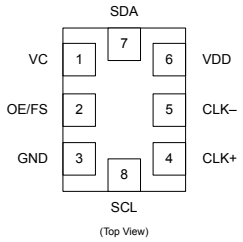
# Ultra Series™ Crystal Oscillator (VCXO) Si569 Data Sheet

## Ultra Low Jitter I2C Programmable VCXO (100 fs), 0.2 to 3000 MHz

The Si569 Ultra Series™ voltage-controlled crystal oscillator utilizes Silicon Laboratories' advanced 4th generation DSPLL® technology to provide an ultra-low jitter, low phase noise clock at any output frequency. The device is user-programmed via simple I2C commands to provide any frequency from 0.2 to 3000 MHz with <1 ppb resolution and maintains exceptionally low jitter for both integer and fractional frequencies across its operating range. On-chip power supply filtering provides industry-leading power supply noise rejection, simplifying the task of generating low jitter clocks in noisy systems that use switched-mode power supplies. Unlike a traditional XO, where a different crystal is required for each output frequency, the Si569 uses one simple crystal and a DSPLL IC-based approach to provide the desired output frequency. The Si569 is factory-configurable for a wide variety of user specifications, including startup frequency, I2C address, output format, and OE pin location/polarity. Specific configurations are factory-programmed at time of shipment, eliminating long lead times associated with custom oscillators.

Pin Assignments



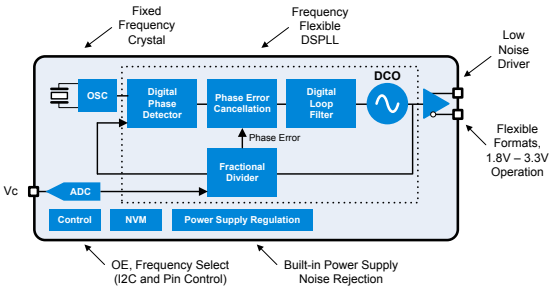

**KEY FEATURES**

- I2C programmable to any frequency from 0.2 to 3000 MHz with < 1 ppb resolution
- Ultra low jitter: 100 fs RMS Typ (12 kHz – 20 MHz)
- Configure up to 2 pin-selectable startup frequencies
- I2C interface supports 100 kbps, 400 kbps, and 1 Mbps (Fast Mode Plus)
- Excellent PSRR and supply noise immunity: –80 dBc Typ
- Programmable Kv (ppm/V) simplifies development
- 3.3 V, 2.5 V and 1.8 V $V_{DD}$ supply operation from the same part number
- LVPECL, LVDS, CML, HCSL, CMOS, and Dual CMOS output options
- 3.2x5, 5x7 mm package footprints
- Samples available with 1-2 week lead times

**APPLICATIONS**

- 100G/200G/400G OTN, coherent optics, PAM4
- 3G-SDI/12G-SDI/24G-SDI broadcast video
- Servers, switches, storage, search acceleration
- FPGA/ASIC clocking

| Pin # | Descriptions |
|---|---|
| 1 | VC = Voltage Control Pin |
| 2 | *Selectable via ordering option*<br>OE = Output enable; FS = Frequency Select |
| 3 | GND = Ground |
| 4 | CLK+ = Clock output |
| 5 | CLK- = Complementary clock output. Not used for CMOS. |
| 6 | VDD = Power supply |
| 7 | SDA = I2C Serial Data |
| 8 | SCL = I2C Serial Clock |



# 1. Ordering Guide

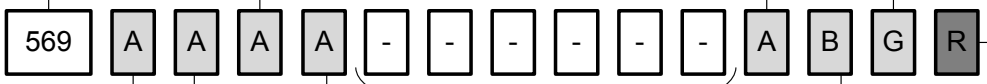
The Si569 XO supports a variety of options including startup frequency, output format, and control voltage tuning slope, as shown in the chart below. Specific device configurations are programmed into the part at time of shipment, and samples are available in 1-2 weeks. Silicon Laboratories provides an online part number configuration utility to simplify this process. Refer to www.silabs.com/oscillators to access this tool and for further ordering instructions.

| VCXO Series | Description |
|---|---|
| 569 | I2C Programmable |

| Code | OE Pin | OE Polarity | FS (Dual) |
|---|---|---|---|
| A | Pin 2 | Active High | -- |
| B | Pin 2 | Active Low | -- |
| C | -- | -- | Pin 2 |

| | Package |
|---|---|
| A | 5x7 mm |
| B | 3.2x5 mm |

| | Temperature Grade |
|---|---|
| G | -40 to 85 °C |




| Signal Format | VDD Range | Order Option |
|---|---|---|
| LVPECL | 2.5, 3.3 V | A |
| LVDS | 1.8, 2.5, 3.3 V | B |
| CMOS | 1.8, 2.5, 3.3 V | C |
| CML | 1.8, 2.5, 3.3 V | D |
| HCSL | 1.8, 2.5, 3.3 V | E |
| Dual CMOS (In-Phase) | 1.8, 2.5, 3.3 V | F |
| Dual CMOS (Complementary) | 1.8, 2.5, 3.3 V | G |
| Custom[1] | 1.8, 2.5, 3.3 V | X |

| Code | Supported Frequency Range |
|---|---|
| A | 0.2-3000 MHz |
| B | 0.2-1500 MHz |
| C | 0.2-800 MHz |
| D | 0.2-325 MHz (CMOS available to 250 MHz) |

| Code | Reel |
|---|---|
| R | Tape and Reel |
| <Blank> | Coil Tape |

| | Temperature Stability = ± 20 ppm | | | |
|---|---|---|---|---|
| | Vc Tuning Slope Kv (ppm/V) | Min APR (± ppm) at VDD [3] | | |
| | | 3.3V | 2.5V | 1.8V |
| A | 60 | 20 | -- | -- |
| B | 75 | 40 | 20 | -- |
| C | 105 | 70 | 40 | 20 |
| D | 150 | 115 | 75 | 45 |
| E | 180 | 145 | 100 | 65 |
| F | 225 | 190 | 135 | 85 |

| Frequency Code [2] | Description |
|---|---|
| xxxxxx | The Si569 supports up to two user-defined startup frequencies in the range selected by the Supported Frequency Range code. A user-defined 7-bit I2C address is supported. Each unique startup configuration and I2C address combination is assigned a 6-digit code. |

**Notes:**

1. Contact Silicon Labs for non-standard configurations.
2. Create custom part numbers at www.silabs.com/oscillators.
3. Min Absolute Pull Range (APR) includes temp stability, initial accuracy, load pulling, VDD variation, and 20 year aging at 70 °C.
   a. For best jitter and phase noise performance, always choose the smallest Kv that meets the application's minimum APR requirements. Unlike SAW-based solutions which require higher Kv values to account for their higher temperature dependence, the Si56x series provides lower Kv options to minimize noise coupling and jitter in real-world PLL designs.
   b. Absolute Pull Range (APR) is the ability of a VCXO to track a signal over the product lifetime. A VCXO with an APR of ±20 ppm is able to lock to a clock with a ±20 ppm stability over 20 years over all operating conditions.
   c. APR (±) = (0.5 x VDD x tuning slope) - (initial accuracy + temp stability + load pulling + VDD variation + aging).
   d. Minimum APR values noted above include absolute worst case values for all parameters.
   e. See application note, "AN266: VCXO Tuning Slope (Kv), Stability, and Absolute Pull Range (APR)" for more information.

## 1.1 Technical Support

| | |
|---|---|
| Frequently Asked Questions (FAQ) | www.silabs.com/Si569-FAQ |
| Oscillator Phase Noise Lookup Utility | www.silabs.com/oscillator-phase-noise-lookup |
| Quality and Reliability | www.silabs.com/quality |
| Development Kits | www.silabs.com/oscillator-tools |

# 2. Electrical Specifications

**Table 2.1. Electrical Specifications**

$V_{DD}$ = 1.8 V, 2.5 or 3.3 V ± 5%, $T_A$ = –40 to 85 ºC

| Parameter | Symbol | Test Condition/Comment | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| Temperature Range | $T_A$ | | –40 | — | 85 | ºC |
| Frequency Range | $F_{CLK}$ | LVPECL, LVDS, CML | 0.2 | — | 3000 | MHz |
| | | HCSL | 0.2 | — | 400 | MHz |
| | | CMOS, Dual CMOS | 0.2 | — | 250 | MHz |
| Supply Voltage | $V_{DD}$ | 3.3 V | 3.135 | 3.3 | 3.465 | V |
| | | 2.5 V | 2.375 | 2.5 | 2.625 | V |
| | | 1.8 V | 1.71 | 1.8 | 1.89 | V |
| Supply Current | $I_{DD}$ | LVPECL (output enabled) | — | 120 | 170 | mA |
| | | LVDS/CML (output enabled) | — | 100 | 167 | mA |
| | | HCSL (output enabled) | — | 95 | 140 | mA |
| | | CMOS (output enabled) | — | 95 | 145 | mA |
| | | Dual CMOS (output enabled) | — | 105 | 155 | mA |
| | | Tristate Hi-Z (output disabled) | — | 83 | — | mA |
| Temperature Stability[1] | | -40 to 85 °C | –20 | — | 20 | ppm |
| Rise/Fall Time (20% to 80% $V_{PP}$) | $T_R/T_F$ | LVPECL/LVDS/CML | — | — | 350 | ps |
| | | CMOS / Dual CMOS ($C_L$ = 5 pF) | — | 0.5 | 1.5 | ns |
| | | HCSL, $F_{CLK}$ >50 MHz | — | — | 450 | ps |
| Duty Cycle | $D_C$ | All formats | 45 | — | 55 | % |
| Output Enable (OE), Frequency Select (FS)[2] | $V_{IH}$ | | 0.7 × $V_{DD}$ | — | — | V |
| | $V_{IL}$ | | — | — | 0.3 × $V_{DD}$ | V |
| | $T_D$ | Output Disable Time, $F_{CLK}$ >10 MHz | — | — | 3 | µs |
| | $T_E$ | Output Enable Time, $F_{CLK}$ >10 MHz | — | — | 20 | µs |
| | $T_{FS}$ | Settling Time after FS Change | — | — | 10 | ms |
| Powerup Time | $t_{OSC}$ | Time from 0.9 × $V_{DD}$ until output frequency ($F_{CLK}$) within spec | — | — | 10 | ms |
| LVPECL Output Option[3] | $V_{OC}$ | Mid-level | $V_{DD}$ – 1.42 | — | $V_{DD}$ – 1.25 | V |
| | $V_O$ | Swing (diff, $F_{CLK} \leq$ 1.5 GHz) | 1.1 | — | 1.9 | $V_{PP}$ |
| | | Swing (diff, $F_{CLK}$ > 1.5 GHz)[6] | 0.55 | — | 1.7 | $V_{PP}$ |

| Parameter | Symbol | Test Condition/Comment | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| LVDS Output Option[4] | $V_{OC}$ | Mid-level (2.5 V, 3.3 V VDD) | 1.125 | 1.20 | 1.275 | V |
| | | Mid-level (1.8 V VDD) | 0.8 | 0.9 | 1.0 | V |
| | $V_O$ | Swing (diff, $F_{CLK} \leq 1.5$ GHz) | 0.5 | 0.7 | 0.9 | $V_{PP}$ |
| | | Swing (diff, $F_{CLK} > 1.5$ GHz) [6] | 0.25 | 0.5 | 0.8 | $V_{PP}$ |
| | | Swing (diff, $F_{CLK} \leq 1.6$ GHz) [7] | 0.6 | 0.8 | 1.0 | $V_{PP}$ |
| HCSL Output Option[5] | $V_{OH}$ | Output voltage high | 660 | 800 | 850 | mV |
| | $V_{OL}$ | Output voltage low | –150 | 0 | 150 | mV |
| | $V_C$ | Crossing voltage | 250 | 410 | 550 | mV |
| CML Output Option (AC-Coupled) | $V_O$ | Swing (diff, $F_{CLK} \leq 1.5$ GHz) | 0.6 | 0.8 | 1.0 | $V_{PP}$ |
| | | Swing (diff, $F_{CLK} > 1.5$ GHz)[6] | 0.3 | 0.55 | 0.9 | $V_{PP}$ |
| CMOS Output Option | $V_{OH}$ | $I_{OH}$ = 8/6/4 mA for 3.3/2.5/1.8V VDD | 0.85 × $V_{DD}$ | — | — | V |
| | $V_{OL}$ | $I_{OL}$ = 8/6/4 mA for 3.3/2.5/1.8V VDD | — | — | 0.15 × $V_{DD}$ | V |

**Notes:**

1. Min APR includes ±20 ppm temperature stability, initial accuracy, load pulling, VDD variation, and aging for 20 yrs at 70 °C.
2. OE includes a 50 kΩ pull-up to VDD for OE active high, or includes a 50 kΩ pull-down to GND for OE active low. FS pin includes a 50 kΩ pull-up to VDD.
3. $R_{term}$ = 50 Ω to $V_{DD}$ – 2.0 V (see Figure 4.1).
4. $R_{term}$ = 100 Ω (differential) (see Figure 4.2).
5. $R_{term}$ = 50 Ω to GND (see Figure 4.2).
6. Refer to the figure below for Typical Clock Output Swing Amplitudes vs Frequency.
7. High drive LVDS swing is supported when following the method shown in section 5.8 Configuring High Drive LVDS Swing.

