

ASD5020 High Speed Mode

Multi-Mode 12-bit 640 MSPS / 8-bit 1000 MSPS Analog to Digital Converter

Features

- 12-bit Modes
 - Single Channel Mode: $F_{Smax} = 640$ MSPS
 - Dual Channel Mode: $F_{Smax} = 320$ MSPS
 - Quad Channel Mode: $F_{Smax} = 160$ MSPS
 - SNR: 71 dB, SFDR: 65 dB
- 8-bit Modes
 - Single Channel Mode: $F_{Smax} = 1000$ MSPS
 - Dual Channel Mode: $F_{Smax} = 500$ MSPS
 - Quad Channel Mode: $F_{Smax} = 250$ MSPS
 - SNR: 49 dB, SFDR: 65 dB
- Integrated Cross Point Switches with instantaneous switching
- Internal low jitter programmable Clock Divider
- Ultra Low Power Dissipation
 - 490mW including I/O at 640 MSPS
- 0.5 μ s start-up time from Sleep, 15 μ s from Power Down
- Internal reference circuitry with no external components required
- Coarse and fine gain control
- Digital fine gain adjustment for each ADC
- Internal offset correction
- 1.8 V supply voltage
- 1.7 - 3.6 V CMOS logic on control interface pins
- Serial LVDS/RSDS output
 - 12, 14, 16 and Dual 8-bit modes available
- 7mm x 7mm 48 QFN Package

Description

The ASD5020 is a versatile high performance low power analog-to-digital converter (ADC), utilizing time-interleaving to increase sampling rate. Integrated Cross Point Switches activate the input selected by the user.

In single channel mode, one of the four inputs can be selected as valid input to the single ADC channel. In dual channel mode, any two of the four inputs can be selected to each ADC channel. In quad channel mode, any input can be assigned to any ADC channel.

An internal, low jitter and programmable clock divider makes it possible to use a single clock source for all operational modes.

The ASD5020 is based on a proprietary structure, and employs internal reference circuitry, a serial control interface and a serial LVDS output data. Data and frame synchronization clocks are supplied for data capture at the receiver. Internal digital fine gain can be set separately for each ADC to calibrate for gain errors.

Various modes and configuration settings can be applied to the ADC through the serial control interface (SPI). Each channel can be powered down independently and output data format can be selected through this interface. A full chip idle mode can be set by a single external pin. Register settings determine the exact function of this pin.

ASD5020 is designed to interface easily with Field Programmable Gate Arrays (FPGAs) from several vendors.

Applications

- Precision Oscilloscopes
- Diversity Receivers
- Hi-End Ultrasound
- Communication Testing
- Non Destructive Testing

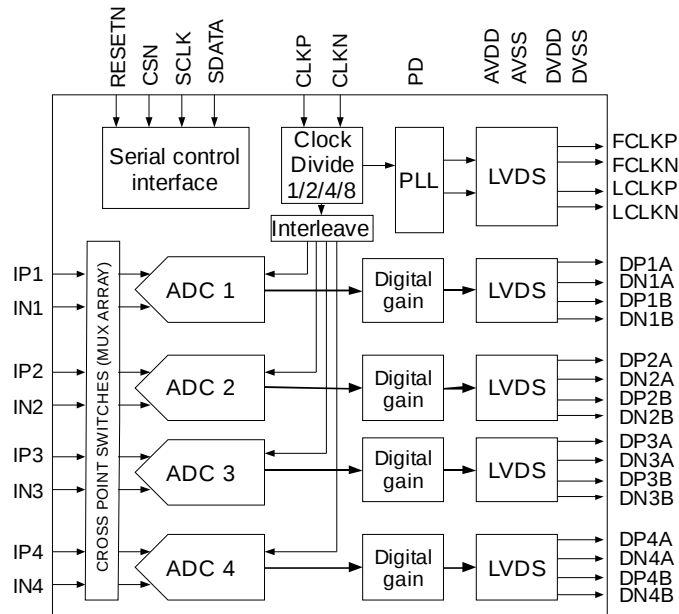


Figure 1: Functional Block Diagram



Table of Contents

| | |
|--|----|
| Blizzard Product family: Products and Relations..... | 3 |
| Specifications..... | 4 |
| ASD5020 High Speed Mode..... | 5 |
| Digital and Switching Specifications..... | 6 |
| Absolute Maximum Ratings..... | 7 |
| Pin Configuration and Description..... | 8 |
| Startup Initialization..... | 10 |
| Serial Interface..... | 10 |
| Timing Diagram..... | 10 |
| Timing Diagrams..... | 11 |
| Register Map..... | 13 |
| Register Description | 15 |
| Software Reset..... | 15 |
| Modes of Operation | 15 |
| Input Select..... | 16 |
| Full-scale Control | 17 |
| Current Control..... | 17 |
| Start-up and Clock Jitter Control..... | 19 |
| LVDS Output Configuration and Control..... | 21 |
| LVDS Drive Strength Programmability..... | 24 |
| LVDS Internal Termination Programmability..... | 24 |
| Power Mode Control | 26 |
| Programmable Gain..... | 27 |
| Analog Input Invert..... | 29 |
| LVDS Test Patterns..... | 29 |
| Theory of Operation..... | 30 |
| Interleaving Effects and Sampling Order..... | 30 |
| Recommended Usage..... | 30 |
| Analog Input..... | 30 |
| DC-Coupling..... | 31 |
| AC-Coupling..... | 31 |
| Clock Input and Jitter Considerations..... | 31 |
| Application Usage Example..... | 32 |
| Start-up Initialization..... | 32 |
| Change Mode..... | 32 |
| Select Analog Input..... | 32 |
| Package Mechanical Data..... | 33 |
| QFN48..... | 33 |
| Product Information..... | 34 |
| Ordering information..... | 34 |
| Datasheet status..... | 34 |
| Objective Product Specification:..... | 34 |
| Preliminary Product Specification:..... | 34 |
| Product Specification:..... | 34 |
| Life Support Applications :..... | 34 |



Blizzard Product family: Products and Relations

ASD5020 is a part of the ASD Blizzard family of ADCs for Instrumentation applications, with two main modes:

- High Speed Mode (ASD5020HS): 12-bit up to 640MSPS
- Precision Mode (ASD5020PM): 14-bit up to 105MSPS

The Blizzard family also includes ASD5010 with 8-bit up to 1GSPS

Blizzard ADCs are pin compatible. The products within the family can be configured with the SPI interface. The ASD5020 modes can be chosen by SPI configuration. Additionally, ASD5020 can be configured as ASD5010. ASD5010 can not be configured as ASD5020.

This relationship is shown in figure 2

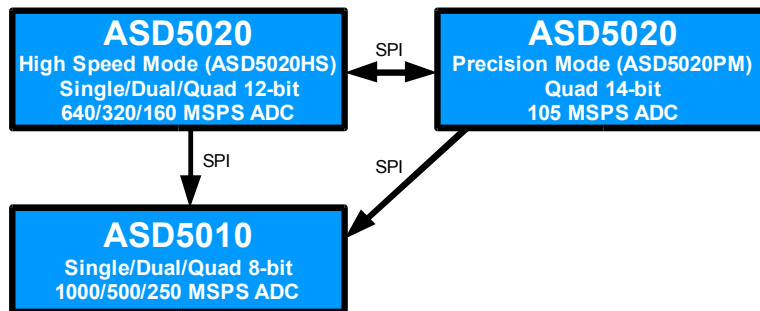


Figure 2: Blizzard Product Family



Specifications

AVDD=DVDD=OVDD=1.8V, $F_s = 160$ MSPS, Quad channel 12-bit High Speed Mode, 50% clock duty cycle, -1dBFS 70 MHz input signal, 1x/0dB digital gain (fine and coarse), unless otherwise noted

| Parameter | Description | Min | Typ | Max | Unit |
|---------------------|--|----------------|--------------|----------------|-----------------|
| DC accuracy | | | | | |
| No missing codes | | Guaranteed | | | |
| Offset | Offset error after internal digital offset correction | | 1 | | LSB |
| G_{abs} | Gain error | | | ± 6 | %FS |
| G_{rel} | Gain matching between channels. $\pm 3\sigma$ value at worst case conditions | | ± 0.5 | | %FS |
| DNL | Differential non linearity | | ± 0.2 | | LSB |
| INL | Integral non linearity | | ± 0.6 | | LSB |
| $V_{CM,out}$ | Common mode voltage output | | $V_{AVDD}/2$ | | |
| Analog Input | | | | | |
| $V_{CM,in}$ | Analog input common mode voltage | $V_{CM} - 0.1$ | | $V_{CM} + 0.2$ | V |
| FSR | Differential input voltage full scale range | | 2.0 | | V _{pp} |
| $C_{in,Q}$ | Differential input capacitance, Quad channel mode | | 5 | | pF |
| $C_{in,D}$ | Differential input capacitance, Dual channel mode | | 7 | | pF |
| $C_{in,S}$ | Differential input capacitance, Single channel mode | | 11 | | pF |
| FPBW | Full Power Bandwidth | | 700 | | MHz |
| F_{INmax_2VPP} | Maximum input frequency @ 2V _{pp} full scale range | | 500 | | MHz |
| F_{INmax_1VPP} | Maximum input frequency @ 1V _{pp} full scale range | | 800 | | MHz |
| Power Supply | | | | | |
| V_{AVDD} | Analog Supply Voltage | 1.7 | 1.8 | 2.0 | V |
| V_{DVDD} | Digital and output driver supply voltage | 1.7 | 1.8 | 2.0 | V |
| V_{OVDD} | Digital CMOS Input Supply Voltage | 1.7 | 1.8 | 3.6 | V |
| Temperature | | | | | |
| T_A | Operating free-air temperature | -40 | | 85 | °C |



ASD5020 High Speed Mode

AVDD=DVDD=OVDD=1.8V, 50% clock duty cycle, -1dBFS 70 MHz input signal, Gain = 1X, 12-bit output, RSDS output data levels, unless otherwise noted

| Parameter | Description | Min | Typ | Max | Unit |
|------------------------|---|--------------------|------|------------------|------|
| Performance | | | | | |
| SNR | Signal to Noise Ratio | | | | |
| | Single Channel Mode , $F_S = 640$ MSPS | | 70 | | dBFS |
| | Single Channel Mode , $F_S = 640$ MSPS Gain = 10X | | 52 | | dBFS |
| | Dual Channel Mode , $F_S = 320$ MSPS | | 70 | | dBFS |
| SINAD | Quad Channel Mode , $F_S = 160$ MSPS | | 70 | | dBFS |
| | Signal to Noise and Distortion Ratio | | | | |
| | Single Channel Mode , $F_S = 640$ MSPS | | 62 | | dBFS |
| | Single Channel Mode , $F_S = 640$ MSPS, Gain = 10X | | 51 | | dBFS |
| SFDR | Dual Channel Mode , $F_S = 320$ MSPS | | 66 | | dBFS |
| | Quad Channel Mode , $F_S = 160$ MSPS | | 67 | | dBFS |
| | Spurious Free Dynamic Range | | | | |
| | Single Channel Mode , $F_S = 640$ MSPS | | 65 | | dBc |
| ENOB | Dual Channel Mode , $F_S = 320$ MSPS | | 65 | | dBc |
| | Quad Channel Mode , $F_S = 160$ MSPS | | 65 | | dBc |
| | Effective number of Bits | | | | |
| X _{tlk,HS2} | Single Channel Mode , $F_S = 640$ MSPS | | 10.0 | | bits |
| | Single Channel Mode , $F_S = 640$ MSPS, Gain = 10X | | 8.0 | | bits |
| | Dual Channel Mode , $F_S = 320$ MSPS | | 10.6 | | bits |
| | Quad Channel Mode , $F_S = 160$ MSPS | | 10.8 | | bits |
| X _{tlk,HS4} | CrossTalk Dual Ch Mode. Signal applied to 1 channel (F_{IN0}). Measurement taken on one channel with full scale at F_{IN1} . $F_{IN1} = 8$ MHz, $F_{IN0} = 9.9$ MHz | | TBD | | dBc |
| | CrossTalk Quad Ch Mode. Signal applied to 1 channel (F_{IN0}). Measurement taken on one channel with full scale at F_{IN1} . $F_{IN1} = 8$ MHz, $F_{IN0} = 9.9$ MHz | | TBD | | dBc |
| Power Supply | Single Ch: $F_S = 640$ MSPS, Dual Ch: $F_S = 320$ MSPS, Quad Ch: $F_S = 160$ MSPS. | | | | |
| I _{AVDD} | Analog Supply Current | | 190 | | mA |
| I _{DVDD} | Digital and output driver Supply Current | | 82 | | mA |
| P _{AVDD} | Analog Power | | 342 | | mW |
| P _{DVDD} | Digital Power | | 148 | | mW |
| P _{TOT} | Total Power Dissipation | | 490 | | mW |
| P _{PD} | Power Down Mode Dissipation | | 15 | | μW |
| P _{SLP} | Deep Sleep Mode Power Dissipation | | 66 | | mW |
| P _{SLPCH} | Power Dissipation with all channels in sleep channel mode (Light Sleep) | | 121 | | mW |
| P _{SLPCH_SAV} | Power Dissipation savings per channel off | | 92 | | mW |
| Clock Inputs | | | | | |
| F _{Smax} | Max. Conversion Rate in Modes: Single / Dual / Quad Channel | 640 / 320 / 160 | | | MSPS |
| F _{Smin} | Min. Conversion Rate in Modes: Single / Dual / Quad Channel | | | 120 / 60 / 30 | MSPS |



