

The Embedded I/O Company



TPMC851

Multifunction I/O

(16 bit ADC/DAC, TTL I/O, Counter)

Version 1.0

User Manual

Issue 1.0.9

September 2014



Ehlbeek 15a
30938 Burgwedel
fon 05139-9980-0
fax 05139-9980-49

www.powerbridge.de
info@powerbridge.de

TEWS TECHNOLOGIES GmbH

Am Bahnhof 7 25469 Halstenbek, Germany

e: +49 (0) 4101 4058 0 Fax: +49 (0) 4101 4058 19

e-mail: info@tews.com www.tews.com

TPMC851-10R

Multifunction I/O

(16 bit ADC/DAC, TTL I/O, Counter)

This document contains information, which is proprietary to TEWS TECHNOLOGIES GmbH. Any reproduction without written permission is forbidden.

TEWS TECHNOLOGIES GmbH has made any effort to ensure that this manual is accurate and complete. However TEWS TECHNOLOGIES GmbH reserves the right to change the product described in this document at any time without notice.

TEWS TECHNOLOGIES GmbH is not liable for any damage arising out of the application or use of the device described herein.

Style Conventions

Hexadecimal characters are specified with prefix 0x, i.e. 0x029E (that means hexadecimal value 029E).

For signals on hardware products, an ‚Active Low’ is represented by the signal name with # following, i.e. IP_RESET#.

Access terms are described as:

| | |
|-----|------------|
| W | Write Only |
| R | Read Only |
| R/W | Read/Write |
| R/C | Read/Clear |
| R/S | Read/Set |

©2014 by TEWS TECHNOLOGIES GmbH

All trademarks mentioned are property of their respective owners.

| Issue | Description | Date |
|--------------|--|----------------|
| 1.0 | Initial Issue | May 2004 |
| 1.1 | Added ADC power-up note | January 2005 |
| 1.2 | Correction Interrupt Control Register ICR, Summarization of Important Notes | February 2006 |
| 1.3 | New address TEWS LLC | September 2006 |
| 1.4 | Changed PCI9030 Local Configuration Clarified Sequencer Start / Stop Control in ADCSEQCONT | September 2006 |
| 1.5 | Added note to ADC Control Register | October 2006 |
| 1.0.6 | New notation for HW Engineering Documentation Releases | March 2009 |
| 1.0.7 | Corrected DAC Data Register Reset Value to 0x0000 | January 2010 |
| 1.0.8 | General Revision | August 2014 |
| 1.0.9 | "PCI Base Address 2 for Local Address Space 0" initial value corrected in Table "PCI9030 Header" | September 2014 |

Table of Contents

| | | |
|----------|---|-----------|
| 1 | PRODUCT DESCRIPTION | 8 |
| 2 | TECHNICAL SPECIFICATION | 10 |
| 3 | LOCAL SPACE ADDRESSING | 12 |
| | 3.1 PCI9030 Local Space Configuration | 12 |
| | 3.2 Local Register Address Space | 12 |
| | 3.2.1 Analog Input Registers | 14 |
| | 3.2.1.1 ADC Control Register ADCCONT (Offset 0x0000) | 14 |
| | 3.2.1.2 ADC Data Register ADCDATA (Offset 0x0004) | 16 |
| | 3.2.1.3 ADC Status Register ADCSTAT (Offset 0x0008) | 17 |
| | 3.2.1.4 ADC Conversion Start Register ADCCONV (Offset 0x000C) | 18 |
| | 3.2.1.5 ADC Sequencer Control Register ADCSEQCONT (Offset 0x0010) | 19 |
| | 3.2.1.6 ADC Sequencer Status Register ADCSEQSTAT (Offset 0x0014) | 20 |
| | 3.2.1.7 ADC Sequencer Timer Register ADCSEQTIME (Offset 0x0018) | 21 |
| | 3.2.1.8 ADC Sequencer Instruction RAM (Offset 0x0020 - Offset 0x009C) | 22 |
| | 3.2.2 Analog Output Registers | 24 |
| | 3.2.2.1 DAC Control Register DACCONT (Offset 0x00C0) | 24 |
| | 3.2.2.2 DAC Sequencer Status Register DACSEQSTAT (Offset 0x00C4) | 25 |
| | 3.2.2.3 DAC Sequencer Timer Register DACSEQTIME (Offset 0x00C8) | 26 |
| | 3.2.2.4 DAC Data Register DACDATA 1-8 (Offset 0x00D0 - Offset 0x00EC) | 26 |
| | 3.2.3 Digital I/O Registers | 27 |
| | 3.2.3.1 Line Direction Register LINEDIR (Offset 0x0100) | 27 |
| | 3.2.3.2 Line Debounce Enable Register LINEDEB (Offset 0x0104) | 28 |
| | 3.2.3.3 Line Debounce Time Register LINEDBT (Offset 0x0108) | 28 |
| | 3.2.3.4 Line Input Register LINEIN (Offset 0x010C) | 29 |
| | 3.2.3.5 Line Output Register LINEOUT (Offset 0x0110) | 30 |
| | 3.2.3.6 Line Interrupt Enable Register LINEIEN (Offset 0x0114) | 31 |
| | 3.2.3.7 Line Interrupt Status Register LINEIST (Offset 0x0118) | 32 |
| | 3.2.4 Counter Registers | 33 |
| | 3.2.4.1 Counter Preload Register CNTPRL (Offset 0x0120) | 33 |
| | 3.2.4.2 Counter Compare Register CNTCMP (Offset 0x0124) | 33 |
| | 3.2.4.3 Counter Data Register CNTDATA (Offset 0x0128) | 33 |
| | 3.2.4.4 Counter Control Register CNTCONT (Offset 0x012C) | 34 |
| | 3.2.4.5 Counter Status Register CNTSTAT (Offset 0x0130) | 36 |
| | 3.2.4.6 Counter Command Register CNTCOM (Offset 0x0134) | 37 |
| | 3.2.5 Interrupt Control Register ICR (Offset 0x0140) | 38 |
| | 3.3 ADC Sequencer Data RAM | 39 |
| | 3.4 ADC/DAC Calibration Data ROM | 39 |
| 4 | PCI9030 TARGET CHIP | 41 |
| | 4.1 PCI Configuration Registers (PCR) | 41 |
| | 4.1.1 PCI9030 Header | 41 |
| | 4.1.2 PCI Base Address Initialization | 42 |
| | 4.2 Local Configuration Register (LCR) | 43 |
| | 4.3 Configuration EEPROM | 44 |
| | 4.4 Local Software Reset | 44 |
| 5 | CONFIGURATION HINTS | 45 |
| | 5.1 Big / Little Endian | 45 |

| | | |
|----------|---|-----------|
| 6 | PROGRAMMING HINTS | 47 |
| 6.1 | Data Correction | 47 |
| 6.1.1 | ADC Correction Formula..... | 47 |
| 6.1.2 | DAC Correction Formula..... | 48 |
| 6.2 | ADC Operating Modes..... | 48 |
| 6.2.1 | Manual Mode | 48 |
| 6.2.2 | Sequencer Mode..... | 49 |
| 6.2.2.1 | Sequencer Errors..... | 50 |
| 6.2.3 | Application Examples..... | 51 |
| 6.2.3.1 | Fastest Conversion of an arbitrary Single Channel | 51 |
| 6.2.3.2 | Fastest Conversion of a specific Single Channel | 51 |
| 6.2.3.3 | Periodic Conversion of multiple Channels | 53 |
| 6.2.3.4 | Continuous Conversion of multiple Channels..... | 54 |
| 6.3 | DAC Operating Modes..... | 55 |
| 6.3.1 | Immediate Update..... | 55 |
| 6.3.2 | Simultaneous Update..... | 55 |
| 6.4 | Digital TTL I/O..... | 56 |
| 6.5 | Counter Operating Modes..... | 57 |
| 6.5.1 | Input Modes | 57 |
| 6.5.1.1 | Timer Mode..... | 57 |
| 6.5.1.2 | Direction Count | 57 |
| 6.5.1.3 | Up/Down Count..... | 57 |
| 6.5.1.4 | Quadrature Count | 58 |
| 6.5.2 | Count Modes..... | 58 |
| 6.5.2.1 | Divide-by-N | 58 |
| 6.5.2.2 | Single Cycle | 58 |
| 6.5.3 | Control Modes..... | 58 |
| 6.5.3.1 | No Control Mode..... | 59 |
| 6.5.3.2 | Load Mode | 59 |
| 6.5.3.3 | Latch Mode | 59 |
| 6.5.3.4 | Gate Mode | 59 |
| 6.5.3.5 | Reset Mode..... | 59 |
| 6.5.4 | Configuration Examples..... | 59 |
| 6.5.4.1 | Watchdog Timer..... | 59 |
| 6.5.4.2 | Event Counting | 60 |
| 6.5.4.3 | Input Pulse Width Measurement..... | 60 |
| 6.5.4.4 | Mechanical System Positioning | 60 |
| 6.6 | Interrupts | 60 |
| 6.6.1 | Interrupt Sources..... | 60 |
| 7 | INSTALLATION..... | 62 |
| 7.1 | ADC Input Wiring | 62 |
| 7.2 | DAC Output Wiring | 63 |
| 7.3 | TTL I/O Interface..... | 63 |
| 8 | PIN ASSIGNMENT – I/O CONNECTOR | 64 |
| 9 | IMPORTANT NOTES | 65 |
| 9.1 | Dummy Conversions after Power-up..... | 65 |
| 9.2 | Open Multiplexer Inputs | 65 |

List of Figures

| | |
|--|----|
| FIGURE 1-1 : BLOCK DIAGRAM..... | 9 |
| FIGURE 6-1 : FLOW: FASTEST CONVERSION OF AN ARBITRARY SINGLE CHANNEL | 51 |
| FIGURE 6-2 : FLOW: FASTEST CONVERSION OF A SPECIFIC SINGLE CHANNEL | 52 |
| FIGURE 6-3 : FLOW: PERIODIC CONVERSION OF MULTIPLE CHANNELS | 53 |
| FIGURE 6-4 : FLOW: CONTINUOUS CONVERSION OF MULTIPLE CHANNELS | 54 |
| FIGURE 6-5: QUADRATURE SIGNALS | 58 |
| FIGURE 7-1 : ADC INPUT WIRING | 62 |
| FIGURE 7-2 : DAC OUTPUT WIRING | 63 |
| FIGURE 7-3 : TTL I/O INTERFACE | 63 |

List of Tables

| | |
|--|----|
| TABLE 2-1 : TECHNICAL SPECIFICATION..... | 11 |
| TABLE 3-1 : PCI9030 LOCAL SPACE CONFIGURATION | 12 |
| TABLE 3-2 : REGISTER WORD/BYTE ACCESSES | 12 |
| TABLE 3-3 : REGISTER ADDRESS SPACE..... | 13 |
| TABLE 3-4 : ADC CONTROL REGISTER | 15 |
| TABLE 3-5 : ADC DATA REGISTER | 16 |
| TABLE 3-6 : ADC DATA CODING | 16 |
| TABLE 3-7 : ADC STATUS REGISTER..... | 17 |
| TABLE 3-8 : ADC SEQUENCER CONTROL REGISTER | 19 |
| TABLE 3-9 : ADC SEQUENCER STATUS REGISTER..... | 20 |
| TABLE 3-10: ERROR FLAG IRQ GENERATION | 20 |
| TABLE 3-11: ADC SEQUENCER TIMER REGISTER..... | 21 |
| TABLE 3-12: ADC SEQUENCER INSTRUCTION RAM REGISTER | 22 |
| TABLE 3-13: ADC SEQUENCER INSTRUCTION RAM REGISTER POSITIONS | 23 |
| TABLE 3-14: DAC CONTROL REGISTER | 24 |
| TABLE 3-15: DAC SEQUENCER STATUS REGISTER..... | 25 |
| TABLE 3-16: DAC SEQUENCER TIMER REGISTER..... | 26 |
| TABLE 3-17: DAC DATA REGISTER | 26 |
| TABLE 3-18: DAC DATA CODING | 26 |
| TABLE 3-19: LINE DIRECTION REGISTER..... | 27 |
| TABLE 3-20: LINE DEBOUNCE ENABLE REGISTER..... | 28 |
| TABLE 3-21: LINE DEBOUNCE TIME REGISTER | 28 |
| TABLE 3-22: LINE INPUT REGISTER..... | 29 |
| TABLE 3-23: LINE OUTPUT REGISTER..... | 30 |
| TABLE 3-24: LINE INTERRUPT ENABLE REGISTER | 31 |
| TABLE 3-25: LINE INTERRUPT STATUS REGISTER | 32 |
| TABLE 3-26: COUNTER PRELOAD REGISTER | 33 |
| TABLE 3-27: COUNTER COMPARE REGISTER | 33 |
| TABLE 3-28: COUNTER DATA REGISTER..... | 33 |

| | |
|--|----|
| TABLE 3-29: COUNTER CONTROL REGISTER..... | 35 |
| TABLE 3-30: COUNTER STATUS REGISTER | 36 |
| TABLE 3-31: COUNTER COMMAND REGISTER | 37 |
| TABLE 3-32: INTERRUPT CONTROL REGISTER..... | 38 |
| TABLE 3-33: ADC SEQUENCER DATA RAM SPACE ADDRESS MAP | 39 |
| TABLE 3-34: ADC/DAC CALIBRATION ROM DATA SPACE ADDRESS MAP | 40 |
| TABLE 4-1 : PCI9030 HEADER..... | 41 |
| TABLE 4-2 : PCI9030 PCI BASE ADDRESS USAGE | 42 |
| TABLE 4-3 : PCI9030 LOCAL CONFIGURATION REGISTER | 43 |
| TABLE 4-4 : CONFIGURATION EEPROM TPMC851-10R..... | 44 |
| TABLE 5-1 : LOCAL BUS LITTLE/BIG ENDIAN | 45 |
| TABLE 6-1 : SEQUENCER ERRORS..... | 50 |
| TABLE 6-2 : DAC LOAD MODES | 55 |
| TABLE 6-3 : INPUT MODES | 57 |
| TABLE 6-4 : CLOCK PRESCALER..... | 57 |
| TABLE 6-5 : COUNT DIRECTIONS..... | 57 |
| TABLE 6-6 : INPUT CONTROL MODE EVENTS | 59 |
| TABLE 6-7: GATE MODE | 59 |
| TABLE 6-8 : INTERRUPT SOURCES | 61 |
| TABLE 8-1 : PIN ASSIGNMENT I/O CONNECTOR..... | 64 |

1 Product Description

The TPMC851 combines 32 single ended / 16 differential channels of 16 bit multiplexed analog input, 8 channels of 16 bit analog output, 16 digital I/O lines and a 32 bit multi-purpose counter on a standard single-width PMC module.

A 16 bit ADC converts 32 single-ended or 16 differential multiplexed ADC input channels. The data acquisition and conversion time is up to 1.25 μ s without channel/gain change and up to 17.25 μ s with channel/gain change. The input multiplexer of the A/D circuit offers analog overvoltage protection of up to 70Vpp. A programmable gain amplifier allows gains of 1, 2, 4 or 8 resulting in input voltage ranges of $\pm 10V$, $\pm 5V$, $\pm 2.5V$ or $\pm 1.25V$.

The ADC part of the TPMC851 can operate in Manual Mode or Sequencer Mode:

- Manual Mode

In Manual Mode the multiplexer, programmable gain amplifier and the converter are fully controlled by the user. A conversion for a selected channel / gain can be started automatically after the settling time has elapsed, or manually by the user.

- Sequencer Mode

In Sequencer Mode each of the A/D channels can be independently enabled and configured for the sequencer. The sequencer can run continuously, at specific time intervals, or it may be triggered by an external event. Conversion data is stored in a data RAM.

The 8 analog output channels are realized by eight 16 bit digital to analog converters (DACs). The conversion time is 10 μ s. An operational amplifier drives the full-scale range of $\pm 10V$ and is capable to drive high capacitive loads. Similar to the analog inputs a sequencer can control the analog outputs. Following operation modes are available:

- Immediate Update: Updates the DAC output immediately when new data is written to the DAC channel.
- Simultaneous Update: DAC data is buffered and all DAC outputs are updated simultaneously on a trigger event:
 - Manual Update: Updates all DAC outputs on a manual event (register write)
 - Trigger Update: Updates all DAC outputs on a external event
 - Sequencer Update: Updates all DAC outputs after the sequencer timer has elapsed

Each TPMC851 is factory calibrated. The calibration data is stored in an EEPROM unique to each TPMC851.

The 16 digital TTL tri-state I/O lines with 4.7k Ω pull up resistors are ESD protected and protected against overvoltage. All 16 lines can be individually programmed as input or output, and can generate an interrupt on negative and positive transitions. Each input has an electronic debounce circuit to prevent short spikes on the input lines to cause an interrupt. The digital inputs can supply the external signals for the ADC and DAC sequencer and the 32 bit counter.

Additionally the TPMC851 offers a 32 bit multi-purpose counter. The counter includes a 32 bit preload register and a 32 bit compare register. The 32 bit counter can be fed with an internal clock or with an external signal supplied by the digital inputs. The 4 counter input modes determine the interpretation of the input signals. Additionally 3 count modes, which describe the behavior of the counter, and 4 control modes are available.

