# ANALOG DEVICES

# 16-Bit, 16-Channel, 1 MSPS/500 kSPS, Easy Drive Multiplexed SAR ADC

## AD4695/AD4696

## FEATURES

**Easy Drive**
- Greatly reduced analog input and reference drive requirements
- Overvoltage protection up to 5 mA on each analog input
- Long acquisition phase, ≥71.5% of cycle time at 1 MSPS

**High performance**
- Sample rate: 500 kSPS/1 MSPS options
- INL: ±1.0 LSB maximum
- Guaranteed 16-bit, no missing codes
- SINAD: 93 dB typical, $f_{IN}$ = 1 kHz
- Oversampled dynamic range, OSR = 64: 111.5 dB

**Small footprint, high channel density**
- 32-lead 5 mm × 5 mm LFCSP
- Easy Drive features support system-level designs with fewer components

**Enhanced digital functionality**
- First conversion accurate, no latency or pipeline delay
- Fast conversion time and dual-SDO mode allows low SPI clock rates
- Customizable channel sequencer
- On-chip oversampling and decimation
- Threshold detection alerts
- Offset and gain correction
- Autonomous conversion (autocycle) mode
- SPI-/QPSI-/MICROWIRE-/DSP-compatible serial interface

**Low power**
- 17.4 mW at 1 MSPS
- 8.7 mW at 500 kSPS
- 5.7 µW standby power dissipation
- Internal LDO enables 3.15 V to 5.5 V, single analog supply operation
- 1.14 V to 1.98 V logic interface

**Wide operating temperature range: −40°C to +125°C**

## APPLICATIONS

Optical communications
Automatic test equipment
Battery-powered equipment
Precision data acquisition
Instrumentation and process control

## GENERAL DESCRIPTION

The AD4695/AD4696 are compact, high accuracy, low power, 16-channel, 16-bit, 500kSPS/1 MSPS, multiplexed input precision successive approximation register (SAR) analog-to-digital converters (ADC) with Easy Drive features and extensive digital functionality.

The AD4695/AD4696 are optimal for use in space-constrained multichannel precision data acquisition systems and monitoring circuits. The AD4695/AD4696 feature a true 16-bit SAR ADC core with no missing codes, a 16-channel, low crosstalk multiplexer, flexible channel sequencer, overvoltage protection clamp circuits on each analog input, on-chip oversampling and decimation, threshold detection and alert indicators, and an autonomous conversion (autocycle) mode.

The AD4695/AD4696 Easy Drive features relax the drive requirements of the analog front end (AFE) and reference circuitry. Analog input high-Z mode and reference input high-Z mode simplify system designs, reduce component count and increase channel density by removing the need for dedicated high-speed ADC drivers and reference buffers.

Input overvoltage protection clamps on each of the analog inputs protect the AD4695/AD4696 from overvoltage events and prevent overvoltage events on one channel from degrading performance on other channels (see Figure 26).

Advanced digital functionality makes the AD4695/AD4696 compatible with a variety of low power digital hosts. The low serial peripheral interface (SPI) clock rate requirements, on-chip customizable channel sequencers, and oversampling and decimation reduces the burden on the digital host system. Autocycle mode and threshold detection features enable low power, interrupt-driven firmware design by performing conversions autonomously and generating alerts based on channel-specific threshold limits.

The AD4695/AD4696 are available in a 5 mm × 5 mm 32-lead LFCSP with operation specified from −40°C to +125°C.

**Document Feedback**

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.
Tel: 781.329.4700      ©2020 Analog Devices, Inc. All rights reserved.
**Technical Support**                                    **www.analog.com**

# TABLE OF CONTENTS

## FUNCTIONAL BLOCK DIAGRAM



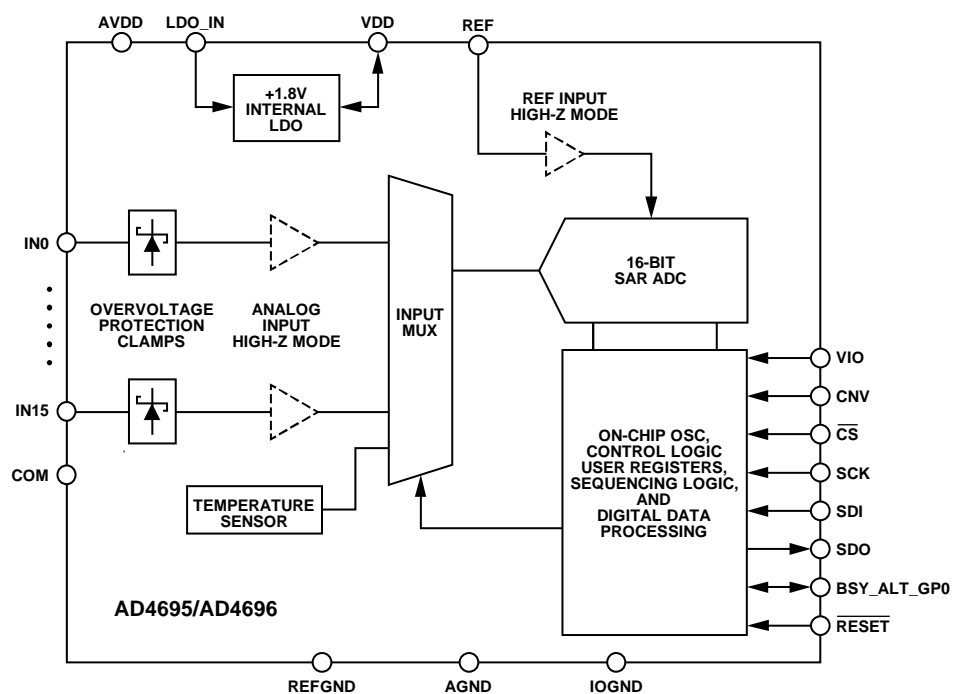*Figure 1. AD4695/AD4696 Block Diagram*

## SPECIFICATIONS

AVDD = 3.15 V to 5.5 V, LDO_IN = 2.4 V to 5.5 V with internal LDO enabled and LDO_IN = AGND with internal LDO disabled, VDD = 1.71 V to 1.89 V with internal LDO disabled, VIO = 1.14 V to 1.98 V, AGND = REFGND = IOGND = 0 V, REFP = reference voltage ($V_{REF}$) = 2.4 V to 5.1 V, sample rate ($f_{CNV}$) = 1 MSPS for the AD4696, $f_{CNV}$ = 500 kSPS for the AD4695 digital output load capacitance[2] = 20 pF, autocycle mode disabled, analog input high-Z mode enabled, reference input high-Z mode enabled, no active overvoltage protection clamps, $T_A$ = −40°C to +125°C, unless otherwise noted.

**Table 1.**

| Parameter | Test Conditions/Comments | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| RESOLUTION | | 16 | | | Bits |
| ANALOG INPUT[1,2] | | | | | |
| Input Voltage Range | Positive ADC input voltage (IN+) – negative ADC input voltage (IN−) | | | | |
| | Unipolar mode | 0 | | $+V_{REF}$ | V |
| | Pseudo-bipolar mode | $−V_{REF}/2$ | | $+V_{REF}/2$ | V |
| Operating Input Voltage | | | | | |
| IN+ − REFGND | IN− = REFGND | 0 | | $+V_{REF}$ | V |
| | IN− = COM, odd-numbered input | −0.1 | | $V_{REF}$ + 0.1 | V |
| IN− − REFGND | IN− = COM, odd-numbered input | | | | |
| | Unipolar mode | −0.1 | | $V_{REF}$ + 0.1 | V |
| | Pseudo-bipolar mode | $V_{REF}/2$ − 0.1 | $V_{REF}/2$ | $V_{REF}/2$ + 0.1 | V |
| Common-Mode Rejection Ratio (CMRR) | $f_{IN}$ = 250 kHz, IN− = COM, odd-numbered input | | 62.5 | | dB |
| Analog Input Leakage Current[3] | | | 10 | | nA |
| SAMPLING DYNAMICS | | | | | |
| Sample Rate ($f_{CNV}$) | Autocycle mode disabled | | | 1 | MSPS |
| Autocycle Sample Period | Autocycle mode enabled | | | | |
| | AC_CYC = 0x0 | TBD | 10 | TBD | µs |
| | AC_CYC = 0x1 | TBD | 20 | TBD | µs |
| | AC_CYC = 0x2 | TBD | 40 | TBD | µs |
| | AC_CYC = 0x3 | TBD | 80 | TBD | µs |
| | AC_CYC = 0x4 | TBD | 100 | TBD | µs |
| | AC_CYC = 0x5 | TBD | 200 | TBD | µs |
| | AC_CYC = 0x6 | TBD | 400 | TBD | µs |
| | AC_CYC = 0x7 | TBD | 800 | TBD | µs |
| Aperture Delay | | | 2 | | ns |
| Aperture Jitter | | | 0.5 | | ps rms |
| DC ACCURACY | | | | | |
| No Missing Codes | | 16 | | | Bits |
| Integral Nonlinearity Error (INL) | $V_{REF}$ = 5 V, oversampling ratio = 1 | −1 | ±0.4 | +1 | LSB |
| Differential Nonlinearity Error (DNL) | $V_{REF}$ = 5 V, oversampling ratio = 1 | −0.6 | ±0.3 | +0.6 | LSB |
| Transition Noise | $V_{REF}$ = 5 V | | | | |
| | Oversampling ratio = 1 | | 0.5 | | LSB |
| | Oversampling ratio = 64 | | TBD | | LSB |
| Offset Error[4] | $T_A$ = 25°C | | ±60 | | µV |
| | $T_A$ = −40°C to +125°C | −360 | | +360 | µV |
| Offset Error Match | $T_A$ = 25°C | | ±100 | | µV |
| | $T_A$ = −40°C to +125°C | −270 | | +270 | µV |
| Gain Error[4] | $T_A$ = 25°C | | ±0.002 | | %FS[5] |
| | $T_A$ = −40°C to +125°C | −0.026 | | +0.026 | %FS |
| Gain Error Match[4] | $T_A$ = 25°C | | ±0.004 | | %FS |
| | $T_A$ = −40°C to +125°C | −0.021 | | +0.021 | %FS |

| Parameter | Test Conditions/Comments | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| AC PERFORMANCE | | | | | |
| Dynamic Range | $V_{REF}$ = 5 V | | | | |
| | Oversampling ratio = 1 | | 93.5 | | dB |
| | Oversampling ratio = 4 | | 99.5 | | dB |
| | Oversampling ratio = 16 | | 105.5 | | dB |
| | Oversampling ratio = 64 | | 111.5 | | dB |
| Input RMS Noise | Oversampling ratio = 1 | | 37.4 | | µV rms |
| | Oversampling ratio = 4 | | 18.7 | | µV rms |
| | Oversampling ratio = 16 | | 9.3 | | µV rms |
| | Oversampling ratio = 64 | | 4.7 | | µV rms |
| 1/f Noise | Bandwidth = 0.1 Hz to 10 Hz | | 5 | | µV p-p |
| Signal-to-Noise Ratio (SNR) | $V_{REF}$ = 5 V, $f_{IN}$ = 1 kHz | TBD | 93 | | dB |
| | $V_{REF}$ = 4.096 V, $f_{IN}$ = 1 kHz | | 91.3 | | dB |
| | $V_{REF}$ = 2.5 V, $f_{IN}$ = 1 kHz | | 87 | | dB |
| Total Harmonic Distortion (THD) | $V_{REF}$ = 5 V, $f_{IN}$ = 1 kHz | | −115 | | dB |
| | $V_{REF}$ = 4.096 V, $f_{IN}$ = 1 kHz | | TBD | | dB |
| | $V_{REF}$ = 2.5 V, $f_{IN}$ = 1 kHz | | TBD | | dB |
| Signal-to-Noise-and-Distortion (SINAD) | $V_{REF}$ = 5 V, $f_{IN}$ = 1 kHz | TBD | 93 | | dB |
| | $V_{REF}$ = 4.096 V, $f_{IN}$ = 1 kHz | | 91.3 | | dB |
| | $V_{REF}$ = 2.5 V, $f_{IN}$ = 1 kHz | | 87 | | dB |
| Spurious-Free Dynamic Range (SFDR) | $V_{REF}$ = 5 V | | 121 | | dB |
| | $V_{REF}$ = 2.5 V | | 121 | | dB |
| Channel-to-Channel Isolation | $f_{IN}$ = 100 kHz | | −123 | | dB |
| Channel-to-Channel Memory | $f_{IN}$ = 100 kHz, $f_{CNV}$ = 1 MSPS | | −102 | | dB |
| | $f_{IN}$ = 100 kHz, $f_{CNV}$ = 500 kSPS | | −112 | | dB |
| −3 dB Input Bandwidth | | | 11 | | MHz |
| REFERENCE | | | | | |
| Reference Voltage ($V_{REF}$) Range | | 2.4 | | AVDD + 0.25 | V |
| REF Leakage Current | $V_{REF}$ = 5 V | | | | |
| | No active overvoltage protection clamps | | 175 | | nA |
| | All clamps active, overvoltage reduced current mode disabled | | 383 | | µA |
| | All clamps active, overvoltage reduced current mode enabled | | 6.3 | | µA |
| REF Average Input Current | $V_{REF}$ = AVDD = 5 V | | | | |
| Reference high-Z mode disabled | $f_{CNV}$ = 10 kSPS, unipolar mode | | 3.8 | | µA |
| | $f_{CNV}$ = 500 kSPS, unipolar mode | | 160 | | µA |
| | $f_{CNV}$ = 1 MSPS, unipolar mode | | 320 | | µA |
| | $f_{CNV}$ = 10 kSPS, pseudo-bipolar mode | | 3.9 | | µA |
| | $f_{CNV}$ = 500 kSPS, pseudo-bipolar mode | | 195 | | µA |
| | $f_{CNV}$ = 1 MSPS, pseudo-bipolar mode | | 390 | | µA |
| Reference high-Z mode enabled | $f_{CNV}$ = 10 kSPS, unipolar mode | | TBD | | µA |
| | $f_{CNV}$ = 500 kSPS, unipolar mode | | 6 | | µA |
| | $f_{CNV}$ = 1 MSPS, unipolar mode | | 12 | | µA |
| | $f_{CNV}$ = 10 kSPS, pseudo-bipolar mode | | TBD | | µA |
| | $f_{CNV}$ = 500 kSPS, pseudo-bipolar mode | | 11 | | µA |
| | $f_{CNV}$ = 1 MSPS, pseudo-bipolar mode | | 22 | | µA |

| Parameter | Test Conditions/Comments | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| TEMPERATURE SENSOR | | | | | |
| Temperature Sensor Voltage | $T_A$ = 25°C | | 680 | | mV |
| | $T_A$ = 0°C | | 725 | | mV |
| Temperature Sensitivity | $T_A$ = −40°C to +125°C | | −1.8 | | mV/°C |
| OVERVOLTAGE CLAMP | | | | | |
| External Series Resistance ($R_{EXT}$)[6] | For stable clamp operation | | | | |
| | Overvoltage reduced current mode disabled | | | 2000 | Ω |
| | Overvoltage reduced current mode enabled | | | 1000 | Ω |
| External Series Capacitance ($C_{EXT}$)[6] | For stable clamp operation | 500 | | | pF |
| Clamp Input Current | For each active clamp | | | 5 | mA |
| Clamp Activation Voltage | | | | $V_{REF}$ + 0.55 | V |
| Clamp Deactivation Voltage | | $V_{REF}$ + 0.1 | | | V |
| Input Clamping Voltage | Clamp current = 5 mA | | $V_{REF}$ + 0.2 | | V |
| Activation Time | | | TBD | | ns |
| Deactivation Time | | | TBD | | ns |
| DIGITAL INPUTS | | | | | |
| Logic Levels | | | | | |
| Input Low Voltage ($V_{IL}$) | | −0.3 | | 0.3 × VIO | V |
| Input High Voltage ($V_{IH}$) | | 0.7 × VIO | | 5.5 | V |
| Input Current ($I_{IL}$) | | −1 | | +1 | µA |
| Input Pin Capacitance | | | TBD | | pF |
| DIGITAL OUTPUTS | | | | | |
| Conversion Mode Data Format | Unipolar mode | | Straight Binary | | |
| | Pseudo-bipolar mode | | Two's Complement | | |
| Logic Levels | | | | | |
| Output Low Voltage ($V_{OL}$) | Digital output current = +500 µA | | | 0.4 | V |
| Output High Voltage ($V_{OH}$) | Digital output current = −500 µA | VIO − 0.3 | | | V |
| POWER REQUIREMENTS | | | | | |
| AVDD − AGND | | 3.15 | | 5.5 | V |
| LDO_IN − AGND | Internal LDO enabled | 2.4 | | 5.5 | V |
| | Internal LDO disabled | | 0 | | V |
| VDD − AGND | Internal LDO disabled | 1.71 | 1.8 | 1.89 | V |
| VIO − IOGND | | 1.14 | | 1.98 | V |
| POWER SUPPLY CURRENT[7] | | | | | |
| Standby Current | | | | | |
| AVDD | AVDD = 5 V | | 160 | | nA |
| LDO_IN | LDO_IN = 5 V | | | | |
| | Internal LDO enabled | | 10 | | µA |
| | Internal LDO disabled | | −1 | | µA |
| VDD | VDD = 1.8 V, internal LDO disabled | | 1.5 | | µA |
| VIO | VIO = 1.8 V | | 250 | | nA |
| AVDD Current (Conversion Mode) | AVDD = 5 V | | | | |
| Reference high-Z mode disabled, analog input high-Z mode disabled | $f_{CNV}$ = 10 kSPS | | 0.77 | | µA |
| | $f_{CNV}$ = 500 kSPS | | TBD | | µA |
| | $f_{CNV}$ = 1 MSPS | | 51 | | µA |
| Reference high-Z mode enabled, analog input high-Z mode enabled | $f_{CNV}$ = 10 kSPS | | 13 | | µA |
| | $f_{CNV}$ = 500 kSPS | | TBD | TBD | µA |

| Parameter | Test Conditions/Comments | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| | $f_{CNV}$ = 1 MSPS | | 1.28 | 1.45 | mA |
| LDO_IN Current (Conversion Mode) | LDO_IN = 5 V, internal LDO enabled | | | | |
| Reference high-Z mode disabled, analog input high-Z mode disabled | $f_{CNV}$ = 10 kSPS | | 52.4 | | μA |
| | $f_{CNV}$ = 500 kSPS | | TBD | | mA |
| | $f_{CNV}$ = 1 MSPS | | 4.0 | | mA |
| Reference high-Z mode enabled, analog input high-Z mode enabled | $f_{CNV}$ = 10 kSPS | | 65 | | μA |
| | $f_{CNV}$ = 500 kSPS | | TBD | TBD | mA |
| | $f_{CNV}$ = 1 MSPS | | 5.2 | 6.5 | mA |
| VDD Current (Conversion Mode) | VDD = 1.8 V, internal LDO disabled | | | | |
| Reference high-Z mode disabled, analog input high-Z mode disabled | $f_{CNV}$ = 10 kSPS | | 41.7 | | μA |
| | $f_{CNV}$ = 500 kSPS | | TBD | | mA |
| | $f_{CNV}$ = 1 MSPS | | 3.9 | | mA |
| Reference high-Z mode enabled, analog input high-Z mode enabled | $f_{CNV}$ = 10 kSPS | | 52.7 | | μA |
| | $f_{CNV}$ = 500 kSPS | | TBD | TBD | mA |
| | $f_{CNV}$ = 1 MSPS | | 5.0 | 6.3 | mA |
| VIO Dynamic Current | VIO = 1.8 V | | | | |
| Register Configuration Mode | Streaming mode, SCK frequency ($f_{SCK}$) = 50 MHz, | | 126.6 | | μA |
| Conversion Mode | Status bits enabled | | | | |
| Single- SDO Mode | $f_{CNV}$ = 10 kSPS | | 3.3 | | μA |
| | $f_{CNV}$ = 500 kSPS | | TBD | | μA |
| | $f_{CNV}$ = 1 MSPS | | 362.5 | 460 | μA |
| Dual-SDO Mode | $f_{CNV}$ = 10 kSPS | | TBD | | μA |
| | $f_{CNV}$ = 500 kSPS | | TBD | | μA |
| | $f_{CNV}$ = 1 MSPS | | TBD | | μA |
| POWER DISSIPATION[7] | AVDD = 5 V, VIO = 1.8 V | | | | |
| Standby Power Dissipation | | | | | |
| Internal LDO disabled | VDD = 1.8 V | | 4 | | μW |
| Internal LDO enabled | LDO_IN = 5 V | | TBD | | μW |
| Power Dissipation, Internal LDO Disabled | LDO_IN = AGND, VDD = 1.8 V | | | | |
| Reference high-Z mode disabled, analog input high-Z mode disabled | $f_{CNV}$ = 10 kSPS | | 85 | | μW |
| | $f_{CNV}$ = 500 kSPS | | TBD | | mW |
| | $f_{CNV}$ = 1 MSPS | | 7.9 | | mW |
| Reference high-Z mode enabled, analog input high-Z mode enabled | $f_{CNV}$ = 10 kSPS | | 165.8 | | μW |
| | $f_{CNV}$ = 500 kSPS | | TBD | TBD | mW |
| | $f_{CNV}$ = 1 MSPS | | 7.1 | 8.1 | mW |
| Power Dissipation, Internal LDO Enabled | LDO_IN = 5 V | | | | |
| Reference high-Z mode disabled, analog input high-Z mode disabled | $f_{CNV}$ = 10 kSPS | | 271.8 | | μW |
| | $f_{CNV}$ = 500 kSPS | | TBD | | mW |
| | $f_{CNV}$ = 1 MSPS | | 20.9 | | mW |
| Reference high-Z mode enabled, analog input high-Z mode enabled | $f_{CNV}$ = 10 kSPS | | 0.4 | | mW |
| | $f_{CNV}$ = 500 kSPS | | TBD | TBD | mW |
| | $f_{CNV}$ = 1 MSPS | | 33.1 | 40.6 | mW |
| Autocycle Mode Power Dissipation | LDO_IN = 5 V, internal LDO enabled, autocycle mode enabled | | | | |
| | AC_CYC = 0x0 | | 2.3 | | mW |
| | AC_CYC = 0x7 | | 0.2 | | mW |

| Parameter | Test Conditions/Comments | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| TEMPERATURE RANGE |  |  |  |  |  |
| Specified Performance | $T_{MIN}$ to $T_{MAX}$ | −40 |  | +125 | °C |

[1] See the Channel Configuration Options section for a detailed description of unipolar mode and pseudo-bipolar mode, and the channel pin assignment options.

[2] IN+ and IN− represent the analog inputs connected to the positive and negative inputs of the AD4695/AD4696 ADC core via the internal multiplexer (see the Multiplexer section and Channel Configuration Options section).

[3] The analog input leakage current specification refers to the input current of the analog input pins during periods when the ADC is not performing conversions and the analog input voltage is already settled.

[4] Offset and gain error specifications taken with the offset and gain correction registers set to their default values, corresponding to no offset or gain correction. See the Offset and Gain Correction section for more information.

[5] %FS is the percentage of the ADC full-scale (see the Transfer Function section for a definition of full-scale).

[6] $R_{EXT}$ and $C_{EXT}$ refer to the resistor and capacitor that make up the recommended external RC filters at the analog inputs, respectively (see the External RC Filter section).

[7] For the power supply current and power dissipation specifications where the analog input high-Z mode is enabled, the analog input high-Z mode is set to be enabled for all channels. The power consumption scales with the percentage of conversions performed with analog input high-Z mode enabled.

## TIMING SPECIFICATIONS

$AVDD^1$ = 3.15 V to 5.5 V, LDO_IN = 2.4 V to 5.5 V with internal LDO enabled and LDO_IN = AGND with internal LDO disabled, VDD = 1.71 V to 1.89 V with internal LDO disabled, VIO = 1.14 V to 1.98 V, AGND = REFGND = IOGND = 0 V, $REF^1$ = reference voltage ($V_{REF}$) = 2.4 V to 5.1 V, sample rate ($f_{CNV}$) = 1 MSPS for the AD4696, $f_{CNV}$ = 500 kSPS for the AD4695, digital output load capacitance[2] = 20 pF, autocycle mode disabled, no active overvoltage protection clamps, $T_A$ = −40°C to +125°C, unless otherwise noted.

**Table 2.**

| Parameter | Symbol | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| Conversion Time | $t_{CONVERT}$ |  | 365 | TBD | ns |
| Acquisition Time | $t_{ACQ}$ |  |  |  |  |
| Two-Cycle Command Mode, Standard Sequencer, or Advanced Sequencer Enabled |  | 715 |  |  | ns |
| Single-Cycle Command Mode[3] Enabled |  |  |  |  |  |
| CNV Period (Time Between Conversions) | $t_{CYC}$ |  |  |  |  |
| AD4696, Autocycle Mode Disabled |  | 1000 |  |  | ns |
| AD4695, Autocycle Mode Disabled |  | 2000 |  |  | ns |
| Autocycle Mode Enabled |  |  |  |  |  |
| AC_CYC = 0x0 |  | TBD | 10 | TBD | µs |
| AC_CYC = 0x1 |  | TBD | 20 | TBD | µs |
| AC_CYC = 0x2 |  | TBD | 40 | TBD | µs |
| AC_CYC = 0x3 |  | TBD | 80 | TBD | µs |
| AC_CYC = 0x4 |  | TBD | 100 | TBD | µs |
| AC_CYC = 0x5 |  | TBD | 200 | TBD | µs |
| AC_CYC = 0x6 |  | TBD | 400 | TBD | µs |
| AC_CYC = 0x7 |  | TBD | 800 | TBD | µs |
| CNV High Time | $t_{CNVH}$ | 10 |  |  | ns |
| CNV Low Time | $t_{CNVL}$ | 80 |  |  | ns |
| $\overline{CS}$ High Time | $t_{CSBH}$ | 5 |  |  | ns |
| $\overline{CS}$ Low to Digital Interface Ready Delay | $t_{EN}$ |  |  | 15 | ns |
| $\overline{CS}$ High to SDO High Impedance Delay | $t_{CSBDIS}$ |  |  | 15 | ns |
| SCK Period | $t_{SCK}$ |  |  |  |  |
| Register Configuration Mode |  |  | 40 |  | ns |
| Conversion Mode |  |  | 12.5 |  | ns |
| SCK Low Time | $t_{SCKL}$ |  |  |  |  |
| Register Configuration Mode |  |  | 16 |  | ns |
| Conversion Mode |  |  | 5 |  | ns |
| SCK High Time | $t_{SCKH}$ |  |  |  |  |
| Register Configuration Mode |  |  | 16 |  | ns |
| Conversion Mode |  |  | 5 |  | ns |
| SDI Data Setup Time Prior to SCK Rising Edge | $t_{SSDI}$ | 2 |  |  | ns |
| SDI Data Hold Time After SCK Rising Edge | $t_{HSDI}$ | 2 |  |  | ns |

| Parameter | Symbol | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| SCK Falling Edge to Data Remains Valid Delay | $t_{HSDO}$ | 1.5 | | | ns |
| SCK Falling Edge to Data Valid Delay | $t_{DSDO}$ | | | 10.5 | ns |
| Last SCK Edge to CNV Rising Edge Delay | $t_{SCKCNV}$ | 80 | | | ns |
| Last SCK Rising Edge to $\overline{CS}$ Rising Edge Delay | $t_{SCKCSB}$ | 1 | | | ns |
| CNV Rising Edge to Busy Indicator Rising Edge Delay (Busy Indicator Enabled on General-Purpose Pin) | $t_{CNVBSY}$ | | | 20 | ns |
| CNV Rising Edge to Alert Indicator Transition (Alert Indicator Enabled on General-Purpose Pin) | $t_{CNVALT}$ | | | 425 | ns |
| Busy Indicator Low Time, Autocycle Mode Enabled (Busy Indicator Enabled on General-Purpose Pin) | $t_{ACBSY}$ | | | | |
| AC_CYC = 0x0 | | | TBD | | µs |
| AC_CYC = 0x1 | | | TBD | | µs |
| AC_CYC = 0x2 | | | TBD | | µs |
| AC_CYC = 0x3 | | | TBD | | µs |
| AC_CYC = 0x4 | | | TBD | | µs |
| AC_CYC = 0x5 | | | TBD | | µs |
| AC_CYC = 0x6 | | | TBD | | µs |
| AC_CYC = 0x7 | | | TBD | | µs |
| Register Configuration Mode Setup Time | $t_{REGCONFIG}$ | 20 | | | ns |
| $\overline{RESET}$ Low Time | $t_{RESETL}$ | 10 | | | ns |
| Hardware Reset Delay (VDD Always Supplied) | $t_{HWR\_DELAY}$ | 300 | | | µs |
| Software Reset Delay | $t_{SWR\_DELAY}$ | 300 | | | µs |
| VDD Power-On Reset Delay | $t_{POR\_VDD}$ | | TBD | | µs |
| VIO Power-On Reset Delay (VDD Supplied Externally) | $t_{POR\_VIO1}$ | | TBD | | µs |
| LDO_IN Power-On Reset Delay | $t_{POR\_LDO}$ | | TBD | | ms |
| VIO Power-On Reset Delay (VDD Supplied by Internal LDO) | $t_{POR\_VIO2}$ | | TBD | | ms |
| LDO Wakeup Command Power-On Reset Delay | $t_{WAKEUP\_SW}$ | | TBD | | ms |
| Hardware Reset Delay (Internal LDO Disabled) | $t_{WAKEUP\_HW}$ | | TBD | | ms |

[1] For all specifications, the relative voltages for AVDD, REF, and REFIN inputs follow the operating conditions specified in the reference and power requirements sections of Table 1.
[2] Digital output load capacitance is the equivalent load capacitance on pins defined as digital outputs in Table 5 and Table 6.
[3] The acquisition time for single-cycle command mode depends on the sample rate and SCK frequency (see the Single-Cycle Command Mode section).

## ABSOLUTE MAXIMUM RATINGS

**Table 3.**

| Parameter | Rating |
|---|---|
| Analog Inputs | |
| INx[1], COM to REFGND | −0.3 V to REF + 0.3 V |
| Reference Inputs | |
| REF, REFIN to AGND, REFGND, IOGND | −0.3 V to +6 V |
| REF to REFIN | −6.3 V to +6.3 V |
| Supply Inputs | |
| AVDD, LDO_IN to AGND, REFGND, IOGND | −0.3 V to +6 V |
| VDD, VIO to AGND, REFGND, IOGND | −0.3 V to +2.1 V |
| AVDD to LDO_IN | −6.3 V to +6.3 V |
| AVDD, LDO_IN to REF, REFIN | −6.3 V to +6.3 V |
| VDD, VIO to AVDD, LDO_IN, REF, REFIN | −6.3 V to +2.4 V |
| VDD to VIO | −2.4 V to +2.4 V |
| Ground | |
| AGND, IOGND to REFGND | −0.3 V to +0.3 V |
| Digital Inputs[2] to IOGND | −0.3 V to +6 V |
| Digital Outputs[2] to IOGND | −0.3 V to VIO + 0.3 V |
| Storage Temperature Range | −65°C to +150°C |
| Junction Temperature | 150°C |
| Lead Temperature Soldering | 260°C reflow as per JEDEC J-STD-020 |
| Electrostatic Discharge (ESD) Ratings | |
| Human Body Model | TBD kV |
| Field Induced Charged Device Model | TBD kV |

[1]INx refers to the analog inputs, Pin IN0 through Pin IN15.
[2]See the Pin Configurations and Function Descriptions section for a list of the digital input and digital output pins.

Stresses at or above those listed under Absolute Maximum Ratings may cause permanent damage to the product. This is a stress rating only; functional operation of the product at these or any other conditions above those indicated in the operational section of this specification is not implied. Operation beyond the maximum operating conditions for extended periods may affect product reliability.

### THERMAL RESISTANCE

Thermal resistance values specified in Table 4 were calculated based on JEDEC specifications and must be used in compliance with JESD51-12. The worst case junction temperature is reported.

The $\theta_{JA}$ is highly dependent on the application and board layout. In applications where high maximum power dissipation exists, close attention to thermal board design is required. The $\theta_{JA}$ value may vary depending on printed circuit board (PCB) material, layout, and environmental conditions.

**Table 4. Thermal Resistance**

| Package Type | $\theta_{JA}$ | $\theta_{JC\_TOP}$ | $\Psi_{JT}$ | Unit |
|---|---|---|---|---|
| CP-32-7 | 40.2[1] | 17.5[2] | 0.2[1] | °C/W |
| CB-36-5 | 41.8[1] | 0.1[2] | 0[1] | °C/W |

[1] Simulated values based on JEDEC 2S2P thermal test board with 9 thermal vias in a JEDEC natural convection environment. See JEDEC JESD51.
[2] Simulated values measured to the package top surface with a cold plate attached directly to the package top surface.

To solve for the worst case $T_J$ in that given environment, use the following equation:

$$T_J = \Psi_{JT} \times P + T_{TOP}$$

where:
$\Psi_{JT}$ is the junction to top thermal characterization number as specified in Table 4.
$P$ refers to total power dissipation in the chip (W).
$T_{TOP}$ refers to the package top temperature (°C) and is measured at the top center of the package in environment of the user.

### ESD CAUTION

**ESD (electrostatic discharge) sensitive device.** Charged devices and circuit boards can discharge without detection. Although this product features patented or proprietary protection circuitry, damage may occur on devices subjected to high energy ESD. Therefore, proper ESD precautions should be taken to avoid performance degradation or loss of functionality.

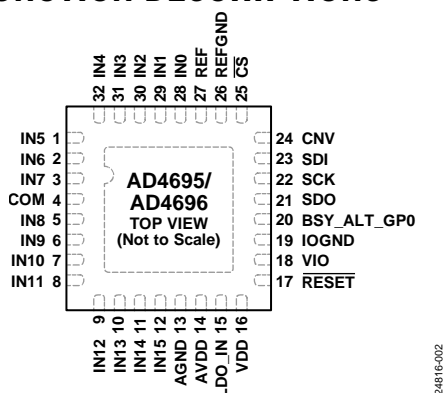## PIN CONFIGURATIONS AND FUNCTION DESCRIPTIONS



*Figure 2. AD4695/AD4696 Pin Configuration*

**Table 5. AD4695/AD4696 Pin Function Descriptions**

| Pin No. | Mnemonic. | Type[1] | Description |
|---|---|---|---|
| 1 | IN5 | AI | Analog Input 5. |
| 2 | IN6 | AI | Analog Input 6. |
| 3 | IN7 | AI | Analog Input 7. |
| 4 | COM | AI | Common Channel Input. IN0–IN15 can be paired with COM so the ADC core samples the differential voltage between them. COM is nominally tied to signal ground (unipolar mode) or $V_{REF}/2$ (pseudo-bipolar mode). See the Channel Configuration Options section for a detailed description on pairing inputs and the unipolar and pseudo-bipolar modes. |
| 5 | IN8 | AI | Analog Input 8. |
| 6 | IN9 | AI | Analog Input 9. |
| 7 | IN10 | AI | Analog Input 10. |
| 8 | IN11 | AI | Analog Input 11. |
| 9 | IN12 | AI | Analog Input 12. |
| 10 | IN13 | AI | Analog Input 13. |
| 11 | IN14 | AI | Analog Input 14. |
| 12 | IN15 | AI | Analog Input 15. |
| 13 | AGND | P | Analog Supply Ground. AVDD, LDO_IN, and VDD are referenced to AGND. |
| 14 | AVDD | P | Analog Power Supply. Nominally TBD V to TBD V. Decouple AVDD with a local 100 nF capacitor to AGND. |
| 15 | LDO_IN | P | Internal LDO Input. Nominally TBD V to TBD V when the internal LDO is enabled. Decouple LDO_IN with a local 100 nF capacitor to AGND. If powering VDD with an external 1.8 V rail, tie LDO_IN to AGND. See the Internal LDO section for more information. |
| 16 | VDD | P | ADC Core Power Supply. Nominally 1.8 V. When powering VDD from the internal LDO, leave VDD floating. When powering VDD from an external rail, disable the internal LDO. Decouple VDD with a local 100 nF capacitor to AGND. |
| 17 | RESET | DI | Hardware Reset Input. Driving this pin low performs a hardware reset of the device, resetting register states to their default values (see the Device Reset section). |
| 18 | VIO | P | Input/Output Interface Digital Power. Nominally at the same supply as the host interface (for example, 1.2 V or 1.8 V). Decouple VIO with a local 100 nF capacitor to IOGND. |
| 19 | IOGND | P | Input/Output Interface Digital Supply Ground. VIO is referenced to IOGND. |
| 20 | BSY_ALT_GP0 | DI/DO | General-purpose pin 0. Can be configured to function as a general-purpose input/output, the threshold detection alert indicator, the busy indicator, or a second serial data output (see the General-Purpose Pin section). |
| 21 | SDO | DO | Serial Data Output. When the device is configured in register configuration mode, SDO is used to read configuration register data during SPI read transactions. When the device is configured in conversion mode, SDO is used to read conversion results. Data output is synchronized to the falling edge of SCK. |
| 22 | SCK | DI | Serial Data Clock Input. SCK is used to clock out data on SDO and clock in data on SDI while the device is configured in either register configuration mode or conversion mode. |
| 23 | SDI | DI | Serial Data Input. When the device is configured in register configuration mode, SDI is used to perform SPI read and write transactions to access the configuration registers. In conversion mode, SDI receives 5-bit commands from the digital host, as shown in Table 14. |

| Pin No. | Mnemonic. | Type[1] | Description |
|---|---|---|---|
| 24 | CNV | DI | Convert Input. When the device is configured in conversion mode, a rising edge on CNV initiates a conversion of the selected analog input. The AD4695/AD4696 can interface to a 4-wire SPI by tying CNV to $\overline{CS}$ (see the Digital Interface Operation section for more information). |
| 25 | $\overline{CS}$ | DI | Chip Select Input. configured in register configuration mode, $\overline{CS}$ frames SPI read and write transactions which accesses the configuration registers. When the device is configured in conversion mode, $\overline{CS}$ can either be held low throughout the entire conversion or can be used to frame SPI transactions reading back conversion results. The AD4695/AD4696 can interface to a 4-wire SPI by tying CNV to $\overline{CS}$ (see the Digital Interface Operation section for more information). |
| 26 | REFGND | P | Reference Ground. REF is referenced to REGND. IN0 to IN15 can be paired with REFGND so the ADC core samples the differential voltage between them. See the Channel Configuration Options section for a detailed description on pairing inputs. |
| 27 | REF | AI | Reference Input. $V_{REF}$ must be provided by an external precision reference voltage between TBD V and TBD V. The REF pin must be decoupled with a minimum 1 μF capacitor for optimal operation. See the Voltage Reference Input section for more information. |
| 28 | IN0 | AI | Analog Input 0. |
| 29 | IN1 | AI | Analog Input 1. |
| 30 | IN2 | AI | Analog Input 2. |
| 31 | IN3 | AI | Analog Input 3. |
| 32 | IN4 | AI | Analog Input 4. |
| 33 | EPAD | NC | Exposed Pad. The exposed pad is not connected internally. For increased reliability of the solder joints, it is recommended that the pad be soldered to the system ground plane. |

[1]AI is analog input, AI/O is analog input/output, DI is digital input, DO is digital output, P is power, and NC is no internal connection.

# TYPICAL PERFORMANCE CHARACTERISTICS

AVDD = LDO_IN = 5 V, VIO = 1.8 V, REF = $V_{REF}$ = 5 V, $f_{CNV}$ = 1 MSPS, $f_{SCK}$ = 50 MHz, unipolar mode, analog input high-Z mode enabled, reference input high-Z mode enabled, internal LDO enabled, $f_{CNV}$ = 1 MSPS for the AD4696, $f_{CNV}$ = 500 kSPS for the AD4695, no active clamps, autocycle mode disabled, OSR = 1, $T_A$ = +25°C, unless otherwise specified.

**TBD**

*Figure 3. INL vs. Code, $V_{REF}$ = 5 V*

**TBD**

*Figure 6. DNL vs. Code, $V_{REF}$ = 5 V*

**TBD**

*Figure 4. INL vs. Code, $V_{REF}$ = 4.096 V*

**TBD**

*Figure 7. DNL vs. Code, $V_{REF}$ = 4.096 V*

**TBD**

*Figure 5. INL vs. Code, $V_{REF}$ = 2.5 V*

**TBD**

*Figure 8. DNL vs. Code, $V_{REF}$ = 2.5 V*

Rev. PrI | Page 13 of 94

*Figure 9. Histogram of a DC Input at Code Center, OSR = 1*

*Figure 12. Histogram of a DC Input at Code Transition, OSR = 1*

*Figure 10. Histogram of a DC Input at Code Center, OSR = 64*

*Figure 13. Histogram of a DC Input at Code Transition, OSR = 64*

*Figure 11. Dynamic Range vs. OSR*

*Figure 14. Analog Input Voltage Step with Analog Input High-Z Mode Disabled and Enabled*

*Figure 15. FFT, $f_{IN}$ = 1 kHz, $V_{REF}$ = 5 V, OSR = 1*

*Figure 18. FFT, $f_{IN}$ = 1 kHz, $V_{REF}$ = 2.5 V, OSR = 1*

*Figure 16. FFT, $f_{IN}$ = 1 kHz, $V_{REF}$ = 5 V, OSR = 64*

*Figure 19. FFT, $f_{IN}$ = 1 kHz, $V_{REF}$ = 2.5 V, OSR = 64*

*Figure 17. SNR, THD, and SINAD vs. Input Frequency*

*Figure 20. THD vs. Source Resistance*

**TBD**

*Figure 21. SNR, SINAD, and ENOB vs. $V_{REF}$, $f_{IN}$ = 1 kHz*

**TBD**

*Figure 24. THD and SFDR vs. $V_{REF}$, $f_{IN}$ = 1 kHz*

**TBD**

*Figure 22. SNR, SINAD, and ENOB vs. Temperature, $f_{IN}$ = 1 kHz*

**TBD**

*Figure 25. THD and SFDR vs. Temperature, $f_{IN}$ = 1 kHz*

**TBD**

*Figure 23. SNR, THD, and SINAD vs. Active Clamps,*
*Reduced Current Mode Disabled*

**TBD**

*Figure 26. SNR, THD, and SINAD vs. Active Clamps,*
*Reduced Current Mode Enabled*

**TBD**

*Figure 27. SINAD vs. $t_{SCKCNV}$*

**TBD**

*Figure 30. Channel-to-Channel Isolation vs. Input Frequency*

**TBD**

*Figure 28. CMRR vs. Input Frequency*

**TBD**

*Figure 31. PSRR vs. Frequency*

**TBD**

*Figure 29. 1/f Noise (0.1 Hz to 10 Hz Bandwidth), X kSPS, X Samples Averaged per Reading*

**TBD**

*Figure 32. Offset Error vs. Temperature*

**TBD**

*Figure 33. Offset Error vs. Number of Active Clamps,*
*Clamp Current = 5 mA, Reduced Current Mode Disabled*

**TBD**

*Figure 36. Offset Error vs. Number of Active Clamps,*
*Clamp Current = 5 mA, Reduced Current Mode Enabled*

**TBD**

*Figure 34. Gain Error vs. Temperature*

**TBD**

*Figure 37. Temperature Sensor Output Voltage vs. Temperature*

**TBD**

*Figure 35. Gain Error vs. Number of Active Clamps,*
*Clamp Current = 5 mA, Reduced Current Mode Disabled*

**TBD**

*Figure 38. Gain Error vs. Number of Active Clamps,*
*Clamp Current = 5 mA, Reduced Current Mode Enabled*

**TBD**

Figure 39. Reference Input Current vs. Sample Rate, $V_{REF}$ = 5 V

**TBD**

Figure 40. Reference Input Current vs. Temperature, $V_{REF}$ = 5 V, $f_{CNV}$ = 1 MSPS

**TBD**

Figure 41. Additional Reference Input Current vs. Number of Active Clamps, Clamp Current = 5 mA, $V_{REF}$ = 5 V

**TBD**

Figure 42. Reference Input Current vs. Reference Voltage, $f_{CNV}$ = 1 MSPS and 500 kSPS

**TBD**

Figure 43. Analog Input Voltage vs. Source Voltage, $R_{EXT}$ = 1 kΩ, $V_{REF}$ = 5 V

**TBD**

Figure 44. Analog Input Current vs. Source Voltage, $R_{EXT}$ = 1 kΩ, $V_{REF}$ = 5 V)

*Figure 45. Power Consumption vs. Sample Rate (Internal LDO Disabled)*

*Figure 48. Power Consumption vs. Sample Rate (Internal LDO Enabled)*

*Figure 46. Power Consumption vs. Temperature, Internal LDO Disabled $f_{CNV} = 1$ MSPS and $f_{CNV} = 500$ kSPS*

*Figure 49. Power Consumption vs. Temperature, Internal LDO Enabled, $f_{CNV} = 1$ MSPS and $f_{CNV} = 500$ kSPS*

*Figure 47. Internal LDO Settling Time vs. $C_{VDD}$*

*Figure 50. Standby Current vs. Temperature*

TBD

*Figure 51. AVDD Current vs. Sample Rate*

TBD

*Figure 54. LDO_IN Current vs. Sample Rate, Internal LDO Enabled*

TBD

*Figure 52. AVDD Current vs. AVDD Voltage*

TBD

*Figure 55. LDO_IN Current vs. LDO_IN Voltage, Internal LDO Enabled*

TBD

*Figure 53. AVDD Current vs. Temperature*

TBD

*Figure 56. LDO_IN Current vs. Temperature, Internal LDO Enabled*

**TBD**

Figure 57. VDD Current vs. Sample Rate, Internal LDO Disabled

**TBD**

Figure 58. VDD Current vs. VDD Voltage, Internal LDO Disabled

**TBD**

Figure 59. VDD Current vs. Temperature, Internal LDO Disabled

**TBD**

Figure 60. VIO Current vs. Sample Rate, Conversion Mode, OSR = 1

**TBD**

Figure 61. VIO Current vs. Temperature, Conversion Mode, OSR = 1

**TBD**

*Figure 62. VIO Current vs. SCK Frequency, Register Configuration Mode, Streaming Mode*

**TBD**

*Figure 63. $t_{DSDO}$ vs. Digital Output Load Capacitance*

# TERMINOLOGY

**Integral Nonlinearity (INL)**
INL is the maximum deviation from a straight line passing through the endpoints of the ADC transfer function. The endpoints of the transfer function are zero scale, a point ½ LSB below the first code transition, and full scale, a point ½ LSB above the last code transition.