Digital and Switching Specifications

AVDD= DVDD=OVDD=1.8V, RSDS output data levels, unless otherwise noted

| Parameter | Description | Min | Typ | Max | Unit |
|---|---|-------------------------------|--------------------------|--------------------------|--------------|
| Clock Inputs | | | | | |
| DC Compliance | Duty Cycle | 40 | | 60 | % high |
| $V_{CK,diff}$ | Differential input voltage swing | +/-200 | | | mVpp |
| $V_{CK,sine}$ | Differential input voltage swing, sine wave clock input | +/-800 | | | mVpp |
| $V_{CK,CMOS}$ | Voltage input range CMOS (CLKN connected to ground) | | V_{OVDD} | | |
| $V_{CM,CK}$ | Input common mode voltage. Keep voltages within ground and voltage of OVDD | 0.3 | | $V_{OVDD}-0.3$ | V |
| C_{CK} | Differential Input capacitance | | 3 | | pF |
| Logic inputs (CMOS) | | | | | |
| V_{HI} | High Level Input Voltage. $V_{OVDD} \geq 3.0V$ | 2 | | | V |
| V_{HI} | High Level Input Voltage. $V_{OVDD} = 1.7V - 3.0V$ | $0.8 \cdot V_{OVDD}$ | | | V |
| V_{LI} | Low Level Input Voltage. $V_{OVDD} \geq 3.0V$ | 0 | | 0.8 | V |
| V_{LI} | Low Level Input Voltage. $V_{OVDD} = 1.7V - 3.0V$ | 0 | | $0.2 \cdot V_{OVDD}$ | V |
| I_{HI} | High Level Input leakage Current | | | +/-10 | μA |
| I_{LI} | Low Level Input leakage Current | | | +/-10 | μA |
| C_i | Input Capacitance | | 3 | | pF |
| Data outputs | | | | | |
| Compliance | | | LVDS / RSDS | | |
| V_{OUT} | Differential output voltage, LVDS | | 350 | | mV |
| V_{OUT} | Differential output voltage, RSDS | | 150 | | mV |
| V_{CM} | Output common mode voltage | | 1.2 | | V |
| Output coding | Default/optional | Offset Binary/ 2's complement | | | |
| Timing Characteristics | | | | | |
| t_A | Aperture delay | | 1.5 | | ns |
| t_j | Aperture jitter, all bits set to '1' in jitter_ctrl<7:0> | | 120 | | fsrms |
| t_j | Aperture jitter, one bit set to '1' in jitter_ctrl<7:0> | | 160 | | fsrms |
| T_{skew} | Timing skew between ADC channels | | 2.5 | | psrms |
| T_{SU} | Start up time from Power Down Mode and Deep Sleep Mode to Active Mode in μs . See section "Clock Frequency" for details. | | 15 | | μs |
| T_{SLPCH} | Start up time from Sleep Channel Mode to Active Mode | | 0.5 | | μs |
| T_{OVR} | Out of range recovery time | | 1 | | clock cycles |
| $T_{LATHSMQ}$ | Pipeline delay, Quad High Speed Mode | | 32 | | clock cycles |
| $T_{LATHSMD}$ | Pipeline delay, Dual High Speed Mode | | 64 | | clock cycles |
| $T_{LATHSMS}$ | Pipeline delay, Single High Speed Mode | | 128 | | clock cycles |
| LVDS Output Timing Characteristics | | | | | |
| t_{data} | LCLK to data delay time (excluding programmable phase shift) | | 50 | | ps |
| T_{PROP} | Clock propagation delay. | $6 \cdot T_{LVDS} + 2.2$ | $7 \cdot T_{LVDS} + 3.5$ | $7 \cdot T_{LVDS} + 5.0$ | ns |
| | LVDS bit-clock duty-cycle | 45 | | 55 | % LCLK cycle |
| | Frame clock cycle-to-cycle jitter | | | 2.5 | % LCLK cycle |
| T_{EDGE} | Data rise- and fall time 20% to 80% | | 0.7 | | ns |
| $T_{CLKEDGE}$ | Clock rise- and fall time 20% to 80% | | 0.7 | | ns |



Absolute Maximum Ratings

Applying voltages to the pins beyond those specified in Table 1 could cause permanent damage to the circuit.

Table 1: Maximum voltage ratings

| Pin | Reference pin | Rating |
|---------------------------|---------------|----------------|
| AVDD | AVSS | -0.3V to +2.3V |
| DVDD | DVSS | -0.3V to +2.3V |
| OVDD | AVSS | -0.3V to +3.9V |
| AVSS / DVSS | DVSS / AVSS | -0.3V to +0.3V |
| Analog inputs and outputs | AVSS | -0.3V to +2.3V |
| CLKx | AVSS | -0.3V to +3.9V |
| LVDS outputs | DVSS | -0.3V to +2.3V |
| Digital inputs | DVSS | -0.3V to +3.9V |

Table 2 shows the maximum external temperature ratings.

Table 2: Maximum temperature ratings

| | |
|---------------------------------|----------------|
| Operating temperature | -40 to +85 °C |
| Storage temperature | -60 to +150 °C |
| Soldering profile qualification | J-STD-020 |



This device can be damaged by ESD. Even though this product is protected with state-of-the-art ESD protection circuitry, damage may occur if the device is not handled with appropriate precautions. ESD damage may range from device failure to performance degradation. Analog circuitry may be more susceptible to damage as vary small parametric changes can result in specification noncompliance.



Pin Configuration and Description

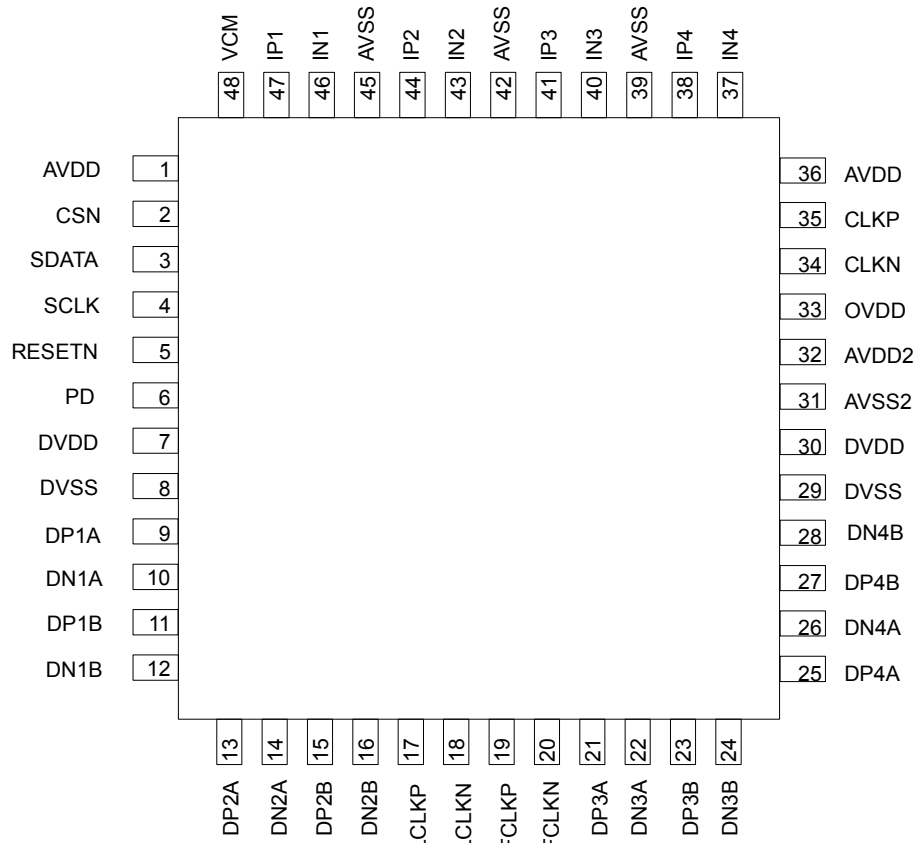


Figure 3: Package diagram

Table 3: Pin descriptions

| PIN NAME | DESCRIPTION | PIN NUMBER | # OF PINS |
|----------|--|------------|-----------|
| AVDD | Analog power supply, 1.8V | 1, 36 | 2 |
| CSN | Chip select enable. Active low | 2 | 1 |
| SDATA | Serial data input | 3 | 1 |
| SCLK | Serial clock input | 4 | 1 |
| RESETN | Reset SPI interface. Active low | 5 | 1 |
| PD | Power-down input. Activate after applying power in order to initialize the ADC correctly. Alternatively use the SPI power down feature | 6 | 1 |
| DVDD | Digital and I/O power supply, 1.8V | 7, 30 | 2 |
| DVSS | Digital ground | 8, 29 | 2 |
| DP1A | LVDS channel 1A, positive output | 9 | 1 |
| DN1A | LVDS channel 1A, negative output | 10 | 1 |
| DP1B | LVDS channel 1B, positive output | 11 | 1 |
| DN1B | LVDS channel 1B, negative output | 12 | 1 |
| DP2A | LVDS channel 2A, positive output | 13 | 1 |
| DN2A | LVDS channel 2A, negative output | 14 | 1 |



| PIN NAME | DESCRIPTION | PIN NUMBER | # OF PINS |
|----------|---|------------|-----------|
| DP2B | LVDS channel 2B, positive output | 15 | 1 |
| DN2B | LVDS channel 2B, negative output | 16 | 1 |
| LCKP | LVDS bit clock, positive output | 17 | 1 |
| LCKN | LVDS bit clock, negative output | 18 | 1 |
| FCLKP | LVDS frame clock (1X), positive output | 19 | 1 |
| FCLKN | LVDS frame clock (1X), negative output | 20 | 1 |
| DP3A | LVDS channel 3A, positive output | 21 | 1 |
| DN3A | LVDS channel 3A, negative output | 22 | 1 |
| DP3B | LVDS channel 3B, positive output | 23 | 1 |
| DN3B | LVDS channel 3B, negative output | 24 | 1 |
| DP4A | LVDS channel 4A, positive output | 25 | 1 |
| DN4A | LVDS channel 4A, negative output | 26 | 1 |
| DP4B | LVDS channel 4B, positive output | 27 | 1 |
| DN4B | LVDS channel 4B, negative output | 28 | 1 |
| AVSS2 | Analog ground domain 2 | 31 | 1 |
| AVDD2 | Analog power supply domain 2, 1.8V | 32 | 1 |
| OVDD | Digital CMOS Inputs supply voltage | 33 | 1 |
| CLKN | Negative differential input clock. | 34 | 1 |
| CLKP | Positive differential input clock | 35 | 1 |
| IN4 | Negative differential input signal, channel 4 | 37 | 1 |
| IP4 | Positive differential input signal, channel 4 | 38 | 1 |
| AVSS | Analog ground | 39, 42, 45 | 3 |
| IN3 | Negative differential input signal, channel 3 | 40 | 1 |
| IP3 | Positive differential input signal, channel 3 | 41 | 1 |
| IN2 | Negative differential input signal, channel 2 | 43 | 1 |
| IP2 | Positive differential input signal, channel 2 | 44 | 1 |
| IN1 | Negative differential input signal, channel 1 | 46 | 1 |
| IP1 | Positive differential input signal, channel 1 | 47 | 1 |
| VCM | Common mode output pin, 0.5*AVDD | 48 | 1 |



Startup Initialization

As part of the ASD5020 power-on sequence both a reset and a power down cycle have to be applied to ensure correct start-up initialization. Reset can be done in one of two ways:

1. By applying a low-going pulse (minimum 20 ns) on the RESETN pin (asynchronous).
2. By using the serial interface to set the 'rst' bit high. Internal registers are reset to default values when this bit is set. The 'rst' bit is self-reset to zero. When using this method, do not apply any low-going pulse on the RESETN pin.

Power down cycling can be done in one of two ways:

1. By applying a high-going pulse (minimum 20 ns) on the PD pin (asynchronous).
2. By cycling the 'pd' bit in register 0F_{hex} to high (reg value '0200'_{hex}) and then low (reg value '0000'_{hex}).

Serial Interface

The ASD5020 configuration registers can be accessed through a serial interface formed by the pins SDATA (serial interface data), SCLK (serial interface clock) and CSN (chip select, active low). The following occurs when CSN is set low:

- Serial data are shifted into the chip
- At every rising edge of SCLK, the value present at SDATA is latched
- SDATA is loaded into the register every 24th rising edge of SCLK

Multiples of 24-bit words data can be loaded within a single active CSN pulse. If more than 24 bits are loaded into SDATA during one active CSN pulse, only the first 24 bits are kept. The excess bits are ignored. Every 24-bit word is divided into two parts:

- The first eight bits form the register address
- The remaining 16 bits form the register data

Acceptable SCLK frequencies are from 20MHz down to a few hertz. Duty-cycle does not have to be tightly controlled.