- Counter input modes:
 - Internal clock with prescaler
 - Up/Down count: first counter input counts up, second input counts down
 - Direction count: first counter input counts, second input sets count direction
 - Quadrature count with 1x, 2x or 4x resolution multiplier
- Count modes:
 - Cycling Counter: normal wrap-around count mode
 - Divide-by-N: counter is reloaded from the preload register upon a borrow or carry
 - Single Cycle: counter stops upon a borrow or carry
- Control Modes:
 - Load on Control: counter is reloaded from the preload register
 - Latch on Control: counter value is latched
 - Gate on Control: counter input is gated
 - Reset on Control: counter is reset

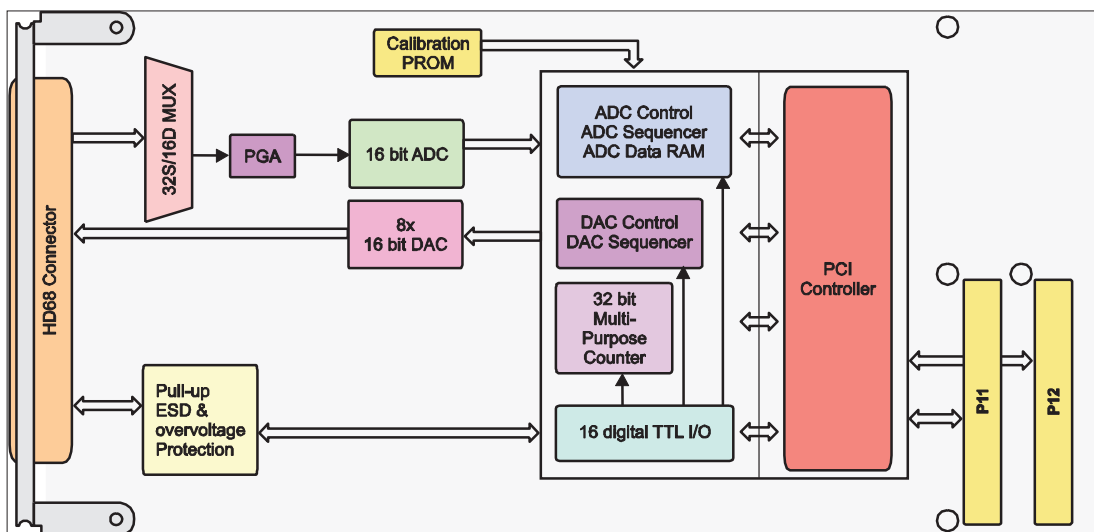


Figure 1-1 : Block Diagram

2 Technical Specification

| | |
|-------------------------------------|--|
| PMC Interface | |
| Mechanical Interface | PCI Mezzanine Card (PMC) Interface Single Size |
| Electrical Interface | PCI Rev. 2.2 compliant 33 MHz / 32 bit PCI 3.3V and 5V PCI Signaling Voltage |
| On Board Devices | |
| PCI Target Chip | PCI9030 (PLX Technology) |
| ADC | AD7663 |
| DAC | DAC7644 |
| Analog Input | |
| Number of Analog Inputs | 32 single-ended or 16 differential channels |
| Input Gain Amplifier | Programmable for gain 1, 2, 4 and 8 |
| Input Voltage Range | ±10V for gain = 1 ±5V for gain = 2 ±2.5V for gain = 4 ±1.25V for gain = 8 |
| Input Overvoltage Protection | 70Vpp |
| Calibration Data | Calibration data for gain and offset correction in ID PROM |
| Resolution | 16 bit with no missing codes |
| Conversion Time | 1.25µs without channel/gain change 17.25µs with channel/gain change 16µs / channel in sequencer mode |
| Accuracy | ±2 LSB @ 25°C after calibration |
| Linearity | ±2 LSB @ 25°C after calibration |
| ADC INL/DNL Error | ±3 LSB |
| Analog Output | |
| Number of Analog Outputs | 8 analog output channels |
| Output Voltage Range | ±10V |
| Calibration Data | Calibration data for gain and offset correction in ID PROM |
| Resolution | 16 bit |
| Conversion Time | 10µs max. |
| Accuracy | ±1 LSB @ 25°C after calibration |
| Linearity | ±2 LSB @ 25°C after calibration |
| DAC INL/DNL Error | ±4/±3 LSB |
| Capacitive Load | Up to 10 000pF |

| | |
|-------------------------------|---|
| Digital I/O | |
| Number of Digital I/O | 16 digital I/O channels |
| Overvoltage Protection | 30V |
| Minimum Pulse Width | Programmable, 100ns – 6.55ms |
| I/O Interface | |
| I/O Connector | Front I/O: HD68 SCSI-3 type female connector (AMP 787082-7) No PMC P14 I/O |
| Physical Data | |
| Power Requirements | 295 mA typical @ +3.3V DC 165 mA typical @ +5V DC 10 mA typical @ -12V DC |
| Temperature Range | Operating -40°C to +85 °C Storage -40°C to +85°C |
| MTBF | 330000 h MTBF values shown are based on calculation according to MIL-HDBK-217F and MIL-HDBK-217F Notice 2; Environment: G _B 20°C. The MTBF calculation is based on component FIT rates provided by the component suppliers. If FIT rates are not available, MIL-HDBK-217F and MIL-HDBK-217F Notice 2 formulas are used for FIT rate calculation. |
| Humidity | 5 – 95 % non-condensing |
| Weight | 82 g |

Table 2-1 : Technical Specification

3 Local Space Addressing

3.1 PCI9030 Local Space Configuration

The local on board addressable regions are accessed from the PCI side by using the PCI9030 local spaces.

| PCI9030 Local Space | PCI9030 PCI Base Address (Offset in PCI Configuration Space) | PCI Space Mapping | Size (Byte) | Port Width (Bit) | Endian Mode | Description |
|---------------------|--|-------------------|-------------|------------------|-------------|------------------------------|
| 0 | 2 (0x18) | MEM | 512 | 32 | BIG | Local Register Address Space |
| 1 | 3 (0x1C) | MEM | 64 | 16 | BIG | Sequencer Data RAM |
| 2 | 4 (0x20) | MEM | 64 | 16 | BIG | ADC/DAC Calibration ROM Data |
| 3 | 5 (0x24) | - | - | - | - | Not Used |

Table 3-1 : PCI9030 Local Space Configuration

3.2 Local Register Address Space

PCI Base Address: PCI9030 PCI Base Address 2 (Offset 0x18 in PCI Configuration Space).

The register offsets stated below refer to 32 bit accesses. The registers shorter than 32 bit are filled with zeros on long word read-accesses. Refer to chapter 'Big/Little Endian' or the following table for correct word- or byte-wide accesses.

| Access Width | Register Bits | | | |
|--------------|-----------------|--------------------|--------------------|--------------------|
| | [31:24] | [23:16] | [15:8] | [7:0] |
| 32 bit | Register Offset | | | |
| 16 bit | Register Offset | | Register Offset +2 | |
| 8 bit | Register Offset | Register Offset +1 | Register Offset +2 | Register Offset +3 |

Table 3-2 : Register word/byte Accesses

| Offset to PCI Base Address 2 | Symbol | Register Name |
|------------------------------|------------|------------------------------------|
| 0x0000 | ADCCONT | ADC Control Register |
| 0x0004 | ADCDATA | ADC Data Register |
| 0x0008 | ADCSTAT | ADC Status Register |
| 0x000C | ADCCONV | ADC Conversion Start Register |
| 0x0010 | ADCSEQCONT | ADC Sequencer Control Register |
| 0x0014 | ADCSEQSTAT | ADC Sequencer Status Register |
| 0x0018 | ADCSEQTIME | ADC Sequencer Timer Register |
| 0x001C | - | Not used |
| 0x0020 – 0x009C | ADCSEQIRAM | ADC Sequencer Instruction RAM 1-32 |
| 0x00A0 – 0x00BF | - | Not used |

| Offset to PCI Base Address 2 | Symbol | Register Name |
|------------------------------|------------|---------------------------------|
| 0x00C0 | DACCONT | DAC Control Register |
| 0x00C4 | DACSEQSTAT | DAC Sequencer Status Register |
| 0x00C8 | DACSEQTIME | DAC Sequencer Timer Register |
| 0x00CC | - | Not used |
| 0x00D0 – 0x00EC | DACDATA | DAC Data Register 1-8 |
| 0x00F0 – 0x00FF | - | Not used |
| 0x0100 | LINEDIR | Line Direction Register |
| 0x0104 | LINEDEB | Line Debounce Enable Register |
| 0x0108 | LINEDBT | Line Debounce Time Register |
| 0x010C | LINEIN | Line Input Register |
| 0x0110 | LINEOUT | Line Output Register |
| 0x0114 | LINEIEN | Line Interrupt Enable Register |
| 0x0118 | LINEIST | Line Interrupt Status Register |
| 0x011C | - | Not used |
| 0x0120 | CNTPRL | Counter Preload Register |
| 0x0124 | CNTCMP | Counter Compare Register |
| 0x0128 | CNTDATA | Counter Data Register |
| 0x012C | CNTCONT | Counter Control Register |
| 0x0130 | CNTSTAT | Counter Status Register |
| 0x0134 | CNTCOM | Counter Command Register |
| 0x0138 – 0x013F | - | Not used |
| 0x0140 | ICR | Interrupt Control Register |
| 0x0144 – 0x014F | - | Factory use only. Do not Write. |
| 0x0150 – 0x01FF | - | Not used |

Table 3-3 : Register Address Space

Addresses “Not used” will return 0 when read.

3.2.1 Analog Input Registers

3.2.1.1 ADC Control Register ADCCONT (Offset 0x0000)

| Bit | Symbol | Description | Access | Reset Value | | | | | | | | | | | | | | | | | | | | |
|-------|------------|---|---------------------|-------------|-------------|---------------------|---|---|---|------|---|---|---|-----|---|---|---|-------|---|---|---|--------|-----|----|
| 31:12 | - | Reserved Write: don't care Read: always reads as '0' | R | 0 | | | | | | | | | | | | | | | | | | | | |
| 11 | RST | ADC Reset If RST is set to '1', the ADC is reset. Current conversions are aborted. This bit is cleared immediately after a write access | R/W | 0 | | | | | | | | | | | | | | | | | | | | |
| 10 | IRQC | IRQ after Conversion Enable 0 = IRQ after Conversion disabled 1 = IRQ after Conversion enabled An interrupt will be generated after a conversion is finished (ADC_BUSY changes from 1 to 0). For pending interrupts and interrupt acknowledge see the ADC Status Register ADCSTAT. | R/W | 0 | | | | | | | | | | | | | | | | | | | | |
| 9 | IRQST | IRQ after Settling Time Enable 0 = IRQ after Settling Time disabled 1 = IRQ after Settling Time enabled An interrupt will be generated after the settling time has elapsed (SETTL_BUSY changes from 1 to 0) For pending interrupts and interrupt acknowledge see the ADC Status Register ADCSTAT. | R/W | 0 | | | | | | | | | | | | | | | | | | | | |
| 8 | ASTC | Automatic Settling Time Control 0 = OFF (Normal Mode) A conversion must be initiated manually in the ADC Conversion Start Register ADCCONV. 1 = ON (Automatic Mode) A conversion is automatically initiated after the settling time has elapsed. The ADC settling time is appr. 16µs. | R/W | 0 | | | | | | | | | | | | | | | | | | | | |
| 7:6 | GAIN [1:0] | Gain Selection (Analog Input Amplifier) <table border="1" data-bbox="488 1290 1161 1473"> <thead> <tr> <th>GAIN1</th> <th>GAIN0</th> <th>Gain Factor</th> <th>Input Voltage Range</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> <td>±10V</td> </tr> <tr> <td>0</td> <td>1</td> <td>2</td> <td>±5V</td> </tr> <tr> <td>1</td> <td>0</td> <td>4</td> <td>±2.5V</td> </tr> <tr> <td>1</td> <td>1</td> <td>8</td> <td>±1.25V</td> </tr> </tbody> </table> | GAIN1 | GAIN0 | Gain Factor | Input Voltage Range | 0 | 0 | 1 | ±10V | 0 | 1 | 2 | ±5V | 1 | 0 | 4 | ±2.5V | 1 | 1 | 8 | ±1.25V | R/W | 00 |
| GAIN1 | GAIN0 | Gain Factor | Input Voltage Range | | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 1 | ±10V | | | | | | | | | | | | | | | | | | | | | |
| 0 | 1 | 2 | ±5V | | | | | | | | | | | | | | | | | | | | | |
| 1 | 0 | 4 | ±2.5V | | | | | | | | | | | | | | | | | | | | | |
| 1 | 1 | 8 | ±1.25V | | | | | | | | | | | | | | | | | | | | | |
| 5 | SE/DIFF | Single/Differential Mode Control 0 = Single-ended mode 32 single-ended input channels available 1 = Differential mode 16 differential input channels available | R/W | 0 | | | | | | | | | | | | | | | | | | | | |

| Bit | Symbol | Description | Access | Reset Value | | | | | | | | | | | | | | | | | | | | | |
|---------|-------------------------------------|---|---------|-------------------------------------|-------------------------------------|-------|-----|-----|-----|-----|-----|-------|------|------|-------|------|-----|-----|-----|-----|-------|------|-----|--|--|
| 4:0 | CS[4:0] | Channel Select (Analog Input Channel) | R/W | 00000 | | | | | | | | | | | | | | | | | | | | | |
| | | <table border="1"> <thead> <tr> <th>CS[4:0]</th> <th>Single-ended Channel SE/DIFF = 0</th> <th>Differential Channel SE/DIFF = 1</th> </tr> </thead> <tbody> <tr> <td>00000</td> <td>CH1</td> <td>CH1</td> </tr> <tr> <td>...</td> <td>...</td> <td>...</td> </tr> <tr> <td>01111</td> <td>CH16</td> <td>CH16</td> </tr> <tr> <td>10000</td> <td>CH17</td> <td>N/A</td> </tr> <tr> <td>...</td> <td>...</td> <td>...</td> </tr> <tr> <td>11111</td> <td>CH32</td> <td>N/A</td> </tr> </tbody> </table> | CS[4:0] | Single-ended Channel SE/DIFF = 0 | Differential Channel SE/DIFF = 1 | 00000 | CH1 | CH1 | ... | ... | ... | 01111 | CH16 | CH16 | 10000 | CH17 | N/A | ... | ... | ... | 11111 | CH32 | N/A | | |
| CS[4:0] | Single-ended Channel SE/DIFF = 0 | Differential Channel SE/DIFF = 1 | | | | | | | | | | | | | | | | | | | | | | | |
| 00000 | CH1 | CH1 | | | | | | | | | | | | | | | | | | | | | | | |
| ... | ... | ... | | | | | | | | | | | | | | | | | | | | | | | |
| 01111 | CH16 | CH16 | | | | | | | | | | | | | | | | | | | | | | | |
| 10000 | CH17 | N/A | | | | | | | | | | | | | | | | | | | | | | | |
| ... | ... | ... | | | | | | | | | | | | | | | | | | | | | | | |
| 11111 | CH32 | N/A | | | | | | | | | | | | | | | | | | | | | | | |
| | | 'N/A' = ADC Input connected to GND | | | | | | | | | | | | | | | | | | | | | | | |

Table 3-4 : ADC Control Register

A write to this register sets the new channel and gain. Subsequent write accesses are ignored until the settling time has elapsed. This register is developed for a word (16 bit) or long word (32 bit) read/write access. Byte accesses on this register are not supported and will fail.

Do NOT write to this register while the sequencer is running! A write to this register while the sequencer is running can lead to unpredictable results!

Interrupts are generated only if the Master Interrupt Enable (MIE) is enabled in the Interrupt Control Register ICR.

3.2.1.2 ADC Data Register ADCDATA (Offset 0x0004)

The ADC Data Register contains the converted data value. The output format is two's complement.

This register is updated if the conversion was initiated with the ADC Conversion Start Register ADCCONV or by the Automatic Settling Time Control ASTC on the falling edge of ADC_BUSY in the ADC Status Register (ADCSTAT). Conversions initiated by the sequencer will have no effect to this register.

| Bit | Symbol | Description | Access | Reset Value |
|-------|--------|--|--------|-------------|
| 31:16 | - | Reserved Write: don't care Read: always reads as '0' | R | 0 |
| 15:0 | | Stores the converted 16 bit data value. | R | 0 |

Table 3-5 : ADC Data Register

| Description | ±10V (Gain 1) | ±5V (Gain 2) | ±2.5V (Gain 4) | ±1.25V (Gain 8) | Two's Complement |
|-----------------------|---------------|--------------|----------------|-----------------|---------------------|
| Full Scale Range | ±10V | ±5V | ±2.5 | ±1.25 | |
| Least Significant Bit | 305.2µV | 152.6µV | 76.2µV | 38.15µV | |
| FSR - 1LSB | 9.999695V | 4.999847V | 2.499924V | 1.249962V | 0x7FFF ¹ |
| Midscale + 1LSB | 305.2µV | 152.6µV | 76.2µV | 38.15µV | 0x0001 |
| Midscale | 0V | 0V | 0V | 0V | 0x0000 |
| Midscale - 1LSB | -305.2µV | -152.6µV | -76.2µV | -38.15µV | 0xFFFF |
| -FSR + 1LSB | -9.999695V | -4.999847V | -2.499924V | -1.249962V | |
| Full Scale (neg.) | -10V | -5V | -2.5V | 1.25V | 0x8000 ² |

¹This is also the code for an overrange analog input.

²This is also the code for an underrange analog input.

Table 3-6 : ADC Data Coding

After power up the on board ADC device is in a random state and requires two dummy conversions before operating correctly. This is based on the chip design of the ADC device.

Software should ignore the data of the first two ADC conversions after power-up.

The software drivers from TEWS TECHNOLOGIES already include these two dummy conversions.

3.2.1.3 ADC Status Register ADCSTAT (Offset 0x0008)

| Bit | Symbol | Description | Access | Reset Value |
|------|------------|---|--------|-------------|
| 31:4 | - | Reserved Write: don't care Read: always read as '0' | R | 0 |
| 3 | SETTL IRQ | Pending Settling Time Interrupts (Read), Interrupt acknowledge (Write) On a read-access this bit shows a pending Settling Time interrupt. Pending interrupts are marked with a '1'. An interrupt is acknowledged by writing a '1' to this bit after the error flags are cleared. | R/C | 0 |
| 2 | ADC IRQ | Pending Conversion Interrupts (Read), Interrupt acknowledge (Write) On a read-access this bit shows a pending Conversion interrupt. Pending interrupts are marked with a '1'. An interrupt is acknowledged by writing a '1' to this bit after the error flags are cleared. | R/C | 0 |
| 1 | SETTL BUSY | SETTL_BUSY Indicates that the required settling time after a write to the CONTREG register is not yet done. This bit is set by writing to the CONTREG register. The bit is cleared when the required settling time has elapsed. This bit must be read as '0' before a conversion is started by a write to the ADC Conversion Start Register ADCCONV. The ADC settling time is appr. 16µs. | R | 0 |
| 0 | ADC BUSY | ADC_BUSY Indicates if an actual data conversion is in progress. If "Automatic Settling Time Mode" is OFF, this bit is set by writing to the ADC Conversion Start Register ADCCONV. If "Automatic Settling Time Mode" is ON, this bit is set automatically after the settling time has elapsed. This bit must be read as '0' before the conversion data is read from the DATAREG register. | R | 0 |

Table 3-7 : ADC Status Register

Interrupts are generated only if the Master Interrupt Enable (MIE) is enabled in the Interrupt Control Register ICR.