**Differential Nonlinearity (DNL)**
In an ideal ADC, code transitions are 1 LSB apart. DNL is the maximum deviation from this ideal value. It is often specified in terms of resolution for which no missing codes are guaranteed.

**Offset Error**
The offset error is the deviation of the measured transition between −FSR and −FSR+1 from the ideal transition, measured in volts. The ideal transition between −FSR and −FSR+1 occurs at an analog input level ½ LSB above the IN− voltage (see the Transfer Function section).

**Offset Error Match**
Offset error match is the difference in offset error between any two input channels.

**Gain Error**
The gain error is the deviation of the measured transition between +FSR−1 and +FSR from the ideal transition, measured in percentage of full-scale (%FS). The ideal transition between +FSR−1 and +FSR occurs for an analog input level 1½ LSB below the nominal full scale (see the Transfer Function section).

**Gain Error Match**
Gain error match is the difference in gain error between any two input channels.

**Spurious-Free Dynamic Range (SFDR)**
SFDR is the difference, in decibels (dB), between the rms amplitude of the input signal and the peak spurious signal.

**Effective Number of Bits (ENOB)**
ENOB is a measurement of the resolution with a sine wave input. It is related to SINAD by the following formula:

$$ENOB = (SINAD_{dB} − 1.76)/6.02$$

ENOB is expressed in bits.

**Noise Free Code Resolution**
Noise free code resolution is the number of bits beyond which it is impossible to distinctly resolve individual codes. Calculate it as follows:

$$Noise\ Free\ Code\ Resolution = \log_2 (2^N/Peak\ to\ Peak\ Noise)$$

Noise free code resolution is expressed in bits.

**Dynamic Range**
Dynamic range is the ratio of the rms value of the full scale to the total rms noise measured with the inputs shorted together. The value for dynamic range is expressed in decibels (dB). It is measured with a signal at −60 dBFS to include all noise sources and DNL artifacts.

**Signal-to-Noise Ratio (SNR)**
SNR is the ratio of the rms value of the actual input signal to the rms sum of all other spectral components below the Nyquist frequency, excluding harmonics and dc. The value for SNR is expressed in decibels (dB).

**Signal-to-Noise-and-Distortion (SINAD) Ratio**
SINAD is the measured ratio of signal-to-noise-and-distortion at the output of the ADC. The signal is the rms amplitude of the fundamental. Noise is the sum of all nonfundamental signals up to half the sampling frequency ($f_S/2$), excluding dc.

The ratio is dependent on the number of quantization levels in the digitization process; the more levels, the smaller the quantization noise. The theoretical SINAD ratio for an ideal N-bit converter with a sine wave input is given by

$$SINAD = (6.02N + 1.76)\ dB$$

Thus, for an ideal 12-bit converter, the SINAD ratio is 74 dB.

**Channel-to-Channel Memory**
Channel-to-channel memory is a measure of the level of crosstalk that occurs when switching between channels in a channel sequence. It is measured by applying a full-scale, 100 kHz signal to one analog input channel and a dc voltage on another analog input channel, and repeatedly switching between the two channels between each conversion. The channel-to-channel memory is the magnitude at 100 kHz in the spectrum measured from the dc channel data.

**Channel-to-Channel Isolation**
Channel-to-channel isolation is a measure of the level of crosstalk from a signal on an inactive channel to an active channel. It is measured by applying a dc input to one analog input channel and a full-scale, 100 kHz sine wave signal to all other analog input channels and performing conversions only on the dc input channel. The channel-to-channel isolation is the magnitude at 100 kHz in the spectrum measured from the dc channel data.

**Total Harmonic Distortion (THD)**
THD is the ratio of the rms sum of harmonics to the fundamental. It is defined as

$$THD(dB) = 20\log\frac{\sqrt{V_2^2 + V_3^2 + V_4^2 + V_5^2 + V_6^2}}{V_1}$$

where:
$V_1$ is the rms amplitude of the fundamental.
$V_2$, $V_3$, $V_4$, $V_5$, and $V_6$ are the rms amplitudes of the second through the sixth harmonic.

**Aperture Delay**
Aperture delay is the measure of the acquisition performance. It is the time between the rising edge of the CNV input and when the input signal is held for a conversion.

# THEORY OF OPERATION

## OVERVIEW

The AD4695/AD4696 are low power, 16-channel, 16-bit, 500 kSPS/1 MSPS multiplexed precision SAR ADCs. The AD4695/AD4696 offer valid first conversion results even after being idle for long periods of time.

The AD4695/AD4696 include features that simplify the design requirements of peripheral circuitry, and facilitate high performance data acquisition system designs with low power consumption and high channel density:

- 16-bit SAR ADC core with no missing codes
- 16 multiplexed analog inputs with low crosstalk multiplexer
- Flexible channel sequencing modes
- Analog input and reference high-Z mode
- Temperature sensor
- Input overvoltage protection clamps on each analog input
- Programmable threshold detection for each analog input
- Autocycle mode for performing conversions autonomously
- First-order offset and gain correction for each analog input
- Oversampling and decimation options for each analog input

When multiplexing between channels, the analog input high-Z mode feature reduces the nonlinear voltage steps that occur at the analog inputs. Analog input high-Z mode relaxes settling and bandwidth requirements of the analog front-end circuitry and allows lower bandwidth and lower power amplifiers to drive the analog inputs directly.

The reference input high-Z mode feature significantly reduces the REF input current while the ADC core performs conversions to relax the drive requirements of the reference circuitry. This feature allows the use of lower power references and smaller reference decoupling capacitors (1 μF) than with traditional SAR ADCs.

Each analog input is equipped with input overvoltage protection clamps to protect the device from overvoltage events. The circuits of the clamps are robust and prevent overvoltage events on an analog input from significantly impacting the performance of the other analog inputs.

The AD4695/AD4696 include a variety of channel sequencing modes that provide a flexible means of performing conversions on a sequence of analog input channels. The standard sequencer and advanced sequencer allow a channel sequence to be preprogrammed and automatically progressed as conversions occur. Two-cycle command mode and single-cycle command mode allow the digital host to manually select from the channels with SPI commands.

The AD4695/AD4696 have an enhanced digital interface that is used to access its register contents and initiate and read conversion results, while providing additional utility. Register configuration mode is used to read and write to the register contents. Conversion mode is used to initiate conversions and read back conversion results. The fast conversion time of the AD4695/AD4696 allows low serial clock rates to read back conversions even when running at full throughput. The AD4695/AD4696 supports 4-wire SPI protocol and has optional dual-SDO mode that enable slower SCK rates by shifting out conversion results on multiple data outputs in parallel.

The power consumption of the AD4695/AD4696 scales with throughput because its ADC core powers down between conversions. When operating at 10 kSPS, for example, the AD4695/AD4696 typically consumes 0.1 mW (with internal LDO, analog input high-Z mode and reference high-Z mode disabled), making it suitable for battery-powered applications.

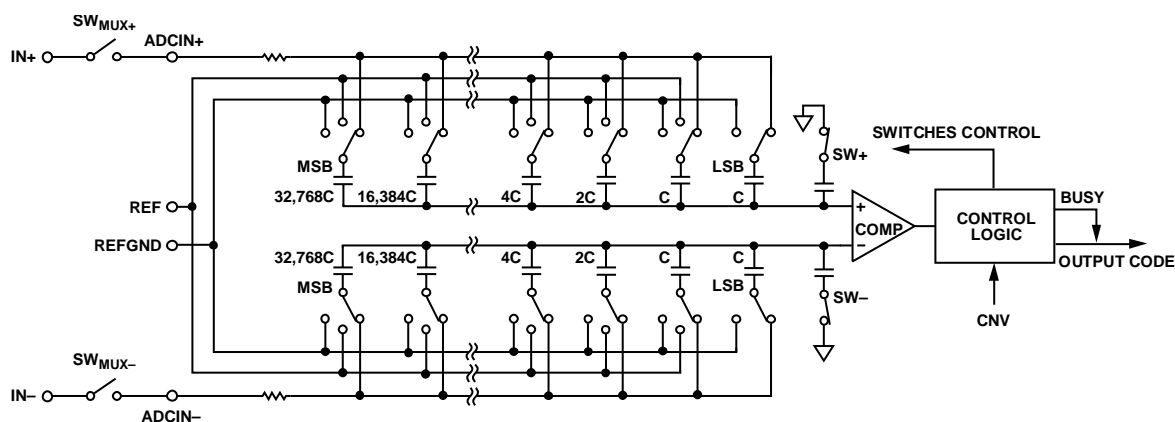The AD4695/AD4696 are available in a 32-lead, 5 mm × 5 mm LFCSP.



Figure 64. ADC Simplified Schematic

## CONVERTER OPERATION

The AD4695/AD4696 contains a SAR-based ADC core which utilizes a charge redistribution digital-to-analog-converter (DAC) to quantize the applied input voltage to an output code. The charge redistribution DAC consists of two identical arrays of 16 binary-weighted capacitors, which are connected to the inputs of a comparator. Figure 64 shows a simplified schematic of the SAR ADC core.

The capacitor arrays are used as sampling capacitors and acquire the analog signal on the selected analog input. The analog inputs and the temperature sensor are connected to the ADC core inputs (ADCIN+ and ADCIN−) via the internal low-crosstalk multiplexer, represented by $SW_{MUX+}$ and $SW_{MUX-}$ in Figure 64. The multiplexer is controlled by the channel sequencing logic to route the desired analog input channel to the ADC core for acquisition and conversion (see the Multiplexer section and the Channel Sequencing Modes section).

The AD4695/AD4696 conversion process includes an acquisition phase and a conversion phase. While in conversion mode, a rising edge on the CNV input initiates the conversion phase and ends the acquisition phase. The conversion result is available to read after the conversion phase is complete, and the delay between the CNV rising edge and data being ready is given by $t_{CONVERT}$ in Table 2. The ADC core enters the acquisition phase before the conversion phase is complete and remains in the acquisition phase until the next CNV rising edge.

During the acquisition phase, the ADC core capacitor arrays track the voltage on the analog input channel selected by the multiplexer. In the acquisition phase, terminals of the array tied to the input of the comparator are connected to REFGND through the SW+ and SW− switches. All switches on the individual capacitors in the array are connected to ADCIN+ and ADCIN−, and ADCIN+ and ADCIN− are connected to the selected analog input channel through $SW_{MUX+}$ and $SW_{MUX-}$.

The conversion phase is initiated on the CNV rising edge (in conversion mode only). When the conversion phase begins, SW+, SW−, $SW_{MUX+}$ and $SW_{MUX-}$ are opened first, sampling the analog input voltage on the capacitor arrays. The two capacitor arrays are then disconnected from ADCIN+ and ADCIN− and connected to REFGND. The sampled voltage is applied to the comparator inputs, causing the comparator to become unbalanced. By switching each element of the capacitor array between REFGND and REF in sequence, the comparator input varies by binary-weighted voltage steps ($V_{REF}/2$, $V_{REF}/4$, …, $V_{REF}/65,536$). The control logic toggles the switches for each capacitor in the array, starting with the MSB, to bring the comparator back into a balanced condition. The state of the comparator is recorded for each binary-weighted voltage step (also known as a bit trial) to produce an ADC output code.

The SAR ADC core generates one output code for each CNV rising edge (when in conversion mode). Multiple output codes are averaged together to generate an oversampled ADC result when the active channel is configured with an oversampling ratio setting greater than 1 (see the Transfer Function and Oversampling and Decimation sections).

During the conversion phase, the ADC generates a busy indicator to communicate to the digital host when a conversion is complete and ready to be read via the SPI (see the Busy Indicator section).

After the conversion process is complete, the ADC returns to the acquisition phase to acquire the signal on the next analog input channel. SW+ and SW− close, the capacitor array switches connect the capacitor terminals to ADCIN+ and ADCIN−, and $SW_{MUX+}$ and $SW_{MUX-}$ are connected to the next analog input channel.

The AD4695/AD4696 ADC core is controlled by an internal clock, and the SPI serial clock (SCK) is not required for the conversion process.

## TRANSFER FUNCTION

Figure 65 shows the AD4695/AD4696 data processing path. The SAR ADC core generates one 16-bit output code per conversion period. The oversampling ratio setting for the selected analog input channel determines how many consecutive 16-bit output codes results are averaged, and then the offset and gain correction settings are applied to generate the final result to be read over the SPI in conversion mode (see the Oversampling and Decimation section and the Offset and Gain Correction section).

The conversion result length is determined by the oversample ratio (OSR) setting. The conversion result resolution can range from 16- to 19-bits for an OSR of 1 and 64, respectively (see the Oversampling and Decimation section).

The conversion result encoding format is determined by the selected polarity mode. The results are in straight binary format for channels configured in unipolar mode, and twos complement for channels configured in pseudo-bipolar mode (see the Channel Configuration Options section).

The AD4695/AD4696 include offset and gain correction for each channel that can be configured to compensate for first-order system errors. The offset and gain correction registers modify the ADC transfer function digitally (see the Offset and Gain Correction section).

The ideal transfer function is shown in Figure 66. The Converting Between Codes and Volts section describes the relationship between output codes and input voltages versus $V_{REF}$, oversampling ratio, polarity modes and offset and gain correction settings. Table 6 through Table 9 show examples of different voltage inputs and corresponding results for each oversampling ratio and polarity mode option assuming an ideal ADC transfer function and with the offset and gain correction values set to their default values.
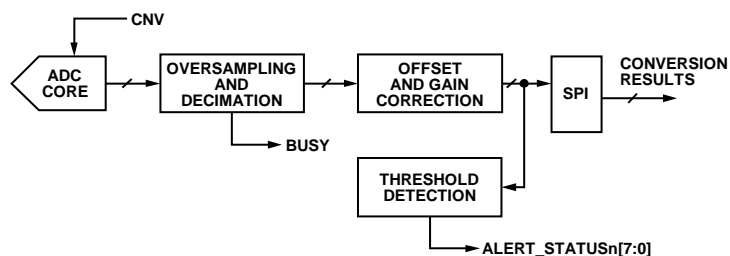
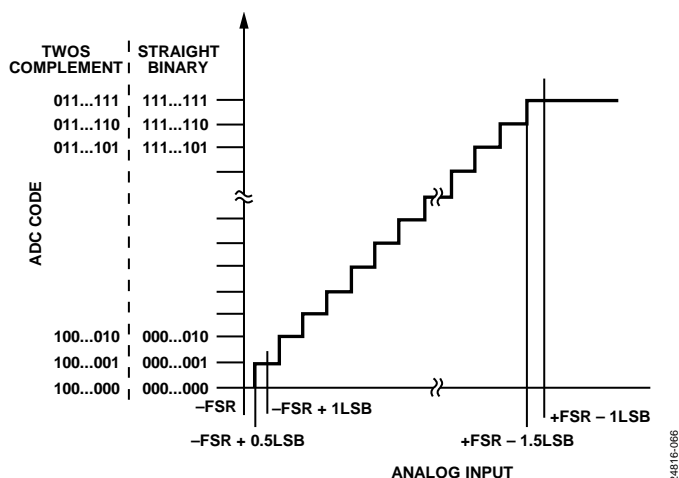*Figure 65. AD4695/AD4696 ADC Data Processing Path*



*Figure 66. ADC Ideal Transfer Function (FSR Is Full-Scale Range)*

**Table 6. Output Codes and Ideal Input Voltages, $V_{REF}$ = 5 V, Oversampling Ratio = 1**

| Description | Input Voltage in Unipolar Mode | Digital Output Code (Straight Binary Hex) | Input Voltage in Pseudo-Bipolar Mode | Digital Output Code (Twos Complement Hex) |
|---|---|---|---|---|
| FSR − 1 LSB | 4.999924 V | 0xFFFF | 2.499924 V | 0x7FFF |
| Midscale + 1 LSB | 2.500076 V | 0x8001 | 76.3 µV | 0x0001 |
| Midscale | 2.5 V | 0x8000 | 0 V | 0x0000 |
| Midscale − 1 LSB | 2.499924 V | 0x7FFF | −76.3 µV | 0xFFFF |
| −FSR + 1 LSB | 76.3 µV | 0x0001 | −2.499924 V | 0x8001 |
| −FSR | 0 V | 0x0000 | −2.5 V | 0x8000 |

**Table 7. Output Codes and Ideal Input Voltages, $V_{REF}$ = 5 V, Oversampling Ratio = 4**

| Description | Input Voltage in Unipolar Mode | Digital Output Code (Straight Binary Hex) | Input Voltage in Pseudo-Bipolar Mode | Digital Output Code (Twos Complement Hex) |
|---|---|---|---|---|
| FSR − 1 LSB | 4.999962 V | 0x1FFFF | 2.499962 V | 0x0FFFF |
| Midscale + 1 LSB | 2.500038 V | 0x10001 | 38.1 µV | 0x00001 |
| Midscale | 2.5 V | 0x10000 | 0 V | 0x00000 |
| Midscale − 1 LSB | 2.499962 V | 0x0FFFF | −38.1 µV | 0x1FFFF |
| −FSR + 1 LSB | 38.1 µV | 0x00001 | −2.499962 V | 0x10001 |
| −FSR | 0 V | 0x00000 | −2.5 V | 0x10000 |

**Table 8. Output Codes and Ideal Input Voltages, $V_{REF}$ = 5 V, Oversampling Ratio = 16**

| Description | Input Voltage in Unipolar Mode | Digital Output Code (Straight Binary Hex) | Input Voltage in Pseudo-Bipolar Mode | Digital Output Code (Twos Complement Hex) |
|---|---|---|---|---|
| FSR − 1 LSB | 4.999981 V | 0x3FFFF[3] | 2.499981 | 0x1FFFF |
| Midscale + 1 LSB | 2.500019 V | 0x20001 | 19.1 µV | 0x00001 |
| Midscale | 2.5 V | 0x20000 | 0 V | 0x00000 |
| Midscale − 1 LSB | 2.499981 V | 0x1FFFF | −19.1 µV | 0x3FFFF |
| −FSR + 1 LSB | 19.1 µV | 0x00001 | −2.499981 | 0x20001 |
| −FSR | 0 V | 0x00000[4] | −2.5 | 0x20000 |

**Table 9. Output Codes and Ideal Input Voltages, $V_{REF}$ = 5 V, Oversampling Ratio = 64**

| Description | Input Voltage in Unipolar Mode | Digital Output Code (Straight Binary Hex) | Input Voltage in Pseudo-Bipolar Mode | Digital Output Code (Twos Complement Hex) |
|---|---|---|---|---|
| FSR − 1 LSB | 4.999910 V | 0x7FFFF[3] | 2.499990 V | 0x3FFFF |
| Midscale + 1 LSB | 2.500010 V | 0x40001 | 9.54 µV | 0x00001 |
| Midscale | 2.5 V | 0x40000 | 0 V | 0x00000 |
| Midscale − 1 LSB | 2.499990 V | 0x3FFFF | −9.54 µV | 0x7FFFF |
| −FSR + 1 LSB | 9.54 µV | 0x00001 | −2.499990 V | 0x40001 |
| −FSR | 0 V | 0x00000[4] | −2.5 V | 0x40000 |

## ANALOG INPUTS

Figure 67 shows an equivalent circuit of the AD4695/AD4696 analog inputs (IN0 thru IN15 and COM).
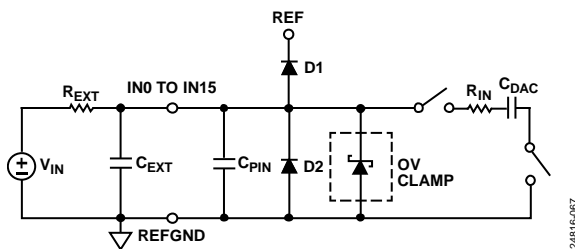


*Figure 67. Equivalent Analog Input Circuit*

A low crosstalk analog multiplexer routes the signals from the analog input pins to the ADC core inputs. The impedance of the analog inputs are modeled as the parallel combination of the pin capacitance ($C_{PIN}$) and the network formed by the series connection of $R_{IN}$ and $C_{DAC}$. $R_{IN}$ represents the ADC input series resistance and the multiplexer switch resistance and is typically 240 Ω. $C_{DAC}$ represents the ADC sampling capacitive DAC shown in Figure 64, and is typically 60 pF.

Each analog input has a unique overvoltage protection clamp circuit, represented by OV CLAMP in Figure 67. The clamps protect the analog inputs from dc overvoltage conditions and eliminate the need for additional external protection diodes. See the Input Overvoltage Protection Clamps section for a detailed description of the overvoltage protection clamps.

$R_{EXT}$ and $C_{EXT}$ in Figure 67 represent an external, resistor-capacitor (RC) low-pass filter, which is included in the system design to limit the bandwidth of the input signal. $R_{EXT}$ can also be used to improve overvoltage protection of the analog inputs. See the External RC Filter section for detailed descriptions of the $R_{EXT}$ and $C_{EXT}$ functions.

### Multiplexer

The AD4695/AD4696 contain a flexible, low crosstalk analog multiplexer for selecting from the 16 analog inputs and internal temperature sensor and routing them to the inputs of the 16-bit, pseudo-differential SAR ADC core. Figure 68 shows a simplified schematic of the internal multiplexer. The $SW_{MUX+}$ and $SW_{MUX−}$ switches shown in Figure 64 and Figure 68 represent the multiplexer switches that route the selected channel to the ADC inputs (labeled ADCIN+ and ADCIN− in Figure 64). $SW_{MUX+}$ and $SW_{MUX−}$ are break-before-make and are controlled by the internal channel sequencing logic (see the Channel Sequencing Modes section).

The multiplexer allows flexible analog input channel configuration. The $SW_{MUX−}$ position is user programmable, and can be assigned to any of the pins shown in Figure 68 (see the Channel Configuration Options section).
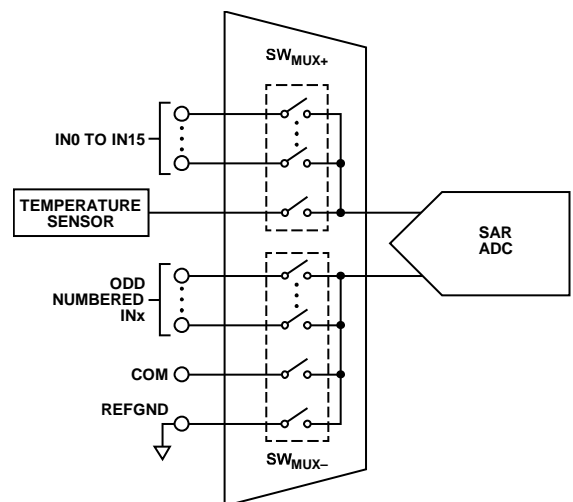


*Figure 68. Multiplexer Simplified Schematic*

### Channel Configuration Options

The AD4695/AD4696 features several channel configuration options that allow it to interface with a variety of signals. The channel configuration can be independently programmed for each of the 16 analog inputs (IN0 through IN15).

The channel configuration settings include pin-pairing assignments and signal polarity modes. The pin-pairing options assign the position of SW$_{MUX-}$ for each position of SW$_{MUX+}$, thereby determining which signal is routed to the negative-side of the SAR ADC core (ADCIN− in Figure 64). The signal polarity modes configure the ADCIN− voltage range. Figure 69 shows the pin-pairing and voltage ranges for the different channel configuration options.

The pin-pairing assignment options include:

- Figure 69 (a): IN0 thru IN15 paired with REFGND
- Figure 69 (b): IN0 thru IN15 paired with COM
- Figure 69 (c): Even-numbered input paired with the next-highest odd-numbered input (i.e. IN0 with IN1, IN2 with IN3, and so on)

The two signal polarity modes are called unipolar mode and pseudo-bipolar mode. When a channel is in unipolar mode, the signal routed to ADCIN− is nominally 0 V (relative to REFGND). When a channel is in pseudo-bipolar mode, the signal routed to ADCIN− is nominally V$_{REF}$/2 V (relative to REFGND). The valid operating input voltage specification for unipolar and pseudo-bipolar modes are shown in Table 1.

When an input is configured in unipolar mode, its output codes are in straight binary format. When an input is configured in pseudo-bipolar mode, its output codes are in two's complement format. See the Transfer Function section for an example of the output code formatting for both unipolar and pseudo-bipolar modes.

The pin-pairing assignments are selected with the IN_PAIR field in the CONFIG_INn registers. The signal polarity modes are selected with the IN_MODE field in the CONFIG_INn registers.

When an even-numbered input is paired with its corresponding odd-numbered input, selecting the odd-numbered input through any of the channel sequencing modes is functionally identical to selecting the even-numbered input. The even-numbered input is always connected to ADCIN+ and the odd-numbered input is always connected to ADCIN−, and only the settings in the even-numbered input CONFIG_INn register are applied. It is recommended to only include the even-numbered input in the channel sequence when it is assigned as part of a channel pair.

When the standard sequencer is enabled, the pin-pairing assignment settings are the same for all 16 analog inputs and are set by the IN_PAIR field in the CONFIG_IN0 register. When the advanced sequencer, two-cycle command mode, or single-cycle command mode is enabled, the pin-pairing assignment settings are independent for all 16 analog inputs

and are set by the IN_PAIR field in each input's corresponding CONFIG_INn register. The polarity mode settings for each analog input are always set by the IN_MODE bits in their corresponding CONFIG_INn registers, regardless of the channel sequencing mode.

Note that pseudo-bipolar mode is not available for channels with the REFGND pin-pairing assignment selected. If a channel pin-pairing assignment is configured as REFGND, then the state of the IN_PAIR field is ignored.
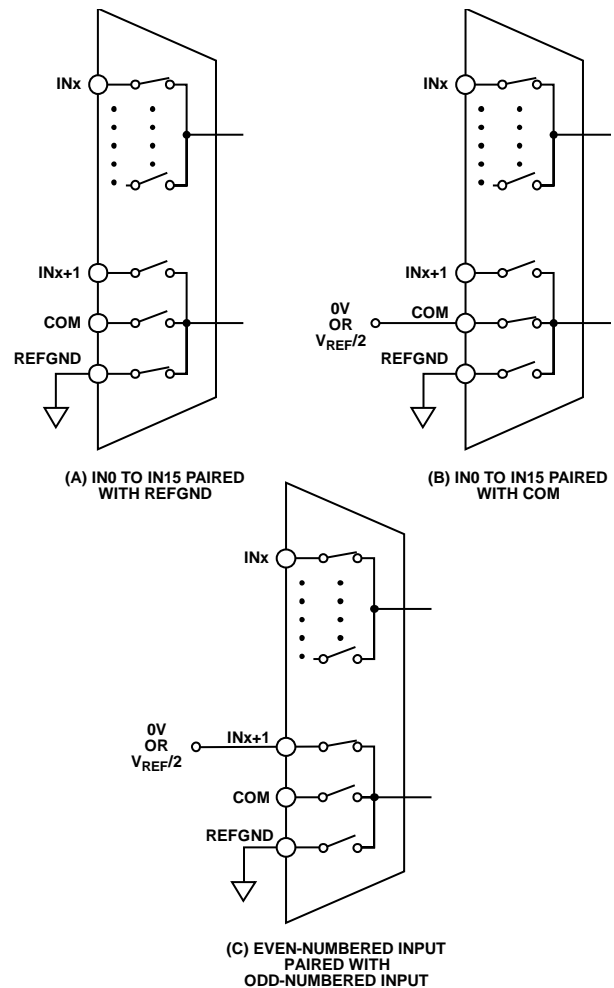


*Figure 69. Channel Configuration Options*

### Analog Input High-Z Mode

To achieve optimal data sheet performance from traditional high resolution SAR ADCs with a multiplexed front-end, system designers are often forced to include dedicated ADC driver amplifiers between the ADC and the remaining analog front end (AFE) to settle the voltage kickback that occurs at the analog inputs between conversions (see the Signal Settling Requirements section). The AD4695/AD4696 analog input high-Z mode simplifies the design requirements of the AFE circuitry driving the analog inputs, and facilitates the design of small footprint, high channel density precision multiplexed SAR ADC signal chains.

Analog input high-Z mode significantly reduces the magnitude of the voltage kickback which occurs at the analog inputs when the ADC and multiplexer switches reconnect at the start of the ADC acquisition phase. The reduction in kickback increases the effective input impedance of the AD4695/AD4696 inputs and reduces the bandwidth requirements of the AFE circuitry to achieved desired settling performance. Figure 14 shows the voltage kickback that occurs on an analog input driven to 5 V after switching from another analog input driven to 0 V, with analog input high-Z mode disabled and enabled.

The relaxed bandwidth requirements offered by analog input high-Z mode simplify AFE circuit design by broadening the selection of compatible amplifiers and external RC filter components. Analog input high-Z mode therefore helps remove the requirement of dedicated ADC driver amplifiers per channel, significantly reducing system footprint and power consumption.

The analog input high-Z mode also reduces performance degradation due to series resistance between the amplifiers and the AD4695/AD4696 analog inputs, allowing the resistor in the external RC filter (shown as $R_{EXT}$ in Figure 67 and Figure 107) to be larger compared to traditional multiplexed SAR ADCs. Using larger $R_{EXT}$ with smaller $C_{EXT}$ alleviates amplifier stability concerns without significantly impacting distortion performance. Figure 20 shows the THD performance for various source impedances with analog input high-Z mode enabled and disabled.

Analog input high-Z mode is enabled with the AINHIZ_EN bit in the CONFIG_INn registers. When the standard sequencer is enabled, analog input high-Z mode is enabled or disabled for all 16 analog inputs and are set by the AINHIZ_EN bit in the CONFIG_IN0 register. When the advanced sequencer is enabled or when using two-cycle or single-cycle command modes, analog input high-Z mode is enabled or disabled for all 16 analog inputs independently and are set by the AINHIZ_EN bit in each input's corresponding CONFIG_INn register. Analog input high-Z mode is always enabled when sampling the temperature sensor.

Performing conversions on a channel with analog input high-Z mode enabled consumes TBD% more power per conversion than if analog input high-Z mode were disabled; however, the total system power consumption is typically less than when using a dedicated high speed ADC driver. The power consumption also scales with sample rate and is relative to the number of times it is used in the channel sequence (for example, if the analog input high-Z mode is only used for 50% of the channels in the sequence, the total additional power consumption is also cut by 50%). Table 1 shows power consumption with analog input high-Z mode disabled and enabled on all channels in the sequence.

Figure 70 and Figure 71 show the AD4695/AD4696 SNR and THD performance while being driven by the ADA4077-1 (supply current per amplifier ($I_{SY}$) = 400 µA), and ADA4610-1 ($I_{SY}$ = 1.50 mA) precision amplifiers with analog input high-Z mode enabled and disabled with various external RC filter

values. These amplifiers achieve TBD dB to TBD dB typical SNR and close to − TBD dB typical THD with analog input high-Z mode enabled for a 2.27 MHz RC bandwidth. THD is approximately TBD dB better with analog input high-Z mode enabled, even for $R_{EXT}$ values greater than TBD Ω. SNR remains close to TBD dB even with a very low RC filter cutoff.



*Figure 70. SNR vs. RC Filter Bandwidths for Various Precision ADC Drivers, $V_{REF}$ = 5 V, $f_{IN}$ = 1 kHz (Analog Input High-Z Mode Enabled/Disabled)*



*Figure 71. THD vs. RC Bandwidths for Various Precision ADC Drivers, $V_{REF}$ = 5 V, $f_{IN}$ = 1 kHz (Analog Input High-Z Mode Enabled/Disabled)*

***Enhanced Acquisition Phase***

The AD4695/AD4696 feature a very fast conversion time of 365 ns (see $t_{CONVERT}$ in Table 2), which results in a longer acquisition phase for a given sample rate compared to previous generations of SAR ADCs. A longer acquisition phase relaxes the settling requirement on the driving circuit, and a lower power, lower bandwidth amplifier can be chosen. The longer acquisition phase also allows the external RC filter bandwidth to be lower, which means a noisier amplifier can also be tolerated. See the External RC Filter section and Signal Settling Requirements section for details on setting the RC filter bandwidth and choosing a suitable amplifier.

## INPUT OVERVOLTAGE PROTECTION CLAMPS

The AD4695/AD4696 include overvoltage protection clamps on IN0 thru IN15 and COM to reduce the risk of device damage from sustained dc overvoltage events. These clamps eliminate the need for external clamping diodes in systems where the input driving circuitry positive supply rail is greater than $V_{REF}$ (see Figure 107).

Table 1 shows the activation, deactivation, and clamping voltages of the overvoltage protection clamps. Figure 43 and Figure 44 show typical behavior of the clamps during overvoltage conditions. The clamp circuits activate when the analog input voltage exceeds the activation voltage. The clamps deactivate when the input voltage drops below the deactivation voltage. While a clamp is active, a flag is set in the status registers that can be read by the digital host. See the Overvoltage Clamp Flags section for a detailed description of the options for reading the status of each clamp.

The overvoltage protection clamps limit the extent to which input overvoltage events disturb the reference source. When active, the clamps limit the voltage on the analog inputs to the specified clamping voltage and conduct the input current to ground rather than through the ESD diode connecting the analog input to the REF input ($D_1$ in Figure 67), thereby preventing overvoltage conditions on one analog input from degrading performance on other analog inputs or other devices sharing the reference. Figure 41 shows the relationship between a single clamp input current and the resulting additional reference input current.

Figure 26, Figure 36, and Figure 38 show the offset error, gain error, and ac performance for one analog input channel vs. the total number of active overvoltage protection clamps on the other inputs.

Each overvoltage protection clamp circuit supports a maximum sustained current of 5 mA. All 17 clamp circuits can sink 5 mA simultaneously without damaging the device. The clamp current is a function of $V_{REF}$, the external series resistance (such as $R_{EXT}$ in Figure 67), and the output voltage of the AFE circuitry. See the Analog Input Overvoltage Protection section for details on how to select $R_{EXT}$ to prevent excess clamp current during overvoltage events.

### *Overvoltage Reduced Current Mode*

The overvoltage reduced current mode further reduces additional reference current during overvoltage events. Figure 41 shows the difference between the additional reference input currents drawn for different clamp input currents with the overvoltage reduced current mode enabled and disabled.

The overvoltage reduced current mode is enabled when the OV_MODE bit in the REF_CTRL register is set to 0. Overvoltage reduced current mode is enabled by default.

Enabling overvoltage reduced current mode changes the maximum value of $R_{EXT}$ and achieves stable clamp operation. See the Overvoltage Protection Clamp Stability section for

more information on the relationship between the external RC filter and clamp operation.

### *Overvoltage Protection Clamp Stability*

In applications where analog input overvoltage events are not a concern, or in applications where clamp stability is not a concern, the $R_{EXT}$ and $C_{EXT}$ values are not required to follow the guidelines described in this section.

The stability of the overvoltage protection clamp circuits depends on the external RC filter component values and whether the overvoltage reduced current mode is enabled or disabled. When a clamp is unstable, it toggles between the active and inactive states during overvoltage events. This instability causes small, modulating currents to flow in both the overdriven input and the reference, which can result in measurement errors in the conversions of other analog inputs if the reference circuitry does not have adequate load regulation to maintain a stable reference voltage in response to the additional reference current. Table 1 and Figure 41 show the additional reference input (REF) current per active clamp.

To ensure stable clamp operation, $C_{EXT}$ in the external RC filter (as shown in Figure 67) must be at least 500 pF. The maximum value of $R_{EXT}$ is 1 kΩ when the overvoltage reduced current mode is enabled, and 2 kΩ when the overvoltage reduced current mode is disabled.

### *Overvoltage Clamp Flags*

The AD4695/AD4696 provide several means to check the status of the overvoltage protection clamps.

The INX_CLAMP_FLAG bits in the CLAMP_STATUS1 and CLAMP_STATUS2 registers indicate the status of the overvoltage protection clamps for IN0 thru IN15. Each INX_CLAMP_FLAG bit is asserted when the corresponding input clamp circuit is active and is de-asserted when it is inactive. The CLAMP_FLAG bit in the STATUS is asserted when any combination of the overvoltage clamps on IN0 thru IN15 are activated (when any of the INX_CLAMP_FLAG bits are asserted). This bit is sticky and is only cleared when it is read while all of the clamps are inactive.

The COM_CLAMP_FLAG bit in the STATUS register is asserted when the COM input overvoltage protection clamp is active and is de-asserted when it is inactive. These bits can be read when in register configuration mode to check the current status of each of the overvoltage input clamp circuits.

The OV_ALT flag in the optional status bits allows checking the status of all of the overvoltage clamps while performing conversions. The OV_ALT flag is the bitwise logical OR of the 16 INX_CLAMP_FLAG bits in the CLAMP_STATUS1 and CLAMP_STATUS2 registers. The OV_ALT flag can also be configured as the logical OR of the overvoltage clamp flags and the general threshold alert indicator (as described in the Threshold Detection and Alert Indicators section). See the Status Bits section for details on configuring the OV_ALT flag.

## TEMPERATURE SENSOR

The AD4695/AD4696 include a temperature sensor which converts the die temperature to an output voltage that can be sampled and converted to an output code by the SAR ADC core. The relationship between the measured die temperature (T) and the temperature sensor output voltage ($V_{TEMP}$) is nominally:

$$V_{TEMP} = \left( -TBD\,\frac{mV}{°C} \times T \right) + TBD\ mV$$

The temperature sensor sensitivity is a measure of the change in output voltage in relation to a change in device temperature, and is typically −TBD mV/°C. At 0°C, the temperature sensor output is typically TBD mV.

When the temperature sensor is selected, the multiplexer $SW_{MUX+}$ switch (see Figure 68) selects the temperature sensor's output and its $SW_{MUX-}$ switch selects REFGND, and the SAR ADC core samples $V_{TEMP}$ to generate a corresponding output code. The analog-to-digital conversion of the temperature sensor's output utilizes the same transfer function as an analog input configured in unipolar mode with OSR = 1 (see the Transfer Function section).

When the standard sequencer or advanced sequencer is enabled, the temperature sensor is sampled at the end of the pre-programmed channel sequence if the TEMP_EN bit in the TEMP_CTRL register is set to 1.

When using either two-cycle command mode or single-cycle command mode, the temperature sensor can be selected by writing the code 0x0F on SDI on the first five rising edges of SCK in the same way analog inputs are selected (see Table 14).

When the temperature sensor is enabled, analog input high-Z mode is always enabled and the oversampling ratio is always 1. The temperature sensor does not have threshold detection alerts.

## VOLTAGE REFERENCE INPUT

The reference voltage ($V_{REF}$) sets the ADC full-scale voltage (see the Transfer Function section). The ADC core samples the voltage on the reference input (REF) during the bit trials in the conversion process to determine the output code result. The AD4695/AD4696 are compatible with reference voltages from 2.4 V to 5.1 V.

The AD4695/AD4696 must be configured for optimal performance with the selected reference voltage. The VREF_SET field in the REF_CTRL register provides five $V_{REF}$ range options, shown in Table 46. This value must be programmed to match the $V_{REF}$ voltage applied to the REF pin.

A common challenge presented by traditional SAR ADCs is in designing reference circuitry with sufficient drive capability to maintain a precise $V_{REF}$ while the REF input dynamically draws input current during the SAR bit trials. Deviations in $V_{REF}$ result in reduction in ADC accuracy and performance, such as higher gain error, distortion, etc. The REF input presents a dynamic load as it pulls charge from the external reference circuitry at different times in the SAR process. This process traditionally requires either voltage references with sufficient load regulation and drive capabilities, or the use of a dedicated reference buffer to drive the REF input, along with a large reference decoupling capacitor. See the Reference Circuitry Design section for more information on properly selecting reference circuitry components.

The AD4695/AD4696 incorporate features which simplify design of the companion reference circuitry, and facilitate the design of small footprint, low power systems. The reference input high-Z mode reduces the REF input current by TBD%, allowing a broader selection of voltage references and amplifiers to drive the REF input without impacting performance (see the Reference Input High-Z Mode section).

The reference input current scales with sample rate (see Table 1 and Figure 39).