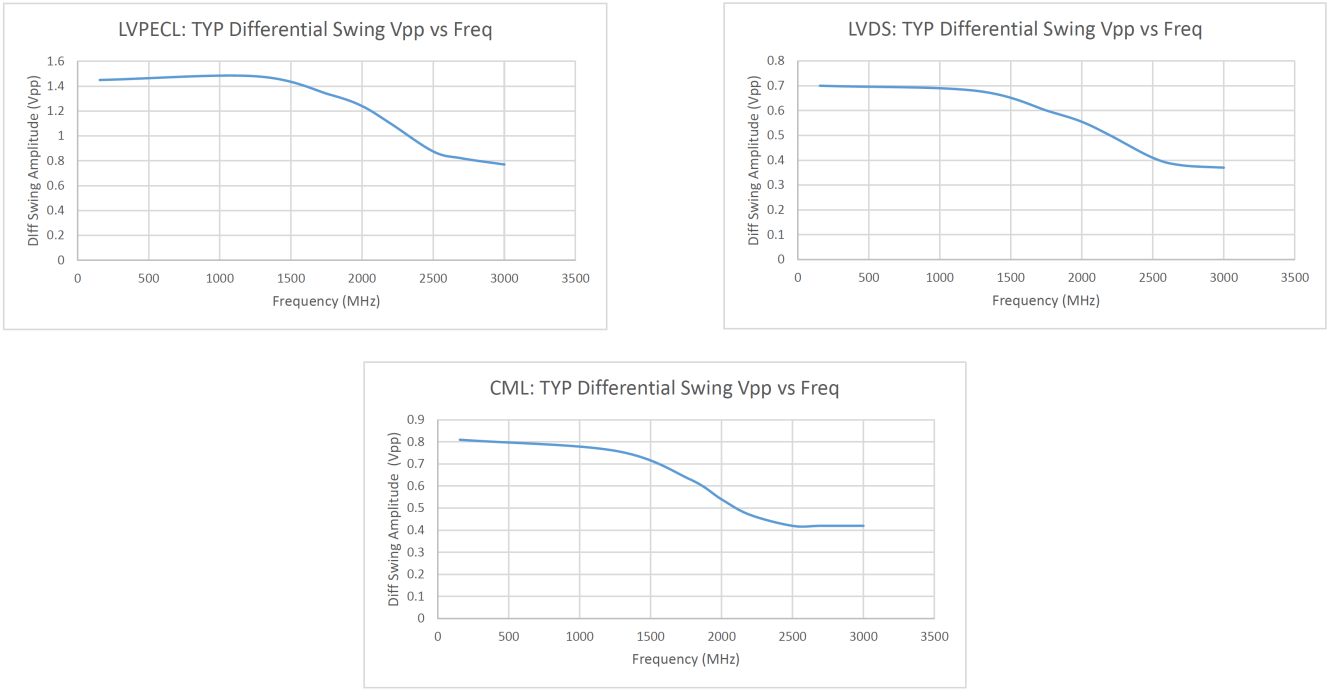



**Figure 2.1. Typical Clock Output Swing Amplitudes vs. Frequency**

**Table 2.2. I2C Characteristics**

$V_{DD}$ = 1.8, 2.5, or 3.3 V ± 5%, $T_A$ = –40 to 85 ºC

| Parameter | Symbol | Test Condition/Comment | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| SDA, SCL Input Voltage High | $V_{IH}$ | | 0.70 x $V_{DD}$ | — | — | V |
| SDA, SCL Input Voltage Low | $V_{IL}$ | | — | — | 0.30 x $V_{DD}$ | V |
| Frequency Reprogramming Resolution | $M_{RES}$ | | — | 0.004 | — | ppb |
| Frequency Range for Small Frequency Change (Continuous Glitchless Output) | | From center frequency | -950 | — | +950 | ppm |
| Settling Time for Small Frequency Change | | < ±950 ppm from center frequency | — | — | 100 | µs |
| Settling Time for Large Frequency Change (Output Squelched during Frequency Transition) | | > ±950 ppm from center frequency | — | — | 10 | ms |

**Table 2.3. $V_C$ Control Voltage Input**

$V_{DD}$ = 1.8, 2.5 or 3.3 V ± 5%, $T_A$ = –40 to 85 ºC

| Parameter | Symbol | Test Condition | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| Control Voltage Range | $V_C$ | | 0.1 x VDD | VDD/2 | 0.9 x VDD | V |
| Control Voltage Tuning Slope (Vc = 10% VDD to 90% VDD) | Kv | Positive slope, ordering option | 60, 75, 105, 150, 180, 225 | | | ppm/V |
| Kv Variation | Kv_var | | — | — | ±10 | % |
| Control Voltage Linearity | LVC | Best Straight Line fit | –1.5 | ±0.5 | +1.5 | % |
| Modulation Bandwidth | BW | | — | 10 | — | kHz |
| Vc Input Impedance | ZVC | | 500 | — | — | kΩ |

**Table 2.4. Clock Output Phase Jitter and PSRR**

$V_{DD}$ = 1.8 V, 2.5 or 3.3 V ± 5%, $T_A$ = –40 to 85 ºC

| Parameter | Symbol | Test Condition/Comment | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| Phase Jitter (RMS, 12 kHz - 20 MHz)[1] All Differential Formats, $F_{CLK}$ ≥ 200 MHz | ϕJ | Kv = 60 ppm/V | — | 100 | 150 | fs |
| | | Kv = 75 ppm/V | — | 103 | — | fs |
| | | Kv = 105 ppm/V | — | 110 | — | fs |
| | | Kv = 150 ppm/V | — | 123 | — | fs |
| | | Kv = 180 ppm/V | — | 132 | — | fs |
| | | Kv = 225 ppm/V | — | 150 | — | fs |
| Phase Jitter (RMS, 12 kHz - 20 MHz) [1] All Diff Formats, 100 MHz ≤ $F_{CLK}$ < 200 MHz | ϕJ | Kv = 60 ppm/V | — | 115 | 170 | fs |
| | | Kv = 75 ppm/V | — | 118 | — | fs |
| | | Kv = 105 ppm/V | — | 125 | — | fs |
| | | Kv = 150 ppm/V | — | 138 | — | fs |
| | | Kv = 180 ppm/V | — | 147 | — | fs |
| | | Kv = 225 ppm/V | — | 165 | — | fs |
| Phase Jitter (RMS, 12 kHz - 20 MHz)[1] LVDS, $F_{CLK}$ = 156.25 MHz | ϕJ | Kv = 60 ppm/V | — | 110 | 130 | fs |
| | | Kv = 75 ppm/V | — | 113 | — | fs |
| | | Kv = 105 ppm/V | — | 120 | — | fs |
| | | Kv = 150 ppm/V | — | 133 | — | fs |
| | | Kv = 180 ppm/V | — | 142 | — | fs |
| | | Kv = 225 ppm/V | — | 160 | — | fs |
| Phase Jitter (RMS, 12 kHz - 20 MHz)[1] CMOS / Dual CMOS Formats | ϕJ | 10 MHz ≤ $F_{CLK}$ < 250 MHz | — | 200 | — | fs |
| Spurs Induced by External Power Supply Noise, 50 mVpp Ripple. LVDS 156.25 MHz Output | PSRR | 100 kHz sine wave | | -83 | | dBc |
| | | 200 kHz sine wave | | -83 | | |
| | | 500 kHz sine wave | | -82 | | |
| | | 1 MHz sine wave | | -85 | | |

**Note:**
1. Jitter inclusive of any spurs.

**Table 2.5. 3.2 x 5 mm Clock Output Phase Noise (Typical)**

| Offset Frequency (f) | 156.25 MHz LVDS | 200 MHz LVDS | 644.53125 MHz LVDS | Unit |
|---|---|---|---|---|
| 100 Hz | –73 | –71 | –60 | |
| 1 kHz | –102 | –102 | –93 | |
| 10 kHz | –130 | –128 | –118 | |
| 100 kHz | –141 | –139 | –129 | dBc/Hz |
| 1 MHz | –150 | –148 | –138 | |
| 10 MHz | –159 | –160 | –153 | |
| 20 MHz | –160 | –162 | –154 | |
| **Offset Frequency (f)** | **156.25 MHz LVPECL** | **200 MHz LVPECL** | **644.53125 MHz LVPECL** | **Unit** |
| 100 Hz | –72 | –71 | –60 | |
| 1 kHz | –103 | –101 | –92 | |
| 10 kHz | –130 | –127 | –117 | |
| 100 kHz | –142 | –139 | –129 | dBc/Hz |
| 1 MHz | –150 | –148 | –138 | |
| 10 MHz | –160 | –162 | –154 | |
| 20 MHz | –161 | –162 | –156 | |

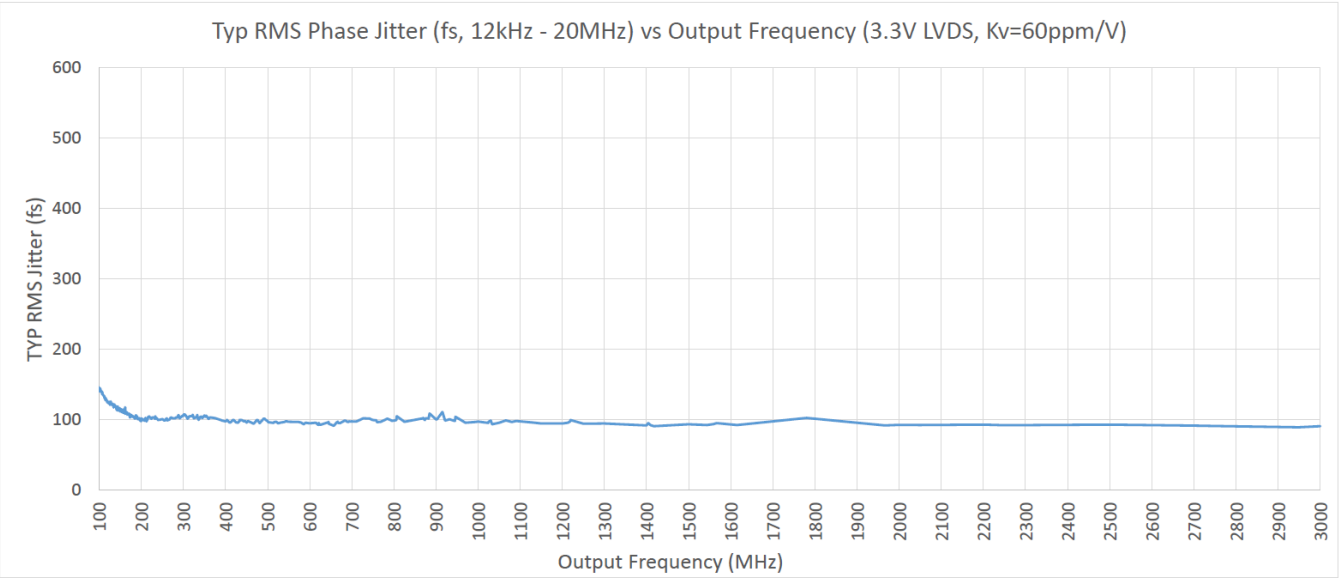



**Figure 2.2. Phase Jitter vs. Output Frequency**

Phase jitter measured with Agilent E5052 using a differential-to-single ended converter (balun or buffer). Measurements collected for >700 commonly used frequencies. Phase noise plots for specific frequencies are available using our free, online Oscillator Phase Noise Lookup Tool at www.silabs.com/oscillators.

**Table 2.6. Environmental Compliance and Package Information**

| Parameter | Test Condition |
|---|---|
| Mechanical Shock | MIL-STD-883, Method 2002 |
| Mechanical Vibration | MIL-STD-883, Method 2007 |
| Solderability | MIL-STD-883, Method 2003 |
| Gross and Fine Leak | MIL-STD-883, Method 1014 |
| Resistance to Solder Heat | MIL-STD-883, Method 2036 |
| Moisture Sensitivity Level (MSL) | 1 |
| Contact Pads | Gold over Nickel |

**Note:**

1. For additional product information not listed in the data sheet (e.g. RoHS Certifications, MDDS data, qualification data, REACH Declarations, ECCN codes, etc.), refer to our "Corporate Request For Information" portal found here: www.silabs.com/support/quality/Pages/RoHSInformation.aspx.