3.2.1.4 ADC Conversion Start Register ADCCONV (Offset 0x000C)

The ADC Conversion Start Register is used to start an ADC conversion when “Automatic Settling Time Mode” is OFF. Any write access starts the conversion. The user must read the SETTL_BUSY bit in the ADC Status Register as '0' or wait for a Settling Time Interrupt before the conversion is started. The ADC_BUSY bit in the ADC Status Register indicates if the conversion data in the ADC Data Register is valid (ADC_BUSY bit = '0').

It is allowed to set up a new channel/gain by writing to the ADC Control Register CONTREG immediately after starting an ADC conversion.

If “Sequencer Mode” is selected (SEQCONT register bit 0 is set to '1') all write accesses to the ADC Conversion Start Register ADCCONV are ignored.

Please pay attention to the chapter “Important Notes”.

3.2.1.5 ADC Sequencer Control Register ADCSEQCONT (Offset 0x0010)

| Bit | Symbol | Description | Access | Reset Value |
|------|-------------|--|--------|-------------|
| 31:3 | - | Reserved Write: don't care Read: always reads as '0' | R | 0 |
| 2 | SEQ INT ENA | Sequencer Interrupt Enable Control 0 = Sequencer Interrupt Disabled 1 = Sequencer Interrupt Enabled An interrupt request will be generated if any error flag is set in the ADCSEQSTAT Register (Sequencer Data Valid or Sequencer Error). For pending interrupts and interrupt acknowledge see ADC Sequencer Status Register ADCSEQSTAT. | R/W | 0 |
| 1 | SEQ TRIG | Sequencer Trigger Source 0 = Sequencer Timer 1 = External Signal (Dig I/O Line 0) | R/W | 0 |
| 0 | SEQ ON | Sequencer Start / Stop Control 0 = Stops the Sequencer after last Instruction (Sequencer Continuous Mode: Sequencer stops immediately). No DATA AV will be issued. 1 = Starts the Sequencer immediately | R/W | 0 |

Table 3-8 : ADC Sequencer Control Register

If an error flag (DATA_OVERFLOW_ERROR, TIMER_ERROR, I-RAM_ERROR) is set in the Sequencer Status Register SEQSTAT, the sequencer will be stopped after the last instruction (SEQ_ON will be set to '0'). The user must clear the status bits and start the sequencer again.

Interrupts are generated only if the Master Interrupt Enable (MIE) is enabled in the Interrupt Control Register ICR.

Please pay attention to the chapter “Important Notes” before using the sequencer.

If ADC_OUT in the Line Direction Register LINEDIR is set to '1', the trigger signal for the sequencer is available for external use on Dig I/O Line 0.

Do not use the External Signal as Sequencer Trigger Source if ADC_OUT is enabled. This will cause the sequencer to lock.

3.2.1.6 ADC Sequencer Status Register ADCSEQSTAT (Offset 0x0014)

| Bit | Symbol | Description | Access | Reset Value |
|------|---------------------|--|--------|-------------|
| 31:5 | - | Reserved Write: don't care Read: always reads as '0' | R | 0 |
| 4 | SEQ IRQ | Pending Sequencer Interrupts (Read), On a read-access this bit shows a pending Sequencer interrupt. Pending interrupts are marked with a '1'. An interrupt is acknowledged when the error flags are cleared. | R | 0 |
| 3 | I-RAM ERROR | Instruction RAM Error Flag Set by the sequencer if the sequencer has been started and there is no instruction in the Sequencer Instruction RAM. To clear this flag the user must write '1' to this bit. | R/C | 0 |
| 2 | TIMER ERROR | Timer Error Flag Set by the sequencer if the sequencer timer has elapsed but the actual sequence is still in progress. To clear the Timer Error Flag the user must write '1' to this bit. If the Sequencer Timer Register is 0 (Sequencer Continuous Mode) the Timer Error Flag always reads as '0'. | R/C | 0 |
| 1 | DATA OVERFLOW ERROR | Data Overflow Error Flag Set by the sequencer if the last sequencer instruction is done and the Data Available Flag of the previous sequence has not yet been cleared by the user. To clear the error flag the user must write '1' to this bit. If the Sequencer Timer Register is 0 (Sequencer Continuous Mode) the Data Overflow Error Flag always reads as '0'. | R/C | 0 |
| 0 | DATA AV | Data Available Flag Set if a sequence is done and new ADC Data is available in the ADC Data RAM. After reading the ADC Data RAM the user must clear the Data Available Flag by writing '1' to this bit. | R/C | 0 |

Table 3-9 : ADC Sequencer Status Register

| Description | Sequencer Timer | Continuous Mode | External Trigger |
|----------------------------|-----------------|-----------------|------------------|
| Data Available Flag | Active, IRQ | Active, no IRQ | Active, IRQ |
| Data Overflow Error Flag | Active, IRQ | Disabled | Active, IRQ |
| Timer Error Flag | Active, IRQ | Disabled | Active, IRQ |
| Instruction RAM Error Flag | Active, IRQ | Active, IRQ | Active, IRQ |

Table 3-10: Error Flag IRQ generation

Also see chapter 'Sequencer Errors' for details.

3.2.1.7 ADC Sequencer Timer Register ADCSEQTIME (Offset 0x0018)

| Bit | Symbol | Description | Access | Reset Value |
|-------|--------|---|------------|-------------|
| 31:16 | - | Reserved Write: don't care Read: always reads as '0' | R | 0 |
| 0:15 | | Sequencer Timer Preload Value | R/W | 0 |

Table 3-11: ADC Sequencer Timer Register

The Sequencer Timer is programmable from 100µs to 6.5535s in 100µs steps. The time base for the sequencer timer is derived from an on board 40 MHz oscillator.

Whenever the timer reaches the programmed value the sequencer starts a new sequence with the first instruction found in the instruction RAM.

Assure that the time needed to complete a sequence is suitable to the chosen sequence timer value. If the sequence timer elapses while a sequence is still in progress, a timer error will be asserted.

If the Sequencer Timer Register is set to '0', the "Sequencer Continuous Mode" is selected. The sequencer will start again with the first instruction of the sequence immediately after the last instruction of the previous sequence has been completed. In this mode the Timer Error Flag (TIMER_ERROR) and the Data Overflow Error Flag (DATA_OVERFLOW_ERROR) are not active and will read as '0'. The Data Available Flag (DATA_AV) will be active, but will not produce an interrupt.

3.2.1.8 ADC Sequencer Instruction RAM (Offset 0x0020 - Offset 0x009C)

| Bit | Symbol | Description | Access | Reset Value | | | | | | | | | | | | | | | | | | | | |
|-------|------------|--|---------------------|-------------|-------------|---------------------|---|---|---|------|---|---|---|-----|---|---|---|-------|---|---|---|--------|-----|----|
| 31:4 | - | Reserved Write: don't care Read: always reads as '0' | R | 0 | | | | | | | | | | | | | | | | | | | | |
| 3:2 | GAIN [1:0] | Gain Selection (Analog Input Amplifier) <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>GAIN1</th> <th>GAIN0</th> <th>Gain Factor</th> <th>Input Voltage Range</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> <td>±10V</td> </tr> <tr> <td>0</td> <td>1</td> <td>2</td> <td>±5V</td> </tr> <tr> <td>1</td> <td>0</td> <td>4</td> <td>±2.5V</td> </tr> <tr> <td>1</td> <td>1</td> <td>8</td> <td>±1.25V</td> </tr> </tbody> </table> | GAIN1 | GAIN0 | Gain Factor | Input Voltage Range | 0 | 0 | 1 | ±10V | 0 | 1 | 2 | ±5V | 1 | 0 | 4 | ±2.5V | 1 | 1 | 8 | ±1.25V | R/W | 00 |
| GAIN1 | GAIN0 | Gain Factor | Input Voltage Range | | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 1 | ±10V | | | | | | | | | | | | | | | | | | | | | |
| 0 | 1 | 2 | ±5V | | | | | | | | | | | | | | | | | | | | | |
| 1 | 0 | 4 | ±2.5V | | | | | | | | | | | | | | | | | | | | | |
| 1 | 1 | 8 | ±1.25V | | | | | | | | | | | | | | | | | | | | | |
| 1 | ENA | Enable this ADC Channel for the Sequencer 0 = Sequencer will pass over the ADC Channel 1 = Sequencer converts this ADC Channel and updates the ADC Data in the Sequencer Data RAM at the end of the sequence. | R/W | 0 | | | | | | | | | | | | | | | | | | | | |
| 0 | SE/DIFF | Single/Differential Mode Control 0 = Single-ended mode 32 single ended channels available 1 = Differential mode 16 differential channels available. Only channels 1-16 can be configured as differential channel. Channels 17 to 32 are used as input for channels 1 to 16. | R/W | 0 | | | | | | | | | | | | | | | | | | | | |

Table 3-12: ADC Sequencer Instruction RAM Register

If a channel is configured as differential channel, the instruction of the associated channel is ignored (see following chart or chapter 'Pin Assignment – I/O Connector' for the associated channels).

Within a sequence mixed single-ended and differential modes are possible.

Example: Channel 1 to channel 8 (with channels 17 to channel 24) are selected as differential inputs, channel 9 to channel 16 and channel 25 to channel 32 as single-ended input channels.

Only the Sequencer Data RAM locations of channels enabled for the sequence are updated at the end of a sequence. All other RAM locations are empty or contain old data.

Example: If only channel 1, channel 2, and channel 8 are enabled, only the three ADC RAM locations for channel 1, channel 2, and channel 8 are updated at the end of the sequence. The user must only read these three ADC RAM locations then.

If a sequence is started with an empty instruction RAM, an I-RAM error is issued.

The Sequencer Instruction RAM is accessible only while the sequencer is not running (SEQ_ON = 0).

The following table shows the position of the Sequencer Instruction RAM bytes in the local memory space.

| Offset | Single Ended | Differential |
|--------|--------------|----------------------------|
| 0x20 | Channel 1 | Channel 1 |
| 0x24 | Channel 2 | Channel 2 |
| 0x28 | Channel 3 | Channel 3 |
| 0x2C | Channel 4 | Channel 4 |
| 0x30 | Channel 5 | Channel 5 |
| 0x34 | Channel 6 | Channel 6 |
| 0x38 | Channel 7 | Channel 7 |
| 0x3C | Channel 8 | Channel 8 |
| 0x40 | Channel 9 | Channel 9 |
| 0x44 | Channel 10 | Channel 10 |
| 0x48 | Channel 11 | Channel 11 |
| 0x4C | Channel 12 | Channel 12 |
| 0x50 | Channel 13 | Channel 13 |
| 0x54 | Channel 14 | Channel 14 |
| 0x58 | Channel 15 | Channel 15 |
| 0x5C | Channel 16 | Channel 16 |
| 0x60 | Channel 17 | N/A (Input for Channel 1) |
| 0x64 | Channel 18 | N/A (Input for Channel 2) |
| 0x68 | Channel 19 | N/A (Input for Channel 3) |
| 0x6C | Channel 20 | N/A (Input for Channel 4) |
| 0x70 | Channel 21 | N/A (Input for Channel 5) |
| 0x74 | Channel 22 | N/A (Input for Channel 6) |
| 0x78 | Channel 23 | N/A (Input for Channel 7) |
| 0x7C | Channel 24 | N/A (Input for Channel 8) |
| 0x80 | Channel 25 | N/A (Input for Channel 9) |
| 0x84 | Channel 26 | N/A (Input for Channel 10) |
| 0x88 | Channel 27 | N/A (Input for Channel 11) |
| 0x8C | Channel 28 | N/A (Input for Channel 12) |
| 0x90 | Channel 29 | N/A (Input for Channel 13) |
| 0x94 | Channel 30 | N/A (Input for Channel 14) |
| 0x98 | Channel 31 | N/A (Input for Channel 15) |
| 0x9C | Channel 32 | N/A (Input for Channel 16) |

Table 3-13: ADC Sequencer Instruction RAM Register positions

3.2.2 Analog Output Registers

3.2.2.1 DAC Control Register DACCONT (Offset 0x00C0)

| Bit | Symbol | Description | Access | Reset Value | | | | | | | | | | | | | | |
|---------------------|--|--|--------------|-------------|------------------|--|----|--|---------------------|--|----|--|----|--|----|--|-----|---|
| 31:6 | - | Reserved Write: don't care Read: always reads as '0' | R | 0 | | | | | | | | | | | | | | |
| 5 | DAC IRQ ENA | DAC Interrupt Enable 0 = DAC Interrupt Disabled 1 = DAC Interrupt Enabled An interrupt is issued on a Sequencer Data Request in the Sequencer Update or Trigger Update mode. For pending interrupts and interrupt acknowledge see DAC Sequencer Status Register DACSEQSTAT. | R/W | 0 | | | | | | | | | | | | | | |
| 4:3 | LOAD SEL | Load Mode Select The Load Mode determines how the DAC outputs are updated. <table border="1" data-bbox="523 779 1241 1254"> <thead> <tr> <th>LOADSEL[1:0]</th> <th>Load Mode</th> </tr> </thead> <tbody> <tr> <td colspan="2">Immediate Update</td> </tr> <tr> <td>00</td> <td>Immediate Update: The DAC outputs are updated immediately on a write access to a DAC Data Register.</td> </tr> <tr> <td colspan="2">Simultaneous Update</td> </tr> <tr> <td>01</td> <td>Manual Update: All DAC outputs are updated simultaneously when LOADDAC is set to '1'.</td> </tr> <tr> <td>10</td> <td>Sequencer Update: All DAC outputs are updated simultaneously when the DAC Sequencer Timer reaches zero.</td> </tr> <tr> <td>11</td> <td>Trigger Update: All DAC outputs are updated simultaneously by an external trigger (Dig I/O Line 1).</td> </tr> </tbody> </table> <p>In the Simultaneous Update modes it is required to load the DACs with data via the DAC Data Registers. This data is buffered until the trigger event on which all DACs are updated simultaneously with the buffered data. If a channel was not loaded with new data, it will use the old data previously buffered.</p> | LOADSEL[1:0] | Load Mode | Immediate Update | | 00 | Immediate Update: The DAC outputs are updated immediately on a write access to a DAC Data Register. | Simultaneous Update | | 01 | Manual Update: All DAC outputs are updated simultaneously when LOADDAC is set to '1'. | 10 | Sequencer Update: All DAC outputs are updated simultaneously when the DAC Sequencer Timer reaches zero. | 11 | Trigger Update: All DAC outputs are updated simultaneously by an external trigger (Dig I/O Line 1). | R/W | 0 |
| LOADSEL[1:0] | Load Mode | | | | | | | | | | | | | | | | | |
| Immediate Update | | | | | | | | | | | | | | | | | | |
| 00 | Immediate Update: The DAC outputs are updated immediately on a write access to a DAC Data Register. | | | | | | | | | | | | | | | | | |
| Simultaneous Update | | | | | | | | | | | | | | | | | | |
| 01 | Manual Update: All DAC outputs are updated simultaneously when LOADDAC is set to '1'. | | | | | | | | | | | | | | | | | |
| 10 | Sequencer Update: All DAC outputs are updated simultaneously when the DAC Sequencer Timer reaches zero. | | | | | | | | | | | | | | | | | |
| 11 | Trigger Update: All DAC outputs are updated simultaneously by an external trigger (Dig I/O Line 1). | | | | | | | | | | | | | | | | | |
| 2 | LOAD DAC | In Manual Update mode writing a '1' loads all DACs simultaneously. This bit is cleared immediately after a write access. | W | 0 | | | | | | | | | | | | | | |
| 1 | RSTSEL | Reset Select. Indicates the action of RSTDAC. A RSTDAC command will set the DAC registers and outputs to midscale. Hardwired to '1' | R | 1 | | | | | | | | | | | | | | |
| 0 | RSTDAC | Reset DACs Writing '1' resets the DACs. The DACs are set to midscale. This bit is cleared immediately after a write access. | W | 0 | | | | | | | | | | | | | | |

Table 3-14: DAC Control Register

DAC analog outputs of all channels are held in a reset state for the time of a PCI or local reset.

Interrupts are generated only if the Master Interrupt Enable (MIE) is enabled in the Interrupt Control Register ICR.

If DAC_OUT in the Line Direction Register LINEDIR is set to '1', the trigger signal for the sequencer is available for external use on Dig I/O Line 1.

Do not use the Trigger Update mode if DAC_OUT is enabled. This will cause the sequencer to lock.

3.2.2.2 DAC Sequencer Status Register DACSEQSTAT (Offset 0x00C4)

| Bit | Symbol | Description | Access | Reset Value |
|------|---------|---|--------|-------------|
| 31:3 | - | Reserved Write: don't care Read: always reads as '0' | R | 0 |
| 2 | DAC IRQ | Pending DAC Interrupts (Read), On a read-access this bit shows a pending DAC interrupt. Pending interrupts are marked with a '1'. An interrupt is acknowledged when the SDR bit is cleared. | R | 0 |
| 1 | SDU | Sequencer Data Underflow (bit is used to signalize a data underflow condition for the sequencer data RAM) 1 = Sequencer Data Underflow (sequencer is ready for the next sequence but the user has not yet confirmed new data in sequencer data RAM). 0 = All DAC Data Registers have been updated with new data. | R | 1 |
| 0 | SDR | Sequencer Data Request/Acknowledge (bit is used to signalize data request for the sequencer data RAM) 1 = Sequencer Data Request (sequencer is requesting new data in the sequencer data RAM) During the Simultaneous/Sequencer/Trigger Update modes this status bit must be cleared after the sequencer data RAM has been updated with data for the next sequence. The bit is cleared by writing a '1'. | R/C | 0 |

Table 3-15: DAC Sequencer Status Register

3.2.2.3 DAC Sequencer Timer Register DACSEQTIME (Offset 0x00C8)

| Bit | Symbol | Description | Access | Reset Value |
|-------|--------|---|--------|-------------|
| 31:16 | - | Reserved Write: don't care Read: always reads as '0' | R | 0 |
| 0:15 | | Sequencer Timer Preload Value | R/W | 0 |

Table 3-16: DAC Sequencer Timer Register

Setting LOADSEL in the DAC Control Register (DACCONT) to 'Sequencer Update' starts the timer. The minimum for the Sequencer Timer Preload Value is 1; the timer will not start if the Sequencer Timer Preload Value is 0.

