ICS-130

OPERATING MANUAL

Interactive Circuits And Systems Ltd.
November 2000

Extra copies of this manual are available from the factory.

ICS Ltd
5430 Canotek Road
Gloucester, Ontario
K1J 9G2 Canada

http://www.ics-ltd.com
support@ics-ltd.com

Tel: (613) 749-9241
USA: (800) 267-9794
Fax: (613) 749-9461

DOC E10523 Rev.D

TABLE OF CONTENTS

1. INTRODUCTION

The ICS-130 is a 32-channel, 16-bit, VMEbus data acquisition card with an output rate of 1.2 MSample/sec. The board offers multiple options for generating and moving data at high speed. This design has been optimized for applications which demand high speed, precision and ease of integration.

ICS offers software drivers for the ICS-130 for a number of platforms (including VxWorks and Solaris). Use of one of these drivers is strongly recommended, since they greatly simplify control and operation of the ICS-130. This will generally save the programmer much time since he/she is relieved of the requirement to understand the complexities of the ICS-130 hardware model and of the Tundra Semiconductor SCV64 VMEbus interface device. Contact ICS for further details about available drivers.

1.1 References

1. OpenBus Interface Components, SCV64 User Manual, Document No. 891078.MD301.01, Newbridge Microsystems, 1993.

2. VxWorks Device Driver Manual for the ICS-130, Document No. E10524, Interactive Circuits and Systems Ltd., March 1998.

2. GENERAL DESCRIPTION

Figure 1 shows a simplified block diagram of the ICS-130 board. The board uses 32 16-bit Sigma-Delta ADCs (Analog Devices AD7723) to provide simultaneous sampling at rates of up to 1.2 Msamples/sec. on each channel. The oversampling ratio of the Sigma-Delta ADC can be selected as 16 or 32.

To allow fast transfer of ADC data, the ICS-130 board includes a VME64 interface capable of in excess of 70 MByte/s (i.e. D64 bus cycles), an optional P2 interface (VSB, RaceWay, Skychannel, etc.), and a 32-bit front panel interface. The transfer rate of the FPDP interface is fully programmable from 1 to 50 MHz. The VME64 interface circuitry uses the Tundra Semiconductor (formerly Newbridge) SCV64 chip to support a Multiplexed Block Transfer (MBLT) master/slave interface. The A32/D32, and A24/D32 protocols are also supported.

ADC data is buffered either in the 4 KSample FIFO (First-In First-Out memory) when data is read-out via the FPDP or the P2 interface, or the dual-ported memories when data is read out via the VMEbus. The sampling clock and the trigger can be either internal or external. The internal ADC clock is user programmable in steps of less than 250Hz at the output rate.

The ADCs can be operated either in continuous or capture modes. In the continuous mode, data is continuously supplied to the selected interface upon application of a trigger signal until the acquisition is disabled. In capture mode, a fixed number of samples are acquired upon each application of the trigger. There are two ways in which this may be done. When using pre-trigger storage, the ICS-130 stores samples continuously before the trigger and acquires a programmable number of samples following the trigger (to a maximum of 32,768 samples/channel if all 32 channels are active). When pre-trigger storage is not used, conversion starts at each application of the trigger and a programmable number of samples (again, to a maximum of 32,768) are acquired.

In capture mode without pre-trigger storage, the acquisition count and buffer length may be programmed separately. Thus it is possible to perform multiple capture sequences at each occurrence of the trigger, until the buffer is filled to the programmed length.

The ICS-130 board can generate VMEbus interrupts at any user programmed interval (number of samples acquired).

All power requirements of the ICS-130 are satisfied with standard VMEbus voltages.

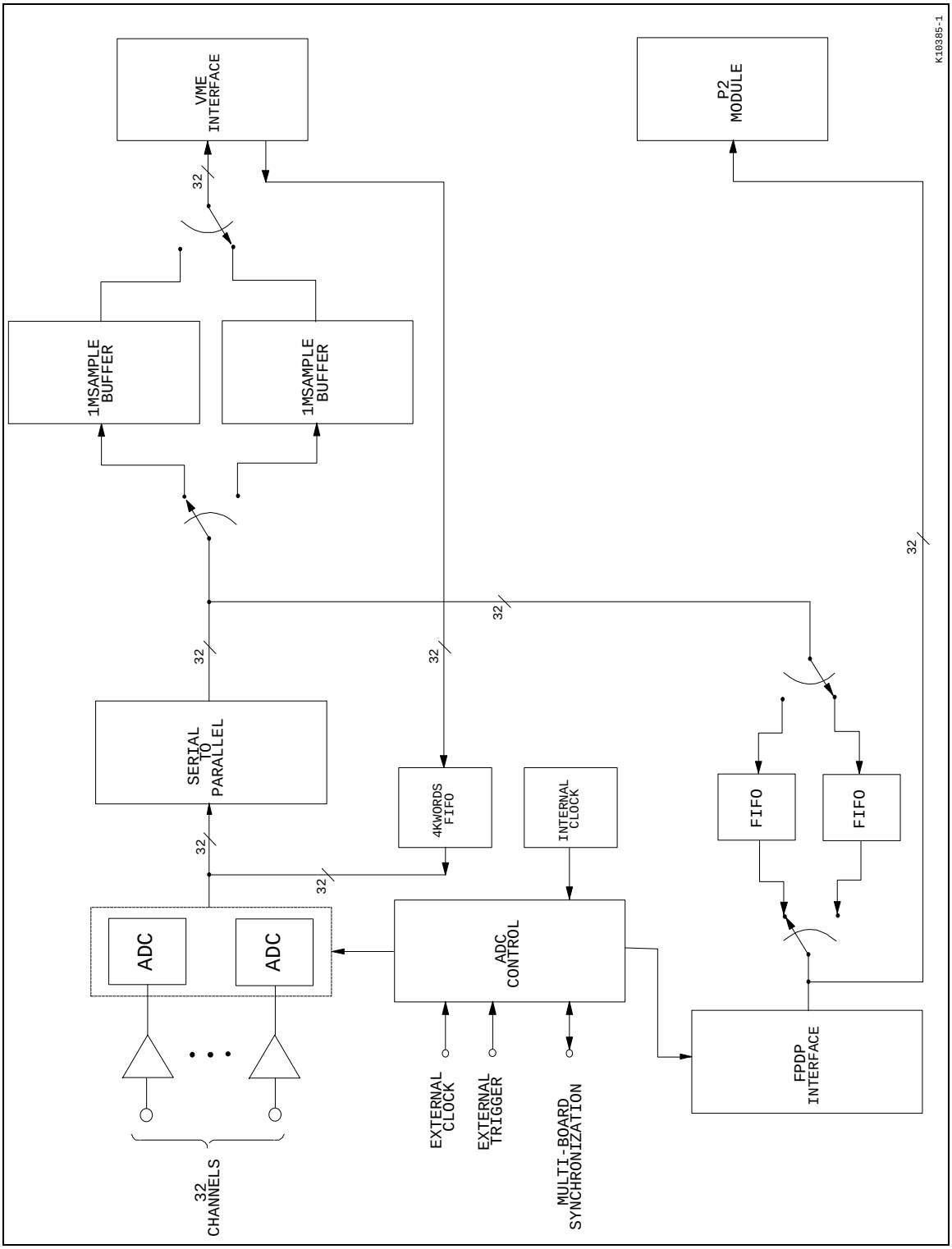


Figure 1 - ICS-130 Simplified Block Diagram

2.1 Board Specifications

Analog Input	
No. of Analog Input Channels:	4, 8, 16 or 32 differential
Input Impedance:	10 kOhm (each wire)
Full Scale Input:	2 Vpp diff. or 4 Vpp single-ended
Max. Input Sample Rate (Fs):	19.2 MHz
Oversampling Ratio:	16 or 32
Max. Input Signal Bandwidth:	0.383 x Fo (Flat passband) 0.478 x Fo (-3dB) where Fo = Channel Output Rate Fo = Fs/16 or Fs/32
Maximum Channel Output Rate:	1.2 MHz/ch. for 32 channels (16x oversampling)
Minimum Channel Output Rate:	31.25 kHz/ch. for 32 channels
Maximum Decimation:	256
Settling Time:	1293/Fs (32x oversampling) 541/Fs (16x oversampling)
Signal/(Noise+Distortion+Crosstalk):	>87dB (Fs/32) >81dB (Fs/16)
Inter-channel Cross-talk:	<-80dB
On-board Storage:	2 MSample (64 KSample/channel, all 32 chan's active)
General	
VMEbus Interface:	A64/A32/A24 D64/D32 MBLT Master/Slave
Front-Panel Interface:	FPDP 32-bit, 160MByte/s
Environmental:	Temp: 0° to +50°C operating (at entry point of forced air, approximately 490 LFM) -40° to +85°C Storage Humidity <95% non-condensing
Power:	2.2 Amps @ +5 V 0.3 Amp @ +12 V (No daughter card modules installed)
Board Size:	VME 6U (233x160mm) Single Slot Width

3. DETAILED DESCRIPTION

3.1 ADC Section

The ICS-130 board uses 32 16-bit Sigma-Delta ADCs (Analog Devices's AD7723). The maximum input clock for the Sigma-Delta ADC is 19.2 MHz. Thus, the maximum output rate (16x oversampling ratio mode) is 1.2 MSamples/sec. When the internal programmable frequency clock is used, the minimum sampling rate for the ADCs is 1.0 MHz. However, the ADC data stream may be decimated by up to 256, making the board's effective minimum output rate 122 Hz. Decimation is accomplished by storing one out of every N samples where N is programmable from 1 to 256. The full scale input signal level is 2 Vpp differential (i.e. 2 Vpp on each wire of the differential pair), with an input impedance of 10kΩ to ground on each input wire. Figure 2 shows the input buffer stage for one channel. Two 44-pin connectors (marked P6 and P7) are provided on the front panel for applying the differential analog input signals. The inputs may also be driven using a single-ended signal; in this case, the full-scale input on the driven wire is 4 Vpp. The unused wire should be connected to ground. If a long cable is used to drive the input, the best arrangement is to use a shielded twisted pair to drive each channel input, with the undriven wire grounded at the transmit end of the cable. This scheme takes advantage of the differential input of the ICS-130 to remove common-mode noise picked up by the cable.

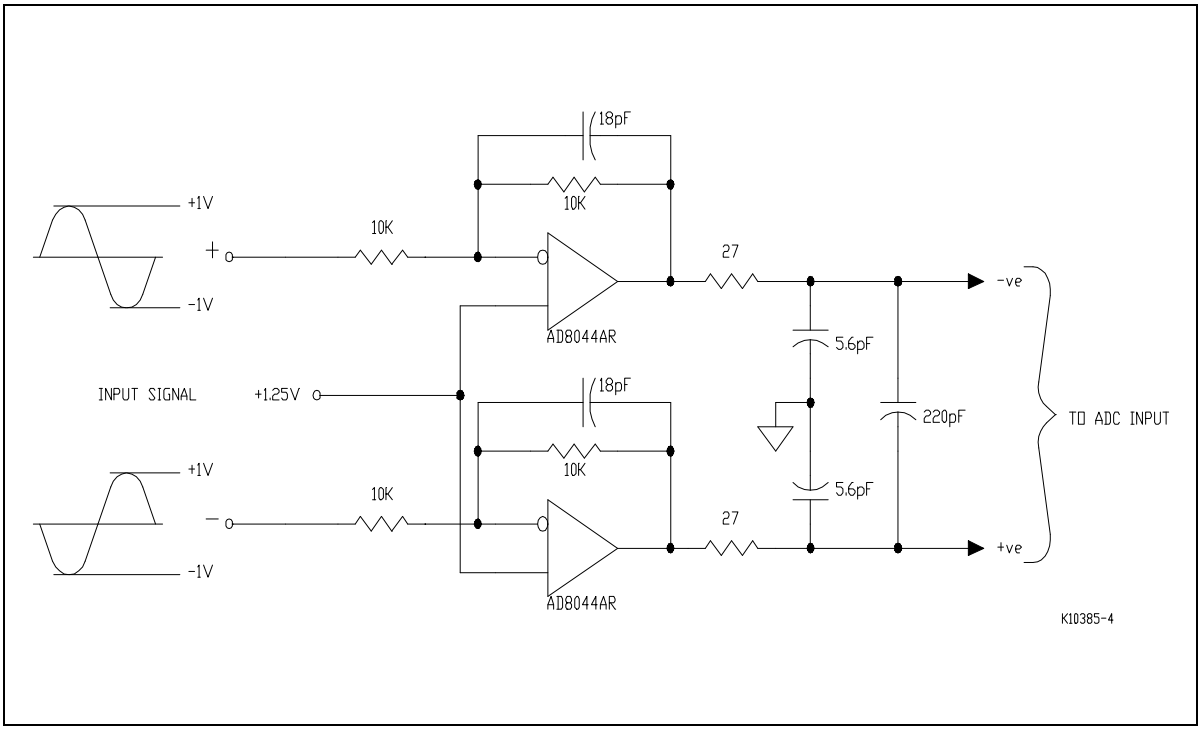


Figure 2 - ICS-130 Analog Input (One Channel Only Shown)

3.2 Input Bandwidth and Sample Rate

The AD7723 Sigma-Delta ADC chip employs a number of digital filters in series in order to achieve a high decimation factor. The composite response of these filters governs the input signal bandwidth. The digital filter response can be either lowpass or bandpass as shown in Figure 3.