**Table 2.7. Thermal Conditions**

| Package | Parameter | Symbol | Test Condition | Value | Unit |
|---|---|---|---|---|---|
| 3.2 × 5 mm 8-pin CLCC | Thermal Resistance Junction to Ambient | $\Theta_{JA}$ | Still Air, 85 ºC | 79.1 | ºC/W |
| | Thermal Resistance Junction to Board | $\Theta_{JB}$ | Still Air, 85 ºC | 49.6 | ºC/W |
| | Max Junction Temperature | $T_J$ | Still Air, 85 ºC | 125 | ºC |
| 5 × 7 mm 8-pin CLCC | Thermal Resistance Junction to Ambient | $\Theta_{JA}$ | Still Air, 85 ºC | 67.1 | ºC/W |
| | Thermal Resistance Junction to Board | $\Theta_{JB}$ | Still Air, 85 ºC | 51.7 | ºC/W |
| | Max Junction Temperature | $T_J$ | Still Air, 85 ºC | 125 | ºC |

**Table 2.8. Absolute Maximum Ratings[1]**

| Parameter | Symbol | Rating | Unit |
|---|---|---|---|
| Maximum Operating Temp. | $T_{AMAX}$ | 95 | ºC |
| Storage Temperature | $T_S$ | –55 to 125 | ºC |
| Supply Voltage | $V_{DD}$ | –0.5 to 3.8 | ºC |
| Input Voltage | $V_{IN}$ | –0.5 to $V_{DD}$ + 0.3 | V |
| ESD HBM (JESD22-A114) | HBM | 2.0 | kV |
| Solder Temperature[2] | $T_{PEAK}$ | 260 | ºC |
| Solder Time at $T_{PEAK}$[2] | $T_P$ | 20–40 | sec |

**Notes:**

1. Stresses beyond those listed in this table may cause permanent damage to the device. Functional operation specification compliance is not implied at these conditions. Exposure to maximum rating conditions for extended periods may affect device reliability.
2. The device is compliant with JEDEC J-STD-020.

## 3. Dual CMOS Buffer

Dual CMOS output format ordering options support either complementary or in-phase signals for two identical frequency outputs. This feature enables replacement of multiple VCXOs with a single Si569 device.

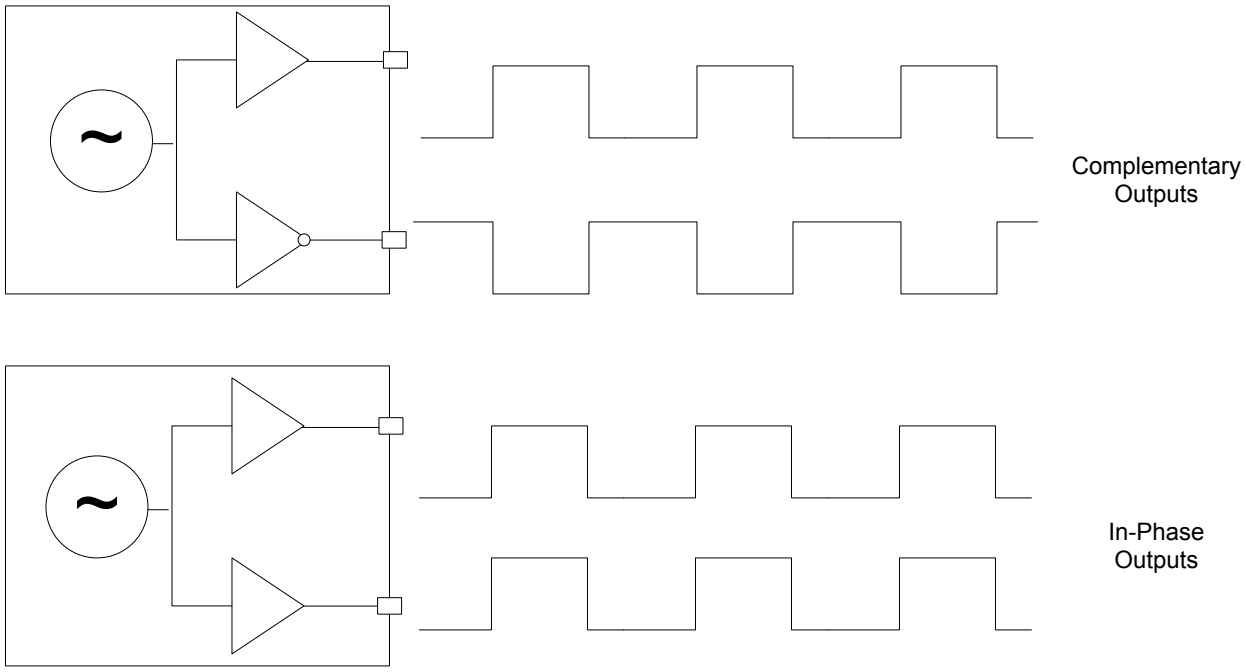



**Figure 3.1. Integrated 1:2 CMOS Buffer Supports Complementary or In-Phase Outputs**

## 4. Recommended Output Terminations

The output drivers support both AC-coupled and DC-coupled terminations as shown in figures below.

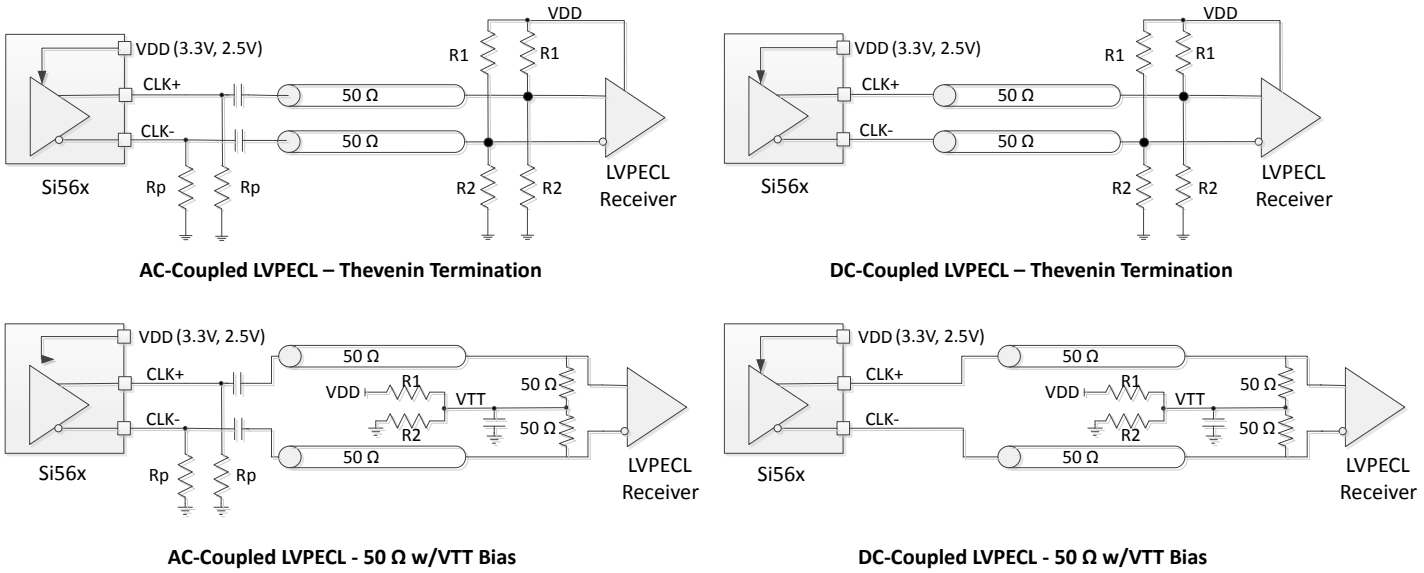



**Figure 4.1. LVPECL Output Terminations**

| AC-Coupled LVPECL Termination Resistor Values | | | |
|---|---|---|---|
| **VDD** | R1 | R2 | Rp |
| **3.3 V** | 127 Ω | 82.5 Ω | 130 Ω |
| **2.5 V** | 250 Ω | 62.5 Ω | 90 Ω |

| DC-Coupled LVPECL Termination Resistor Values | | |
|---|---|---|
| **VDD** | R1 | R2 |
| **3.3 V** | 127 Ω | 82.5 Ω |
| **2.5 V** | 250 Ω | 62.5 Ω |

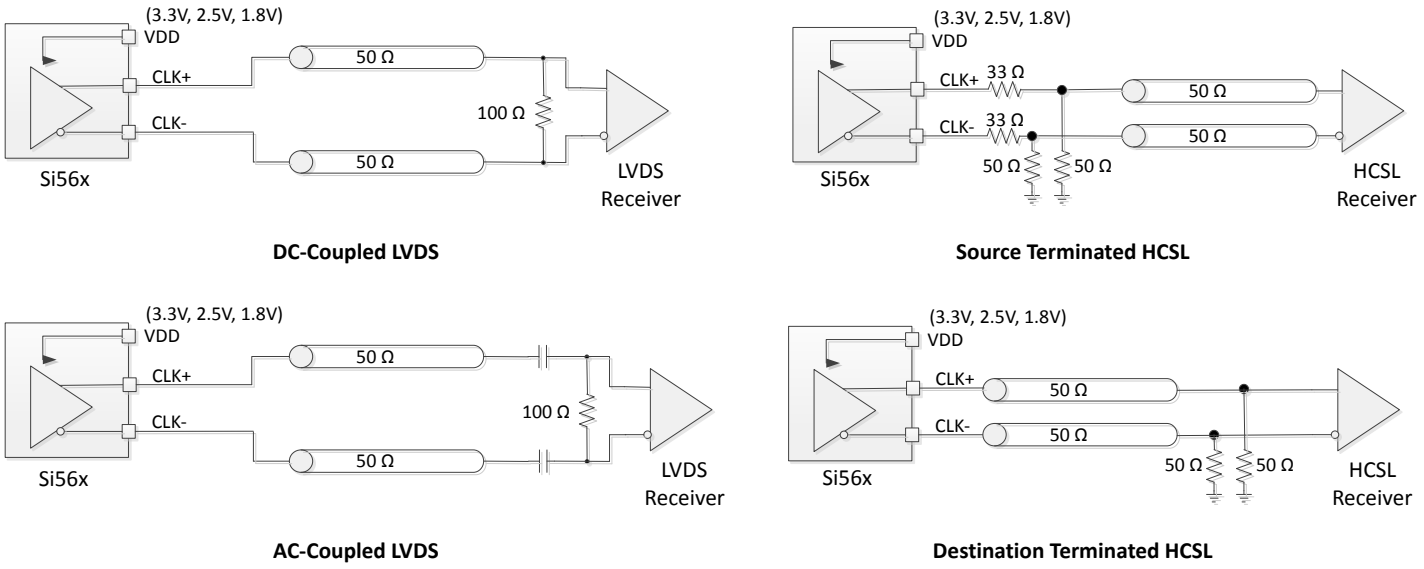



**Figure 4.2. LVDS and HCSL Output Terminations**

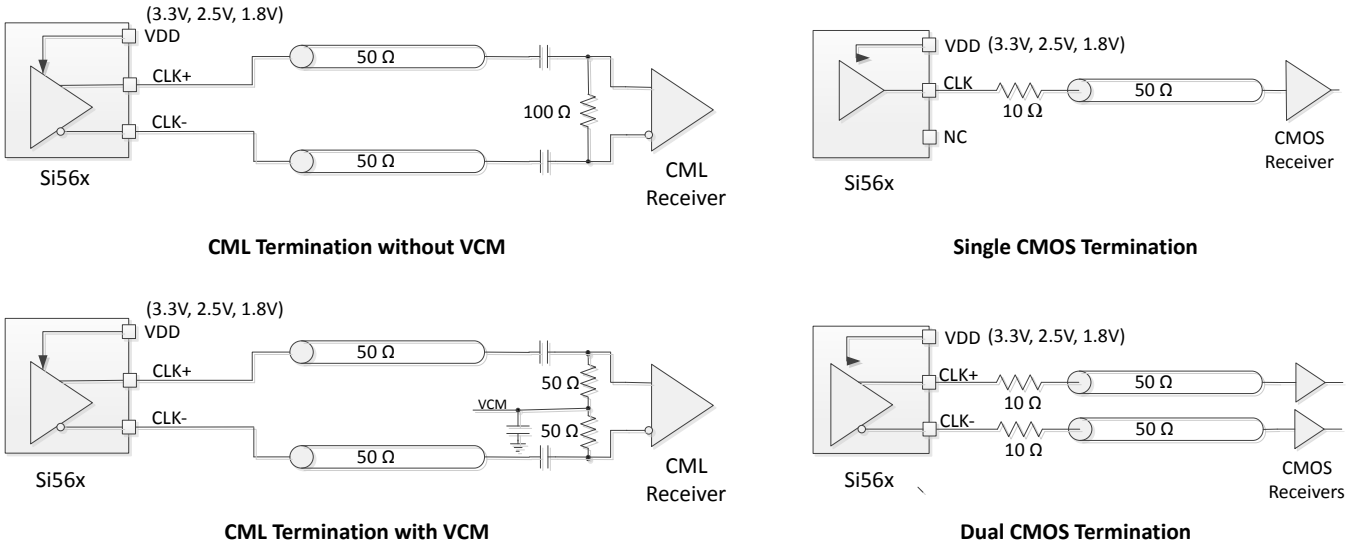



**Figure 4.3. CML and CMOS Output Terminations**

# 5. Configuring Si569 via I2C

The Si569 VCXO device contains a fixed frequency crystal and frequency synthesis IC using Silicon Labs patented DSPLL™ technology, all enclosed in a standard hermetically sealed voltage controlled crystal oscillator (VCXO) package. The internal crystal provides the reference frequency used by the DSPLL frequency synthesis IC. The center output frequency of the Si569 voltage controlled oscillator is set via I2C register settings in the DSPLL frequency synthesis IC. The output frequency is then pulled higher or lower by applying a voltage above or below VDD/2 to the VC pin. The amount of output frequency change per volt is based on a programmed ppm/V (Kv) register setting. DSPLL technology provides unmatched frequency flexibility with superior output jitter/phase noise performance and part per trillion frequency accuracy. This section describes how to calculate the required Si569 register values used to set device output frequency and Kv gain, and how to load these values into the Si569 device.