The sequencer timer is programmable from 100 μ s to 6.5535s in 100 μ s steps. The sequencer timer is loaded with the value in the Sequencer Timer Register when a sequence is started. In sequencer timer mode the start of the next sequence is delayed for: Register Value * 100 μ s.

The time base for the sequencer timer is derived from an on board 40 MHz oscillator.

3.2.2.4 DAC Data Register DACDATA 1-8 (Offset 0x00D0 - Offset 0x00EC)

| Bit | Symbol | Description | Access | Reset Value |
|-------|--------------|---|--------|-------------|
| 31:16 | - | Reserved Write: don't care Read: always reads as '0' | R | 0 |
| 0:15 | DAC DATA 1-8 | DAC Data Value When LOADSEL in the DAC Control Register is set to "00", the DAC output is updated immediately. | R/W | 0x0000 |

Table 3-17: DAC Data Register

After power-on or reset the value of all DAC Data registers are set to 0x8000. The analog output voltage is set to Midscale 0V DC.

The settling time for a full range voltage step is typically 8 μ s and maximum 10 μ s.

| Description | Output Voltage | DAC Data Value |
|-----------------------|----------------|----------------|
| Full Scale Range | $\pm 10V$ | |
| Least Significant Bit | 305.2 μ V | |
| FSR - 1LSB | 9.999695V | 0x7FFF |
| Midscale + 1LSB | 305.2 μ V | 0x0001 |
| Midscale | 0V | 0x0000 |
| Midscale - 1LSB | -305.2 μ V | 0xFFFF |
| -FSR + 1LSB | -9.999695V | 0x8001 |
| Full Scale (neg.) | -10V | 0x8000 |

Table 3-18: DAC Data Coding

3.2.3 Digital I/O Registers

3.2.3.1 Line Direction Register LINEDIR (Offset 0x0100)

| Bit | Symbol | Description | Access | Reset Value |
|-------|-----------|--|--------|-------------|
| 31:18 | - | Reserved Write: don't care Read: always reads as '0' | R | 0 |
| 17 | DAC OUT | Enable DAC Sequencer trigger output 1 = Override LINEDIR1 and output DAC Sequencer trigger on I/O Line 1 0 = Normal LINEDIR1 operation | R/W | 0 |
| 16 | ADC OUT | Enable ADC Sequencer trigger output 1 = Override LINEDIR0 and output ADC Sequencer trigger on I/O Line 0 0 = Normal LINEDIR0 operation | R/W | 0 |
| 15 | LINEDIR15 | I/O Line 15 | R/W | 0 |
| 14 | LINEDIR14 | I/O Line 14 | | 0 |
| 13 | LINEDIR13 | I/O Line 13 | | 0 |
| 12 | LINEDIR12 | I/O Line 12 | | 0 |
| 11 | LINEDIR11 | I/O Line 11 | | 0 |
| 10 | LINEDIR10 | I/O Line 10 | | 0 |
| 9 | LINEDIR9 | I/O Line 9 | | 0 |
| 8 | LINEDIR8 | I/O Line 8 | | 0 |
| 7 | LINEDIR7 | I/O Line 7 | | 0 |
| 6 | LINEDIR6 | I/O Line 6 | | 0 |
| 5 | LINEDIR5 | I/O Line 5 | | 0 |
| 4 | LINEDIR4 | I/O Line 4 | | 0 |
| 3 | LINEDIR3 | I/O Line 3 | | 0 |
| 2 | LINEDIR2 | I/O Line 2 | | 0 |
| 1 | LINEDIR1 | I/O Line 1 | | 0 |
| 0 | LINEDIR0 | I/O Line 0 | | 0 |

Table 3-19: Line Direction Register

The reset value of the Line Direction Register is 0x0000. That means all TTL I/O lines are inputs.

3.2.3.2 Line Debounce Enable Register LINEDEB (Offset 0x0104)

| Bit | Symbol | Description | Access | Reset Value |
|-------|-----------|---|--------|-------------|
| 31:16 | - | Reserved Write: don't care Read: always reads as '0' | R | 0 |
| 15 | LINEDEB15 | I/O Line 15 | R/W | 0 |
| 14 | LINEDEB14 | I/O Line 14 | | 0 |
| 13 | LINEDEB13 | I/O Line 13 | | 0 |
| 12 | LINEDEB12 | I/O Line 12 | | 0 |
| 11 | LINEDEB11 | I/O Line 11 | | 0 |
| 10 | LINEDEB10 | I/O Line 10 | | 0 |
| 9 | LINEDEB9 | I/O Line 9 | | 0 |
| 8 | LINEDEB8 | I/O Line 8 | | 0 |
| 7 | LINEDEB7 | I/O Line 7 | | 0 |
| 6 | LINEDEB6 | I/O Line 6 | | 0 |
| 5 | LINEDEB5 | I/O Line 5 | | 0 |
| 4 | LINEDEB4 | I/O Line 4 | | 0 |
| 3 | LINEDEB3 | I/O Line 3 | | 0 |
| 2 | LINEDEB2 | I/O Line 2 | | 0 |
| 1 | LINEDEB1 | I/O Line 1 | | 0 |
| 0 | LINEDEB0 | I/O Line 0 | | 0 |

Table 3-20: Line Debounce Enable Register

For the enabled I/O lines a digital debounce filter will be applied. The debounce duration can be programmed in the Line Debounce Time Register LINEDBT. The digital input signal must have duration greater than the programmed debounce duration in order to be recognized as a valid input signal.

The disabled I/O lines will not be filtered.

3.2.3.3 Line Debounce Time Register LINEDBT (Offset 0x0108)

| Bit | Symbol | Description | Access | Reset Value |
|-------|--------|---|--------|-------------|
| 31:16 | - | Reserved Write: don't care Read: always reads as '0' | R | 0 |
| 0:15 | | Line Debounce Time Value | R/W | 0 |

Table 3-21: Line Debounce Time Register

The Line Debounce Time Register allows to program the debounce duration from 100ns to 6.55ms. Use following formula to calculate the debounce duration:

$$\text{Debounce duration} = (\text{Line Debounce Time Value} * 100\text{ns}) + 100\text{ns}$$

The debounce time is common for all 16 inputs.

3.2.3.4 Line Input Register LINEIN (Offset 0x010C)

| Bit | Symbol | Description | Access | Reset Value |
|-------|----------|--|--------|-------------|
| 31:16 | - | Reserved Write: don't care Read: always reads as '0' | R | 0 |
| 15 | LINEIN15 | I/O Line 15 | R | 0 |
| 14 | LINEIN14 | I/O Line 14 | | 0 |
| 13 | LINEIN13 | I/O Line 13 | | 0 |
| 12 | LINEIN12 | I/O Line 12 | | 0 |
| 11 | LINEIN11 | I/O Line 11 | | 0 |
| 10 | LINEIN10 | I/O Line 10 | | 0 |
| 9 | LINEIN9 | I/O Line 9 | | 0 |
| 8 | LINEIN8 | I/O Line 8 | | 0 |
| 7 | LINEIN7 | I/O Line 7 | | 0 |
| 6 | LINEIN6 | I/O Line 6 | | 0 |
| 5 | LINEIN5 | I/O Line 5 | | 0 |
| 4 | LINEIN4 | I/O Line 4 | | 0 |
| 3 | LINEIN3 | I/O Line 3 | | 0 |
| 2 | LINEIN2 | I/O Line 2 | | 0 |
| 1 | LINEIN1 | I/O Line 1 | | 0 |
| 0 | LINEIN0 | I/O Line 0 | | 0 |

Reflects the actual input state of the debounced TTL I/O lines at all times, including I/O lines configured as output in the LINEDIR (to provide readback functionality)
0 = TTL I/O line logic low
1 = TTL I/O line logic high

Table 3-22: Line Input Register

3.2.3.5 Line Output Register LINEOUT (Offset 0x0110)

| Bit | Symbol | Description | Access | Reset Value |
|-------|-----------|--|--------|-------------|
| 31:16 | - | Reserved Write: don't care Read: always reads as '0' | R | 0 |
| 15 | LINEOUT15 | I/O Line 15 | R/W | 0 |
| 14 | LINEOUT14 | I/O Line 14 | | 0 |
| 13 | LINEOUT13 | I/O Line 13 | | 0 |
| 12 | LINEOUT12 | I/O Line 12 | | 0 |
| 11 | LINEOUT11 | I/O Line 11 | | 0 |
| 10 | LINEOUT10 | I/O Line 10 | | 0 |
| 9 | LINEOUT9 | I/O Line 9 | | 0 |
| 8 | LINEOUT8 | I/O Line 8 | | 0 |
| 7 | LINEOUT7 | I/O Line 7 | | 0 |
| 6 | LINEOUT6 | I/O Line 6 | | 0 |
| 5 | LINEOUT5 | I/O Line 5 | | 0 |
| 4 | LINEOUT4 | I/O Line 4 | | 0 |
| 3 | LINEOUT3 | I/O Line 3 | | 0 |
| 2 | LINEOUT2 | I/O Line 2 | | 0 |
| 1 | LINEOUT1 | I/O Line 1 | | 0 |
| 0 | LINEOUT0 | I/O Line 0 | | 0 |

Sets output state of the TTL I/O lines. Writes on I/O lines configured as input in the LINEDIR are allowed but will take no effect.
0 = TTL I/O line logic low
1 = TTL I/O line logic high

Table 3-23: Line Output Register

3.2.3.6 Line Interrupt Enable Register LINEIEN (Offset 0x0114)

| Bit | Symbol | Description | Access | Reset Value |
|-----|------------|-------------|--------|-------------|
| 31 | LINEIENN15 | I/O Line 15 | R/W | 0 |
| 30 | LINEIENN14 | I/O Line 14 | | 0 |
| 29 | LINEIENN13 | I/O Line 13 | | 0 |
| 28 | LINEIENN12 | I/O Line 12 | | 0 |
| 27 | LINEIENN11 | I/O Line 11 | | 0 |
| 26 | LINEIENN10 | I/O Line 10 | | 0 |
| 25 | LINEIENN9 | I/O Line 9 | | 0 |
| 24 | LINEIENN8 | I/O Line 8 | | 0 |
| 23 | LINEIENN7 | I/O Line 7 | | 0 |
| 22 | LINEIENN6 | I/O Line 6 | | 0 |
| 21 | LINEIENN5 | I/O Line 5 | | 0 |
| 20 | LINEIENN4 | I/O Line 4 | | 0 |
| 19 | LINEIENN3 | I/O Line 3 | | 0 |
| 18 | LINEIENN2 | I/O Line 2 | | 0 |
| 17 | LINEIENN1 | I/O Line 1 | | 0 |
| 16 | LINEIENN0 | I/O Line 0 | | 0 |
| 15 | LINEIENP15 | I/O Line 15 | R/W | 0 |
| 14 | LINEIENP14 | I/O Line 14 | | 0 |
| 13 | LINEIENP13 | I/O Line 13 | | 0 |
| 12 | LINEIENP12 | I/O Line 12 | | 0 |
| 11 | LINEIENP11 | I/O Line 11 | | 0 |
| 10 | LINEIENP10 | I/O Line 10 | | 0 |
| 9 | LINEIENP9 | I/O Line 9 | | 0 |
| 8 | LINEIENP8 | I/O Line 8 | | 0 |
| 7 | LINEIENP7 | I/O Line 7 | | 0 |
| 6 | LINEIENP6 | I/O Line 6 | | 0 |
| 5 | LINEIENP5 | I/O Line 5 | | 0 |
| 4 | LINEIENP4 | I/O Line 4 | | 0 |
| 3 | LINEIENP3 | I/O Line 3 | | 0 |
| 2 | LINEIENP2 | I/O Line 2 | | 0 |
| 1 | LINEIENP1 | I/O Line 1 | | 0 |
| 0 | LINEIENP0 | I/O Line 0 | | 0 |

Table 3-24: Line Interrupt Enable Register

Interrupts are generated only if the Master Interrupt Enable (MIE) is enabled in the Interrupt Control Register ICR.

3.2.3.7 Line Interrupt Status Register LINEIST (Offset 0x0118)

| Bit | Symbol | Description | Access | Reset Value |
|-----|------------|-------------|--------|-------------|
| 31 | LINEISTN15 | I/O Line 15 | R/C | 0 |
| 30 | LINEISTN14 | I/O Line 14 | | 0 |
| 29 | LINEISTN13 | I/O Line 13 | | 0 |
| 28 | LINEISTN12 | I/O Line 12 | | 0 |
| 27 | LINEISTN11 | I/O Line 11 | | 0 |
| 26 | LINEISTN10 | I/O Line 10 | | 0 |
| 25 | LINEISTN9 | I/O Line 9 | | 0 |
| 24 | LINEISTN8 | I/O Line 8 | | 0 |
| 23 | LINEISTN7 | I/O Line 7 | | 0 |
| 22 | LINEISTN6 | I/O Line 6 | | 0 |
| 21 | LINEISTN5 | I/O Line 5 | | 0 |
| 20 | LINEISTN4 | I/O Line 4 | | 0 |
| 19 | LINEISTN3 | I/O Line 3 | | 0 |
| 18 | LINEISTN2 | I/O Line 2 | | 0 |
| 17 | LINEISTN1 | I/O Line 1 | | 0 |
| 16 | LINEISTN0 | I/O Line 0 | | 0 |
| 15 | LINEISTP15 | I/O Line 15 | R/C | 0 |
| 14 | LINEISTP14 | I/O Line 14 | | 0 |
| 13 | LINEISTP13 | I/O Line 13 | | 0 |
| 12 | LINEISTP12 | I/O Line 12 | | 0 |
| 11 | LINEISTP11 | I/O Line 11 | | 0 |
| 10 | LINEISTP10 | I/O Line 10 | | 0 |
| 9 | LINEISTP9 | I/O Line 9 | | 0 |
| 8 | LINEISTP8 | I/O Line 8 | | 0 |
| 7 | LINEISTP7 | I/O Line 7 | | 0 |
| 6 | LINEISTP6 | I/O Line 6 | | 0 |
| 5 | LINEISTP5 | I/O Line 5 | | 0 |
| 4 | LINEISTP4 | I/O Line 4 | | 0 |
| 3 | LINEISTP3 | I/O Line 3 | | 0 |
| 2 | LINEISTP2 | I/O Line 2 | | 0 |
| 1 | LINEISTP1 | I/O Line 1 | | 0 |
| 0 | LINEISTP0 | I/O Line 0 | | 0 |

Table 3-25: Line Interrupt Status Register

3.2.4 Counter Registers

3.2.4.1 Counter Preload Register CNTPRL (Offset 0x0120)

| Bit | Symbol | Description | Access | Reset Value |
|------|--------|--|--------|-------------|
| 31:0 | CNTPRL | Counter Preload Register The value of this register can be loaded into the counter by: <ul style="list-style-type: none"> - Setting bit 4 (LCNT) of the Channel Command Register - An impulse on the Control-Input when the 'Load on Control'-mode is active - Automatically in the 'Divide-by-N'-mode every time the counter creates a borrow or a carry | R/W | 0 |

Table 3-26: Counter Preload Register

3.2.4.2 Counter Compare Register CNTCMP (Offset 0x0124)

| Bit | Symbol | Description | Access | Reset Value |
|------|--------|---|--------|-------------|
| 31:0 | CNTCMP | Counter Compare Register Every time the counter matches the Counter Compare Register value, bit 2 (MAT) of the Channel Status Register is set to '1' and, if enabled, a Match Interrupt is generated. | R/W | -1 |

Table 3-27: Counter Compare Register

3.2.4.3 Counter Data Register CNTDATA (Offset 0x0128)

| Bit | Symbol | Description | Access | Reset Value |
|------|---------|---|--------|-------------|
| 31:0 | CNTDATA | Counter Data Register This Register contains the actual Counter Data Value. A Latch Mode event loads the current counter value into the Counter Data Register. The counter value is latched until the Counter Data Register is read or when bit 5 (CDLT) of the Channel Status Register is set to '1'. | R | 0 |

Table 3-28: Counter Data Register

The Counter Data Register will not load again while the latch is active. If a Latch Mode event occurs while the Counter Data Register Latch is active, the OVFL bit in the Counter Status Register will be set to indicate that data was lost.

To avoid data inconsistencies this register is developed for a long word (32 bit) read/write access. Byte or word accesses on this register are not supported and will fail.

