,6V
C62
0.1uF
D2
1N5817
L2
47uH
22uF,20V
C41
C35
C34
C45
0.1uF
L1
18uH
22uF,20V
C67
C74
C44
0.1uF
C65
0.1uF
-15V
TP2
-15V
GND
TP1
AGND
+5V
TP3
+5V
L4
18uH
C73
47uF,6V
C72
0.1uF
U27
MAX762
SHDN
LBI
REF
FB
GND
LX
V+
LBO
C56
0.1uF
D3
1N5817
C64
22uF,20V
C58
22uF,20V
C53
0.1uF
L3
18uH
C48
22uF,20V
C49
22uF,20V
C52
0.1uF
+15V
TP4
C40
0.1uF
+15V
+5V
+5V
DGND
SENSORAY COMPANY, INCORPORATED
Title
DC-DC CONVERTERS, 425 BOARD
Size
B
Document Number
FILE: 425-4.SCH
Rev
C
Date: Wednesday, July 10, 1996
Sheet 4 of 7


MISC BUS BUFFER
BOARD ADDRESS DECODE
DATA BUS BUFFER
P5
U31
74ACT244
U32
74HCT688
U33
74HCT688
U29
74ACT245
U35
74ACT245
RN4
RN5
10K-SIP
IRQ
IRQDRV
A[0..4]
SA[0..4]
WR*
RD*
DGND
BDSEL*
BDSELH*
PUP[0..5]
AEN
SRST
SD[0..15]
D[0..15]
GBUF*
DIR
IO16*
CHCHK*
ALARM
ALARM*
C75
10uF,6V
C77
10uF,6V
C71
C79
C51
C60
C81
C78
C8
C29
0.1uF
SENSORAY COMPANY, INCORPORATED
Title
SYSTEM BUS INTERFACE, MODEL 425
Size
B
Document Number
FILE: 425-5.SCH
Rev
C
Date:
Wednesday, July 10, 1996
Sheet
5
7 of


FAULT
U4
TA0-
TA0+
TB0-
TB0+
IX0-
IX0+
IN0 IN0 IN1 IN1 IN2 IN2 IN3 IN3
OUT0 OUT1 OUT2 OUT3 VCC E E GND
TX0
TX1
TX2
26C32
TX[0..2]
U10
TA1-
TA1+
TB1-
TB1+
IX1-
IX1+
TY0
TY1
TY2
TY[0..2]
U16
TA2-
TA2+
TB2-
TB2+
IX3-
IX3+
TZ0
TZ1
TZ2
TZ[0..2]
+5V
DGND
C17
C7
C25
0.1uF
J1
IDC
COUNTER
INPUT
CONNECTOR
HEADER26M
HDR26
SENSORAY COMPANY, INCORPORATED
Title
COUNTER INPUT CONDITIONER, MODEL 425
Size B
Document Number
FILE: 425-6.SCH
Rev C
Date: Wednesday, October 01, 1997
Sheet 6 of 7


SENSORAY COMPANY, INCORPORATED
Title
GATE ARRAY & SUPPORT CIRCUITRY, MODEL 425
Size
B
Document Number
FILE: 425-7.SCH
Rev
C
Date:
Sheet 7 7 of
U12
XC3195-PQ160
U5
425A
27C256
150NS
Socket
Y1
6MHZ
RESONATOR
R11
680
R12
1M
TP5
6MHZ
XTALA
XTALB
C33 C2 C55 C54 C57 C37 C30 C9 C10 C11 C20 C21 C31
0.1uF
+5V
DGND
ENCODER INPUTS
TX[0..2]
TY[0..2]
TZ[0..2]
LOCAL DBUS
D[0..15]
DAC CONTROLS
EDAC*
LDAC*
CSLSB*
CSMSB*
BOARD
DECODE
BDSEL*
RD*
FAULT
HDC
HOST IRQ
IRQ*
A/D MODE
UNI*
WR*
RELAY R/W
STROBES
WDIO*[0..2]
GDIO*[1..2]
BDRST*
HOST ABUS
A[0..4]
WATCHDOG
TIMEOUT
WDRST*
A/D CHAN NO.
CH[0..2]
RELAY CHAN
15:0 INPUTS
DI[0..15]
A/D CONTROLS
ADCS*
ADCLK
ADOUT
LDC*