For the lowpass filter case, as shown in Figure 3(a), the output rate, Fo, can be either Fs/16 or Fs/32. Note that for the lowpass case (which is the case for most applications) and for a flat pass band, the sample rate can be selected as:

Fs = 32 x BW/0.383 for BW $\leq$ 230 kHz
or Fs = 16 x BW/0.383 for BW $\leq$ 460 kHz

For the bandpass case, as shown in Figure 3(b), the output rate must be Fs/32. The band shown in the figure (0.308Fo to 0.383Fo) is shifted down to DC.

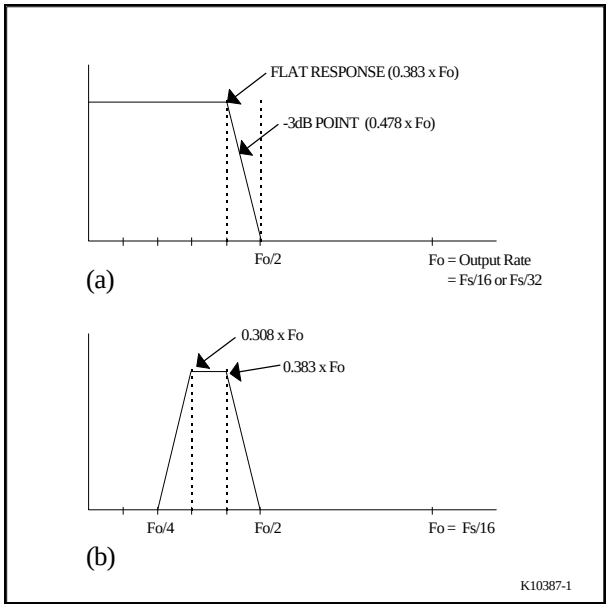


Figure 3 - Lowpass and Bandpass response options

The choice of filter type and decimation (oversampling) ratio may be selected using the ADC Mode field of the ICS-130 Control register (see section 5.7.11). Further details on the converter characteristics are available from the manufacturer, Analog Devices, at the following web site address:
http://products.analog.com/products/info.asp?product=AD7723

The outputs of the ADCs are combined to produce up to sixteen 32-bit words, depending on the number of channels selected to be active, in the Channel Count register (see section 5.8). The selected data are transmitted either to the FPDP/P2 FIFO and/or to the VME swing buffer. The output data format is:

D31	D0	
Channel 1	Channel 2	Word 1
Channel 3	Channel 4	Word 2
• • •	• • •	•
Channel N-1	Channel N	•
Channel 1	Channel 2	•
• • •	• • •	•

3.3 Clock/Trigger Options

The ICS-130 offers a number of clock and trigger options. The card has an internal sampling clock provided by a programmable oscillator giving a resolution of less than 250Hz at the output rate over the entire 1.0 MHz to 19.2 MHz range. Alternatively, an external sampling clock applied at pin 25 of the P4 front panel connector may be used. In this case, the clock must be at the oversampling rate (either 16x or 32x the required output rate).

The trigger can be programmed to be internal (i.e. software controlled) or external. If using the external trigger, the user should supply a positive-going TTL pulse on pin 23 of the P4 connector; the pulse must be at least one sample period long. The trigger is internally synchronized to the sampling clock by the ICS-130. Acquisition starts with the first valid data word after the application of the trigger.

Both external clock and trigger signals must conform to standard TTL levels and drive capability. The relevant connections are listed in section 6.3.

3.4 Data Path Selection

The ICS-130 provides three basic options for reading acquired data. The destination for the ADC data can be VMEbus, FPDP, or P2. One or more of these interfaces may be enabled at a time.

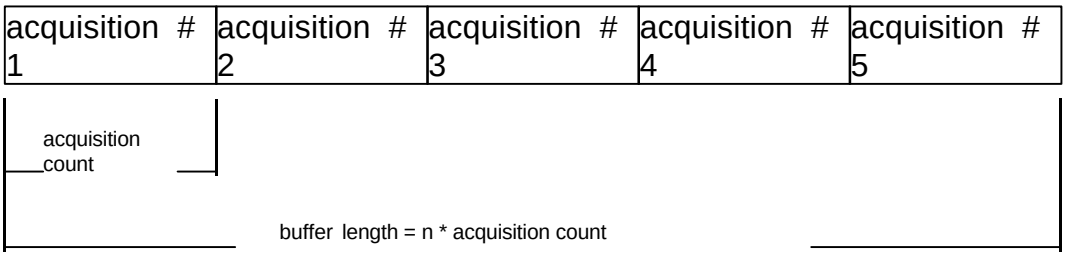
3.5 Modes of Operation

Three modes of operation are provided in the ICS-130 design:

- capture mode without pre storage,
- capture mode with pre storage,
- and continuous mode, which has no pre storage option.

In the capture mode *without* pre storage, data is acquired for a programmable number of samples following the application of each trigger. The maximum number of samples that can be stored for all active channels is 1048576. Because the size of the memory buffer, and the count of samples acquired are both programmable, multiple capture acquisitions may be stored in the ICS-130 memory.

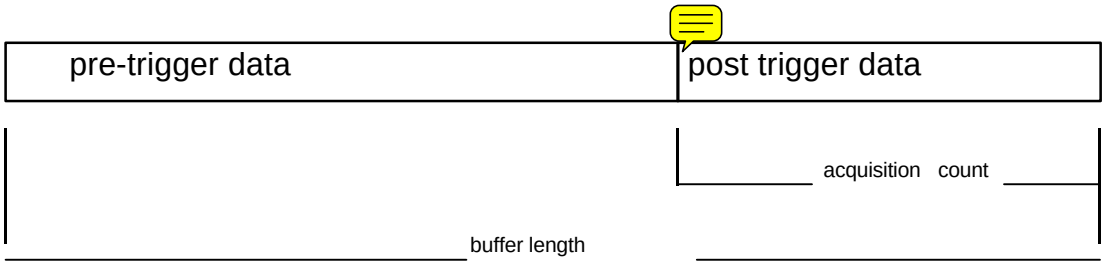
Important note: When using capture mode without pre-trigger storage, the memory buffer length programmed must *always* be an integral number of acquisition counts.



In the capture mode *with* pre storage, the ICS-130 memory is used as a circular buffer of programmable length. The ICS-130 is "armed" by the user, and the control logic continuously fills the circular buffer with fresh data samples in anticipation of the trigger signal. When the trigger signal is recieved, the final "acquisition count" number of samples are stored in memory, and acquisition is automatically terminated.

To select pre-trigger storage of data in Capture mode, the Arm register must be written to as the last action of configuring the board. This will cause the board to start acquiring data. If the Arm register is not written, pre-trigger storage will not occur.

The basic idea: When using pre-trigger storage, acquisition count and memory buffer size may be independantly set. Data stored in the memory buffer will be divided into two sections: data acquired *before* the application of the trigger, and data acquired *after* the application of the trigger.



In the Continous mode of operation, acquisition begins upon application of the trigger and continues until the board is disabled.

3.6 VMEbus Interface

The ICS-130 implements a VMEbus Master/Slave A64/D64 A32/D64 A24/D32 interface using the Tundra Semiconductor SCV64 integrated circuit. Master BLT (Block Transfer) and MBLT (Multiplexed Block Transfer) cycles are also supported. On power-up, default A32 and A24 slave images are loaded by the SCV64 with base addresses determined from on-board switches (see section 4 for details of the hardware configuration). The VMEbus Host can configure the SCV64 internal registers by accessing the SCV64 register block

using either of these slave images. Note that the slave base address of the SCV64 can be reprogrammed by the host by loading appropriate values in the SCV64 VMEbus Base Address Register.

The SCV64 can be configured to perform VMEbus Master Block Transfers by loading the SCV64 internal registers with the appropriate VMEbus Start Address, the local (to the ICS-130) start address, and the transfer count. After the DMAGO bit in the SCV64 Control Register is set, the SCV64 will acquire the VMEbus and perform the requested transfer, freeing the host for other tasks. Note that the VMEbus specification limits block transfers to a maximum of 256 Bytes, however the ICS-130 design does not prevent transfers of larger blocks. Transfers can be in either D64 (MBLT) or D32 (BLT) data path width. D64 transfer rates as high as 70 MB/s can be attained. A programmable VMEbus interrupt is available to indicate that a transfer has been completed. Programmable interrupts are also available for the ADC Data Ready condition.

3.7 Cascading Multiple Boards

The ICS-130 provides simultaneous sampling not only on all channels on one board, but also on all channels across multiple boards. The ICS-130's PLL clock circuitry allows multiple board systems to have simultaneous triggering (+/- 0 samples) and less than 1.5 ns board to board sampling skew.

In order to achieve multiple board synchronization, one board is designated as the "Master" and provides clock and trigger signals to the other (“Slave”) boards in the group. As with single boards, either internal or external clock and trigger signals may be used with multiple board configurations. In the case of external clock and/or trigger, the user supplies the external signal(s) to the master, which in turn distributes the clock and trigger to the slaves. Master/slave status can be programmed in the Control Register (see 5.8). The 20 pin header on the front panel of the ICS-130 provides access to all signals necessary for multi card synchronization. Details of the P4 pinout are given in appendix 6.3. The term Master in this context should not be confused with VMEbus bus mastership.

Important note: All boards to be synchronized must be located in the same VMEbus chassis in order to avoid violation of timing requirements.

3.8 FPDP Interface

A connector on the front panel of the ICS-130, designated P3, is compatible with the ICS FPDP Interface. The **ICS** **F**ront **P**anel **D**ata **P**ort (**FPDP**) is an industry standard interconnection for board to board or system to system data transfer. This interface standard has gained acceptance in the industry for use in a broad range of signal processing applications. It is directly compatible to CSPI's SC130/P100, Mercury's RIN-T/ROUT-T, SKY's SKYburst 160 and Ixthos' IXI2S32-F interfaces.

The FPDP is a high performance 32 bit parallel interface configured with a ribbon cable to connect boards or systems together. The simple and well-defined physical and electrical interface provides the basis for integrating many different types of boards and sub-systems. The maximum clock rate of the FPDP interface is 20 MHz when using the TTL Strobe signal, providing a sustained data rate of up to 80 MBytes/s. When using the optional PECL-level differential Strobe signals, the interface supports clock rates up to 40 MHz for a sustained data rate of up to 160 MB/s. The ICS-130 provides the optional PECL Strobe, and can typically be run up to 50 MHz.

The ICS-130 FPDP interface design allows multiple boards to be connected on the same FPDP cable. Data from all selected channels are inserted into the FPDP Data frame in the correct time slot. It is necessary to connect a cable between the P4 Local Bus connectors of each ICS-130 in order to ensure that all boards are correctly synchronized (see section 3.7 above). One ICS-130 is configured as the Sampling Master for this purpose, by programming Control Register bit CR<6>, and generates the timing signals used by the other (Slave) boards. The same board must also be configured as FPDP Master using Control Register bit CR<8>. The board designated as Master must be at one physical end of the FPDP and Local Bus cables.

3.9 Light-Emitting Diodes

The ICS-130 is fitted with a set of light-emitting diodes (LEDs) which indicate board operation and error conditions. The LEDs are provided for diagnosis of major operational problems and can only be viewed when the board is on an extender card.

The diodes are located on the component side of the board (the right hand side when the board is installed in vertical orientation), between the VME connectors. Their position on the board is shown in Fig. 3. There are four LEDs installed in a line. Table 3.1 describes the ICS-130 LEDs starting from the LED closest to the P1 VME connector (i.e. the uppermost LED when the board is installed in normal vertical orientation).

Note that the intensity of illumination of each LED will depend on the frequency of the access to that interface. It may not be possible to detect infrequent accesses to an interface.