3.2.4.4 Counter Control Register CNTCONT (Offset 0x012C)

| Bit | Symbol | Description | Access | Reset Value | | | | | | | | | | | | | | | |
|--------|-----------------|--|--------|--------------|-----------------|-----------------|----------------|---------------------------------|-----|----------------|---------------------------------|-----------|----------------|---------------------------------|-----|-----|-------|-----|----|
| 31:15 | - | Reserved Write: don't care Read: always reads as '0' | - | 0 | | | | | | | | | | | | | | | |
| 14:12 | POL | Input Polarity The Input Polarity Control can be used to adapt the counter inputs to the input source polarity. <table border="1" data-bbox="560 524 1206 669"> <thead> <tr> <th>POL</th> <th>Input</th> <th>Polarity</th> </tr> </thead> <tbody> <tr> <td>14</td> <td>TTL I/O line 4</td> <td>0 = high active, 1 = low active</td> </tr> <tr> <td>13</td> <td>TTL I/O line 3</td> <td>0 = high active, 1 = low active</td> </tr> <tr> <td>12</td> <td>TTL I/O line 2</td> <td>0 = high active, 1 = low active</td> </tr> </tbody> </table> | POL | Input | Polarity | 14 | TTL I/O line 4 | 0 = high active, 1 = low active | 13 | TTL I/O line 3 | 0 = high active, 1 = low active | 12 | TTL I/O line 2 | 0 = high active, 1 = low active | R/W | 000 | | | |
| POL | Input | Polarity | | | | | | | | | | | | | | | | | |
| 14 | TTL I/O line 4 | 0 = high active, 1 = low active | | | | | | | | | | | | | | | | | |
| 13 | TTL I/O line 3 | 0 = high active, 1 = low active | | | | | | | | | | | | | | | | | |
| 12 | TTL I/O line 2 | 0 = high active, 1 = low active | | | | | | | | | | | | | | | | | |
| 11 | CIEN | Enable Control Mode Interrupt 0 = Control Mode Interrupt disabled 1 = Control Mode Interrupt enabled An interrupt will be generated on a control mode event. For pending interrupts and interrupt acknowledge see the Counter Status Register CNTSTAT. | R/W | 0 | | | | | | | | | | | | | | | |
| 10 | MIEN | Enable Match Interrupt 0 = Match Interrupt disabled 1 = Match Interrupt enabled An interrupt will be generated when the counter value matches the Counter Compare Register CNTCMP. For pending interrupts and interrupt acknowledge see the Counter Status Register CNTSTAT. | R/W | 0 | | | | | | | | | | | | | | | |
| 9:7 | ZCM | Control Mode The Control Mode determines how the counter interprets the Control Input (I/O line 4). <table border="1" data-bbox="681 1180 1083 1400"> <thead> <tr> <th>ZCM</th> <th>Control Mode</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>No Control Mode</td> </tr> <tr> <td>001</td> <td>Load Mode</td> </tr> <tr> <td>010</td> <td>Latch Mode</td> </tr> <tr> <td>011</td> <td>Gate Mode</td> </tr> <tr> <td>100</td> <td>Reset Mode</td> </tr> </tbody> </table> See chapter 'Control Modes' for details. | ZCM | Control Mode | 000 | No Control Mode | 001 | Load Mode | 010 | Latch Mode | 011 | Gate Mode | 100 | Reset Mode | R/W | 000 | | | |
| ZCM | Control Mode | | | | | | | | | | | | | | | | | | |
| 000 | No Control Mode | | | | | | | | | | | | | | | | | | |
| 001 | Load Mode | | | | | | | | | | | | | | | | | | |
| 010 | Latch Mode | | | | | | | | | | | | | | | | | | |
| 011 | Gate Mode | | | | | | | | | | | | | | | | | | |
| 100 | Reset Mode | | | | | | | | | | | | | | | | | | |
| 6:5 | SCM | Count Mode The Count Mode determines the behavior of the counter. <table border="1" data-bbox="681 1503 1083 1648"> <thead> <tr> <th>SCM</th> <th>Count Mode</th> </tr> </thead> <tbody> <tr> <td>00</td> <td>Cycling Counter</td> </tr> <tr> <td>01</td> <td>Divide-by-N</td> </tr> <tr> <td>10</td> <td>Single Cycle</td> </tr> </tbody> </table> See chapter 'Count Modes' for details. | SCM | Count Mode | 00 | Cycling Counter | 01 | Divide-by-N | 10 | Single Cycle | R/W | 00 | | | | | | | |
| SCM | Count Mode | | | | | | | | | | | | | | | | | | |
| 00 | Cycling Counter | | | | | | | | | | | | | | | | | | |
| 01 | Divide-by-N | | | | | | | | | | | | | | | | | | |
| 10 | Single Cycle | | | | | | | | | | | | | | | | | | |
| 4:3 | CLKDIV | Internal Clock Prescaler <table border="1" data-bbox="601 1720 1163 1904"> <thead> <tr> <th>CLKDIV</th> <th>Prescaler</th> <th>Clock frequency</th> </tr> </thead> <tbody> <tr> <td>00</td> <td>1x</td> <td>40 MHz</td> </tr> <tr> <td>01</td> <td>2x</td> <td>20 MHz</td> </tr> <tr> <td>10</td> <td>4x</td> <td>10 MHz</td> </tr> <tr> <td>11</td> <td>8x</td> <td>5 MHz</td> </tr> </tbody> </table> | CLKDIV | Prescaler | Clock frequency | 00 | 1x | 40 MHz | 01 | 2x | 20 MHz | 10 | 4x | 10 MHz | 11 | 8x | 5 MHz | R/W | 00 |
| CLKDIV | Prescaler | Clock frequency | | | | | | | | | | | | | | | | | |
| 00 | 1x | 40 MHz | | | | | | | | | | | | | | | | | |
| 01 | 2x | 20 MHz | | | | | | | | | | | | | | | | | |
| 10 | 4x | 10 MHz | | | | | | | | | | | | | | | | | |
| 11 | 8x | 5 MHz | | | | | | | | | | | | | | | | | |
| 2:0 | INPUT | Input Mode | R/W | 000 | | | | | | | | | | | | | | | |

| Bit | Symbol | Description | Access | Reset Value | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------|---------------------|---|--------|-------------|--------------|-----|------------------|---|-----|---------------|--------------------------|-----|-----------------|--------------------------|-----|-----------------|----------------|-----|---------------|----------------|-----|---------------------|----------------|-----|---------------------|----------------|-----|---------------------|----------------|--|--|
| | | <p>The Input Mode determines the input source for the counter and how the counter interprets these input signals. The quadrature mode can be used with a 1x, 2x, or 4x resolution multiplier.</p> <table border="1"> <thead> <tr> <th>INPUT</th> <th>Input Mode</th> <th>Input Source</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>Counter disabled</td> <td>-</td> </tr> <tr> <td>001</td> <td>Timer Mode Up</td> <td>Internal Clock Prescaler</td> </tr> <tr> <td>010</td> <td>Timer Mode Down</td> <td>Internal Clock Prescaler</td> </tr> <tr> <td>011</td> <td>Direction Count</td> <td>I/O line 2 & 3</td> </tr> <tr> <td>100</td> <td>Up/Down Count</td> <td>I/O line 2 & 3</td> </tr> <tr> <td>101</td> <td>Quadrature Count 1x</td> <td>I/O line 2 & 3</td> </tr> <tr> <td>110</td> <td>Quadrature Count 2x</td> <td>I/O line 2 & 3</td> </tr> <tr> <td>111</td> <td>Quadrature Count 4x</td> <td>I/O line 2 & 3</td> </tr> </tbody> </table> <p>See chapter 'Input Modes' for details.</p> | INPUT | Input Mode | Input Source | 000 | Counter disabled | - | 001 | Timer Mode Up | Internal Clock Prescaler | 010 | Timer Mode Down | Internal Clock Prescaler | 011 | Direction Count | I/O line 2 & 3 | 100 | Up/Down Count | I/O line 2 & 3 | 101 | Quadrature Count 1x | I/O line 2 & 3 | 110 | Quadrature Count 2x | I/O line 2 & 3 | 111 | Quadrature Count 4x | I/O line 2 & 3 | | |
| INPUT | Input Mode | Input Source | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 000 | Counter disabled | - | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 001 | Timer Mode Up | Internal Clock Prescaler | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 010 | Timer Mode Down | Internal Clock Prescaler | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 011 | Direction Count | I/O line 2 & 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 100 | Up/Down Count | I/O line 2 & 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 101 | Quadrature Count 1x | I/O line 2 & 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 110 | Quadrature Count 2x | I/O line 2 & 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 111 | Quadrature Count 4x | I/O line 2 & 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

Table 3-29: Counter Control Register

Interrupts are generated only if the Master Interrupt Enable (MIE) is enabled in the Interrupt Control Register ICR.

3.2.4.5 Counter Status Register CNTSTAT (Offset 0x0130)

| Bit | Symbol | Description | Access | Reset Value |
|-------|--------|---|--------|-------------|
| 31:10 | - | Reserved Read: always reads as '0' | R | 0 |
| 9 | CIRQ | Pending Control Mode Interrupts (Read), Interrupt acknowledge (Write) On a read-access this bit shows a pending Control Mode Interrupt. Pending interrupts are marked with a '1'. An interrupt is acknowledged by writing a '1' to this bit. | R/C | 0 |
| 8 | MIRQ | Pending Match Interrupts (Read), Interrupt acknowledge (Write) On a read-access this bit shows a pending Match Interrupt. Pending interrupts are marked with a '1'. An interrupt is acknowledged when the MAT bit is cleared. | R | 0 |
| 7 | SGL | Single Cycle active In Single Cycle counting mode this bit is set to '1' if the counter is enabled. | R | 0 |
| 6 | OVFL | Counter Data Register Latch Overflow If a Latch Mode event occurs while the Counter Data Register Latch is active, this bit will be set to indicate that data was lost. This bit must be reset by writing a '1' to this bit. | R/C | 0 |
| 5 | CDLT | Counter Data Register Latch This bit is set to '1', when the Counter Data Register is latched due to a 'Latch on Control'. This bit is cleared after a read access to the Output Register or by writing a '1' to this bit. | R/C | 0 |
| 4 | DIR | Count Direction This bit indicates the direction, the counter is counting to. '1' indicates up, '0' indicates down. In the 'Up/Down Count' mode this bit indicates the direction at the last count. In the 'Direction Count' mode this bit corresponds to the Y-input. | R | 0 |
| 3 | SGN | Sign The Sign bit is set to '1' when the counter overflows, and set to '0' when the counter underflows. After reset or power-up this bit should be considered as "don't care" until the first Carry or Borrow occurred. | R | 0 |
| 2 | MAT | Match This bit is set to '1' when the counter value matches the value of the Counter Compare Register. This bit must be reset by writing a '1' to this bit. | R/C | 0 |
| 1 | CRY | Carry This bit is set to '1' when the counter changes from 0xFFFFFFFF to 0x00000000. This bit must be reset by writing a '1' to this bit. | R/C | 0 |
| 0 | BOR | Borrow This bit is set to '1' when the counter changes from 0x00000000 to 0xFFFFFFFF. This bit must be reset by writing a '1' to this bit. | R/C | 0 |

Table 3-30: Counter Status Register

3.2.4.6 Counter Command Register CNTCOM (Offset 0x0134)

| Bit | Symbol | Description | Access | Reset Value |
|------|--------|--|--------|-------------|
| 31:2 | - | Reserved Write: don't care | - | - |
| 1 | LCNT | Load Counter Write '1' to load the counter with the value of the Counter Preload Register CNTPRL. This bit is cleared immediately after a write access. | W | 0 |
| 0 | RCNT | Reset Counter Write '1' to reset the counter. This bit is cleared immediately after a write access. | W | 0 |

Table 3-31: Counter Command Register

3.2.5 Interrupt Control Register ICR (Offset 0x0140)

| Bit | Symbol | Description | Access | Reset Value |
|------|---------|--|--------|-------------|
| 31:8 | - | Reserved Write: don't care Read: always reads as '0' | R | 0 |
| 7 | CIRQ | Pending Control Mode Interrupt This interrupt must be acknowledged in the Counter Status Register CNTSTAT | R | 0 |
| 6 | MIRQ | Pending Match Interrupt This interrupt must be acknowledged in the Counter Status Register CNTSTAT | R | 0 |
| 5 | LINEIST | Pending Digital I/O Line IRQ This interrupt must be acknowledged in the Line Interrupt Status Register LINEIST | R | 0 |
| 4 | DAC IRQ | Pending DAC Sequencer Interrupt This interrupt must be acknowledged in the DAC Sequencer Status Register DACSEQSTAT | R | 0 |
| 3 | SEQ IRQ | Pending ADC Sequencer Interrupt This interrupt must be acknowledged in the ADC Sequencer Status Register ADCSEQSTAT | R | 0 |
| 2 | IRQST | Pending IRQ after Settling Time This interrupt must be acknowledged in the ADC Status Register ADCSTAT | R | 0 |
| 1 | IRQC | Pending IRQ after Conversion This interrupt must be acknowledged in the ADC Status Register ADCSTAT | R | 0 |
| 0 | MIE | Master Interrupt Enable 0 = All interrupts requests are disabled 1 = Interrupts requests are enabled | R/W | 0 |

Table 3-32: Interrupt Control Register

The interrupt flags in this register are informational only. All interrupts must be acknowledged in their associated registers.

Interrupts are generated only if the Master Interrupt Enable (MIE) is enabled in the Interrupt Control Register ICR.

3.3 ADC Sequencer Data RAM

PCI Base Address: PCI9030 PCI Base Address 3 (Offset 0x1C in PCI Configuration Space).

The ADC Sequencer Data RAM is a 32 x 16 bit wide RAM storing the converted data values.

Each channel has its own ADC Data location.

The ADC Sequencer Data RAM is updated after every full sequencer cycle. Only the channels enabled in the ADC Sequencer Instruction RAM are updated.

| Offset to PCI Base Address 3 | Description | Size (Bit) |
|------------------------------|-------------------------------|------------|
| 0x00 | Channel 1 ADC Sequencer Data | 16 |
| 0x02 | Channel 2 ADC Sequencer Data | 16 |
| ... | ... | ... |
| 0x3C | Channel 31 ADC Sequencer Data | 16 |
| 0x3E | Channel 32 ADC Sequencer Data | 16 |

Table 3-33: ADC Sequencer Data RAM Space Address Map

3.4 ADC/DAC Calibration Data ROM

PCI Base Address: PCI9030 PCI Base Address 4 (Offset 0x20 in PCI Configuration Space).

The calibration data values are determined at factory and stored in this ROM space.

There is one Offset Error value and one Gain Error value for each ADC gain which are valid for all 32 channels. For the DACs again there is an Offset Error value and a Gain Error value for each DAC channel.

See the Programming Hints chapter for data correction formulas.

The calibration data is read only.

The calibration data is available 500 μ s after reset.

| Offset to PCI Base Address 4 | Description | Size (Bit) |
|------------------------------|---------------------------------------|------------|
| 0x00 | ADC Offset _{ERROR} Gain 1 | 16 |
| 0x02 | ADC Gain _{ERROR} Gain 1 | 16 |
| 0x04 | ADC Offset _{ERROR} Gain 2 | 16 |
| 0x06 | ADC Gain _{ERROR} Gain 2 | 16 |
| 0x08 | ADC Offset _{ERROR} Gain 4 | 16 |
| 0x0A | ADC Gain _{ERROR} Gain 4 | 16 |
| 0x0C | ADC Offset _{ERROR} Gain 8 | 16 |
| 0x0E | ADC Gain _{ERROR} Gain 8 | 16 |
| 0x10 | DAC Channel 1 Offset _{ERROR} | 16 |
| 0x12 | DAC Channel 1 Gain _{ERROR} | 16 |
| 0x14 | DAC Channel 2 Offset _{ERROR} | 16 |
| 0x16 | DAC Channel 2 Gain _{ERROR} | 16 |

| | | |
|------|---------------------------------------|----|
| 0x18 | DAC Channel 3 Offset _{ERROR} | 16 |
| 0x1A | DAC Channel 3 Gain _{ERROR} | 16 |
| 0x1C | DAC Channel 4 Offset _{ERROR} | 16 |
| 0x1E | DAC Channel 4 Gain _{ERROR} | 16 |
| 0x20 | DAC Channel 5 Offset _{ERROR} | 16 |
| 0x22 | DAC Channel 5 Gain _{ERROR} | 16 |
| 0x24 | DAC Channel 6 Offset _{ERROR} | 16 |
| 0x26 | DAC Channel 6 Gain _{ERROR} | 16 |
| 0x28 | DAC Channel 7 Offset _{ERROR} | 16 |
| 0x2A | DAC Channel 7 Gain _{ERROR} | 16 |
| 0x2C | DAC Channel 8 Offset _{ERROR} | 16 |
| 0x2E | DAC Channel 8 Gain _{ERROR} | 16 |

Table 3-34: ADC/DAC Calibration ROM Data Space Address Map

4 PCI9030 Target Chip

4.1 PCI Configuration Registers (PCR)

4.1.1 PCI9030 Header

| PCI CFG Register Address | Write '0' to all unused (Reserved) bits | | | | | | | PCI writeable | Initial Values (Hex Values) | |
|--------------------------|---|----|-------------|----|---------------------|---------------|-----------------|---------------|-----------------------------|-------------|
| | 31 | 24 | 23 | 16 | 15 | 8 | 7 | | | 0 |
| 0x00 | Device ID | | | | Vendor ID | | | | N | 0353 1498 |
| 0x04 | Status | | | | Command | | | | Y | 0280 0000 |
| 0x08 | Class Code | | | | | Revision ID | | | N | 118000 00 |
| 0x0C | BIST | | Header Type | | PCI Latency Timer | | Cache Line Size | | Y[7:0] | 00 00 00 00 |
| 0x10 | PCI Base Address 0 for MEM Mapped Config. Registers | | | | | | | Y | FFFFFFF80 | |
| 0x14 | PCI Base Address 1 for I/O Mapped Config. Registers | | | | | | | Y | FFFFFFF81 | |
| 0x18 | PCI Base Address 2 for Local Address Space 0 | | | | | | | Y | FFFFFFE00 | |
| 0x1C | PCI Base Address 3 for Local Address Space 1 | | | | | | | Y | FFFFFFC0 | |
| 0x20 | PCI Base Address 4 for Local Address Space 2 | | | | | | | Y | FFFFFFC0 | |
| 0x24 | PCI Base Address 5 for Local Address Space 3 | | | | | | | Y | 00000000 | |
| 0x28 | PCI CardBus Information Structure Pointer | | | | | | | N | 00000000 | |
| 0x2C | Subsystem ID | | | | Subsystem Vendor ID | | | N | 000A 1498 | |
| 0x30 | PCI Base Address for Local Expansion ROM | | | | | | | Y | 00000000 | |
| 0x34 | Reserved | | | | | New Cap. Ptr. | | N | 000000 40 | |
| 0x38 | Reserved | | | | | | | N | 00000000 | |
| 0x3C | Max_Lat | | Min_Gnt | | Interrupt Pin | | Interrupt Line | | Y[7:0] | 00 00 01 00 |
| 0x40 | PM Cap. | | | | PM Nxt Cap. | | PM Cap. ID | | N | 4801 00 01 |
| 0x44 | PM Data | | PM CSR EXT | | PM CSR | | | Y | 00 00 0000 | |
| 0x48 | Reserved | | HS CSR | | HS Nxt Cap. | | HS Cap. ID | | Y[23:16] | 00 00 00 00 |
| 0x4C | VPD Address | | | | VPD Nxt Cap. | | VPD Cap. ID | | Y[31:16] | 0000 00 03 |
| 0x50 | VPD Data | | | | | | | Y | 00000000 | |

Table 4-1 : PCI9030 Header

4.1.2 PCI Base Address Initialization

PCI Base Address Initialization is scope of the PCI host software.

PCI9030 PCI Base Address Initialization:

1. Write 0xFFFF_FFFF to the PCI9030 PCI Base Address Register.
2. Read back the PCI9030 PCI Base Address Register.
3. For PCI Base Address Registers 0:5, check bit 0 for PCI Address Space.
 - Bit 0 = '0' requires PCI Memory Space mapping
 - Bit 0 = '1' requires PCI I/O Space mapping

For the PCI Expansion ROM Base Address Register, check bit 0 for usage.

 - Bit 0 = '0': Expansion ROM not used
 - Bit 0 = '1': Expansion ROM used
4. For PCI I/O Space mapping, starting at bit location 2, the first bit set determines the size of the required PCI I/O Space size.