Timing Diagram

Figure 4 shows the timing of the serial port interface. Table 4 explains the timing variables used in figure 4.

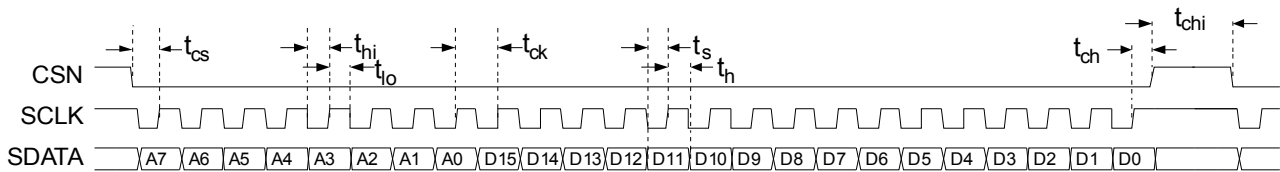


Figure 4: Serial Port Interface timing

Table 4: Serial Port Interface timing definitions

| Parameter | Description | Minimum value | Unit |
|-----------|---------------------------------|---------------|------|
| t_{cs} | Setup time between CSN and SCLK | 8 | ns |
| t_{ch} | Hold time between CSN and SCLK | 8 | ns |
| t_{hi} | SCLK high time | 20 | ns |
| t_{lo} | SCLK low time | 20 | ns |
| t_{ck} | SCLK period | 50 | ns |
| t_s | Data setup time | 5 | ns |
| t_h | Data hold time | 5 | ns |



Timing Diagrams

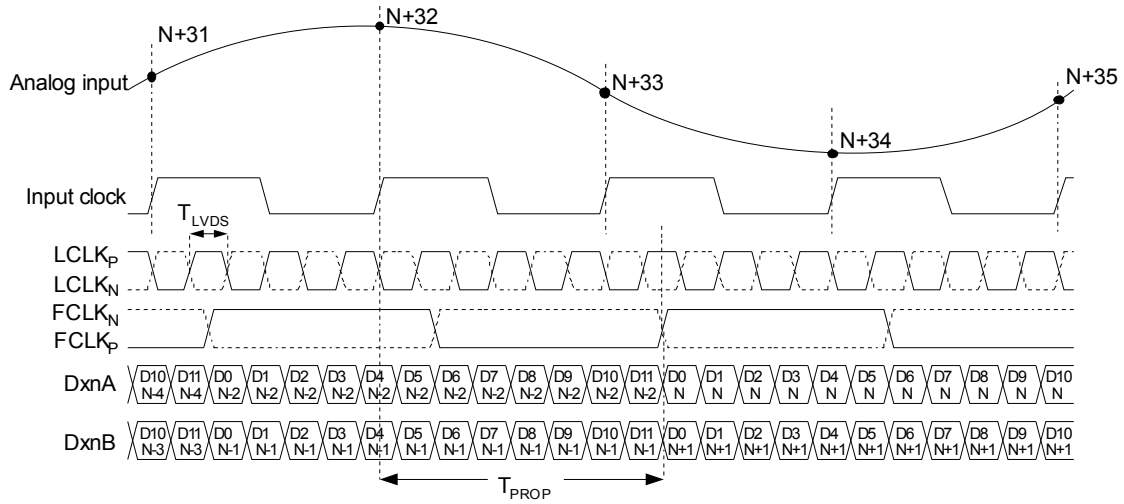


Figure 5: Quad channel - LVDS timing 12-bit output

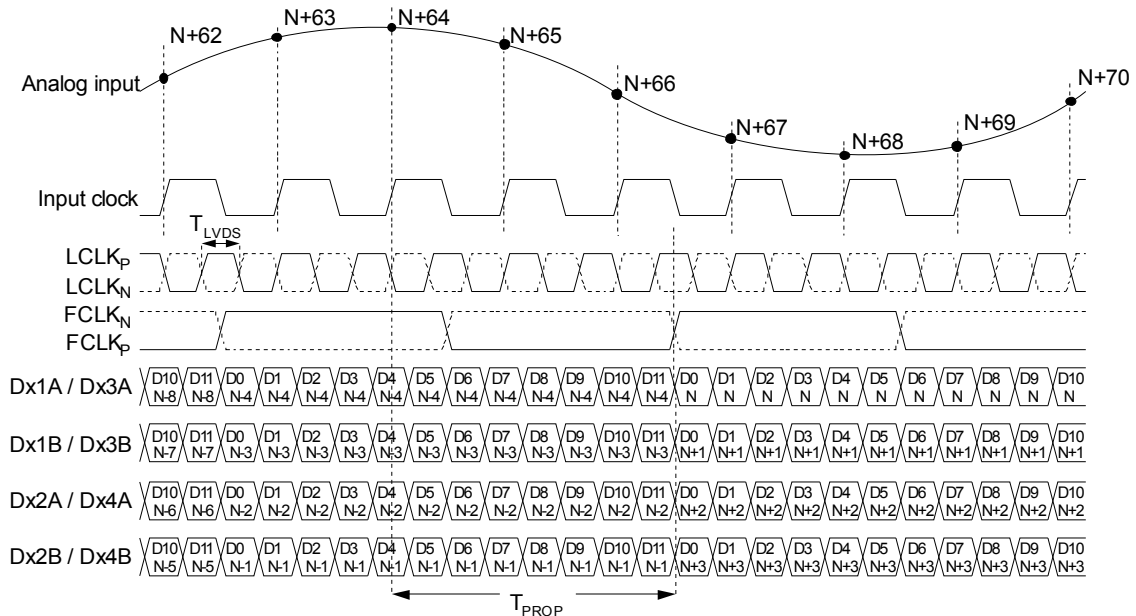


Figure 6: Dual channel - LVDS timing 12-bit output

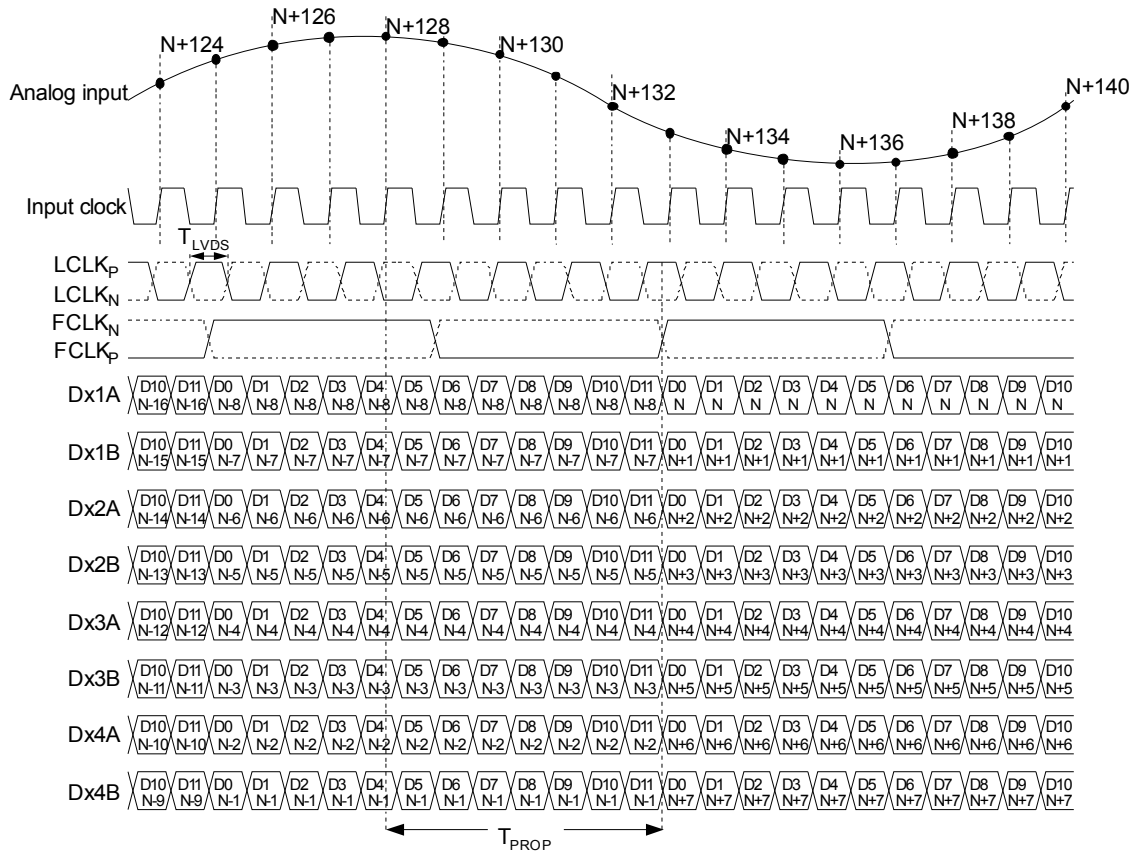


Figure 7: Single channel - LVDS timing 12-bit output

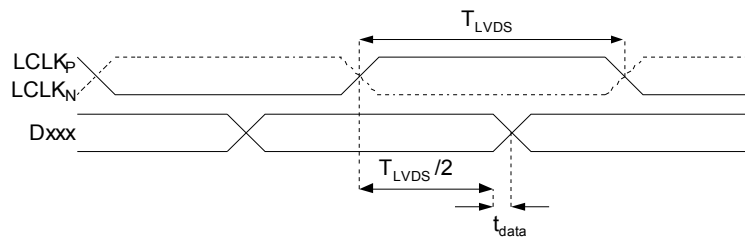


Figure 8: LVDS data timing



Register Map

Table 5: Register map - ASD5020

| Name | Description | Default | D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex Address |
|---------------------|---|---------------------------------------|-----|-----|-----|-----|-----|-----|----|----|----|----|----|----|----|----|----|------|-------------|
| rst * | Self-clearing software reset. | Inactive | | | | | | | | | | | | | | | X | | 0x00 |
| sleep4_ch<4:1> | Channel-specific sleep mode for a Quad Channel setup. | Inactive | | | | | | | | | | | | | X | X | X | X | 0x0F |
| sleep2_ch<2:1> | Channel-specific sleep mode for a Dual Channel setup. | Inactive | | | | | | | | | | X | X | | | | | | |
| sleep1_ch1 | Channel-specific sleep mode for a Single Channel setup. | Inactive | | | | | | | | | X | | | | | | | | |
| sleep | Go to sleep-mode. | Inactive | | | | | | | | X | | | | | | | | | |
| pd | Go to power-down. | Inactive | | | | | | | X | | | | | | | | | | |
| pd_pin_cfg<1:0> | Configures the PD pin function. | PD pin configured for power-down mode | | | | | X | X | | | | | | | | | | | |
| ilvds_lclk<2:0> | LVDS current drive programmability for LCLKP and LCLKN pins. | 3.5 mA drive | | | | | | | | | | | | | X | X | X | 0x11 | |
| ilvds_frame<2:0> | LVDS current drive programmability for FCLKP and FCLKN pins. | 3.5 mA drive | | | | | | | | | X | X | X | | | | | | |
| ilvds_dat<2:0> | LVDS current drive programmability for output data pins. | 3.5 mA drive | | | | | X | X | X | | | | | | | | | | |
| en_lvds_term | Enables internal termination for LVDS buffers. | Termination disabled | X | | | | | | | | | | | | | | | 0x12 | |
| term_lclk<2:0> | Programmable termination for LCLKN and LCLKP buffers. | Termination disabled | 1 | | | | | | | | | | | | X | X | X | | |
| term_frame<2:0> | Programmable termination for FCLKN and FCLKP buffers. | Termination disabled | 1 | | | | | | | X | X | X | | | | | | | |
| term_dat<2:0> | Programmable termination for output data buffers. | Termination disabled | 1 | | | | X | X | X | | | | | | | | | | |
| invert4_ch<4:1> | Channel specific swapping of the analog input signal for a Quad Channel setup. | IPx is positive input | | | | | | | | | | | | | X | X | X | X | 0x24 |
| invert2_ch<2:1> | Channel specific swapping of the analog input signal for a Dual Channel setup. | IPx is positive input | | | | | | | | | X | X | | | | | | | |
| invert1_ch1 | Channel specific swapping of the analog input signal for a Single Channel setup. | IPx is positive input | | | | | | | | X | | | | | | | | | |
| en_ramp | Enables a repeating full-scale ramp pattern on the outputs. | Inactive | | | | | | | | | X | 0 | 0 | | | | | 0x25 | |
| dual_custom_pat | Enable the mode wherein the output toggles between two defined codes. | Inactive | | | | | | | | | 0 | X | 0 | | | | | | |
| single_custom_pat | Enables the mode wherein the output is a constant specified code. | Inactive | | | | | | | | | 0 | 0 | X | | | | | | |
| bits_custom1 <15:0> | Bits for the single custom pattern and for the first code of the dual custom pattern. <0> is the LSB. | 0x0000 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | 0x26 |
| bits_custom2 <15:0> | Bits for the second code of the dual custom pattern. | 0x0000 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | 0x27 |
| cgain4_ch1 <3:0> | Programmable coarse gain channel 1 in a Quad Channel setup. | 1x gain | | | | | | | | | | | | | X | X | X | X | 0x2A |
| cgain4_ch2 <3:0> | Programmable coarse gain channel 2 in a Quad Channel setup. | 1x gain | | | | | | | | | X | X | X | X | | | | | |
| cgain4_ch3 <3:0> | Programmable coarse gain channel 3 in a Quad Channel setup. | 1x gain | | | | X | X | X | X | | | | | | | | | | |
| cgain4_ch4 <3:0> | Programmable coarse gain channel 4 in a Quad Channel setup. | 1x gain | X | X | X | X | | | | | | | | | | | | | |
| cgain2_ch1 <3:0> | Programmable coarse gain channel 1 in a Dual Channel setup. | 1x gain | | | | | | | | | | | | | X | X | X | X | 0x2B |
| cgain2_ch2 <3:0> | Programmable coarse gain channel 2 in a Dual Channel setup. | 1x gain | | | | | | | | | X | X | X | X | | | | | |