### Reference Input High-Z Mode

When enabled, reference input high-Z mode reduces the average REF current by TBD% to TBD µA/MSPS. The reduction in REF current allows the AD4695/AD4696 to tolerate larger series resistance between the reference source and the REF input without compromising performance. Reference input high-Z mode therefore allows for voltage references with higher load regulation specifications to directly drive the REF input without the need for a dedicated reference buffer.

The REF input requires a decoupling capacitor ($C_{REF}$). When reference input high-Z mode is disabled, $C_{REF}$ must be 10 µF or larger. When reference input high-Z mode is enabled, $C_{REF}$ can be as small as 1 µF.

See the Reference Circuitry Design section for more reference circuit design recommendations.

Reference input high-Z mode is enabled and disabled by setting the value of the REFHIZ_EN bit in the REF_CTRL register and is enabled by default.

When reference input high-Z mode is enabled, the AD4695/AD4696 consume approximately TBD% more power; however, this is still significantly lower than using higher power references or discrete reference buffers that are traditionally required to drive the ADC's reference input. The power consumption also scales with sample rate. Table 1 shows power consumption with reference input high-Z mode disabled and enabled.

## POWER SUPPLIES

The AD4695/AD4696 have three power supply pins: an analog supply (AVDD), an ADC core supply (VDD), and a digital input/output interface supply (VIO). The AD4695/AD4696 also include an internal LDO that can be used to provide the VDD rail with a wider variety of supply voltages (or in single-supply systems by tying LDO_IN to AVDD). Table 1 shows the specified power supply voltage requirements.

AVDD can range from 3.15 V to 5.5 V and powers the analog front-end features of the AD4695/AD4696, including the analog input high-Z mode and reference input high-Z mode circuitry.

VDD is nominally 1.8 V, and powers both the ADC core and the device register memory. When power is first applied to VDD, the ADC core initializes and device register contents are set to their default state (as shown in the Register Information section).

VIO can range from 1.2 V to 1.8 V and sets the input and output levels for the digital interface pins. VIO allows direct interfacing with digital controller logic levels between 1.2 V and 1.8 V(see the Digital Interface section for more information).

Decouple AVDD to AGND and VIO to IOGND with at least 100 nF and decouple VDD to AGND with at least 1 µF. When not using the internal LDO to supply VDD, LDO_IN does not require decoupling.

The AD4695/AD4696 is independent of the power supply sequencing between VIO, VDD and AVDD (and LDO_IN when the internal LDO is enabled). When VIO and VDD are first supplied, a power-on reset is initiated (see the Device Reset section). Additionally, the AD4695/AD4696 is insensitive to power supply ripple over a wide frequency range, as shown in Figure 31.

### Internal LDO

To minimize the number of system supply rails required to power the AD4695/AD4696, the internal LDO can be used to supply the VDD voltage internally. LDO_IN can be tied to AVDD to enable a single supply to power the entire device (excluding VIO, which should be powered by the digital host I/O voltage).

To enable the internal LDO, LDO_IN must be driven to at least 2.4 V and VIO must already be powered. The internal LDO is enabled by setting the LDO_EN bit in the SETUP register to 1. The internal LDO is enabled by default on device power-up and after device resets. When the internal LDO is enabled, its output drives VDD internally. When the internal LDO is disabled, its output is high impedance.

It is not possible to power the VIO supply with the internal LDO output. VIO must be supplied by the digital host or other system supply rail.

When using the internal LDO, VDD must be floating, and the VDD supply voltage is driven by the internal LDO's output automatically when LDO_IN and VIO are supplied. When not using the internal LDO, LDO_IN should be tied to AGND, and VDD must be supplied externally.

The internal LDO output is designed to withstand being powered up with VDD either driven by a separate 1.8 V supply or inadvertently shorted to AGND. It is still recommended to ensure VDD is disconnected from any other rails or loads. The internal LDO is not intended to power additional devices. It is recommended to clear the LDO_EN bit when powering VDD externally, even if the LDO_IN input is shorted to AGND (see the Device Configuration Recommendations section).

The internal LDO can be disabled to put the AD4695/AD4696 in a low-power state without disabling the AVDD, LDO_IN or VIO rails. When the internal LDO is disabled while VDD is not powered by an external supply, the ADC core shuts down and configuration register contents are erased. The internal LDO can be enabled again either with a wake-up command over the SPI, or with a hardware reset. The wake-up command is 0x81 and is identical to performing a software reset (see the Device Reset section for detailed descriptions of hardware and software resets). The digital interface requires that VIO still be supplied to accept the wake-up command, and the internal LDO will not be enabled if VIO is not within the specified range (see Table 1).

## OVERSAMPLING AND DECIMATION

The AD4695/AD4696 include an oversampling and decimation engine that averages consecutive ADC samples to generate an oversampled result with higher effective resolution and lower effective noise (see Table 1).

Each analog input channel can be configured with oversampling ratios of 1, 4, 16, or 64. Conversion results generated for channels with oversampling ratios of 4, 16, or 64 are 17-, 18-, and 19-bits long, as shown in Table 17, Table 18, Table 19, and the Transfer Function section.

When a given analog input channel is selected by the channel sequencing logic, the multiplexer continues to select that channel until the specified number of conversions have been performed, and the results of each of those conversions are averaged together to generate a single output code. For example, if IN0 is configured with an oversampling ratio of 64, then one averaged result is produced after the 64th consecutive CNV rising edge (when the AD4695/AD4696 is in conversion mode). Configuring a channel with an oversampling ratio of 1 is equivalent to performing no oversampling on that channel.

When enabled on a general-purpose pin or the serial data output(s), the busy indicator acts as a "data ready" signal, and only transitions low when the oversampled result is available (see the Busy Indicator section). Figure 75 shows the relative timing of the busy indicator when the oversampling ratio for a channel is set to a value other than 1.

The effective sample period of a given channel is equal to the conversion period ($t_{CYC}$ in Table 2) multiplied by its oversampling ratio. Figure 75 shows the relative timing of the CNV signal and the availability of the oversampled result. Consider the oversampling ratios of each channel when designing the channel sequence to achieve the desired certain effective sample rates for each channel (see the Effective Channel Sample Rate section).

The oversampling ratio is configured via the OSR fields in the CONFIG_INn registers (see Table 54).

When the standard sequencer is enabled, the oversampling ratio for all analog input channels is the same and is set by the OSR field in the CONFIG_IN0 register. When the advanced sequencer is enabled, each of the 16 analog input channels can be configured with different oversampling ratios with the OSR fields in their corresponding CONFIG_INn registers.

Oversampling is not supported in two-cycle command mode or single-cycle command mode. Set the OSR fields for all active channels to 0x0 when using two-cycle command mode or single-cycle command mode.

When autocycle mode is enabled, the conversion signal is generated internally by the AD4695/AD4696, and the oversampling engine still waits for OSR conversion periods before generating an output result.

## OFFSET AND GAIN CORRECTION

The AD4695/AD4696 includes offset and gain error correction functionality to correct for first-order non-idealities in full analog front-end signal chain. Offset and gain error correction digitally adjusts the offset and gain of the overall ADC transfer function (see the Transfer Function section).

The final output code is calculated with the following expression:

$$OUT = (IN + B) \times M$$

where:
$OUT$ is the final output code result.
$IN$ is the result generated by the ADC (after oversampling).
$B$ is the offset correction value.
$M$ is the gain correction value.

The gain correction value ($M$) for each analog input is set with the GAIN field in its corresponding GAIN_INn register. The GAIN field is 16 bits wide and is in straight binary format. The range of gain correction values is 0 to +1.99997, and is calculated with the following expression:

$$M = GAIN/2^{15}$$

where:
$M$ is the gain correction value.
$GAIN$ is the value written to the GAIN field.

The offset correction value ($B$) for each analog input is set with the OFFSET field in its corresponding OFFSET_INn register. The OFFSET field is 16 bits wide and is in twos complement format to enable positive and negative offset correction. The range of offset correction values is ±FSR/8 for all oversampling ratio options, so the MSB of the OFFSET field always corresponds to the MSB – 3 bit of the ADC result. For example, when the oversampling ratio for a given analog input channel is 1, then the offset correction value is equal to OFFSET[15:3], and when the oversampling ratio is 64, the offset correction value is OFFSET[15:0]. Table 10 shows the offset correction value for each oversampling ratio option.

Offset and gain correction are always enabled for all analog input channels. When the OFFSET field for a given analog input is set to 0x0000, the offset correction value is 0, and it is equivalent to applying no offset correction. When the GAIN field for a given analog input is set to 0x8000, the gain correction value is 1, and it is equivalent to applying no gain correction.

Table 10. Offset Correction Value vs. Oversampling Ratio

| Oversample Ratio | Offset Correction Value (B) |
|---|---|
| 1 | OFFSET[15:3] |
| 4 | OFFSET[15:2] |
| 16 | OFFSET[15:1] |
| 64 | OFFSET[15:0] |

## THRESHOLD DETECTION AND ALERT INDICATORS

The AD4695/AD4696 include a threshold detection feature with alert indicators that notify the digital host system when a conversion result violates user-defined upper and lower limits.

The TD_EN bit in the CONFIG_INn registers enables or disables threshold detection for the corresponding analog input. When the standard sequencer is enabled, threshold detection is enabled or disabled for all analog inputs with the TD_EN bit in the CONFIG_IN0 register. When the advanced sequencer, two-cycle command mode or single-cycle command mode are enabled, threshold detection is enabled or disabled for each analog input independently with the TD_EN bit in each of their corresponding CONFIG_INn registers.

When threshold detection is enabled for a given analog input, ADC results generated for that analog input are compared against a upper threshold value and lower threshold value. Upper and lower threshold values can be independently assigned for each of the 16 analog inputs. The upper and lower threshold values for the 16 analog inputs are set with the UPPER and LOWER fields in the UPPER_INn and LOWER_INn registers. The UPPER and LOWER fields are 12 bits wide and correspond to the 12 most significant bits of the ADC results, for all oversampling ratio options. For example, setting UPPER to 0xFFF corresponds to an upper threshold value of 0xFFF0 when the oversampling ratio of that channel is 1, and 0x7FF80 when the oversampling ratio of that channel is 64 (see the Oversampling and Decimation section).

When an analog input is configured in unipolar mode, its corresponding UPPER and LOWER fields are in straight binary format. When an analog input is configured in pseudo-bipolar mode, its corresponding UPPER and LOWER fields are in twos complement format.

### Alert Indicator Registers

The ALERT_STATUS1 thru ALERT_STATUS4 registers contain the upper alert indicators (HI_INn) and lower alert indicators (LO_INn) for all 16 analog inputs. The TD_ALERT bit in the STATUS register is the logical OR of the HI_INn and LO_INn bits. When the ADC result is greater than or equal to the upper threshold value, the corresponding HI_INn flag is set to 1. When the ADC result is less than or equal to the lower threshold value, the corresponding LO_INn flag is set to 1. When the oversampling ratio of an INn analog input is greater than 1, the state of its corresponding HI_INn and LO_INn flags update after the oversampled result is generated.

Reading the TD_ALERT bit indicates to the digital host whether any upper or lower threshold was violated and reading the HI_INn and LO_INn bits indicate which specific type of threshold was violated on which channel. The AD4695/AD4696 must be in register configuration mode to read from the registers that contain these alert indicator bits, but the state of TD_ALERT can also be read via the status bits or a general-purpose pin when these options are enabled (see the Status Bits and Alert Indicator on  sections).

The HI_INn and LO_INn bits are read-to-clear bits and are automatically reset to 0 after being read in a SPI read transaction (in register configuration mode).

When the ALERT_MODE bit in the SETUP register is set to 0, the HI_INn and LO_INn bits are also automatically cleared based on user programmable hysteresis settings. The HYSTERESIS fields in the 16 HYSTERESIS_INn registers sets the hysteresis value for the corresponding analog input. Each analog input can be programmed with different hysteresis

values. When this option is selected, each HI_INn bit is automatically cleared when the corresponding analog input generates a conversion result that is less than the upper threshold value minus its hysteresis value. Each LO_INn bit is automatically cleared when the corresponding analog input generates a conversion result that is greater than the lower threshold value plus its hysteresis value. Figure 72 shows how the HI_INn and LO_INn bits are set and cleared when ALERT_MODE is set to 0 and 1 as conversion results are generated on the corresponding analog input channel. ALERT_MODE is set to 0 by default.

### Alert Indicator on BSY_ALT_GP0

When the alert indicator is enabled on BSY_ALT_GP0, the state of the TD_ALERT bit is driven on the BSY_ALT_GP0 pin, allowing threshold violations to be detected without interrupting conversions. The combination of the alert indicator on a general-purpose pin and autocycle mode allows the digital host serial interface to remain idle until a threshold violation is detected (see the Autocycle Mode section).

Figure 91 through Figure 98 show the relative timing of CNV rising edges and when the state of a general-purpose pin configured as the alert indicator is updated.

Set the ALERT_GP_EN bit in the GP_MODE register to 1 to enable the alert indicator on the BSY_ALT_GP0 pin.

The BSY_ALT_GP0 pin can also be configured to perform other functions besides the alert indicator, and all other higher-priority functions must be disabled to configure them as the busy indicator. See the General-Purpose Pin section for details on configuring the general-purpose pin functionality.
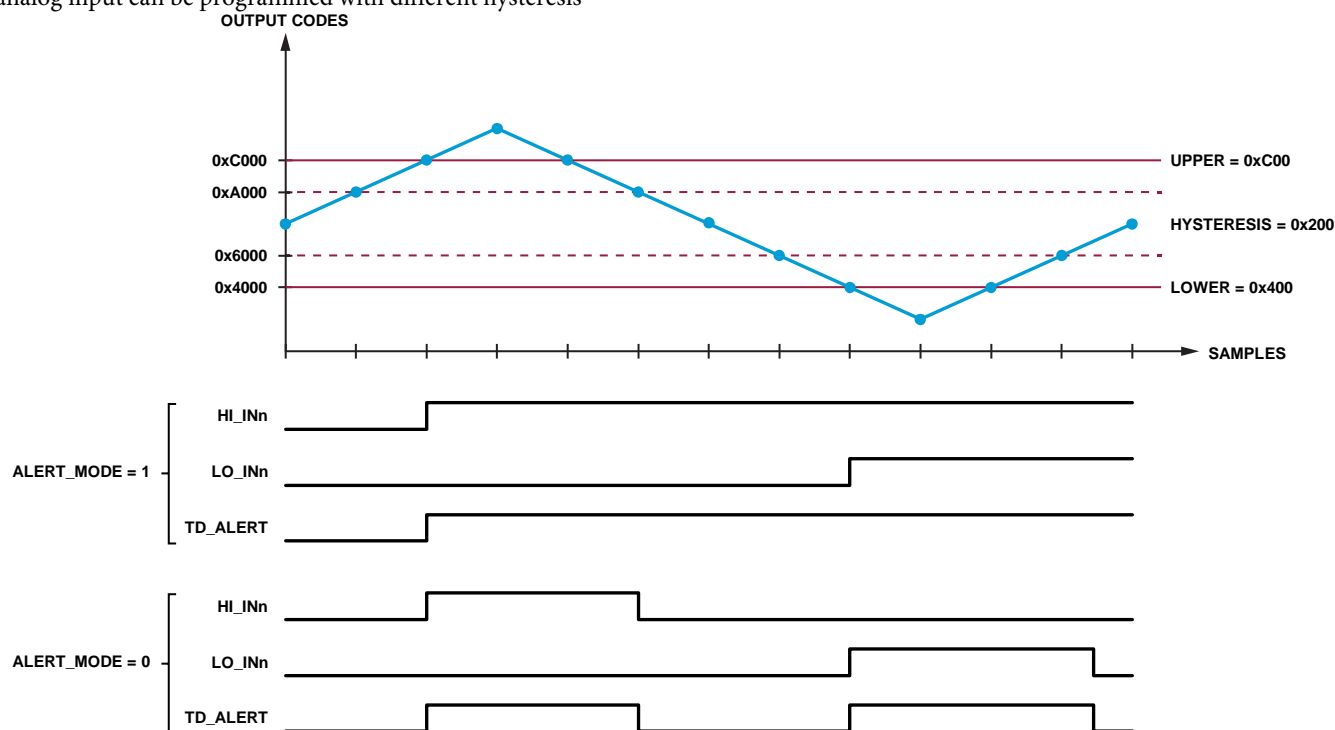


Figure 72. Alert Indicator Behavior with Hysteresis Enabled and Disabled (Unipolar Mode, Oversampling Ratio = 1)

## BUSY INDICATOR

The busy indicator acts as a data-ready signal that can be used to trigger an interrupt service routine on the digital host to initiate an SPI transaction to read the ADC result (see the Conversion Mode section and SPI Peripheral Synchronization in Conversion Mode section). The busy indicator can be enabled on the serial data outputs and on the BSY_ALT_GP0 pin.

### Busy Indicator on Serial Data Outputs

When the busy indicator is enabled on the serial data outputs, the serial data outputs are high-impedance while the ADC is in the conversion phase, and transition low when the ADC result is ready. Set the SDO_STATE bit in the SETUP register to 1 to enable the busy indicator on the serial data outputs.

Figure 91 through Figure 98 show the relative timing of CNV rising edges to the busy indicator on the serial data output(s).

The serial data output mode selected by the SDO_MODE field determines which pins are assigned as serial data outputs (see the Serial Data Output Modes section). When SDO_STATE is set to 1, the busy indicator is enabled on all pins assigned as serial data outputs. When single-SDO mode is selected, the busy indicator is only output on SDO. When dual-SDO mode is selected, the busy indicator is output on both SDO and BSY_ALT_GP0.

When enabling the busy indicator on the serial data outputs, place pull-up resistors (2 kΩ minimum) on each utilized pin to ensure the serial data output lines are pulled high until the ADC result is ready.

The serial data outputs are forced to a high-impedance state whenever the $\overline{CS}$ pin is driven high. If the $\overline{CS}$ pin is high when the ADC result is ready, the serial data outputs will remain high impedance until the $\overline{CS}$ pin is brought low (see the Digital Interface section).

### Busy Indicator on BSY_ALT_GP0

When the busy indicator is enabled on the BSY_ALT_GP0 pin, BSY_ALT_GP0 is driven high while the ADC is in the conversion phase, and transition low when the ADC result is ready. Set the BUSY_GP_EN bit in the GP_MODE register to 1 to enable the busy indicator on BSY_ALT_GP0.

Figure 91 through Figure 98 show the relative timing of CNV rising edges to the busy indicator rising and falling edges.

When the BSY_ALT_GP0 pin is assigned as the busy indicator, it is not forced to high impedance when the $\overline{CS}$ pin is high. This allows the digital host to leave its serial interface completely disabled until a busy indicator falling edge is registered (see the SPI Peripheral Synchronization in Conversion Mode section).

The BSY_ALT_GP0 pin can also be configured to perform other functions besides the busy indicator, and all other higher-priority functions must be disabled to configure them as the busy indicator. See the General-Purpose Pin section for details on configuring the general-purpose pins.

## CHANNEL SEQUENCING MODES

In conversion mode, the AD4695/AD4696 multiplexer channel updates once per conversion period at the start of the ADC core's acquisition phase, as described in the Converter Operation section. The multiplexer is controlled by internal channel sequencing logic, and there are four options for programming the channel sequence.

The standard sequencer and advanced sequencer automates progression through a pre-programmed channel sequence. When either the standard sequencer or advanced sequencer are enabled, the digital host is not required to provide channel sequencing instructions while reading conversion results over the SPI, which reduces the digital resource requirements.

Two-cycle command mode and single-cycle command mode allow the digital host to directly control the channel sequence via 5-bit commands written over the serial interface during conversion data readback frames. Two-cycle command mode and single-cycle command mode enable systems with dynamic and adaptive channel sequencing requirements, such as control loop applications.

Table 11 shows the configuration settings used to select from the four channel sequencing modes. Both the STD_SEQ_EN bit and the NUM_SLOTS_AS field are located in the SEQ_CTRL register. The CYC_CTRL bit is located in the SETUP register.

As noted in the Channel Configuration Options section, when even- and odd-numbered inputs are paired, selecting the odd-numbered input using any of the four channel sequencing modes will result in the same behavior as if the even-numbered input were selected instead. For this reason, it is recommended to only include the even-numbered input in the channel sequence.

### Standard Sequencer

The standard sequencer automates progression through a pre-programmed set of enabled channels. The standard sequencer is the simplest of the four channel sequencing modes, and is ideal for systems with fixed, static channel sequences.

The standard sequencer advances through each of the enabled channels in ascending order and repeats the sequence until the device is put out of conversion mode. The multiplexer channel is updated to the next enabled channel each time a conversion result is ready. Figure 73 shows an example where the standard sequencer is enabled and three analog inputs (IN0, IN2, IN15,) and the temperature sensor are enabled in the sequence (with no oversampling on any channel).

The bits in the STD_SEQ_CONFIG register control which channels are included in the channel sequence when the standard sequencer is enabled. Each bit in the STD_SEQ_CONFIG register corresponds to one of the 16 analog inputs, and each channel is enabled if its corresponding bit is set to 1. If the TEMP_EN bit in the TEMP_CTRL register is set to 1, then the temperature sensor is added to the end of sequence as well. For the example in Figure 73, the value programmed into the

STD_SEQ_CONFIG register is 0x1005, and the TEMP_EN bit is set to 1.

To enable the standard sequencer, set the STD_SEQ_EN bit in the SEQ_CTRL register to 1 and the CYC_CTRL bit in the SETUP register to 0 (see Table 11). The standard sequencer is enabled by default.

While the AD4695/AD4696 are in register configuration mode, when the STD_SEQ_EN bit in the SEQ_CTRL register is set to 1, the multiplexer automatically connects the first enabled channel in the sequence to the ADC core inputs, which allows the ADC to acquire the signal on that channel even before the device enters conversion mode.

When the standard sequencer is enabled, the control bits in the CONFIG_IN0 register determine the configuration settings for all INn analog inputs (except for the polarity mode, which is set for each INn analog input independently with the IN_MODE bit in their corresponding CONFIG_INn register). Therefore, all analog inputs have the same pin-pairing options, analog input easy drive enable settings, oversampling ratio settings, and threshold detection enable settings.

The multiplexer does not advance to the next channel in the sequence until the required number of conversions dictated by the selected channel's oversampling ratio setting has been completed. For example, if the oversampling ratio is set to 16, then 16 CNV rising edges are required before the conversion result is ready and the multiplexer selects the next channel in the sequence. Figure 74 shows an example timing diagram where the oversampling ratio for all channels is set to N. See the Oversampling and Decimation section for more information.

When the standard sequencer is enabled, each enabled analog input is sampled once per each sequence iteration, so each analog input has the same effective sample rate. See the Effective Channel Sample Rate section for more information.

**Table 11. Register Settings for Channel Sequencing Modes**

| Channel Sequencing Mode | STD_SEQ_EN | NUM_SLOTS_AS | CYC_CTRL |
|---|---|---|---|
| Two-cycle command mode | 0 | 0 | 0 |
| Single-cycle command mode | 0 | 0 | 1 |
| Standard sequencer | 1 | Don't Care | 0 |
| Advanced sequencer | 0 | 1 to 127 | 0 |

### Advanced Sequencer

The advanced sequencer automates progression through a pre-programmed channel sequence where the order of channels is completely customizable. The advanced sequencer enables highly flexible sequences of channels with minimal digital overhead.

The advanced sequencer steps through a set of channel "slots," where each slot can be assigned to any of the 16 analog inputs, and sequences can be between two and 128 slots. The sequence progresses through the enabled slots in ascending order starting from slot 0, and the sequence is repeated until the device is put

out of conversion mode. Figure 74 shows an example where the advanced sequencer is enabled with four slots enabled and assigned to IN6, IN10, IN6 and IN3 and the temperature sensor is enabled (with no oversampling on any channel).

The number of slots in the sequence is set with the NUM_SLOTS_AS field in the SEQ_CTRL register. Each slot's channel assignment is set with the SLOT_INX fields in the AS_SLOTn registers (located at register addresses 0x100 to 0x17F), where AS_SLOT0 corresponds to slot 0, AS_SLOT1 corresponds to slot 1, etc. Table 60 shows the values of SLOT_INX for each of the 16 analog inputs.

If the TEMP_EN bit in the TEMP_CTRL register is set to 1, then the temperature sensor is appended to the end of sequence. The temperature sensor cannot be selected with the SLOT_INX fields in the AS_SLOTn registers.

To enable the advanced sequencer, set the STD_SEQ_EN bit to 0, the CYC_CTRL bit to 0, and the NUM_SLOTS_AS field to any value between 1 and 127 (see Table 11).

While the AD4695/AD4696 are in register configuration mode, when the STD_SEQ_EN bit in the SEQ_CTRL register is set to 0, the multiplexer automatically connects the channel specified in the AS_SLOT0 register to the ADC core inputs, which allows the ADC to acquire the signal on that channel even before the device enters conversion mode.

When the advanced sequencer is enabled, the configuration settings for each channel are set with their corresponding CONFIG_INn register. Therefore, all analog inputs can have different channel configuration options, analog input easy drive enable settings, oversampling ratio settings, and threshold detection enable settings. Configure each of the CONFIG_INn registers before entering conversion mode and initiating conversions.

The multiplexer does not advance to the next channel in the sequence until the required number of conversions dictated by the selected channel's oversampling ratio setting has been completed. When the oversampling ratio of a channel in the sequence is set to a value other than 1 (when the OSR field in the corresponding CONFIG_INn register is not set to 0x0), then the advanced sequencer does not advance to the next channel in the sequence and the busy indicator does not transition low until the required number of conversions has been completed. For example, if the oversampling ratio is set to 16, then 16 CNV rising edges are required before the conversion result is ready and the multiplexer selects the next channel in the sequence. Figure 75 shows an example timing diagram where the oversampling ratio for IN0 is set to N. See the Oversampling and Decimation section for more information.

When the advanced sequencer is enabled, the channel sequence can be configured to achieve different effective sample rates for each channel. See the Effective Channel Sample Rate section for more information.

*Two-Cycle Command Mode*

Two-cycle command mode allows the digital host system to manually control the next channel in the sequence on-the-fly and enables dynamic channel sequencing without interrupting conversions.

In two-cycle command mode, the channel sequence is determined by 5-bit commands transmitted from the digital host during conversion result readback frames. The 5-bit commands are clocked in on SDI on the first five SCK rising edges in the frame and latched into memory on the 6th SCK falling edge in the frame. If a valid channel command is received, the conversion result for that channel will be available after two conversion periods. Figure 76 shows the relative timing between the 5-bit commands (represented by "CMD") and the corresponding acquisition phase, conversion phase, and conversion result readback in two-cycle command mode.

Two-cycle command mode maximizes the acquisition time for all channels because the 5-bit channel commands are latched in before the multiplexer switches select the corresponding channel and begin the ADC acquisition phase.

Table 14 shows the valid commands for selecting IN0 thru IN15 or the temperature sensor. Commands other than those listed in Table 14 are treated as no-operation (NOOP) commands, and result in the multiplexer repeating the previous channel.

When two-cycle command mode is enabled, the first analog input channel selected is the one specified in the AS_SLOT0 register. The channel will only update after a valid CMD code is received.

To enable two-cycle command mode, set the STD_SEQ_EN bit to 0 and the NUM_SLOTS_AS field to 0x00 and the CYC_CTRL bit to 0 (see Table 11).

While the AD4695/AD4696 are in register configuration mode, when the STD_SEQ_EN bit in the SEQ_CTRL register is set to 0, the multiplexer automatically connects the channel specified in the AS_SLOT0 register to the ADC core inputs, which allows the ADC to acquire the signal on that channel even before the device enters conversion mode.

When two-cycle command mode is enabled, the configuration settings for each channel are set with their corresponding CONFIG_INn register. Therefore, all analog inputs can have different channel configuration options, analog input easy drive enable settings, and threshold detection enable settings. Configure each of the CONFIG_INn registers before entering conversion mode and initiating conversions.

Oversampling is not supported when two-cycle command mode is enabled. Set the oversampling ratio for all analog inputs to 1 before entering conversion mode with two-cycle command mode enabled (see the Oversampling and Decimation section).

*Single-Cycle Command Mode*

Single-cycle command mode allows the digital host system to manually control the next channel in the sequence on-the-fly

and enables dynamic channel sequencing without interrupting conversions.

In single-cycle command mode, the channel sequence is determined by 5-bit commands transmitted from the digital host during conversion result readback frames. The 5-bit commands are clocked in on SDI on the first five SCK rising edges in the frame and latched into memory on the 6th SCK falling edge in the frame.

If a valid channel command is received, the conversion result for that channel will be available in only one conversion period. Figure 77 shows the relative timing between the 5-bit commands (represented by "CMD") and the corresponding acquisition phase, conversion phase, and conversion result readback in single-cycle command mode.

Single-cycle command mode minimizes the latency between the 5-bit channel commands and the corresponding ADC data because the multiplexer switches select the specified channel immediately after the 5-bit command is latched into memory. As a result, the acquisition time is reduced and the analog front-end circuitry has less time to settle the analog input signals compared to the other three channel sequencing modes. Figure 95 shows a conversion mode timing diagram with single-cycle command mode enabled, and Table 2 lists the relevant timing specifications. The acquisition time ($t_{ACQ}$) in single-cycle command mode is a function of the CNV period ($t_{CYC}$) and SCK period ($t_{SCK}$), and can be calculated with the following expression:

$$t_{ACQ} = t_{CYC} - t_{CONVERT} - (5.5 \times t_{SCK})$$

Table 14 shows the valid commands for selecting IN0 thru IN15 or the temperature sensor. Commands other than those listed in Table 14 are treated as no-operation (NOOP) commands, and result in the multiplexer repeating the previous channel.

When single-cycle command mode is enabled, the first analog input channel selected is the one specified in the AS_SLOT0 register. The channel will only update after a valid command is received.

To enable single-cycle command mode, set the STD_SEQ_EN bit to 0 and the NUM_SLOTS_AS field to 0x00 and the CYC_CTRL bit to 1 (see Table 11).

When single-cycle command mode is enabled, the configuration settings for each channel are set with their corresponding CONFIG_INn register. Therefore, all analog inputs can have different channel configuration options, analog input easy drive enable settings, and threshold detection enable settings. Configure each of the CONFIG_INn registers before entering conversion mode and initiating conversions.

Oversampling is not supported when single-cycle command mode is enabled. Set the oversampling ratio for all analog inputs to 1 before entering conversion mode with single-cycle command mode enabled (see the Oversampling and Decimation section).
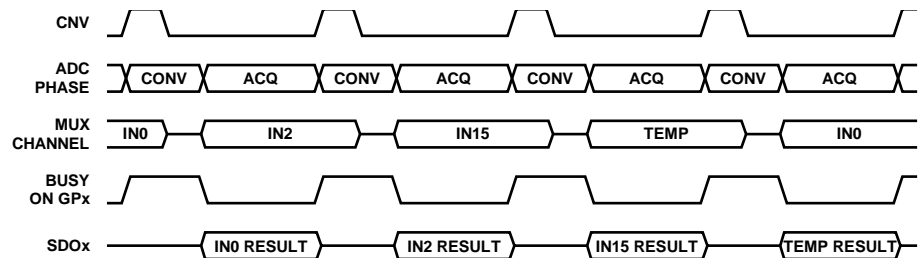
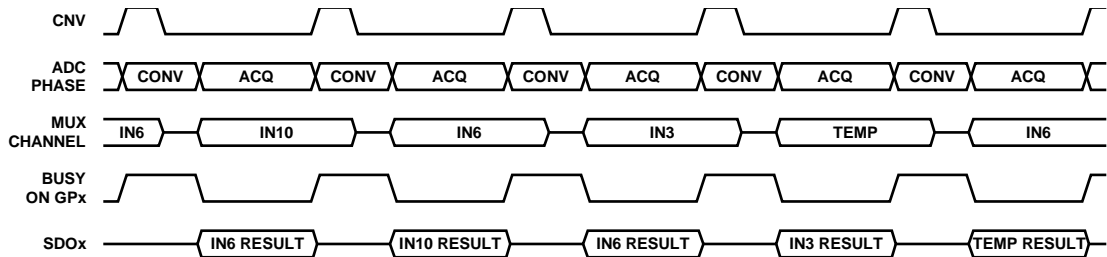Figure 73. Standard Sequencer Example with Oversampling Ratio = 1



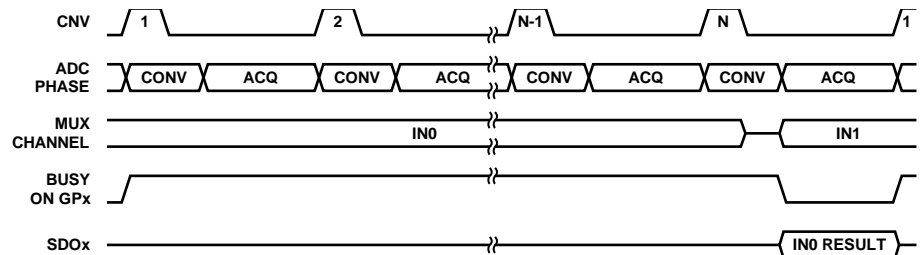Figure 74. Advanced Sequencer Example with Oversampling Ratio = 1 for All Channels



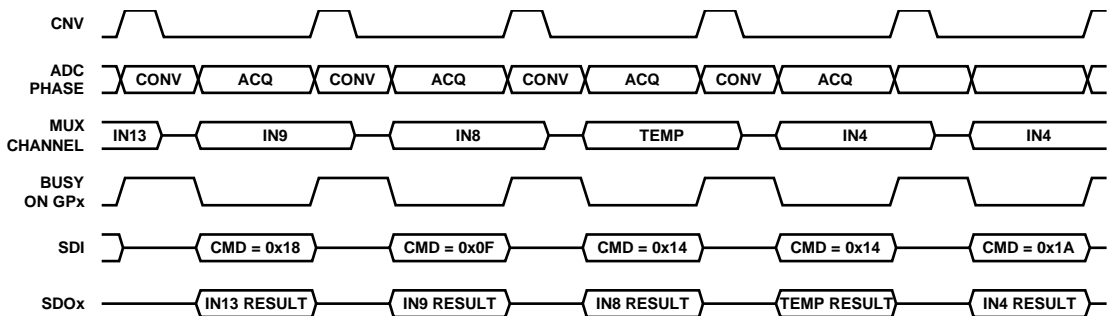Figure 75. Standard Sequencer and Advanced Sequencer SPI Frames with IN0 Oversampling Ratio = N


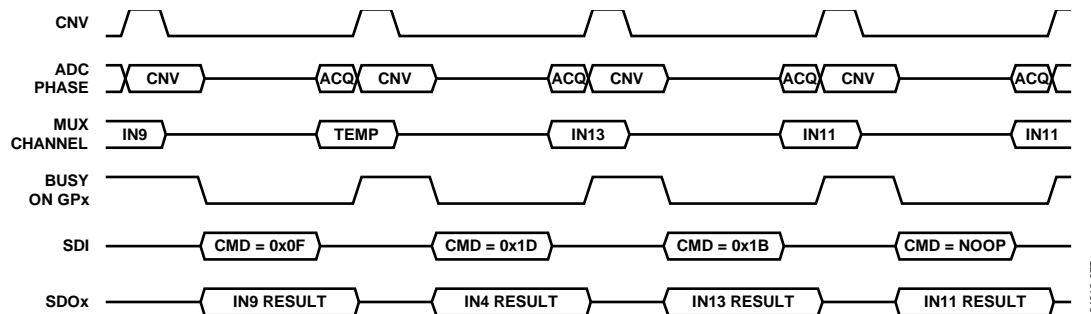
Figure 76. Two-Cycle Command Mode Timing



Figure 77. Single-Cycle Command Mode Timing

# DIGITAL INTERFACE

The AD4695/AD4696 digital interface includes a 4-wire serial peripheral interface (SPI), a convert-start input (CNV), an active-low reset input ($\overline{\text{READY}}$), and general-purpose (GP) pin(s).

The AD4695/AD4696 digital interface has two operating modes: register configuration mode and conversion mode. In register configuration mode, the SPI is used to read from and write to the configuration registers. In conversion mode, the SPI is used to read conversion results and optional status bits. The Register Configuration Mode and Conversion Mode sections describe these operating modes in detail.

The interface logic level is set by the VIO voltage, and can range from 1.2 V to 1.8 V. The AD4695/AD4696 uses SPI mode 3 (CPHA = CPOL = 1).

## REGISTER CONFIGURATION MODE

When in register configuration mode, the digital host can read from and write to the AD4695/AD4696 configuration registers via the SPI. The device must be in register configuration mode to perform register read and write instructions. Register configuration mode is the default mode of operation upon device power-up and reset.

The register configuration mode protocol is flexible and can be configured for efficient access of large blocks of the configuration register map. Each SPI frame consists of at least one instruction phase and one data phase, and an optional 8-bit cyclic redundancy check (CRC) checksum (see the Checksum Protection section). Data is transmitted over the SPI MSB first. The format and order of instruction and data phases is configurable, as described in the subsections of the Register Configuration Mode section. Figure 78 shows an example of a basic SPI frame consisting of the instruction phase, data phase, and optional CRC checksum.

A $\overline{\text{CS}}$ falling edge starts an SPI frame, and a subsequent $\overline{\text{CS}}$ rising edge ends the SPI frame. Data is latched on SDI on SCK rising edges and shifted out on SDO on SCK falling edges. For all SPI transactions, data is aligned MSB first.

Figure 90 shows a detailed timing diagram for register read and write operations via the SPI when the AD4695/AD4696 are in register configuration mode. See Table 2 for the timing specifications shown.

See the Register Details section for a detailed description of the addresses and functions of the AD4695/AD4696 configuration registers.

The 5-bit register configuration mode command switches the device from conversion mode into register configuration mode (see the Register Configuration Mode Command section).
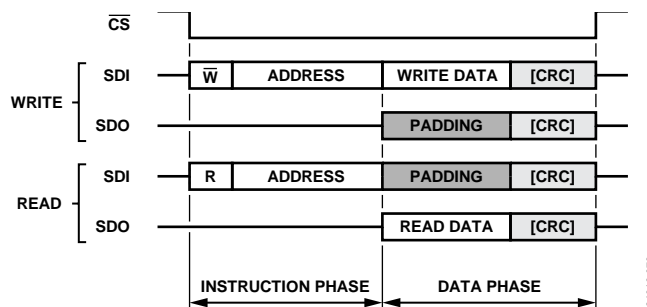


*Figure 78. Basic SPI Frame*

### Instruction Phase

Every SPI frame starts with the instruction phase. The instruction phase immediately follows a $\overline{\text{CS}}$ falling edge (see Figure 78). The instruction phase consists of a read/write bit (R/$\overline{\text{W}}$) followed by a register address word (ADDR). Set the R/$\overline{\text{W}}$ bit high to initiate a read instruction or set the R/$\overline{\text{W}}$ bit low to initiate a write instruction. The register address word specifies the address of the register to be accessed. The register address word is 15 bits in length (long addressing) by default but can be changed to 7 bits in length (short addressing) with the ADDR_LEN bit in the SPI_CONFIG_B register.

When using single instruction mode, each register read or write transaction in an SPI frame begins with an instruction phase. When using streaming mode, only one instruction phase is required per SPI frame to access a set of contiguous registers. See the Single Instruction Mode section and the Streaming Mode section for instructions on selecting and using these modes.

### Data Phase

During the data phase, register data is either shifted out on SDO on SCK falling edges (for register reads) or latched in on SDI on SCK rising edges (for register writes). The data phase can include the data for an entire register or individual bytes of the register (see the Multibyte Register Access section).

If CRC is disabled, register contents are updated immediately after the final SCK rising edge of the data phase. If CRC is enabled, register contents are updated immediately after the final SCK rising edge of the checksum (if the checksum value matches the data in the data phase).

### Address Direction Options

The address direction options control whether the address is set to automatically increment or decrement when accessing multiple bytes of data in a single data phase (for example, when accessing multibyte registers or when streaming mode is enabled). Figure 79 and Figure 80 show SPI frames with both address direction options.

Select between the two address direction options with the ADDR_DIR bit in the SPI_CONFIG_A register. When the ADDR_DIR bit is set to 0, the descending address option is selected and the address decrements after each byte is accessed. When the ADDR_DIR bit is set to 1, the ascending address option is selected and the address increments after each byte is accessed. The descending address option is selected by default.

### Multibyte Register Access

Some of the AD4695/AD4696 configuration registers contain multiple bytes of data stored in adjacent address locations in memory. These registers are referred to as multibyte registers. The address of each multibyte register is defined as the address of its least significant byte (LSByte), but the multibyte register contents extend across multiple register addresses. For example, the STD_SEQ_CONFIG register (address 0x024) is two bytes long, and the address of its LSByte is 0x024 and the address of its MSByte is 0x025. Table 25 specifies whether registers are single-byte or multibyte.

The state of the MB_STRICT bit in the SPI_CONFIG_C register determines whether multibyte registers are treated as a single unit of memory with one register address or if they are treated as multiple registers that are each one byte long with individual register addresses.

When the MB_STRICT bit is set to 0, each byte of a multibyte register must be read from or written to individually, allowing the digital host to access one byte of a multibyte register without accessing the other byte(s). With this setting, all data phases in an SPI frame consist of a single byte rather than the entire multibyte register, and each byte in a multibyte register is directly addressable. The contents of either byte will be updated by a SPI write transaction as long as new data is provided for that entire byte. Figure 82 and Figure 83 show examples where individual bytes in a multibyte register (Address = 0x0043) are accessed over multiple SPI transactions in single instruction mode with MB_STRICT = 0.

When the MB_STRICT bit is set to 1, all bytes of a multibyte register must be read from or written to in the same SPI transaction. With this setting, the data phase includes all bytes when accessing a multibyte register. If the digital host fails to read from or write to the entire multibyte register, the SPI transaction is considered invalid, and the MB_ERROR flag in the SPI_STATUS register is set to 1. This setting ensures that all modes or enable bits associated with a multibyte register are updated simultaneously. The MB_STRICT bit is set to 1 by default.

When the MB_STRICT bit is set to 1, the order in which each byte of a multibyte register is read from or written to depends on the selected address direction option (see the Address Direction Options section). With the descending addresses option selected, the first byte accessed in the data phase is the MSByte of the multibyte register, and each subsequent byte corresponds to the data in the next lowest address. With the ascending addresses option selected, the first byte accessed in the data phase is the LSByte of the multibyte register, and each

subsequent byte corresponds to the data in the next highest address. Figure 79 and Figure 80 show generalized read and write transactions of a multibyte register for both address direction options.