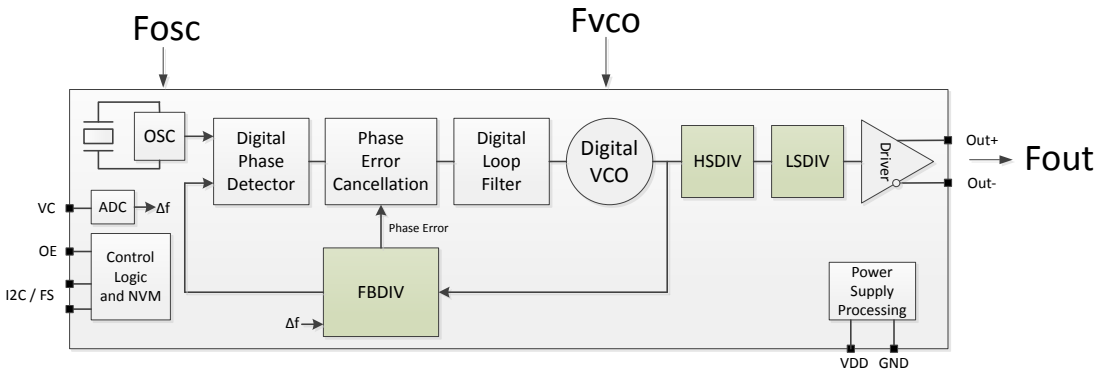



**Figure 5.1. Si569 Block Diagram**

The figure above is a simplified high-level block diagram of the Si569 VCXO device. The output frequency is set by a combination of three divider blocks highlighted in the above block diagram.

1. FBDIV - DSPLL™ Feedback Divider used to set Digital VCO frequency
2. HSDIV - High-Speed Output Divider
3. LSDIV - Low-Speed Output Divider

The final device output frequency (Fout) is based on the digital VCO frequency (Fvco) divided by the product of the HSDIV and LSDIV divider values. The digital VCO frequency is based on the crystal reference frequency (OSC) multiplied by the feedback divider setting (FBDIV). The FBDIV value is set via I2C registers and is modulated depending on the voltage on the Vc pin. The amount of digital VCO frequency variation for a given Vc voltage in ppm/V depends on the Kv register setting. The limits of each of these internal blocks (digital VCO and dividers) determines the valid operating frequency range of the device.

The FBDIV divider is a fractional fixed-point divider with a total length of 43 bits consisting of an 11-bit integer field (FBINT) and a 32 bit fractional field (FBFRAC) where total FBDIV = [FBINT].[FBFRAC] with an implied decimal point as shown. This bit format is known as an 11.32 fixed point format where the integer portion is 11 bits and fractional portion is 32 bits, for a total of 43 bits.

The HSDIV divider is an integer divider, 11 bits in length, containing a binary divider value. One noteworthy feature of the HSDIV divider is a special duty cycle correction circuit that allows **odd** divide ratios of lower divider values (4-33 only) with 50% duty cycle output. This feature is useful when LSDIV divide ratio is set to 1.

The LSDIV divider performs power-of-2 divides ranging from divide by 1 ($2^0$) to divide by 32 ($2^5$). The register controlling the LSDIV divider is 3 bits in length, holding the power-of-2 divide ratio (divider exponent). For example, if the LSDIV register = 3 the LSDIV divide ratio is 2^3 = 8. Note that LSDIV has a maximum value of 32 and therefore LSDIV register settings of 5, 6 or 7 will all result in the maximum divide-by-32 LSDIV operation.

The tables below summarize the divider limits for LSDIV, HSDIV, FBDIV. These limits and restrictions **must** be observed when deriving divider register values as will be explained in later sections.

**Table 5.1. Si569 Divider Range Limits**

| Divider | Upper Limit | Lower Limit |
|---|---|---|
| HSDIV[10:0] (unsigned) | 2046 | 4 |
| LSDIV[2:0][1] (unsigned) | 32 (2^5) | 1 (2^0) |
| FBDIV[42:0] hex (unsigned) | 7FDFFFFFFFF | 03C00000000 |
| FBDIV[42:0] int.frac (unsigned) | 2045.99999999976 | 60.0 |

**Note:**
1. LSDIV is power of 2 divider. See LSDIV table below for actual divide ratio based on LSDIV register value.

**Table 5.2. Additional LSDIV and HSDIV Divider Restrictions**

| LSDIV Register Value | Divide Ratio | HSDIV Value Restrictions |
|---|---|---|
| 0 | 1 | 4-33 even or odd values[1], 34-2046 **even values only** |
| 1 | 2 | 4-2046 even or odd values |
| 2 | 4 | 4-2046 even or odd values |
| 3 | 8 | 4-2046 even or odd values |
| 4 | 16 | 4-2046 even or odd values |
| 5 | 32 | 4-2046 even or odd values |
| 6 | 32 | 4-2046 even or odd values |
| 7 | 32 | 4-2046 even or odd values |

**Note:**
1. HSDIV can implement low value (4-33) **odd** divide ratios while providing a 50% duty cycle output due to special duty cycle correction circuit.

Note that all divider values (FBDIV, HSDIV, LSDIV) are **unsigned** and contain only positive values.

The Si569 high-performance VCXO family has four different speed grade offerings, each covering a specific frequency range. The table below outlines the output frequency range coverage by each speed grade, the corresponding min and max VCO frequency for that speed grade, and the nominal crystal frequency. The information in the table below is needed when calculating divider settings for a given device, speed grade, and output frequency.

**Table 5.3. Si569 Speed Grades, Crystal Frequency, and VCO Range Limits**

| Device | Speed Grade | Xtal freq (MHz) | Min Output Freq (MHz) | Max Output Freq (MHz) | Min Fvco (GHz) | Max Fvco (GHz) |
|---|---|---|---|---|---|---|
| Si569 | A | 152.6 | 0.2 | 3000 | 10.8 | 13.122222022 |
| | B | 152.6 | 0.2 | 1500 | 10.8 | 12.511886114 |
| | C | 152.6 | 0.2 | 800 | 10.8 | 12.206718160 |
| | D | 152.6 | 0.2 | 325 | 10.8 | 12.206718160 |

### 5.1 Output Frequency and Kv Gain Calibration Equations

The basic equations used to derive the output frequency are given below and can be inferred from the device block diagram in Figure 5.2 Si569 Frequency Definition Block Diagram on page 14. Equation 1 is the relationship between the output frequency (Fout), and the VCO frequency (Fvco) and total output divider ratio (HSDIV * LSDIV). Equation 2 is the relationship between the VCO frequency (Fvco), the fixed crystal oscillator frequency (Fosc), and the feedback divider (FBDIV). Equation 2 also includes frequency adjustment (Δf) using the input control voltage (Vc) and the ppm/V control voltage gain (Kv).

$$Fout = Fvco / (HSDIV \times LSDIV)$$

**Equation 1**

$$Fvco = (Fosc \times FBDIV) \times (1 + \Delta f) \text{ (offset freq in ppm)}$$

**Equation 2a**

$$Fvco = (Fosc \times FBDIV) \times (1 + [(Vc - VDD/2) * Kv])$$

**Equation 2b**

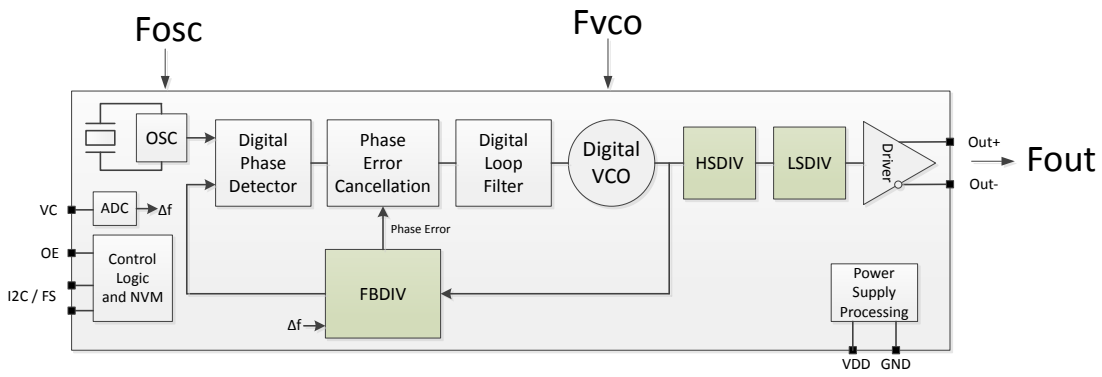



**Figure 5.2. Si569 Frequency Definition Block Diagram**

Equation 3a is a rearranged Equation 1 to solve for the total output divider (HSDIV *LSDIV) given Fout and Fvco. Equation 3b is rearranged again solving for Fvco given Fout and (HSDIV * LSDIV).

$$(HSDIV \times LSDIV) = Fvco / Fout$$

**Equation 3a**

$$Fvco = Fout \times (HSDIV \times LSDIV)$$

**Equation 3b**

Equation 4a is a rearranged Equation 2b to now solve for FBDIV given Fvco, Vc, VDD, Kv, and Fosc. Equation 4b simplifies Equation 4a to determine the FBDIV value for the center output frequency when Vc = VDD/2.

$$FBDIV = (Fvco / Fosc) / [1 + (Vc - VDD/2) * Kv]$$

**Equation 4a**

$$FBDIV = Fvco / Fosc \text{ for } Vc = VDD/2 \text{ (center frequency)}$$

**Equation 4b**

Equations 3a, 3b, 4a, and 4b will be used in the process of deriving the required divider values to provide a desired center output frequency (Fout). The basic process is outlined in the next section.

Whenever the Fvco frequency is modified from the factory default, it is necessary to re-calibrate Kv gain. This is because the Vc ADC input sampling rate is tied to Fvco and is factory calibrated to 80 MHz based on the factory Fvco setting. Whenever Fvco is modified to change the output center frequency, the Vc ADC sampling rate is also changed so the full-scale Kv gain must be re-calculated.

CADC_FSGAIN = round (128 * nominal Vc ADC sampling rate / new Vc ADC sampling rate)

**Equation 5a**

CADC_FSGAIN = round (128 * 80e6 / (Fvco / NFXDIV / 8))

**Equation 5b**

Equations 5a and 5b are used along with the table below to re-calculate the Kv full scale gain. This process is also outlined in the next section.