TABLE 3.1 - Light Emitting Diodes

	LED Function	Colour	Description
1.	VME Access	Green	Illuminated at each valid VMEbus access to ICS130
2.	P2 Access	Green	Illuminated at each valid P2 access to ICS-130
3.	FPDP Out	Green	Illuminated at each valid FPDP access to ICS-130
4.	ADC Running	Green	Illuminated for each ADC sample acquired

4. HARDWARE PREPARATION

This section provides information necessary for hardware preparation of both single board and multiple board ICS-130 systems. Figure 3 shows the position of switch and jumper (wire link) blocks on the ICS-130.

Prior to board installation, the user should examine the VMEbus chassis where the ICS-130 board is to be installed. Some VMEbus signals are of the "daisy chain" type, i.e. they are routed through each board in turn using one connector pin for input to a board, and another pin for output; the signals in question are BG0IN/OUT, BG1IN/OUT, BG2IN/OUT, BG3IN/OUT and IACKIN/IACKOUT. For slots where boards will not be installed, one of two mechanisms is normally employed to ensure continuity for these signals. Either the user must install a set of jumpers on the VMEbus backplane, or the connectors are of a type which can sense the presence of a board and make or break the connection accordingly. If the backplane in use is a jumpered type, the user should remove all jumpers for the slot where an ICS-130 will be installed. All unused slots between slot 1 and the slot where the ICS-130 is installed must be jumpered to ensure correct operation.

4.1 System Configuration

The ICS-130 is designed to allow simultaneous sampling across all channels on a board, and also across all channels on multiple boards. To facilitate multiple board clusters, timing and external clock/trigger signals are bussed on the front panel P4 Local Bus cable. In addition, if the Front Panel Data Port (FPDP) is used for data output, multiple ICS-130 boards may be bussed on the same FPDP cable.

Two multiple board cluster configurations are possible:

- All ICS-130 boards on one Local Bus cable and on one FPDP cable
- All ICS-130 boards on one Local Bus cable, but on two or more FPDP cables

The latter case addresses the situation where a single FPDP cable would not have sufficient throughput, or where the FPDP/R- (receiver) board has insufficent throughtput and multiple receiving boards must be used.

For the purposes of correctly driving and terminating bussed front panel signals, there are four possible hardware switch configurations for an ICS-130 board. These are:

- Master - the board drives timing signals and terminates received signals
- Mid-Slave - the board does not terminate or drive signals except data lines
- End-Slave - the board terminates timing signals and drives data lines
- Stand-alone Master - the board drives and terminates all signals

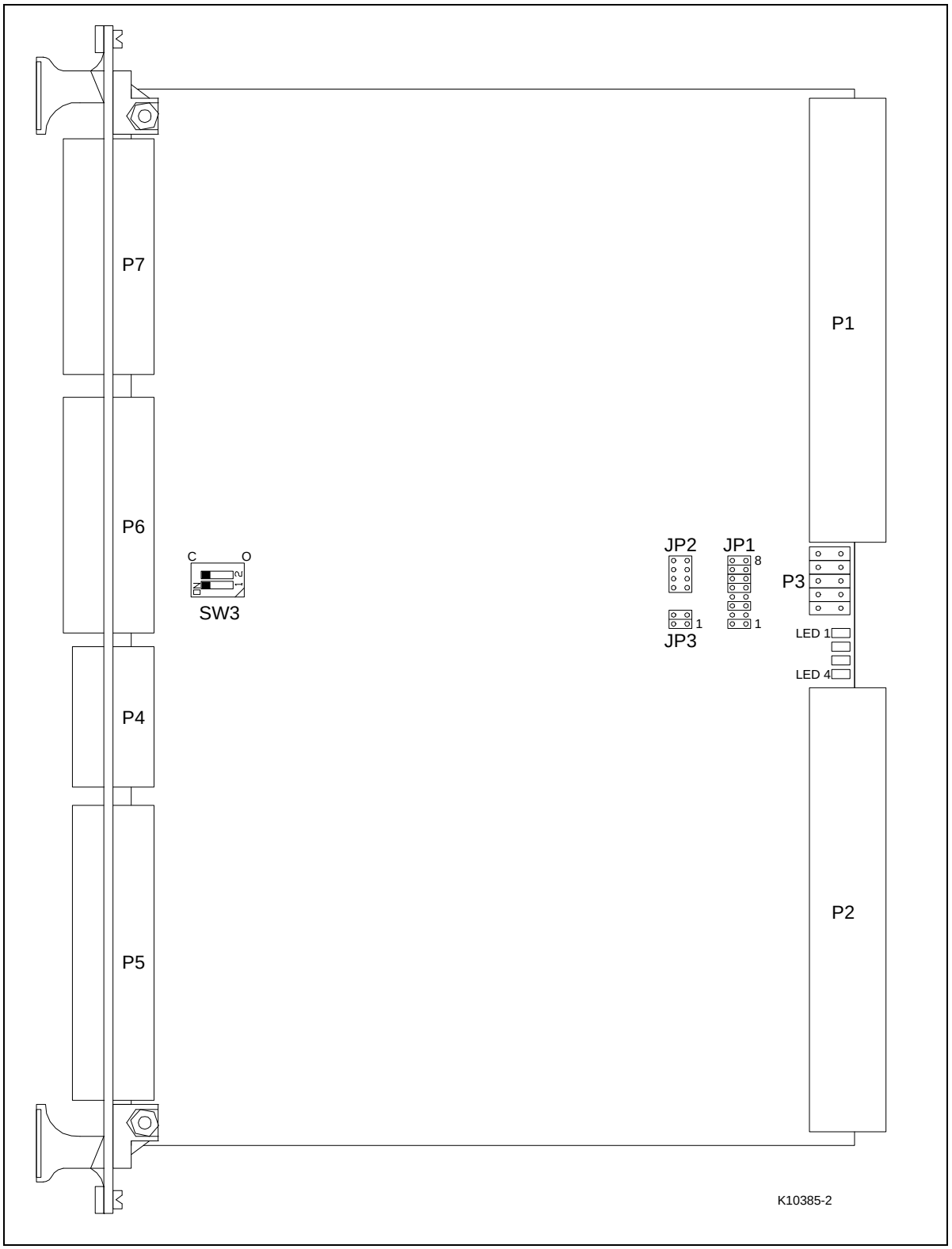


Figure 3 - ICS-130 Switch and Jumper Locations

Note that the term “End Slave” refers to an ICS-130 board, not the FPDP receiving board. These classifications are applicable to both Local Bus and FPDP operation. It is possible therefore, for a board to be a Local Bus Mid-Slave but an FPDP Master, FPDP End-Slave or FPDP Stand-alone Master. The Master board must be located at one end of the cable. The End-Slave is located at the opposite end of the cable to the Master. The Mid-Slaves are located in between the Master and End-Slave boards. In a two board system, only Master and End-Slave configurations apply. Finally, the Stand-Alone Master refers to a one board system. The required register settings are determined by which of these four cases applies to the board in question. Also, switch and register settings are affected by whether a configuration uses internal or external clock (EXT_CLK+/-) and trigger (EXT_ACQ+/-) signals.

If the ICS-130 board/s is/are connected to one or more Digital Signal Processor (DSP) or Array Processor boards using the FPDP, the DSP board/s should be installed in the chassis at the End-Slave end of the FPDP cable (i.e. the ordering should be Master, Mid-slaves, End-Slave, DSP). If FPDP is not being used, the position of the DSP board/s in the chassis with respect to the ICS-130 boards is not important, however it is recommended that the ICS-130 boards are installed in adjacent slots.

4.2 Jumper and Switch Settings

A number of hardware configuration jumpers (wire links) and one switch block need to be configured on the ICS-130. Figure 3 shows the locations of the jumpers and switches on the component side of the board. Switches and jumpers are numbered thus: SWx-y and JPx-y, referring to position y of switch or jumper x, respectively.

4.2.1 VMEbus Base Address Selection

The VMEbus A24 and A32 Base Address is set with jumpers JP1 (8 positions) and JP3 (2 positions). JP2 is an unused jumper block available for storage of spare jumpers. The values of these switches are loaded to SCV64 register VMEBAR at power up. This register may be subsequently reprogrammed by the user (see section 5.2). The ICS-130 address map occupies a 384KB (Hexadecimal 0x60000) space. In A24 mode, the user may specify the values of A23 to A19, allowing the base address to be set to any value between 0x00080000 and 0x00F80000, in increments of 0x80000.

The default A24 base address is 0x500000. This is the setting shown in Fig. 3.

The A24 base address is set as follows:

Jumper	VME A24 Address Bit
JP1-1	A23
JP1-2	A22
JP1-3	A21
JP1-4	A20
JP1-5	A19

In A32 mode, the user may specify the values of A31 to A27, allowing the base address to be set to any value between 0x08000000 and 0xF8000000, in increments of 0x8000000. The A32 base address is set as follows:

Switch	VME A32 Address Bit
JP1-6	A31
JP1-7	A30
JP1-8	A29
JP3-1	A28
JP3-2	A27

Note: An inserted jumper selects a zero in that address bit.

The VMEbus A64 base address is programmed after power up; details of the procedure for doing this are given in section 5.2

4.2.2 P4 Local Bus Interface

Switch block SW3 is used to connect parallel (pull-up/pull-down) resistive terminations to the External Clock and Trigger signals, as shown in the following table. These are required if the user chooses not to use serial terminating resistors at the signal transmitter/s. Other functions of the P4 Local Bus can be programmed by software using the Control Register (see section 5.7). When connecting external clock and/or trigger signals to a multiple board FPDP configuration, it is recommended that the terminations should be enabled on the master board only, and that the signals should be driven from the End Slave end of the cable so that the terminations are at the far end of the cable.

SWITCH	**FUNCTION (When switch in "ON" or Closed Position)**
SW3-1	Terminate External Clock Input
SW3-2	Terminate External Trigger Input

5. PROGRAMMING MODEL

The ICS-130 VMEbus Memory Map is shown in Table 5.1 below. The individual bit fields of the registers are shown in Fig. 4. All programming and control of the ICS-130 is accomplished through the VMEbus interface. All control register bits that are not defined have no effect on the operation of the ICS-130, but will always be read as zero. All other bits are undefined, and may be read as zero or one.

5.1 General Notes

All transfers to and from the ICS-130 control and status registers should be done using D32 VMEbus cycles. The A24 and A32 VMEbus base address is programmed with jumpers (wire links) on the board as described in section 4. The ICS-130 responds to both SUPERVISORY and NON-PRIVILEGED VMEbus cycles. Individual bits in the 32 bit registers are referred to in braces. e.g. ADCCR<0> corresponds to ADC Control Register bit 0, and ADCDEC<4:0> corresponds to ADC Decimation Register bits 4 through 0.

TABLE 5.1 - ICS-130 Memory Map

REGISTER	OFFSET	TYPE
ADC DATA/DIAGNOSTIC FIFO	BASE + 0x00000	READ ONLY/WRITE ONLY
SCV64	BASE + 0x40000	READ/WRITE
P2 MODULE	BASE + 0x48000	READ/WRITE
STATUS	BASE + 0x50000	READ ONLY
INTERRUPT MASK	BASE + 0x50004	READ/WRITE
CONTROL REGISTER	BASE + 0x50008	READ/WRITE
CHANNEL COUNT	BASE + 0x5000C	READ/WRITE
BUFFER LENGTH	BASE + 0x50010	READ/WRITE
ACQUISITION COUNT	BASE + 0x50014	READ/WRITE
DECIMATION	BASE + 0x50018	READ/WRITE
FRAME COUNT	BASE + 0x5001C	READ/WRITE
ADC CLOCK	BASE + 0x50020	WRITE ONLY
FPDP CLOCK	BASE + 0x50024	WRITE ONLY
ARM	BASE + 0x50028	WRITE ONLY
ADC RESET	BASE + 0x50030	WRITE ONLY
BOARD RESET	BASE + 0x50034	WRITE ONLY
MASTER CONTROL	BASE + 0x58000	READ/WRITE