For PCI Memory Space mapping, starting at bit location 4, the first bit set to '1' determines the size of the required PCI Memory Space size.

For PCI Expansion ROM mapping, starting at bit location 11, the first bit set to '1' determines the required PCI Expansion ROM size.

For example, if bit 5 of a PCI Base Address Register is detected as the first bit set to '1', the PCI9030 is requesting a 32 byte space (address bits 4:0 are not part of base address decoding).
5. Determine the base address and write the base address to the PCI9030 PCI Base Address Register. For PCI Memory Space mapping the mapped address region must comply with the definition of bits 3:1 of the PCI9030 PCI Base Address Register.

After programming the PCI9030 PCI Base Address Registers, the software must enable the PCI9030 for PCI I/O and/or PCI Memory Space access in the PCI9030 PCI Command Register (Offset 0x04). To enable PCI I/O Space access to the PCI9030, set bit 0 to '1'. To enable PCI Memory Space access to the PCI9030, set bit 1 to '1'.

| Offset in Config. | Description | Usage |
|-------------------|-----------------------|----------|
| 0x10 | PCI9030 LCR's MEM | Used |
| 0x14 | PCI9030 LCR's I/O | Used |
| 0x18 | PCI9030 Local Space 0 | Used |
| 0x1C | PCI9030 Local Space 1 | Used |
| 0x20 | PCI9030 Local Space 2 | Used |
| 0x24 | PCI9030 Local Space 3 | Not used |
| 0x30 | Expansion ROM | Not used |

Table 4-2 : PCI9030 PCI Base Address Usage

4.2 Local Configuration Register (LCR)

After reset, the PCI9030 Local Configuration Registers are loaded from the on board serial configuration EEPROM.

The PCI base address for the PCI9030 Local Configuration Registers is PCI9030 PCI Base Address 0 (PCI Memory Space) (Offset 0x10 in the PCI9030 PCI Configuration Register Space) or PCI9030 PCI Base Address 1 (PCI I/O Space) (Offset 0x14 in the PCI9030 PCI Configuration Register Space).

Do not change hardware dependent bit settings in the PCI9030 Local Configuration Registers.

| Offset from PCI Base Address | Register | Value | Description |
|------------------------------|--------------------------------------|-------------|------------------------------|
| 0x00 | Local Address Space 0 Range | 0x0FFF_FE00 | 512 Bytes Memory Space |
| 0x04 | Local Address Space 1 Range | 0x0FFF_FFC0 | 64 Bytes Memory Space |
| 0x08 | Local Address Space 2 Range | 0x0FFF_FFC0 | 64 Bytes Memory Space |
| 0x0C | Local Address Space 3 Range | 0x0000_0000 | Not used |
| 0x10 | Local Exp. ROM Range | 0x0000_0000 | Not used |
| 0x14 | Local Re-map Register Space 0 | 0x0000_0001 | Enabled, Base Address 0x0000 |
| 0x18 | Local Re-map Register Space 1 | 0x0000_0201 | Enabled, Base Address 0x0200 |
| 0x1C | Local Re-map Register Space 2 | 0x0000_0301 | Enabled, Base Address 0x0300 |
| 0x20 | Local Re-map Register Space 3 | 0x0000_0000 | Not used |
| 0x24 | Local Re-map Register ROM | 0x0000_0000 | Not used |
| 0x28 | Local Address Space 0 Descriptor | 0x0180_6022 | Local Space 0 Configuration |
| 0x2C | Local Address Space 1 Descriptor | 0x0140_0022 | Local Space 1 Configuration |
| 0x30 | Local Address Space 2 Descriptor | 0x0140_0022 | Local Space 2 Configuration |
| 0x34 | Local Address Space 3 Descriptor | 0x0000_0000 | Not used |
| 0x38 | Local Exp. ROM Descriptor | 0x0000_0000 | Not used |
| 0x3C | Chip Select 0 Base Address | 0x0000_0101 | Chip Select Local Space 0 |
| 0x40 | Chip Select 1 Base Address | 0x0000_0221 | Chip Select Local Space 1 |
| 0x44 | Chip Select 2 Base Address | 0x0000_0321 | Chip Select Local Space 2 |
| 0x48 | Chip Select 3 Base Address | 0x0000_0000 | Not used |
| 0x4C | Interrupt Control/Status | 0x0043 | Local IRQ1 & PCI IRQ enabled |
| 0x4E | EEPROM Write Protect Boundary | 0x0030 | Standard write protection |
| 0x50 | Miscellaneous Control Register | 0x0078_0000 | Retry delay = max |
| 0x54 | General Purpose I/O Control | 0x026D_B6DB | No GPIO |
| 0x70 | Hidden1 Power Management data select | 0x0000_0000 | Not used |
| 0x74 | Hidden 2 Power Management data scale | 0x0000_0000 | Not used |

Table 4-3 : PCI9030 Local Configuration Register

4.3 Configuration EEPROM

After power-on or PCI reset, the PCI9030 loads initial configuration register data from the on board configuration EEPROM.

The configuration EEPROM contains the following configuration data:

- Address 0x00 to 0x27 : PCI9030 PCI Configuration Register Values
- Address 0x28 to 0x87 : PCI9030 Local Configuration Register Values
- Address 0x88 to 0xFF : Reserved

See the PCI9030 Manual for more information.

| Address | Offset | | | | | | | |
|---------|--------|--------|--------|--------|--------|--------|--------|--------|
| | 0x00 | 0x02 | 0x04 | 0x06 | 0x08 | 0x0A | 0x0C | 0x0E |
| 0x00 | 0x0353 | 0x1498 | 0x0280 | 0x0000 | 0x1180 | 0x0000 | s.b. | 0x1498 |
| 0x10 | 0x0000 | 0x0040 | 0x0000 | 0x0100 | 0x4801 | 0x4801 | 0x0000 | 0x0000 |
| 0x20 | 0x0000 | 0x4C06 | 0x0000 | 0x0003 | 0x0FFF | 0xFE00 | 0x0FFF | 0xFFC0 |
| 0x30 | 0x0FFF | 0xFFC0 | 0x0000 | 0x0000 | 0x0000 | 0x0000 | 0x0000 | 0x0001 |
| 0x40 | 0x0000 | 0x0201 | 0x0000 | 0x0301 | 0x0000 | 0x0000 | 0x0000 | 0x0000 |
| 0x50 | 0x0180 | 0x0022 | 0x0140 | 0x0022 | 0x0140 | 0x0022 | 0x0000 | 0x0000 |
| 0x60 | 0x0000 | 0x0000 | 0x0000 | 0x0101 | 0x0000 | 0x0221 | 0x0000 | 0x0321 |
| 0x70 | 0x0000 | 0x0000 | 0x0030 | 0x0041 | 0x0078 | 0x0000 | 0x026D | 0xB6D2 |
| 0x80 | 0x0000 | 0x0000 | 0x0000 | 0x0000 | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF |
| 0x90 | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF |
| 0xA0 | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF |
| 0xB0 | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF |
| 0xC0 | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF |
| 0xD0 | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF |
| 0xE0 | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF |
| 0xF0 | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF |

Table 4-4 : Configuration EEPROM TPMC851-10R

Subsystem-ID Value (Offset 0x0C): TPMC851-10R 0x000A

4.4 Local Software Reset

The PCI9030 Local Reset Output LRESETo# is used to reset the on board local logic.

The PCI9030 local reset is active during PCI reset or if the PCI Adapter Software Reset bit is set in the PCI9030 local configuration register CNTRL (offset 0x50).

CNTRL[30] PCI Adapter Software Reset:

Value of '1' resets the PCI9030 and issues a reset to the Local Bus (LRESETo# asserted). The PCI9030 remains in this reset condition until the PCI Host clears this bit. The contents of the PCI9030 PCI and Local Configuration Registers are not reset. The PCI9030 PCI Interface is not reset.

5 Configuration Hints

5.1 Big / Little Endian

- PCI – Bus (Little Endian)

| | |
|--------|------------|
| Byte 0 | AD[7..0] |
| Byte 1 | AD[15..8] |
| Byte 2 | AD[23..16] |
| Byte 3 | AD[31..24] |

- Every Local Address Space (0...3) and the Expansion ROM Space can be programmed to operate in Big or Little Endian Mode.

| Big Endian | | Little Endian | |
|--------------------------|-----------|---------------|-----------|
| 32 Bit | | 32 Bit | |
| Byte 0 | D[31..24] | Byte 0 | D[7..0] |
| Byte 1 | D[23..16] | Byte 1 | D[15..8] |
| Byte 2 | D[15..8] | Byte 2 | D[23..16] |
| Byte 3 | D[7..0] | Byte 3 | D[31..24] |
| 16 Bit upper lane | | 16 Bit | |
| Byte 0 | D[31..24] | Byte 0 | D[7..0] |
| Byte 1 | D[23..16] | Byte 1 | D[15..8] |
| 16 Bit lower lane | | | |
| Byte 0 | D[15..8] | | |
| Byte 1 | D[7..0] | | |
| 8 Bit upper lane | | 8 Bit | |
| Byte 0 | D[31..24] | Byte 0 | D[7..0] |
| 8 Bit lower lane | | | |
| Byte 0 | D[7..0] | | |

Table 5-1 : Local Bus Little/Big Endian

Standard use of the TPMC851:

Local Address Space 0 32 bit bus in Big Endian Mode

Local Address Space 1 16 bit bus in Big Endian Mode

Local Address Space 2 16 bit bus in Big Endian Mode

Local Address Space 3 not used

Expansion ROM Space not used

To change the Endian Mode use the Local Configuration Registers for the corresponding Space. Bit 24 of the according register sets the mode. A value of 1 indicates Big Endian and a value of 0 indicates Little Endian.

Use the PCI Base Address 0 + Offset or PCI Base Address 1 + Offset:

| Short cut | Offset | Name |
|-----------|--------|---|
| LAS0BRD | 0x28 | Local Address Space 0 Bus Region Description Register |
| LAS1BRD | 0x2C | Local Address Space 1 Bus Region Description Register |
| LAS2BRD | 0x30 | Local Address Space 2 Bus Region Description Register |
| LAS3BRD | 0x34 | Local Address Space 3 Bus Region Description Register |
| EROMBRD | 0x38 | Expansion ROM Bus Region Description Register |

You could also use the PCI - Base Address 1 I/O Mapped Configuration Registers.

6 Programming Hints

6.1 Data Correction

There are two errors which affect the DC accuracy of the ADC and the DACs.

- Offset Error:

ADC: The data value when converting with the input connected to its own ground in single-ended mode, or with shorted inputs in differential mode. This error is corrected by subtracting the known error from the readings.

DAC: The data value that is required to get a zero voltage output signal. This error is corrected by subtracting the known error from the data value.

- Gain Error:

ADC: The difference between the ideal gain and the actual gain of the programmable gain amplifier and the ADC. It is corrected by multiplying the reading with a correction factor.

DAC: The difference between the ideal gain and the actual gain of the DAC. It is corrected by multiplying the data value with a correction factor.

The data correction values are obtained during factory calibration and are stored in the Calibration Data ROM.

6.1.1 ADC Correction Formula

Please use the total 16 bit data register value for the ADC correction formula.

The basic formula for correcting any ADC reading for the TPMC851 (bipolar input voltage range) is:

$$Value = Reading \cdot \left(1 - \frac{Gain_{corr}}{131072} \right) - \frac{Offset_{corr}}{4}$$

Value is the corrected result.

Reading is the data read from the ADC Data Register.

Gain_{corr} and *Offset_{corr}* are the ADC correction factors from the Calibration Data ROM stored for each gain factor and are the same for all channel 1-32.

The correction values are stored as two's complement 16 bit values in the range -32768 to 32767. For higher accuracy they are scaled to ¼ LSB.

Floating point arithmetic or scaled integer arithmetic is necessary to avoid rounding error while computing above formula.

6.1.2 DAC Correction Formula

The basic formula for correcting any DAC value is:

$$Data = Value \cdot \left(1 - \frac{Gain_{corr}}{131072} \right) - \frac{Offset_{corr}}{4}$$

Value is the desired DAC value.

Data is the corrected DAC value that must be sending to the DAC.

Gain_{corr} and *Offset_{corr}* are the DAC correction values from the Calibration Data ROM. They are stored separately for each of the 8 DAC channels.

The correction values are stored as two's complement byte wide values in the range from -32768 to +32767. For higher accuracy they are scaled to ¼ LSB.

Floating point arithmetic or scaled integer arithmetic must be used to avoid rounding errors in computing above formula.

6.2 ADC Operating Modes

The ADC part of the TPMC851 can operate in two modes: the Manual Mode, with little or none support through automation, and the Sequencer Mode, with large support through automation.

- Manual Mode

In this mode, the converter operation relies on the user. The channel and gain are set by the user, and the user has large influence on the converter operation.

Use this mode to convert specific channels and to control conversion timing or to read a channel repeatedly without the need to await the settling time.

- Sequencer Mode

In this mode almost everything is automated and the converter operation is transparent to the user.

Use this mode to convert all channels at specific time intervals, or to always have current data available.

6.2.1 Manual Mode

The Manual Mode is useful if direct control of converter operation is needed. Setup the desired channel and gain in the ADCCONT register. If the "Automatic Settling Time Control" is deactivated, the user has to wait until the ADC_SETTL flag reads '0'. Then the conversion can be started with a write access to the ADC Conversion Start Register ADCCONV. If the Automatic Settling Time Control is activated, the conversion starts automatically after the settling time has elapsed.

It is possible to select the next channel and/or gain in the ADCCONT register immediately after the write to the ADC Conversion Start Register ADCCONV. The conversion and the settling time will then proceed simultaneously.

The conversion data is available in the ADCDATA register when the ADC_BUSY flag in the ADCSTAT register reads as '0'.

If interrupts are enabled, two interrupts will be generated: the first interrupt is generated when the settling time has elapsed (and the Automatic Settling Time Control is deactivated); the second interrupt is generated when the conversion has finished. Using the interrupts exempts from polling the ADC_SETTL and ADC_BUSY flags.

Without Automatic Settling Time Control:

- Setup the conversion in the ADC Control Register ADCCONT
- Poll for SETTL_BUSY flag
- After settling time has elapsed, write to the ADC Conversion Start Register ADCCONV to start conversion
- Poll for ADC_BUSY flag
- After conversion time has elapsed, read conversion data in the ADC Data Register ADCDATA

With Automatic Settling Time Control:

- Setup the conversion in the ADC Control Register ADCCONT
- Poll for SETTL_BUSY flag
- Poll for ADC_BUSY flag
- After conversion time has elapsed, read conversion data in the ADC Data Register ADCDATA

6.2.2 Sequencer Mode

The Sequencer Mode is very useful for periodic measurements or to always provide actual conversion data. The sequencer converts all enabled ADC channels and stores the results in the Sequencer Data RAM. After a programmable time the sequencer repeats the sequence.

To use the sequencer, all channels must be configured for the sequence in the Sequencer Instruction RAM. In the Sequencer Instruction RAM the channels are enabled for the sequence, and the gain and the mode (single-ended or differential) are selected.

Once the sequencer is started, all enabled channels are converted and the results are stored in the Sequencer Data RAM. When the last sequencer instruction has been completed, the Data Available Flag DATA_AV in the Sequencer Status Register (ADCSEQSTAT) is set to '1' and, if enabled, an interrupt request will be asserted. The user can now read the ADC data from the Sequencer Data RAM (note: only the enabled channels are updated; Channels not enabled in the Sequencer Instruction RAM are not updated and may contain invalid data from former conversions). After that, the DATA_AV flag must be cleared by writing a '1' to the Sequencer Status Register ADCSEQSTAT bit 0.

The repeat frequency of the sequencer can be programmed in the Sequencer Timer Register. The Sequencer Timer is programmable from 100µs to 6.5535s in steps of 100µs. Whenever the timer reaches the programmed value, the sequencer starts a new sequence.

A special function is the Sequencer Continuous Mode. It is activated if the Sequencer Timer Register is set to 0x0000. In this mode the sequencer immediately starts a new sequence when the actual sequence has been completed. If the sequencer is in Sequencer Continuous Mode, the user can read valid data from the Sequencer Data RAM at any time. The Sequencer Data RAM locations of the enabled ADC channels are updated with every sequence.

If the Sequencer Continuous Mode is active, the Timer Error Flag (TIMER_ERROR) and the Data Overflow Error Flag (DATA_OVERFLOW_ERROR) are not active and will read as '0'. Clearing the Data Available Flag DATA_AV is not necessary in the Sequencer Continuous Mode (it will not produce an interrupt), but it may be useful in order to monitor the completion of the sequences.

The sequencer may also be triggered by an external signal instead of the timer. In this case the Timer Error Flag (TIMER_ERROR) is not active. A new external trigger is accepted only after a sequence has completed. An external trigger while the sequencer runs is ignored.

The update rate depends on the number of enabled channels:

Update Rate = 16µs · number of enabled channels

6.2.2.1 Sequencer Errors

If the sequencer detects an error, it will stop after the last instruction and sets the corresponding error flag in the Sequencer Status Register SEQSTAT.

| Error | Description | Sequencer Action | User Action |
|-----------------------|---|---|---|
| Data Overflow Error | Error occurs if the sequencer has new data to store but the user has not yet acknowledged that the data from the previous sequence has been read out. (not in Sequencer Continuous Mode) | Sequencer stops after the last instruction is done. Data Overflow Error Flag is set. If enabled, an interrupt request will be asserted. | Write a '1' to the Sequencer Status Register bit 1. Make sure the Sequencer Data is read and acknowledged within the programmed Sequence Time. Start Sequencer again. |
| Timer Error | Error occurs if the programmed Sequencer Time is shorter than the sequence itself. (not in Sequencer Continuous Mode) | Sequencer stops after the last instruction is done. Timer Error Flag is set. If enabled, an interrupt request will be asserted. | Write a '1' to the Sequencer Status Register bit 2. Program a larger Sequence Time. Start Sequencer again. |
| Instruction RAM Error | Error occurs if the no channel is enabled for the sequence (bit 1 of Sequencer Instruction RAM byte) and the Sequencer is started. | Sequencer stops after the last instruction is done. Instruction RAM Error Flag is set. If it is enabled, an interrupt request will be asserted. | Write a '1' to the Sequencer Status Register bit 3. Correct the Sequencer Instruction RAM setting. Start Sequencer again. |

Table 6-1 : Sequencer Errors

If the Sequence Timer Register is set to 0x0000 (Sequencer Continuous Mode) the sequencer ignores the data overflow. The Data Overflow Error Flag is always read as '0' in this mode.