| Name | Description | Default | D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex Address |
|--------------------------|---|--------------------------------|-----|-----|-----|-----|-----|-----|----|----|----|----|----|----|----|----|----|----|-------------|
| cgain1_ch1 <3:0> | in a Dual Channel setup. Programmable coarse gain channel 1 in a Single Channel setup. | 1x gain | | | | | X | X | X | X | | | | | | | | | |
| jitter_ctrl<7:0> | Clock jitter adjustment. | 160 fsrms | | | | | | | | | X | X | X | X | X | X | X | X | 0x30 |
| precision_mode * | Enable Quad Channel 14 bits precision mode. | Inactive | | | | | | | | | | | | | | X | | | |
| high_speed_mode * <2:0> | Enable high speed mode, Single, Dual or Quad channel. | High speed mode – Quad Channel | | | | | | | | | | | | | | X | X | X | 0x31 |
| clk_divide<1:0> * | Define clock divider factor: 1, 2, 4 or 8 | Divide by 1 | | | | | | X | X | | | | | | | | | | |
| coarse_gain_cfg | Configures the coarse gain setting | x-gain enabled | | | | | | | | | | | | | | | | X | |
| fine_gain_en | Enable use of fine gain. | Disabled | | | | | | | | | | | | | | | | X | 0x33 |
| fgain_branch1<6:0> | Programmable fine gain for branch1. | 1x / 0dB gain | | | | | | | | | X | X | X | X | X | X | X | X | |
| fgain_branch2<6:0> | Programmable fine gain for branch 2. | 1x / 0dB gain | X | X | X | | X | X | X | X | | | | | | | | | 0x34 |
| fgain_branch3<6:0> | Programmable fine gain for branch 3. | 1x / 0dB gain | | | | | | | | | X | X | X | X | X | X | X | X | |
| fgain_branch4<6:0> | Programmable fine gain for branch 4. | 1x / 0dB gain | X | X | X | | X | X | X | X | | | | | | | | | 0x35 |
| fgain_branch5<6:0> | Programmable fine gain for branch 5. | 1x / 0dB gain | | | | | | | | | X | X | X | X | X | X | X | X | |
| fgain_branch6<6:0> | Programmable fine gain for branch 6. | 1x / 0dB gain | X | X | X | | X | X | X | X | | | | | | | | | 0x36 |
| fgain_branch7<6:0> | Programmable fine gain for branch 7. | 1x / 0dB gain | | | | | | | | | X | X | X | X | X | X | X | X | |
| fgain_branch8<6:0> | Programmable fine gain for branch 8. | 1x / 0dB gain | X | X | X | | X | X | X | X | | | | | | | | | 0x37 |
| inp_sel_adc1<4:0> | Input select for adc 1. | Signal input: IP1/IN1 | | | | | | | | | | | X | X | X | X | X | X | |
| inp_sel_adc2<4:0> | Input select for adc 2. | Signal input: IP2/IN2 | X | X | X | X | X | X | X | | | | | | | | | | 0x3A |
| inp_sel_adc3<4:0> | Input select for adc 3. | Signal input: IP3/IN3 | | | | | | | | | | | X | X | X | X | X | X | |
| inp_sel_adc4<4:0> | Input select for adc 4. | Signal input: IP4/IN4 | X | X | X | X | X | X | X | | | | | | | | | | 0x3B |
| phase_ddr<1:0> | Controls the phase of the LCLK output relative to data. | 90 degrees | | | | | | | | | X | X | | | | | | | 0x42 |
| pat_deskew | Enable deskew pattern mode. | Inactive | | | | | | | | | | | | | | 0 | X | | |
| pat_sync | Enable sync pattern mode. | Inactive | | | | | | | | | | | | | | X | 0 | | 0x45 |
| btc_mode | Binary two's complement format for ADC output data. | Straight offset binary | | | | | | | | | | | | | | X | | | |
| msb_first | Serialized ADC output data comes out with MSB first. | LSB first | | | | | | | | | | | | | | X | | | 0x46 |
| adc_curr<2:0> | ADC current scaling. | Nominal | | | | | | | | | | | | | | X | X | X | |
| ext_vcm_bc<1:0> | VCM buffer driving strength control. | Nominal | | | | | | | | | X | X | | | | | | | 0x50 |
| lvds_pd_mode | Controls LVDS power down mode | High z-mode | | | | | | | | | | | | | | | | X | 0x52 |
| lvds_output_mode <2:0> * | Sets the number of LVDS output bits. | 12 bit | | | | | | | | | | | | | | X | X | X | |
| low_clk_freq * | Low clock frequency used. | Inactive | | | | | | | | | | | | | | X | | | |
| lvds_advance | Advance LVDS data bits and frame clock by one clock cycle | Inactive | | | | | | | | | 0 | X | | | | | | | 0x53 |
| lvds_delay | Delay LVDS data bits and frame clock by one clock cycle | Inactive | | | | | | | | | X | 0 | | | | | | | |
| fs_cntrl<5:0> | Fine adjust ADC full scale range | 0% change | | | | | | | | | X | X | X | X | X | X | X | X | 0x55 |
| startup_ctrl<2:0> * | Controls start-up time. | '000' | | | | | | | | | | | | | | X | X | X | 0x56 |

Undefined register addresses must not be written to; incorrect behavior may be the result.
 Unused register bits (blank table cells) must be set to '0' when programming the registers.
 All registers can be written to while the chip is in power down.
 *) These registers requires a power down cycle when written to (See Startup Initialization).



Register Description

Software Reset

| Name | Description | Default | D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex Address |
|------|-------------------------------|----------|-----|-----|-----|-----|-----|-----|----|----|----|----|----|----|----|----|----|----|-------------|
| rst | Self-clearing software reset. | Inactive | | | | | | | | | | | | | | | | X | 0x00 |

Setting the *rst* register bit to '1', restores the default value of all the internal registers including the *rst* register bit itself.

Modes of Operation

| Name | Description | Default | D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex Address |
|----------------------|---|--------------------------------|-----|-----|-----|-----|-----|-----|----|----|----|----|----|----|----|----|----|----|-------------|
| high_speed_mode<2:0> | Enable high speed mode, Single, Dual or Quad channel. | High speed mode – Quad Channel | | | | | | | | | | | | | 0 | X | X | X | |
| clk_divide<1:0> | Define clock divider factor: 1, 2, 4 or 8 | Divide by 1 | | | | | | | X | X | | | | | | | | | |

The ASD5020 has three main high speed operating modes controlled by the register bit *high_speed_mode* as defined in table 6. Power down mode, as described in section 'Startup Initialization', must be activated after or during a change of operating mode to ensure correct operation. The high speed modes all utilize interleaving to achieve high sampling speed. Quad channel mode interleaves 2 ADC branches, dual channel mode interleaves 4 ADC branches, while single channel mode interleaves all 8 ADC branches.

Table 6: Modes of operation

| high_speed_mode<2:0> | Mode of operation | Description |
|----------------------|---------------------------------------|---|
| 0 0 1 | Single channel 12-bit high speed mode | Single channel by interleaving ADC1 to ADC4 |
| 0 1 0 | Dual channel 12-bit high speed mode | Dual channel where channel 1 is made by interleaving ADC1 and ADC2, channel 2 by interleaving ADC3 and ADC4 |
| 1 0 0 | Quad channel 12-bit high speed mode | Quad channel where channel 1 corresponds to ADC1, channel2 to ADC2, channel3 to ADC3 and channel 4 to ADC4 |

Only one of the 3 bits should be activated at the same time.

clk_divide<1:0> allows the user to apply an input clock frequency higher than the sampling rate. The clock divider will divide the input clock frequency by a factor of 1, 2, 4, or 8, defined by the *clk_divide<1:0>* register. By setting the *clk_divide<1:0>* value relative to the *channel_num<2:0>* value, the same input clock frequency can be used for all settings on number of channels. e.g: When increasing the number of channels from 1 to 4, the maximum sampling rate is reduced by a factor of 4. By letting *clk_divide<1:0>* follow the *channel_num<2:0>* value, and change it from 1 to 4, the internal clock divider will provide the reduction of the sampling rate without changing the input clock frequency.

Table 7: Clock Divider Factor

| clk_divide<1:0> | Clock Divider Factor | Sampling rate (FS) |
|-----------------|----------------------|---------------------------|
| 00 (default) | 1 | Input clock frequency / 1 |
| 01 | 2 | Input clock frequency / 2 |
| 10 | 4 | Input clock frequency / 4 |
| 11 | 8 | Input clock frequency / 8 |



Input Select

| Name | Description | Default | D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex Address |
|-------------------|-------------------------|-----------------------|-----|-----|-----|-----|-----|-----|----|----|----|----|----|----|----|----|----|----|-------------|
| inp_sel_adc1<4:0> | Input select for adc 1. | Signal input: IP1/IN1 | | | | | | | | | | | X | X | X | X | 0 | | 0x3A |
| inp_sel_adc2<4:0> | Input select for adc 2. | Signal input: IP2/IN2 | | X | X | X | X | 0 | | | | | | | | | | | |
| inp_sel_adc3<4:0> | Input select for adc 3. | Signal input: IP3/IN3 | | | | | | | | | | X | X | X | X | 0 | | | 0x3B |
| inp_sel_adc4<4:0> | Input select for adc 4. | Signal input: IP4/IN4 | | X | X | X | X | 0 | | | | | | | | | | | |

Each ADC is connected to the four input signals via a full flexible cross point switch, set up by *inp_sel_adcx*. In single channel mode, any one of the four inputs can be selected as valid input to the single ADC channel. In dual channel mode, any two of the four inputs can be selected to each ADC channel. In quad channel mode, any input can be assigned to any ADC channel. The switching of inputs can be done during normal operation, and no additional actions are needed. The switching will occur instantaneously at the end of each SPI command.

Table 8: ADC input select

| inp_sel_adcx<4:0> | Selected input |
|-------------------|----------------|
| 0001 0 | IP1/IN1 |
| 0010 0 | IP2/IN2 |
| 0100 0 | IP3/IN3 |
| 1000 0 | IP4/IN4 |
| other | Do not use |

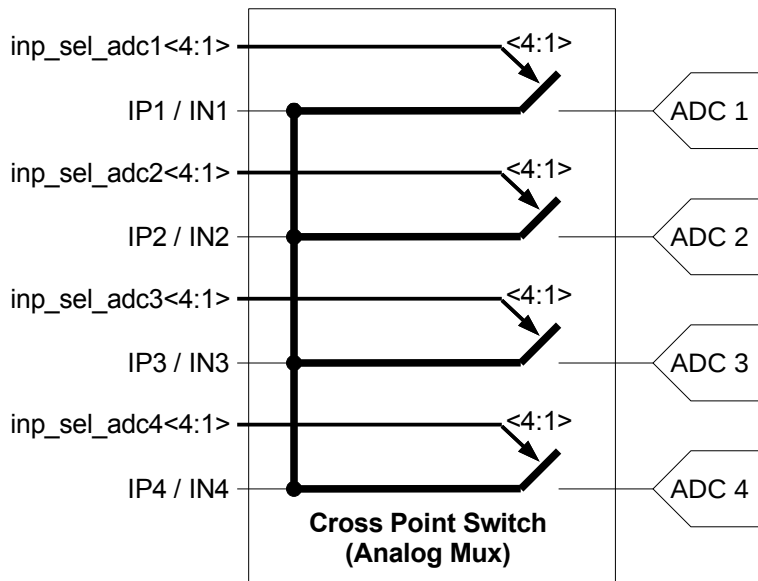


Figure 9: ADC input signals through Cross Point Switch



Full-scale Control

| Name | Description | Default | D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex Address |
|---------------|----------------------------------|-----------|-----|-----|-----|-----|-----|-----|----|----|----|----|----|----|----|----|----|----|-------------|
| fs_cntrl<5:0> | Fine adjust ADC full scale range | 0% change | | | | | | | | | | X | X | X | X | X | X | X | 0x55 |

The full-scale voltage range of ASD5020 can be adjusted using an internal 6-bit DAC controlled by the *fs_cntrl* register. Changing the value in the register by one step, adjusts the full-scale range by approximately 0.3%. This leads to a maximum range of ±10% adjustment. Table 9 shows how the register settings correspond to the full-scale range. Note that the values for full-scale range adjustment are approximate. The DAC is, however, guaranteed to be monotonous.