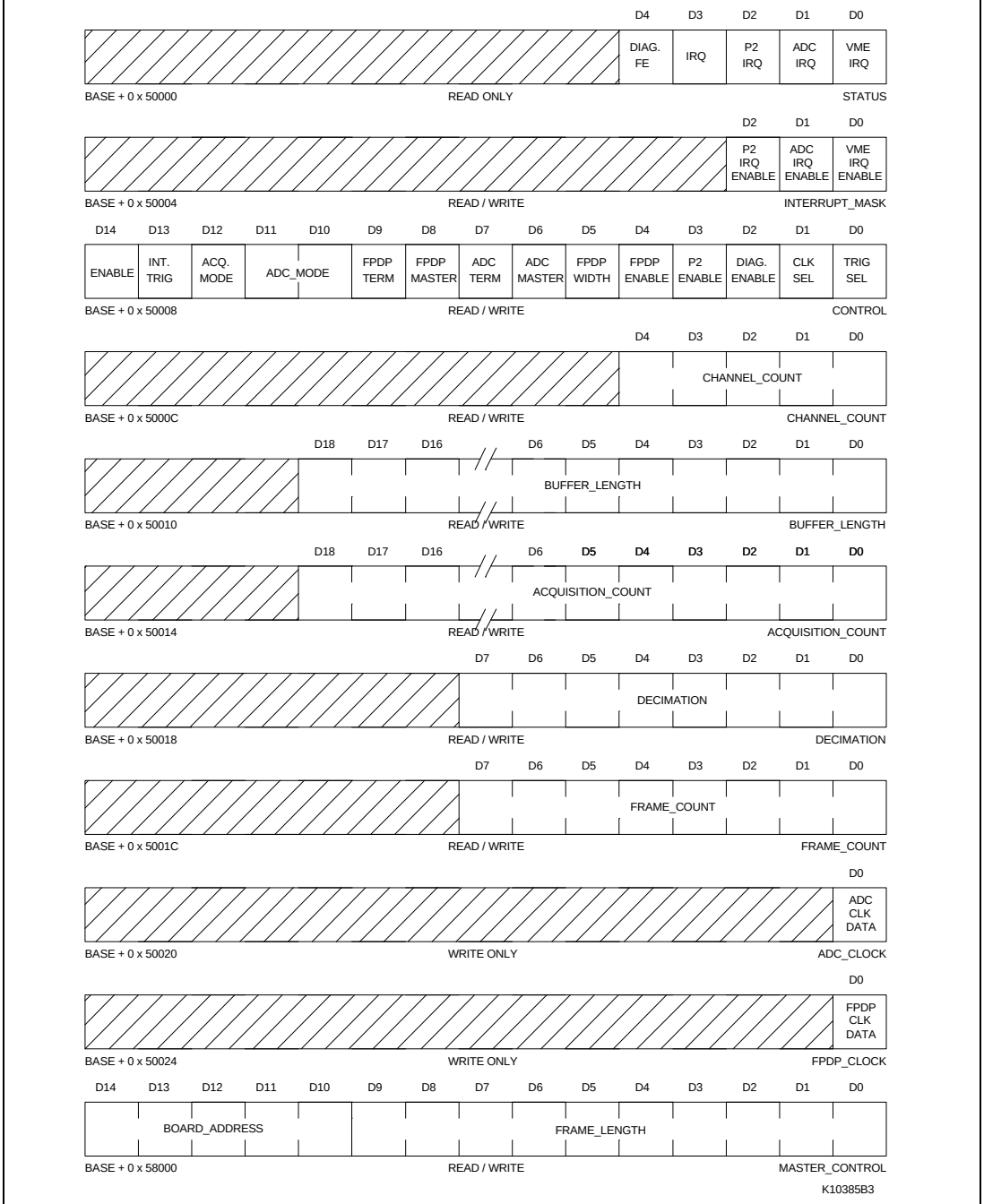


Figure 4 - ICS-130 Register Map

5.2 SCV64 Registers

The ICS-130 uses the Tundra Semiconductor SCV64 VMEbus interface chip to handle all VMEbus communications. Full details of the SCV64 may be found in the SCV64 User Manual (See Ref. 1). Table 5.2 gives descriptions for the SCV64 registers which may be needed when programming the ICS-130. Unless the Master BLT/MBLT or the A64 capability of the SCV64 is to be utilized, the power-up defaults of the register contents are sufficient, with the following exception: the MODE register should be programmed to the hexadecimal value 0x9480e401. This value is chosen to optimize speed of transfer between the ICS-130 and the VMEbus. Descriptions of some of the bits of the MODE register are given in Table 5.2. For a more detailed description, see Ref. 1.

If the user wishes to employ VMEbus 64-bit address cycles (A64 addressing mode) when addressing the ICS-130, it is necessary to program the most significant 32 bits of the ICS-130 base address (i.e. bits A<63:32>) to SCV64 register SA64BAR. The procedure for doing this is as follows:

i) Set MODE<12> to '1' (coupled mode).
ii) Write most significant 32 bits of VMEbus address to SA64BAR.
iii) Clear MODE<12>.

The least significant 32 bits of the 64-bit base address are taken from the A32 base address values configured for the board as described in section 4. These jumper settings are loaded to the SCV64 VMEBAR register at power up, but can be subsequently reprogrammed by the user.

The register map of the SCV64 is complex, as it contains functionality not useful to the operation of the ICS-130. Only the relevant register assignments are discussed here. For this reason, accesses to the SCV64 memory space other than to documented offsets may have unpredictable consequences and should be avoided. The SCV64 has two sets of address and data busses, one connected to the VMEbus and the other to the ICS-130 local bus. VMEbus accesses are mapped from VMEbus space to local bus according to the programming of the SCV64. When the SCV64 is used to perform VMEbus Master transfers, it simultaneously becomes the local bus Master, and the VMEbus Master. When VMEbus Slave cycles occur, the SCV64 is the VMEbus Slave, but the local bus Master. When performing VMEbus Master transfers, the SCV64 can be configured to use normal cycles (VMEbus address broadcast between each cycle), BLT (D32 block transfer), or MBLT (D64 multiplexed block transfer). All register offsets are given with respect to the beginning of the SCV64 register window (BASE + 0x40000).

When the SCV64 is programmed to operate as the VMEbus Master, some caution must be taken when considering transfer counts. This is discussed in section 5.3.

TABLE 5.2 - SCV64 Register Descriptions

Register	Offset	Description
DMALAR: DMA Local Address	0x0	This register contains the local bus address used by the SCV64 when performing a VMEbus Master transfer. This register must be re-programmed to zero before each Master transfer cycle is initiated.
DMAVAR: DMA VMEbus Address	0x4	This register contains the VMEbus address accessed by the SCV64 when it is performing a VMEbus Master transfer cycle. If not re-programmed between cycles, it continues from the next consecutive address.
DMATC: DMA Transfer Count	0x08	This register contains the DMA transfer count (Longwords) used by the SCV64 when it is performing a VMEbus Master transfer cycle. Register width is 20 bits (DMATC<19:00>).
DCSR: Control and Status Register	0x0C	The following describes the bits of the SCV64 Control and Status Register used by the ICS-130. All other bits should be set to 0 during a write. The register should always be cleared before starting to set up a DMA transfer.
		DCSR<16> When asserted, this bit indicates an SCV64 configuration error
		DCSR<12> A64 base address ready. Must be programmed to 1 after the SA64BAR and MA64BAR registers have been programmed.
		DCSR<5> If asserted, a bus error has occurred during a DMA transfer. Write a '0' to clear the error.
		DCSR<3:2> If either of these bits are asserted, the previous DMA cycle failed. This can result from transferring too much data (See SCV Registers) or BERR was asserted on the VMEbus during the transfer. Writing '0' to these bits clears them.
		DCSR<1> When asserted, the previous DMA cycle was successfully completed. Writing '0' to this bit clears it.

		DCSR<0> Writing a '1' to this bit starts the DMA cycle. Reading a '1' indicates that a DMA cycle is in progress. A DMA cycle may be aborted while in progress by clearing this bit. In this case, DCSR<1> will be set, and a transfer complete interrupt will be requested.
VMEBAR: VMEbus Slave Base Address Register	0x10	This register is used to set the VME A24 and A32 Slave Base Address values. The A32 SBA is also used as the LS 32 bits of the A64 Slave Base Address. On power up, the A24 and A32 values are loaded from the on-board switches (see section 4).
		VMEBAR<22:21> Size of A24 slave image: 0 - 512K 1 - 1M 2 - 2M 3 - 4M
		VMEBAR<20:16> Base address of A24 slave image. These bits form bits A23-A19 of base address. Address bits A17 and A16 are forced to zero according to setting of VMEBAR<22:21> described above.
		VMEBAR<08:05> A32 slave image size. Selectable in powers of two from 4K to 128M. Program 0 for 4K, 0xF for 4M.
		VMEBAR<04:00> A32 base address. Selects base address in increments of 0x0800.0000 from 0x0000.0000 to 0xF800.0000. Program 0 for A32 address 0x0000.0000.
IVECT: VMEbus Interrupter Vector Register	0x24	This register contains the interrupt vector the SCV64 responds with during an interrupt acknowledge cycle.
MODE: Mode Control Register	0x3C	The following describes the bits of the SCV64 Mode Control Register which may need to be accessed when using the ICS-130. All other bits should be set to '0' during a write.

		MODE<31>	This bit controls the maximum transfer size of the SCV64. When set to '0', the maximum programmable transfer size is 4K (12 bit transfer count). When set to '1', the maximum is 2M (20 bit transfer count).
		MODE<28>	This bit should always be written as '1'.
		MODE<26>	This bit should always be written as '1'.
		MODE<23>	When set to '0', the SCV64 uses non-privileged AM codes when doing Master transfers. When set to '1' it uses supervisory AM codes. Must be cleared if MODE<20> is set to '1'.
		MODE<20>	When set to '1' SCV64 Master transfer cycles are performed using A64 addressing on the VMEbus. Otherwise A32 or A24 cycles are used, as determined by MODE<9>.
		MODE<19>	When set to '1', SCV64 Master transfer cycles are performed using MBLT (D64) cycles on the VMEbus. Must be cleared if MODE<9> is set to '1' or if MODE<18> is set to '1'.
		MODE<18>	When set to '1' SCV64 Master transfer cycles are performed using BLT cycles on the VMEbus. Must be cleared if MODE<19> is set to '1'.
		MODE<16>	When set to '0' SCV64 Master transfer cycles are performed from ADC memory to the VMEbus, and from the VMEbus to DAC memory when '1'.
		MODE<15:13>	These bits should always be written as all '1's.
		MODE<12>	When set to '1', the receive FIFOs couple VME and local buses. When '0', busses are uncoupled. Should normally be written as '0', except when accessing SA64BAR and MA64BAR registers (see decriptions below).
		MODE<10>	When set to '0' the VMEbus Slave image of the ICS-130 is disabled

			(will not respond), and enabled when set to '1'.
		MODE<9>	When set to '0', the SCV64 Master operates in A32 or A64 mode (as determined by MODE<20>), and in A24 mode when set to '1'.
		MODE<0>	If set to '0', all Slave images of the SCV64 are disabled. IN THIS EVENT, THE ICS-130 CANNOT BE ACCESSED UNTIL A VMEBUS RESET OCCURS.
SA64BAR: Slave A64 Base Address Register	0x40	This register sets bits 63-32 of the VMEbus A64 slave base address. The remaining bits (31-0) are taken from the A32 image. Before accessing this register, MODE<12> must be set to '1'. After accessing this register, MODE<12> should be cleared.	
MA64BAR Master A64 Base Address Register	0x44	This register sets bits 63-32 of the VMEbus A64 master base address. The remaining bits (31-0) are taken from the A32 image. Before accessing this register, MODE<12> must be set to '1'. After accessing this register, MODE<12> should be cleared.	
VINT: VMEbus Interrupter Request Register	0x8C	The least significant three bits (D2:0) set the interrupt level of generated by the SCV64. Programming D3 high enables the interrupt. This bit is reset when the interrupt is acknowledged. Caution: Changing the interrupt level while D3 is in the enable state may cause improper operation.	
VREQ: VMEbus Requester Register	0x90	The following describes the contents of VREQ:	
		VREQ<7>	VMEbus Ownership Timer Enable. Condition 1 after reset. 0 = Disable Timer, 1 = Enable Timer.
		VREQ<6>	Bus Clear Recognition Control. Condition 0 after reset. 0 = Ignore BCLR* signal. 1 = Release bus if BCLR* asserted.
		VREQ<5>	VMEbus Release Mode Control. Condition 1 after reset. 0 = Release on Request (ROR), 1 = Release when Done (RWD).

		VREQ<4>	VMEbus Request Mode Control. Condition 1 after reset. 0 = Fair, 1 = Demand.
		VREQ<3:2>	VMEbus Ownership Timer (Time-out Period). Condition 3 after reset. 0=Zero, 1=2microsec, 2=4microsec, 3=8microsec.
		VREQ<1:0>	VMEbus Request Level. Condition 3 after reset. 0 = Level Zero.

5.3 Performing Block Transfers

When performing block transfers, it is important that the transfer count is correctly programmed. If not, memory buffer overflow may occur, or the ADC data window size may be exceeded. The size of the transfer should be the size of the data block currently available or the size of the ADC data window, whichever is smaller. For VME Master transfers, the count is specified in (32 bit) longwords. Note that the VMEbus specification limits block transfers to a maximum of 256 Bytes, however, the ICS-130 design does not prevent transfers of larger blocks as described above.