When CRC is enabled, a checksum follows the data phase for each SPI transaction. When the MB_STRICT bit is set to 0, the checksum occurs after each byte of a multibyte register is accessed (see Figure 82 and Figure 87). When the MB_STRICT bit is set to 1, the checksum only occurs after all bytes of the multibyte register are accessed (see Figure 83 and Figure 88).
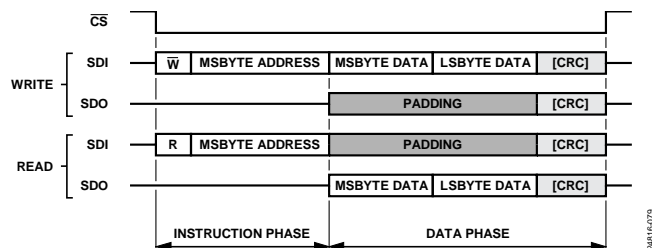


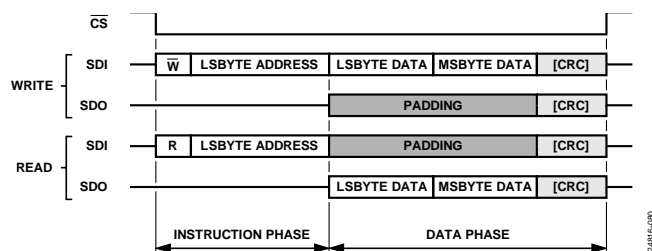*Figure 79. Multibyte Register Access with MB_STRICT = 1 and Descending Address*



*Figure 80. Multibyte Register Access with MB_STRICT = 1 and Ascending Address*

### Streaming Mode

When the INST_MODE bit in the SPI_CONFIG_B register is set to 0, streaming mode is enabled. In streaming mode, only one instruction phase is required per SPI frame, and the register address being read from or written to is automatically updated after each data phase (based on the selected address direction option). The instruction phase is followed by multiple data phases for each register being accessed until the end of the SPI frame. Streaming mode enables efficient access to large, contiguous sections of the configuration register map, such as when updating the advanced sequencer slot registers (AS_SLOTn) to configure the advanced sequencer.

Figure 81 shows a generalized SPI frame for performing multiple register read and write transactions with streaming mode selected. Since there is only one instruction phase per frame in streaming mode, all SPI transactions in a given SPI frame are either all reads or all writes. The checksum is included in each data phase only if CRC is enabled (see the Checksum Protection section).

Figure 82 thru Figure 84 show examples of accessing different parts of the register map with both address direction options and with both MB_STRICT options (see the Multibyte Register Access section).

When streaming mode is active, a specified number of registers can be looped to repeatedly access the same registers multiple times in a single SPI frame. The LOOP_COUNT field in the LOOP_MODE register determines how many registers are accessed before the register address is reset to the starting address (the one specified in the instruction phase). When the MB_STRICT bit is set to 1, a multibyte register is considered one register when looping. When the MB_STRICT bit is set to 0, each byte of a multibyte register is considered one register when looping. Figure 85 shows an example using looping to repeatedly read from the CLAMP_STATUSn registers.

If LOOP_COUNT is set to 0x0, then looping is disabled. If looping is disabled and the descending address option is selected, the address decrements until it reaches Address 0x0000, and the address is set to the highest valued register address available (address 0x013F) on the subsequent byte access. If looping is disabled and the ascending address option is selected, the address increments until it reaches the highest valued register address available (Address 0x013F), and the address is set to Address 0x0000 on the subsequent byte access. Looping is disabled by default.

Note that even when using 7-bit addressing, registers with addresses larger than 0xFF are still accessible in streaming mode, but accessing these registers is generally more efficient using 15-bit addressing

### Single Instruction Mode

When the INST_MODE bit in the SPI_CONFIG_B register is set to 1, single instruction mode is selected. In single instruction mode, each SPI read or write transaction includes an instruction phase to specify whether the transaction is a read or a write and what address is being accessed. Single instruction mode allows the digital host to quickly read from and write to registers with nonadjacent register addresses in a single SPI frame, as opposed to streaming mode, which allows exclusively reading from or writing to registers with adjacent addresses without starting a new SPI frame.

Figure 86 shows a generalized SPI frame for performing multiple register read and write transactions with single instruction mode selected. The checksum is included in each data phase only if CRC is enabled (see the Checksum Protection section).

Figure 87 shows an example of reading from and writing to the MSByte and LSByte of the UPPER_IN1 register (MB_STRICT = 0). Figure 88 and Figure 89 show examples of reading from the UPPER_IN1 register and writing to the UPPER_IN0 register in the same frame with both address direction options (MB_STRICT = 1). Note that the UPPER_INn registers are multibyte registers, and when MB_STRICT is set to 1, then both bytes must be read from or written to in one data phase (see the Multibyte Register Access register section).
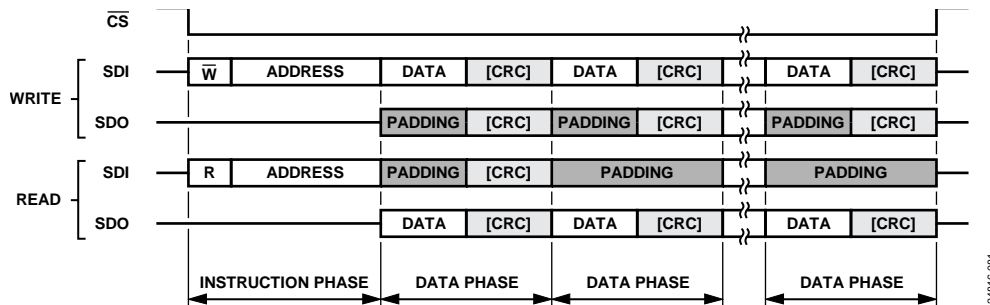


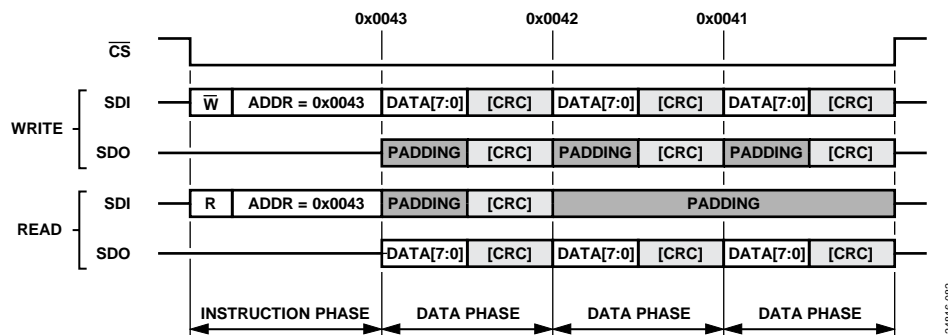*Figure 81. Streaming Mode SPI Frame*



*Figure 82. Streaming Mode SPI Frame, Looping Disabled, Descending Address, MB_STRICT = 0*
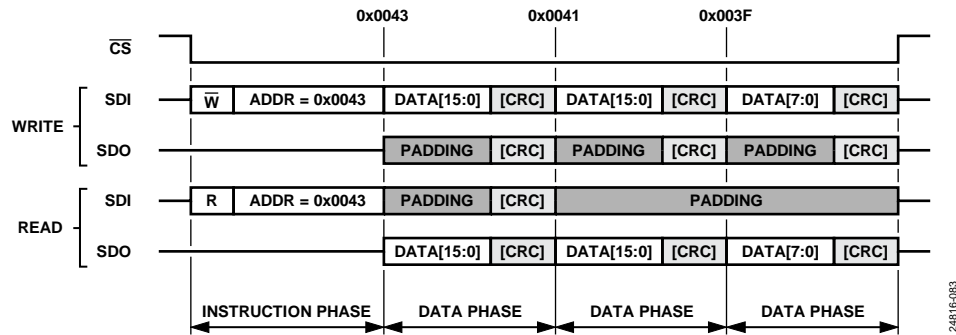
Figure 83. Streaming Mode SPI Frame, Looping Disabled, Descending Address, MB_STRICT = 1
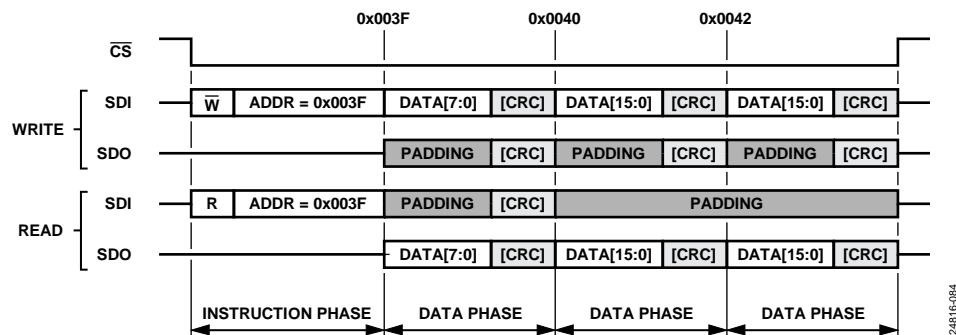


Figure 84. Streaming Mode SPI Frame, Looping Disabled, Ascending Address, MB_STRICT = 1
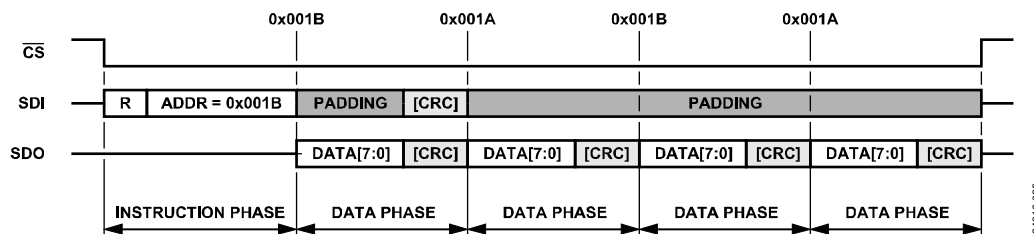


Figure 85. Streaming Mode SPI Frame, Looping Enabled, LOOP_COUNT = 7, Descending Address
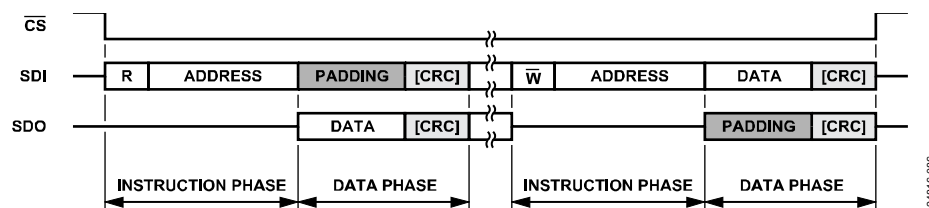


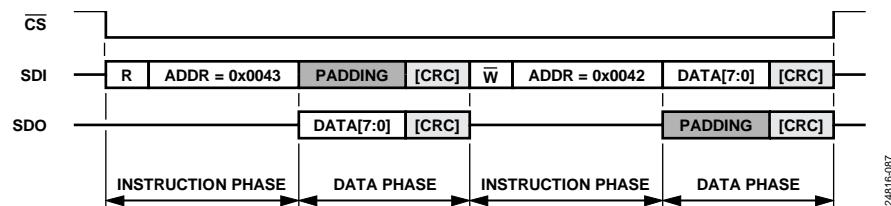Figure 86. Single Instruction Mode SPI Frame



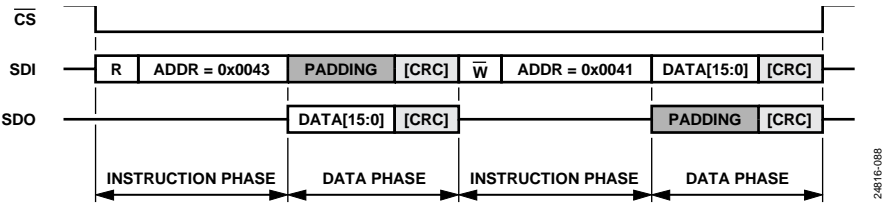Figure 87. Single Instruction Mode SPI Frame, MB_STRICT = 0

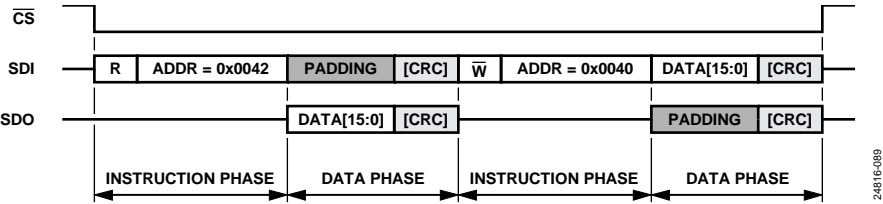*Figure 88. Single Instruction Mode SPI Frame, MB_STRICT = 1, Descending Address*



*Figure 89. Single Instruction Mode SPI Frame, MB_STRICT = 1, Ascending Address*

### Checksum Protection

The AD4695/AD4696 includes optional error checking based on an 8-bit CRC in register configuration mode. When CRC is enabled, an 8-bit checksum code is appended to the data phase of each register read or write transaction. The value of the checksum is calculated from the data read or written over the SPI, and therefore allows the AD4695/AD4696 and the digital host to detect corrupted data. If the checksum does not match the corresponding register data, the register read or write is considered invalid.

Figure 81 shows a generalized SPI frame for performing register reads and writes with streaming mode selected, including the CRC checksum. Figure 86 shows a generalized SPI frame for performing register reads and writes with single instruction mode selected, including the CRC checksum. Note that the checksums on SDI shown in both Figure 81 and Figure 86 are sent from the digital host to the AD4695/AD4696, and the digital host must send a valid checksum during the SPI read and write transactions pictured. The only exception is when performing multiple register reads with streaming mode selected, where the digital host is only required to send a CRC on SDI for the first transaction (see Figure 81).

When the AD4695/AD4696 receives a checksum that does not match its corresponding SPI transaction, the transaction is considered invalid, and the CRC_ERROR bit in the SPI_STATUS register is set to 1. The CRC_ERROR bit is a write-1-to-clear bit, meaning it must be written to 1 to be cleared.

When a write transaction is considered invalid, then register contents are not updated. When a read transaction is considered invalid, then the digital host should ignore the received register data and attempt the register read transaction again. Read-to-clear bits are also not cleared unless the register read transaction is considered valid (for example, the HI_INn and LO_INn bits in the ALERT_STATUSn registers).

When streaming mode and CRC are both enabled, if an invalid checksum is received for a given SPI transaction, all subsequent SPI transactions are considered invalid for the remainder of the SPI frame (until $\overline{CS}$ is brought high).

CRC is enabled with the CRC_EN and CRC_EN_N fields in the SPI_CONFIG_C register. To enable CRC, CRC_EN must be set to 0x1 and CRC_EN_N must be set to 0x2. CRC is disabled for all other combinations of CRC_EN and CRC_EN_N.

The AD4695/AD4696 expects checksums to be included in each SPI transaction immediately after CRC is enabled. Write to the SPI_CONFIG_C register to enable CRC before writing to any other registers, then read the SPI_CONFIG_C register assuming that CRC has been enabled. If the SPI master receives the correct state of the CRC_EN and CRC_EN_N fields and a valid checksum, CRC has been enabled, and the SPI master can begin configuring the remaining configuration registers.

The following CRC polynomial is used to calculate the checksums:

$$X^8 + x^2 + x + 1$$

Each SPI transaction has a corresponding code, and the polynomial is applied to that code to generate a checksum. The codes consist of an 8-bit seed value appended to data from the SPI transaction. Table 12 shows the data and seed values for each possible type of SPI transaction.

In single instruction mode, the seed for all CRCs is 0xA5. In streaming mode, the seed for the first CRC in the frame is also 0xA5, but the seed for the remaining CRCs in the frame is the LSByte of the register address being accessed. If MB_STRICT is set to 1 and a multibyte register is accessed, the register address used for the seed depends on the selected address direction option: with descending address, the address of the MSByte is used, and with ascending address, the address of the LSByte is used. For example, in both Figure 83 and Figure 84, the second data phase includes data from the UPPER_IN0 register, but the seed used for the checksum is 0x41 with the descending address option (Figure 83) and 0x40 with the ascending address option (Figure 84).

**Table 12. CRC Input Values for SPI Modes and Transactions**

| SPI Transaction Type | Pin | Single Instruction Mode or First CRC with Streaming Mode | Subsequent CRCs with Streaming Mode |
|---|---|---|---|
| Write | SDI | SPI Data = [Instruction Phase, Data Phase], Seed = 0xA5 | SPI Data = Data Phase, Seed = LSByte of Current Register Address |
|  | SDO | SPI Data = [Instruction Phase, Data Phase], Seed = 0xA5 | SPI Data = Data Phase, Seed = LSByte of Current Register Address |
| Read | SDI | SPI Data = [Instruction Phase, Padding], Seed = 0xA5 | n/a |
|  | SDO | SPI Data = [Instruction Phase, Data Phase], Seed = 0xA5 | SPI Data = Data Phase, Seed = LSByte of Current Register Address |

### Register Read and Write Timing Diagrams

Figure 90 shows a timing diagram for the SPI when the AD4695/AD4696 is in register configuration mode. See Table 2 for the timing specifications pictured in Figure 90.

Register read and write transactions are framed by $\overline{CS}$. While $\overline{CS}$ is high, SCK edges are ignored, and SDO is high impedance. A falling edge on $\overline{CS}$ begins an SPI frame, and data on SDI is latched on SCK rising edges while data is shifted out on SDO on SCK falling edges. A rising edge on $\overline{CS}$ ends the SPI frame and forces SDO to high impedance.

The first phase of an SPI frame immediately following a $\overline{CS}$ is the instruction phase. The instruction phase is followed by the data phase. For SPI read transactions, the register contents are shifted out on SDO during the data phase. For SPI write transactions, the register contents are latched in on SDI during the data phase. See the Streaming Mode and Single Instruction Mode sections for a detailed description of the order of instruction and data phases in each SPI frame.

The length of the address in the instruction phase (represented by M in Figure 90) is set by the ADDR_LEN bit in the SPI_CONFIG_B register (see the Instruction Phase section).

The length of the data phase (represented by N in Figure 90) depends on whether CRC is enabled and the length of the register being accessed (see the Checksum Protection and Multibyte Register Access sections).

The AD4695/AD4696 ignore the state of CNV when in register configuration mode. The Entering Conversion Mode section describes the process for putting the AD4695/AD4696 in conversion mode.

### Entering Conversion Mode

To put the AD4695/AD4696 in conversion mode, set the SPI_MODE bit in the SETUP register to 1. Once the SPI_MODE bit is set to 1, the SPI frame immediately terminates, and the device enters conversion mode. No further register reads or writes can occur until the AD4695/AD4696 enter register configuration mode again.

The digital host must provide a delay specified by $t_{SCKCNV}$ after the final SCK rising edge of the register write before initiating conversions with a CNV rising edge (see Table 2 and Figure 90).
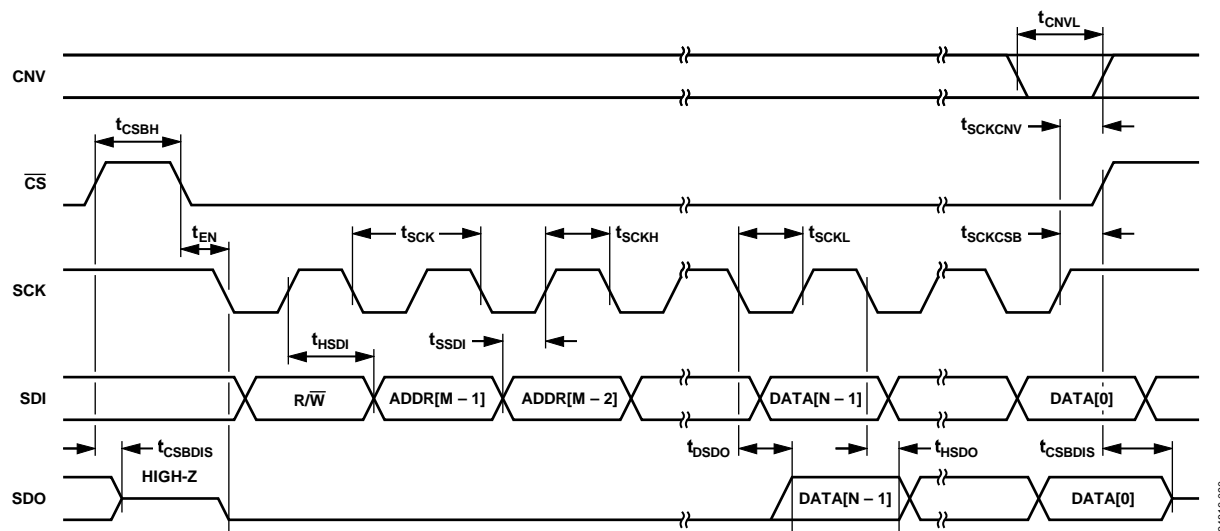


*Figure 90. Register Configuration Mode Timing Diagram*

## CONVERSION MODE

When the AD4695/AD4696 are in conversion mode, CNV rising edges initiate conversions on the selected channel, and the channel sequencing logic updates the multiplexer to the next channel (see the Converter Operation and Channel Sequencing Modes sections). The AD4695/AD4696 enters conversion mode when the SPI_MODE bit in the SETUP register is set to 1.

In conversion mode, the AD4695/AD4696 SPI is used to read ADC results and write the 5-bit SDI commands shown in Table 14. Figure 91 thru Figure 96 show timing diagrams for SPI frames relative to performing conversions. The CNV and $\overline{CS}$ pins can be tied together to enable interfacing with a single 4-wire SPI port (see Figure 96). Each ADC result is available until the next CNV rising edge occurs.

An optional set of 5 status bits can be appended to the ADC data. The status bits include channel information, the overvoltage clamp flag and a threshold detection alert indicator. See the Status Bits section for a description of the status bits and how they are enabled.

In conversion mode, the general-purpose pin(s) on the AD4695/AD4696 can be assigned as additional serial data outputs, reducing the SCK frequency required to shift out the ADC result plus optional status bits before the next conversion occurs. See the Serial Data Output Modes section for a description of the options available on both package options of the AD4695/AD4696 and how to enable these modes.

The general-purpose pin(s) on the AD4695/AD4696 can also be assigned as either the busy indicator or the threshold detection alert indicator. Figure 91 thru Figure 96 show the relative timing of the CNV signal and the busy and alert indicators when they are assigned to general-purpose pin(s). The General-Purpose Pin section describes how to set the general-purpose pin(s) to the desired function.

When autocycle mode is enabled, the AD4695/AD4696 generate their own internal convert-start signal to autonomously perform conversions without a CNV signal from the digital host (see the Autocycle Mode section).

### Status Bits

A set of five status bits can be appended to the end of each conversion result. The status bits allow the digital host to monitor the status of the analog inputs without interrupting analog-to-digital conversions. Table 13 shows the names and descriptions of the status bits.

By default, the OV_ALT status bit indicates the status of the overvoltage clamp flags (the bitwise logical OR of the CLAMP_FLAG and COM_CLAMP_FLAG bits in the STATUS register). When the OV_ALT_MODE bit in the GP_MODE register is set to 1, then the OV_ALT status bit is the logical OR of the CLAMP_FLAG bit and the threshold detection alert indicator (TD_ALERT bit in the STATUS register). The digital host can monitor the state of the OV_ALT bit to detect and respond to out-of-range events.

The INX bits indicate which of the 16 analog inputs the conversion result corresponds to. The values for INX range from 0 to 15, and correspond to IN0–IN15, respectively. An INX value of 15 corresponds to either IN15 or the temperature sensor. The INX bits can be used by the digital host to align the ADC data with the sequence of analog input channels.

Set the STATUS_EN bit in the SETUP register to 1 to enable the status bits. The status bits are disabled by default. When the status bits are enabled, the serial data output word extends to 24 bits, where Bit 20 to Bit 24 contain the status bits.

Table 13. Status Bits Names and Descriptions

| Status Word Index | Bit Name | Description |
|---|---|---|
| Bit 4 | OV_ALT | Active High. Indicates the status of the overvoltage protection clamp flag and (if enabled) the status of the threshold detection alert indicator. |
| Bits [3:0] | INX[3:0] | Indicates what analog input channel the ADC data corresponds to (IN0 thru IN15). |

Table 14. Conversion Mode Commands

| Channel Sequencing Mode | CMD | Description |
|---|---|---|
| Two Cycle Command Mode and Single Cycle Command Mode | 0x00 thru 0x09, 0x0B thru 0x0E | No-op |
| | 0x0A | Register Configuration Mode Command |
| | 0x0F | Temperature Sensor Channel Selection |
| | 0x10 thru 0x1F | IN0 thru IN15 Channel Selection |
| Standard Sequencer and Advanced Sequencer | 0x00 thru 0x09, 0x0B thru 0x1F | No-op |
| | 0x0A | Register Configuration Mode Command |

### Serial Data Output Modes

The AD4695/AD4696 digital interface allows for clocking out ADC data on more than one serial data output, reducing the number of SCK periods required to access the full ADC result and allowing slower SCK frequencies. The two serial data output modes include single-SDO mode and dual-SDO mode. In single-SDO mode, the ADC results are only shifted out on SDO. In dual-SDO mode, ADC results are shifted out on SDO and BSY_ALT_GP0 in parallel.

Table 15 shows which general-purpose pins are used for each serial data output signal for each of the serial data output modes. Table 17, Table 18, and Table 19 show the formatting of conversion results for all combinations of serial data output modes, status bits, and oversampling ratio options. The value of the blank cells in Table 17, Table 18, and Table 19 depends on the setting of SDO_STATE, as described in the Conversion Mode Timing Diagrams section.

The serial data output modes only apply when the AD4695/AD4696 is in conversion mode. In register configuration mode, register read data is always shifted out serially on SDO only.

The SDO_MODE field in the SETUP register determines which of the serial data output modes are selected. Table 16 shows the values of SDO_MODE and the corresponding serial data output modes.

**Table 15. Serial Data Output Mode Pin Assignments**

| Mode | Signal | LFCSP Pins |
|---|---|---|
| Single-SDO Mode | SDO0 | SDO |
| Dual-SDO Mode | SDO1 | BSY_ALT_GP0 |
| | SDO0 | SDO |

**Table 16. SDO_MODE Values vs. Serial Data Output Mode**

| SDO_MODE | Mode (LFCSP) |
|---|---|
| 0x0 | Single-SDO Mode |
| 0x1 | Dual-SDO Mode |
| 0x2 | Single-SDO Mode |
| 0x3 | Single-SDO Mode |

**Table 17. Single-SDO Mode Data Output Format**

| OSR Setting | Status Bits | Signal | 1 | 2 | 3 | … | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | | | **SCK Falling Edge Number** | | | | |
| 1 | Disabled | SDO0 | D15 | D14 | D13 | … | D1 | D0 | | | | | | | | |
| 4 | Disabled | SDO0 | D16 | D15 | D14 | … | D2 | D1 | D0 | | | | | | | |
| 16 | Disabled | SDO0 | D17 | D16 | D15 | … | D3 | D2 | D1 | D0 | | | | | | |
| 64 | Disabled | SDO0 | D18 | D17 | D16 | … | D4 | D3 | D2 | D1 | D0 | | | | | |
| 1 | Enabled | SDO0 | D15 | D14 | D15 | … | D1 | D0 | 0 | 0 | 0 | OV_ALT | INX[3] | INX[2] | INX[1] | INX[0] |
| 4 | Enabled | SDO0 | D16 | D15 | D16 | … | D2 | D1 | D0 | 0 | 0 | OV_ALT | INX[3] | INX[2] | INX[1] | INX[0] |
| 16 | Enabled | SDO0 | D17 | D16 | D17 | … | D3 | D2 | D1 | D0 | 0 | OV_ALT | INX[3] | INX[2] | INX[1] | INX[0] |
| 64 | Enabled | SDO0 | D18 | D17 | D16 | … | D4 | D3 | D2 | D1 | D0 | OV_ALT | INX[3] | INX[2] | INX[1] | INX[0] |

**Table 18. Dual-SDO Mode Data Output Format**

| OSR Setting | Status Bits | Signal | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | **SCK Falling Edge Number** | | | | | | | |
| 1 | Disabled | SDO1 | D15 | D13 | D11 | D9 | D7 | D5 | D3 | D1 | | | | |
| | | SDO0 | D14 | D12 | D10 | D8 | D6 | D4 | D2 | D0 | | | | |
| 4 | Disabled | SDO1 | D16 | D14 | D12 | D10 | D8 | D6 | D4 | D2 | D0 | | | |
| | | SDO0 | D15 | D13 | D11 | D9 | D7 | D5 | D3 | D1 | | | | |
| 16 | Disabled | SDO1 | D17 | D15 | D13 | D11 | D9 | D7 | D5 | D3 | D1 | | | |
| | | SDO0 | D16 | D14 | D12 | D10 | D8 | D6 | D4 | D2 | D0 | | | |
| 64 | Disabled | SDO1 | D18 | D16 | D14 | D12 | D10 | D8 | D6 | D4 | D2 | D0 | | |
| | | SDO0 | D17 | D15 | D13 | D11 | D9 | D7 | D5 | D3 | D1 | | | |
| 1 | Enabled | SDO1 | D15 | D13 | D11 | D9 | D7 | D5 | D3 | D1 | 0 | 0 | INX[3] | INX[2] |
| | | SDO0 | D14 | D12 | D10 | D8 | D6 | D4 | D2 | D0 | 0 | OV_ALT | INX[1] | INX[0] |
| 4 | Enabled | SDO1 | D16 | D14 | D12 | D10 | D8 | D6 | D4 | D2 | D0 | 0 | INX[3] | INX[2] |
| | | SDO0 | D15 | D13 | D11 | D9 | D7 | D5 | D3 | D1 | 0 | OV_ALT | INX[1] | INX[0] |
| 16 | Enabled | SDO1 | D17 | D15 | D13 | D11 | D9 | D7 | D5 | D3 | D1 | 0 | INX[3] | INX[2] |
| | | SDO0 | D16 | D14 | D12 | D10 | D8 | D6 | D4 | D2 | D0 | OV_ALT | INX[1] | INX[0] |
| 64 | Enabled | SDO1 | D18 | D16 | D14 | D12 | D10 | D8 | D6 | D4 | D2 | D0 | INX[3] | INX[2] |
| | | SDO0 | D17 | D15 | D13 | D11 | D9 | D7 | D5 | D3 | D1 | OV_ALT | INX[1] | INX[0] |

### Conversion Mode Timing Diagrams

Figure 91 thru Figure 96 show detailed timing diagrams for performing analog-to-digital conversions when the AD4695/AD4696 are in conversion mode with each of the serial data output mode options (with autocycle mode disabled).

When the AD4695/AD4696 are in conversion mode, a CNV rising edge initiates a conversion and enters the conversion phase (see the Converter Operation section). When a conversion is initiated, it continues until completion regardless of the state of CNV. When the standard sequencer, advanced sequencer, or two cycle command mode are enabled, the AD4695/AD4696 enter the acquisition phase before the conversion phase is complete. When single cycle command mode is enabled, the AD4695/AD4696 enter the acquisition phase after the 6th SCK rising edge in the SPI frame. Figure 91 through Figure 94 and Figure 96 show the acquisition time ($t_{ACQ}$) when the standard sequencer, advanced sequencer or two cycle command mode are enabled. Figure 95 shows $t_{ACQ}$ when single cycle command mode is enabled.

$\overline{CS}$ frames the conversion result data. While $\overline{CS}$ is high, SCK edges are ignored, and all pins assigned as serial data outputs (SDO and general-purpose pins) are high impedance. While $\overline{CS}$ is low, data is clocked out with the MSB first on the serial data output(s) on SCK falling edges, and data is latched in on SDI on SCK rising edges.

CNV and $\overline{CS}$ can be tied together and driven by the chip-select of the SPI master to minimize the number of digital signals required to interface with the AD4695/AD4696 (see the SPI Peripheral Connections section). Figure 96 shows a timing diagram of the AD4695/AD4696 interfacing with a 4-wire SPI with the CNV and $\overline{CS}$ signals tied together.

The conversion phase must be complete before the digital host provides the first SCK falling edge. The digital host can use the busy indicator falling edge to detect the end of the conversion phase and to begin clocking out the ADC results. Otherwise, the digital host must include a delay dictated by the conversion time specification ($t_{CONVERT}$ in Table 2) between the CNV rising edge and the first SCK falling edge.

The 5-bit SDI commands shown in Table 14 are latched in on SDI on the first five SCK rising edges in the SPI frame. The register configuration mode command instructs the AD4695/AD4696 to exit conversion mode and enter register configuration mode (see the Register Configuration Mode Command section). The channel select commands in Table 14 are only used when two-cycle command mode or single-cycle command mode are enabled, and are interpreted as no-op commands when the standard sequencer or the advanced sequencer are enabled (see the Channel Sequencing Modes section).

To ensure optimal performance, there must be a sufficient delay between the final SCK edge and the next CNV rising edge, and there must be no SCK activity until the conversion time has elapsed (see $t_{SCKCNV}$ in Table 2 and Figure 91 thru Figure 96).

The SDO_STATE bit in the SETUP register determines the behavior of the serial data output(s) at the beginning and the end of the conversion mode SPI frames. When the SDO_STATE bit is set to 0, the serial data output(s) hold their final value(s) until the MSB of the next conversion result is clocked out. The serial data output(s) remain in this state even if multiple extra SCK falling edges occur after the full result is shifted out. The serial data output(s) are forced to high impedance when $\overline{CS}$ is brought high, but they return to their previous state after $\overline{CS}$ is brought low again. Figure 91, Figure 93, and Figure 101 show the behavior of the serial data output(s) when SDO_STATE is set to 0. SDO_STATE is set to 0 by default.

When SDO_STATE is set to 1, the busy indicator is enabled on the serial data output(s) (see the Busy Indicator section). The serial data output(s) are forced to high impedance if any SCK falling edges occur after the final bit(s) of the result are already clocked out, or when CNV or $\overline{CS}$ is brought high. After a CNV rising edge initiates a conversion, the serial data output(s) remain high impedance until the conversion phase is complete and the result is available to be read over the SPI. The serial data output(s) are driven low when the data is ready. If the current selected channel has an oversampling ratio greater than 1, then the serial data output(s) are driven low after the oversampled result is ready. Note that $\overline{CS}$ must be driven low for the busy indicator to appear on the serial data output(s).

When the busy indicator is enabled on a general purpose pin, the general purpose pin is driven high after a CNV rising edge, and is driven low when the conversion is complete (see the Busy Indicator on section). The signal labeled BUSY in Figure 91 thru Figure 95 represents the general-purpose pin(s) assigned as the busy indicator. Figure 75 in the Channel Sequencing Modes section shows the relative timing of the CNV rising edge and the busy indicator on a general purpose pin for oversampling ratios of 1 and greater than 1.

When the threshold detection alert indicator is enabled on a general purpose pin, the general purpose pin reflects the value of the TD_ALERT bit in the STATUS register, and if the general purpose pin is driven high, it indicates that the most recent conversion result violated one of the threshold detection limits. The signal labeled ALERT in Figure 91 thru Figure 95 represents the general-purpose pin(s) assigned as the alert indicator. Figure 75 in the Channel Sequencing Modes section show the relative timing of the CNV rising edge and the alert indicator on a general purpose pin for oversampling ratios of 1 and greater than 1.

*Register Configuration Mode Command*

The register configuration mode command is a 5-bit command written on SDI that instructs the device to exit conversion mode and enter register configuration mode. The register configuration mode command is 0x0A. Figure 97 shows the relative timing of the register configuration mode command and the AD4695/AD4696 entering register configuration mode.

The register configuration mode command is clocked in on SDI on the first five SCK rising edges after a conversion. When the register configuration mode command is received, the subsequent rising edge on $\overline{CS}$ places the AD4695/AD4696 in register configuration mode. The digital host must wait for the $t_{REGCONFIG}$ delay (shown in Figure 97 and Table 2) to elapse between the 5th SCK rising edge and the $\overline{CS}$ rising edge.
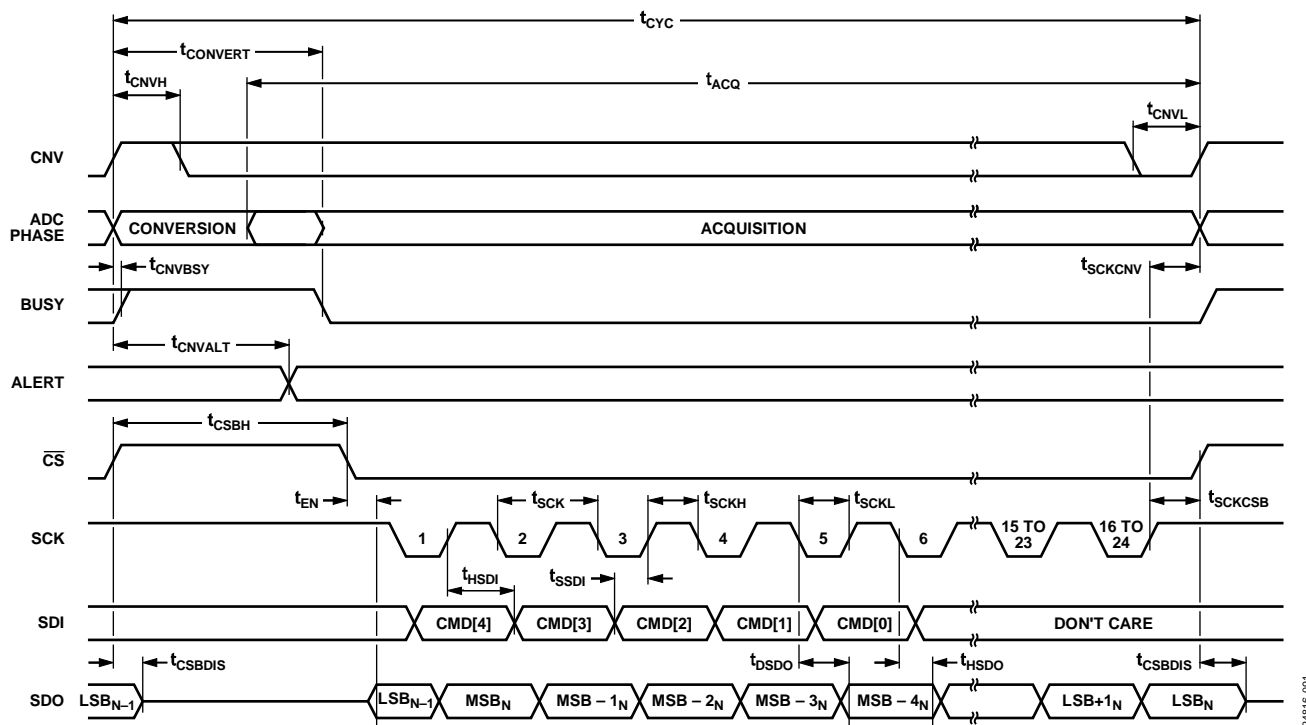


*Figure 91. Conversion Mode Timing Diagram, Single-SDO Mode, SDO_STATE = 0*
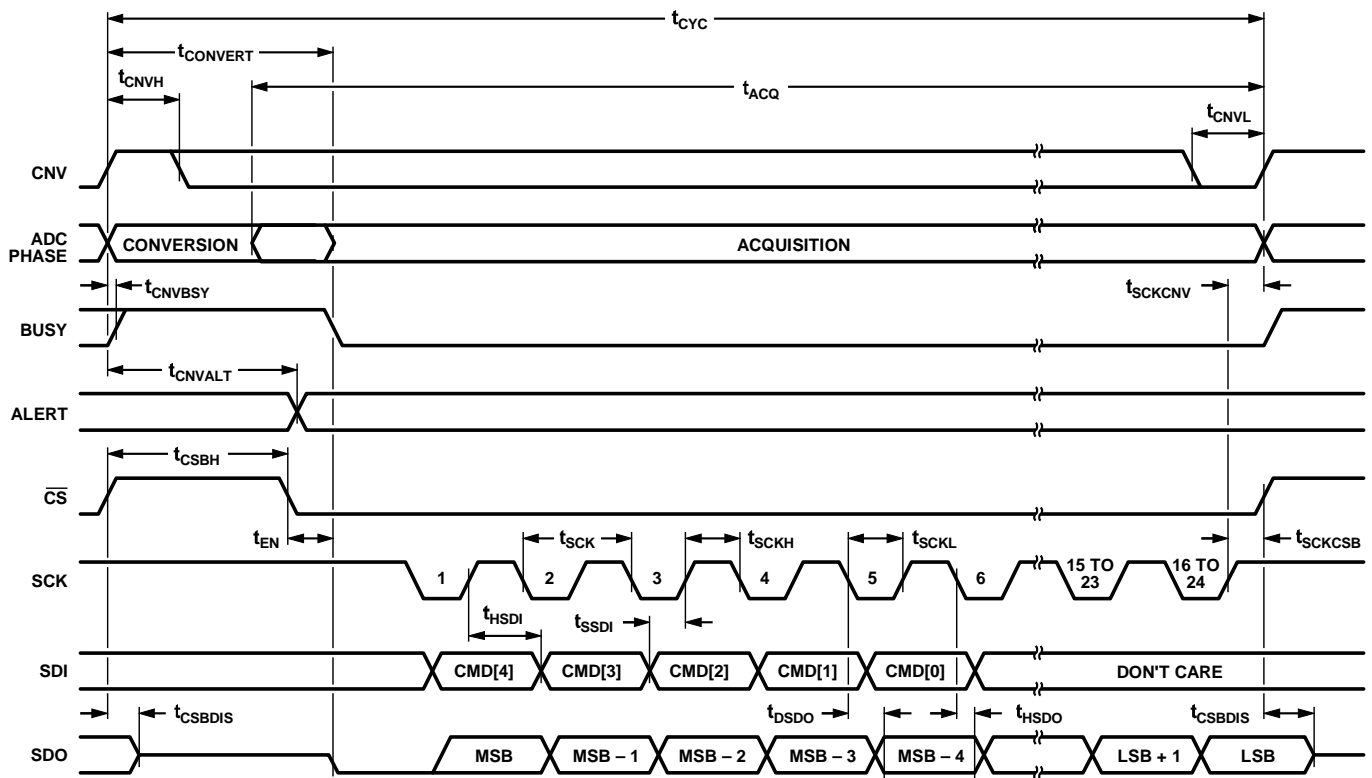
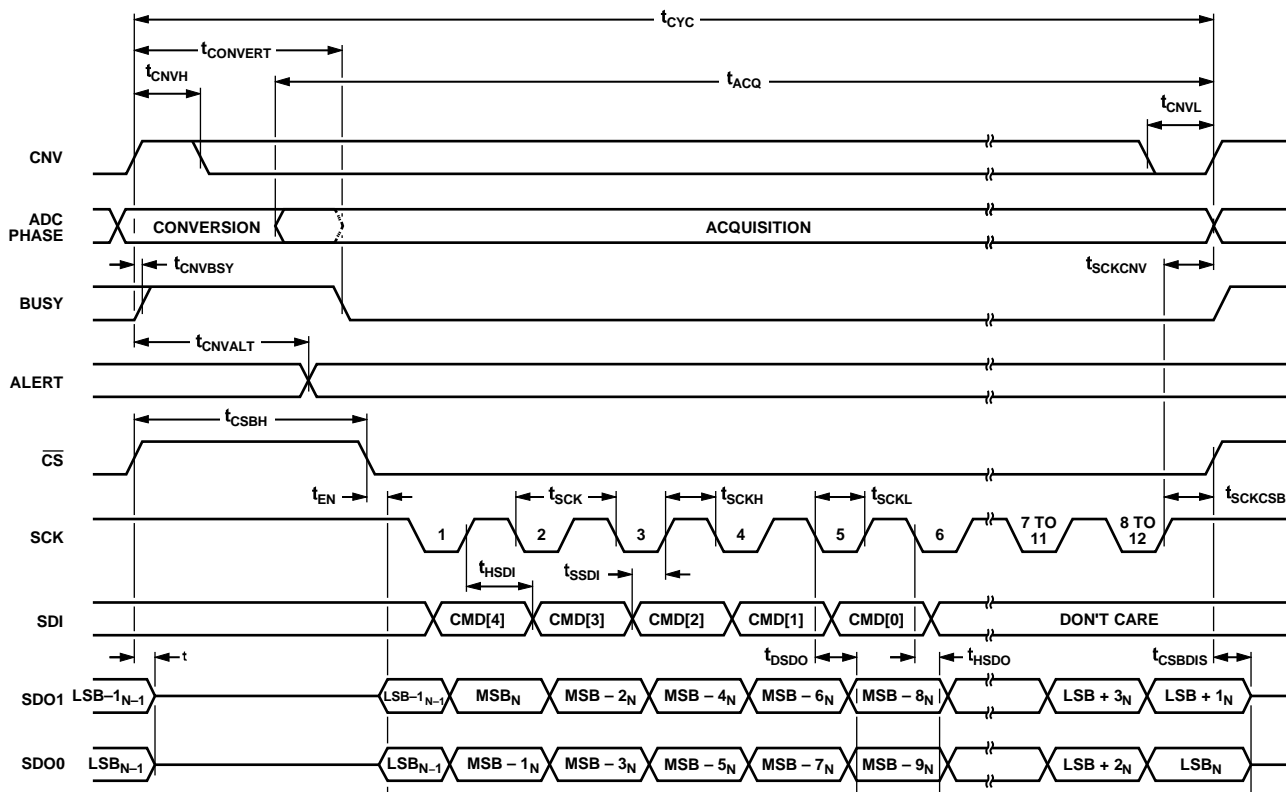Figure 92. Conversion Mode Timing Diagram, Single-SDO Mode, SDO_STATE = 1