6.2.3 Application Examples

The following examples all use interrupts. The use of interrupts can be replaced by polling the according status flags in the ADC Status Register or the Sequencer Status Register.

6.2.3.1 Fastest Conversion of an arbitrary Single Channel

- Program the desired channel and gain in the ADC Control Register ADCCONT. Activate the Automatic Settling Time Control (ASTC = '1') and the IRQ after Conversion (IRQC = '1')
- The channel is now converted without any further user action. After completion of the conversion an interrupt is issued which signals that the conversion data is available in the ADCDATA Register
- Acknowledge the Interrupt in the ADC Status Register ADCSTAT (ADC_READY = '1') and read ADCDATA

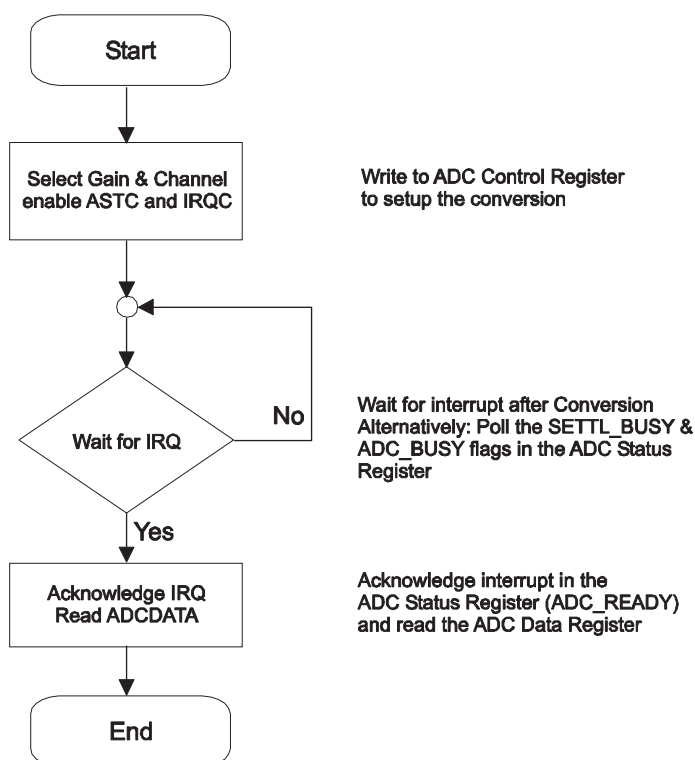


Figure 6-1 : Flow: Fastest conversion of an arbitrary single channel

Conversion time is approx. 17.25 μ s.

6.2.3.2 Fastest Conversion of a specific Single Channel

- Program the ADC Control Register with desired channel and gain and activate the IRQ after Settling Time (IRQS = '1') and the IRQ after Conversion (IRQC = '1')
- When the IRQ after Settling Time is issued, the channel is ready for conversion. Write to the ADC Conversion Start Register ADCCONV and acknowledge the Interrupt in the ADC Status Register ADCSTAT (SETTL_IRQ = '1')

- After completion of the conversion an IRQ after Conversion is issued which signals that the conversion data is available in the ADCDATA Register
- Write again to the ADC Conversion Start Register ADCCONV to start the next conversion of this channel, acknowledge the Interrupt in the ADC Status Register ADCSTAT (ADC_IRQ = '1') and read ADCDATA.

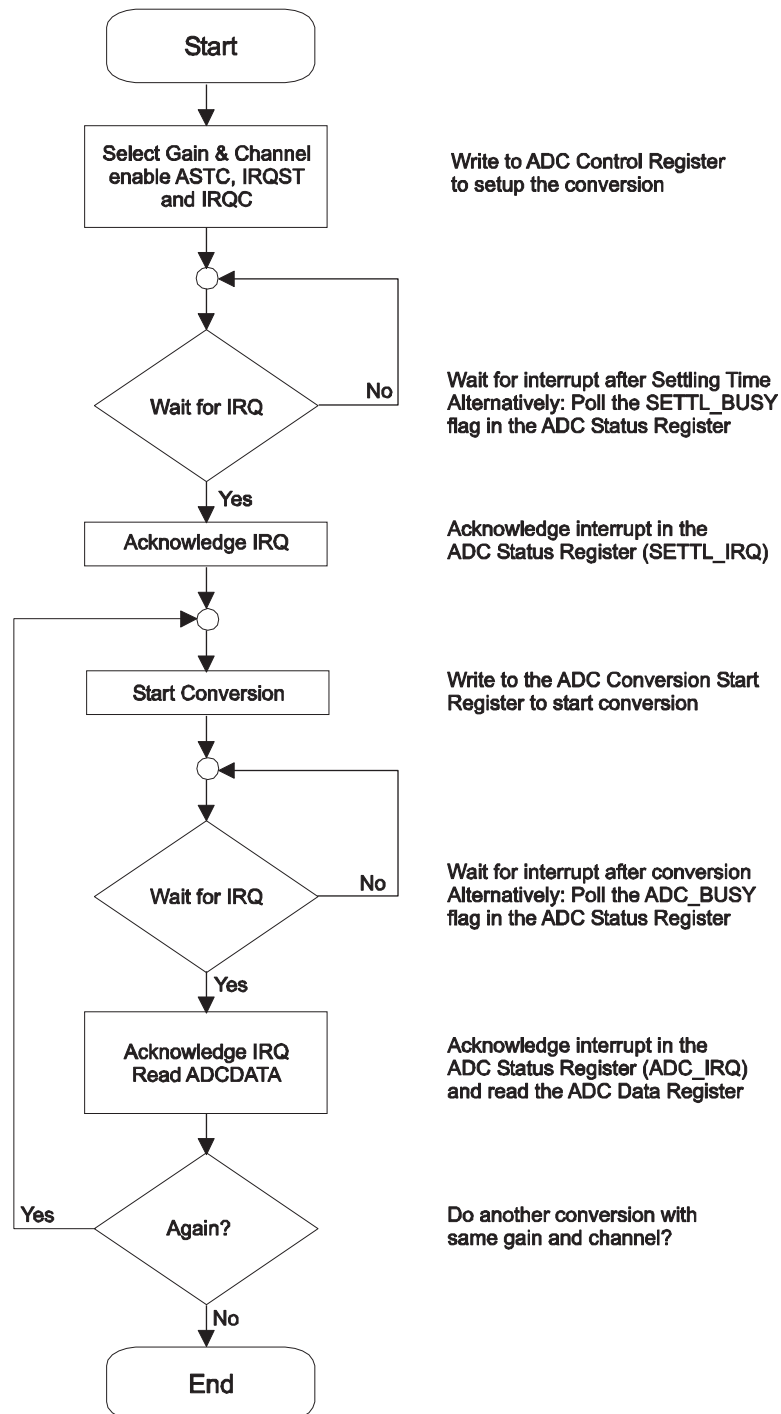


Figure 6-2 : Flow: Fastest conversion of a specific single channel

Conversion time is approx 1.25µs, as long as neither the channel nor the gain is changed.

6.2.3.3 Periodic Conversion of multiple Channels

- Activate the channels to be converted and program the gain in the Sequencer Instruction RAM
- Set the sequencer period in the ADC Sequencer Timer Register ADCSEQTIMER
- Enable the Sequencer Interrupt SEQ_INT_ENA and start the Sequencer in the Sequencer Control Register ADCSEQCONT
- After completion of the sequence a Sequencer Interrupt is issued which signals that the conversion data is available in the Sequencer Data RAM
- Acknowledge the Interrupt and clear the DATA_AV flag in the ADC Sequencer Status Register ADCSEQSTAT (SEQ_IRQ = '1') and read the Sequencer Data RAM

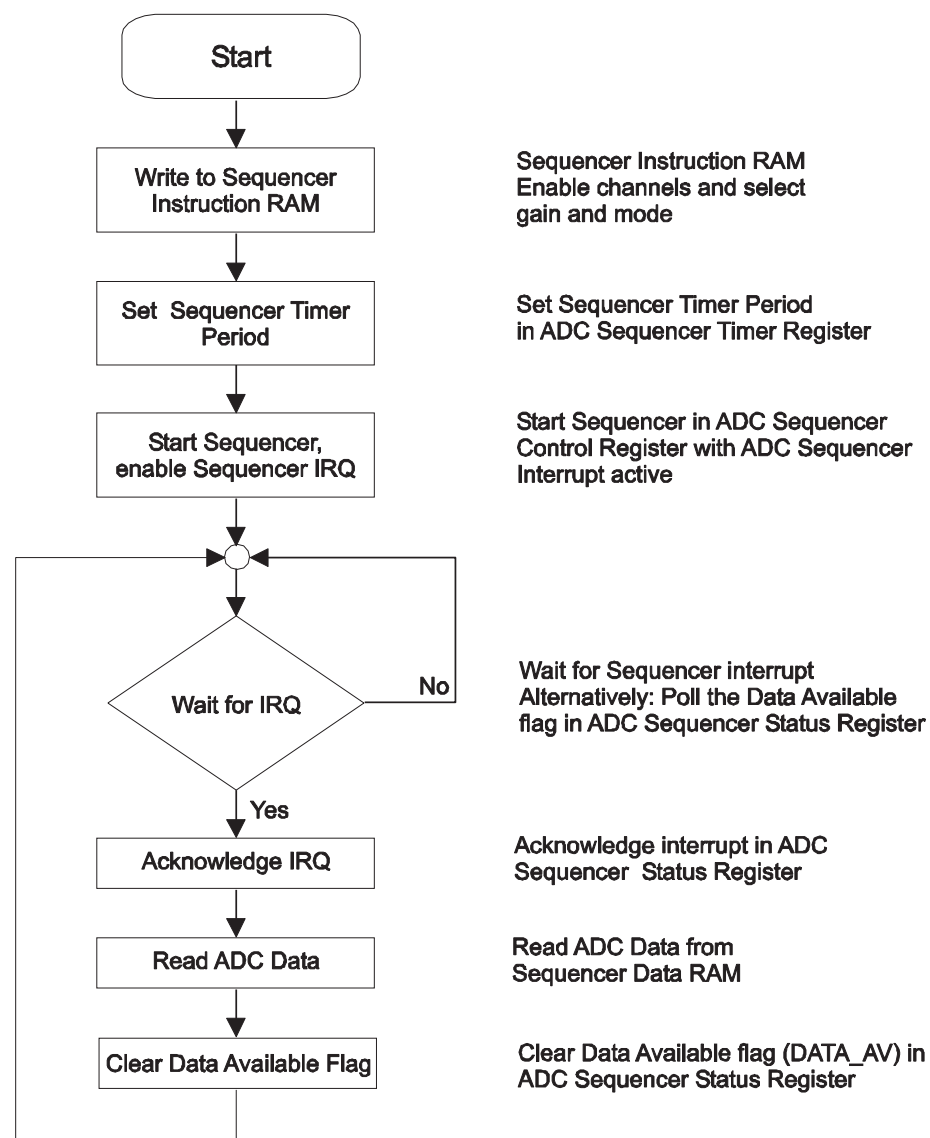


Figure 6-3 : Flow: Periodic conversion of multiple channels

6.2.3.4 Continuous Conversion of multiple Channels

- Activate the channels to be converted and program the gain and mode of these channels in the Sequencer Instruction RAM
- Set the Sequencer Timer Register ADCSEQTIMER to 0x0000
- Start the Sequencer in the Sequencer Control Register ADCSEQCONT
- Read the data from the Sequencer Data RAM as needed.

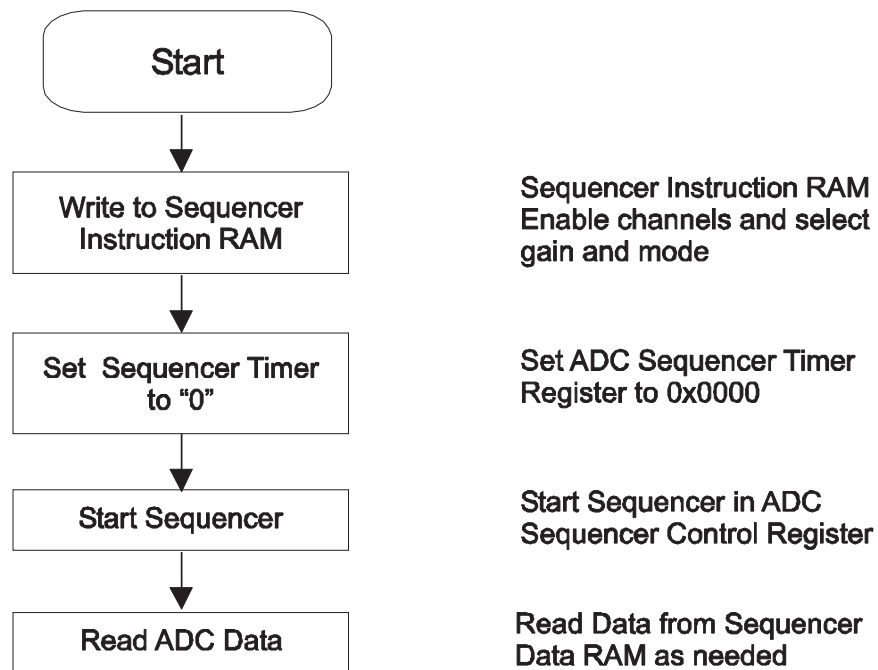


Figure 6-4 : Flow: Continuous conversion of multiple channels

6.3 DAC Operating Modes

The DAC part of the TPMC851 can operate in two modes – Immediate Update or Simultaneous Update. These modes are configured with the Load Mode Select (LOADSEL) bits in the DAC Control Register DACCONT.

| LOADSEL[1:0] | Load Mode |
|---------------------|---|
| Immediate Update | |
| 00 | Immediate Update: The DAC outputs are updated immediately on a write access to a DAC Data Register |
| Simultaneous Update | |
| 01 | Simultaneous Update: All DAC outputs are updated simultaneously when LOADDAC is set to '1' |
| 10 | Sequencer Update: All DAC outputs are updated simultaneously when the DAC Sequencer Timer reaches zero |
| 11 | Trigger Update: All DAC outputs are updated simultaneously by an external trigger (Dig I/O Line 1) |

Table 6-2 : DAC Load Modes

6.3.1 Immediate Update

The DACs are updated immediately on a write access to a DAC Data Register. Use this mode if single channels need to be updated on an irregular base.

6.3.2 Simultaneous Update

Write accesses to the DAC Data Registers don't immediately update the DACs. Instead this data is buffered until a trigger event. This trigger event simultaneously updates all DACs with the buffered data. The Simultaneous Update Mode divides into three sub modes. These modes vary in the source for the trigger event:

- Simultaneous Update: The DACs are updated simultaneously when LOADDAC in the DAC Control Register DACCONT is set to '1'. Use this mode if all channels need to be updated on an irregular base.
- Sequencer Update: All DAC outputs are updated simultaneously when the DAC Sequencer Timer reaches zero. Use this mode if all channels need to be updated on a regular base.
- Trigger Update: All DAC outputs are updated simultaneously by an external trigger (Dig I/O Line 1). Use this mode if all channels need to be updated on an external event.

Before a simultaneous update all DACs should be loaded with new data via the DAC Data Registers. If a channel was not loaded with new data, it will use the old data previously buffered.

Allow the DAC outputs to settle for approx. 10µs before performing a new DAC update.

6.4 Digital TTL I/O

Note that some digital inputs are internally used in the other parts of the TPMC851:

TTL I/O line 0 as ADC sequencer trigger

TTL I/O line 1 as DAC sequencer trigger

TTL I/O line 2 and TTL I/O line 3 as counter inputs

TTL I/O line 4 as counter control input

Normally the usage of these digital I/O lines is not subject to any restrictions, as long as they are used as inputs (i.e. as trigger for a sequencer). Exceptions are I/O lines 0 and 1, these can also be used as sequencer trigger output.

If ADC_OUT in the Line Direction Register LINEDIR is set to '1', the trigger signal for the ADC sequencer is available for external use on Dig I/O Line 0.

If DAC_OUT in the Line Direction Register LINEDIR is set to '1', the trigger signal for the DAC sequencer is available for external use on Dig I/O Line 1.

In this case the settings in the Line Direction Register for these I/O lines (LINEDIR0 resp. LINEDIR1) are overridden and they are always used as output. Writes in the Line Output Register to these I/O lines (LINEOUT0 resp. LINEOUT1) have no effect.

Avoid to use this configuration while the sequencer is set to wait for an external trigger (ADC: SEQ_TRIG = 1; DAC: LOAD_SEL = 11). This will cause the sequencer to lock.

If it is required to use the I/O line as sequencer trigger input and output at the same time, this can be achieved with following solution:

- Configure the sequencer to wait for an external trigger (ADC: SEQ_TRIG = 1; DAC: LOAD_SEL = 11)
- Configure the I/O Line as output (ADC: LINEDIR0 = 1; DAC: LINEDIR1 = 1)
- Trigger the sequencer with a write to the Line Output Register (ADC: LINEOUT0 = 1; DAC: LINEOUT1 = 1)

6.5 Counter Operating Modes

The general purpose counter of the TPMC851 offers 3 Input Modes, 2 Count Modes and 4 Control Modes.

6.5.1 Input Modes

The Input Mode determines the input source for the counter and how the counter interprets these input signals:

| Input Mode | I/O Line 2 | I/O Line 3 | I/O Line 4 |
|------------------|--------------|---------------------------|-----------------------------|
| Timer Mode | - | - | Available for Control Modes |
| Direction Count | Count | Count direction (up/down) | |
| Up/Down Count | Count UP | Count DOWN | |
| Quadrature Count | Quadrature X | Quadrature Y | |

Table 6-3 : Input Modes

Changing the input mode does not affect the counter reading. If no input mode is selected, the counter is disabled.

6.5.1.1 Timer Mode

In Timer mode the counter uses an internal clock prescaler as input:

| Bits | Prescaler | Clock frequency |
|------|-----------|-----------------|
| 00 | 1x | 40 MHz |
| 01 | 2x | 20 MHz |
| 10 | 4x | 10 MHz |
| 11 | 8x | 5 MHz |

Table 6-4 : Clock Prescaler

6.5.1.2 Direction Count

The counter acts as up/down counter. Counting pulses are generated when a transition from low to high of the I/O Line 2 is detected. The I/O Line 3 determines the count direction. I/O Line 3 = '1' is up, I/O Line 3 = '0' is down.

| I/O Line 3 | Count Direction |
|------------|-----------------|
| 0 | Down |
| 1 | Up |

Table 6-5 : Count Directions

6.5.1.3 Up/Down Count

The counter acts as up/down counter. Counting pulses are generated when a transition from low to high of either the I/O Line 2 or the I/O Line 3 is detected. The I/O Line 2 counts up, the I/O Line 3 counts down. Simultaneous transitions on both I/O lines do not generate a counting pulse.