The full-scale control and the programmable gain features differ in two major ways:

1. The full-scale control feature controls the full-scale voltage range in an analog fashion, whereas the programmable gain is a digital feature.
2. The programmable gain feature has much coarser gain steps and larger range than the full-scale control.

Table 9: Register values with corresponding change in full-scale range

| fs_cntrl<5:0> | Full-scale range adjustment |
|---------------|-----------------------------|
| 111111 | +9.7% |
| 111110 | +9.4% |
| ... | ... |
| 100001 | +0.3% |
| 100000 | +0% |
| 011111 | -0.3% |
| ... | ... |
| 000001 | -9.7% |
| 000000 | -10% |

Current Control

| Name | Description | Default | D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex Address |
|-----------------|--------------------------------------|---------|-----|-----|-----|-----|-----|-----|----|----|----|----|----|----|----|----|----|----|-------------|
| adc_curr<2:0> | ADC current scaling. | Nominal | | | | | | | | | | | | | X | X | X | | 0x50 |
| ext_vcm_bc<1:0> | VCM buffer driving strength control. | Nominal | | | | | | | | | | X | X | | | | | | |

There are two registers that impact performance and power dissipation.

The *adc_curr* register scales the current consumption in the ADC core. The performance is guaranteed at the nominal setting. Lower power consumption can be achieved by reducing the *adc_curr* value, see table 10. The impact on performance will depend on the ADC sampling rate.

Table 10: ADC current control settings

| adc_curr<2:0> | ADC core current |
|---------------|------------------|
| 100 | -40% |
| 101 | -30% |
| 110 | -20% |
| 111 | -10% |
| 000 (default) | Nominal |
| 001 | Do not use |
| 010 | Do not use |
| 011 | Do not use |



The `ext_vcm_bc` register controls the driving strength in the buffer supplying the voltage on the VCM pin. If this pin is not in use, the buffer can be switched off. If current is drawn from the VCM pin, the driving strength can be increased to keep the voltage on this pin at the correct level.

Table 11: External common mode voltage buffer driving strength

| ext_vcm_bc<1:0> | VCM buffer driving strength [μA] Max current sinked/sourced from VCM pin with < 20 mV voltage change. |
|------------------------------|---|
| 00 | Off (VCM floating) |
| 01 (default) | +/-20 |
| 10 | +/-400 |
| 11 | +/-700 |



Start-up and Clock Jitter Control

| Name | Description | Default | D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex Address |
|-------------------|--------------------------|-----------|-----|-----|-----|-----|-----|-----|----|----|----|----|----|----|----|----|----|----|-------------|
| startup_ctrl<2:0> | Controls start-up time. | '000' | | | | | | | | | | | | | X | X | X | | 0x56 |
| jitter_ctrl<7:0> | Clock jitter adjustment. | 160 fsrms | | | | | | | | | X | X | X | X | X | X | X | X | 0x30 |

To optimize start up time, a register is provided where the start-up time in number of clock cycles can be set. Some internal circuitry have start up times that are clock frequency independent. Default counter values are set to accommodate these start up times at the maximum clock frequency (sampling rate). This will lead to increased start up times at low clock frequencies. Setting the value of this register to the nearest higher clock frequency will reduce the count values of the internal counters, to better fit the actual start up time, such that the start up time will be reduced. The start up times from power down and sleep modes are changed by this register setting. If the clock divider is used (set to other than 1), the input clock frequency must be divided by the divider factor to find the correct clock frequency range (see table 7).

Table 12: Start-up time control settings

| Quad channel | | | | Dual channel | | | |
|-------------------|------------------------------|------------------------------|--------------------|-------------------|------------------------------|------------------------------|--------------------|
| startup_ctrl<2:0> | Clock frequency range [MSPS] | Startup delay [clock cycles] | Startup delay [μs] | startup_ctrl<2:0> | Clock frequency range [MSPS] | Startup delay [clock cycles] | Startup delay [μs] |
| 100 | 160 - 250 | 3072 | 12.3 – 19.2 | 100 | 320 - 500 | 6144 | 12.3 – 19.2 |
| 000 | 100 - 160 | 1984 | 12.4 - 19.8 | 000 | 200 - 320 | 3968 | 12.4 - 19.8 |
| 001 | 65 - 100 | 1280 | 12.8 - 19.7 | 001 | 130 – 200 | 2560 | 12.8 - 19.7 |
| 010 | 40 - 65 | 840 | 12.9 - 21 | 010 | 80 - 130 | 1680 | 12.9 - 21 |
| 011 | 30 - 40 | 520 | 13 - 17.3 | 011 | 60 – 80 | 1040 | 13 - 17.3 |
| other | Do not use | - | - | other | Do not use | - | - |

| Single channel | | | |
|-------------------|------------------------------|------------------------------|--------------------|
| startup_ctrl<2:0> | Clock frequency range [MSPS] | Startup delay [clock cycles] | Startup delay [μs] |
| 100 | 640 - 1000 | 12288 | 12.3 – 19.2 |
| 000 | 400 - 640 | 7936 | 12.4 - 19.8 |
| 001 | 260 - 400 | 5120 | 12.8 - 19.7 |
| 010 | 160 - 260 | 3360 | 12.9 - 21 |
| 011 | 120 - 160 | 2080 | 13 - 17.3 |
| other | Do not use | - | - |



jitter_ctrl<7:0> allows the user to set a trade-off between power consumption and clock jitter. If all bits in the register is set low, the clock signal is stopped. The clock jitter depends on the number of bits set to '1' in the *jitter_ctrl*<7:0> register. Which bits are set high does not affect the result.

Table 13: Clock jitter performance

| Number of bits to '1' in <i>jitter_ctrl</i><7:0> | Clock jitter performance High speed mode [fsrms] | Module current consumption [mA] |
|---|---|--|
| 1 | 160 | 1 |
| 2 | 150 | 2 |
| 3 | 136 | 3 |
| 4 | 130 | 4 |
| 5 | 126 | 5 |
| 6 | 124 | 6 |
| 7 | 122 | 7 |
| 8 | 120 | 8 |
| 0 | Clock stopped | |



LVDS Output Configuration and Control

| Name | Description | Default | D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex Address |
|-----------------------|---|------------------------|-----|-----|-----|-----|-----|-----|----|----|----|----|----|----|----|----|----|----|-------------|
| lvds_output_mode<2:0> | Sets the number of LVDS output bits. | 12 bit | | | | | | | | | | | | | X | X | X | | 0x53 |
| low_clk_freq | Low clock frequency used. | Inactive | | | | | | | | | | | | | | X | | | |
| lvds_advance | Advance LVDS data bits and frame clock by one clock cycle | Inactive | | | | | | | | | | 0 | X | | | | | | |
| lvds_delay | Delay LVDS data bits and frame clock by one clock cycle | Inactive | | | | | | | | | | X | 0 | | | | | | |
| phase_ddr<1:0> | Controls the phase of the LCLK output relative to data. | 90 degrees | | | | | | | | | X | X | | | | | | | 0x42 |
| btc_mode | Binary two's complement format for ADC output data. | Straight offset binary | | | | | | | | | | | | | | X | | | 0x46 |
| msb_first | Serialized ADC output data comes out with MSB first. | LSB first | | | | | | | | | | | | | X | | | | |

The ASD5020 serial LVDS output has four different modes selected by the register *lvds_output_mode* as defined in table 14. Power down mode, as described in section 'Startup Initialization', must be activated after or during a change in the number of output bits to ensure correct behavior.

Table 14: Number of bits in LVDS output

| lvds_output_mode<2:0> | Number of bits | Comment |
|-----------------------|----------------|--------------------------------------|
| 000 | 8 bit | 8 bit mode, up to 1GSPS (ASD5010) |
| 001 | 12 bit | Default setting for High Speed Modes |
| 010 | 14 bit | |
| 011 | 16 bit | |
| Other | Do not use | |

For the high speed modes only the 12, 14 and 16-bit LVDS output modes are available. When 14 or 16 bit LVDS output mode is selected the output data will be a 13 bit left justified word filled up with '0's on the LSB side. The different high speed modes uses the LVDS outputs as defined by table 15.

Table 15: High speed modes and use of LVDS outputs

| High speed modes/ channels | LVDS outputs used |
|----------------------------|--|
| Single channel | D1A, D1B, D2A, D2B, D3A, D3B, D4A, D4B |
| Dual channel, channel 1 | D1A, D1B, D2A, D2B |
| Dual channel, channel 2 | D3A, D3B, D4A, D4B |
| Quad channel, channel 1 | D1A, D1B |
| Quad channel, channel 2 | D2A, D2B |
| Quad channel, channel 3 | D3A, D3B |
| Quad channel, channel 4 | D4A, D4B |

Maximum data output bit-rate for the ASD5020 is 1 Gb/s. The maximum sampling rate for the different configurations is given by table 16. The sampling rate is set by the frequency of the input clock (F_s). The frame-rate, i.e. the frequency of the FCLK signal on the LVDS outputs, depends on the selected mode and the sampling frequency (F_s) as defined in table 17.



Table 16: Maximum sampling rate vs number of output bits for different ASD5020 configurations

| Number of bits | Single channel High speed [MSPS] | Dual channel High speed [MSPS] | Quad channel High speed [MSPS] |
|----------------|--|--------------------------------------|--------------------------------------|
| 12 | 660 | 330 | 165 |
| 14 | 560 | 280 | 140 |
| 16 | 500 | 250 | 125 |
| Dual 8 | - | - | - |

Table 17: Output data frame rate

| Mode of operation | Frame-rate (FCLK frequency) |
|----------------------------|-----------------------------|
| High speed, single channel | $F_s / 8$ |
| High speed, dual channel | $F_s / 4$ |
| High speed, quad channel | $F_s / 2$ |

If the ASD5020 device is used at a low sampling rate the register bit *low_clk_freq* has to be set to '1'. See table 18 for when to use this register bit for the different modes of operation.

Table 18: Use of register bit *low_clk_freq*

| Mode of operation | Limit when <i>low_clk_freq</i> should be activated |
|----------------------------|--|
| High speed, single channel | $F_s < 240$ MHz |
| High speed, dual channel | $F_s < 120$ MHz |
| High speed, quad channel | $F_s < 60$ MHz |



To ease timing in the receiver when using multiple ASD5020, the device has the option to adjust the timing of the output data and the frame clock. The propagation delay with respect to the ADC input clock can be moved one LVDS clock cycle forward or backward, by using *lvds_delay* and *lvds_advance*, respectively. See figure 10 for details. Note that LCLK is not affected by *lvds_delay* or *lvds_advance* settings.

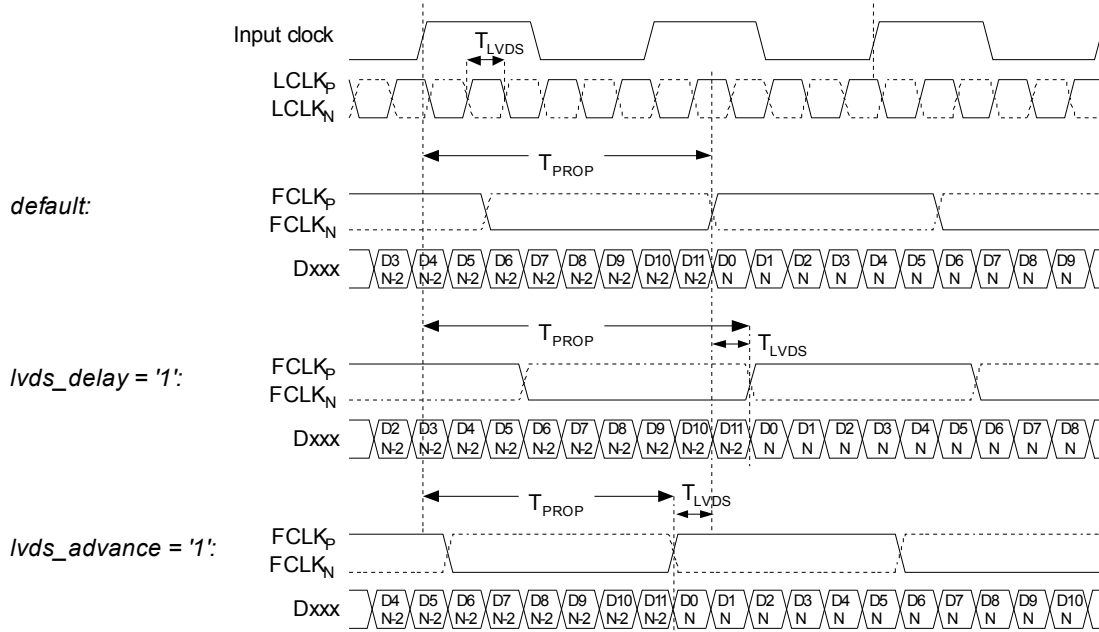


Figure 10: LVDS output timing adjustment

The LVDS output interface of ASD5020 is a DDR interface. The default setting is with the LCLK rising and falling edge transitions in the middle of alternate data windows. The phase for LCLK can be programmed relative to the output frame clock and data bits using *phase_ddr<1:0>*. The LCLK phase modes are shown in figure 11. The default timing is identical to setting *phase_ddr<1:0>='10'*.