**Table 5.4. Si569 NFXDIV Values for Different FBDIV Integer Values**

| FBDIV Min | FBDIV Max | NFXDIV Value |
|---|---|---|
| — | 71.999999… | 16 |
| 72.000000… | 78.999999… | 18 |
| 79.000000… | 85.999999… | 20 |
| 86.000000… | — | 22 |

### 5.2 General Process Steps for Divider Calculations and Kv Gain Calibration

1. Estimate a theoretical total output divider value (HSDIV * LSDIV) based on desired Fout while targeting the minimum valid Fvco frequency using Equation 3a and Table 5.3 Si569 Speed Grades, Crystal Frequency, and VCO Range Limits on page 13. Use floating point calculations for this step.
   - Result: Floating point value of total output divider (HSDIV * LSDIV) for Fvco minimum.
2. Derive a valid LSDIV divider value based on LSDIV and HSDIV divider limitations. Use the lowest possible integer value for LSDIV. For example, if the floating point output divider (HSDIV * LSDIV) for Fvco minimum = 8.22, use LSDIV = 1 and HSDIV = 8.22 versus LSDIV = 2 and HSDIV = 4.11.
   - Result: Valid integer LSDIV value.
3. Using the LSDIV value from #2 above, find the nearest valid **integer** HSDIV divider value resulting in Fvco being **equal to or greater than Fvco min**, observing all HSDIV limitations. Use Equations 3a/3b as necessary.
   - Result: Valid integer HSDIV value.
4. With valid integer HSDIV and LSDIV values, calculate the target Fvco center frequency with Equation. 3b. (Fvco must remain in the valid range per Table 5.3 Si569 Speed Grades, Crystal Frequency, and VCO Range Limits on page 13.)
   - Result: Valid Fvco frequency.
5. With the derived valid Fvco frequency, use Equation 4b to calculate the required FBDIV based on the device specific Fosc frequency from Table 5.3 Si569 Speed Grades, Crystal Frequency, and VCO Range Limits on page 13. Assume Vc = VDD/2 to calculate an FBDIV value for the center Fout frequency.
   - Result: Valid fractional FBDIV value
6. At this point all FBDIV, HSDIV and LSDIV values required to generate the desired center output frequency have been calculated. These three divider values must be now be appropriately formatted to fit the register format expected by the device. This is described in a later section.
   - Result: Valid register values for FBDIV, HSDIV, LSDIV
7. To re-calibrate Kv gain, first determine the integer portion of the new FBDIV value in step #5 above using truncation (not rounding) and then use that value to select the correct NFXDIV value using Table 5.4 Si569 NFXDIV Values for Different FBDIV Integer Values on page 15.
   - Result: Valid NFXDIV value
8. To complete Kv gain calibration, calculate the new Kv gain calibration value (CADC_FSGAIN) using Equation 5b. This Kv gain calibration value must be appropriately formatted to fit the register format expected by the device. This is described in a later section.
   - Result: Valid CADC_FSGAIN value

### 5.3 Example: Deriving Si569 Divider Settings for 156.75 MHz Output

The general process of deriving divider values for a specific output frequency is outlined in the previous section and now will be used in this example. To reiterate, all calculations must be done while observing divider limits and valid VCO frequency range limits for your device. In this example, the device is Si569 and with a desired output frequency of 156.75 MHz, the speed grade required will be “D” or better. (One important note: All divider and register settings derived for any speed grade will work without modification for all **faster** speed grades on the same base part number device.)

**Example VB code that implements the following divider calculation process is given in 5.10 Si569 Frequency Planner VB Code and can be used for implementing any supported output frequency.**

**Step 1:** Find the valid theoretical lower limit of the total output divider (HSDIV*LSDIV) based on the desired output frequency and lowest valid VCO frequency. This will bias the divider solution to the lowest possible VCO frequency since this will provide the best performance solution.

Given the valid Si569 VCO range is 10.8000 GHz to 12.206718160 GHz, the minimum theoretical values for (HSDIV * LSDIV) for the example 156.75 MHz output frequency are given in Equation 3:

Minimum (HSDIV*LSDIV) = (10.8000 GHz / 156.75 MHz) = **68.89952…**

**Step 2:** Find valid LSDIV divisor value given minimum (HSDIV*LSDIV) from step 1. For best performance, preference should be given to implementation of the total output divider (HSDIV*LSDIV) using HSDIV with LSDIV divide ratio = 1, if possible. Use LSDIV divide ratios > 1 only if HSDIV alone cannot implement the required output divider. Since the total (HSDIV*LSDIV) value of 68.8995… is less than the HSDIV maximum divider value of 2046, the LSDIV divide **ratio** value will be 1, which corresponds to a LSDIV register setting of 0, since the LSDIV divider can only be a power of 2 value (see Table 5.2 Additional LSDIV and HSDIV Divider Restrictions on page 13 for valid LSDIV settings).

LSDIV divide ratio = 1, therefore **LSDIV register value = 0**

**Step 3:** Find HSDIV divisor value. Given LSDIV = 1, HSDIV must implement 68.8995… or greater. Since HSDIV is an integer divider, the next greatest integer is 69. But, checking valid HSDIV values when LSDIV divide ratio = 1, we see 69 is NOT valid since it is greater than 33 and an odd value. This means the next greater integer value must be used, which is 70 (now even value). Note that 68 would **not** be valid since 68 is less than 68.8995… and would result in a VCO frequency below the lower VCO frequency limit.

HSDIV divide ratio = 70, which gives **HSDIV register value = 70 decimal (or hex value = 0x46)**

**Step 4:** Calculate a valid VCO frequency and corresponding floating point FBDIV value. Given the calculated output divider value (HSDIV*LSDIV) = 70, the VCO frequency must be set to (156.75 MHz * 70) = **10.9725 GHz**. Note that 10.9725 GHz is indeed within the valid VCO frequency range per Table 5.3 Si569 Speed Grades, Crystal Frequency, and VCO Range Limits on page 13.

Fvco = **10.9725 GHz**

**Step 5:** Calculate the FBDIV value necessary to provide a 10.9725 GHz Fvco using a 152.6 MHz crystal as reference (Si569 device). The floating point FBDIV value required to attain 10.9725 GHz with a 152.6 MHz crystal reference can be calculated as follows:

**FBDIV (float)** = 10.9725 GHz / 152.6 MHz = **71.9036697247707**

**Step 6:** Format each divider value into the required register format. LSDIV and HSDIV are simply binary values and can be directly used. FBDIV must first be put into 11.32 fixed point format. Converting the floating point FBDIV value into the 11.32 fixed point hex value required by the Si569 is done as follows:

**Integer** value = **71** decimal. Convert **71** to 11 bit hex = **0x047**. This is FBINT.

**Fractional** value = **0.9036697247707**. Multiply fractional value by 2^32 = 3881231914.2752. Now extract only the **integer** part of the result which is 3881231914. Convert 3881231914 to 32 bit hex = **0xE756E62A.** This is FBFRAC.

The resulting 11.32 fixed point hex number is therefore:

**FBDIV = FBINT.FBFRAC** = **0x047E756E62A**

At this point we have calculated all the required divider values. The table below summarizes the resulting divider values for implementing a 156.75 MHz output clock on the Si569.

**Table 5.5. Divider Register Values for Si569 Configured for 156.75 MHz Output Clock**

| Divider Register | Decimal Value | Hex Value | Reg Length (bits) |
|---|---|---|---|
| LSDIV | 0 | 0x0 | 3 |
| HSDIV | 70 | 0x046 | 11 |
| FBDIV | 71.9036697247707 | 0x047E756E62A | 43 (11+32) |

### 5.4 Example: Deriving Si569 Kv Gain Settings for 156.75 MHz Output

Whenever the Fvco frequency is modified from the factory default it is necessary to re-calibrate Kv gain.

**Step 1:** Find the Fvco and FBDIV values from the new configuration to be used for Equation 5b.

Fvco = **10.9725 GHz**

FBDIV (float) = 10.9725 GHz / 152.6 MHz = **71.9036697247707**

**Step 2:** Use the integer portion of FBDIV to find the correct value for NFXDIV using Table 5.4 Si569 NFXDIV Values for Different FBDIV Integer Values on page 15. Do not round up the integer portion of FBDIV, instead truncate FBDIV down via the floor function.

FBDIV (int) = floor (10.9725 GHz / 152.6 MHz) = floor (71.9036697247707) = **71**

Excerpt from Table 5.4 Si569 NFXDIV Values for Different FBDIV Integer Values on page 15:

| FBDIV Min | FBDIV Max | NFXDIV Value |
|---|---|---|
| — | 71.999999… | **16** |
| 72.000000… | 78.999999… | 18 |

**Step 3:** Calculate the new CADC_FSGAIN calibration value using Fvco, FBDIV (int) and NFXDIV.

CADC_FSGAIN = round (128 * 80e6 / (Fvco / NFXDIV / 8))

CADC_FSGAIN = round (128 * 80e6 / (10.9725e9 / 16 / 8))

**CADC_FSGAIN** = round (119.455) = **119** = **0x77**

### 5.5 Mapping Divider Settings into Register Values

For the previous 156.75 MHz example, the divider value to register mapping is shown in the table below. Note that Register 24 is a packed register and contains bits from both LSDIV and HSDIV registers as follows: LSDIV[2:0] maps into Reg24[6:4] and HSDIV[10:8] maps into Reg24[2:0]. Note that bits Reg24[7] and Reg24[3] are not used and indicated with 'x' in the RegName field below. See also the Register Map Reference section for specific bit positioning within registers.

**Table 5.6. Si569 Divider Register Values for 156.75 MHz Output Clock Configuration**

| Register (Decimal) | Hex Value | Reg Name |
|---|---|---|
| 23 | 46 | HSDIV[7:0] |
| 24 | 00 | x:LSDIV[2:0]:x:HSDIV[10:8] |
| | | |
| 26 | 2A | FBDIV[7:0] |
| 27 | E6 | FBDIV[15:8] |
| 28 | 56 | FBDIV[23:16] |
| 29 | E7 | FBDIV[31:24] |
| 30 | 47 | FBDIV[39:32] |
| 31 | 00 | FBDIV[42:40] |
| 35 | 77 | CADC_FSGAIN[7:0] |

### 5.6 I2C Register Write Procedure to Set Output Frequency

After the frequency setting registers (Reg 23-Reg31) are calculated, there is a procedure that must be followed involving other specific control registers for the device to properly use the new frequency setting registers. Simply writing Reg23-Reg31 is not enough. The following procedure must be performed as shown to properly configure the Si569 for the desired output frequency. In other words, all the following register writes must be done, and **in the exact sequence shown**.

This programming sequence consists of three distinct phases.

1. Writing to specific registers to get the device ready to be updated.
2. Writing the calculated frequency (divider) settings for the desired output frequency.
3. Writing to specific registers necessary to start-up the device after divider registers have been updated. The new output frequency will appear on output.

The divider values shown in the table below are for the previously described Si569 example for an output frequency of 156.75 MHz (for other frequencies, replace the divider values in registers 23-31 with values specific to your frequency requirements).

**Table 5.7. Si569 Register Write Sequence to Set Output Frequency**

| Register (decimal) | Write Data (hex) | Description | Purpose |
|---|---|---|---|
| 255 | 0x00 | Set page register to point to page 0 | Get Device Ready for Update |
| 69 | 0x00 | Disable FCAL override (to allow FCAL for this Freq Update) | |
| 17 | 0x00 | Synchronously disable output | |
| 23 | 0x46 | HSDIV[7:0] | Update Dividers |
| 24 | 0x00 | LSDIV[2:0]:HSDIV[10:8] | |
| 26 | 0x2A | FBDIV[7:0] | |
| 27 | 0xE6 | FBDIV[15:8] | |
| 28 | 0x56 | FBDIV[23:16] | |
| 29 | 0xE7 | FBDIV[31:24] | |
| 30 | 0x47 | FBDIV[39:32] | |
| 31 | 0x00 | FBDIV[42:40] | |
| 35 | 0x77 | CADC_FSGAIN[7:0] | Update Kv Gain |
| 7 | 0x08 | Start FCAL using new divider values | Startup Device |
| 17 | 0x01 | Synchronously enable output | |

**Note:** Refer to the device data sheet for default Si569 I2C address or to the device data sheet addendum for your specific I2C address.

### 5.7 Digitally Controlled Oscillator – ADPLL: Small, Fast Frequency Changes

The Si569 can make small, fast frequency adjustments over a range of +/- 950 ppm (parts-per-million) around the device output frequency (set as described in previous sections). This mode is typically used in applications requiring a digitally controlled oscillator (DCO) for digital PLL or other types of frequency control loops. We refer to this type of application as an all-digital PLL or ADPLL.

The ADPLL mode uses a single 24 bit register, ADPLL_DELTA_M[23:0], to add an offset to the VCO frequency to affect the small frequency change. This offset is added in a synchronous fashion to prevent frequency discontinuities and can be updated as fast as the max I2C bus speed of 1 MHz will allow. The frequency offset can be positive or negative over a range of -950 ppm to +950 ppm with 0.0001164 ppm resolution.

The equation for this frequency change is simply,

$$ADPLL_DELTA_M[23:0] = \Delta\ FoutPPM / 0.0001164$$

Where $\Delta\ Fout_{PPM}$ is the desired ppm change in output frequency, ADPLL_DELTA_M[23:0] is a two's complement 24 bit value, and 0.0001164 is a constant per-bit ppm value. The 24 bit ADPLL_DELTA_M[23:0] value is written into three sequential 8 bit registers in LSByte to MSByte order via I2C. Upon writing the MSByte, the frequency change takes effect. Below is an example VB to implement this feature. (Note that writing ADPLL_DELTA_M[23:0] = 0x000 will result in no frequency offset and return to the nominal output frequency.)