6.5.1.4 Quadrature Count

The counter acts as quadrature counter. The quadrature inputs can be interpreted as 1x, 2x or 4x counting. 1x lets the counter count once for each full cycle of the quadrature inputs, 2x lets the counter count once for each half cycle of the quadrature inputs and 4x lets the counter count once for each quarter cycle of the quadrature inputs. The count direction (increase or decrease) is determined by the relative phase of the Quadrature X- and Quadrature Y-signals.

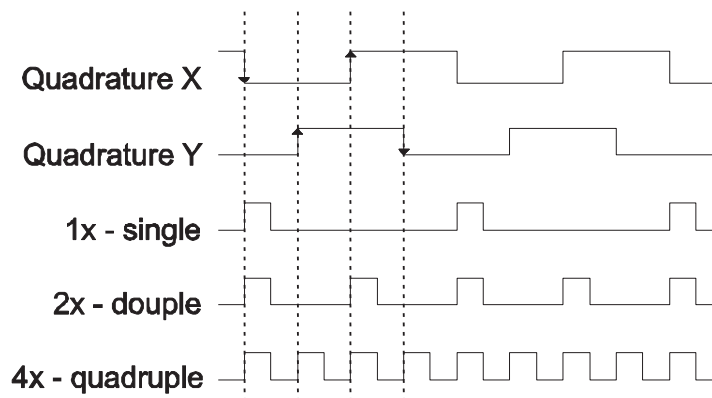


Figure 6-5: Quadrature Signals

6.5.2 Count Modes

In normal operation, the counter is a cycling counter. Additional two Count Modes are available. The Count Modes are available for every Input Mode.

6.5.2.1 Divide-by-N

The counter is enabled in the Channel Control Register and will run until disabled. The counter is loaded with the content of the preload register every time the counter creates a borrow or a carry. This mode assumes that the counter counts to one direction only.

6.5.2.2 Single Cycle

The counter is enabled in the Channel Control Register and will start on a manual preload or reset in the Channel Command Register or on a control mode event in load or reset mode.

The counter stops when it creates a borrow or a carry.

6.5.3 Control Modes

The Control Mode determines how I/O Line 4 is interpreted. A control mode event generates interrupts, if the Control Mode Interrupt is enabled in the Counter Control Register.

With the exception of the Gate Mode, all modes react on a level change on the I/O Line 4. Note that if the digital debounce filter is applied, a change in the input level is only detected when the input line is stable for duration greater than the programmed debounce duration.

The following table gives an overview of the control mode events.

| Input Control Mode | Polarity | |
|--------------------|----------------------------------|---------------------------------|
| | high active (CNTCONT[14] = 0) | low active (CNTCONT[14] = 1) |
| No Control Mode | - | - |
| Load Mode | Rising edge | Falling edge |
| Latch Mode | Rising edge | Falling edge |
| Gate Mode | High level | Low Level |
| Reset Mode | Rising edge | Falling edge |

Table 6-6 : Input Control Mode events

6.5.3.1 No Control Mode

In this mode I/O Line 4 is ignored.

6.5.3.2 Load Mode

An event on I/O Line 4 loads the counter with the content of the Counter Preload Register.

If the 'Single Cycle' mode is active, the event on I/O Line 4 starts the counter.

The counter can also be preloaded by writing '1' to the 'Load Counter' (LCNT) bit in the Channel Command Register.

6.5.3.3 Latch Mode

An event on I/O Line 4 latches the actual counter value in the Counter Data Register. It will remain latched until the Counter Data Register is read or the latch is released with the CDLT bit in the Counter Status Register.

6.5.3.4 Gate Mode

The I/O Line 4 enables or disables counting.

| I/O Line 4 | Counter |
|------------|----------|
| 0 | Disabled |
| 1 | Enabled |

Table 6-7: Gate Mode

In this mode an interrupt is generated (if enabled) if the gate is closed.

6.5.3.5 Reset Mode

An event on I/O Line 4 resets the counter.

If the 'Single Cycle' mode is active, the event on I/O Line 4 starts the counter.

The counter can also be reset by writing '1' to the Reset Counter (RCNT) bit in the Channel Command Register.

6.5.4 Configuration Examples

6.5.4.1 Watchdog Timer

The watchdog timer counts down from a programmed value until it reaches 0. The counter must be reloaded on a regular base either internal via the Counter Command Register or external via I/O Line 4. Failure to cause a reload would generate a timeout and an interrupt.

| Input Mode | Count Mode | Control Mode | Additional Configuration |
|-----------------|--------------|--------------|---|
| Timer Mode Down | Single Cycle | Load Mode | Set CNTPRL to watchdog time, CNTCMP to zero, enable counter match IRQ |

6.5.4.2 Event Counting

In this configuration, input pulses or events occurring at the input pin of the counter are counted up to a programmed count limit. Upon reaching the count limit, the counter output will generate an interrupt. To arm the counter for the next measurement, reset the counter in the Counter Command Register.

| Input Mode | Count Mode | Control Mode | Additional Configuration |
|------------|--------------|--------------|---|
| U/D Count | Single Cycle | none | Set CNTCMP to limit, enable counter match IRQ |

6.5.4.3 Input Pulse Width Measurement

For pulse-width measurement, the pulse-width being measured serves as a clock gate for an up-counter. To arm the counter for a measurement, reset the counter in the Counter Command Register. The measurement starts with the next rising edge and ends with the falling edge of the gate signal. An interrupt is generated upon completion of the pulse-width measurement. The up-counter uses the internal clock prescaler at the counter's clock pin. Use following formula to calculate the pulse-width:

Pulse Width = Counter Value / Clock Frequency

| Input Mode | Count Mode | Control Mode | Additional Configuration |
|---------------|--------------|--------------|--------------------------|
| Timer Mode Up | Single Cycle | Gate Mode | Enable control mode IRQ |

6.5.4.4 Mechanical System Positioning

If a mechanical system provides position switches, these can be used for various purposes:

Establish a known home

| Input Mode | Count Mode | Control Mode | Additional Configuration |
|------------|---------------|--------------|--------------------------|
| U/D Count | Cycle Counter | Reset Mode | Enable control mode IRQ |

Establish a known reference position

| Input Mode | Count Mode | Control Mode | Additional Configuration |
|------------|---------------|--------------|--------------------------|
| U/D Count | Cycle Counter | Load Mode | Enable control mode IRQ |

Capture a position

| Input Mode | Count Mode | Control Mode | Additional Configuration |
|------------|---------------|--------------|--------------------------|
| U/D Count | Cycle Counter | Latch Mode | Enable control mode IRQ |

6.6 Interrupts

All Interrupts are requested at the LINT1 input of the PCI9030 Target Chip.

Interrupts are generated only if the Master Interrupt Enable (MIE) is enabled in the Interrupt Control Register ICR.

6.6.1 Interrupt Sources

| IRQ | IRQ Description | Enable IRQ | Acknowledge IRQ |
|-----|-----------------|------------|-----------------|
|-----|-----------------|------------|-----------------|

| | | | |
|---------|-----------------------------------|--|---|
| IRQST | ADC settling time elapsed | ADC Control Register ADCCONT | ADC Status Register ADCSTAT |
| IRQC | ADC conversion finished | ADC Control Register ADCCONT | ADC Status Register ADCSTAT |
| SEQ IRQ | ADC sequencer interrupt | ADC Sequencer Control Register ADCSEQCONT | ADC Sequencer Status Register ADCSEQSTAT |
| DAC IRQ | DAC sequencer interrupt | DAC Control Register DACCONT | DAC Sequencer Status Register DACSEQSTAT |
| LINEIST | Digital I/O line IRQ | Line Interrupt Enable Register LINEIEN | Line Interrupt Status Register LINEIST |
| CIRQ | Counter control mode interrupt | Counter Control Register CNTCONT | Counter Status Register CNTSTAT |
| MIRQ | Counter match interrupt | Counter Control Register CNTCONT | Counter Status Register CNTSTAT |

Table 6-8 : Interrupt sources

The Interrupt Control Register can give a quick overview which interrupt source caused an interrupt.

7 Installation

7.1 ADC Input Wiring

The TPMC851 provides 32 single-ended or 16 differential multiplexed analog inputs. The desired input channel and the mode (single-ended or differential) are selected by programming the input multiplexer.

A software programmable gain amplifier with gain settings of 1, 2, 4 and 8 allows a direct connection of a wide range of sensors and instrumentation. The maximum analog input voltage range is $\pm 10V$ at a gain of 1.

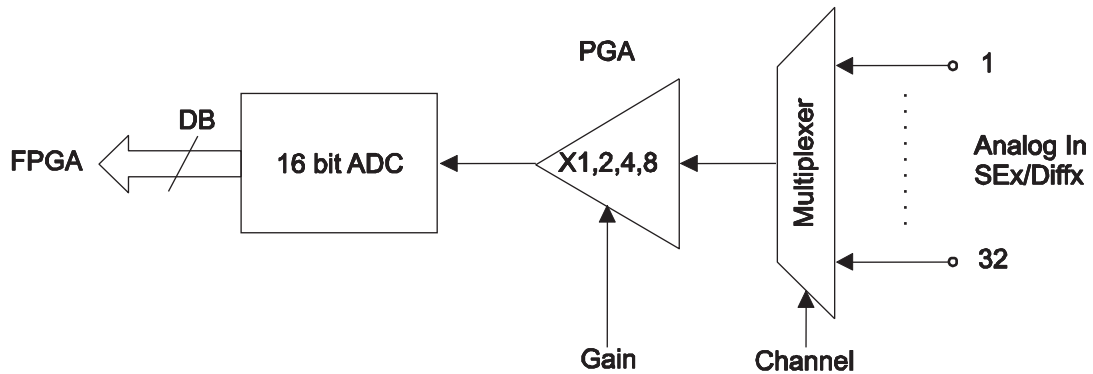


Figure 7-1 : ADC Input Wiring

Make sure that all unused analog input pins are tied to the GND signal level (or any other valid signal level within the analog input voltage range). This is required even if the unused channels are turned off by software.

If unused analog inputs are left floating, they could badly degrade the performance of the active channels.

7.2 DAC Output Wiring

The DAC is used to generate the $\pm 10V$ analog output voltage. The operational amplifier is able to drive high capacitance loads and protects the DAC.

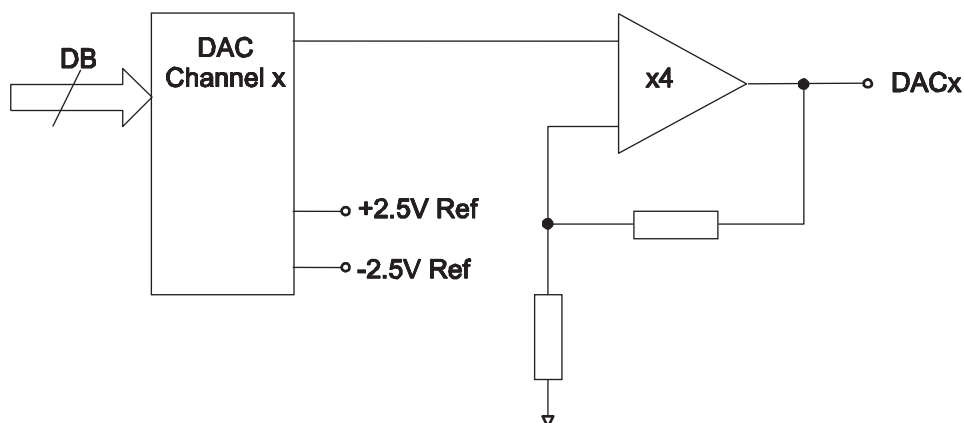


Figure 7-2 : DAC Output Wiring

7.3 TTL I/O Interface

The 16 TTL I/O lines are realized with an input and output buffer with a 4.7k Ω pull up resistor for the tri-state output function and an electronic protection array for ESD and overvoltage protection. See the following figure for more information of electrical circuitry.

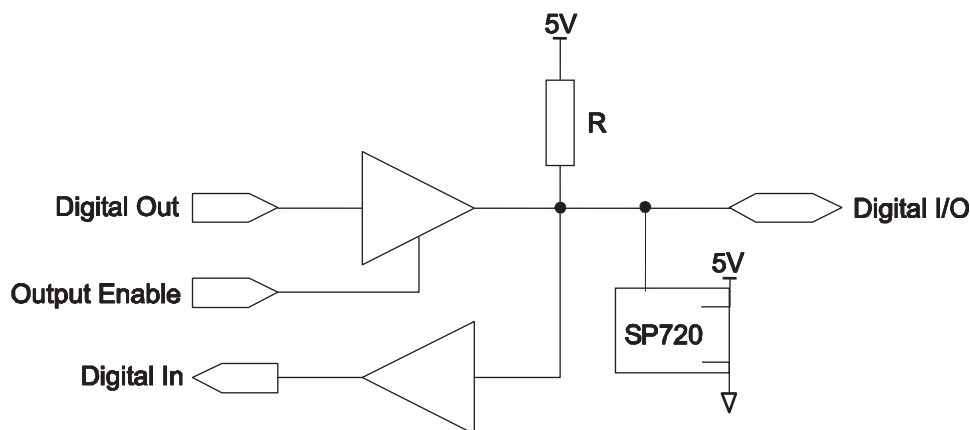


Figure 7-3 : TTL I/O Interface

While power-up and after reset all digital I/O lines are configured as inputs. Because of the pull-up resistor an unused (open) digital I/O line will be read as '1'. A digital I/O line configured as input will appear as a high state to the outside, as long it is not driven from the outside.

8 Pin Assignment – I/O Connector

Connector: HD68 SCSI-3 type female connector (AMP 787082-7)

| Pin | Signal | | Pin | Signal | |
|-----|---------------------------|--------------|-----|--------------|--------------|
| 1 | ADC SE 1 | ADC Diff 1+ | 35 | ADC SE 17 | ADC Diff 1- |
| 2 | ADC SE 2 | ADC Diff 2+ | 36 | ADC SE 18 | ADC Diff 2- |
| 3 | ADC SE 3 | ADC Diff 3+ | 37 | ADC SE 19 | ADC Diff 3- |
| 4 | ADC SE 4 | ADC Diff 4+ | 38 | ADC SE 20 | ADC Diff 4- |
| 5 | ADC SE 5 | ADC Diff 5+ | 39 | ADC SE 21 | ADC Diff 5- |
| 6 | ADC SE 6 | ADC Diff 6+ | 40 | ADC SE 22 | ADC Diff 6- |
| 7 | ADC SE 7 | ADC Diff 7+ | 41 | ADC SE 23 | ADC Diff 7- |
| 8 | ADC SE 8 | ADC Diff 8+ | 42 | ADC SE 24 | ADC Diff 8- |
| 9 | GND | | 43 | GND | |
| 10 | ADC SE 9 | ADC Diff 9+ | 44 | ADC SE 25 | ADC Diff 9- |
| 11 | ADC SE 10 | ADC Diff 10+ | 45 | ADC SE 26 | ADC Diff 10- |
| 12 | ADC SE 11 | ADC Diff 11+ | 46 | ADC SE 27 | ADC Diff 11- |
| 13 | ADC SE 12 | ADC Diff 12+ | 47 | ADC SE 28 | ADC Diff 12- |
| 14 | ADC SE 13 | ADC Diff 13+ | 48 | ADC SE 29 | ADC Diff 13- |
| 15 | ADC SE 14 | ADC Diff 14+ | 49 | ADC SE 30 | ADC Diff 14- |
| 16 | ADC SE 15 | ADC Diff 15+ | 50 | ADC SE 31 | ADC Diff 15- |
| 17 | ADC SE 16 | ADC Diff 16+ | 51 | ADC SE 32 | ADC Diff 16- |
| 18 | GND | | 52 | GND | |
| 19 | DAC 1 Output | | 53 | DAC 5 Output | |
| 20 | DAC 2 Output | | 54 | DAC 6 Output | |
| 21 | GND | | 55 | GND | |
| 22 | DAC 3 Output | | 56 | DAC 7 Output | |
| 23 | DAC 4 Output | | 57 | DAC 8 Output | |
| 24 | GND | | 58 | GND | |
| 25 | TTL I/O 0 (ADC Seq. Trg.) | | 59 | TTL I/O 8 | |
| 26 | TTL I/O 1 (DAC Seq. Trg.) | | 60 | TTL I/O 9 | |
| 27 | TTL I/O 2 (Counter input) | | 61 | TTL I/O 10 | |
| 28 | TTL I/O 3 (Counter input) | | 62 | TTL I/O 11 | |
| 29 | GND | | 63 | GND | |
| 30 | TTL I/O 4 (Counter input) | | 64 | TTL I/O 12 | |
| 31 | TTL I/O 5 | | 65 | TTL I/O 13 | |
| 32 | TTL I/O 6 | | 66 | TTL I/O 14 | |
| 33 | TTL I/O 7 | | 67 | TTL I/O 15 | |
| 34 | GND | | 68 | GND | |

Table 8-1 : Pin Assignment I/O Connector

9 Important Notes

9.1 Dummy Conversions after Power-up

After power-up the ADC's logic will be in a random state and may not perform correctly. This has two consequences:

1. The first conversion results are not valid and should be ignored.
2. The ADC starts in a mode that prevents a correct start of the sequencer.

Therefore, two dummy conversions are required after each power-up, whose results should be ignored.

Use the ADC Conversion Start Register (ADCCONV) Register to perform the dummy conversions.

If the sequencer is to be used, these two dummy conversions are absolutely necessary.

If one of TEWS TECHNOLOGIES software drivers is used, these two dummy conversions are already included.

9.2 Open Multiplexer Inputs

Unused Multiplexer inputs can pick up stray signals which are injected into the device's substrate. This turns on spurious substrate devices which badly degrade the performance of the whole multiplexer device.

Make sure that all unused analog input pins are tied to the analog ground signal level (or any other valid signal level within the analog input voltage range). This is required even if the unused channels are turned off by software.