Figure 93. Conversion Mode Timing Diagram, Dual-SDO Mode, SDO_STATE = 0
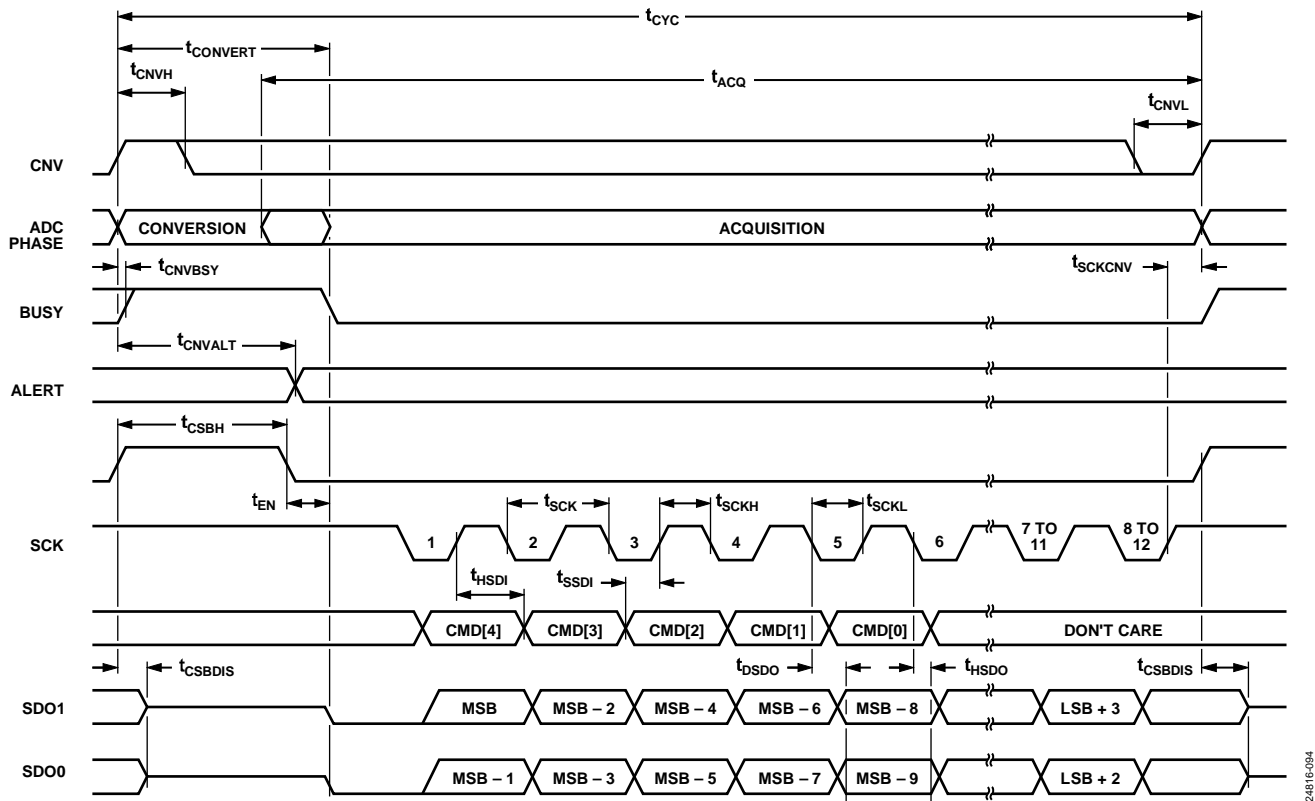
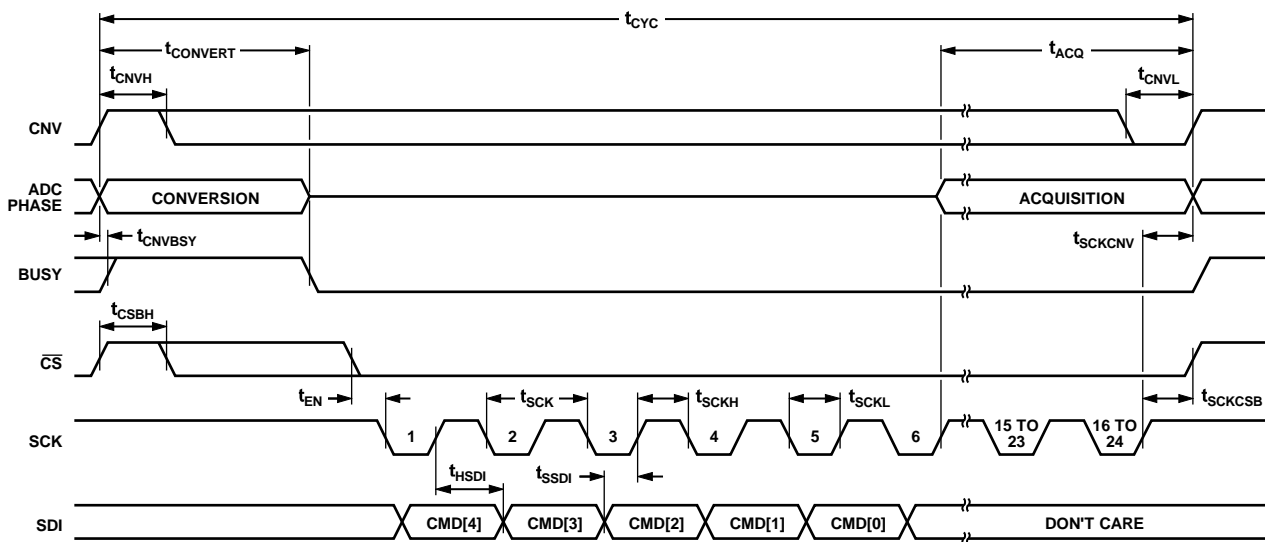*Figure 94. Conversion Mode Timing Diagram, Dual-SDO Mode, SDO_STATE = 1*



*Figure 95. Conversion Mode Timing Diagram, Single Cycle Command Mode Enabled*
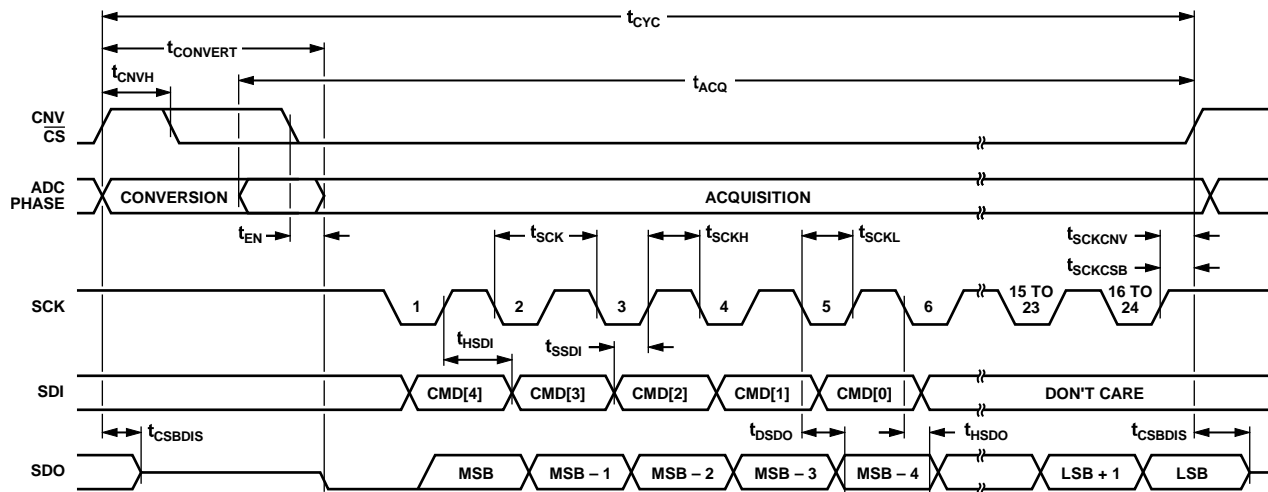
*Figure 96. Conversion Mode Timing Diagram with 4-Wire SPI, Single-SDO Mode, SDO_STATE = 1*
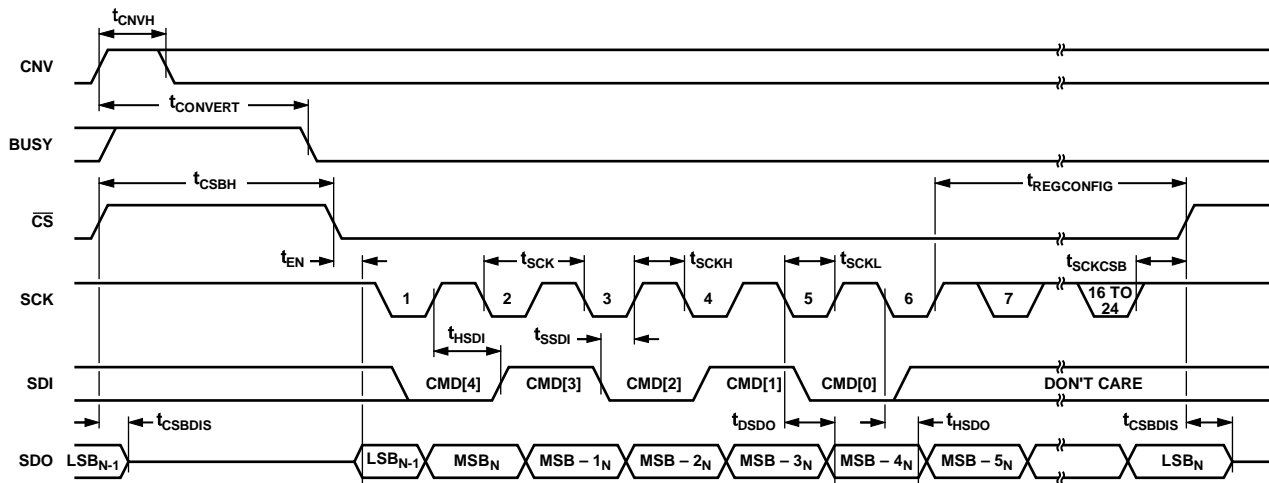
*Figure 97. Conversion Mode Timing Diagram, Register Configuration Mode Command*

## AUTOCYCLE MODE

The AD4695/AD4696 can be configured to convert autonomously on a programmable sequence of its analog inputs, making it the ideal mode of operation for system monitoring. When autocycle mode is enabled, the AD4695/AD4696 generate an internal clock which acts as the convert-start signal, and the digital host is not required to generate a signal on CNV. The internal convert-start clock is enabled when conversion mode is selected, and disabled when register configuration mode is selected, so conversions only occur when the AD4695/AD4696 are in conversion mode.

Autocycle mode is enabled when the AC_EN bit in the AC_CTRL register is set to 1 when the AD4695/AD4696 enter conversion mode. There are eight options for the period of the internal convert-start signal. The convert-start signal period is selected with the AC_CYC field in the AC_CTRL register. Table 19 shows the conversion period and corresponding sample rates for each value of AC_CYC.

**Table 19. Autocycle Mode Conversion Period Options**

| AC_CLK[2:0] Value | Conversion Period | Sample Rate |
|---|---|---|
| 0x0 | 10 µs | 100 kSPS |
| 0x1 | 20 µs | 50 kSPS |
| 0x2 | 40 µs | 25 kSPS |
| 0x3 | 80 µs | 12.5 kSPS |
| 0x4 | 100 µs | 10 kSPS |
| 0x5 | 200 µs | 5 kSPS |
| 0x6 | 400 µs | 2.5 kSPS |
| 0x7 | 800 µs | 1.25 kSPS |

Autocycle mode can be used in conjunction with the busy indicator, threshold detection alerts, and the standard or advanced sequencers to significantly reduce overhead for the digital host system. The threshold detection alert indicator can be assigned to a general-purpose pin and used as an interrupt to indicate a pre-determined out-of-bounds event. The threshold detection interrupt service routine can optionally trigger an SPI instruction to read back the most recent

conversion result and exit conversion mode to determine the specific type of out-of-bounds event using the alert indicator registers (ALERT_STATUS1 thru ALERT_STATUS4).

Note that the SPI transactions when autocycle mode is enabled must adhere to the timing specifications of conversion mode (see the Conversion Mode section and Table 2). Figure 98 shows a timing diagram with the recommended general-purpose pin assignments to synchronize the digital host with the AD4695/AD4696 with autocycle mode enabled. Either the alert indicator or the busy indicator can be assigned to the BSY_ALT_GP0 pin to determine when the digital host can initiate the SPI transaction. See the General-Purpose Pin section for a description of configuring the BSY_ALT_GP0 to output the busy indicator or the alert indicator.

As shown in Figure 98, SPI transactions when using autocycle mode must not start before the conversion time ($t_{CONVERT}$) has elapsed. The busy indicator or the alert indicator must be used to ensure the digital host is synchronized to internal convert-start clock (see the SPI Peripheral Synchronization in Autocycle Mode section). The SCK rate must also be fast enough to complete the

desired SPI transaction before the next conversion begins (see the Conversion Mode SPI Clock Frequency Requirements section).

The $t_{ACBSY}$ specification dictates how long the busy indicator is low between two conversions when autocycle mode is enabled. The $t_{SCKCNV}$ specification dictates how much time must be given between the final SCK rising edge of the SPI transaction and the start of the next conversion.

The $t_{CNVALT}$ specification indicates the delay between the start of the conversion and when the alert indicator state is updated. A rising edge of the alert indicator does not directly imply that the AD4695/AD4696 interface is ready for an SPI transaction, but it can be used as an interrupt to trigger an SPI transaction if the transaction is completed before the remainder of the $t_{CYC}$ timing specification elapses.

Autocycle mode is intended to be used with the standard sequencer and advanced sequencer to minimize the overhead for the digital host. Autocycle mode can be used with two-cycle command mode and single-cycle command mode, but this requires that the digital host transmit the 5-bit SDI commands for selecting channels.
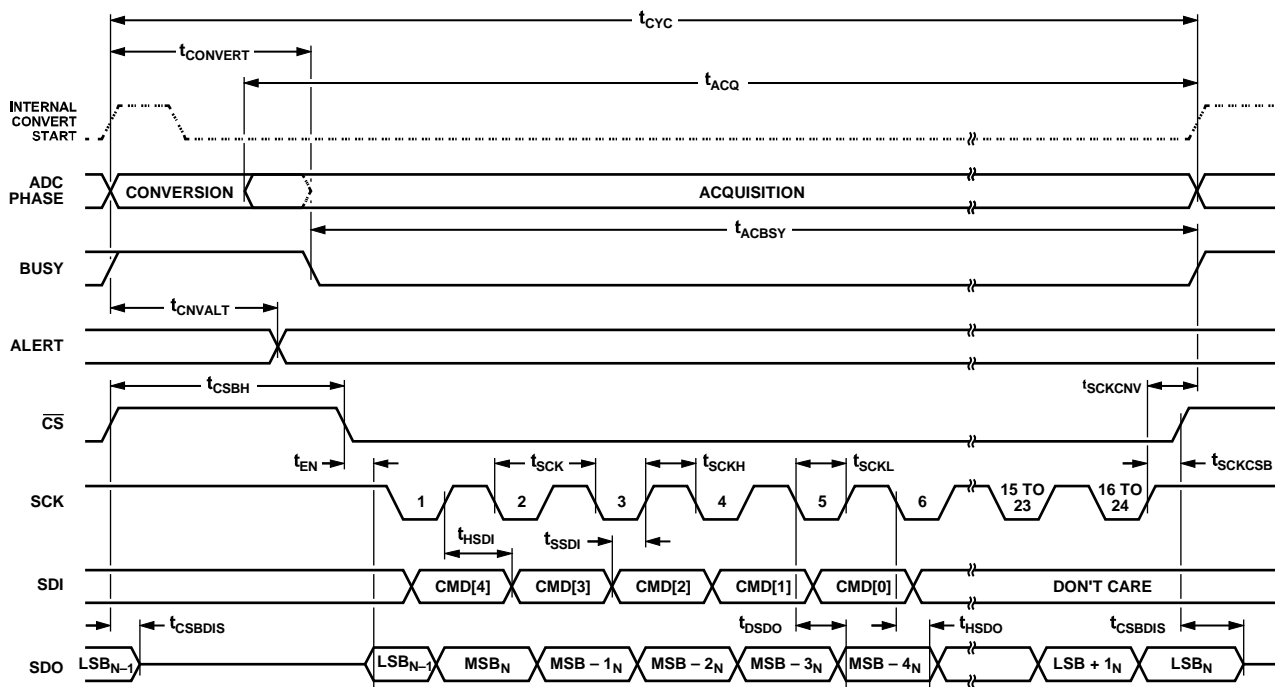


*Figure 98. Conversion Mode Timing Diagram with Autocycle Mode Enabled (Single-SDO Mode, SDO_STATE = 0)*

## GENERAL-PURPOSE PIN

Table 20 shows the functions available on the BSY_ALT_GP0 pin, plus their priority (lower numbers indicate higher priority). To configure the BSY_ALT_GP0 for a given function, all higher priority functions must be disabled. The Busy Indicator, Threshold Detection and Alert Indicators section, the Serial Data Output Modes section and the GPIO section describe the behavior of the BSY_ALT_GP0 when it is configured for each of the functions shown in Table 20.

When the BSY_ALT_GP0 pin is configured for any function other than a general-purpose input, it functions as a digital output. If another device attempts to drive the BSY_ALT_GP0 pin while it is configured as a digital output, contention will occur and could potentially damage the AD4695/AD4696. The BSY_ALT_GP0 pin is configured as a digital input by default.

### GPIO

The BSY_ALT_GP0 pin can be configured as a GPIO using the GPIn_EN and GPOn_EN bits in the GPIO_CTRL register (see Table 50). BSY_ALT_GP0 is configured as an input when its corresponding GPIn_EN bit is set to 1 and is configured as an output when its corresponding GPOn_EN bit is set to 1.

The GPIO functionality on BSY_ALT_GP0 allows the digital host to monitor logic outputs or control logic inputs of other devices in the system through the AD4695/AD4696 SPI, rather than using additional digital host GPIO pins. The AD4695/AD4696 GPIO functionality is especially useful in digitally isolated applications, since they reduce the number of required digital isolation channels.

When BSY_ALT_GP0 is configured as a general-purpose input, BSY_ALT_GP0 can be connected to a logic output of another device in the system, and the digital host can read the GPIO_STATE register to monitor its state. The GPI_READ bit in the GPIO_STATE register indicates the state of the BSY_ALT_GP0 (see Table 52). The logic input thresholds for the BSY_ALT_GP0 pin are specified in Table 1 as $V_{IL}$ and $V_{IH}$.

When BSY_ALT_GP0 is configured as a general-purpose output, BSY_ALT_GP0 can be connected to a logic input of another device in the system, such as other multiplexers or programmable gain amplifiers, and the digital host can write to the GPIO_STATE register to set the state of this signal. The GPO_WRITE bit in the GPIO_STATE register controls the state of the BSY_ALT_GP0 when configured as a general-purpose output (see Table 52). The logic output thresholds for the BSY_ALT_GP0 pin are specified in Table 1 as $V_{OL}$ and $V_{OH}$.

## DEVICE RESET

A device reset reinitializes the AD4695/AD4696 configuration registers. The AD4695/AD4696 provide several options for performing a device reset, including a hardware reset, a software reset, and power-on resets.

Hardware resets, software resets, and PORs all assert the RST_FLAG bit in the STATUS register. The RST_FLAG bit is a read-to-clear bit and is automatically set to 0 after successfully reading from the STATUS register. RST_FLAG can be used by the digital host to confirm the AD4695/AD4696 has successfully executed a device reset, or if a reset was performed unintentionally.

All device reset methods require a delay between the start of the reset instruction and when the AD4695/AD4696 SPI is ready to receive communications from the digital host. The device reset delays are shown in Figure 99 through Figure 106 and in Table 2. When the digital host attempts to perform an SPI read or write transaction before the AD4695/AD4696 are ready, the transaction is considered invalid and the NOT_RDY_ERROR bit in the SPI_STATUS is set to 1. The NOT_RDY_ERROR bit is a write-1-to-clear bit and is only reset by setting it to 1 with a valid register write transaction.

### Hardware Reset

A hardware reset is initiated by $\overline{RESET}$ falling edge. Figure 99 shows a timing diagram for performing a hardware reset. $t_{RESETL}$ is the minimum amount of time that $\overline{RESET}$ must be driven low, and $t_{HWR\_DELAY}$ is the time that the digital host must wait between a $\overline{RESET}$ falling edge and starting an SPI frame (see Table 2).

If the internal LDO is supplying VDD, and the internal LDO is disabled before a hardware reset, then the internal LDO will be enabled by the hardware reset, and an additional delay is required to account for the internal LDO output reaching the VDD minimum required voltage (see the Power-On Resets section).

### Software Reset

A software reset is initiated by setting the SW_RST_MSB and SW_RST_LSB bits in the SPI_CONFIG_A register both to 1. A software reset reinitializes the state of all configuration registers listed in the Register Information section to their default values, except for the SPI_CONFIG_A register. After the software reset is complete, the SW_RST_MSB and SW_RST_LSB bits are automatically cleared. Figure 100 shows the timing requirements for performing a software reset. $t_{SWR\_DELAY}$ is the time that the digital host must wait between the software reset and starting a new SPI frame (see Table 2).

### Power-On Resets

A power-on reset (POR) is initiated when VDD or VIO is first supplied. When a POR event is detected, the AD4695/AD4696 configuration registers are initialized to their default values, but it is still recommended to perform either a hardware reset or a software reset after a POR.

Figure 101 shows a timing diagram of a VDD POR where VIO is already supplied. $t_{POR\_VDD}$ is the time that the digital host must wait between VDD first being supplied and starting an SPI frame (see Table 2). Figure 102 shows a timing diagram of a VIO POR where VDD is already supplied. $t_{POR\_VIO1}$ is the time that the digital host must wait between VIO first being supplied and starting an SPI frame (see Table 2).

When VDD is supplied by the internal LDO, then the VDD POR is triggered when the internal LDO output has driven VDD to at least the minimum VDD specification. The internal LDO output is only enabled when both LDO_IN and VIO are supplied, and when the LDO_EN bit in the SETUP register is set to 1 (see the Internal LDO section).

Figure 103 shows a timing diagram of an LDO_IN POR where VIO is already supplied. $t_{POR\_LDO}$ is the time that the digital host must wait between LDO_IN first being supplied and starting an SPI frame.

Figure 104 shows a timing diagram of a VIO POR where the internal LDO is used to supply VDD. $t_{POR\_VIO2}$ is the time that the digital host must wait between VIO being supplied and starting an SPI frame.

When the internal LDO is supplying VDD, a POR also occurs when the internal LDO is enabled by the LDO wakeup command or by a hardware reset, if the internal LDO was previously disabled (by setting the LDO_EN bit to 0). Figure 105 shows a timing diagram of a POR when the internal LDO is enabled by the LDO wakeup command. $t_{WAKEUP\_SW}$ is the time that the digital host must wait between the LDO wakeup command and starting a new SPI frame. Figure 106 shows a timing diagram of a POR when the internal LDO is enabled by a hardware reset. $t_{WAKEUP\_HW}$ is the time that the digital host must wait between the hardware reset and starting an SPI frame.

$t_{POR\_LDO}$, $t_{POR\_VIO2}$, $t_{WAKEUP\_HW}$, and $t_{WAKEUP\_SW}$ all depend on the VDD decoupling capacitance ($C_{VDD}$). Larger values of $C_{VDD}$ increase the amount of time it takes for the internal LDO output voltage to reach the minimum VDD supply voltage to trigger a VDD POR. Table 2 provides typical values for these reset delay specifications with $C_{VDD} = 1$ μF.

**Table 20. General-Purpose Pin Functions and Function Priority**

| Pin | Function Priority | | | |
|---|---|---|---|---|
| | 1 (Highest Priority) | 2 | 3 | 4 (Lowest Priority) |
| BSY_ALT_GP0 | SDO1 (Dual-SDO Mode) | Alert Indicator | Busy Indicator | GPIO |
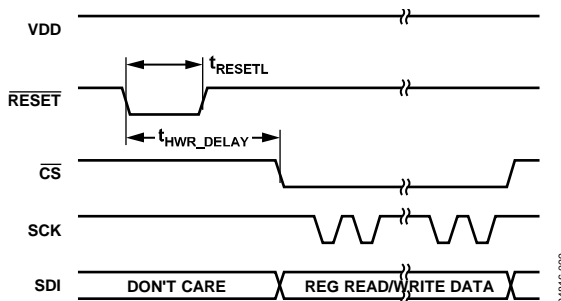

Figure 99. Hardware Reset Timing Diagram


Figure 100. Software Reset Timing Diagram
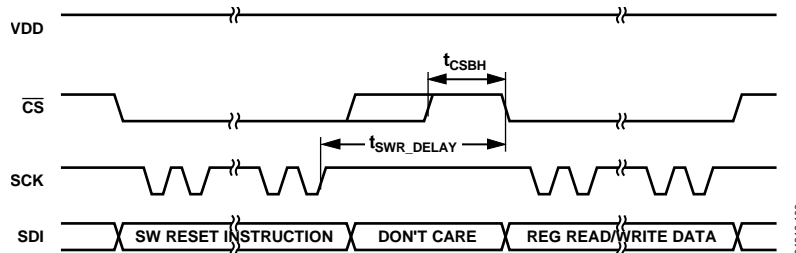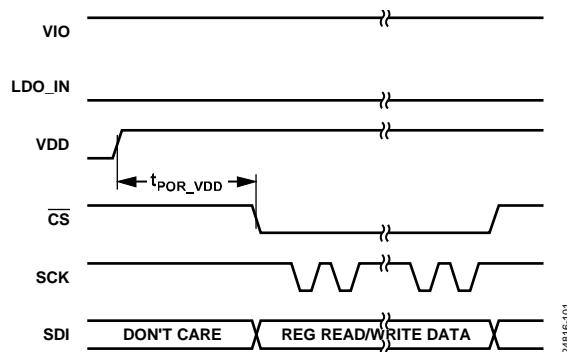
*Figure 101. VDD POR Timing Diagram*
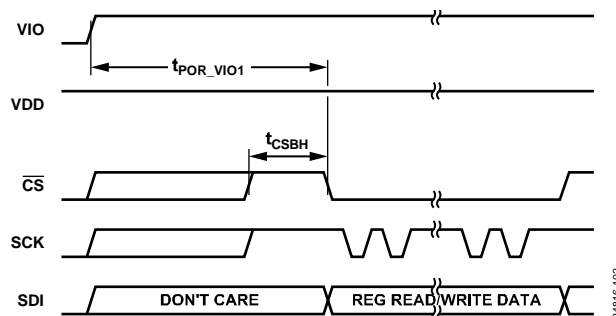


*Figure 102. VIO POR Timing Diagram (VDD Supplied Externally)*
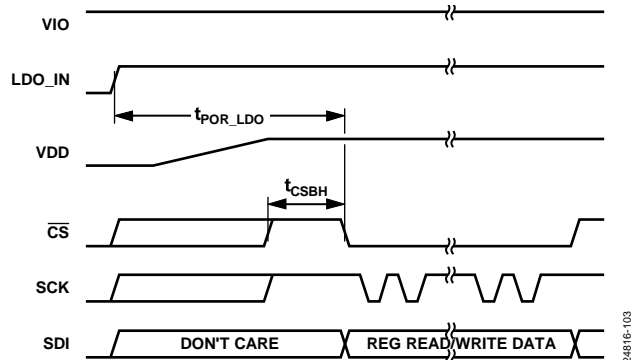


*Figure 103. LDO_IN POR Timing Diagram (Internal LDO Supplying VDD)*
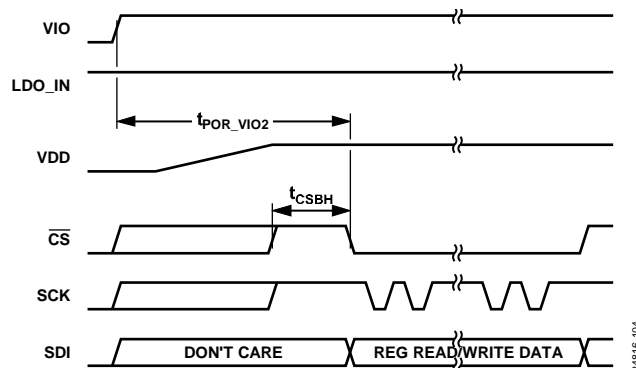


*Figure 104. VIO POR Timing Diagram (Internal LDO Supplying VDD)*
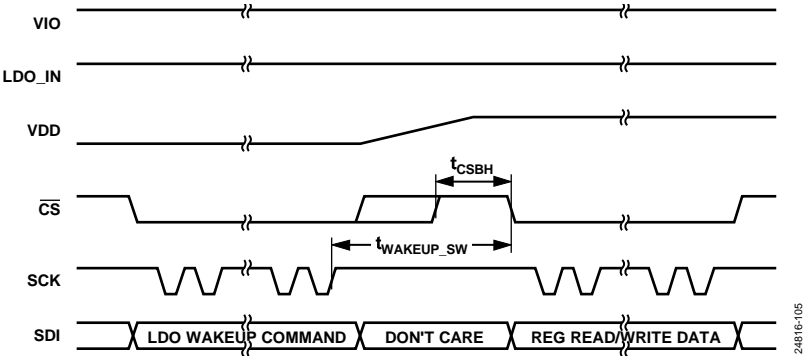
*Figure 105. LDO Wakeup Command POR Timing Diagram*
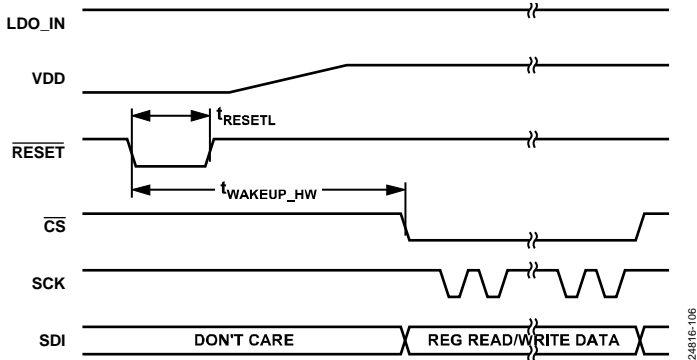


*Figure 106. POR with Internal LDO Enabled by Hardware Reset Timing Diagram*

# APPLICATIONS INFORMATION

Figure 107 shows an example of the recommended connection diagram for the AD4695/AD4696 companion circuitry.

The AD4695/AD4696 companion circuitry includes power supplies, voltage reference circuitry, analog front-end signal conditioning, and a SPI-compatible digital controller (plus

optional digital isolation). The following sections provide recommendations and suggestions for selecting and connecting the AD4695/AD4696 companion circuitry based on common application requirements.
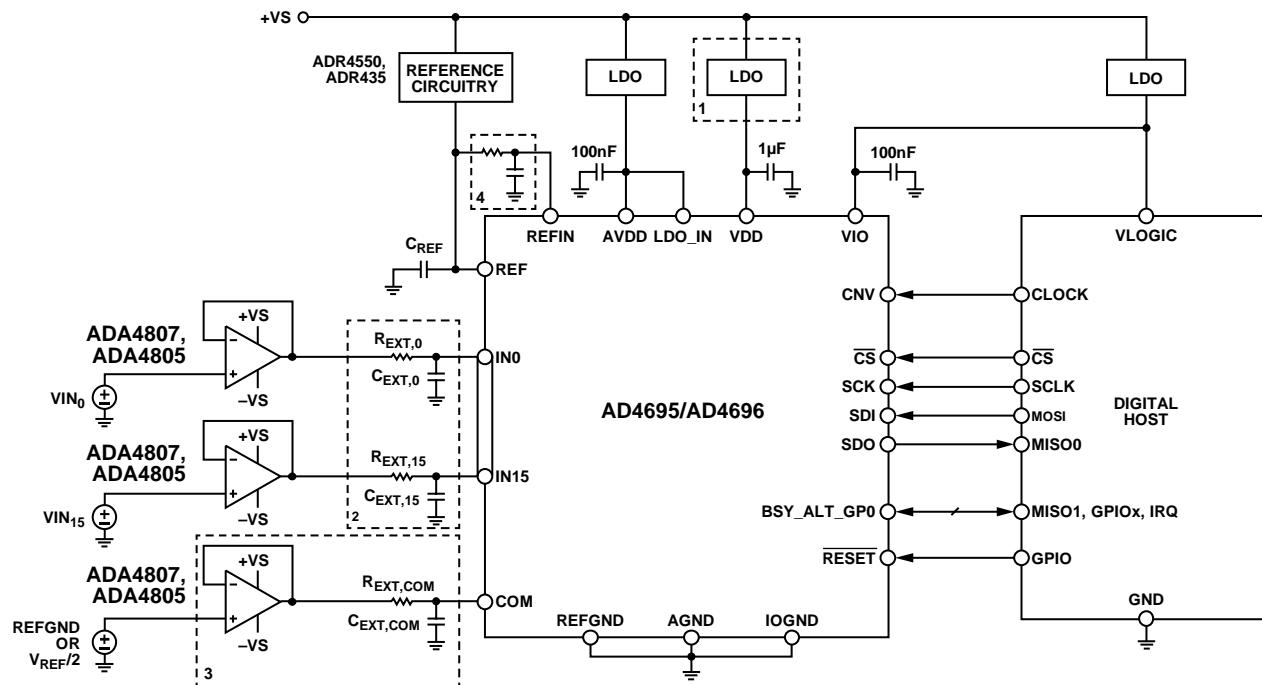


1. DEDICATED +1.8V SUPPLY FOR VDD NOT REQUIRED WHEN USING INTERNAL LDO
2. $R_{EXT,X}$ AND $C_{EXT,X}$ REPRESENT EXTERNAL INPUT RC FILTERS FOR ANALOG INPUT INx OR COM
3. DEDICATED ADC DRIVER FOR COM INPUT NOT REQUIRED IF CONNECTING COM TO REFGND
4. OPTIONAL RC FILTER BETWEEN VOLTAGE REFERENCE AND REFIN INPUT (REFERENCE BUFFER ENABLED ONLY)

*Figure 107. AD4695/AD4696 Typical Connection Diagram*

Rev. PrI | Page 59 of 94

## ANALOG FRONT-END DESIGN

The analog front-end companion circuitry for the AD4695/AD4696 normally includes an external RC filter and an ADC driver or precision operational amplifier between the signal being measured and the AD4695/AD4696 analog inputs.

The component selection and design of the analog front-end circuitry driving the AD4695/AD4696 analog inputs have a direct impact on overall system performance. The analog front-end must be designed with the system target noise, accuracy, distortion and settling requirements of the end application. The following sections provide recommendations for designing analog front-end and signal conditioning circuits based on these requirements.

### External RC Filter

The external RC low-pass filter consists of an external resistor and capacitor (represented by $R_{EXT}$ and $C_{EXT}$ in Figure 67 and Figure 107). These components act to reduce the wideband noise from the analog front-end circuitry, reduce the nonlinear voltage kickback that occur at the analog inputs, and protect the analog inputs from overvoltage events. Selecting the appropriate values of $R_{EXT}$ and $C_{EXT}$ for these functions is described in the Analog Front-End Noise Considerations section, the Signal Settling section, and the Analog Input Overvoltage Protection section.

The $C_{EXT}$ capacitors should be an NP0 ceramic capacitor to limit distortion artifacts, and the printed circuit board layout should minimize the parasitic impedance between $C_{EXT}$ and the analog input pin. See the Layout Guidelines section for more information.

### Signal Settling Requirements

As described in the Converter Operation and Analog Inputs sections, the AD4695/AD4696 analog inputs (IN0 thru IN15 and COM) are routed to the ADC core inputs via the internal analog multiplexer.

As shown in Figure 67, the ADC core capacitive DAC can be represented by a switched capacitive load.

At the start of the conversion phase, the multiplexer switches are disconnected and the voltage on the currently selected analog input is sampled on the capacitive DAC. When the conversion phase is complete and the acquisition phase begins, the multiplexer connects the capacitive DAC to the next channel in the sequence. The voltage on the capacitive DAC at the start of the acquisition phase is therefore often different from the voltage on the selected analog input pin. The result is that at the start of the acquisition phase of the ADC, the selected analog input exhibits a voltage glitch, referred to as kickback. The voltage kickback must be adequately settled to ensure accurate conversion results. Signal settling requirements therefore dictate the minimum allowable analog front-end bandwidth and constrain the driver amplifier and external RC filter selection.

Figure 108 shows a representation of the signal settling dynamics of the AD4695/AD4696 analog inputs (drawn as a single-ended input for simplicity).
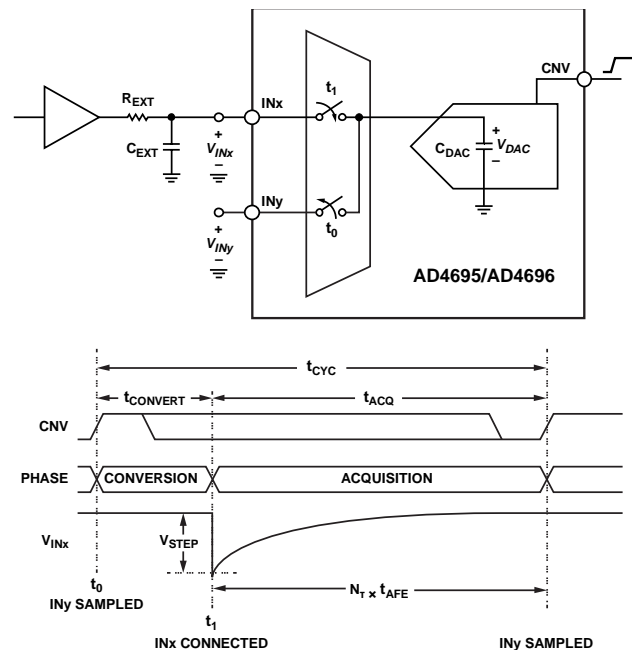


*Figure 108. AD4695/AD4696 Signal Settling Model*

The minimum analog front-end bandwidth ($BW_{AFE}$) required for optimal settling depends on the available acquisition time ($t_{ACQ}$), the magnitude of the voltage kickback ($V_{STEP}$), and the desired settling accuracy in LSBs. The minimum $BW_{AFE}$ can be estimated by approximating the settling behavior of the analog front-end as a single-pole RC filter with time constant $\tau_{AFE}$.

The number of time constants ($N_\tau$) required to settle $V_{STEP}$ to N-bit accuracy is:

$$N_\tau = ln\left(\frac{V_{STEP}}{V_{REF} / 2^N}\right)$$

$\tau_{AFE}$ and $BW_{AFE}$ can be calculated from $N_\tau$ versus $t_{ACQ}$:

$$\tau_{AFE} = t_{ACQ}/N_\tau$$

and

$$BW_{AFE} = 1/(2\pi \times \tau_{AFE}) = 1/(2\pi(t_{ACQ}/N_\tau))$$

$t_{ACQ}$ is equal to the conversion period ($t_{CYC}$) minus the conversion time ($t_{CONVERT}$) when the standard sequencer, advanced sequencer, or two-cycle command mode are enabled, as shown in Figure 91 through Figure 98When single-cycle command mode is enabled, $t_{ACQ}$ is also a function of the SCK period, as shown in Figure 95 and the Single-Cycle Command Mode section. $t_{ACQ}$ is therefore shorter for faster sample rates and when single-cycle command mode is enabled. Table 2 gives the minimum acquisition time for the maximum sample rate of the AD4695/AD4696.

Larger magnitudes of $V_{STEP}$ also increase the minimum required $BW_{AFE}$. The magnitude of $V_{STEP}$ depends on the difference between the voltage on the capacitive DAC ($V_{DAC}$) and the

voltage on the selected analog input ($V_{IN}$), the capacitance of $C_{EXT}$ in the external RC filter, and whether analog input high-Z mode is enabled or disabled (see Figure 108):

$$V_{STEP} = (V_{DAC} - V_{IN}) \times C_{IN}/(C_{IN} + C_{EXT})$$

where $C_{IN}$ is the equivalent capacitance of the AD4695/AD4696 ADC core capacitive DAC, which is roughly 60 pF (see Figure 67).

The magnitude of $V_{STEP}$ is therefore smaller when $V_{DAC}$ is closer to $V_{IN}$ and for larger $C_{EXT}$. The polarity of the voltage kickback is positive when $V_{DAC}$ is larger than $V_{IN}$ and negative when $V_{DAC}$ is smaller than $V_{IN}$.

The voltage on $V_{DAC}$ at the start of the acquisition phase depends on whether analog input high-Z mode is enabled or disabled. When analog input high-Z mode is disabled, $V_{DAC}$ is roughly equal to the voltage sampled from the previous channel, and the worst-case $V_{STEP}$ occurs when:

$$|V_{DAC} - V_{IN}| = |V_{INx} - V_{INy}| = V_{REF}$$

where $V_{INx}$ is the voltage sampled during the conversion phase of the previous channel in the sequence and $V_{INy}$ is the voltage on the current channel.

When analog input high-Z mode is enabled, $V_{STEP}$ is significantly reduced, thereby reducing the bandwidth required to settle the voltage kickback to the desired resolution (see Figure 14). The reduced $V_{STEP}$ also reduces, consequently allowing lower bandwidth amplifiers and external RC filters.