VB Code example for ADPLL (small frequency change) calculation and operation:

```
nAddr = Device I2C address
PPM_Delta = desired PPM frequency shift

    Function Set_ADPLL(ByVal nAddr As UInteger, ByVal PPM_Delta As Double) As Integer
        Dim ADPLL_PPM_StepSize As Double = 0.0001164
        Dim ADPLL_Delta_M As Integer
        Dim Reg231 As UInteger = 0
        Dim Reg232 As UInteger = 0
        Dim Reg233 As UInteger = 0
        Dim ReturnCode As Integer = 0  '1=OK, -1 PPM requested is out of bounds
        If (PPM_Delta <= 950 And PPM_Delta >= -950) Then
            ADPLL_Delta_M = (PPM_Delta / ADPLL_PPM_StepSize)
            Reg231 = (ADPLL_Delta_M And &HFF)
            Reg232 = (ADPLL_Delta_M >> 8) And &HFF
            Reg233 = (ADPLL_Delta_M >> 16) And &HFF
            I2C_Write(nAddr, 0, 231, Reg231)    'write “Reg231” value to register 231 at nAddr, page 0 (LSByte)
            I2C_Write(nAddr, 0, 232, Reg232)    'write “Reg232” value to register 232 at nAddr, page 0
               I2C_Write(nAddr, 0, 233, Reg233)      'write “Reg233” value to register 233 at nAddr, page 0
(MSByte)
            ReturnCode = 1
        Else
            ReturnCode = -1
        End If
        Return (ReturnCode)
    End Function
```

### 5.8 Configuring High Drive LVDS Swing

The Si569 LVDS clock output swing can be increased 100 mV via I2C to have the same swing as AC-coupled CML. This is done by programming the three registers as shown in the table below.

**Table 5.8. LVDS and CML Output Drive Settings**

| Register Address (dec) | Output Drive | LVDS (dec) | High Drive LVDS / CML (dec) |
|---|---|---|---|
| 16 [5:0] | OD_DRV_TRIM_V3P3[5:0] | 17 | 20 |
| 125 [5:0] | OD_DRV_TRIM_V2P5[5:0] | 20 | 23 |
| 126 [5:0] | OD_DRV_TRIM_V1P8[5:0] | 22 | 25 |

## 5.9 Register Map Reference

**Table 5.9. Register Map Reference Summary**

| Register (decimal) | Register Bit | | | | | | | | Type | Reset Value |
|---|---|---|---|---|---|---|---|---|---|---|
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| 7 | RESET | <Reserved> = 3'b000 | | | MS_ICAL 2 | <Reserved> = 3'b000 | | | R/W | 0x00 |
| 17 | <Unused> | | | | | | | ODC_OE | R/W | 0x01 |
| 23 | HSDIV[7:0] | | | | | | | | R/W | 0x54 |
| 24 | <Unused> | LSDIV[2:0] | | | <Unused> | HSDIV[10:8] | | | R/W | 0x00 |
| 26 | FBDIV[7:0] | | | | | | | | R/W | 0x00 |
| 27 | FBDIV[15:8] | | | | | | | | R/W | 0x00 |
| 28 | FBDIV[23:16] | | | | | | | | R/W | 0x00 |
| 29 | FBDIV[31:24] | | | | | | | | R/W | 0x00 |
| 30 | FBDIV[39:32] | | | | | | | | R/W | 0x64 |
| 31 | <Unused> | | | | | FBDIV[42:40] | | | R/W | 0x00 |
| 32 | <Unused> | | | KV_VCXO[4:0] | | | | | R/W | 0x06 |
| 35 | CADC_FSGAIN[7:0] | | | | | | | | R/W | 0x80 |
| 69 | FCAL_OVR | <Reserved> = 7'b0000001 | | | | | | | R/W | 0x01 |
| 231 | ADPLL_DELTA_M[7:0] | | | | | | | | R/W | 0x00 |
| 232 | ADPLL_DELTA_M[15:8] | | | | | | | | R/W | 0x00 |
| 233 | ADPLL_DELTA_M[23:16] | | | | | | | | R/W | 0x00 |
| 255 | <Reserved> = 6'b000000 | | | | | | PAGE[1:0] | | R/W | 0x00 |

**Table 5.10. Register Bit Field Summary**

| Register Bit Field Name | Bit Field (#bits) | Register | Description |
|---|---|---|---|
| RESET | 1 | 7 | Set to 1 to reset device. Self clearing. |
| MS_ICAL2 | 1 | 7 | Set to 1 to initiate FCAL. Self clearing. |
| HSDIV[10:0] | 11 | 23-24 | HSDIV is High-speed output divider value in unsigned 11-bit binary format. Valid divide values are from 5 to 2046, with values of 5-33 even or odd, and values 34-2046 restricted to even values only. |
| LSDIV[2:0] | 3 | 24 | LSDIV sets a power-of-2 output divider. Values of 0,1,2,3,4,5,6,7 result in divide ratio of 1,2,4,8,16,32,32,32 respectively. Note that a value of 0 (divide-by-1) essentially bypasses this divider. |

| Register Bit Field Name | Bit Field (#bits) | Register | Description |
|---|---|---|---|
| FBDIV[42:0] | 43 | 26-31 | The main DSPLL system feedback divide (FBDIV) value for Si56x. This 43 bit value is composed of an unsigned 11-bit integer value (FBDIV[42:32]) concatenated with a 32-bit fractional value (FBDIV[31:0]), for an 11.32 fixed point binary format. The valid range of the 11-bit integer part is from 60 to 2045. |
| KV_VCXO[4:0] | 5 | 32 | Sets Vc voltage control gain Kv (ppm/V). Multiply the register value in decimal by 7.5 to get the actual Kv in ppm/V. |
| CADC_FSGAIN[7:0] | 8 | 35 | Full-scale Kv gain parameter. Used to set the (ppm/V) full-scale of the Vc input depending on the programmed VCO frequency. |
| FCAL_OVR | 1 | 69 | FCAL Override: If set to 1, FCAL is bypassed. Clear to 0 to allow FCAL. |
| ADPLL_DELTA_M[23:0] | 24 | 231-233 | Digital word to effect small frequency shifts to base frequency. Value is 24 bit 2's complement causing a 0.0001164 ppm per bit shift in frequency. Positive values = positive freq shift, negative values = negative freq shift. Valid range is -8161513 to +8161512, representing a max PPM shift range of -950 ppm to +950 ppm, with 0 value representing 0 PPM shift. Writing a new ADPLL_DELTA_M value will take effect upon writing to the MSByte (Register 233). Therefore, value updates should follow the sequence of writing in register order Reg 231...Reg 232...Reg 233. |
| PAGE[1:0] | 2 | 255 | Sets which page of registers the I2C port is reading/writing. The size of a page is 256 bytes which is the addressable range of an I2C "**set address**" command. The value of PAGE is multiplied by 256 and added to what "**set address**" has set. Physically, the 2 PAGE bits become bits [9:8] of the device's internal register map address. This mechanism allows for more than 256 registers to be addressed within the 8 bit I2C "**set address**" limitation. |

### 5.10 Si569 Frequency Planner VB Code

```
---------------------------------------------------------------------------------------------
Module Main
    '
    ' Si56x Frequency Planner Code
    '
    '
    'Set Target device type, Speed grade, and desired output frequency
    '
    Public Device As Integer = 569     '
    Public SpeedGrade As String = "D"  'Can only be "A" or "B" or "C" or “D”
    Public Output_Freq As Double = 312500000.0  'Output frequency in Hz (initially set to 312.5 MHz)

    'Set in 'SetLimits" function...
    Public Fvco_max As Double  'Fvco Max per Table 3
    Public Fvco_min As Double  'Fvco Min per Table 3
    Public Xtal_freq As Double 'Xtal_Freq per Table 3
    Public Fout_min As Double  'Minimum output frequency
    Public Fout_max As Double  'Maximum output frequency

    Sub Main()
        '
        ' Device divider limits (see Tables 1 & 2)
        '
        Dim HSDIV_UpperLimit As Integer = 2046
        Dim HSDIV_LowerLimit As Integer = 4
        Dim HSDIV_LowerLimit_Odd As Integer = 5     'min count for odd HSDIV divisor
        Dim HSDIV_UpperLimit_Odd As Integer = 33    'max count for odd HSDIV divisor
        Dim LSDIV_UpperLimit As Integer = 5
        Dim LSDIV_LowerLimit As Integer = 0
        Dim FBDIV_UpperLimit As Double = 2045 + ((2 ^ 32 - 1) / (2 ^ 32))
        Dim FBDIV_LowerLimit As Double = 60.0
        '
        ' Working variables
        '
        Dim Min_HSLS_Div As Double
        Dim LSDIV_Div As Double     ' actual LSDIV divide ratio
        Dim LSDIV_Reg As Integer    ' LSDIV as encoded in power of 2 for device register use
        Dim HSDIV As Double
        Dim FBDIV As Double
        Dim Fvco As Double
        Dim FBDIV_Int As UInteger
        Dim FBDIV_Frac As UInteger
        Dim Reg23 As UInteger = 0    'HSDIV[7:0]
        Dim Reg24 As UInteger = 0    'OD_LSDIV[2:0],HSDIV[10:8]   (*2^4,/2^8)
        Dim Reg26 As UInteger = 0    'FBDIV[7:0]
        Dim Reg27 As UInteger = 0    'FBDIV[15:8]   (/2^8)
        Dim Reg28 As UInteger = 0    'FBDIV[23:16]  (/2^16)
        Dim Reg29 As UInteger = 0    'FBDIV[31:24]  (/2^24)
        Dim Reg30 As UInteger = 0    'FBDIV[39:32]  (/2^32)
        Dim Reg31 As UInteger = 0    'FBDIV[42:40]  (/2^40)

        '
        ' Set device limits based on device type and speed grade.
        ' (Checks if desired output frequency is valid based on device and speed grade)
        '
        If SetLimits(Device, SpeedGrade, Output_Freq) = 0 Then
            '
            ' If limits are set and output frequency is valid, calculate frequency plan...

            '*********************************************************************************************
            '  Step 1: Find theoretical HSDIV *LSDIV value based on lowest valid VCO frequency...
              '           (Assumes "Output_Freq" has been tested and is in valid range for the device grade
according to Table 3)
            '
              Min_HSLS_Div = Fvco_min / Output_Freq      ' Floating point HS*LS div value. Remember to first
bounds check Output_Freq!

            'Step 2: Find LSDIV divisor value given Min_HSLS_Div value
            '
               LSDIV_Div = Math.Ceiling(Min_HSLS_Div / HSDIV_UpperLimit)   ' Divisor value of LSDIV, NOT yet
```

```
encoded as power of 2
            If (LSDIV_Div > 32) Then LSDIV_Div = 32 ' clip at 32 (max LSDIV divisor)
            '
              'Encode LSDIV divisor value into next nearest 'power of 2' value if not already. This will be
LSDIV_Reg
            '
            LSDIV_Reg = Math.Ceiling(Math.Log(LSDIV_Div, 2))        ' LSDIV_Reg now encoded as proper power of
2. Will range from 0 to 5.

            ' Adjust LSDIV_Div (holder of divisor) based on rounded power of 2 value in LSDIV_Reg
            LSDIV_Div = 2 ^ LSDIV_Reg 'LSDIV_Div divisor now synchronized to actual LSDIV_Reg.
            '
            'Step 3: Find HSDIV divisor value using known LSDIV divisor
            '
            HSDIV = Math.Ceiling(Min_HSLS_Div / LSDIV_Div)
            If ((LSDIV_Reg > 0) Or ((HSDIV >= HSDIV_LowerLimit_Odd) And (HSDIV <= HSDIV_UpperLimit_Odd))) Then
                HSDIV = HSDIV  ' Leaves HSDIV as even or odd only if LSDIV_Div = 1 and HSDIV is from 4 to 33.
            Else
                If ((HSDIV Mod 2) <> 0) Then    'If HSDIV is an odd value...
                    HSDIV = HSDIV + 1           '...make it even by rounding up
                End If                          'If already even, leave it alone
            End If
            '
            ' Step 4: Now calculate Fvco and FBDIV
            '
            Fvco = (HSDIV * LSDIV_Div * Output_Freq)    'Calculate Fvco based on valid HSDIV,LSDIV, and Fout
            FBDIV = Fvco / Xtal_freq                    'Finally, calculate FBDIV based on xtal freq

'Calculate 11.32 fixed point FBDIV value (MCTL_M)
'Extract Integer part
FBDIV_Int = Int(FBDIV)
'Extract fractional part
FBDIV = (FBDIV - FBDIV_Int)
FBDIV = FBDIV * (2 ^ 32)
FBDIV_Frac = Int(FBDIV)
'
'Generate Register values based on LSDIV, HSDIV, and FBDIV (MCTL_M)
'
Reg23 = (HSDIV And &HFF)
Reg24 = ((HSDIV >> 8) And &H7) Or ((LSDIV_Reg And &H7) << 4)

Reg26 = (FBDIV_Frac And &HFF)
Reg27 = (FBDIV_Frac >> 8) And &HFF
Reg28 = (FBDIV_Frac >> 16) And &HFF
Reg29 = (FBDIV_Frac >> 24) And &HFF
Reg30 = (FBDIV_Int) And &HFF
Reg31 = (FBDIV_Int >> 8) And &H7

            '*********************************************************************
        Else
            Console.WriteLine("*** Device invalid or Device limits exceeded. Frequency plan not calculated.")
        End If

    End Sub
    '
    ' Sets device limits according to Table 3
    '   Returns 0 if limits are set and output frequency is valid
    '   Returns -1 if device not found or output frequency/speed grade is invalid
     Function SetLimits(ByVal Device As Integer, ByVal SpeedGrade As String, ByVal Output_Freq As Double) As
Integer
        Dim ReturnCode As Integer
        ReturnCode = 0
        If Device = 569 Then
            Xtal_freq = 152600000.0
            If SpeedGrade = "A" Then
                Fvco_min = 10800000000.0
                Fvco_max = 13122222022.0
                Fout_min = 200000.0
                Fout_max = 3000000000.0
                If ((Output_Freq < Fout_min) Or (Output_Freq > Fout_max)) Then
                    ReturnCode = -1
                End If
            ElseIf SpeedGrade = "B" Then
```

```
                Fvco_min = 10800000000.0
                Fvco_max = 12511886114.0
                Fout_min = 200000.0
                Fout_max = 1500000000.0
                If ((Output_Freq < Fout_min) Or (Output_Freq > Fout_max)) Then
                    ReturnCode = -1
                End If
            ElseIf SpeedGrade = "C" Then
                Fvco_min = 10800000000.0
                Fvco_max = 12206718160.0
                Fout_min = 200000.0
                Fout_max = 800000000.0
                If ((Output_Freq < Fout_min) Or (Output_Freq > Fout_max)) Then
                    ReturnCode = -1
                End If
            ElseIf SpeedGrade = "D" Then
                Fvco_min = 10800000000.0
                Fvco_max = 12206718160.0
                Fout_min = 200000.0
                Fout_max = 325000000.0
                If ((Output_Freq < Fout_min) Or (Output_Freq > Fout_max)) Then
                    ReturnCode = -1
                End If
            Else
                ReturnCode = -1   'Speed Grade not found
            End If
        Else
            ReturnCode = -1    'Device type not found
        End If
        Return (ReturnCode)
    End Function
End Module
-------------------------------------------------------------------------------
```