5.3.1 VMEbus DMA Master

The SCV64 device controls all VMEbus Block transfers. These are also known as Direct Memory Access (DMA) transfers. With A24 or A32 addressing, standard VMEbus Block transfer (BLT) cycles may be used. With A32 or A64 addressing, Multiplexed Block transfer (MBLT) cycles may be used.

The order of programming for Master BLT and MBLT transfers is as follows. All register references are to SCV64 registers. For register details, please refer to Table 5.2 above.

1. Clear the DCSR register.
2. Set up the addressing and data transfer modes by writing to the MODE register. The default to be used for setting this register should be 0x9480e401. The following bits of the register may need to be changed - please refer to the descriptions in Table 5.2:
   - MODE<23>
   - MODE<20>
   - MODE<19>
   - MODE<18>
   - MODE<17>
   - MODE<16>
   - MODE<15:13>
   - MODE<12>
   - MODE<9>
3. Clear the DMALAR (local bus address) register.
4. Write the VMEbus address (least significant 32 bits for A64 address) to DMAVAR. This is the destination address for a write operation from the ADC.
5. For A64 address cycles only, write the most significant 32 bits of the VMEbus (source or destination) address to MA64BAR, using the following procedure:
   i) Set MODE<12> to '1' (coupled mode).
   ii) Write most significant 32 bits of VMEbus address to MA64BAR.
   iii) Clear MODE<12>.
6. Write the transfer count (number of 32 bit longwords) to DMATC.

7. Set DCSR<0>. This is the 'DMAGO' bit. The transfer will start at this point.
8. At the completion of the transfer DMAGO will be read as clear. The DONE bit, DCSR<1>, should be read as set, and a VME interrupt will be asserted. The user should clear DCSR<1> in order to clear the interrupt. If a completion interrupt occurs but the DONE bit is not read as set, an error has occurred. The type of error is indicated by bits DCSR<3:2> and DCSR<5>.

5.3.2 VMEbus DMA Slave

When performing a VMEbus BLT or MBLT block transfer with the ICS-130 as the slave device, no set-up activity is necessary unless the programmer wishes to use A64 address cycles. The set-up procedure for doing this is given in section 5.3 above.

The maximum possible block transfer is 256KB. When initiating a transfer, the programmer should ensure that there are sufficient samples available in the buffer, and that the buffer is of sufficient size for the size of transfer contemplated.

5.4 ADC Data

This section of the ICS-130's VMEbus memory map is used to read data from the ADC memory over the VMEbus. The VMEbus ADC Data area is 0x40000 bytes (256KB) in size. The ADC memory buffers are strictly read only. The same area of the memory map allows test pattern data to be written to the diagnostic FIFO, which is strictly write only. See section 5.16 for details on using built in diagnostics.

The data is organized as one sample in each 16-bit word, with "big endian" ordering. i.e. odd channels occupy the most significant 16 bits and even channels occupy the least significant 16 bits of a 32 bit longword. Channels are numbered from 1 to 32.

The data area appears to the user as FIFO type memory. In other words, random access to samples in the memory is not available. Data access is always sequential regardless of the address used for read or write, as long as the address used falls within the ADC Data area.

Each time a data word is read from the buffer, the data is removed from memory and buffer pointers are modified. The user must be careful to read the exact number of 32-bit words corresponding to the programmed buffer length, otherwise buffer overflow or underflow will occur.

The user should perform a reset of the memory (ADC Reset register) after programming the ADC configuration and before enabling acquisition. This is necessary to ensure that the buffer pointers are correctly aligned prior to buffer access by the ADC circuits.

Addresses presented by the bus master when reading data may be either incremental or repetitive (i.e. always the same address). Provided that addresses fall within the ADC Data

area, consecutive data will be read from the board regardless of the addressing mode. This allows for standard addressing and block transfers with devices which either increment addresses or repetitively present the same address.

5.5 Status Register (SR)

Read only

The Status Register contains information about the state of the ICS-130, including the status of events which may cause VMEbus interrupts, if the appropriate bits of the Interrupt Mask register (IMR) are set. The status register should be used to determine the interrupt status of the ICS-130, and the source of the interrupt. If an interrupt is masked in the IMR, the status of the associated event may still be read here.

5.5.1 SR<0> - VMEbus Master IRQ

This bit reflects the interrupt status of the SCV64 VMEbus Master interface. An interrupt is asserted following completion of a DMA transfer (BLT or MBLT) for which the ICS-130 was master.

SR<0>	VMEbus Master Interrupt Request Status
READ ONLY	
0	SCV64 is not asserting an IRQ
1	SCV64 is asserting an IRQ

5.5.2 SR<1> - ADC IRQ

This bit reflects the status of the ADC control unit. When the Swing Buffer swaps, the ADC control unit will assert this flag, to indicate that new data is available. If ADC interrupts are enabled, an interrupt request will occur.

SR<1>	ADC Interrupt Request
READ ONLY	
0	ADC control unit is not asserting IRQ
1	ADC control unit is asserting IRQ

5.5.3 SR<2> - P2 IRQ

This bit reflects the status of the ADC control unit. When the Swing Buffer swaps, the ADC control unit will assert this flag, to indicate that new data is available. If ADC interrupts are enabled, an interrupt request will occur.

SR<2>	P2 Module Interrupt Request
READ ONLY	
0	P2 Module is not asserting IRQ
1	P2 Module is asserting IRQ

5.5.4 SR<3> - IRQ

This bit indicates that the ICS-130 is asserting a VMEbus interrupt from either source (i.e. Master transfer completion or ADC Swing Buffer swap). It is therefore an OR of the state of bits 0 and 1 of this register.

SR<3>	VMEbus Interrupt Request
READ ONLY	
0	VMEbus interrupt is not asserted
1	VMEbus interrupt is asserted

5.5.5 SR<4> - Diag FIFO empty

This bit indicates that the Diagnostics FIFO is empty and is ready for more data.

SR<4>	Diagnostic FIFO Empty
READ ONLY	
0	FIFO is not empty
1	FIFO is empty

5.6 Interrupt Mask Register (IMR)

Read/Write

The Interrupt Mask register is used to enable and disable VMEbus interrupts. The status of masked interrupts may be read in the corresponding bits of the status register, but no interrupt will occur even when the interrupt is asserted.

5.6.1 IMR<0> - VMEbus IRQ Mask

This bit enables the SCV64 VMEbus Master controller to generate VMEbus interrupts upon completion of a transfer.

IMR<0>	VMEbus Master Interrupt Enable Mask
READ/WRITE	
0	SCV64 Interrupts are disabled
1	SCV64 Interrupts are enabled

5.6.2 IMR<1> - ADC IRQ Mask

This bit enables the ADC control unit controller to generate VMEbus interrupts.

IMR<1>	ADC Interrupt Enable Mask
READ/WRITE	
0	ADC Interrupts are disabled
1	ADC Interrupts are enabled

5.6.3 IMR<2> - P2 Module IRQ Mask

This bit enables the P2 Module to generate VMEbus interrupts.

IMR<2>	P2 Module Interrupt Enable Mask
READ/WRITE	
0	P2 Module Interrupts are disabled
1	P2 Module Interrupts are enabled

5.7 Control Register (CR)

Read/Write

The Control Register controls the overall configuration of the ICS-130.

5.7.1 CR<0> - Trigger Select

This bit determines if the external ADC Trigger is used to start an acquisition. When using an external trigger, triggering occurs following the rising edge of the trigger signal. The signal must remain high for at least one complete acquisition (clock) cycle.

CR<0>	ADC Trigger Select
READ/WRITE	
0	Use Internal (software) Trigger
1	Use External Trigger input

5.7.2 CR<1> - Sampling Clock Select

This bit selects between internal and external sampling clocks for the ADC.

CR<1>	ADC Sampling Clock Select
READ/WRITE	
0	Internal sample clock selected.
1	External sample clock selected.

5.7.3 CR<2> - Diagnostic Mode Enable

This bit is used to enable the Diagnostic mode. In this mode, VMEbus Data can be written to the swing buffer. Once cleared, the data can be read back from the swing buffer.

CR<2>	Diagnostics enable
READ/WRITE	
0	Diagnostic Mode disabled
1	Diagnostic Mode enabled

5.7.4 CR<3> - P2 Enable

This bit enables data output to the optional P2 Daughterboard interface.

CR<3>	P2 Enable
READ/WRITE	
0	P2 Interface Disabled
1	P2 Interface Enabled

5.7.5 CR<4> - FPDP Enable

This bit enables data output to the FPDP interface.

CR<4>	FPDP Enable
READ/WRITE	
0	FPDP Disabled
1	FPDP Enabled

5.7.6 CR<5> - FPDP Word Width

This bit selects the width of the data word used in FPDP transfers. If the unpacked option is selected, one 16-bit sample is sent in each cycle of the FPDP Strobe (clock) signal; the sample is aligned to the most significant 16 bits of the 32-bit FPDP data path width (bits D<31:16>). The natural channel order is maintained (i.e. Channel 1, Channel 2, Channel 3, etc.). If the packed option is selected, Data is presented on the FPDP in the same format as in the internal 32-bit memory, i.e. two samples per FPDP Strobe cycle, with the lower-numbered channel in the most significant 16 bits of the data path.

CR<5>	FPDP Word Width
READ/WRITE	
0	Packed format (two samples per Strobe cycle)
1	Unpacked format (one sample per Strobe cycle)

5.7.7 CR<6> ADC Sampling Master Enable

This bit selects the board as a Sampling Master or Slave, when operating in a multiple ICS-130 board acquisition cluster. In the cluster, only one board can be set as Master. The Master board sends to all slaves the ADC_CLK (pin 1 and 2 on the P4 connector) and the Trigger signal (pin 10 on the P4 connector). The ADC_CLK signal is in PECL format. Slaves receive the ADC_CLK on the P4 connector pin 1 and 2 and the Trigger signal on pin 10. The Master must be located at one physical end of the P4 Local Bus cable. When using a single ICS-130, the board should also be configured as Sampling Master.

CR<6>	Mode
READ/WRITE	
0	ICS-130 is Slave.
1	ICS-130 is Master.

5.7.8 CR<7> - ADC Clock Termination

This bit connects the resistive termination to the P4 ADC_CLK line. This must be enabled on the End Slave board only, and only in multiple board configurations.

CR<7>	ADC Clock Termination
READ/WRITE	
0	Termination disconnected
1	Termination connected

5.7.9 CR<8> - FPDP Master Enable

This bit selects between FPDP Master and FPDP Slave mode, in a multiple board acquisition cluster when using FPDP output. In the cluster or part cluster connected on a single FPDP cable, only one board can be set as FPDP Master. The Master board sends to all slaves the FPDP_CLK (pin 12 and 13 on the P4 connector) and the Data Valid (DVALID*) signal (pin 15 on the P4 connector). The FPDP_CLK signal is in PECL format. Slaves receive the FPDP_CLK on the P4 connector pin 12 and 13 and the Data Valid signal on pin 15.

CR<8>	FPDP Master Enabled
READ/WRITE	
0	ICS-130 is FPDP Slave
1	ICS-130 is FPDP Master

5.7.10 CR<9> - FPDP Termination

This bit enables the resistive termination on the FPDP NRDY* and SUSPEND* signals. These must be connected when using the FPDP interface, on the Master board only. This applies to single board and multiple board configurations.

CR<9>	FPDP Termination
READ/WRITE	
0	FPDP terminations disconnected
1	FPDP terminations connected

5.7.11 CR<11:10> - ADC Mode

These bits select the operating mode of the Sigma-Delta converters. All converters on the board are programmed to the same mode.

CR<11:10>	ADC Mode
READ/WRITE	
0 0	32x oversampling, Band Pass Filter(FCK_IN/63...FCK_IN/33.4 at -3dB)
0 1	32x oversampling, Low Pass Filter (0...FCK_IN/67 at -3dB)
1 1	16x oversampling, Low Pass Filter (0...FCK_IN/33.5 at -3dB)

5.7.12 CR<12> - Acquisition Mode

This bit selects between the Continuous and Capture modes of acquisition. The user should refer to section 3.4 for a detailed description of the operation and capabilities of the board in these modes.

CR<12>	Acquisition Mode
READ/WRITE	
0	Continuous
1	Capture

5.7.13 CR<13> - Trigger

This bit is the internal trigger signal. The “enable” bit CR<14> must have been previously set, and CR<0> must be set to internal trigger. This bit is automatically cleared after acquisition is started.