With analog input high-Z mode enabled, the AD4695/AD4696 can be driven by a larger variety of precision amplifiers with lower small signal bandwidth specifications than traditional multiplexed SAR ADCs. In addition, external RC filters with lower cutoff frequencies can be used to further reduce analog front-end noise and tolerate higher noise amplifiers (see the Analog Front-End Noise Considerations), and smaller $C_{EXT}$ capacitance can be used, reducing stability challenges for the amplifier. It is recommended to enable analog input high-Z mode for channels with an effective bandwidth less than TBD MHz.

Figure 70 and Figure 71 in the Analog Input High-Z Mode show ac performance achieved with various analog front-end components (amplifiers and external RC filter components) with analog input high-Z mode enabled and disabled.

### Analog Front-End Noise Considerations

The magnitude of the analog front-end noise directly impacts the dynamic range and SNR performance of the overall AD4695/AD4696 signal chain. The analog front-end components and configuration should be selected to achieve the target noise specification for the overall system.

Figure 109 illustrates the primary noise sources in a typical analog front-end driver circuit.
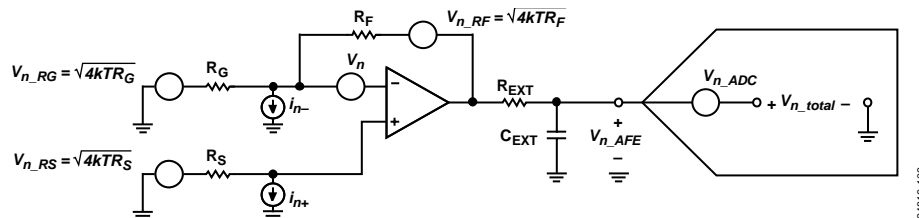


Figure 109. Noise Sources in Typical ADC Analog Front-End Circuit

Assuming all noise sources are Gaussian and uncorrelated, the total system rms noise ($v_{n\_total}$) is the root-sum-square (rss) of the referred-to-output (RTO) rms noise of the analog front-end ($v_{n\_AFE}$) and the AD4695/AD4696 input-referred rms noise ($v_{n\_ADC}$):

$$v_{n\_total} = \sqrt{V_{n\_AFE}^2 + V_{n\_ADC}^2}$$

The estimated system dynamic range ($DR_{total}$) is a measure of the system rms noise and the full-scale input range.

$$DR_{total} = 20\log\left(\frac{V_{REF}/\left(2\sqrt{2}\right)}{v_{n\_total}}\right)$$

The AD4695/AD4696 input-referred rms noise specification ($v_{n\_ADC}$) is typically 37.4 µVrms (see Table 1). Figure 110 shows the typical system dynamic range versus $v_{n\_AFE}$ with $v_{n\_ADC}$ = 37.4 µV rms and $V_{REF}$ = 5 V. For $v_{n\_AFE}$ less than TBD µV rms, the overall system dynamic range remains within 0.5 dB of the AD4695/AD4696 dynamic range specification (see Table 1).



*Figure 110. $DR_{total}$ vs. $v_{n\_AFE}$ with AD4695/AD4696*

The analog front-end RTO noise ($V_{n\_AFE}$) is equal to the rms noise of each of the constituent components in the analog front-end, referred to the output of the external RC filter ($R_{EXT}$ and $C_{EXT}$ in Figure 107 and in the External RC Filter section). Assuming the RC filter bandwidth is much lower than the bandwidth of the amplifier circuit, then $v_{n\_AFE}$ is equal to the noise spectral density of each of these components (referred to the amplifier output) multiplied by the effective noise bandwidth of the RC filter ($ENBW_{RC}$), where:

$$ENBW_{RC} = \sqrt{\frac{\pi}{2} \times \frac{1}{2\pi R_{EXT} C_{EXT}}}$$

and

$$v_{n\_AFE} = ENBW_{RC} \times$$

$$\sqrt{4kTR_F + \left(1 + \frac{R_F}{R_G}\right)^2 \left[4kTR_S + i_{n+}{}^2 R_S{}^2 + v_n{}^2\right] + \left(\frac{R_F}{R_G}\right)^2 4kTR_G + i_{n-}{}^2 R_F{}^2}$$

where:
$k$ is the Boltzmann constant.
$T$ is the absolute temperature in Kelvin.
$R_F$ and $R_G$ are the feedback network resistors, as shown in Figure 109.
$R_S$ is the source resistance, as shown in Figure 109.
$i_{N+}$ and $i_{N-}$ represent the amplifier input current noise spectral density in pA/√Hz.
$v_n$ represents the amplifier input voltage noise spectral density in nV/√Hz.

See MT-049 and MT-050 for detailed derivations of $v_{n\_AFE}$ vs. analog front-end components and configurations.

### Analog Front-End Noise in Pseudo-Bipolar Mode

When configuring a channel in pseudo-bipolar mode, typically a second analog front-end circuit is required to drive the negative-side input to $V_{REF}$/2 V (as described in the Channel Configuration Options section). In this case, the RTO rms noise of the additional analog front-end ($v_{n\_AFE2}$) is added to the rss equation to calculate the total system rms noise:

$$v_{n\_total} = \sqrt{V_{n\_AFE}^2 + V_{n\_AFE2}^2 + V_{n\_ADC}^2}$$

Note that the bandwidth of the RC filter and values of $R_{EXT}$ and $C_{EXT}$ cannot be set arbitrarily low due to the settling requirements of the AD4695/AD4696 analog inputs. Refer to the Signal Settling Requirements section for guidelines on selecting the optimal RC filter components for the target sample rate.

### Guidelines for Driver Amplifier Selection

The following is a list of guidelines for selecting the amplifier(s) used in the AD4695/AD4696 analog front-end based on the end-system requirements.

The amplifier voltage and current noise specifications should be sufficiently low to achieve the desired rms noise and dynamic range performance, as described in the Analog Front-End Noise Considerations section.

The distortion performance of the amplifier should be sufficient to achieve desired THD performance. To meet the AD4695/AD4696 THD data sheet specification, the amplifier circuit must have better or comparable distortion specifications.

The small signal bandwidth of the amplifier should be sufficiently higher than the minimum bandwidth required to adequately settle the voltage steps that occur when switching between two analog input channels, as described in the Signal Settling section.

The amplifier should also have sufficient supply headroom to adequately output a full-scale signal to the AD4695/AD4696 analog inputs (see the input voltage range specification in Table 1). Refer to the input and output headroom requirements in the amplifier data sheet to determine the supply voltages required to support the desired full-scale range for the given channel.

The ADA4805 and ADA4807 are suitable amplifiers for channels acquiring ac waveforms, due to their exceptionally low noise and distortion and high bandwidth.

The ADA4610 and ADA4077 are suitable amplifiers for channels monitoring dc or low frequency signals that require high precision. It is recommended to enable analog input high-Z mode on AD4695/AD4696 analog input channels when being driven directly by the ADA4610, ADA4077, or amplifiers with similar bandwidth specifications to ensure adequate settling performance (see the Signal Settling Requirements section and Analog Input High-Z Mode section).

## ANALOG INPUT OVERVOLTAGE PROTECTION

The external resistor in the external RC filter (represented by $R_{EXT}$ in Figure 67, Figure 107, and Figure 111) works with the input overvoltage protection clamps to provide overvoltage protection to the analog inputs (see the Input Overvoltage Protection Clamps section).

An overvoltage event is defined as an event where the overvoltage protection clamps are activated as a result of the input voltage on IN0 to IN15 or COM exceeding the clamp activation voltage specification ($V_{ACTa}$ in Figure 111). The maximum $V_{ACT}$ voltage specification is $V_{REF} + 0.55$ V (see Table 1).
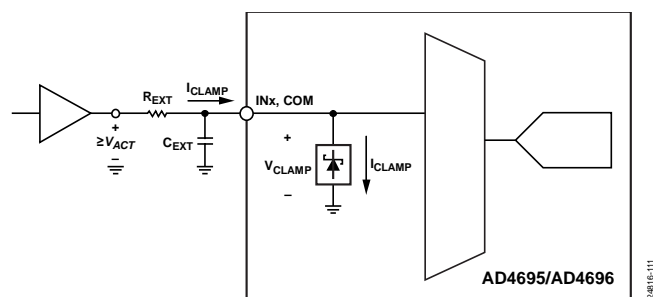


*Figure 111. Analog Input Overvoltage Event*

When activated, the clamp on the given channel sinks current from the source to ground (see $I_{CLAMP}$ in Figure 111), resulting in a voltage drop across $R_{EXT}$. The AD4695/AD4696 overvoltage protection clamps support a maximum sustained $I_{CLAMP}$ current of 5 mA (see Table 1). $R_{EXT}$ therefore isolates the analog input pin voltage from the applied voltage ($v_{IN}$). The maximum $v_{IN}$ voltage that can be supported for a given analog input is a function of $V_{REF}$ and $R_{EXT}$. The following relation can be used to determine the required value of $R_{EXT}$ to limit the clamp current to the maximum supported current (5 mA) given the reference voltage ($V_{REF}$) and maximum expected $v_{IN}$ voltage:

$$R_{EXT} + v_{IN,max} - V_{REF}/5 \text{ mA } (\Omega)$$

For example, if the analog input source can swing to +7.5 V and $V_{REF} = 5$ V, then $R_{EXT}$ must be approximately 500 Ω to limit the clamp current to 5 mA. If this resistor is being sized based upon the clamping current limits, $C_{EXT}$ must be carefully chosen to ensure adequate input bandwidth is achieved (see the Analog Front-End Noise Considerations and Signal Settling sections for more information).

The value of $R_{EXT}$ also must be selected to ensure stability of the overvoltage protection clamp circuit, if desired. See the Overvoltage Protection Clamp Stability section for more information.

## REFERENCE CIRCUITRY DESIGN

The AD4695/AD4696 reference voltage ($V_{REF}$) sets the full-scale range of the ADC core and determines the resulting output code for a given analog input voltage (see the Transfer Function section). The $V_{REF}$ voltage therefore has a direct impact on the overall system accuracy and ac performance. The reference companion circuitry for the AD4695/AD4696 must have adequate noise performance, accuracy, drift and signal settling characteristics for the end application.

The REF input is a dynamic current load which pulls charge from the reference circuitry during the conversion phase of the ADC core, and the reference circuit must be able to maintain a stable $V_{REF}$ while the ADC is performing conversions to maintain performance (i.e. gain error).

The AD4695/AD4696 reference input high-Z mode significantly reduces the magnitude of the average current of the REF input when enabled. The reference input high-Z mode significantly reduces the drive requirements of the reference circuitry, allowing system designers to better prioritize dc accuracy, power, and system footprint targets.

Figure 112 shoes the typical connection diagram for the AD4695/AD4696 companion reference circuit. The reference circuitry consists of a voltage reference, reference decoupling capacitor ($C_{REF}$), and any accompanying reference buffer or analog low-pass filtering. A reference buffer is required if the selected voltage reference does not have an adequate load regulation to drive the REF input at the desired ADC sample rate (see the Reference Circuit Design for Driving REF Input section).

$C_{REF}$ supplies the charge necessary for the ADC core to perform the bit trials as part of the conversion phase and filters noise from the other reference circuitry. $C_{REF}$ must be sufficiently large to prevent deviations in $V_{REF}$ during the ADC bit trials. When reference input high-Z mode is enabled, the amount of charge pulled by the REF input is significantly reduced, thereby reducing the minimum $C_{REF}$ capacitance. When reference input high-Z mode is disabled, a 10 µF $C_{REF}$ is recommended. When reference input high-Z mode is enabled, a 1 µF $C_{REF}$ is recommended.

The PCB layout of the reference circuitry relative to the AD4695/AD4696 REF input is critical to ensuring optimal performance. The Layout Guidelines section provides recommendations and guidelines for the layout of the reference circuit components.

### Reference Circuit Design for Driving REF Input

Figure 112 shows a typical connection diagram for the reference circuitry driving the REF input of the AD4695/AD4696.



<sup>1</sup>WHEN DRIVING REF DIRECTLY ON WLCSP OPTION, SHORT REFIN TO REF AND ENSURE THE INTERNAL REFERENCE BUFFER AND REFERENCE BUFFER BYPASS OPTION ARE DISABLED. (REFBUF_EN = REFBUF_BP = 0)
<sup>2</sup>ADDITIONAL LOW-PASS FILTERING MUST NOT BE IMPLEMENTED WITHOUT A REFERENCE BUFFER.
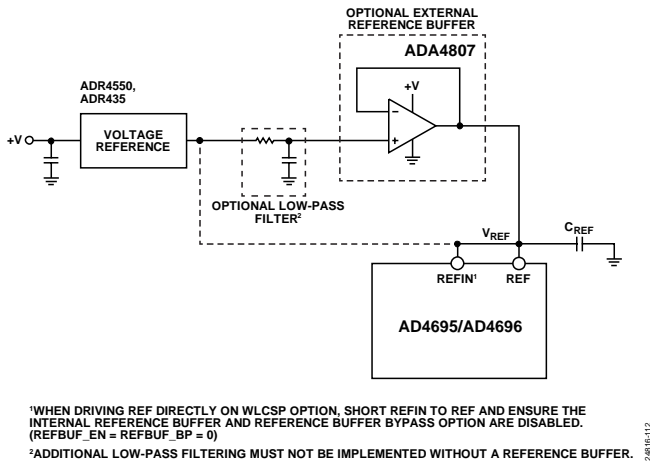
*Figure 112. Typical Connection Diagram for Driving REF Input*

The device driving the REF input must have sufficiently low output impedance so the reference input current does not cause $V_{REF}$ to deviate enough to violate the system performance targets. To achieve data sheet performance, $V_{REF}$ must remain within

half an LSB. The maximum output impedance of the device driving the REF input ($R_{o\_max}$) is therefore:

$$R_{o\_max} = \frac{V_{REF}/2^{(16+1)}}{I_{REF}}$$

where $I_{REF}$ is the average REF input current.

Most voltage references specify load regulation in ppm/mA, which can be converted to effective output impedance with the following:

$$L_{max} = 1000 \times \frac{R_{o\_max}}{V_{REF}}$$

where $L_{max}$ is the load regulation specification for the voltage reference in ppm/mA that corresponds to the calculated $R_{o\_max}$.

$I_{REF}$ is typically 6 µA at 1 MSPS with reference input high-Z mode enabled and 315 µA at 1 MSPS with reference input high-Z mode disabled. $I_{REF}$ scales linearly with the ADC sample rate (see Table 1 and Figure 39). The output impedance and load regulation requirements of the reference circuitry are therefore less strict at lower sample rates. Table 21 shows calculated $R_{o\_max}$ and $L_{max}$ for $V_{REF}$ = 5 V and for different sample rates and with reference input high-Z mode disabled and enabled. Table 21 also provides recommendations for voltage references and discrete reference buffers for each of these conditions.

**Table 21. Reference Circuitry Recommendations, REF Input**

| Sample Rate | Reference Input High-Z Mode | $I_{REF}$ (µA) | $R_{o\_max}$ (Ω) | $L_{max}$ (ppm/mA) | Recommended Voltage References and Reference Buffers |
|---|---|---|---|---|---|
| 1 MSPS | Disabled | 315 | 0.121 | 24.22 | ADR4550 with ADA4807, ADR445 with ADA4807, ADR435 |
| 1 MSPS | Enabled | 6 | 6.358 | 1271 | ADR4550, ADR445, ADR435 |
| 500 kSPS | Disabled | 157.5 | 0.242 | 48.44 | ADR445 with ADA4807, ADR4550, ADR435 |
| 500 kSPS | Enabled | 3 | 12.72 | 2542 | ADR4550, ADR445, ADR435 |

## CONVERTING BETWEEN CODES AND VOLTS

The Transfer Function section describes the ideal transfer function between the analog input voltage sampled by the AD4695/AD4696 ADC core and the resulting output code. The analog input voltage ($V_{INx}$) corresponding to each possible output code value ($CODE_{OUT}$) is a function of the $V_{REF}$ voltage and the oversampling ratio setting and polarity mode for the selected channel:

$$V_{INx} = LSB \times CODE_{OUT} = \frac{V_{REF}}{2^N} \times CODE_{OUT}$$

where LSB is the LSB size and N is the resolution of the output code. The AD4695/AD4696 ADC core outputs 16-bit results (N = 16), but the output code resolution is a function of the oversampling ratio selected for the given channel (DR):

$$N = 16 \log_4(DR)$$

DR can be set to 1, 4, 16, or 64, which correspond to an output code resolution of 16, 17, 18, and 19, respectively. Table 6 thru Table 9 show the negative and positive full-scale output code values for each oversampling ratio. See the Oversampling and Decimation section for details on configuring the oversampling ratio for each channel.

The polarity mode for the selected channel determines whether $CODE_{OUT}$ uses straight binary or twos complement format. When unipolar mode is selected, $CODE_{OUT}$ is in straight binary, and is therefore an unsigned integer value. When pseudo-bipolar mode is selected, $CODE_{OUT}$ uses twos complement encoding, and is therefore a signed integer value. See the Channel Configuration Options for details on configuring the polarity mode for each channel.

The offset and gain correction settings for each channel modify the transfer function of the AD4695/AD4696 to correct for first-order inaccuracies in the system that cause the observed transfer function to deviate from the ideal. The OFFSET and GAIN fields for each channel should be updated during system calibration. The Offset and Gain Correction section describes how the OFFSET and GAIN fields modify

## OVERSAMPLING FOR NOISE REDUCTION

The AD4695/AD4696 includes on-chip oversampling and decimation as a means to reduce the total effective Gaussian noise of the system in the digital domain (see the Oversampling and Decimation section). Assuming the analog front-end noise is Gaussian, the effective system noise after oversampling ($v_{n\_OSR}$) is:

$$v_{n\_OSR} = \frac{v_{n\_total}}{\sqrt{D}}$$

where D is the oversampling ratio (decimation rate) setting for the given analog input channel and $v_{n\_total}$ is the RTO system noise (defined in the Analog Front-End Noise Considerations section). When the oversampling ratio is set to 1, no oversampling occurs, and the effective noise remains $v_{n\_total}$.

When oversampling ratios of 4, 16, and 64 are used, the noise is attenuated by a factor of 2, 4, and 8 respectively.

The resulting dynamic range when utilizing oversampling ($DR_{OSR}$) is:

$$DR_{OSR} = DR_{total} + 10 \log(D)$$

where $DR_{total}$ is the system dynamic range for an oversampling ratio of 1 (defined in the Analog Front-End Noise Considerations section).

The effective number of bits (ENOB) of the system increases by 1 every time the noise is halved. As a result, ENOB increases by 1 bit every time the oversampling ratio is increased by a factor of 4. To reflect this, when an AD4695/AD4696 channel is configured with oversampling ratios of 4, 16, or 64, the resolution of the conversion results for that channel is extended to 17-bits, 18-bits, and 19-bits, respectively (see the Transfer Function section and Serial Data Output Modes section).

Note that oversampling and decimation only reduces voltage noise for uniformly distributed Gaussian noise sources and has no effect on other types of noise sources (such as 1/f noise).

## DIGITAL INTERFACE OPERATION

Figure 107 shows a typical connection diagram of the AD4695/AD4696 digital interface connected to a digital host. The AD4695/AD4696 can be operated by a single 4-wire SPI-compatible host, but some features require additional digital resources, such as GPIOs, timers, etc.

The following sections provide recommendations for digital interface connections and operation to interact with the AD4695/AD4696 interface and feature set.

### ADC Convert-Start Signal Options

The CNV input is analogous to an edge-triggered interrupt pin which instructs the AD4695/AD4696 ADC core to perform a conversion (see the Converter Operation section). The CNV input is active only when in Conversion Mode and is ignored when in Register Configuration Mode. The period of the signal driving CNV sets the sample rate of the AD4695/AD4696, and must conform to the $t_{CYC}$ specification in Table 2 and in the timing diagram figures in the Conversion Mode Timing Diagrams section.

The ADC core samples the analog input voltage on the selected channel on the rising edge of CNV. The signal driving the CNV input therefore must have sufficiently low jitter and fast edge rates to achieve the desired noise performance at the target input frequencies. The layout of the trace connecting the AD4695/AD4696 CNV input to the digital host should be as short as possible with minimal vias to minimize trace impedance (see the Layout Guidelines section).

In conversion mode, the digital host SPI master peripheral must be synchronous to the CNV signal and follow the timing requirements specified in the Conversion Mode Timing Diagrams section. See the SPI Peripheral Synchronization in

Conversion Mode section for connection diagram and firmware architecture for maintaining proper SPI timing.

Embedded clock divider or timer peripherals can typically output an integer division of the system clock. When utilizing embedded clock divider peripherals, connect the digital host clock output to CNV and set the clock output frequency to the desired sample rate. The clock output must be enabled while the AD4695/AD4696 are in conversion mode, but it can be either enabled or disabled while in Register Configuration Mode.

The CNV input can be connected to the $\overline{CS}$ output of the SPI master peripheral, provided the $\overline{CS}$ rising edge timing is deterministic and periodic (see Figure 115). Note that for oversampling ratio settings greater than 1, multiple CNV rising edges are required before the result is available to be read out on the SPI (see the Oversampling and Decimation section). The SPI will output all 0s during the CNV/$\overline{CS}$ frames prior to the data being ready.

An external crystal oscillator with a CMOS clock driver can also drive the CNV input. With this option, either the oscillator output or the busy indicator from the AD4695/AD4696 must be routed to the digital host and used as a timer or interrupt trigger to achieve synchronization between the CNV signal and the SPI master peripheral (see the SPI Peripheral Synchronization in Conversion Mode section).

Note that when autocycle mode is enabled, the CNV input is ignored, and conversions are instead triggered by an internal timer in the AD4695/AD4696, as described in the Autocycle Mode section. When exclusively using autocycle mode, the CNV input should be tied to IOGND. The busy indicator is required to synchronize the AD4695/AD4696 to the SPI master peripheral when using autocycle mode (see the SPI Peripheral Synchronization in Autocycle Mode section).

### SPI Peripheral Connections

The AD4695/AD4696 offer multiple serial data output modes which allow for one or two MISO lines to output conversion results (see the Serial Data Output Modes section). When single-SDO mode is selected, only the SDO pin functions as a serial data output. When dual-SDO mode is selected, both SDO and BSY_ALT_GP0 function as serial data outputs.
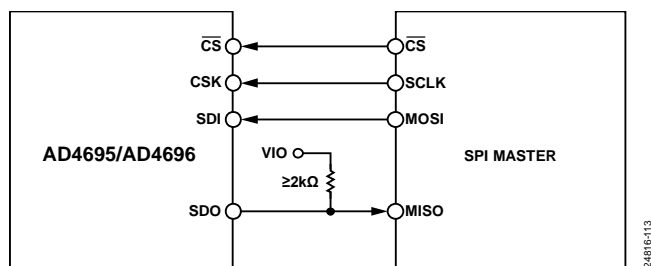


*Figure 113. AD4695/AD4696 SPI Connection Diagram (Single-SDO Mode)*

Figure 113 shows a connection diagram for interfacing the AD4695/AD4696 SPI to the digital host SPI master peripheral when configured in single-SDO mode. It is recommended to include a pull-up resistor (2 kΩ minimum) to VIO on the SDO line, especially when the busy indicator is enabled on SDO (see the Busy Indicator on Serial Data Outputs section).
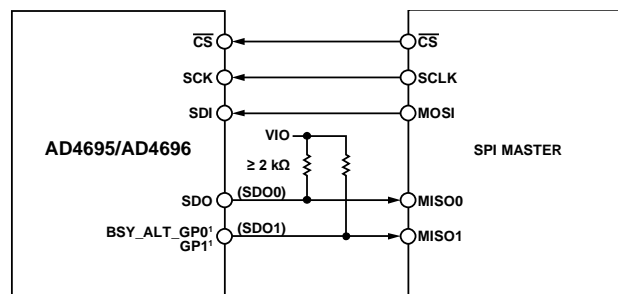


*Figure 114. AD4695/AD4696 SPI Connection Diagram (Dual-SDO Mode)*

Figure 114 shows a connection diagram for interfacing the AD4695/AD4696 SPI to a digital host SPI master peripheral when configured in dual-SDO mode. BSY_ALT_GP0 should be routed to the second MISO input on the digital host (MISO1). It is recommended to include pull-up resistors on both SDO0 and SDO1 lines, especially when the busy indicator is enabled on the serial data outputs (see the Busy Indicator on Serial Data Outputs section).

### SPI Peripheral Synchronization in Conversion Mode

The AD4695/AD4696 have a 4-wire SPI in SPI mode 3 for accessing register contents and ADC results. The digital host must at minimum include a 4-wire SPI-compatible peripheral to operate the AD4695/AD4696 (see the SPI Peripheral Connections section).

In conversion mode, the SPI transfers must begin after the conversion time ($t_{CONVERT}$) has elapsed and must complete within $t_{SCKCNV}$ before the next CNV rising edge (see Table 2 and in the timing diagrams in the Conversion Mode Timing Diagrams section). To ensure the conversion mode timing requirements are met, the digital host SPI master peripheral must either be synchronized to the clock source generating the CNV signal or to the busy indicator output from the AD4695/AD4696. The SCK frequency must also be sufficiently high to ensure all conversion mode results are clocked out before the start of the next conversion frame (see the Conversion Mode SPI Clock Frequency Requirements section).
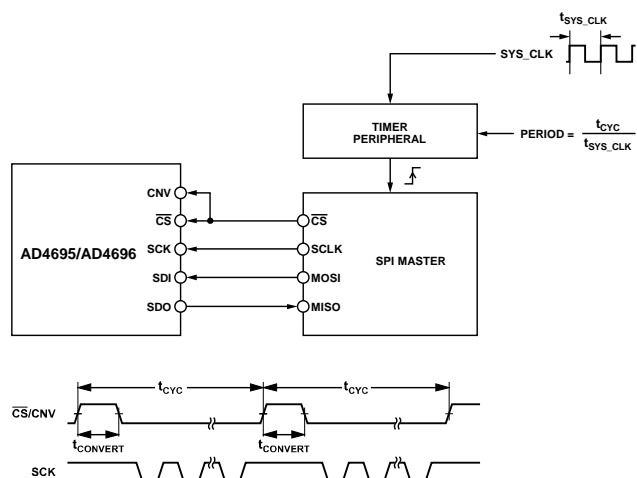
Figure 115. 4-Wire SPI Operation Diagram

Figure 115 shows a simplified connection diagram and software architecture for operating the AD4695/AD4696 with only a 4-wire SPI. The CNV input is driven by the $\overline{CS}$ output from the digital host SPI master peripheral. The configuration in Figure 115 requires the $\overline{CS}$ signal to be periodic with deterministic rising edge timing to achieve the necessary jitter for the application. The SPI frames should be synchronized to a timer peripheral, and the $\overline{CS}$ output should have a well-defined duty cycle. Figure 96 shows a SPI timing diagram using the configuration in Figure 115.



Figure 116. SPI Synchronization with Countdown Timer Peripheral

Figure 116 shows a simplified connection diagram and software architecture for using the digital host countdown timer peripheral to synchronize the SPI master peripheral to the CNV signal source. The countdown timer is configured to trigger on a CNV rising edge, wait for $t_{CONVERT}$ to elapse, and then trigger an interrupt service routine which calls the SPI master to perform a transfer. The countdown timer is programmed with an integer value (COUNT) which specifies the number of system clock (SYS_CLK) periods to wait before calling the SPI transfer

interrupt routine. It is recommended to implement a delay corresponding to the maximum $t_{CONVERT}$ specification given in Table 2. In practice, most digital hosts exhibit some latency between the interrupt service routine triggers and execution, which increases the delay between the CNV rising edge and the start of the SPI transfer. Refer to the digital host specifications to determine the optimal COUNT value for the given application.



Figure 117. SPI Synchronization with Busy Indicator

Figure 117 shows a simplified connection diagram and software architecture for utilizing the AD4695/AD4696 busy indicator to synchronize the SPI master peripheral to the ADC conversion timing. The busy indicator must be enabled on the BSY_ALT_GP0 pin as described in the Busy Indicator on section, and the digital host must have a digital input that can be configured as a trigger for interrupt service routines. Route the busy indicator to the interrupt input on the digital host and configure the interrupts to trigger on the busy indicator falling edge. Since the busy indicator falling edge is interpreted as the "data ready" signal, the digital host is not required to implement any further delays between the busy indicator falling edge and the start of the SPI frame.

The configuration in Figure 117 is optimal when utilizing oversampling, as the busy indicator does not go low until the oversampled result is ready, reducing the number of redundant SPI transfers that would otherwise occur without additional logic (see Figure 75).

*Figure 118. SPI Synchronization with Alert Indicator*
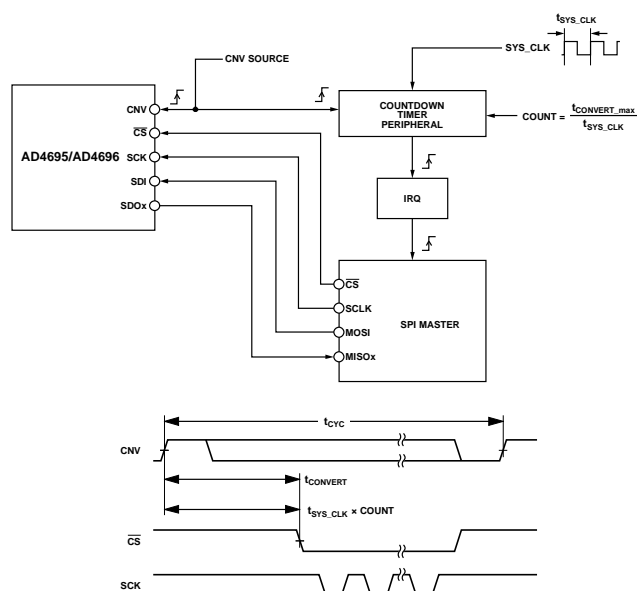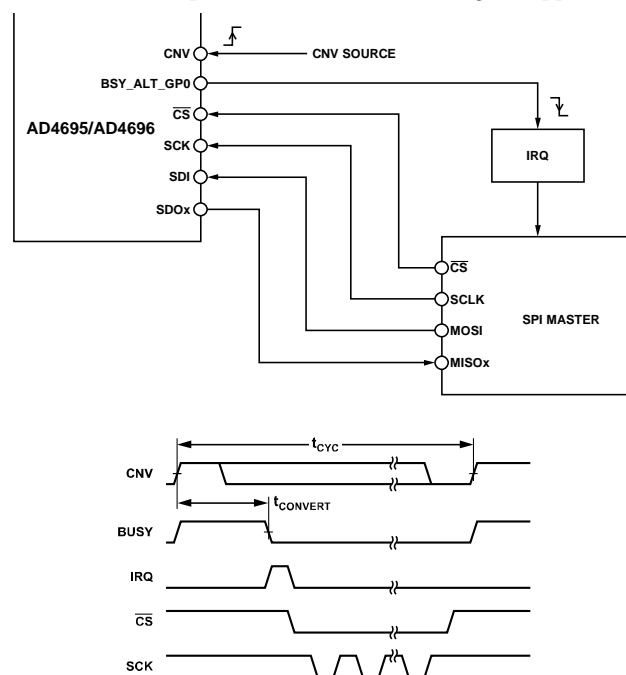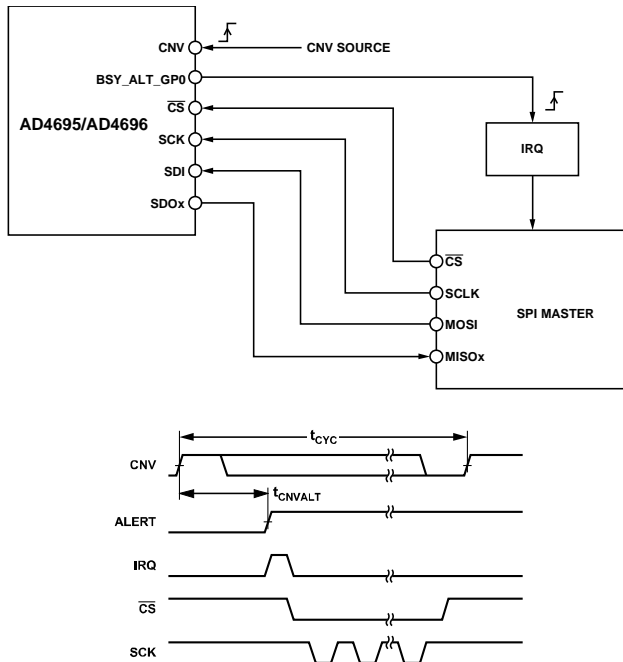
Figure 118 shows a simplified connection diagram and software architecture for utilizing the AD4695/AD4696 threshold detection alert indicator to synchronize the SPI master peripheral to the ADC conversion timing. The alert indicator must be enabled on the BSY_ALT_GP0 pin as described in the Alert Indicator on  section. The configuration in Figure 118 is ideal in autonomous conversion applications, where the SPI is idle while the ADC continuously converts until a user-defined out-of-bounds condition occurs. The alert indicator is updated at the end of the conversion phase of the ADC and can therefore be used as the trigger to start the SPI frame, so long as the SPI frame can be completed before the start of the next conversion. Typically, the interrupt service routine called by the alert indicator rising edge will call the SPI to read back the conversion result and send the register configuration mode command over SDI to put the AD4695/AD4696 into register configuration mode.

The configurations in Figure 118 is ideal when operating the AD4695/AD4696 in autocycle mode because it allows the digital host to be completely idle until an out-of-bounds condition occurs, and guarantees the digital host can remain synchronized to the internal conversion timing (see the SPI Peripheral Synchronization in Autocycle Mode section).

### SPI Peripheral Synchronization in Autocycle Mode

If autocycle mode is enabled when the AD4695/AD4696 enters conversion mode, the convert-start instructions for the ADC core are generated by an internal oscillator (see the Autocycle Mode section). Autocycle mode is therefore ideal for autonomous conversion applications, where the digital host is idle or in a sleep state until a user-programmed threshold detection event occurs.

The digital host SPI must not attempt to read/write data while the AD4695/AD4696 are still in the conversion phase. In autocycle mode, the convert-start signal is generated internally, and the digital host must therefore reference either the busy indicator or the alert indicator via the BSY_ALT_GP0 pin to synchronize the AD4695/AD4696 and digital host SPIs and ensure SPI frames occur between ADC conversion phases. Figure 98 shows the required SPI frame timing relative to the busy indicator and alert indicator when autocycle mode is enabled.

The busy indicator can be used to trigger an interrupt service routine to read the most recent conversion result and send the 5-bit SDI commands (see Figure 117). The busy indicator transitions low at the end of each conversion phase and transitions high at the start of each next conversion phase. The digital host must begin the SPI frame following the busy indicator falling edge, and the SCK rate must be sufficiently fast to complete the SPI frame at least 80 ns prior to the next busy indicator rising edge to conform to the $t_{SCKCNV}$ specification in Figure 98 and Table 2 (see the Conversion Mode SPI Clock Frequency Requirements section). The time duration between busy indicator falling edge and rising edge is given by the $t_{ACBSY}$ specification in Table 2.

The alert indicator can be used as a one-shot trigger for an interrupt service routine on the digital host to instruct the SPI master peripheral to send the register configuration mode command and poll the alert registers (see Figure 118). The alert indicator state is updated following the completion of the conversion phase. Therefore an alert indicator rising edge can signify to the digital host that the AD4695/AD4696 SPI is ready for a SPI frame. The alert indicator only transitions when a threshold violation is detected on a given channel, however, and therefore the digital host will not be able to read conversion results except for those that cause the alert indicator to go high (see the Alert Indicator on BSY_ALT_GP0 section).

As described in the SPI Peripheral Synchronization in Conversion Mode section, the digital host must complete the SPI frame before the start of the next conversion. Refer to Figure 98 and the Conversion Mode SPI Clock Frequency Requirements section for guidelines on minimum SCK frequency and overall system latency to achieve appropriate SPI transfer rates for the selected sample rate.

### Conversion Mode SPI Clock Frequency Requirements

Conversion results for a given sample are available until the start of the next conversion phase. The SCK frequency must therefore be fast enough to read the data from the AD4695/AD4696 SPI before the following CNV rising edge (or internal convert-start signal when autocycle mode is enabled).

The minimum required SCK frequency is a function of the sample rate in use, the length of the SPI frame (in bits), and the serial data output mode in use. Faster sample rates require faster SCK frequencies because the time between conversions is shorter. Dual-SDO mode significantly reduces the required SCK frequency for a given sample rate by doubling the number

of bits output on the SPI per SCK period (see the Serial Data Output Modes section).

The number of SCK periods required per conversion mode frame ($N_{SCK}$) is a function of the number of bits per frame ($N_{BITS}$) and the number of serial data outputs ($N_{SDO}$):

$$N_{SCK} = N_{BITS}/N_{SDO}$$

$N_{BITS}$ depends on the maximum oversampling ratio in use and whether the status bits are enabled (see Table 17, Table 18, and Table 19). $N_{SDO}$ is 1 for single-SDO mode and 2 for dual-SDO mode.

The Conversion Mode Timing Diagrams section shows timing diagrams for the SPI frames in conversion mode. The start of the conversion mode SPI frame must not occur before the $t_{CONVERT}$ time has elapsed and must complete early enough to adhere to the minimum $t_{SCKCNV}$ specification (see Table 2). The amount of time given to complete an SPI frame in conversion mode ($t_{FRAME}$) is:

$$t_{FRAME} = t_{CYC} - t_{CONVERT\_max} - t_{SCKCNV} = 1/f_{CNV} - t_{CONVERT\_max} - t_{SCKCNV}$$

where $t_{CYC}$ is the sample period, $t_{CONVERT\_max}$ is the maximum $t_{CONVERT}$ specification and $t_{SCKCNV}$ is the SCK to CNV rising edge delay specification (see Table 2).

The SCK frequency ($f_{SCK}$) is a function of $t_{FRAME}$ and $N_{SCK}$:

$$f_{SCK} > N_{SCK}/t_{FRAME} = \frac{N_{BITS}}{N_{SDO} \times (t_{CYC} - t_{CONVERT} - t_{SCKCNV})}$$

Table 22 shows examples of the minimum SCK frequency required for several sample rates for each serial data output

mode with status bits disabled and enabled and oversampling ratio set to 1.

Note when single-cycle command mode is enabled, the multiplexer does not update channels until the 5-bit channel command is clocked in on SDI. The SCK frequency therefore impacts the acquisition time ($t_{ACQ}$) when single-cycle command mode is enabled (see the Single-Cycle Command Mode section).

When autocycle mode is enabled, $t_{CYC}$ is determined by the internal convert-start signal, the period of which is set by the AC_CLK field, and the digital host must use either the busy indicator or the alert indicator to synchronize the SPI frames with the internal conversion timing (see the SPI Peripheral Synchronization in Autocycle Mode section).

The digital host SPI master peripheral may provide more SCK periods than required per conversion mode SPI frame. The behavior of SDO when additional SCK falling edges occur after the LSB is clocked out depends on the SDO_STATE bit setting. When SDO_STATE = 0, SDO maintains its state when extra SCKs occur. When SDO_STATE = 1, SDO transitions to high-impedance when extra SCKs occur.

Note that the minimum SCK period is longer for register configuration mode than for conversion mode (see $t_{SCK}$ in Table 2). In conversion mode, the minimum $t_{SCK}$ is 12.5 ns, corresponding to a maximum $f_{SCK}$ of 80 MHz. In register configuration mode, the minimum $t_{SCK}$ is 40 ns, corresponding to a maximum $f_{SCK}$ of 25 MHz. Therefore for applications requiring conversion mode SCK frequencies of greater than 25 MHz, ensure the SPI master peripheral serial clock rate is programmed accordingly while the AD4695/AD4696 are in register configuration mode.

Table 22. $f_{SCK}$ Requirements vs. Sample Rate and Serial Data Output Modes (Oversampling Ratio = 1)

| Sample Rate (kSPS) | Status Bits[1] | Single-SDO Mode | Dual-SDO Mode |
|---|---|---|---|
| 1000 (AD4696 Only) | Disabled | 30.8 | 15.4 |
| 1000 (AD4696 Only) | Enabled | 46.2 | 23.1 |
| 500 | Disabled | 10.5 | 5.3 |
| 500 | Enabled | 15.8 | 7.9 |
| 100 | Disabled | 1.7 | 0.8 |
| 100 | Enabled | 2.5 | 1.3 |

[1] In the calculations in Table 22, $N_{BITS}$ = 16 when status bits are disabled and $N_{BITS}$ = 24 when status bits are enabled.

## RESET Connection Recommendations

The RESET input allows the digital host to trigger a full device reset with a GPIO (see the Hardware Reset section). The RESET input is active low and must be driven low to initiate a hardware reset. The AD4695/AD4696 remain in the reset state until the RESET input is driven high.

Hardware resets are not required to operate the AD4695/AD4696, since the SPI provides a software reset option (see the Software Reset section). For systems not utilizing hardware reset functionality, tie the RESET input to VIO on board to ensure it is pulled high during device operation.

To utilize hardware resets, connect the RESET input to a GPIO or equivalent digital output from the digital host. The signal driving RESET must idle high. It is recommended to also include a weak pull-up resistor to VIO on the RESET input to ensure it is pulled high until the digital host output is in a defined state. The host firmware function for performing hardware resets must pulse RESET low following the timing requirements in Figure 99.

## DEVICE CONFIGURATION RECOMMENDATIONS

The following are recommendations for configuring the desired AD4695/AD4696 features and settings via the configuration registers described in the Register Information section.

The AD4695/AD4696 must be in register configuration mode to access the configuration registers via the SPI. The AD4695/AD4696 enter register configuration mode on device power-up and following device resets. The settings in the configuration registers must be properly programmed for the specific application prior to entering conversion mode and performing conversions.

On device power up it is recommended to perform either a hardware or software reset as described in the Device Reset section.

Program the contents of the SPI_CONFIG_A, SPI_CONFIG_B and SPI_CONFIG_C registers to the desired settings first to ensure the AD4695/AD4696 SPI protocol is configured to be compatible with the digital host (see the Register Configuration Mode section). The scratch pad register (SCRATCH_PAD) allows the digital host to validate communications with the AD4695/AD4696 by writing test values and reading them back without affecting device settings.

The device identification registers (DEVICE_TYPE, DEVICE_ID_L, DEVICE_ID_H, DEVICE_INFO, VENDOR_L and VENDOR_H) are provided for the digital host to verify that an AD4695/AD4696 device is connected.

Next, when powering VDD externally, it is recommended to disable the internal LDO by setting the LDO_EN bit in the SETUP register to 0 (see the Internal LDO section). Note that setting the SPI_MODE bit to a 1 will put the AD4695/AD4696 into conversion mode. Ensure that SPI_MODE is set to 0 until the remaining configuration registers are properly configured.

Next, configure the channel sequencing registers for the desired channel sequencing mode. The SEQ_CTRL register contains the STD_SEQ_EN bit and NUM_SLOTS_AS field, which must be configured to select the desired channel sequencing mode. By default, the STD_SEQ_EN bit is set to 1, which selects the standard sequencer (see Table 47).