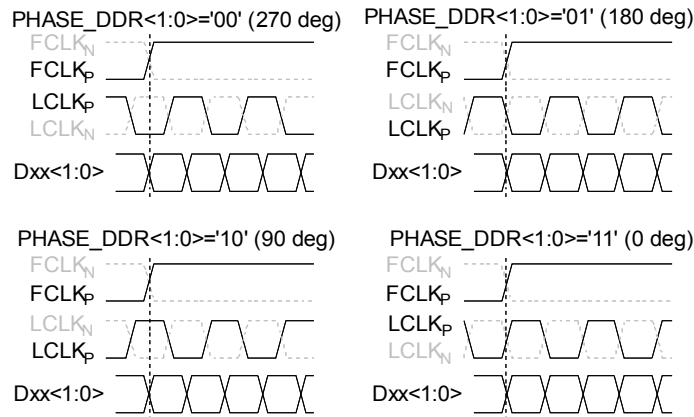


Figure 11: Phase programmability modes for LCLK

The default data output format is offset binary. Two's complement mode can be selected by setting the *btc_mode* bit to '1' which inverts the MSB.

The first bit of the frame (following the rising edge of FCLKP) is the LSB of the ADC output for default settings. Programming the *msb_first* mode results in reverse bit order, and the MSB is output as the first bit following the FCLKP rising edge.



LVDS Drive Strength Programmability

| Name | Description | Default | D15 D14 D13 D12 | D11 D10 D9 D8 | D7 D6 D5 D4 | D3 D2 D1 D0 | Hex Address |
|------------------|--|--------------|-----------------|---------------|-------------|-------------|-------------|
| ilvds_lclk<2:0> | LVDS current drive programmability for LCLKP and LCLKN pins. | 3.5 mA drive | | | | X X X | 0x11 |
| ilvds_frame<2:0> | LVDS current drive programmability for FCLKP and FCLKN pins. | 3.5 mA drive | | | X X X | | |
| ilvds_dat<2:0> | LVDS current drive programmability for output data pins. | 3.5 mA drive | | X X X | | | |

The current delivered by the LVDS output drivers can be configured as shown in table 19. The default current is 3.5mA, which is what the LVDS standard specifies.

Setting the *ilvds_lclk<2:0>* register controls the current drive strength of the LVDS clock output on the LCLKP and LCLKN pins.

Setting the *ilvds_frame<2:0>* register controls the current drive strength of the frame clock output on the FCLKP and FCLKN pins.

Setting the *ilvds_dat<2:0>* register controls the current drive strength of the data outputs on the D[8:1]P and D[8:1]N pins.

Table 19: LVDS output drive strength for LCLK, FCLK and data

| ilvds_*<2:0> | LVDS drive strength |
|--------------|---------------------|
| 000 | 3.5 mA (default) |
| 001 | 2.5 mA |
| 010 | 1.5 mA (RSDS) |
| 011 | 0.5 mA |
| 100 | 7.5 mA |
| 101 | 6.5 mA |
| 110 | 5.5 mA |
| 111 | 4.5 mA |

LVDS Internal Termination Programmability

| Name | Description | Default | D15 D14 D13 D12 | D11 D10 D9 D8 | D7 D6 D5 D4 | D3 D2 D1 D0 | Hex Address |
|-----------------|---|----------------------|-----------------|---------------|-------------|-------------|-------------|
| en_lvds_term | Enables internal termination for LVDS buffers. | Termination disabled | X | | | | 0x12 |
| term_lclk<2:0> | Programmable termination for LCLKN and LCLKP buffers. | Termination disabled | 1 | | | X X X | |
| term_frame<2:0> | Programmable termination for FCLKN and FCLKP buffers. | Termination disabled | 1 | | X X X | | |
| term_dat<2:0> | Programmable termination for output data buffers. | Termination disabled | 1 | X X X | | | |

The off-chip load on the LVDS buffers may represent a characteristic impedance that is not perfectly matched with the PCB traces. This may result in reflections back to the LVDS outputs and loss of signal integrity. This effect can be mitigated by enabling an internal termination between the positive and negative outputs of each LVDS buffer. Internal termination mode can be selected by setting the *en_lvds_term* bit to '1'. Once this bit is set, the internal termination values for the bit clock, frame clock, and data buffers can be independently programmed using sets of three bits. Table 20 shows how the internal termination of the LVDS buffers are programmed. The values are typical values and can vary by up to ±20% from device to device and across temperature.

*Table 20: LVDS output internal termination for LCLK, FCLK and data*

| term_*<2:0> | LVDS Internal Termination |
|--------------------------|----------------------------------|
| 000 | Termination disabled |
| 001 | 260Ω |
| 010 | 150Ω |
| 011 | 94Ω |
| 100 | 125Ω |
| 101 | 80Ω |
| 110 | 66Ω |
| 111 | 55Ω |



Power Mode Control

| Name | Description | Default | D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex Address |
|-----------------|---|---------------------------------------|-----|-----|-----|-----|-----|-----|----|----|----|----|----|----|----|----|----|------|-------------|
| sleep4_ch<4:1> | Channel-specific sleep mode for a Quad Channel setup. | Inactive | | | | | | | | | | | | | X | X | X | X | 0x0F |
| sleep2_ch<2:1> | Channel-specific sleep mode for a Dual Channel setup. | Inactive | | | | | | | | | | X | X | | | | | | |
| sleep1_ch1 | Channel-specific sleep mode for a Single Channel setup. | Inactive | | | | | | | | | X | | | | | | | | |
| sleep | Go to sleep-mode. | Inactive | | | | | | | X | | | | | | | | | | |
| pd | Go to power-down. | Inactive | | | | | | X | | | | | | | | | | | |
| pd_pin_cfg<1:0> | Configures the PD pin function. | PD pin configured for power-down mode | | | | | X | X | | | | | | | | | | | |
| lvds_pd_mode | Controls LVDS power down mode | High z-mode | | | | | | | | | | | | | | | X | 0x52 | |

The ASD5020 device has several modes for power management, from sleep modes with short start up time to full power down with extremely low power dissipation. There are two sleep modes, both with the LVDS clocks (FCLK, LCLK) running, such that the synchronization with the receiver is maintained. The first is a light sleep mode (*sleep*_ch*) with short start up time, and the second a deep sleep mode (*sleep*) with the same start up time as full power down.

Setting *sleep4_ch<n>* = '1' sets channel <n> in a Quad Channel setup in sleep mode. Setting *sleep2_ch<n>* = '1' sets channel <n> in a Dual Channel setup in sleep mode. Setting *sleep1_ch1* = '1' sets the ADC channel in a Single Channel setup in sleep mode. This is a light sleep mode with short start up time.

Setting *sleep* = '1', puts all channels to sleep, but keeps FCLK and LCLK running to maintain LVDS synchronization. The start up time is the same as for complete power down. Power consumption is significantly lower than for setting all channels to sleep by using the *sleep*_ch* register.

Setting *pd* = '1' completely powers down the chip, including the band-gap reference circuit. Start-up time from this mode is significantly longer than from the *sleep*_ch* mode. The synchronization with the LVDS receiver is lost since LCLK and FCLK outputs are put in high-Z mode.

Setting *pdn_pin_cfg<1:0>* = 'x1' configures the circuit to enter sleep channel mode (all channels off) when the PD pin is set high. This is equal to setting all channels to sleep by using *sleep*_ch*. The channels can not be powered down separately using the PD pin. Setting *pdn_pin_cfg<1:0>* = '10' configures the circuit to enter (deep) sleep mode when the PD pin is set high (equal to setting *sleep*='1'). When *pdn_pin_cfg <1:0>* = '00', which is the default, the circuit enters the power down mode when the PD pin is set high.

The *lvds_pd_mode* register configures whether the LVDS data output drivers are powered down or kept alive in sleep and sleep channel modes. LCLK and FCLK drivers are not affected by this register, and are always on in sleep and sleep channel modes. If *lvds_pd_mode* is set low (default), the LVDS output is put in high Z mode, and the driver is completely powered down. If *lvds_pd_mode* is set high, the LVDS output is set to constant 0, and the driver is still on during sleep and sleep channel modes.



Programmable Gain

| Name | Description | Default | D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex Address |
|--------------------|---|----------------|-----|-----|-----|-----|-----|-----|----|----|----|----|----|----|----|----|----|----|-------------|
| cgain_cfg | Configures the coarse gain setting | x-gain enabled | | | | | | | | | | | | | | | | X | 0x33 |
| fine_gain_en | Enable use of fine gain. | Disabled | | | | | | | | | | | | | | | | X | |
| cgain4_ch1 <3:0> | Programmable coarse gain channel 1 in a Quad Channel setup. | 1x gain | | | | | | | | | | | | | X | X | X | X | 0x2A |
| cgain4_ch2 <3:0> | Programmable coarse gain channel 2 in a Quad Channel setup. | 1x gain | | | | | | | | | X | X | X | X | | | | | |
| cgain4_ch3 <3:0> | Programmable coarse gain channel 3 in a Quad Channel setup. | 1x gain | | | | | X | X | X | X | | | | | | | | | |
| cgain4_ch4 <3:0> | Programmable coarse gain channel 4 in a Quad Channel setup. | 1x gain | X | X | X | X | | | | | | | | | | | | | |
| cgain2_ch1 <3:0> | Programmable coarse gain channel 1 in a Dual Channel setup. | 1x gain | | | | | | | | | | | | | X | X | X | X | 0x2B |
| cgain2_ch2 <3:0> | Programmable coarse gain channel 2 in a Dual Channel setup. | 1x gain | | | | | | | | | X | X | X | X | | | | | |
| cgain1_ch1 <3:0> | Programmable coarse gain channel 1 in a Single Channel setup. | 1x gain | | | | | X | X | X | X | | | | | | | | | |
| fgain_branch1<6:0> | Programmable fine gain for branch1. | 1x / 0dB gain | | | | | | | | | X | X | X | X | X | X | X | X | 0x34 |
| fgain_branch2<6:0> | Programmable fine gain for branch 2. | 1x / 0dB gain | X | X | X | X | X | X | X | X | | | | | | | | | |
| fgain_branch3<6:0> | Programmable fine gain for branch 3. | 1x / 0dB gain | | | | | | | | | X | X | X | X | X | X | X | X | 0x35 |
| fgain_branch4<6:0> | Programmable fine gain for branch 4. | 1x / 0dB gain | X | X | X | X | X | X | X | X | | | | | | | | | |
| fgain_branch5<6:0> | Programmable fine gain for branch 5. | 1x / 0dB gain | | | | | | | | | X | X | X | X | X | X | X | X | 0x36 |
| fgain_branch6<6:0> | Programmable fine gain for branch 6. | 1x / 0dB gain | X | X | X | X | X | X | X | X | | | | | | | | | |
| fgain_branch7<6:0> | Programmable fine gain for branch 7. | 1x / 0dB gain | | | | | | | | | X | X | X | X | X | X | X | X | 0x37 |
| fgain_branch8<6:0> | Programmable fine gain for branch 8. | 1x / 0dB gain | X | X | X | X | X | X | X | X | | | | | | | | | |

The device includes a digital programmable gain in addition to the Full-scale control. The programmable gain of each channel can be individually set using a four bit code, indicated as *cgain*<3:0>*. The gain is configured by the register *cgain_cfg*, when *cgain_cfg* equals '0' a gain in dB steps is enabled as defined in table 21 otherwise if *cgain_cfg* equals '1' the gain is defined by table 22.

Table 21: Gain setting – dB step

| cgain_cfg | cgain*<3:0> | Implemented gain [dB] |
|-----------|-------------|-----------------------|
| 0 | 0000 | 0 |
| 0 | 0001 | 1 |
| 0 | 0010 | 2 |
| 0 | 0011 | 3 |
| 0 | 0100 | 4 |
| 0 | 0101 | 5 |
| 0 | 0110 | 6 |
| 0 | 0111 | 7 |
| 0 | 1000 | 8 |
| 0 | 1001 | 9 |
| 0 | 1010 | 10 |
| 0 | 1011 | 11 |
| 0 | 1100 | 12 |
| 0 | 1101 | Not used |
| 0 | 1110 | Not used |
| 0 | 1111 | Not used |



Table 22: Gain setting - x step

| cgain_cfg | cgain*<3:0> | Implemented gain factor [x] |
|-----------|-------------|-----------------------------|
| 1 | 0000 | 1 |
| 1 | 0001 | 1.25 |
| 1 | 0010 | 2 |
| 1 | 0011 | 2.5 |
| 1 | 0100 | 4 |
| 1 | 0101 | 5 |
| 1 | 0110 | 8 |
| 1 | 0111 | 10 |
| 1 | 1000 | 12.5 |
| 1 | 1001 | 16 |
| 1 | 1010 | 20 |
| 1 | 1011 | 25 |
| 1 | 1100 | 32 |
| 1 | 1101 | 50 |
| 1 | 1110 | Not used |
| 1 | 1111 | Not used |

There is a digital fine gain implemented for each ADC branch to adjust the fine gain errors between the branches. The gain is controlled by *fgain_branch** as defined in table 23. For the high speed interleaved modes, there will be no missing codes when using digital fine gain, due to higher resolution internally (1 bit). The relationship between channels and ADC branches is shown in tables 24-26.