CR<13>	Trigger
READ/WRITE	
0	** Automatically cleared **
1	Begin acquisition

5.7.14 CR<14> - Enable

This bit is used to enable acquisition. If external triggering is selected, the acquisition begins following the next rising edge of the external trigger input, otherwise it starts with the first valid data after the trigger bit, CR<13>, has been set.

CR<14>	Enable
READ/WRITE	
0	ADC acquisition disabled
1	ADC acquisition enabled

5.8 Channel Count Register

Read/Write

This register determines the number of active channels on the board. Only the samples coming from the active channels are stored in the swing buffer and moved to the selected output interface. A value (N-1) written in this register will activate channels 1 to N. N must be an even number. For FPDP transfer, N must be at least 4. The valid range of numbers that may be programmed to this register is 1 to 31 (3 to 31 for FPDP output).

5.9 Buffer Length Register

Read/Write

The Buffer Length register is used to determine the number of samples stored in the ICS-130 Swing Buffer. The register must be programmed with the number of 32-bit words (i.e. pairs of samples) to be acquired in the buffer prior to each interrupt or to completion. In continuous mode, the 19-bit buffer length value determines the number of 32-bit words written to the swing buffer by the ADCs before the buffer banks are swapped. In Capture mode, the value programmed determines the total number of sample pairs acquired for all active channels before acquisition terminates. When operating in Capture mode without pre-trigger storage, the buffer length selected must be an integral multiple of the Acquisition Count. The value must be written to this register as one less than the required buffer length. If the ADC interrupt is enabled, a VMEbus interrupt will occur when the programmed length is reached. The valid range of numbers that may be programmed to this register is 0 to 524287. The maximum value is equivalent to the maximum swing buffer capacity of 1Msample (each side).

The user must perform an ADC Reset (by writing to the ADC Reset register) after loading the Buffer Length, Acquisition Count, Decimation and Frame Count registers in order for the new values to take effect.

5.10 Acquisition Count Register

Read/Write

The Acquisition Count register determines the number of acquisition frames (a frame contains one sample for each channel currently active) stored in the swing buffer following each application of a trigger, when operating in Capture mode with no pre-trigger storage. When operating in Capture mode with pre-trigger storage, it determines the number of frames acquired after the (single) occurrence of the trigger. The valid range of numbers that may be programmed to this register is 1 to 524287.

The user must perform an ADC Reset (by writing to the ADC Reset register) after loading the Buffer Length, Acquisition Count, Decimation and Frame Count registers in order for the new values to take effect.

5.11 Decimation Count Register

Read/Write

This register programs the eight bit decimation count. The real output rate is the ADC output rate divided by the decimation factor. The decimation factor is one greater than the value written to this register. Therefore, setting this register to zero disables decimation, while writing one to this register will result in decimation by a factor of 2. This feature enables the output rate to be reduced, allowing operation of the ICS-130 with an effective sample output rate below 31.25 kHz. However, the user is cautioned that decimating the output in this way counteracts the advantages of the Sigma-Delta technology of the converters. The valid range of numbers that may be programmed to this register is 0 to 255, corresponding to decimation factors of 1 to 256.

The user must perform an ADC Reset (by writing to the ADC Reset register) after loading the Buffer Length, Acquisition Count, Decimation and Frame Count registers in order for the new values to take effect.

5.12 Frame Count Register

Read/Write

Data for output to the FPDP and P2 interfaces is stored in two FIFO buffers which are arranged in a swing buffer mode of operation (see Fig. 1). One buffer receives the acquired data while the other is the source for the data transmitted on the FPDP and/or P2 interface. When the acquisition path completes the writing of N acquisition frames (a frame consists of a sample for each active channel), the two buffers are automatically swapped. In general, the user should use a frame count of one. The valid range of numbers that may be programmed to this register is 1 to 255.

The user must perform an ADC Reset (by writing to the ADC Reset register) after loading the Buffer Length, Acquisition Count, Decimation and Frame Count registers in order for the new values to take effect.

5.13 ADC Clock Frequency Register

Write Only

This register allows the user to program the internal sampling (conversion) frequency. This is done by writing a 22-bit programming word to the on-board programmable oscillator. Data is written serially to the oscillator, least significant bit first. One write to the register must occur for each bit of data to be written to the oscillator, with the data to be

programmed in bit 0. Note that the frequency programmed must be the oversampling frequency rather than the output rate. See section 6.2 for details of programming the oscillator.

5.14 FPDP Clock Frequency Register

Write Only

This register allows the user to program the FPDP Strobe (clock) frequency, when the FPDP is used for output. This is done by writing a 22-bit programming word to the on-board programmable oscillator. Data is written serially to the oscillator, least significant bit first. One write must occur for each bit of data to be written to the oscillator, with the data to be programmed in bit 0. See section 6.2 for details of programming the oscillator Clock Generator. The theoretical minimum FPDP Strobe frequency required is the sample output rate multiplied by the number of channels selected; this value can be divided by two is packed output mode is used (see section 5.7.6). The actual frequency used must be at least 15% higher than the calculated minimum. Under certain conditions (e.g. small frame sizes), it may be necessary to further increase the Strobe frequency used.

5.15 Arm Register

Write Only

Writing to this register initiates pre-storage of data when the ICS-130 is used in Capture mode with pre-trigger storage. See section 3.4 for details on using pre-trigger storage.

5.16 ADC Reset Register

Write only

A write to this register resets the ADC swing buffer memory pointers. It does not alter memory contents or the values of control registers. The value written to the register is unimportant. This register must be written to ***after*** configuring the ADC section and ***before*** enabling acquisition in order for the Swing Buffer memory to be correctly initialized.

5.17 Board Reset Register

Write only

A write to this register masks all interrupts, resets all control register bits to their power-up defaults (zero) and resets all on-board memory. SCV64 registers and status are unaffected. The data written to the register is unimportant.

5.18 Master Control Register (MCR)

Read/Write

This register is used to program the FPDP interface configuration.The term FPDP cluster refers to all ICS-130 boards connected on the same FPDP cable, which may be one or more. The End Slave is the final ICS-130 board on the cable, in multiple ICS-130 configurations. A Stand-alone Master is the name given to a single ICS-130 configuration.

5.18.1 MCR<09:00> - Frame Length

This field sets the number of words to be sent in a FPDP frame, which is related to the total number of active channels across all boards. This is value is given by the following expression:

$$W = (C \times P / 2) - 1$$

where C = total number of channels selected on all ICS-130 boards in cluster
P = packing factor selected in CR<5>, either 2 for unpacked or 1 for packed data

The calculated value must be programmed to all ICS-130 boards in the cluster. Note that, in an FPDP Cluster, all boards except the End Slave must be programmed for 32 active channels. An End Slave or Stand-alone Master may have any number of active channels (multiples of two only, minimum of four). The valid range of values for this field is 0 to 1023.

5.18.2 MCR<14:10> - FPDP board address

These bits program the FPDP address for the current board in an FPDP cluster. The range is 0 to (N-1), where is N is the number of ICS-130 boards connected on the FPDP and P4 cables. The valid range of values for this field is 0 to 31. The FPDP Master or Stand-alone Master must always have FPDP board address zero.

5.19 Using Diagnostic Mode

The ICS-130 has built in digital diagnostic circuitry wich allows the user to test all functionality of the board with the exception of the ADCs themselves. This is accomplished by feeding the ICS-130 with simulated ADC data from an on-board 4 KWord FIFO memory.

The diagnostic mode is used by first configuring the ICS-130 registers and then setting the DIAG ENABLE bit (CR<2>) of the Control Register. Data may then be written to the board in the ADC Data window in blocks not exceeding 4096 words. Between each block the board must be enabled until the Diag FIFO Empty bit (SR<4>) in the Status register is set. The board can then be disabled and the next block written. Once all the data has been written, the DIAG ENABLE bit can then be reset, and the data written to the board can then be read back.

When data is written to the board in diagnostic mode, the data flows through a hardware chain which includes a serial to parallel converter. As a result, the data read back has a different appearance from that written. The data read back can be considered to be in blocks of thirty-two 32-bit words, where the first word of each block is formed from the least significant bits (bit 00) of the first thirty-two words written to the board, the second word read back is formed from bit 01 of the first thirty-two words written, and so on.

The data read back can be converted to that of the written format using the following 'C' language routine (buflen must be a multiple of 16). Data obtained in the buffer rbufd should be the same as the data that was written to the board in Diagnostic mode:

```
int dataconv(
          int *rbuf,  /* Write data buffer */
          int *rbufd  /* Buffer for data converted to Read format */
          int buflen  /* Buffer length (32-bit words) */
          )
{
int    i, j, k, mask, l[32];

for (i=0; i < buflen/16; i++)
       {
       for (j=0; j<32; j++) l[j] = 0;
       for (j=0; j<16; j++)
              {
              mask = 1;
              for (k=0; k<32; k++)
                     {
                      if (rbuf[i*16 + j] & mask)
                                    l[k] = (l[k] << 1) | 0x1;
                      else
                                    l[k] = l[k] << 1;
                     mask = mask << 1;
                     }
              }
       for (j=0; j<16; j++) rbufd[i*16 + j] = (l[2*j]<<16) | l[2*j +1];
       }
} /* dataconv */
```


APPENDICES

6. APPENDICES

6.1 Typical Order of Operations for Simple Acquisition

1: Configure switches to desired VME address and master/slave configuration. Install ICS-130 in VME system chassis.
2: Reset board; clears control resgisters.
3: Set interrupt vector and level in SCV64.
4: Set control register:
- 4.1: Select trigger source (internal / external).
- 4.2: Select clock source (internal / external).
  - 4.2.1 If using external clock, activate the line termination
- 4.3: Select the ADC mode
- 4.4: Select operating mode (capture / continuous)
- 4.5: Select output data path (VME, FPDP, P2).
  - 4.5.1 If using FPDP, set the Frame Count Register
  - 4.5.2 If using FPDP, set the board address and the frame length (MCR, see section 5.18)
- 4.6: Disable diagnostic mode (see section 5.7).
- 4.7: Clear Enable bit.
- 4.8: Deassert internal trigger.

5: Set interrupt mask register.
6: Set decimation.
7: Set acquisition count.
8: Set Buffer length.
9: Program sample rate if using internal clock.
11: Reset ADC. (Loads acquisition count, buffer length, frame count and decimation values).
12: Set Enable bit in Control register.
13: If using pre-trigger capture mode, write to ARM register (starts pre-trigger storage).
14: If using internal trigger, set trigger bit in control register.
15: If using external trigger, expect external trigger signal/s to occur here.
16: Wait for interrupt from ICS-130 (VME/P2 output) or FPDP receiver.
17: Read back acquired data.
18: If using continuous mode:
IF enough data acquired, disable acquisition.
ELSE repeat steps 16..17.

If using capture mode:
IF enough data acquired, disable acquisition.
ELSE repeat steps 11..17.

6.2 Programming the Internal Clock Generator and FPDP Clock

6.2.1 Introduction

The frequency of the ICS-130 internal sampling clock is controlled by programming the programmable oscillator through the Clock Frequency register of the VMEbus memory map (see Fig. 4 above). The device is programmed not in engineering units, but by means of a complex programming word whose construction is described below; a series of control words must also be written to the device. Data is transferred to the device serially; it is necessary to write the data to the ICS-130 ADC Clock register one bit at a time; bit 0 of the register is the relevant bit. Thus the programming sequence should normally be done by repeatedly writing each control or programming word to the ADC Clock register, shifting the data right by one bit after each write, until all bits of the word has been written. An identical operation is used with the FPDP Clock register to program the FPDP data rate.

Automatic calculation of the oscillator programming word is provided by the ics130calcFoxWord() function and other functions in the 'C' language function library supplied with the optional ICS-130 software drivers, available for Solaris 2 and VxWorks environments. These routines generate a 22-bit formatted programming word for the oscillator equivalent to the nearest possible frequency to that supplied as input by the user (in MHz), and also supply the actual frequency represented by the programming word, as an output.

6.2.2 Programming Summary

The oscillator device contains two registers, the Control Register and the Programming Word Register. The programming sequence is as follows:

1. Write to Control Register to configure device and prepare the device to receive the Programming Word.

2. Write Programming Word.

3. Write to Control Register to load Programming Word data into device.

4. Wait at least 10ms for device Voltage Controlled Oscillator (VCO) frequency to stabilize.

5. Write to Control Register to enable device output of new frequency.

6.2.3 Control Register

When writing data to the Control Register, it is necessary to include a Protocol Field to identify the data as Control Register data.