If using the standard sequencer ensure the STD_SEQ_EN bit is set to 1, and then program the STD_SEQ_CONFIG register and TEMP_CTRL register to select the channels for the sequence (see Table 49 and Table 53). If including the temperature sensor in the channel sequence

If using the advanced sequencer, update the SEQ_CTRL register to set the STD_SEQ_EN bit to 0 and set the NUM_SLOTS_AS field to the desired number of advanced sequencer slots, and program the appropriate number of AS_SLOTn registers and TEMP_CTRL register to implement the desired channel sequence (see Table 53 and Table 60).

If using either two-cycle command mode or single-cycle command mode, update the SEQ_CTRL register to set the STD_SEQ_EN bit to 0 but keep the NUM_SLOTS_AS field set to 0x0. The CYC_CTRL bit must also be set to select between two-cycle and single-cycle command modes, but since CYC_CTRL is in the SETUP register, it can be configured in the same frame as the SPI_MODE bit is set to put the device in conversion mode.

After the channel sequencing mode settings are configured, update the CONFIG_INn register settings as needed to select the channel configuration settings, including threshold detection alert enable setting, polarity mode, pin pairing option, analog input high-Z mode enable setting, and oversampling ratio. When the standard sequencer is enabled, the settings programmed into the CONFIG_INn register bits are applied to all analog input channels. When any other channel sequencing mode is selected, the settings in each CONFIG_INn register are applied to their corresponding INn channel. See Table 54 for a detailed description of the bits in the CONFIG_INn registers.

When enabling threshold detection for any set of channels, update the values in the corresponding UPPER_INn and LOWER_INn registers to implement the desired upper and lower threshold limits (see Table 55 and Table 56). The ALERT_MODE bit must be updated to enable or disable hysteresis, but since ALERT_MODE is in the SETUP register, it can be configured in the same frame as the SPI_MODE bit is set to put the device in conversion mode. If enabling hysteresis, then the HYSTERESIS_INn registers must be updated to implement the desired hysteresis settings.

If utilizing any of the general purpose pin functions described in the General-Purpose Pin section, update the GPIO_CTRL and GP_MODE register contents accordingly (see Table 50 and Table 51). If configuring a general purpose pin as a general purpose output, set the corresponding GPO_WRITE bit in the GPIO_STATE register (see Table 52).

If using autocycle mode, update the settings in the AC_CTRL to enable autocycle mode and select the desired sample rate (see Table 48). Autocycle mode is disabled by default so if not using autocycle mode, it is not necessary to update the AC_CTRL register after a device reset.

If utilizing offset and gain correction, update the settings in the OFFSET_INn and GAIN_INn registers accordingly. If a calibration routine is required to determine the necessary offset and gain correction values for each channel, update the OFFSET_INn and GAIN_INn registers after putting the AD4695/AD4696 into conversion mode to collect enough conversion data.

After all other necessary configuration register settings have been updated, put the AD4695/AD4696 in conversion mode by setting the SPI_MODE bit in the SETUP register to 1. Ensure that all other bits in the SETUP register are set to achieve the desired device settings (see Table 45).

Prior to updating the SETUP register to put the device in conversion mode, the digital host may optionally check the state of the SPI_ERROR bit in the STATUS register to verify that there were no errors in updating the configuration registers. The host can also check the state of the CLAMP_STATUS1 and CLAMP_STATUS2 registers to check if any of the AD4695/AD4696 analog input channels are experiencing overvoltage events prior to putting the device into conversion mode.

While the AD4695/AD4696 is in conversion mode, the SPI cannot be used to update the configuration registers. If any of the configuration registers need to be read from or updated while the device is already in conversion mode, send the register configuration mode command during a conversion mode SPI frame to put the device back into register configuration mode (see the Register Configuration Mode Command section).

## EFFECTIVE CHANNEL SAMPLE RATE

The AD4695/AD4696 analog inputs are multiplexed to a single ADC core, and the state of the multiplexer is updated at the end of the conversion phase. The effective sample rate for each channel in the channel sequence is therefore some fraction of the sample rate of the ADC (set by $f_{CNV}$). The effective sample rate for a channel is defined as the frequency at which each new conversion results are generated for that channel.

In order for an analog input to have an effective sample rate, new results must be generated at a constant rate for the entire channel sequence or at least for a long enough time span to perform the necessary analysis. For example, to calculate an FFT and perform ac analysis on the ADC data for a given channel, the sampling interval between each sample gathered for that channel must be constant. The effective sample rate for an analog input ($f_{S\_INx}$) is a function of the frequency of the CNV signal ($f_{CNV}$) and the number of CNV periods between each time it is sampled ($N_{CNV}$). The following relationship applies for each of the 16 analog inputs (IN0–IN15) and for the temperature sensor:

$$f_{S\_INx} = f_{CNV}/N_{CNV}$$

The required $f_{S\_INx}$ for each analog input is determined by its input signal frequency range. The Nyquist frequency for a given analog input (which is half of $f_{S\_INx}$) must be greater than the highest signal frequency being measured to avoid aliasing.

When the standard sequencer is enabled, each of the enabled channels in the STD_SEQ_CONFIG register are sampled once per sequence iteration. $f_{S\_INx}$ is therefore always constant for each enabled channel when the standard sequencer is enabled, and is calculated as:

$$f_{S\_INx} = f_{CNV}/(N_{CNV} \times D)$$

where $N_{EN}$ is the number of inputs included in the channel sequence and D is the oversampling ratio selected by the OSR field in the CONFIG_IN0 register. $N_{EN}$ can range from 1 (only one channel enabled) to 17 (when all channels and the temperature sensor are enabled). In the example provided in Figure 73, $N_{EN}$ = 4 and D = 1, so $f_{S\_INx}$ is $f_{CNV}/4$. If the oversampling ratio was programmed to 4 in this example, then $f_{S\_INx}$ would be $f_{CNV}/16$.

When the advanced sequencer, two-cycle command mode or single-cycle command mode are enabled, the sequence of analog inputs is more flexible, and the channel sequence can be designed to implement multiple effective sample rates. This is useful in applications with a combination of channels with low-frequency or dc signals and channels with high-frequency or ac signals. The Implementing Two Effective Channel Sample Rates section describes how to design a channel sequence which achieves two effective sample rates for two sets of channels.

Table 23 and Figure 119 shows an example of a sequence which achieves three effective sample rates with four analog inputs. The sequence in Table 23 and Figure 119 can be implemented with the advanced sequencer, two-cycle command mode, or single-cycle command mode.

The advanced sequencer, two-cycle command mode, and single-cycle command mode can also be utilized to perform aperiodic conversions on analog inputs, for example when all channels have dc-type signals, or when the channel sequencing involves adaptive control logic.

**Table 23. Multiple Effective Sample Rates Example**

| Sequence Position | Input | Effective Sample Rate of Input |
|---|---|---|
| 0 | IN0 | $f_{CNV}/2$ |
| 1 | IN1 | $f_{CNV}/4$ |
| 2 | IN0 | $f_{CNV}/2$ |
| 3 | IN2 | $f_{CNV}/8$ |
| 4 | IN0 | $f_{CNV}/2$ |
| 5 | IN1 | $f_{CNV}/4$ |
| 6 | IN0 | $f_{CNV}/2$ |
| 7 | IN3 | $f_{CNV}/8$ |
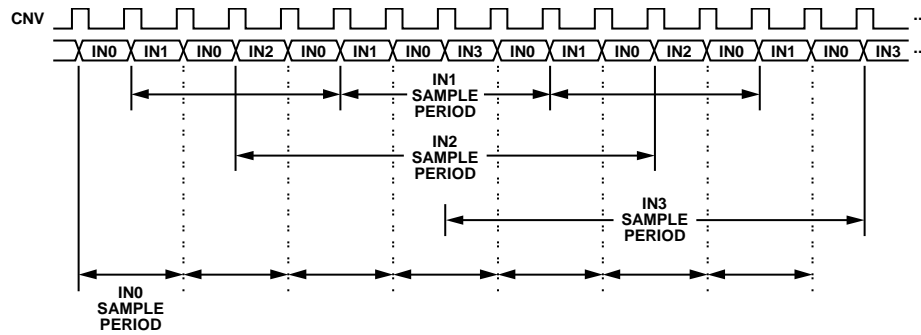
*Figure 119. Multiple Effective Sample Rates Example*

### Implementing Two Effective Channel Sample Rates

In multichannel data acquisition systems, the ADC may be monitoring a mix of higher-frequency and lower-frequency or dc-type signals. Channels with higher maximum input frequencies require higher Nyquist frequencies, and therefore require higher effective sample rates than channels with lower maximum input frequencies. To maximize the effective sample rate for analog input channels with higher-frequency input signals, the channel sequence can be designed to implement two different effective sample rates.

In a custom channel sequence which implements two effective sample rates, each of the AD4695/AD4696 channels included in the sequence are categorized as either high sample rate (HSR) channels or low sample rate (LSR) channels. Figure 120 shows a generalized channel sequence implementing HSR and LSR channels.
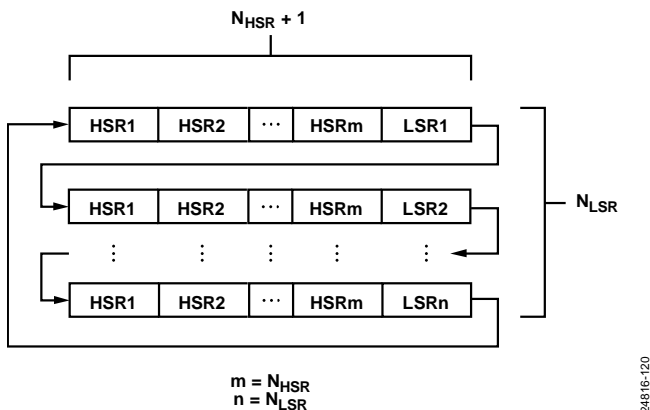


*Figure 120. Sequence of HSR and LSR Inputs with Two Effective Sample Rates*

The full channel sequence in Figure 120 consists of a repeating "sub-sequence" of all HSR channels, followed by one LSR channel. The sub-sequences repeat until all LSR channels are sampled once, and then the entire sequence starts again. As a result, the LSR channels are sampled only once per sequence iteration, while the HSR channels are sampled once for each LSR channel in the sequence.

The number of HSR channels ($N_{HSR}$) and the number of LSR channels ($N_{LSR}$) dictate their effective sample rates, as well as the number of sequence positions required to implement the two

sample rates. The number of sequence positions required ($N_S$) follows the relation:

$$N_S = N_{LSR} \times (N_{HSR} + 1)$$

where $N_{HSR}$ and $N_{LSR}$ are the number of HSR and LSR inputs, respectively. When the advanced sequencer is enabled, the maximum value of $N_S$ is limited by the number of AS_SLOTn registers. When two-cycle command mode or single-cycle command mode are enabled, $N_S$ can be arbitrarily large.

Since the LSR channels are only sampled once per full sequence iteration, their effective sample rate ($f_{S\_LSR}$) is the sample rate of the ADC core ($f_{CNV}$) divided by $N_S$:

$$f_{S\_LSR} = f_{CNV}/N_S$$

Since the HSR inputs are sampled once for each LSR input in the sequence, the effective sample rate for the HSR inputs ($f_{S\_HSR}$) is:

$$f_{S\_HSR} = (f_{CNV} \times N_{LSR})/N_S$$

Table 24 shows an example where IN5, IN9, and IN14 are HSR channels and IN2, IN10, and the temperature sensor (TEMP) are LSR channels.

**Table 24. Sequence with Two Effective Channel Sample Rates**

| Sequence Position | Input | Effective Sample Rate of Input |
|---|---|---|
| 0 | IN5 $f_{CNV}$ | $f_{CNV}/4$ |
| 1 | IN9 | $f_{CNV}/4$ |
| 2 | IN14 | $f_{CNV}/4$ |
| 3 | IN2 | $f_{CNV}/12$ |
| 4 | IN5 | $f_{CNV}/4$ |
| 5 | IN9 | $f_{CNV}/4$ |
| 6 | IN14 | $f_{CNV}/4$ |
| 7 | IN10 | $f_{CNV}/12$ |
| 8 | IN5 | $f_{CNV}/4$ |
| 9 | IN9 | $f_{CNV}/4$ |
| 10 | IN14 | $f_{CNV}/4$ |
| 11 | TEMP | $f_{CNV}/12$ |

Note that implementing the sequence in Table 24 with the advanced sequencer requires the following register configuration settings:

- STD_SEQ_EN = 0
- NUM_SLOTS_AS = 10
- TEMP_EN = 1

The first 11 advanced sequencer slots (AS_SLOT0–AS_SLOT10) are also programmed with the analog inputs listed in Table 24. These settings are because the temperature sensor is enabled via the TEMP_EN bit instead of via the advanced sequencer slots.

Note that when using the advanced sequencer, the temperature sensor cannot be assigned as an HSR channel because it cannot be assigned with the AS_SLOTn registers, but it can be included as an LSR channel by enabling it via the TEMP_EN bit in the TEMP_CTRL register, as demonstrated in Table 24.

## LAYOUT GUIDELINES

The following are suggested layout techniques for achieving optimal performance of the AD4695/AD4696 populated on a printed circuit board (PCB). An example PCB layout with the AD4696 is provided in the User Guide for the AD4696 evaluation board (EVAL-AD4696FMCZ).

Analog traces (i.e. traces connected to the analog inputs and reference input) should be physically separated from the digital traces (i.e. traces to the CNV input, SPI, and general purpose pins) to limit cross-coupling from fast-switching digital signals into the analog input signals. Add ground fill between analog and digital traces on the same PCB layer. Do not cross digital traces over analog traces or the AD4695/AD4696 device without a ground plane PCB layer in between. The analog and digital pins on the AD4695/AD4696 are arranged to facilitate separation of analog and digital traces.

The AD4695/AD4696 analog inputs (IN0–IN15) have a dynamic input impedance due to the multiplexer and ADC core input switches which toggle between conversions. An external capacitor is recommended to reduce nonlinear voltage steps at the analog inputs. These external capacitors should be placed as close to the analog inputs as possible to minimize parasitic impedances between the two which could degrade performance. See the Analog Front-End Design section for more information.

The AD4695/AD4696 voltage reference input, REF, also has a dynamic input impedance. The effective impedance between the reference drive circuitry output and the REF input must be very low, and a decoupling capacitor must be placed as close to the REF pin as possible. The external reference circuitry should be connected to the REF pin with wide traces to minimize the trace impedance. (See the Reference Circuitry Design section.)

The power supplies of the AD4695/AD4696 must be decoupled with low ESR ceramic capacitors placed close to the supply pins, and connected using short, wide traces to provide low impedance paths and to reduce the effect of glitches on the power supply lines (see the Power Supplies section). If LDO_IN is powered from the same supply as AVDD, the pins should be shorted with a wide common trace, and a single 100 nF capacitor can be used to decouple both pins.

## EVALUATING AD4695/AD4696 PERFORMANCE

The AD4695/AD4696 evaluation tool offerings include a fully assembled and tested evaluation board including the AD4696 (EVAL-AD4696FMCZ), evaluation software for controlling the board from a PC, and support documentation for the hardware and software. The evaluation software requires the EVAL-SDP-CH1Z controller board to establish communication between the PC and the EVAL-AD4696FMCZ board.

The EVAL-AD4696FMCZ board allows for prototyping the analog front-end circuitry and reference circuitry with the various digital features offered by the AD4696. It also features a standard 160-pin FMC connector and 12-pin extended SPI PMOD connector which allow for prototyping communication between the on-board AD4696 and many 3rd party FPGA and SoC development boards.

# REGISTER INFORMATION

## REGISTER OVERVIEW

The AD4695/AD4696 have programmable configuration registers which contain the bits and fields used to monitor device status and configure the device. Reading or writing to these bits and fields requires reading or writing to the registers that contain them. The AD4695/AD4696 SPI is used to read and write to the configuration registers (see the Register Configuration Mode section).

The AD4695/AD4696 register map memory space is divided into bytes. Each byte of memory has a unique address, ranging from 0x000 to 0x17F. Table 25 shows the register memory address assignments for all of the AD4695/AD4696 configuration registers.

Each configuration register is a single byte or multiple bytes in length. Registers that are multiple bytes long are called multibyte registers. The address of each multibyte register is defined as the address of its least significant byte, but each byte in a multibyte register has a unique address in the register map memory space. For example, the STD_SEQ_CONFIG register is two bytes long, and its least significant byte (LSByte) address is 0x24 and its most significant byte (MSByte) address is 0x25. The state of the MB_STRICT bit in the SPI_CONFIG_C register determines whether all bytes in a multibyte register

must be read or written in a single SPI transaction, or if each individual byte must be read or written in separate SPI transactions (see the Multibyte Register Access section).

Bits and fields in the AD4695/AD4696 configuration registers are defined as read-only, read/write or write-1-to-clear. Read-only bits can only be read from and cannot be updated by SPI writes from the SPI master. Read/write bits can be read from or written to. Write-1-to-clear bits can be read from and are only reset to 0 when the digital host writes a 1 in their memory location.

In the access column of Table 25, registers which contain exclusively read-only bits are represented with "R" and registers with writeable bits are represented with "R/W." In the access column of Table 26 through Table 60, read-only bits are represented with "R," read/write bits are represented with "R/W," and write-1-to-clear bits are represented with "R/W1C."

The SPI_STATUS register contains various error flags which indicate whether a SPI read or write transaction violated one of several aspects of the protocols outlined in the Register Configuration Mode section (see Table 37). The SPI_ERROR bit in the STATUS is the bitwise logical OR of the error flags in the SPI_STATUS register (see Table 38).

**Table 25. Configuration Register Names and Descriptions**

| Address | Name | Description | Length | Reset | Access |
|---|---|---|---|---|---|
| 0x000 | SPI_CONFIG_A | Interface Configuration A. | Single-Byte | 0x10 | R/W |
| 0x001 | SPI_CONFIG_B | Interface Configuration B. | Single-Byte | 0x00 | R/W |
| 0x003 | DEVICE_TYPE | Device Type. | Single-Byte | 0x07 | R |
| 0x004 | DEVICE_ID_L | Device ID (Lower Byte). | Single-Byte | [1] | R |
| 0x005 | DEVICE_ID_H | Device ID (Upper Byte). | Single-Byte | 0x00 | R |
| 0x006 | DEVICE_INFO | Device Information. | Single-Byte | 0x00 | R |
| 0x00A | SCRATCH_PAD | Scratch Pad. | Single-Byte | 0x00 | R/W |
| 0x00C | VENDOR_L | Vendor ID (Lower Byte). | Single-Byte | 0x56 | R |
| 0x00D | VENDOR_H | Vendor ID (Upper Byte). | Single-Byte | 0x04 | R |
| 0x00E | LOOP_MODE | Loop Mode. | Single-Byte | 0x00 | R/W |
| 0x010 | SPI_CONFIG_C | Interface Configuration C. | Single-Byte | 0x23 | R/W |
| 0x011 | SPI_STATUS | Interface Status. | Single-Byte | 0x00 | R/W |
| 0x014 | STATUS | Device Status. | Single-Byte | 0x20 | R |
| 0x015 | ALERT_STATUS1 | Alert Status (IN0-IN3). | Single-Byte | 0x00 | R |
| 0x016 | ALERT_STATUS2 | Alert Status (IN4-IN7). | Single-Byte | 0x00 | R |
| 0x017 | ALERT_STATUS3 | Alert Status (IN8-IN11). | Single-Byte | 0x00 | R |
| 0x018 | ALERT_STATUS4 | Alert Status (IN12-IN15). | Single-Byte | 0x00 | R |
| 0x01A | CLAMP_STATUS1 | Clamp Status (IN0-IN7). | Single-Byte | 0x00 | R |
| 0x01B | CLAMP_STATUS2 | Clamp Status (IN8-IN15). | Single-Byte | 0x00 | R |
| 0x020 | SETUP | Device Setup. | Single-Byte | 0x10 | R/W |
| 0x021 | REF_CTRL | Reference Control. | Single-Byte | 0x12 | R/W |
| 0x022 | SEQ_CTRL | Sequencer Control. | Single-Byte | 0x80 | R/W |
| 0x023 | AC_CTRL | Autocycle Control. | Single-Byte | 0x00 | R/W |
| 0x024 | STD_SEQ_CONFIG | Standard Sequencer Configuration. | Multibyte | 0x0001 | R/W |

[1] The default value of the DEVICE_ID_L register is different between the AD4695 and AD4696 (see Table 30).

| Address | Name | Description | Length | Reset | Access |
|---|---|---|---|---|---|
| 0x026 | GPIO_CTRL | GPIO Enable. | Single-Byte | 0x00 | R/W |
| 0x027 | GP_MODE | General-Purpose Pin Function Control. | Single-Byte | 0x00 | R/W |
| 0x028 | GPIO_STATE | GPIO State. | Single-Byte | 0x00 | R/W |
| 0x029 | TEMP_CTRL | Temperature Sensor Control. | Single-Byte | 0x00 | R/W |
| 0x030 - 0x03F | CONFIG_INn | Analog Input Settings Configuration. | Single-Byte | 0x08 | R/W |
| 0x040 - 0x05E | UPPER_INn | Upper Threshold Value. | Multibyte | 0x07FF | R/W |
| 0x060 - 0x07E | LOWER_INn | Lower Threshold Value. | Multibyte | 0x0000 | R/W |
| 0x080 - 0x09E | HYST_INn | Hysteresis Setting. | Multibyte | 0x0010 | R/W |
| 0x0A0 - 0x0BE | OFFSET_INn | INn Offset Correction. | Multibyte | 0x0000 | R/W |
| 0x0C0 - 0x0DE | GAIN_INn | INn Gain Correction. | Multibyte | 0x8000 | R/W |
| 0x100 - 0x17F | AS_SLOTn | Advanced Sequencer Slot. | Single-Byte | 0x00 | R/W |

## REGISTER DETAILS

### SPI Configuration A Register

**Address: 0x000, Reset: 0x10, Name: SPI_CONFIG_A**
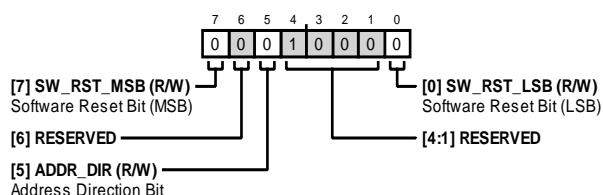


**Table 26. Bit Descriptions for SPI_CONFIG_A**

| Bits | Bit Name | Description | Reset | Access |
|---|---|---|---|---|
| 7 | SW_RST_MSB | Software Reset Bit (MSB). Setting both the SW_RST_MSB and SW_RST_LSB bits to 1 initiates a software reset of the device, resetting all registers except INTERFACE_CONFIG_A to the default power up state (see the Software Reset section). | 0x0 | R/W |
| 6 | RESERVED | Reserved. | 0x0 | R |
| 5 | ADDR_DIR | Address Direction Bit. Determines sequential addressing behavior when performing register reads and writes on multiple bytes of data in a single data phase (see the Address Direction Options section). 0: Selects descending address option. 1: Selects ascending address option. | 0x0 | R/W |
| [4:1] | RESERVED | Reserved. | 0x8 | R |
| 0 | SW_RST_LSB | Software Reset Bit (LSB). Setting both the SW_RST_MSB and SW_RST_LSB bits to 1 initiates a software reset of the device, resetting all registers except INTERFACE_CONFIG_A to the default power up state (see the Software Reset section). | 0x0 | R/W |

### SPI Configuration B Register

**Address: 0x001, Reset: 0x00, Name: SPI_CONFIG_B**
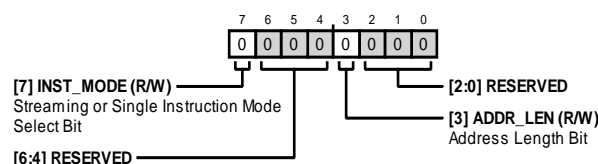


**Table 27. Bit Descriptions for SPI_CONFIG_B**

| Bits | Bit Name | Description | Reset | Access |
|---|---|---|---|---|
| 7 | INST_MODE | Streaming or Single Instruction Mode Select Bit. Selects between streaming mode and single instruction mode (see the Streaming Mode section and the Single Instruction Mode section). 0: Enables streaming mode. 1: Enables single instruction mode. | 0x0 | R/W |
| [6:4] | RESERVED | Reserved. | 0x0 | R |

Rev. PrI | Page 75 of 94

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| 3 | ADDR_LEN | Address Length Bit. Sets the length of the register address in the instruction phase to 7 bits or 15 bits. See the Instruction Phase section.<br>0: 15-bit Addressing.<br>1: 7-Bit Addressing. | 0x0 | R/W |
| [2:0] | RESERVED | Reserved. | 0x0 | R |

### Device Type Register

**Address: 0x003, Reset: 0x07, Name: DEVICE_TYPE**



**[7:4] RESERVED**

**[3:0] DEVICE_TYPE (R)**
Device Type Indicator Field
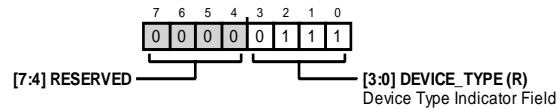
**Table 28. Bit Descriptions for DEVICE_TYPE**

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| [7:4] | RESERVED | Reserved. | 0x0 | R |
| [3:0] | DEVICE_TYPE | Device Type Indicator Field. Identifies the Analog Devices, Inc. product category that the device belongs to. The value 0x7 corresponds to precision ADCs. Read this field, the DEVICE_ID field, and the VENDOR_ID field to identify this device as the AD4695 or AD4696. The AD4695 is identified with VENDOR_ID = 0x0456, DEVICE_TYPE = 0x7, DEVICE_ID = 0x0015. The AD4696 is identified by VENDOR_ID = 0x0456, DEVICE_TYPE = 0x7, DEVICE_ID = 0x0016. | 0x7 | R |

### Device ID (Lower Byte) Register

**Address: 0x004, Reset: 0x10, Name: DEVICE_ID_L**



**[7:0] DEVICE_ID[7:0] (R)**
Device Identification Field
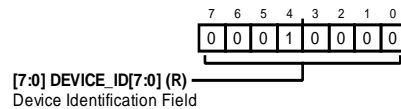
**Table 29. Bit Descriptions for DEVICE_ID_L**

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| [7:0] | DEVICE_ID[7:0] | Device Identification Field. The DEVICE_ID[15:0] field is unique to each Analog Devices, Inc. precision ADC. Read this field, the DEVICE_TYPE field, and the VENDOR_ID field to identify this device as the AD4695 or AD4696. The AD4695 is identified with VENDOR_ID = 0x0456, DEVICE_TYPE = 0x7, DEVICE_ID = 0x0015. The AD4696 is identified by VENDOR_ID = 0x0456, DEVICE_TYPE = 0x7, DEVICE_ID = 0x0016. | AD4695: 0x05, AD4696: 0x06 | R |

### Device ID (Upper Byte) Register

**Address: 0x005, Reset: 0x00, Name: DEVICE_ID_H**



**[7:0] DEVICE_ID[15:8] (R)**
Device Identification Field
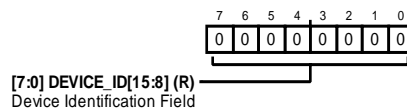
**Table 30. Bit Descriptions for DEVICE_ID_H**

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| [7:0] | DEVICE_ID[15:8] | Device Identification Field. The DEVICE_ID[15:0] field is unique to each Analog Devices, Inc. precision ADC. Read this field, the DEVICE_TYPE field, and the VENDOR_ID field to identify this device as the AD4695 or AD4696. The AD4695 is identified with VENDOR_ID = 0x0456, DEVICE_TYPE = 0x7, DEVICE_ID = 0x0015. The AD4696 is identified by VENDOR_ID = 0x0456, DEVICE_TYPE = 0x7, DEVICE_ID = 0x0016. | 0x00 | R |

### Device Information Register

**Address: 0x006, Reset: 0x00, Name: DEVICE_INFO**



**[7:4] PACKAGE (R)** — Device Package Indicator.
**[3:0] RESERVED**
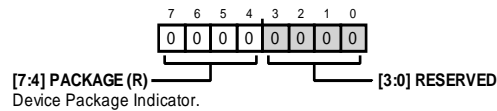
**Table 31. Bit Descriptions for DEVICE_INFO**

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| [7:4] | PACKAGE | Device Package Indicator. Indicates the package option for the connected device. 0x0: LFCSP. | 0x0 | R |
| [3:0] | RESERVED | Reserved. | 0x0 | R |

### Scratch Pad Register

**Address: 0x00A, Reset: 0x00, Name: SCRATCH_PAD**



**[7:0] SCRATCH_VALUE (R/W)** — Scratchpad Field
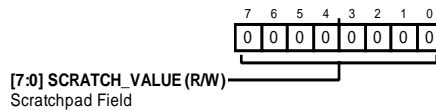
**Table 32. Bit Descriptions for SCRATCH_PAD**

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| [7:0] | SCRATCH_VALUE | Scratchpad Field. Values written to this register have no impact on device behavior. Use this register to test SPI communications with the device. | 0x00 | R/W |

### Vendor ID (Lower Byte) Register

**Address: 0x00C, Reset: 0x56, Name: VENDOR_L**



**[7:0] VENDOR_ID[7:0] (R)** — Vendor Identification Field
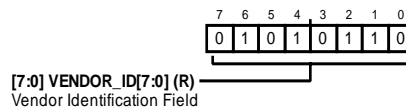
**Table 33. Bit Descriptions for VENDOR_L**

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| [7:0] | VENDOR_ID[7:0] | Vendor Identification Field. The VENDOR_ID[15:0] field is the same value (0x0456) for all Analog Devices, Inc. products. Read this field, the DEVICE_TYPE field, and the DEVICE_ID field to identify this device as the AD4695 or AD4696. The AD4695 is identified with VENDOR_ID = 0x0456, DEVICE_TYPE = 0x7, DEVICE_ID = 0x0015. The AD4696 is identified by VENDOR_ID = 0x0456, DEVICE_TYPE = 0x7, DEVICE_ID = 0x0016. | 0x56 | R |

### Vendor ID (Upper Byte) Register

**Address: 0x00D, Reset: 0x04, Name: VENDOR_H**



**[7:0] VENDOR_ID[15:8] (R)** — Vendor Identification Field
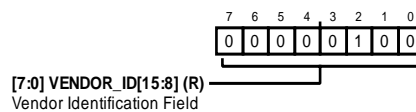
**Table 34. Bit Descriptions for VENDOR_H**

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| [7:0] | VENDOR_ID[15:8] | Vendor Identification Field. The VENDOR_ID[15:0] field is the same value (0x0456) for all Analog Devices, Inc. products. Read this field, the DEVICE_TYPE field, and the DEVICE_ID field to identify this device as the AD4695 or AD4696. The AD4695 is identified with VENDOR_ID = 0x0456, DEVICE_TYPE = 0x7, DEVICE_ID = 0x0015. The AD4696 is identified by VENDOR_ID = 0x0456, DEVICE_TYPE = 0x7, DEVICE_ID = 0x0016. | 0x04 | R |

### Loop Mode Register

**Address: 0x00E, Reset: 0x00, Name: LOOP_MODE**

```
7 6 5 4 3 2 1 0
0 0 0 0 0 0 0 0
```
**[7:0] LOOP_COUNT (R/W)**
Loop Count Field
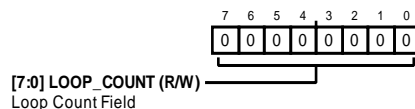
**Table 35. Bit Descriptions for LOOP_MODE**

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| [7:0] | LOOP_COUNT | Loop Count Field. Specifies the number of registers to loop through for each SPI frame when streaming mode is selected (see the Streaming Mode section). A value of 0x00 disables looping. Values between 0x01 and 0xFF sets the number of registers to loop through before returning to the original register address. | 0x00 | R/W |

### SPI Configuration C Register

**Address: 0x010, Reset: 0x23, Name: SPI_CONFIG_C**

```
7 6 5 4 3 2 1 0
0 0 1 0 0 0 1 1
```
**[7:6] CRC_EN (R/W)**　　　　**[1:0] CRC_EN_N (R/W)**
CRC Enable Field　　　　Inverted CRC Enable Field
**[5] MB_STRICT (R/W)**　　　**[4:2] RESERVED**
Multibyte Access Control Bit
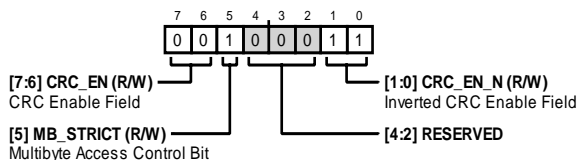
**Table 36. Bit Descriptions for SPI_CONFIG_C**

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| [7:6] | CRC_EN | CRC Enable Field. Enables CRC when set to 0x1 (if CRC_EN_N is also set to 0x2). Disables CRC when set to a value other than 0x1 (see the Checksum Protection section).<br>0: Disables CRC.<br>1: Enables CRC if CRC_EN_N = 0x2. | 0x0 | R/W |
| 5 | MB_STRICT | Multibyte Access Control Bit. Sets the SPI transaction requirements for multibyte registers (see the Multibyte Register Access section).<br>0: Individual bytes in multibyte registers are read from or written to in individual data phases.<br>1: All bytes in multibyte registers are read from or written to in a single data phase. | 0x1 | R/W |
| [4:2] | RESERVED | Reserved. | 0x0 | R |
| [1:0] | CRC_EN_N | Inverted CRC Enable Field. Enables CRC when set to 0x2 (if CRC_EN is also set to 0x1). Disables CRC when set to a value other than 0x2. (See the Checksum Protection section.) | 0x3 | R/W |

### Interface Status Register

**Address: 0x011, Reset: 0x00, Name: SPI_STATUS**

```
7 6 5 4 3 2 1 0
0 0 0 0 0 0 0 0
```
**[7] NOT_RDY_ERROR (R/W1C)**　　　**[0] ADDR_INVALID (R/W1C)**
Interface Not Ready Error Flag　　　Invalid Address Error Flag
**[6:5] RESERVED**　　　　　　　　　**[1] MB_ERROR (R/W1C)**
　　　　　　　　　　　　　　　　　　Multibyte Register Access Error Flag
**[4] SCK_ERROR (R/W)**
SPI Clock Count Error Flag　　　　　**[2] WRITE_INVALID (R/W1C)**
　　　　　　　　　　　　　　　　　　Invalid Write Error Flag
**[3] CRC_ERROR (R/W1C)**
CRC Error Flag
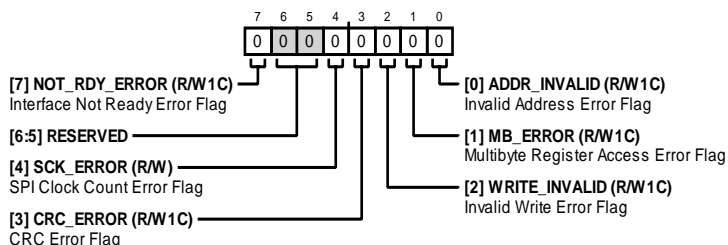
**Table 37. Bit Descriptions for SPI_STATUS**

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| 7 | NOT_RDY_ERROR | Interface Not Ready Error Flag. NOT_RDY_ERROR is set to 1 when the digital host initiates a SPI transaction before the AD4695/AD4696 interface was ready to respond, for example, before a device reset is finished. | 0x0 | R/W1C |
| [6:5] | RESERVED | Reserved. | 0x0 | R |

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| 4 | SCK_ERROR | SPI Clock Count Error Flag. SCK_ERROR is set to 1 when an incorrect number of serial clock edges is received in an SPI read or write transaction, for example, if the SPI frame ended in the middle of a data phase. | 0x0 | R/W |
| 3 | CRC_ERROR | CRC Error Flag. CRC_ERROR is set to 1 when the AD4695/AD4696 receives a checksum that does not match its expected value (see the Checksum Protection section). This error flag is only active when CRC is enabled. | 0x0 | R/W1C |
| 2 | WRITE_INVALID | Invalid Write Error Flag. WRITE_INVALID is set to 1 when the digital host attempts an SPI write to a register that contains exclusively read-only bits. | 0x0 | R/W1C |
| 1 | MB_ERROR | Multibyte Register Access Error Flag. MB_ERROR is set to 1 when a SPI transaction does not access all bytes of a multibyte register. This error flag is only active when the MB_STRICT bit is set to 1. | 0x0 | R/W1C |
| 0 | ADDR_INVALID | Invalid Address Error Flag. ADDR_INVALID is set to 1 when an SPI transaction attempts to access a nonexistent register (a register with an address outside of the specified range of values in Table 25). | 0x0 | R/W1C |

### Device Status Register

**Address: 0x014, Reset: 0x20, Name: STATUS**
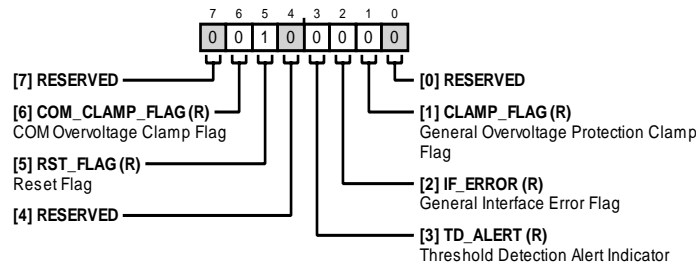


Table 38. Bit Descriptions for STATUS

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| 7 | RESERVED | Reserved. | 0x0 | R |
| 6 | COM_CLAMP_FLAG | COM Overvoltage Clamp Flag. Indicates if the COM overvoltage protection clamp is active due to an overvoltage event. This bit is not sticky and is cleared when the COM overvoltage protection clamp is inactive. <br> 0: COM overvoltage protection clamp inactive. <br> 1: COM overvoltage protection clamp active. | 0x0 | R |
| 5 | RST_FLAG | Reset Flag. Indicates whether a hardware reset, software reset, or POR has occurred since the last time this bit was read (see the Device Reset section). This bit is automatically cleared when read. <br> 0: No device reset occurred since this bit was last read. <br> 1: A device reset has occurred since this bit was last read. | 0x1 | R |
| 4 | RESERVED | Reserved. | 0x0 | R |
| 3 | TD_ALERT | Threshold Detection Alert Indicator. Indicates if any combination of the upper or lower alert indicators for IN0 thru IN15 are asserted. TD_ALERT is the logical OR of all HI_INn and LO_INn bits in the ALERT_STATUS1 thru ALERT_STATUS4 registers. This bit is not sticky. <br> 0: No upper or lower alert indicators asserted. <br> 1: At least one upper or lower alert indicator asserted. | 0x0 | R |
| 2 | SPI_ERROR | General Interface Error Flag. Indicates if any of the error flags in the SPI_STATUS register are asserted. This bit is the bitwise logical OR of all bits in the SPI_STATUS register. <br> 0: No interface error detected. <br> 1: One or more interface errors detected. | 0x0 | R |
| 1 | CLAMP_FLAG | General Overvoltage Protection Clamp Flag. Indicates if any of the IN0 thru IN15 overvoltage protection clamps were activated by an overvoltage event (if any of the INX_CLAMP_FLAG bits are asserted). This bit is sticky and is only cleared if all of the INX_CLAMP_FLAG bits are de-asserted when it is read. <br> 0: IN0 thru IN15 overvoltage clamps all inactive. <br> 1: At least one of the IN0 thru IN15 overvoltage clamps are active. | 0x0 | R |
| 0 | RESERVED | Reserved. | 0x0 | R |

### Alert Status (IN0-IN3) Register

Address: 0x015, Reset: 0x00, Name: ALERT_STATUS1
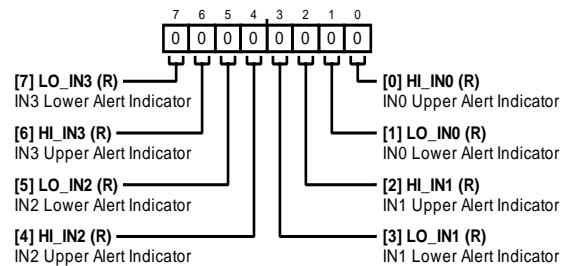


**Table 39. Bit Descriptions for ALERT_STATUS1**

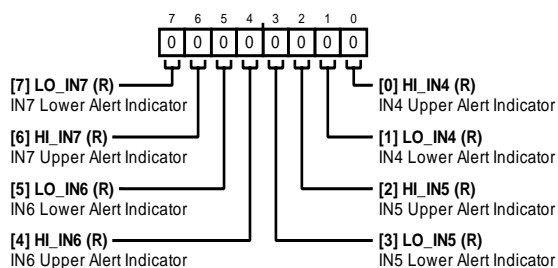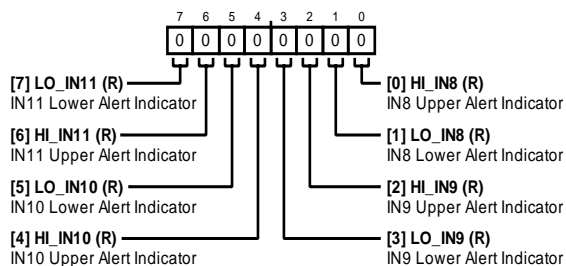| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| 7 | LO_IN3 | IN3 Lower Alert Indicator. LO_IN3 is set to 1 when a conversion result for IN3 is less than or equal to the IN3 lower threshold value. This indicator is only active if the threshold detection is enabled on IN3 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN3 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN3 register (see the Alert Indicator Registers section). | 0x0 | R |
| 6 | HI_IN3 | IN3 Upper Alert Indicator. HI_IN3 is set to 1 when a conversion result for IN3 is greater than or equal to the IN3 upper threshold value. This indicator is only active if the threshold detection is enabled on IN3 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN3 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN3 register (see the Alert Indicator Registers section). | 0x0 | R |
| 5 | LO_IN2 | IN2 Lower Alert Indicator. LO_IN2 is set to 1 when a conversion result for IN2 is less than or equal to the IN2 lower threshold value. This indicator is only active if the threshold detection is enabled on IN2 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN2 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN2 register (see the Alert Indicator Registers section). | 0x0 | R |
| 4 | HI_IN2 | IN2 Upper Alert Indicator. HI_IN2 is set to 1 when a conversion result for IN2 is greater than or equal to the IN2 upper threshold value. This indicator is only active if the threshold detection is enabled on IN2 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN2 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN2 register (see the Alert Indicator Registers section). | 0x0 | R |
| 3 | LO_IN1 | IN1 Lower Alert Indicator. LO_IN1 is set to 1 when a conversion result for IN1 is less than or equal to the IN1 lower threshold value. This indicator is only active if the threshold detection is enabled on IN1 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN1 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN1 register (see the Alert Indicator Registers section). | 0x0 | R |
| 2 | HI_IN1 | IN1 Upper Alert Indicator. HI_IN1 is set to 1 when a conversion result for IN1 is greater than or equal to the IN1 upper threshold value. This indicator is only active if the threshold detection is enabled on IN1 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN1 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN1 register (see the Alert Indicator Registers section). | 0x0 | R |
| 1 | LO_IN0 | IN0 Lower Alert Indicator. LO_IN0 is set to 1 when a conversion result for IN0 is less than or equal to the IN0 lower threshold value. This indicator is only active if the threshold detection is enabled on IN0 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN0 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN0 register (see the Alert Indicator Registers section). | 0x0 | R |
| 0 | HI_IN0 | IN0 Upper Alert Indicator. HI_IN0 is set to 1 when a conversion result for IN0 is greater than or equal to the IN0 upper threshold value. This indicator is only active if the threshold detection is enabled on IN0 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN0 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN0 register (see the Alert Indicator Registers section). | 0x0 | R |