To enable the fine gain function the register bit *fine_gain_en* has to be activated, set to '1'.

Table 23: Fine gain setting

| fgain_branchx<6:0> | Arithmetic function | Implemented gain (x) | Gain (dB) |
|--------------------|---|----------------------|-----------|
| 0 1 1 1 1 1 1 | $OUT=(1+2^{-8}+2^{-9}+2^{-10}+2^{-11}+2^{-12}+2^{-13})\cdot IN$ | 1.0077 | 0.0665 |
| 0 1 1 1 1 1 0 | $OUT=(1+2^{-8}+2^{-9}+2^{-10}+2^{-11}+2^{-12})\cdot IN$ | 1.0076 | 0.0655 |
| 0 1 1 1 1 0 1 | $OUT=(1+2^{-8}+2^{-9}+2^{-10}+2^{-11}+2^{-13})\cdot IN$ | 1.0074 | 0.0644 |
| 0 1 1 1 1 0 0 | $OUT=(1+2^{-8}+2^{-9}+2^{-10}+2^{-11})\cdot IN$ | 1.0073 | 0.0634 |
| | ... | ... | ... |
| | ... | ... | ... |
| | ... | ... | ... |
| 0 0 0 0 0 1 1 | $OUT=(1+2^{-12}+2^{-13})\cdot IN$ | 1.0004 | 0.0031 |
| 0 0 0 0 0 1 0 | $OUT=(1+2^{-12})\cdot IN$ | 1.0002 | 0.0021 |
| 0 0 0 0 0 0 1 | $OUT=(1+2^{-13})\cdot IN$ | 1.0001 | 0.0010 |
| 0 0 0 0 0 0 0 | $OUT=IN$ | 1.0000 | 0.0000 |
| 1 1 1 1 1 1 1 | $OUT=IN$ | 1.0000 | 0.0000 |
| 1 1 1 1 1 1 0 | $OUT=(1-2^{-13})\cdot IN$ | 0.9999 | -0.0011 |
| 1 1 1 1 1 0 1 | $OUT=(1-2^{-12})\cdot IN$ | 0.9998 | -0.0021 |
| 1 1 1 1 1 0 0 | $OUT=(1-2^{-12}-2^{-13})\cdot IN$ | 0.9996 | -0.0032 |
| | ... | ... | ... |
| | ... | ... | ... |
| | ... | ... | ... |
| 1 0 0 0 0 1 1 | $OUT=(1-2^{-8}-2^{-9}-2^{-10}-2^{-11})\cdot IN$ | 0.9927 | -0.0639 |
| 1 0 0 0 0 1 0 | $OUT=(1-2^{-8}-2^{-9}-2^{-10}-2^{-11}-2^{-13})\cdot IN$ | 0.9926 | -0.0649 |
| 1 0 0 0 0 0 1 | $OUT=(1-2^{-8}-2^{-9}-2^{-10}-2^{-11}-2^{-12})\cdot IN$ | 0.9924 | -0.0660 |
| 1 0 0 0 0 0 0 | $OUT=(1-2^{-8}-2^{-9}-2^{-10}-2^{-11}-2^{-12}-2^{-13})\cdot IN$ | 0.9923 | -0.0670 |



Analog Input Invert

| Name | Description | Default | D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex Address |
|-----------------|--|-----------------------|-----|-----|-----|-----|-----|-----|----|----|----|----|----|----|----|----|----|----|-------------|
| invert4_ch<4:1> | Channel specific swapping of the analog input signal for a Quad Channel setup. | IPx is positive input | | | | | | | | | | | | | X | X | X | X | 0x24 |
| invert2_ch<2:1> | Channel specific swapping of the analog input signal for a Dual Channel setup. | IPx is positive input | | | | | | | | | | X | X | | | | | | |
| invert1_ch1 | Channel specific swapping of the analog input signal for a Single Channel setup. | IPx is positive input | | | | | | | | | X | | | | | | | | |

The IPx pin represents the positive analog input pin, and INx represents the negative (complementary) input. Setting the bits marked *invertx_ch<n:1>* (individual control for each channel) causes the inputs to be swapped. INx would then represent the positive input, and IPx the negative input.

LVDS Test Patterns

| Name | Description | Default | D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Hex Address |
|---------------------|---|----------|-----|-----|-----|-----|-----|-----|----|----|----|----|----|----|----|----|----|----|-------------|
| en_ramp | Enables a repeating full-scale ramp pattern on the outputs. | Inactive | | | | | | | | | X | 0 | 0 | | | | | | 0x25 |
| dual_custom_pat | Enable the mode wherein the output toggles between two defined codes. | Inactive | | | | | | | | | 0 | X | 0 | | | | | | |
| single_custom_pat | Enables the mode wherein the output is a constant specified code. | Inactive | | | | | | | | | 0 | 0 | X | | | | | | |
| bits_custom1 <15:0> | Bits for the single custom pattern and for the first code of the dual custom pattern. <0> is the LSB. | 0x0000 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | 0x26 |
| bits_custom2 <15:0> | Bits for the second code of the dual custom pattern. | 0x0000 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | 0x27 |
| pat_deskew | Enable deskew pattern mode. | Inactive | | | | | | | | | | | | | | | 0 | X | 0x45 |
| pat_sync | Enable sync pattern mode. | Inactive | | | | | | | | | | | | | | | X | 0 | |

To ease the LVDS synchronization setup of ASD5020, several test patterns can be set up on the outputs. Normal ADC data are replaced by the test pattern in these modes. Setting *en_ramp* to '1' sets up a repeating full-scale ramp pattern on all data outputs. The ramp starts at code zero and is increased 1LSB every clock cycle. It returns to zero code and starts the ramp again after reaching the full-scale code.

A constant value can be set up on the outputs by setting *single_custom_pat* to '1', and programming the desired value in *bits_custom1<15:0>*. In this mode, *bits_custom1<15:0>* replaces the ADC data at the output, and is controlled by LSB-first and MSB-first modes in the same way as normal ADC data are.

The device may also be set up to alternate between two codes by programming *dual_custom_pat* to '1'. The two codes are the contents of *bits_custom1<15:0>* and *bits_custom2<15:0>*.

Since *bit_custom*<15:0>* is a 16 bit word there will be a truncation at the LSB side when using less than 16 bits in the LVDS output word. If 12-bit output is selected bit <15:4> will be used, if 14-bit output is used bit <15:2> will be used and if dual 8-bit is selected bit<15:8> will be put on the LVDS 'A' output and bit <7:0> will be put on the LVDS 'B' output.

Two preset patterns can also be selected:

1. Deskew pattern: Set using *pat_deskew*, this mode replaces the ADC output with a pattern consisting of alternating zeros and ones - MSB will be a zero. For a 12-bit output the pattern will be: '0101010101'
2. Sync pattern: Set using *pat_sync*, the normal ADC word is in this mode replaced by a fixed synchronization pattern where the output word is split in two and the upper part of the word is ones and the lower part is zeros. For a 12-bit output the pattern will be: '111111000000' .

Note: Only one of the above patterns should be selected at the same time.



Theory of Operation

ASD5020 is a multi Mode high-speed, CMOS ADC, consisting of 8 ADC branches, configured in different channel modes, using interleaving to achieve high speed sampling. For all practical purposes, the device can be considered to contain 4 ADCs. Fine gain is adjusted for each of the eight branches separately.

ASD5020 utilizes a LVDS output, described in 'Register Description, LVDS Output Configuration and Control'. The clocks needed (FCLK, LCLK) for the LVDS interface are generated by an internal PLL.

The ASD5020 operate from one clock input, which can be differential or single ended. The sampling clocks for each of the four channels are generated from the clock input using a carefully matched clock buffer tree. Internal clock dividers are utilized to control the clock for each ADC during interleaving. The clock tree is controlled by the Mode of operations.

ASD5020 uses internally generated references. The differential reference value is 1V. This results in a differential input of -1V to correspond to the zero code of the ADC, and a differential input of +1V to correspond to the maximum code.

The ADC employs a Pipeline converter architecture. Each Pipeline Stage feeds its output data into the digital error correction logic, ensuring excellent differential linearity and no missing codes.

ASD5020 operates from two sets of supplies and grounds. The analog supply and ground set is identified as AVDD and AVSS, while the digital set is identified by DVDD and DVSS.

Interleaving Effects and Sampling Order

Interleaving ADCs will generate interleaving artifacts caused by gain, offset and timing mismatch between the ADC branches. The design of ASD5020 has been optimized to minimize these effects. It is not possible, though, to eliminate mismatch completely, such that additional compensation may be needed. The internal digital fine gain control may be used to compensate for gain errors between the ADC branches. Due to the optimization of ASD5020 there is not a one-to-one correspondence between the sampling order, LVDS output order and the branch number. Tables 24, 25 and 26 give an overview of the corresponding branches, LVDS outputs and sampling order for the different high speed modes.

Table 24: Quad channel mode

| Channel # | Sampling order | LVDS output | Fine gain branch |
|-----------|----------------|-------------|------------------|
| 1 | 1 | D1A | 1 |
| | 2 | D1B | 2 |
| 2 | 1 | D2A | 3 |
| | 2 | D2B | 4 |
| 3 | 1 | D3A | 5 |
| | 2 | D3B | 6 |
| 4 | 1 | D4A | 7 |
| | 2 | D4B | 8 |

Table 25: Dual channel mode

| Channel # | Sampling order | LVDS output | Fine gain branch |
|-----------|----------------|-------------|------------------|
| 1 | 1 | D1A | 1 |
| | 2 | D1B | 3 |
| | 3 | D2A | 2 |
| | 4 | D2B | 4 |
| 2 | 1 | D3A | 5 |
| | 2 | D3B | 7 |
| | 3 | D4A | 6 |
| | 4 | D4B | 8 |

Table 26: Single channel mode

| Channel # | Sampling order | LVDS output | Fine gain branch |
|-----------|----------------|-------------|------------------|
| 1 | 1 | D1A | 1 |
| | 2 | D1B | 6 |
| | 3 | D2A | 2 |
| | 4 | D2B | 5 |
| | 5 | D3A | 8 |
| | 6 | D3B | 3 |
| | 7 | D4A | 7 |
| | 8 | D4B | 4 |

Recommended Usage

Analog Input

The analog input to ASD5020 is a switched capacitor track-and-hold amplifier optimized for differential operation. Operation at common mode voltages at mid supply is recommended even if performance will be good for the ranges specified. The VCM pin provides a voltage suitable as common mode voltage reference. The internal buffer for the VCM voltage can be switched off, and driving capabilities can be changed programming the ext_vcm_bc<1:0> register.

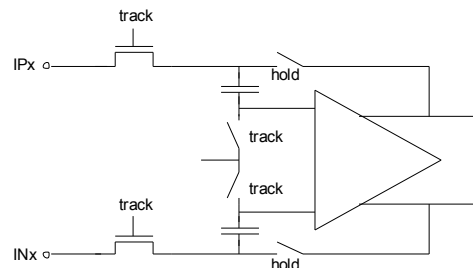


Figure 12: Input configuration

Figure 12 shows a simplified drawing of the input network. The signal source must have sufficiently low output impedance to charge the sampling capacitors within one clock cycle. A small external resistor (e.g. 22 ohm) in series with each input is recommended as it helps reducing transient currents and dampens ringing behavior. A small differential shunt capacitor at the chip



side of the resistors may be used to provide dynamic charging currents and may improve performance. The resistors form a low pass filter with the capacitor, and values must therefore be determined by requirements for the application.

DC-Coupling

Figure 13 shows a recommended configuration for DC-coupling. Note that the common mode input voltage must be controlled according to specified values. Preferably, the CM_EXT output should be used as reference to set the common mode voltage.

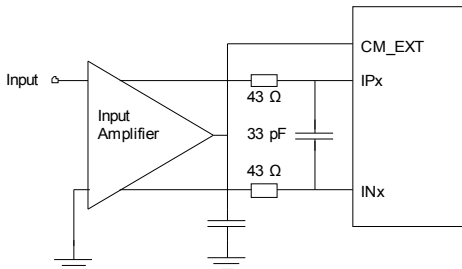


Figure 13: DC coupled input

The input amplifier could be inside a companion chip or it could be a dedicated amplifier. Several suitable single ended to differential driver amplifiers exist in the market. The system designer should make sure the specifications of the selected amplifier is adequate for the total system, and that driving capabilities comply with ASD5020 input specifications.

Detailed configuration and usage instructions must be found in the documentation of the selected driver, and the values given in figure 13 must be adjusted according to the recommendations for the driver.