Protocol Field (6 bits) = 0 1 1 1 1 0

It is important that the sequence of four ones contained in the protocol field never be sent except as part of the Protocol Field. Thus, when writing programming word data to the device, it is necessary to use a technique of inserting an extra zero after a run of three ones; this technique is called "bit stuffing", and is described in more detail in section 6.2.4.

The control register contains eight bits, which are defined as shown in Table 6.1.

Table 6.1

Bit No.	Description	Function	Power-up Default
0	Enable Programming Word register to be written by next data	0 = Program Register Disabled 1 = Program Register Enabled to Receive Data	0
1	Internal Output Disable	0 = Output is VCO or f_{REF} 1 = Output is tri-stated	0
2	Internal Multiplexer	0 = Output is VCO frequency 1 = Output is f_{REF}	1
3	Device pin 7 usage	Set to zero only	0
4 - 7	Reserved	Set to 0	0000

Control register data is written to the device starting with bit 0 of the control word, continuing to bit 7 of the control word, and followed by the 6 bits of the Protocol field, for a total of 14 bits.

Note that the default configuration of the device following power-up results in the output being at the frequency of the reference oscillator which is 14.31818 MHz.

6.2.4 Programming Register

The Programming register is written with a 22-bit word describing the required frequency of the output. However, it is essential that programming words do not mimic the Control word Protocol field (see section 6.2.3 above) by containing runs of four consecutive ones. Thus,

the device specification requires that a zero must be inserted in the word after each occasion when three one's have been transmitted to the device, regardless of whether the next bit is a 0 or a 1. This procedure is known as "bit stuffing". For this reason, the actual length of a programming word may vary between 22 and 29 bits.
For example, to send this programming data:

Last Bit				**First Bit**
1111	**0101**	**0111**	**1110**	**111111**

Transmit this serial bit stream:

Last Bit				**First Bit**
10111	**0101**	**00111**	**01110**	**01110111**

The fields of the programming word are described in Table 6.2.

Table 6.2

Field	**Bits**	**# of Bits**	**Notes**
P Counter value (P)	<21:15>	7	MSB (Most Significant Bits)
Duty Cycle Adjust Up (D)	<14>	1	Set to logic 0
Mux (M)	<13:11>	3	
Q Counter Value (Q)	<10:4>	7	
Index (I)	<3:0>	4	LSB (Least Significant Bits)

The frequency of the programmable oscillator f_{VCO}, and the output frequency f_{OUT} are determined by these fields as follows:

$$f_{VCO} = 2 * f_{REF} * (P+3)/(Q+2)$$

where, f_{REF} = Reference frequency (i.e. 14.31818 MHz)

$$f_{OUT} = f_{VCO} / 2^{M}$$

The values of the P and Q parameters must be selected so that f_{VCO} remains between 50 MHz and 150 MHz, inclusive. The value programmed to the M field programs a division register to allow sub-multiples of the VCO frequency to be obtained at the output; the maximum divisor possible is 128.

The **Index field (I)** is used to preset the VCO to an appropriate range. The value for this field should be chosen from the following table. (Note that this table is referenced to f_{VCO} rather than to the desired output frequency.)

I	f_{vco} (MHz)
0000	50 – 80
1000	80 - 150

If the desired VCO frequency is exactly 80 MHz, then either index value may be used (since both limits are tested). However, the manufacturer recommends using the setting corresponding to the higher frequency range.

6.2.5 VCO Programming Constraints

There are three primary programming constraints the user must be aware of:

1: 50MHz <= Fvco <= 150MHz
2: 1 <= P <= 127
3: 1 <= Q <= 127

The constraints have to do with the trade-offs between optimum speed with lowest noise, VCO stability and factors affecting the loop equation.

6.2.6 Program Register Example

The following is an example of how to calculate a clock programming word:

Derive the proper programming word for 12.8 MHz clock frequency.

Since 12.8 MHz < 50 MHz, quadruple it to 51.2 MHz
Set M to 010_2
Set I to 0000_2

The result:

Fout = 12.8 = (2*14.31818*(P+3)/(Q+2))/2^M

where M = 0, 1, 2, 3, 4, 5, 6, 7
since M = 2: (P+3)/(Q+2) = 1.787936735

The two choices of P and Q giving the nearest to the required frequency are:

P	Q	f_{vco}	Error (PPM)
56	31	51.19834	32
115	64	51.19834	32
90	50	51.21503	293

Taking the first set of values, i.e. (P,Q) = (56,31):

P = 56 decimal = 0111000 binary = 0(0)111000

Q = 31 decimal = 0011111 binary = 0011(0)111

NOTE: The presence of three ones in a row in both P and Q values causes zero bit-stuff values to be inserted in each. However, it is necessary to examine the values in the previous (I) field before inserting a zero in the least significant bits of the Q value. If the I field had contained a 1 in the most significant position the inserted zero would have been in a different position.

The full programming word, W is therefore:

W = P, D, M, Q, I = 00111000, 0, 010, 00110111, 0000 = 001110000010001101110000 (382370 Hex)

6.2.7 Oscillator Programming Example

The oscillator requires three control words plus the programming word in order to program a new output frequency (see programming sequence given in section 6.2.2 above). The following paragraphs provide an overview of how the control words are build along side the programming word.

All data is written to the oscillator serially through the ADC Clock Frequency or FPDP Clock Frequency register bit 0. The data is written least significant bit first. An example of

programming the oscillator to an initial or new frequency is as follows:

1. Load Control register to enable loading of the Programming word register, enable the output and select the reference frequency for output from the Internal Multiplexer. The Protocol word follows the control word. All data is shifted in LSB (Least Significant Bit) first.

	Last Bit **First Bit**
Control word	0 1 1 1 1 0 0 0 0 0 0 0 1 0 1
	Protocol Word Control Reg. Data

2. Shift in the desired output frequency value via a programming word that is 22 bits in length, plus any required bit stuffs. (Up to 29 bits can be obtained with bit-stuffing). The example programming word for 12.8 MHz, calculated in section 6.2.6 above, is shown below.

	Last Bit **First Bit**
Programming word for 12.8 MHz (0x382370)	0 0 1 1 1 0 0 0 0 0 1 0 0 0 1 1 0 1 1 1 0 0 0 0

3. Load the Control register to disable further loading of the Programming word register. This causes the required output frequency value to be programmed while keeping the output set to the reference frequency for the time being.

	Last Bit **First Bit**
Control word	0 1 1 1 1 0 0 0 0 0 0 1 0 0
	Protocol Word Control Reg. Data

4. Wait at least 10ms for the VCO to settle to the new frequency. The value will be accurate to within 0.1% within this time.

5. Load the Control register to disable the Internal Multiplexer. This will cause the output to immediately swing to the new frequency.

	Last Bit **First Bit**
Control word	0 1 1 1 1 0 0 0 0 0 0 0 0 0 0
	Protocol Word Control Reg. Data

6.3 Analog 1-16 Connector Details

Connector on board: ODD44F4R7NTX
Mating connector: ODD44M..... (consult Manufacturer's data for details)
Manufacturer: Positronics Industries Inc., (417)866-2322

PIN	SIGNAL	PIN	SIGNAL
32	CHANNEL 1-	33	CHANNEL 1+
18	CHANNEL 2-	19	CHANNEL 2+
3	CHANNEL 3-	4	CHANNEL 3+
34	CHANNEL 4-	35	CHANNEL 4+
20	CHANNEL 5-	21	CHANNEL 5+
5	CHANNEL 6-	6	CHANNEL 6+
36	CHANNEL 7-	37	CHANNEL 7+
22	CHANNEL 8-	23	CHANNEL 8+
7	CHANNEL 9-	8	CHANNEL 9+
38	CHANNEL 10-	39	CHANNEL 10+
24	CHANNEL 11-	25	CHANNEL 11+
9	CHANNEL 12-	10	CHANNEL 12+
40	CHANNEL 13-	41	CHANNEL 13+
26	CHANNEL 14-	27	CHANNEL 14+
11	CHANNEL 15-	12	CHANNEL 15+
42	CHANNEL 16-	43	CHANNEL 16+

Note 1: All other pins are connected to analog ground on the board.

6.4 Analog 17-32 Connector Details

Connector on board: Positronics ODD44F4R7NTX (see above)

PIN	SIGNAL	PIN	SIGNAL
32	CHANNEL 17-	33	CHANNEL 17+
18	CHANNEL 18-	19	CHANNEL 18+
3	CHANNEL 19-	4	CHANNEL 19+
34	CHANNEL 20-	35	CHANNEL 20+
20	CHANNEL 21-	21	CHANNEL 21+
5	CHANNEL 22-	6	CHANNEL 22+
36	CHANNEL 23-	37	CHANNEL 23+
22	CHANNEL 24-	23	CHANNEL 24+
7	CHANNEL 25-	8	CHANNEL 25+
38	CHANNEL 26-	39	CHANNEL 26+
24	CHANNEL 27-	25	CHANNEL 27+
9	CHANNEL 28-	10	CHANNEL 28+
40	CHANNEL 29-	41	CHANNEL 29+
26	CHANNEL 30-	27	CHANNEL 30+
11	CHANNEL 31-	12	CHANNEL 31+
42	CHANNEL 32-	43	CHANNEL 32+

Note 1: All other pins are connected to analog ground on the board.

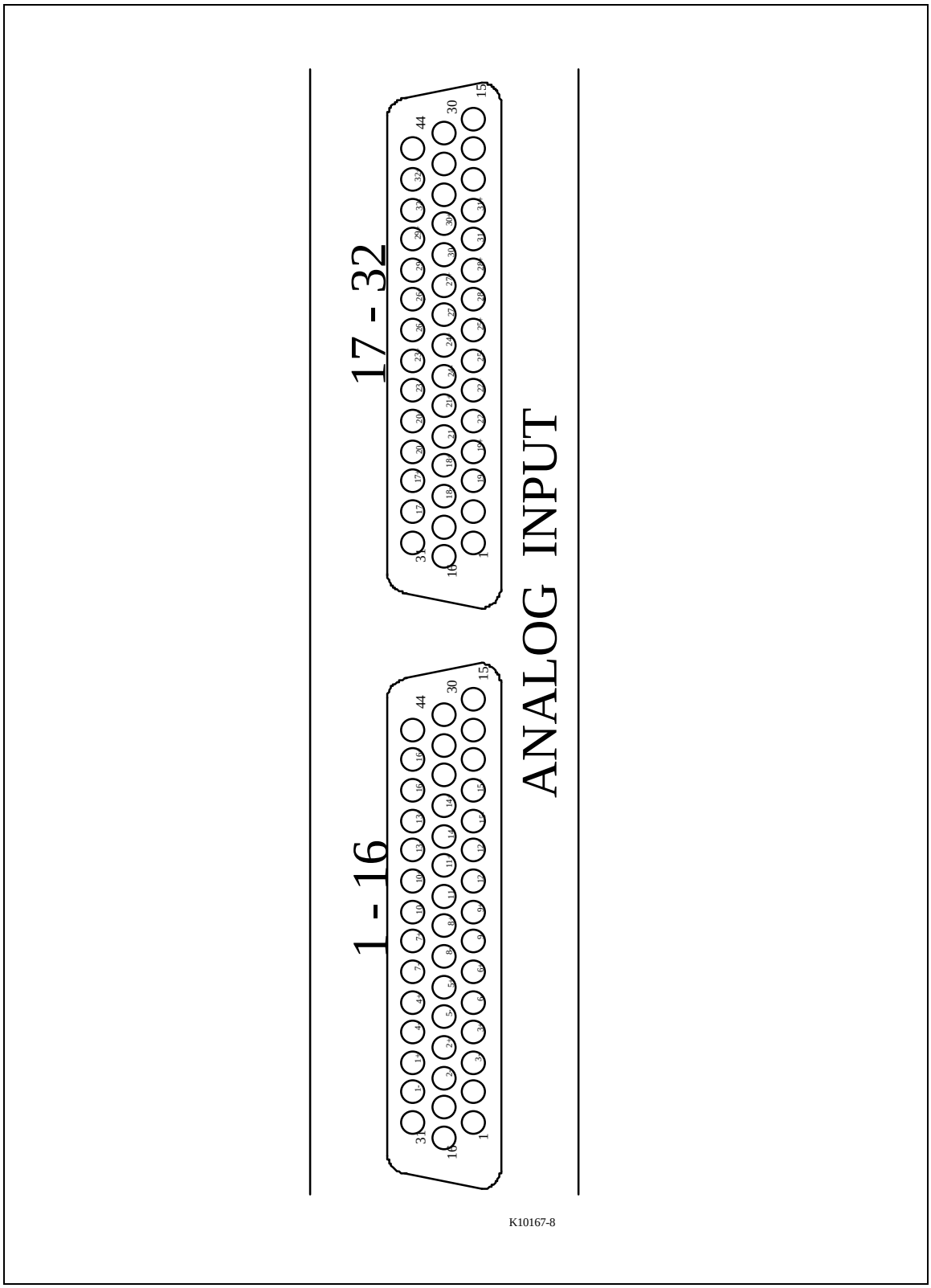


Figure 5 - ICS-130 Analog Connector Pinout

6.5 P4 Local Bus Connector Details

Suitable mating connectors are available from a number of manufacturers. The one listed is an example only.