*Alert Status (IN4-IN7) Register*

**Address: 0x016, Reset: 0x00, Name: ALERT_STATUS2**



**Table 40. Bit Descriptions for ALERT_STATUS2**

| Bits | Bit Name | Description | Reset | Access |
|---|---|---|---|---|
| 7 | LO_IN7 | IN7 Lower Alert Indicator. LO_IN7 is set to 1 when a conversion result for IN7 is less than or equal to the IN7 lower threshold value. This indicator is only active if the threshold detection is enabled on IN7 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN7 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN7 register (see the Alert Indicator Registers section). | 0x0 | R |
| 6 | HI_IN7 | IN7 Upper Alert Indicator. HI_IN7 is set to 1 when a conversion result for IN7 is greater than or equal to the IN7 upper threshold value. This indicator is only active if the threshold detection is enabled on IN7 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN7 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN7 register (see the Alert Indicator Registers section). | 0x0 | R |
| 5 | LO_IN6 | IN6 Lower Alert Indicator. LO_IN6 is set to 1 when a conversion result for IN6 is less than or equal to the IN6 lower threshold value. This indicator is only active if the threshold detection is enabled on IN6 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN6 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN6 register (see the Alert Indicator Registers section). | 0x0 | R |
| 4 | HI_IN6 | IN6 Upper Alert Indicator. HI_IN6 is set to 1 when a conversion result for IN6 is greater than or equal to the IN6 upper threshold value. This indicator is only active if the threshold detection is enabled on IN6 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN6 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN6 register (see the Alert Indicator Registers section). | 0x0 | R |
| 3 | LO_IN5 | IN5 Lower Alert Indicator. LO_IN5 is set to 1 when a conversion result for IN5 is less than or equal to the IN5 lower threshold value. This indicator is only active if the threshold detection is enabled on IN5 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN5 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN5 register (see the Alert Indicator Registers section). | 0x0 | R |
| 2 | HI_IN5 | IN5 Upper Alert Indicator. HI_IN5 is set to 1 when a conversion result for IN5 is greater than or equal to the IN5 upper threshold value. This indicator is only active if the threshold detection is enabled on IN5 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN5 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN5 register (see the Alert Indicator Registers section). | 0x0 | R |
| 1 | LO_IN4 | IN4 Lower Alert Indicator. LO_IN4 is set to 1 when a conversion result for IN4 is less than or equal to the IN4 lower threshold value. This indicator is only active if the threshold detection is enabled on IN4 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN4 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN4 register (see the Alert Indicator Registers section). | 0x0 | R |
| 0 | HI_IN4 | IN4 Upper Alert Indicator. HI_IN4 is set to 1 when a conversion result for IN4 is greater than or equal to the IN4 upper threshold value. This indicator is only active if the threshold detection is enabled on IN4 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN4 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN4 register (see the Alert Indicator Registers section). | 0x0 | R |

### *Alert Status (IN8-IN11) Register*

Address: 0x017, Reset: 0x00, Name: ALERT_STATUS3



**Table 41. Bit Descriptions for ALERT_STATUS3**

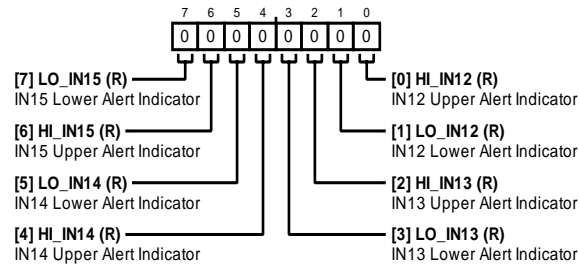| Bits | Bit Name | Description | Reset | Access |
|---|---|---|---|---|
| 7 | LO_IN11 | IN11 Lower Alert Indicator. LO_IN11 is set to 1 when a conversion result for IN11 is less than or equal to the IN11 lower threshold value. This indicator is only active if the threshold detection is enabled on IN0 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN11 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN11 register (see the Alert Indicator Registers section). | 0x0 | R |
| 6 | HI_IN11 | IN11 Upper Alert Indicator. HI_IN11 is set to 1 when a conversion result for IN11 is greater than or equal to the IN11 upper threshold value. This indicator is only active if the threshold detection is enabled on IN11 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN11 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN11 register (see the Alert Indicator Registers section). | 0x0 | R |
| 5 | LO_IN10 | IN10 Lower Alert Indicator. LO_IN10 is set to 1 when a conversion result for IN10 is less than or equal to the IN10 lower threshold value. This indicator is only active if the threshold detection is enabled on IN0 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN10 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN10 register (see the Alert Indicator Registers section). | 0x0 | R |
| 4 | HI_IN10 | IN10 Upper Alert Indicator. HI_IN10 is set to 1 when a conversion result for IN10 is greater than or equal to the IN10 upper threshold value. This indicator is only active if the threshold detection is enabled on IN10 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN10 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN10 register (see the Alert Indicator Registers section). | 0x0 | R |
| 3 | LO_IN9 | IN9 Lower Alert Indicator. LO_IN9 is set to 1 when a conversion result for IN9 is less than or equal to the IN9 lower threshold value. This indicator is only active if the threshold detection is enabled on IN9 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN9 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN9 register (see the Alert Indicator Registers section). | 0x0 | R |
| 2 | HI_IN9 | IN9 Upper Alert Indicator. HI_IN9 is set to 1 when a conversion result for IN9 is greater than or equal to the IN9 upper threshold value. This indicator is only active if the threshold detection is enabled on IN9 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN9 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN9 register (see the Alert Indicator Registers section). | 0x0 | R |
| 1 | LO_IN8 | IN8 Lower Alert Indicator. LO_IN8 is set to 1 when a conversion result for IN8 is less than or equal to the IN8 lower threshold value. This indicator is only active if the threshold detection is enabled on IN8 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN8 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN8 register (see the Alert Indicator Registers section). | 0x0 | R |
| 0 | HI_IN8 | IN8 Upper Alert Indicator. HI_IN8 is set to 1 when a conversion result for IN8 is greater than or equal to the IN8 upper threshold value. This indicator is only active if the threshold detection is enabled on IN8 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN8 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN8 register (see the Alert Indicator Registers section). | 0x0 | R |

### Alert Status (IN12-IN15) Register

**Address: 0x018, Reset: 0x00, Name: ALERT_STATUS4**



**Table 42. Bit Descriptions for ALERT_STATUS4**

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| 7 | LO_IN15 | IN15 Lower Alert Indicator. LO_IN15 is set to 1 when a conversion result for IN15 is less than or equal to the IN15 lower threshold value. This indicator is only active if the threshold detection is enabled on IN0 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN15 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN15 register (see the Alert Indicator Registers section). | 0x0 | R |
| 6 | HI_IN15 | IN15 Upper Alert Indicator. HI_IN15 is set to 1 when a conversion result for IN15 is greater than or equal to the IN15 upper threshold value. This indicator is only active if the threshold detection is enabled on IN15 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN15 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN15 register (see the Alert Indicator Registers section). | 0x0 | R |
| 5 | LO_IN14 | IN14 Lower Alert Indicator. LO_IN14 is set to 1 when a conversion result for IN14 is less than or equal to the IN14 lower threshold value. This indicator is only active if the threshold detection is enabled on IN0 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN14 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN14 register (see the Alert Indicator Registers section). | 0x0 | R |
| 4 | HI_IN14 | IN14 Upper Alert Indicator. HI_IN14 is set to 1 when a conversion result for IN14 is greater than or equal to the IN14 upper threshold value. This indicator is only active if the threshold detection is enabled on IN14 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN14 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN14 register (see the Alert Indicator Registers section). | 0x0 | R |
| 3 | LO_IN13 | IN13 Lower Alert Indicator. LO_IN13 is set to 1 when a conversion result for IN13 is less than or equal to the IN13 lower threshold value. This indicator is only active if the threshold detection is enabled on IN0 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN13 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN13 register (see the Alert Indicator Registers section). | 0x0 | R |
| 2 | HI_IN13 | IN13 Upper Alert Indicator. HI_IN13 is set to 1 when a conversion result for IN13 is greater than or equal to the IN13 upper threshold value. This indicator is only active if the threshold detection is enabled on IN13 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN13 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN13 register (see the Alert Indicator Registers section). | 0x0 | R |
| 1 | LO_IN12 | IN12 Lower Alert Indicator. LO_IN12 is set to 1 when a conversion result for IN12 is less than or equal to the IN12 lower threshold value. This indicator is only active if the threshold detection is enabled on IN0 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN12 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN12 register (see the Alert Indicator Registers section). | 0x0 | R |

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| 0 | HI_IN12 | IN12 Upper Alert Indicator. HI_IN12 is set to 1 when a conversion result for IN12 is greater than or equal to the IN12 upper threshold value. This indicator is only active if the threshold detection is enabled on IN12 (see the Threshold Detection and Alert Indicators section). This bit is read-to-clear. When the ALERT_MODE bit in the SETUP register is set to 1, this bit also automatically clears if a subsequent IN12 conversion is within the range set by the HYSTERESIS field in the HYSTERESIS_IN12 register (see the Alert Indicator Registers section). | 0x0 | R |

### *Clamp Status (IN0-IN7) Register*

**Address: 0x01A, Reset: 0x00, Name: CLAMP_STATUS1**

```
      7   6   5   4   3   2   1   0
     [0] [0] [0] [0] [0] [0] [0] [0]
      └───────────────────────────┘
[7:0] INX_CLAMP_FLAG[7:0] (R)
INx Overvoltage Clamp Flags
```
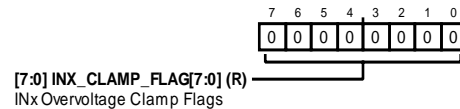
**Table 43. Bit Descriptions for CLAMP_STATUS1**

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| [7:0] | INX_CLAMP_FLAG[7:0] | INx Overvoltage Clamp Flags. Indicates if the INx overvoltage protection clamps are active due to an overvoltage event. Each bit corresponds to one of the analog inputs (IN0 thru IN15), where INX_CLAMP_FLAG[x] corresponds to the INx overvoltage protection clamp status. INX_CLAMP_FLAG[x] is set to 1 while the INx overvoltage protection clamp is active. These bits are not sticky and are automatically cleared when their corresponding overvoltage protection clamp deactivates. | 0x0 | R |

### *Clamp Status (IN8-IN15) Register*

**Address: 0x01B, Reset: 0x00, Name: CLAMP_STATUS2**

```
      7   6   5   4   3   2   1   0
     [0] [0] [0] [0] [0] [0] [0] [0]
      └───────────────────────────┘
[7:0] INX_CLAMP_FLAG[15:8] (R)
INx Overvoltage Clamp Flags
```
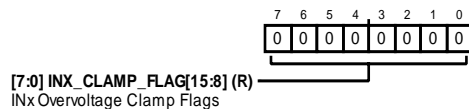
**Table 44. Bit Descriptions for CLAMP_STATUS2**

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| [7:0] | INX_CLAMP_FLAG[15:8] | INx Overvoltage Clamp Flags. Indicates if the INx overvoltage protection clamps are active due to an overvoltage event. Each bit corresponds to one of the analog inputs (IN0 thru IN15), where INX_CLAMP_FLAG[x] corresponds to the INx overvoltage protection clamp status. INX_CLAMP_FLAG[x] is set to 1 while the INx overvoltage protection clamp is active. These bits are not sticky and are automatically cleared when their corresponding overvoltage protection clamp deactivates. | 0x0 | R |

### Device Setup Register

**Address: 0x020, Reset: 0x10, Name: SETUP**
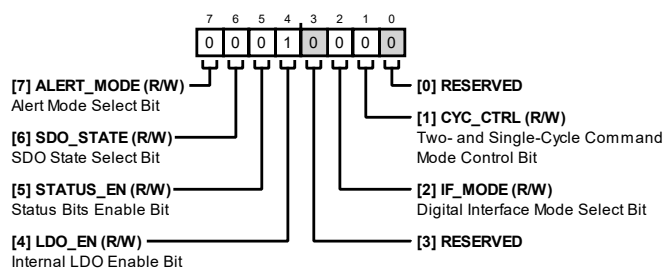


**Table 45. Bit Descriptions for SETUP**

| Bits | Bit Name | Description | Reset | Access |
|---|---|---|---|---|
| 7 | ALERT_MODE | Alert Mode Select Bit. Determines how the upper and lower alert indicators (HI_INn and LO_INn) are cleared (see the Alert Indicator Registers section).<br>0: Hysteresis enabled.<br>1: Hysteresis disabled. | 0x0 | R/W |
| 6 | SDO_STATE | SDO State Select Bit. Determines the behavior of serial data output(s) at the beginning and end of conversion mode SPI frames (see the Conversion Mode Timing Diagrams section).<br>0: Serial data output(s) hold their final value until the MSB of the next conversion data is clocked out.<br>1: Busy indicator enabled on the serial data output(s). | 0x0 | R/W |
| 5 | STATUS_EN | Status Bits Enable Bit. Determines whether the status bits are appended to conversion data when in conversion mode (see the Status Bits section).<br>0: Status bits disabled.<br>1: Status bits enabled. | 0x0 | R/W |
| 4 | LDO_EN | Internal LDO Enable Bit. Enables or disables the internal LDO. Disable the internal LDO when driving VDD with an external +1.8 V supply. When the internal LDO is supplying VDD, disabling the internal LDO will remove power to VDD, and disable the ADC core and configuration registers (see the Internal LDO section).<br>0: Internal LDO disabled.<br>1: Internal LDO enabled. | 0x1 | R/W |
| 3 | RESERVED | Reserved. | 0x0 | R/W |
| 2 | SPI_MODE | Digital Interface Mode Select Bit. Determines whether the device is in register configuration mode or conversion mode. Set SPI_MODE to 1 to enter conversion mode. SPI_MODE is set to 0 after receiving the register configuration mode command (see the Register Configuration Mode Command section).<br>0: Selects register configuration mode.<br>1: Selects conversion mode. | 0x0 | R/W |
| 1 | CYC_CTRL | Two- and Single-Cycle Command Mode Control Bit. Selects between two-cycle command mode and single-cycle command mode. This bit must be set to 0 when using two-cycle command mode, the standard sequencer, or the advanced sequencer (see the Channel Sequencing Modes section).<br>0: Selects two-cycle command mode.<br>1: Selects single-cycle command mode. | 0x0 | R/W |
| 0 | RESERVED | Reserved. | 0x0 | R/W |

### Reference Control Register

Address: 0x021, Reset: 0x12, Name: REF_CTRL
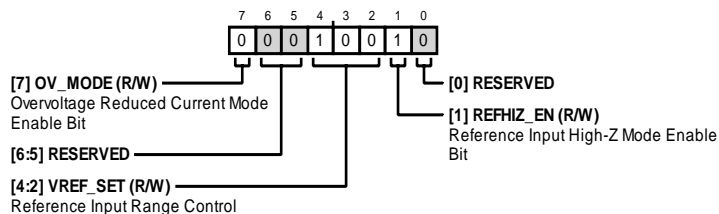


**Table 46. Bit Descriptions for REF_CTRL**

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| 7 | OV_MODE | Overvoltage Reduced Current Mode Enable Bit. Enables or disables overvoltage reduced current mode (see the Input Overvoltage Protection Clamps section).<br>0: Reduce REF current during clamping.<br>1: Do not reduce REF current during clamping. | 0x0 | R/W |
| [6:5] | RESERVED | Reserved. | 0x0 | R/W |
| [4:2] | VREF_SET | Reference Input Range Control. Configures the device to optimize performance based on the reference voltage in use. This field must be programmed to match the VREF voltage applied to the REF pin (see the Voltage Reference Input section).<br>0x0: 2.4 V ≤ VREF ≤ 2.75 V.<br>0x1: 2.75V < VREF ≤ 3.25V.<br>0x2: 3.25V < VREF ≤ 3.75V.<br>0x3: 3.75V < VREF ≤ 4.50V.<br>0x4: 4.5V < VREF ≤ 5.10 V. | 0x4 | R/W |
| 1 | REFHIZ_EN | Reference Input High-Z Mode Enable Bit. Enables or disables reference input high-Z mode (see the Reference Input High-Z Mode section).<br>0: Disables reference input high-Z mode.<br>1: Enables reference input high-Z mode. | 0x1 | R/W |
| 0 | RESERVED | Reserved. | 0x0 | R/W |

### Sequencer Control Register

Address: 0x022, Reset: 0x80, Name: SEQ_CTRL
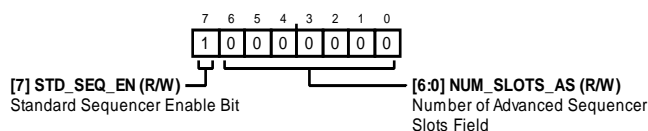


**Table 47. Bit Descriptions for SEQ_CTRL**

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| 7 | STD_SEQ_EN | Standard Sequencer Enable Bit. Enables or disables the standard sequencer (see the Channel Sequencing Modes section).<br>0: Standard sequencer disabled.<br>1: Standard sequencer enabled. | 0x1 | R/W |
| [6:0] | NUM_SLOTS_AS | Number of Advanced Sequencer Slots Field. Determines the number of slots in a sequence when the advanced sequencer is enabled. The number of slots is equal to NUM_SLOTS_AS + 1. NUM_SLOTS_AS must be set to 0x00 to enable two-cycle command mode or single-cycle command mode (see the Channel Sequencing Modes section). | 0x0 | R/W |

*Autocycle Control Register*

**Address: 0x023, Reset: 0x00, Name: AC_CTRL**
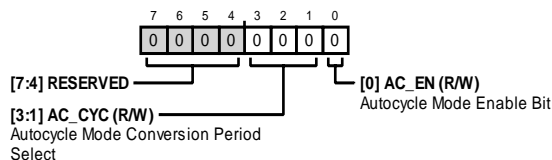


**Table 48. Bit Descriptions for AC_CTRL**

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| [7:4] | RESERVED | Reserved. | 0x0 | R |
| [3:1] | AC_CYC | Autocycle Mode Conversion Period Select. Sets the period of the internal convert-start signal when autocycle mode is enabled (see the Autocycle Mode section). <br> 0x0: Autocycle Conversion Period = 10 µs. <br> 0x1: Autocycle Conversion Period = 20 µs. <br> 0x2: Autocycle Conversion Period = 40 µs. <br> 0x3: Autocycle Conversion Period = 80 µs. <br> 0x4: Autocycle Conversion Period = 100 µs. <br> 0x5: Autocycle Conversion Period = 200 µs. <br> 0x6: Autocycle Conversion Period = 400 µs. <br> 0x7: Autocycle Conversion Period = 800 µs. | 0x0 | R/W |
| 0 | AC_EN | Autocycle Mode Enable Bit. Enables or disables autocycle mode (see the Autocycle Mode section). <br> 0: Autocycle mode disabled. <br> 1: Autocycle mode enabled. | 0x0 | R/W |

### Standard Sequencer Configuration Register

Address: 0x024, Reset: 0x0001, Name: STD_SEQ_CONFIG



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

**[15] IN15_EN (R/W)**
IN15 Standard Sequencer Enable Bit

**[14] IN14_EN (R/W)**
IN14 Standard Sequencer Enable Bit

**[13] IN13_EN (R/W)**
IN13 Standard Sequencer Enable Bit

**[12] IN12_EN (R/W)**
IN12 Standard Sequencer Enable Bit

**[11] IN11_EN (R/W)**
IN11 Standard Sequencer Enable Bit

**[10] IN10_EN (R/W)**
IN10 Standard Sequencer Enable Bit

**[9] IN9_EN (R/W)**
IN9 Standard Sequencer Enable Bit

**[8] IN8_EN (R/W)**
IN8 Standard Sequencer Enable Bit

**[0] IN0_EN (R/W)**
IN0 Standard Sequencer Enable Bit

**[1] IN1_EN (R/W)**
IN1 Standard Sequencer Enable Bit

**[2] IN2_EN (R/W)**
IN2 Standard Sequencer Enable Bit

**[3] IN3_EN (R/W)**
IN3 Standard Sequencer Enable Bit

**[4] IN4_EN (R/W)**
IN4 Standard Sequencer Enable Bit

**[5] IN5_EN (R/W)**
IN5 Standard Sequencer Enable Bit

**[6] IN6_EN (R/W)**
IN6 Standard Sequencer Enable Bit

**[7] IN7_EN (R/W)**
IN7 Standard Sequencer Enable Bit

**Table 49. Bit Descriptions for STD_SEQ_CONFIG**

| Bits | Bit Name | Description | Reset | Access |
|---|---|---|---|---|
| 15 | IN15_EN | IN15 Standard Sequencer Enable Bit. When IN15_EN is set to 1, IN15 is included in the channel sequence when the standard sequencer is enabled (see the Standard Sequencer section). | 0x0 | R/W |
| 14 | IN14_EN | IN14 Standard Sequencer Enable Bit. When IN14_EN is set to 1, IN14 is included in the channel sequence when the standard sequencer is enabled (see the Standard Sequencer section). | 0x0 | R/W |
| 13 | IN13_EN | IN13 Standard Sequencer Enable Bit. When IN13_EN is set to 1, IN13 is included in the channel sequence when the standard sequencer is enabled (see the Standard Sequencer section). | 0x0 | R/W |
| 12 | IN12_EN | IN12 Standard Sequencer Enable Bit. When IN12_EN is set to 1, IN12 is included in the channel sequence when the standard sequencer is enabled (see the Standard Sequencer section). | 0x0 | R/W |
| 11 | IN11_EN | IN11 Standard Sequencer Enable Bit. When IN11_EN is set to 1, IN11 is included in the channel sequence when the standard sequencer is enabled (see the Standard Sequencer section). | 0x0 | R/W |
| 10 | IN10_EN | IN10 Standard Sequencer Enable Bit. When IN10_EN is set to 1, IN10 is included in the channel sequence when the standard sequencer is enabled (see the Standard Sequencer section). | 0x0 | R/W |
| 9 | IN9_EN | IN9 Standard Sequencer Enable Bit. When IN9_EN is set to 1, IN9 is included in the channel sequence when the standard sequencer is enabled (see the Standard Sequencer section). | 0x0 | R/W |
| 8 | IN8_EN | IN8 Standard Sequencer Enable Bit. When IN8_EN is set to 1, IN8 is included in the channel sequence when the standard sequencer is enabled (see the Standard Sequencer section). | 0x0 | R/W |
| 7 | IN7_EN | IN7 Standard Sequencer Enable Bit. When IN7_EN is set to 1, IN7 is included in the channel sequence when the standard sequencer is enabled (see the Standard Sequencer section). | 0x0 | R/W |
| 6 | IN6_EN | IN6 Standard Sequencer Enable Bit. When IN6_EN is set to 1, IN6 is included in the channel sequence when the standard sequencer is enabled (see the Standard Sequencer section). | 0x0 | R/W |
| 5 | IN5_EN | IN5 Standard Sequencer Enable Bit. When IN5_EN is set to 1, IN5 is included in the channel sequence when the standard sequencer is enabled (see the Standard Sequencer section). | 0x0 | R/W |
| 4 | IN4_EN | IN4 Standard Sequencer Enable Bit. When IN4_EN is set to 1, IN4 is included in the channel sequence when the standard sequencer is enabled (see the Standard Sequencer section). | 0x0 | R/W |
| 3 | IN3_EN | IN3 Standard Sequencer Enable Bit. When IN3_EN is set to 1, IN3 is included in the channel sequence when the standard sequencer is enabled (see the Standard Sequencer section). | 0x0 | R/W |
| 2 | IN2_EN | IN2 Standard Sequencer Enable Bit. When IN2_EN is set to 1, IN2 is included in the channel sequence when the standard sequencer is enabled (see the Standard Sequencer section). | 0x0 | R/W |
| 1 | IN1_EN | IN1 Standard Sequencer Enable Bit. When IN1_EN is set to 1, IN1 is included in the channel sequence when the standard sequencer is enabled (see the Standard Sequencer section). | 0x0 | R/W |
| 0 | IN0_EN | IN0 Standard Sequencer Enable Bit. When IN0_EN is set to 1, IN0 is included in the channel sequence when the standard sequencer is enabled (see the Standard Sequencer section). | 0x1 | R/W |

### GPIO Enable Register

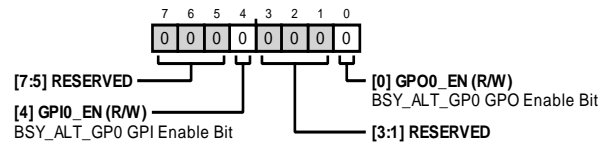**Address: 0x026, Reset: 0x00, Name: GPIO_CTRL**



**Table 50. Bit Descriptions for GPIO_CTRL**

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| [7:5] | RESERVED | Reserved. | 0x0 | R/W |
| 4 | GPI0_EN | BSY_ALT_GP0 GPI Enable Bit. Configures the BSY_ALT_GP0 pin as a general-purpose input if the higher priority functions are disabled (see the General-Purpose Pin section).<br><br>0: General-purpose input function on BSY_ALT_GP0 disabled.<br>1: General-purpose input function on BSY_ALT_GP0 enabled. | 0x0 | R/W |
| [3:1] | RESERVED | Reserved. | 0x0 | R/W |
| 0 | GPO0_EN | BSY_ALT_GP0 GPO Enable Bit. Configures the BSY_ALT_GP0 pin as a general-purpose output if all higher priority functions are disabled (see the General-Purpose Pin section).<br>0: General-purpose output function on BSY_ALT_GP0 disabled.<br>1: General-purpose output function on BSY_ALT_GP0 enabled. | 0x0 | R/W |

### General-Purpose Pin Function Control Register

**Address: 0x027, Reset: 0x00, Name: GP_MODE**
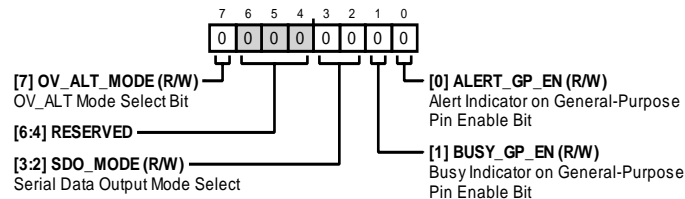


**Table 51. Bit Descriptions for GP_MODE**

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| 7 | OV_ALT_MODE | OV_ALT Mode Select Bit. Configures the OV_ALT bit in the status bits to report the state of the threshold detection alert indicator (see the Status Bits section).<br>0: Does not configure the OV_ALT bit to report the state of the TD_ALERT bit.<br>1: Configures the OV_ALT bit to report the state of the TD_ALERT bit. | 0x0 | R/W |
| 6 | RESERVED | Reserved. | 0x0 | R |
| [5:4] | RESERVED | Reserved. | 0x0 | R/W |
| [3:2] | SDO_MODE | Serial Data Output Mode Select. Selects the serial data output mode.<br>0: Single-SDO mode enabled.<br>1: Dual-SDO mode enabled.<br>01: Single-SDO mode enabled.<br>11: Single-SDO mode enabled. | 0x0 | R/W |
| 1 | BUSY_GP_EN | Busy Indicator on General-Purpose Pin Enable Bit. Enables or disables the busy indicator on the BSY_ALT_GP0 pin if all higher priority functions are disabled (see the General-Purpose Pin section).<br>0: Busy indicator on general-purpose pin function disabled.<br>1: Busy indicator on general-purpose pin function enabled. | 0x0 | R/W |
| 0 | ALERT_GP_EN | Alert Indicator on General-Purpose Pin Enable Bit. Enables or disables the alert indicator on the BSY_ALT_GP0 pin if all higher priority functions are disabled (see the General-Purpose Pin section).<br>0: Alert indicator on general-purpose pin function disabled.<br>1: Alert indicator on general-purpose pin function enabled. | 0x0 | R/W |

*GPIO State Register*
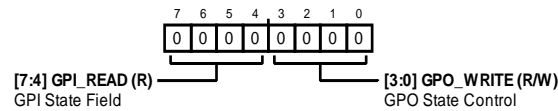
**Address: 0x028, Reset: 0x00, Name: GPIO_STATE**

```
                            7  6  5  4  3  2  1  0
                           [0][0][0][0][0][0][0][0]
    [7:4] GPI_READ (R) ──────┘           └────── [3:0] GPO_WRITE (R/W)
    GPI State Field                               GPO State Control
```

**Table 52. Bit Descriptions for GPIO_STATE**

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| [7:5] | RESERVED | Reserved. | 0x0 | R |
| 4 | GPI_READ | GPI State Indicator. Displays the state of the BSY_ALT_GP0 pin when configured as a general-purpose input. | 0x0 | R |
| [3:1] | RESERVED | Reserved. | 0x0 | R/W |
| 0 | GPO_WRITE | GPO State Control. Sets the state of the BSY_ALT_GP0 pin when configured as a general purpose output. 0000: BSY_ALT_GP0 is driven to logic low voltage. 0001: BSY_ALT_GP0 is driven to logic high voltage. | 0x0 | R/W |

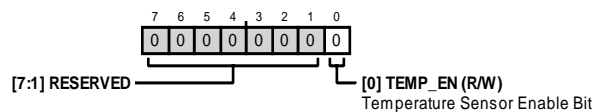*Temperature Sensor Control Register*

**Address: 0x029, Reset: 0x00, Name: TEMP_CTRL**

```
                            7  6  5  4  3  2  1  0
                           [0][0][0][0][0][0][0][0]
    [7:1] RESERVED ──────────┘              └────── [0] TEMP_EN (R/W)
                                                    Temperature Sensor Enable Bit
```

**Table 53. Bit Descriptions for TEMP_CTRL**

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| [7:1] | RESERVED | Reserved. | 0x0 | R |
| 0 | TEMP_EN | Temperature Sensor Enable Bit. Enables or disables the temperature sensor in the channel sequence when the standard sequencer or advanced sequencer are enabled (see the Temperature Sensor section). 0: Temperature sensor not included in the channel sequence. 1: Temperature sensor included in the channel sequence. | 0x0 | R/W |

*Analog Input Settings Configuration Register*

**Address: 0x030 - 0x03F (Increments of 0x001), Reset: 0x08, Name: CONFIG_INn**

```
                            7  6  5  4  3  2  1  0
                           [0][0][0][0][1][0][0][0]
    [7] TD_EN (R/W) ─────────┘                 └──── [1:0] OSR (R/W)
    INn Threshold Detection Enable Bit                INn Oversampling Ratio Select
    [6] IN_MODE (R/W) ───────┘              └──── [2] RESERVED
    INn Polarity Mode Select Bit
    [5:4] IN_PAIR (R/W) ─────┘          └──── [3] AINHIZ_EN (R/W)
    INn Pin-Pairing Select                      INn Analog Input High-Z Mode Enable
                                                Bit
```
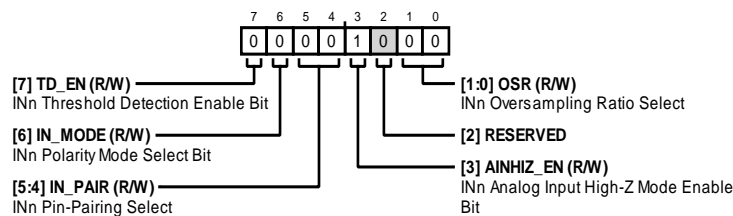
**Table 54. Bit Descriptions for CONFIG_INn**

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| 7 | TD_EN | INn Threshold Detection Enable Bit. When the standard sequencer is enabled, the TD_EN bit in the CONFIG_IN0 register enables or disables threshold detection for IN0 thru IN15. When the advanced sequencer is enabled, the TD_EN bit in each CONFIG_INn register enables or disables threshold detection only for its corresponding INn analog input. The HI_INn and LO_INn alert indicator bits are active when threshold detection is enabled on their corresponding INn analog input (see the Threshold Detection and Alert Indicators section). 0: Disables threshold detection for INn. 1: Enables threshold detection for INn. | 0x0 | R/W |

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| 6 | IN_MODE | INn Polarity Mode Select Bit. Selects the polarity mode for the corresponding INn analog input (see the Channel Configuration Options section). Unlike the other control bits in the CONFIG_INn registers, the polarity mode for each INn analog input is always set by the IN_MODE bit in its corresponding CONFIG_INn register, regardless of the channel sequencing mode.<br>0: Selects unipolar mode for INn.<br>1: Selects pseudo-bipolar mode for INn. | 0x0 | R/W |
| [5:4] | IN_PAIR | INn Pin-Pairing Select. Selects the pin-pairing option for the corresponding INn analog input (see the Channel Configuration Options section). When the standard sequencer is enabled, the IN_PAIR field in the CONFIG_IN0 register sets the pin-pairing option for IN0 thru IN15. When the advanced sequencer is enabled, the IN_PAIR bit in each CONFIG_INn register sets the pin-pairing option only for its corresponding INn analog input.<br>0x0: INn paired with REFGND.<br>0x1: INn paired with COM.<br>0x2: Even- and Odd-Input Paired.<br>0x3: Invalid. | 0x0 | R/W |
| 3 | AINHIZ_EN | INn Analog Input High-Z Mode Enable Bit. When the standard sequencer is enabled, the AINHIZ_EN bit in the CONFIG_IN0 register enables or disables analog input high-Z mode for IN0 thru IN15. When the advanced sequencer is enabled, the AINHIZ_EN bit in each CONFIG_INn register enables or disables analog input high-Z mode only for its corresponding INn analog input. See the Analog Input High-Z Mode section.<br>0: Disables analog input high-Z mode for INn.<br>1: Enables analog input high-Z mode for INn. | 0x1 | R/W |
| 2 | RESERVED | Reserved. | 0x0 | R |
| [1:0] | OSR | INn Oversampling Ratio Select. When the standard sequencer is enabled, the OSR field in the CONFIG_IN0 register sets the oversampling ratio for IN0 thru IN15. When the advanced sequencer is enabled, the OSR field in each CONFIG_INn register sets the oversampling ratio only for its corresponding INn analog input. Set the OSR fields in all CONFIG_INn registers to 0x0 when two-cycle command mode or single-cycle command mode are enabled. See the Oversampling and Decimation section.<br>0x0: INn oversampling ratio = 1 (no oversampling). Output code resolution is 16 bits.<br>0x1: INn oversampling ratio = 4. Output code result resolution increases to 17 bits.<br>0x2: INn oversampling ratio = 16. Output code result resolution increases to 18 bits.<br>0x3: INn oversampling ratio = 64. Output code result resolution increases to 19 bits. | 0x0 | R/W |

## Upper Threshold Value Register

**Address: 0x040 - 0x05E (Increments of 0x002), Reset: 0x07FF, Name: UPPER_INn**

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

[15:12] RESERVED

[11:0] UPPER (R/W)
INn Upper Threshold Value Setting
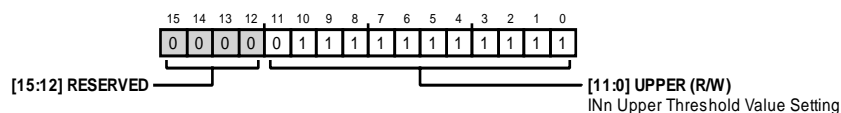
**Table 55. Bit Descriptions for UPPER_INn**

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| [15:12] | RESERVED | Reserved. | 0x0 | R |
| [11:0] | UPPER | INn Upper Threshold Value Setting. Determines the upper threshold value for the corresponding INn analog input (see the Threshold Detection and Alert Indicators section). The value in the UPPER fields corresponds to the 12 MSBs of the ADC result. | 0x7FF | R/W |

### Lower Threshold Value Register

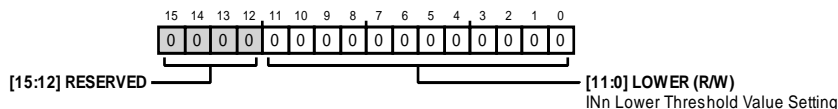**Address: 0x060 - 0x07E (Increments of 0x002), Reset: 0x0000, Name: LOWER_INn**

```
15 14 13 12 | 11 10  9  8  7  6  5  4  3  2  1  0
 0  0  0  0 |  0  0  0  0  0  0  0  0  0  0  0  0
```
[15:12] RESERVED ——————     [11:0] LOWER (R/W)
                                            INn Lower Threshold Value Setting

**Table 56. Bit Descriptions for LOWER_INn**

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| [15:12] | RESERVED | Reserved. | 0x0 | R |
| [11:0] | LOWER | INn Lower Threshold Value Setting. Determines the lower threshold value for the corresponding INn analog input (see the Threshold Detection and Alert Indicators section). The value in the LOWER fields corresponds to the 12 MSBs in the ADC result. | 0x0 | R/W |

### Hysteresis Setting Register

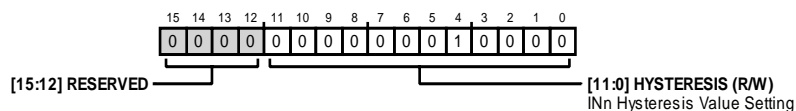**Address: 0x080 - 0x09E (Increments of 0x002), Reset: 0x0010, Name: HYST_INn**

```
15 14 13 12 | 11 10  9  8  7  6  5  4  3  2  1  0
 0  0  0  0 |  0  0  0  0  0  0  0  1  0  0  0  0
```
[15:12] RESERVED ——————     [11:0] HYSTERESIS (R/W)
                                            INn Hysteresis Value Setting

**Table 57. Bit Descriptions for HYST_INn**

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| [15:12] | RESERVED | Reserved. | 0x0 | R |
| [11:0] | HYSTERESIS | INn Hysteresis Value Setting. Determines the hysteresis value for the corresponding INn analog input (see the Threshold Detection and Alert Indicators section). The value in the HYSTERESIS fields corresponds to the 12 MSBs in the ADC result. | 0x10 | R/W |

### INn Offset Correction Register

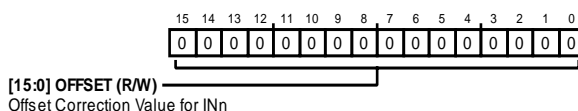**Address: 0x0A0 - 0x0BE (Increments of 0x002), Reset: 0x0000, Name: OFFSET_INn**

```
15 14 13 12 11 10  9  8  7  6  5  4  3  2  1  0
 0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
```
[15:0] OFFSET (R/W)
Offset Correction Value for INn

**Table 58. Bit Descriptions for OFFSET_INn**

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| [15:0] | OFFSET | Offset Correction Value for INn. Sets the offset correction applied to results from the INn channel. See the Offset and Gain Correction section for a detailed description of offset correction. | 0x0 | R/W |

### INn Gain Correction Register

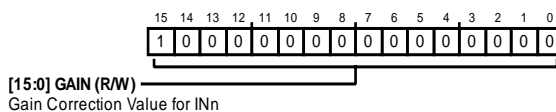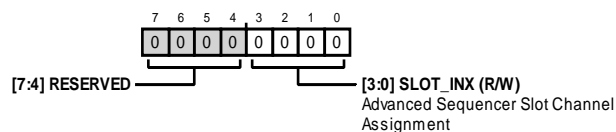**Address: 0x0C0 - 0x0DE (Increments of 0x002), Reset: 0x8000, Name: GAIN_INn**

```
15 14 13 12 11 10  9  8  7  6  5  4  3  2  1  0
 1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
```
[15:0] GAIN (R/W)
Gain Correction Value for INn

**Table 59. Bit Descriptions for GAIN_INn**

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| [15:0] | GAIN | Gain Correction Value for INn. Sets the gain correction applied to results from the INn channel. See the Offset and Gain Correction section for a detailed description of gain correction. | 0x8000 | R/W |

### Advanced Sequencer Slot Register

**Address: 0x100 - 0x17F (Increments of 0x001), Reset: 0x00, Name: AS_SLOTn**

```
      7   6   5   4   3   2   1   0
    ┌───┬───┬───┬───┬───┬───┬───┬───┐
    │ 0 │ 0 │ 0 │ 0 │ 0 │ 0 │ 0 │ 0 │
    └───┴───┴───┴───┴───┴───┴───┴───┘
[7:4] RESERVED                [3:0] SLOT_INX (R/W)
                              Advanced Sequencer Slot Channel
                              Assignment
```

**Table 60. Bit Descriptions for AS_SLOTn**

| Bits | Bit Name | Description | Reset | Access |
|------|----------|-------------|-------|--------|
| [7:4] | RESERVED | Reserved. | 0x0 | R |
| [3:0] | SLOT_INX | Advanced Sequencer Slot Channel Assignment. Determines which of the 16 analog inputs are assigned to slot n (see the Advanced Sequencer section).<br>0x0: IN0.<br>0x1: IN1.<br>0x2: IN2.<br>0x3: IN3.<br>0x4: IN4.<br>0x5: IN5.<br>0x6: IN6.<br>0x7: IN7.<br>0x8: IN8.<br>0x9: IN9.<br>0xA: IN10.<br>0xB: IN11.<br>0xC: IN12.<br>0xD: IN13.<br>0xE: IN14.<br>0xF: IN15. | 0x0 | R/W |

## OUTLINE DIMENSIONS
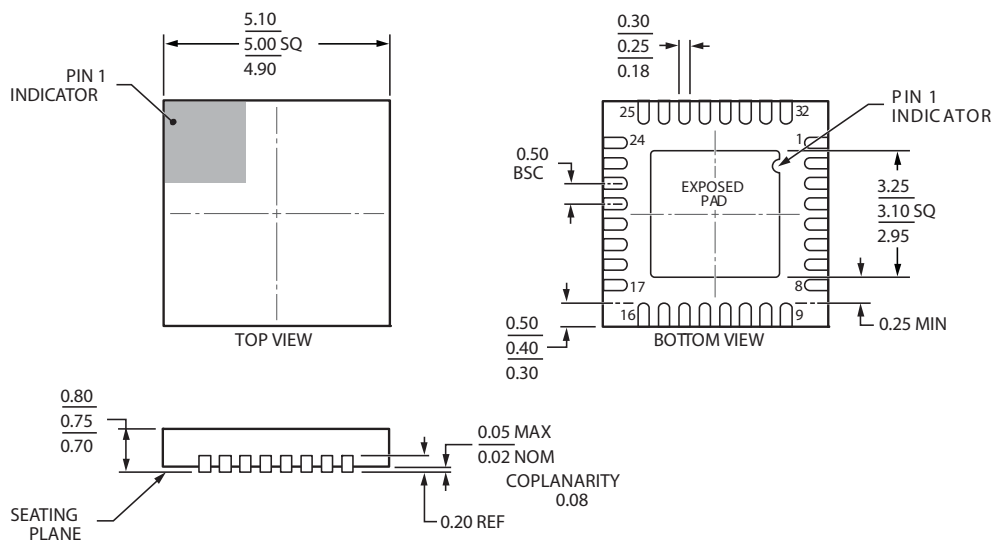


COMPLIANT TO JEDEC STANDARDS MO-220-WHHD-2

*Figure 121. 32-Lead Lead Frame Chip Scale Package [LFCSP_WQ]*
*5 x 5 mm Body, Very Very Thin Quad*
*(CP-32-7)*
*Dimensions shown in millimeters*

**ANALOG DEVICES**

w w w . a n a l o g . c o m