AC-Coupling

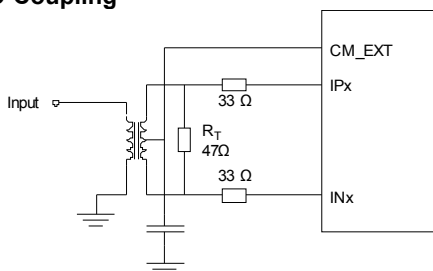


Figure 14: Transformer coupled input

A signal transformer or series capacitors can be used to make an AC-coupled input network. Figure 14 shows a recommended configuration using a transformer. Make sure that a transformer with sufficient linearity is selected, and that the bandwidth of the transformer is appropriate. The bandwidth should preferably exceed the sampling rate of the ADC several times. It is also important to minimize phase mismatch between the differential ADC inputs for good HD2 performance. This type of transformer coupled input is the preferred configuration for high frequency signals as most differential amplifiers do not have adequate performance at high frequencies. Magnetic coupling between the transformers and PCB traces may impact channel crosstalk, and must hence be taken into account during PCB layout.

If the input signal is traveling a long physical distance from the signal source to the transformer (for example a long cable), kick-backs from the ADC will also travel along this distance. If these kick-backs are not terminated properly at the source side, they are reflected and will add to the input signal at the ADC input. This could reduce the ADC performance. To avoid this effect, the source must effectively terminate the ADC kick-backs, or the traveling distance should be very short.

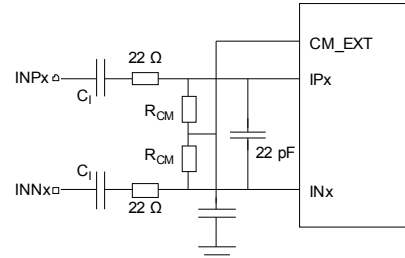


Figure 15: AC coupled input

Figure 15 shows AC-coupling using capacitors. Resistors from the CM_EXT output, R_{CM}, should be used to bias the differential input signals to the correct voltage. The series capacitor, C₁, form the high-pass pole with these resistors, and the values must therefore be determined based on the requirement to the high-pass cut-off frequency.

Note that Start Up Time from Sleep Mode and Power Down Mode will be affected by this filter as the time required to charge the series capacitors is dependent on the filter cut-off frequency.

Clock Input and Jitter Considerations

Typically high-speed ADCs use both clock edges to generate internal timing signals. In ASD5020 only the rising edge of the clock is used.

The input clock can be supplied in a variety of formats. The clock pins are AC-coupled internally, hence a wide common mode voltage range is accepted. Differential clock sources such as LVDS, LVPECL or differential sine wave can be connected directly to the input pins. For CMOS inputs, the CLKN pin should be connected to ground, and the CMOS clock signal should be connected to CLKP. For differential sine wave clock input the amplitude must be at least +/- 0.8 V_{pp}. No additional configuration is needed to set up the clock source format.

The quality of the input clock is extremely important for high-speed, high-resolution ADCs. The contribution to SNR from clock jitter with a full scale signal at a given frequency is shown in equation 1.

$$SNR_{jitter} = 20 \cdot \log(2 \cdot \pi \cdot f_{IN} \cdot \epsilon_t) \quad (1)$$

where f_{IN} is the signal frequency, and ϵ_t is the total rms jitter measured in seconds. The rms jitter is the total of all jitter sources including the clock generation circuitry, clock distribution and internal ADC circuitry.

For applications where jitter may limit the obtainable performance, it is of utmost importance to limit the clock jitter. This can be obtained by using precise and stable clock references (e.g. crystal oscillators with good jitter specifications) and make sure the clock distribution is well controlled. It might be advantageous to use analog power and ground planes to ensure low noise on the supplies to



all circuitry in the clock distribution. It is of utmost importance to avoid crosstalk between the ADC output bits and the clock and between the analog input signal and the clock since such crosstalk often results in harmonic distortion.

The jitter performance is improved with reduced rise and fall times of the input clock. Hence, optimum jitter performance is obtained with LVDS or LVPECL clock with fast edges. CMOS and sine wave clock inputs will result in slightly degraded jitter performance.

If the clock is generated by other circuitry, it should be re-timed with a low jitter master clock as the last operation before it is applied to the ADC clock input.

Application Usage Example

This section gives an overview on how ASD5020 can be used in an application utilizing all active modes with a single clock source. The example assumes that a low jitter 500MHz clock source is applied. A differential clock should be used, and can be generated from a single ended low jitter crystal oscillator, using a transformer or balun in conjunction with ac-coupling to convert from single ended to differential signal.

Start-up Initialization

The start-up sequence will be as follows:

- Apply power
- Apply reset (RESETN low, then high, or SPI command 0x00 0x0001)
- Set power down (PD pin high or SPI command 0x0F 0x0200)
- Set LVDS bit clock phase (phase_dds, register 0x42) if other than default must be used (depends on the receiver).
- Select operating mode, for instance dual channel mode, and clock divider factor 2 (SPI command 0x31 0x0102).
- Set active mode (PD pin low or SPI command 0x0F 0x0000)
- Select analog inputs, for instance input 1 on channel 1 and input 3 on channel 2 (SPI commands 0x3A 0202 and 0x3B 0808)

Change Mode

When changing operational mode, power down must be

activated due to internal synchronization routines. A typical mode change will then be like this:

- Set power down (PD pin high or SPI command 0x0F 0x0200)
- Change mode to for example Single channel mode, and set the clock divider factor to 1 (SPI command 0x31 0x0001)
- Set active mode (PD pin low or SPI command 0x0F 0x0000)
- Select analog inputs, for instance Input 1 (SPI commands 0x3A 0202 and 0x3B 0202)

Table 27 gives an overview of the operational modes in this example and the SPI commands to apply for each mode.

Table 27: Overview of operating modes and setup conditions

| Operating mode | Sampling speed [MSPS] | Clock divider factor | SPI command for mode selection and clock divider |
|----------------|-----------------------|----------------------|--|
| Single channel | 500 | 1 | 0x31 0x0001 |
| Dual channel | 250 | 2 | 0x31 0x0102 |
| Quad channel | 125 | 4 | 0x31 0x0204 |

Select Analog Input

When an operational mode is selected, the analog inputs can be changed 'on-the-fly'. To change analog input one merely have to apply the dedicated SPI commands. The change will occur instantaneously at the end of each SPI command.

Table 28: Example of some analog input selections

| Operating mode | Signal input selection | SPI commands |
|----------------|--|----------------------|
| Single channel | IP4/IN4 | 0x3A 1010, 0x3B 1010 |
| Dual channel | Ch1: IP2/IN2 Ch2: IP3/IN3 | 0x3A 0404, 0x3B 0808 |
| Quad channel | Ch1: IP4/IN4 Ch2: IP3/IN3 Ch3: IP2/IN2 Ch4: IP1/IN1 | 0x3A 1008, 0x3B 0402 |
| Quad channel | Ch1: IP1/IN1 Ch2: IP2/IN2 Ch3: IP3/IN3 Ch4: IP4/IN4 | 0x3A 0204, 0x3B 0810 |



Package Mechanical Data

QFN48

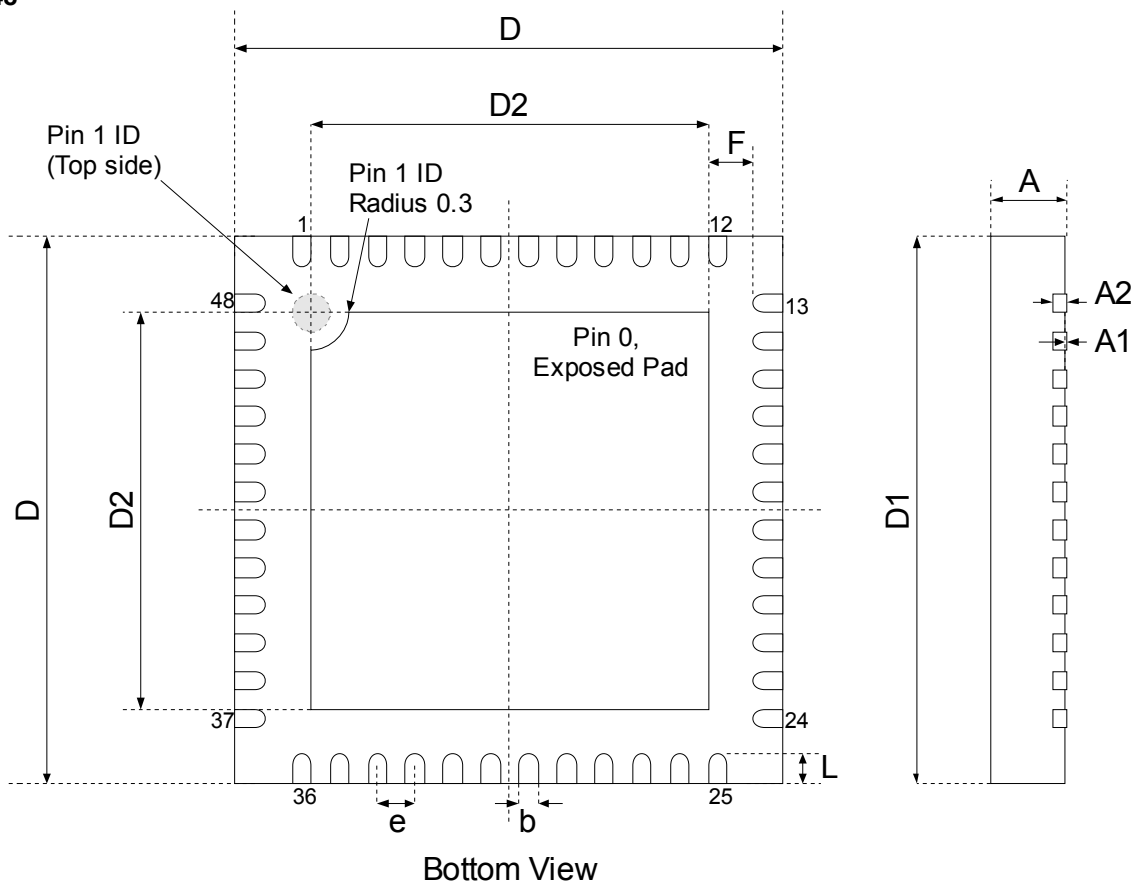


Table 29: QFN48 Dimensions

| Symbol | Millimeter | | | Inch | | |
|--------|------------|----------|------|-------|-----------|-------|
| | Min | Typ | Max | Min | Typ | Max |
| A | 0.8 | 0.9 | 1.0 | 0.031 | 0.035 | 0.039 |
| A1 | 0.00 | 0.02 | 0.05 | 0.000 | 0.0008 | 0.002 |
| A2 | | 0.2 | | | 0.008 | |
| b | 0.18 | 0.25 | 0.3 | 0.007 | 0.010 | 0.012 |
| D | | 7.00 bsc | | | 0.276 bsc | |
| D2 | 5.15 | 5.3 | 5.4 | 0.203 | 0.209 | 0.213 |
| L | 0.3 | 0.4 | 0.5 | 0.012 | 0.016 | 0.020 |
| e | | 0.50 bsc | | | 0.020 bsc | |
| F | 0.6 | | | 0.024 | | |



Product Information

| Product | Status | Datasheet revision | Date |
|---------|-----------------------------------|--------------------|------------|
| ASD5020 | Preliminary Product Specification | 2.0 | 2010.11.08 |

Ordering information

| Model | Temp. range | Package type | Package drawing | MSL, Peak temp (1) | Transport Media |
|----------------|---------------|--------------|-----------------|--------------------|-----------------|
| ASD5020L640INT | -40 to +85 °C | 48 pin QFN | QFN48 | Level 2A | Tray |

(1) MSL, Peak Temp: The moisture sensitivity level rating classified according to the JEDEC industry standard and to peak solder temperature.

Datasheet status

Objective Product Specification:

The values and functionality describe design targets only. Specifications and functionality can be changed without notice

Preliminary Product Specification:

The specifications are based on initial design results. Specifications and functionality can be changed without notice.

Product Specification:

Information is current as of publication data. Products conform to specifications according to the terms of Arctic Silicon Devices AS standard warranty. Production does not necessarily require all parameters to be tested.



asd[®]
arctic silicon devices

Arctic Silicon Devices AS
Vestre Rosten 81
N-7075 Tiller
Norway
Tel: +47 73 10 29 00
Fax: +47 73 10 29 19

Information provided in this document is believed to be accurate and reliable. However, no responsibility is assumed by Arctic Silicon Devices AS for its use. Neither is any responsibility assumed for any infringement of patents or other third party rights that may result from the use of the product or information described herein. No license is implicitly or otherwise granted under any patent or patent right of Arctic Silicon Devices AS.

Arctic Silicon Devices AS specifically disclaims any and all liability, including without limitation incidental or consequential damages. It is the responsibility of the user to ensure that in all respects the application in which Arctic Silicon Devices AS products are used is suited to the purpose of the end user.

Life Support Applications :

Products of Arctic Silicon Devices AS (ASD) are not designed for use in life support appliances, devices or systems, where malfunction can result in personal injury. Customers using or selling ASD products for use in such applications do so at their own risk and agree to fully indemnify ASD for any damages resulting from such improper use or sale.

All rights reserved ©. Reproduction in whole or in part is prohibited without the prior written permission of the copyright holder.

template rev. date: 2009.06.12