Connector on board:	8831E-026-170L (KEL Corporation)
	P50E-026P1-RR1-TG (Robinson-Nugent)
Mating connector:	8825E-026-175 KEL (with strain relief)
	8825R-026-175 KEL (without strain relief)
	P25E-026S-TG Robinson-Nugent
Manufacturers:	KEL Corporation, (408)720-9044
	Robinson-Nugent, (812)945-0211

Pin No.	Signal Name	Description
1	ADC_CLK	Distribution ADC Clock +ve. This PECL signal is used by sampling master to ensure all slaves sample at the same time. This signal is bussed from the sampling master to all slaves, and is driven by the master only.
2	/ADC_CLK	Distribution ADC Clock -ve. This PECL signal is used by sampling master to ensure all slaves sample at the same time. This signal is bussed from the sampling master to all slaves, and is driven by the master only.
3	GND	Digital Ground
4	SYNC	ADC Frame Sync. In multiple board systems, this signal is used by the sampling master to ensure the ADC's output syncronisation on all boards. This signal is bussed from the sampling master to all slaves, and is driven by the master only.
5	GND	Digital Ground
6	-	Not Used
7	GND	Digital Ground
8	FSYNC	Decimation Sync. In multiple board systems, this signal is used by the samling master to ensure all boards decimate on the same sample. This signal is bussed from the sampling master to all slaves, and is driven by the master only.
9	GND	Digital Ground
10	TRIG	Distribution Trigger. This signal is used by the sampling master to ensule all boards trigger at the same time. This signal is bussed from the sampling master to all slaves, and is driven by the master only.
11	GND	Digital Ground
12	FP_CLK	Distribution FPDP Clock +ve. This PECL signal is used by FPDP master to ensure FPDP data transfers. This signal is bussed from the FPDF master to all slaves, and is driven by the master only.
13	PECL /FP_CLK	Distribution FPDP Clock -ve. This PECL signal is used by FPDP master to ensure

		FPDP data transfers. This signal is bussed from the FPDF master to all slaves, and is driven by the master only.
14	GND	Digital Ground
15	/FP_DV	FPDP Data Valid. This line is asserted by the FPDP transmitter. It signals valid data on the FPDP bus.
16	GND	Digital Ground
17	CHAN5	Signal used in FPDF board addressing
18	CHAN6	Signal used in FPDF board addressing
19	CHAN7	Signal used in FPDF board addressing
20	CHAN8	Signal used in FPDF board addressing
21	CHAN9	Signal used in FPDF board addressing
23	EXT_TRIG	External Trigger. The user drives this TTL signal to supply an external trigger. In multiple board systems, only the sampling master receives the external trigger signal.
24	GND	Digital Ground
25	EXT_CLK	External Clock. The user drives this TTL signal to supply an external sampling clock. In multiple board systems, only the sampling master receives the external clock signal.
26	GND	Digital Ground

6.6 P3 FPDP Details

Connector on board:	8831E-080-170L (KEL Corporation)
	P50E-080P1-SR1-TG (Robinson-Nugent)
Mating connector:	8825E-080-175 KEL (with strain relief)
	8825R-080-175 KEL (without strain relief)
	P25E-080S-TG Robinson-Nugent
Manufacturers:	KEL Corporation, (408)720-9044
	Robinson-Nugent, (812)945-0211

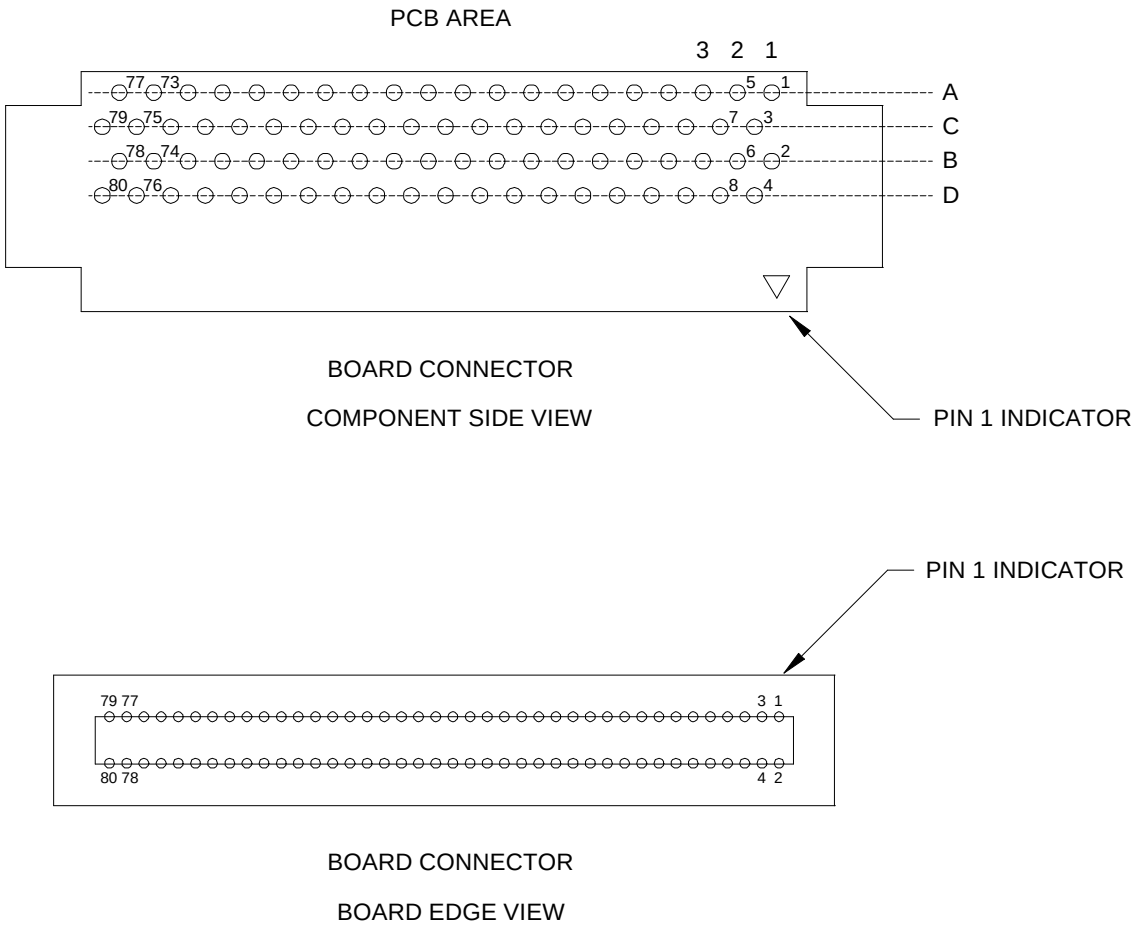
Note 1: The pinout list given in Table 6.3 below corresponds to the wire numbers on the ribbon cable, starting from the connector pin 1 index mark (see Figure 14). Previous ICS documentation referred to the board connector row and column numbers. These are also shown in the Figure to allow cross-reference.

Note 2: When connecting an ICS-130 board to a DSP board, the user should be aware that some DSP implementations have the FPDP connector inverted. For the ICS-130, the connector index mark appears at the bottom right hand side of the connector when the board is mounted vertically in a chassis. In order to prevent the need to fold the FPDP ribbon cable when connecting to the DSP board, the cable is supplied with the connector at one end inverted. The pinouts at the DSP FPDP connector are therefore reversed, compared to the ones given below. In other words, at the DSP connector pin 1 connects to pin 80 at the ICS-130, while pin 2 connects to pin 79 at the ICS-130.

The FPDP is a high performance 32 bit parallel interface configured with a ribbon cable to connect boards or systems together. The simple and well-defined physical and electrical interface provides the basis for integrating many different types of boards and sub-systems. The maximum clock rate of the ICS-130 FPDP interface is 41 MHz, providing a sustained data rate of up to 164 MBytes/s.

6.6.1 FPDP Connector Pin Assigments

The FPDP interface connector is an 80-pin high density connector available from KEL and Robinson-Nugent. The connector on the board is a KEL 8831E-080-170L or R-N P50E-080P1-RR1-TG and is shown in Figure 6. The mating cable connector is a KEL 8825E-080-175S or R-N P25E-080S-TG, and takes a single ribbon cable of 80 conductors on 0.025 inch pitch. Table 6.3 defines the connector pin assignments.



Connector row and column designations are shown outside connector outline.

Cable numbers are shown inside connector outline.

FPDP BOARD-MOUNTED CONNECTOR
KEL 8831E-080-170L
or
Robinson-Nugent P50E-080P1-SR1-TG

K10128B2

Figure 6 - FPDP Board Connector

TABLE 6.3 ICS-130 FPDP P3 connector pin assignments

PIN	SIGNAL	PIN	SIGNAL	PIN	SIGNAL	PIN	SIGNAL
1	GND	2	STROBE	3	GND	4	GND
5	GND	6	GND	7	NRDY*	8	GND
9	DIR*	10	GND	11	RESERVED	12	GND
13	SUSPEND*	14	GND	15	GND	16	GND
17	PIO2	18	GND	19	PIO1	20	GND
21	RESERVED	22	GND	23	RESERVED	24	GND
25	PSTROBE	26	GND	27	PSTROBE*	28	GND
29	SYNC*	30	GND	31	DVALID*	32	GND
33	D31	34	D30	35	GND	36	D29
37	D28	38	GND	39	D27	40	D26
41	GND	42	D25	43	D24	44	GND
45	D23	46	D22	47	GND	48	D21
49	D20	50	GND	51	D19	52	D18
53	GND	54	D17	55	D16	56	GND
57	D15	58	D14	59	GND	60	D13
61	D12	62	GND	63	D11	64	D10
65	GND	66	D09	67	D08	68	GND
69	D07	70	D06	71	GND	72	D05
73	D04	74	GND	75	D03	76	D02
77	GND	78	D01	79	D00	80	GND

6.6.2 FPDP Signals

A description of FPDP signals is given in Table 6.4.

Further details concerning the FPDP design are given in ICS INPUT Technical Note No.15, which is available from the factory or from the ICS Web site at:

www.ics-ltd.com/technotes_white_papers.html

TABLE 6.4 FPDP Signal Descriptions

Signal/s	Signal Name	Description
D31:00	Data Bus	32 bit data bus driven by the data source.
DIR*	Data Direction	The data source asserts DIR* low.
DVALID*	Data Valid	When asserted, DVALID* indicates that the data bus has valid data. This signal is generated by the data source with each data sample.
STROB	Data Strobe	STROB is a free running clock supplied by the data source. The receiving end should clock the data with the rising edge of STROB.
NRDY*	Not Ready	NRDY* is asserted by the receiver when it is not ready to receive data. The data source must sample this signal until the receiver brings it high, at which time the transfer can commence. Since NRDY* is asynchronous to STROB, the data source should double-synchronize to it before sampling its state; this avoid metastability problems.
PIO1,PIO2	Prog. I/O	The PIO signals are programmable I/O lines for user-defined functions. They can be configured as inputs or outputs.
PSTROBE	+ PECL Data Strobe	This signal along with PSTROBE* are generated by the data source as an optional differential PECL (Positive Emitter-Coupled Logic) data strobe. PSTROBE is the positive version of the differential clock and has the same polarity as STROB. For high data rate applications, the differential PECL data strobe should be used instead of STROB. The user must configure the board appropriately; see section 4.2.
PSTROBE*	- PECL Data Strobe	This signal is the negative version of the differential PECL data strobe.
RESERVED		Do not connect to reserved signals.
SUSPEND*	Suspend Data	SUSPEND* is generated by the receiver to inform the data source of a pending FIFO overflow condition. The data source is allowed as many as 16 cycles before suspending the transfer. Since SUSPEND* is asynchronous to STROB, the data source should be double-synchronized to it before sampling its state; this avoids metastability problems.
SYNC*	Sync Pulse	The data source can provide a sync pulse to the receiver to synchronize data transfers. The receiver waits for the sync pulse before accepting data. The receiver starts accepting data on the first Data Valid period following the sync pulse.