### 5.11 Table of Common Frequencies for Si569 (152.6 MHz xtal)

| Fout (MHz) | LSDIV | HSDIV | FBDIV | Fvco (GHz) | Reg 23 | Reg 24 | Reg 26 | Reg 27 | Reg 28 | Reg 29 | Reg 30 | Reg 31 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 70.656 | 0 | 154 | 71.30422018 | 10.881024 | 9Ah | 00h | BAh | 5Fh | E1h | 4Dh | 47h | 00h |
| 100 | 0 | 108 | 70.77326343 | 10.8 | 6Ch | 00h | A7h | 97h | F4h | C5h | 46h | 00h |
| 122.88 | 0 | 88 | 70.86133683 | 10.81344 | 58h | 00h | 04h | 92h | 80h | DCh | 46h | 00h |
| 125 | 0 | 88 | 72.08387942 | 11 | 58h | 00h | 34h | 1Fh | 79h | 15h | 48h | 00h |
| 148.351648 | 0 | 74 | 71.93985552 | 10.97802195 | 4Ah | 00h | 07h | 5Fh | 9Ah | F0h | 47h | 00h |
| 148.5 | 0 | 74 | 72.01179554 | 10.989 | 4Ah | 00h | 63h | 08h | 05h | 03h | 48h | 00h |
| 148.945454 | 0 | 74 | 72.22780862 | 11.0219636 | 4Ah | 00h | 7Dh | AAh | 51h | 3Ah | 48h | 00h |
| 150 | 0 | 72 | 70.77326343 | 10.8 | 48h | 00h | A7h | 97h | F4h | C5h | 46h | 00h |
| 153.6 | 0 | 72 | 72.47182176 | 11.0592 | 48h | 00h | 84h | 4Fh | C9h | 78h | 48h | 00h |
| 155.52 | 0 | 70 | 71.33944954 | 10.8864 | 46h | 00h | 46h | 2Ah | E6h | 56h | 47h | 00h |
| 156.25 | 0 | 70 | 71.67431193 | 10.9375 | 46h | 00h | D8h | B4h | 9Fh | ACh | 47h | 00h |
| 168.04 | 0 | 66 | 72.67785059 | 11.09064 | 42h | 00h | C2h | 9Dh | 87h | ADh | 48h | 00h |
| 168.75 | 0 | 64 | 70.77326343 | 10.8 | 40h | 00h | A7h | 97h | F4h | C5h | 46h | 00h |
| 200 | 0 | 54 | 70.77326343 | 10.8 | 36h | 00h | A7h | 97h | F4h | C5h | 46h | 00h |
| 212.5 | 0 | 52 | 72.41153342 | 11.05 | 34h | 00h | 17h | 41h | 5Ah | 69h | 48h | 00h |
| 245.76 | 0 | 44 | 70.86133683 | 10.81344 | 2Ch | 00h | 04h | 92h | 80h | DCh | 46h | 00h |
| 250 | 0 | 44 | 72.08387942 | 11 | 2Ch | 00h | 34h | 1Fh | 79h | 15h | 48h | 00h |
| 270 | 0 | 40 | 70.77326343 | 10.8 | 28h | 00h | A7h | 97h | F4h | C5h | 46h | 00h |
| 311.04 | 0 | 36 | 73.37771953 | 11.19744 | 24h | 00h | 1Ch | 3Ah | B2h | 60h | 49h | 00h |
| 312.5 | 0 | 36 | 73.72214941 | 11.25 | 24h | 00h | A3h | C8h | DEh | B8h | 49h | 00h |
| 322.265625 | 0 | 34 | 71.80230177 | 10.95703125 | 22h | 00h | 14h | A6h | 63h | CDh | 47h | 00h |
| 400 | 0 | 27 | 70.77326343 | 10.8 | 1Bh | 00h | A7h | 97h | F4h | C5h | 46h | 00h |
| 425 | 0 | 26 | 72.41153342 | 11.05 | 1Ah | 00h | 17h | 41h | 5Ah | 69h | 48h | 00h |
| 491.52 | 0 | 22 | 70.86133683 | 10.81344 | 16h | 00h | 04h | 92h | 80h | DCh | 46h | 00h |
| 500 | 0 | 22 | 72.08387942 | 11 | 16h | 00h | 34h | 1Fh | 79h | 15h | 48h | 00h |
| 614.4 | 0 | 18 | 72.47182176 | 11.0592 | 12h | 00h | 84h | 4Fh | C9h | 78h | 48h | 00h |
| 622.08 | 0 | 18 | 73.37771953 | 11.19744 | 12h | 00h | 1Ch | 3Ah | B2h | 60h | 49h | 00h |
| 644.53125 | 0 | 17 | 71.80230177 | 10.95703125 | 11h | 00h | 14h | A6h | 63h | CDh | 47h | 00h |
| 750 | 0 | 15 | 73.72214941 | 11.25 | 0Fh | 00h | A3h | C8h | DEh | B8h | 49h | 00h |
| 800 | 0 | 14 | 73.39449541 | 11.2 | 0Eh | 00h | C0h | A6h | FDh | 64h | 49h | 00h |

### 5.12 I2C Interface

Configuration and operation of the Si569 is controlled by reading and writing to the RAM space using the I2C interface. The device operates in slave mode with 7-bit addressing and can operate in Standard-Mode (100 kbps), Fast-Mode (400 kbps), or Fast-Mode Plus (1 Mbps). Burst data transfer with auto address increments are also supported.

The I2C bus consists of a bidirectional serial data line (SDA) and a serial clock input (SCL). Both the SDA and SCL pins must be connected to the VDD supply via an external pull-up as recommended by the I2C specification. The Si569 7-bit I2C slave address is user-customized during the part number configuration process.

Data is transferred MSB first in 8-bit words as specified by the I2C specification. A write command consists of a 7-bit device (slave) address + a write bit, an 8-bit register address, and 8 bits of data as shown in the figure below.

A write burst operation is also shown where every additional data word is written using an auto-incremented address.

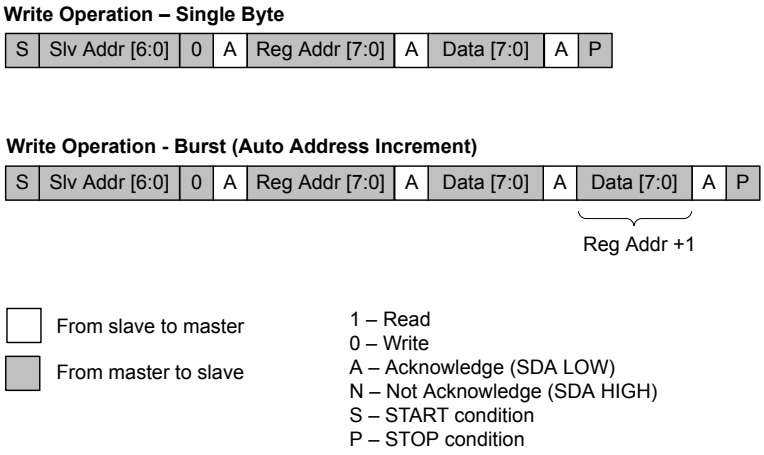



**Figure 5.3. I2C Write Operation**

A read operation is performed in two stages. A data write is used to set the register address, then a data read is performed to retrieve the data from the set address. A read burst operation is also supported. This is shown in the figure below.

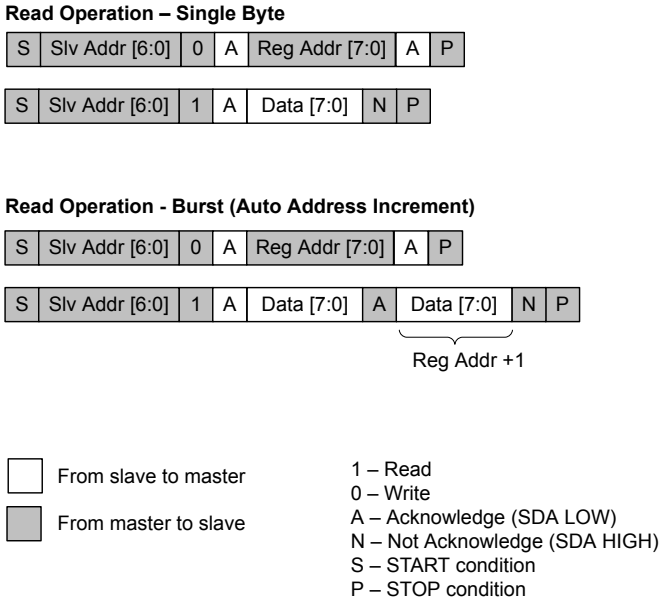



**Figure 5.4. I2C Read Operation**

The timing specifications and timing diagram for the I2C bus is compatible with the I2C-Bus standard. SDA timeout is supported for compatibility with SMBus interfaces.

The I2C bus can be operated at a bus voltage of 1.71 to 3.63 V and should be the same voltage as the Si569 VDD.

# 6. Package Outline

## 6.1 Package Outline (5x7 mm)

The figure below illustrates the package details for the 5x7 mm Si569. The table below lists the values for the dimensions shown in the illustration.

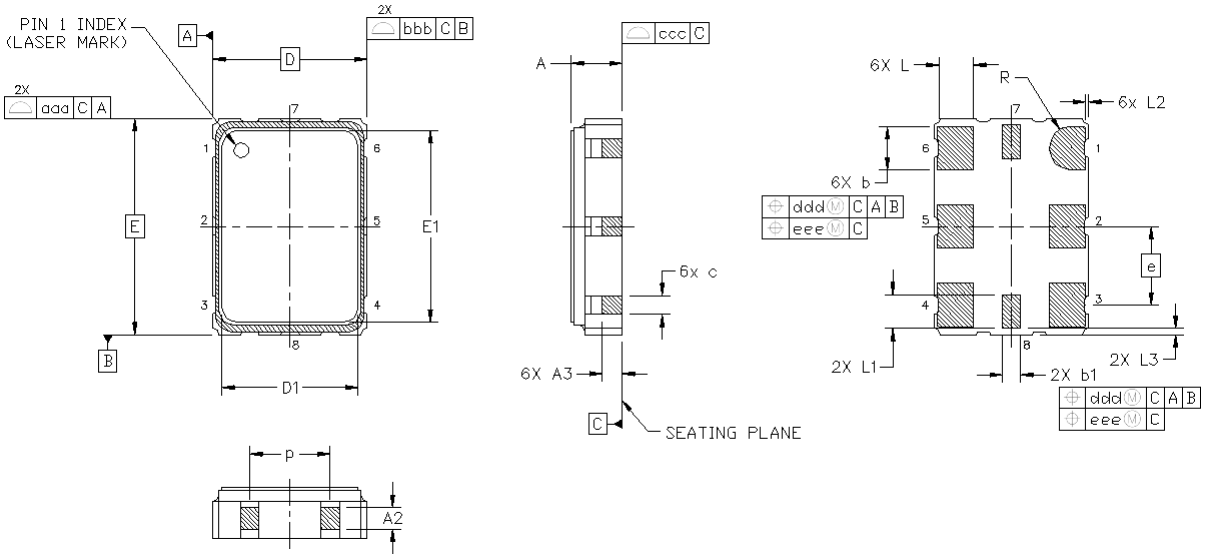



**Figure 6.1. Si569 (5x7 mm) Outline Diagram**

**Table 6.1. Package Diagram Dimensions (mm)**

| Dimension | Min | Nom | Max | | Dimension | Min | Nom | Max |
|---|---|---|---|---|---|---|---|---|
| A | 1.07 | 1.18 | 1.33 | | E1 | 6.10 | 6.20 | 6.30 |
| A2 | 0.40 | 0.50 | 0.60 | | L | 1.07 | 1.17 | 1.27 |
| A3 | 0.45 | 0.55 | 0.65 | | L1 | 1.00 | 1.10 | 1.20 |
| b | 1.30 | 1.40 | 1.50 | | p | 1.70 | -- | 1.90 |
| b1 | 0.50 | 0.60 | 0.70 | | R | 0.70 REF | | |
| c | 0.50 | 0.60 | 0.70 | | aaa | 0.15 | | |
| D | 5.00 BSC | | | | bbb | 0.15 | | |
| D1 | 4.30 | 4.40 | 4.50 | | ccc | 0.08 | | |
| e | 2.54 BSC | | | | ddd | 0.10 | | |
| E | 7.00 BSC | | | | eee | 0.05 | | |

**Notes:**

1. All dimensions shown are in millimeters (mm) unless otherwise noted.
2. Dimensioning and Tolerancing per ANSI Y14.5M-1994.

### 6.2 Package Outline (3.2x5 mm)

The figure below illustrates the package details for the 5x3.2 mm Si569. The table below lists the values for the dimensions shown in the illustration.

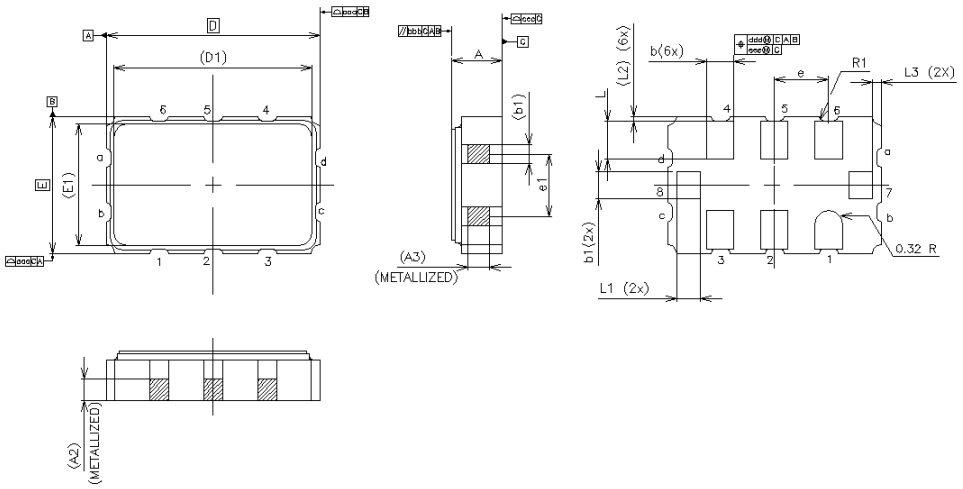



**Figure 6.2. Si569 (3.2x5 mm) Outline Diagram**

**Table 6.2. Package Diagram Dimensions (mm)**

| Dimension | MIN | NOM | MAX | | Dimension | MIN | NOM | MAX |
|---|---|---|---|---|---|---|---|---|
| A | 1.02 | 1.17 | 1.33 | | E1 | | 2.85 BSC | |
| A2 | 0.50 | 0.55 | 0.60 | | L | 0.8 | 0.9 | 1.0 |
| A3 | 0.45 | 0.50 | 0.55 | | L1 | 0.45 | 0.55 | 0.65 |
| b | 0.54 | 0.64 | 0.74 | | L2 | 0.05 | 0.10 | 0.15 |
| b1 | 0.54 | 0.64 | 0.75 | | L3 | 0.15 | 0.20 | 0.25 |
| D | | 5.00 BSC | | | aaa | | 0.15 | |
| D1 | | 4.65 BSC | | | bbb | | 0.15 | |
| e | | 1.27 BSC | | | ccc | | 0.08 | |
| e1 | | 1.625 TYP | | | ddd | | 0.10 | |
| E | | 3.20 BSC | | | eee | | 0.05 | |

**Notes:**

1. All dimensions shown are in millimeters (mm) unless otherwise noted.
2. Dimensioning and Tolerancing per ANSI Y14.5M-1994.

# 7. PCB Land Pattern

## 7.1 PCB Land Pattern (5x7 mm)

The figure below illustrates the 5x7 mm PCB land pattern for the Si569. The table below lists the values for the dimensions shown in the illustration.

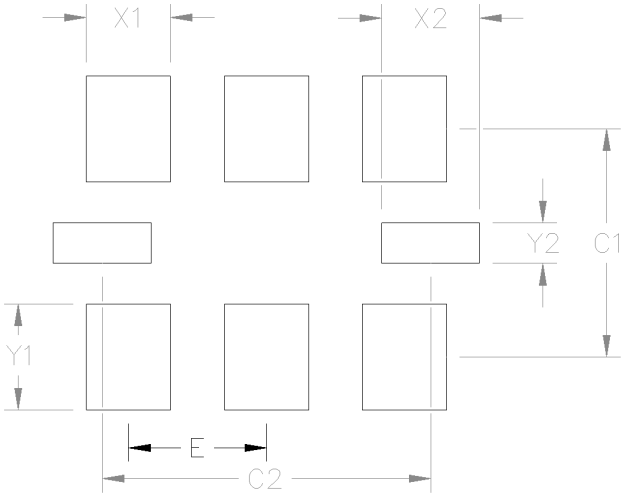



**Figure 7.1. Si569 (5x7 mm) PCB Land Pattern**

**Table 7.1. PCB Land Pattern Dimensions (mm)**

| Dimension | (mm) | | Dimension | (mm) |
|---|---|---|---|---|
| C1 | 4.20 | | Y1 | 1.95 |
| C2 | 6.05 | | X2 | 1.80 |
| E | 2.54 | | Y2 | 0.75 |
| X1 | 1.55 | | | |

**Notes:**

**General**

1. All dimensions shown are in millimeters (mm) unless otherwise noted.
2. Dimensioning and Tolerancing is per the ANSI Y14.5M-1994 specification.
3. This Land Pattern Design is based on the IPC-7351 guidelines.
4. All dimensions shown are at Maximum Material Condition (MMC). Least Material Condition (LMC) is calculated based on a Fabrication Allowance of 0.05 mm.

**Solder Mask Design**

1. All metal pads are to be non-solder mask defined (NSMD). Clearance between the solder mask and the metal pad is to be 60 µm minimum, all the way around the pad.

**Stencil Design**

1. A stainless steel, laser-cut and electro-polished stencil with trapezoidal walls should be used to assure good solder paste release.
2. The stencil thickness should be 0.125 mm (5 mils).
3. The ratio of stencil aperture to land pad size should be 1:1.

**Card Assembly**

1. A No-Clean, Type-3 solder paste is recommended.
2. The recommended card reflow profile is per the JEDEC/IPC J-STD-020D specification for Small Body Components.

### 7.2 PCB Land Pattern (3.2x5 mm)

The figure below illustrates the 3.2x5.0 mm PCB land pattern for the Si569. The table below lists the values for the dimensions shown in the illustration.

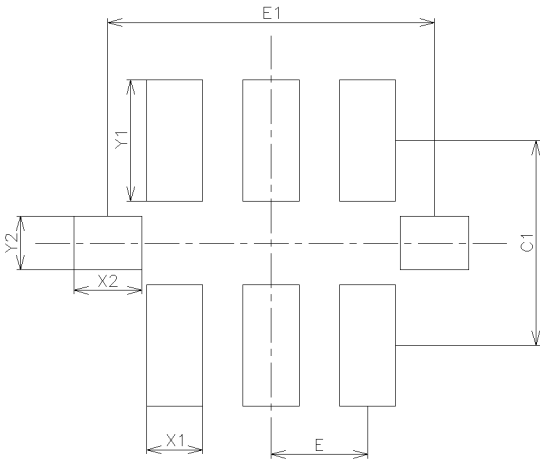



**Figure 7.2. Si569 (3.2x5 mm) PCB Land Pattern**

**Table 7.2. PCB Land Pattern Dimensions (mm)**

| Dimension | (mm) | | Dimension | (mm) |
|---|---|---|---|---|
| C1 | 2.70 | | X2 | 0.90 |
| E | 1.27 | | Y1 | 1.60 |
| E1 | 4.30 | | Y2 | 0.70 |
| X1 | 0.74 | | | |

**Notes:**

**General**

1. All dimensions shown are in millimeters (mm) unless otherwise noted.
2. Dimensioning and Tolerancing is per the ANSI Y14.5M-1994 specification.
3. This Land Pattern Design is based on the IPC-7351 guidelines.
4. All dimensions shown are at Maximum Material Condition (MMC). Least Material Condition (LMC) is calculated based on a Fabrication Allowance of 0.05 mm.

**Solder Mask Design**

1. All metal pads are to be non-solder mask defined (NSMD). Clearance between the solder mask and the metal pad is to be 60 µm minimum, all the way around the pad.

**Stencil Design**

1. A stainless steel, laser-cut and electro-polished stencil with trapezoidal walls should be used to assure good solder paste release.
2. The stencil thickness should be 0.125 mm (5 mils).
3. The ratio of stencil aperture to land pad size should be 1:1.

**Card Assembly**

1. A No-Clean, Type-3 solder paste is recommended.
2. The recommended card reflow profile is per the JEDEC/IPC J-STD-020C specification for Small Body Components.

## 8. Top Marking

The figure below illustrates the mark specification for the Si569. The table below lists the line information.

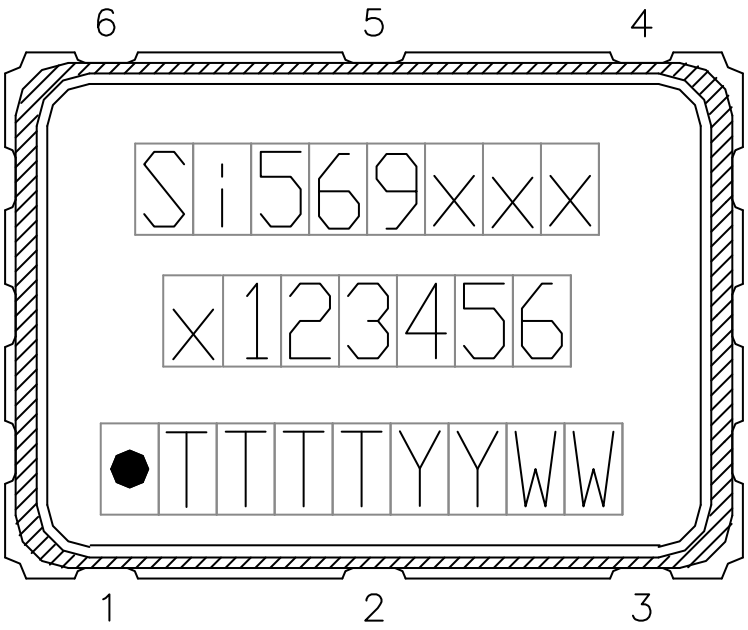



**Figure 8.1. Mark Specification**

**Table 8.1. Si569 Top Mark Description**

| Line | Position | Description |
|---|---|---|
| 1 | 1–8 | "Si569", xxx = Ordering Option 1, Option 2, Option 3 (e.g. Si569AAA) |
| 2 | 1 | x = Frequency Range Supported as described in the 1. Ordering Guide |
| | 2–7 | 6-digit custom Frequency Code as described in the 1. Ordering Guide |
| 3 | **Trace Code** | |
| | Position 1 | Pin 1 orientation mark (dot) |
| | Position 2 | Product Revision (B) |
| | Position 3–5 | Tiny Trace Code (3 alphanumeric characters per assembly release instructions) |
| | Position 6–7 | Year (last two digits of the year), to be assigned by assembly site (ex: 2017 = 17) |
| | Position 8–9 | Calendar Work Week number (1–53), to be assigned by assembly site |

# 9. Revision History

### Revision 1.1

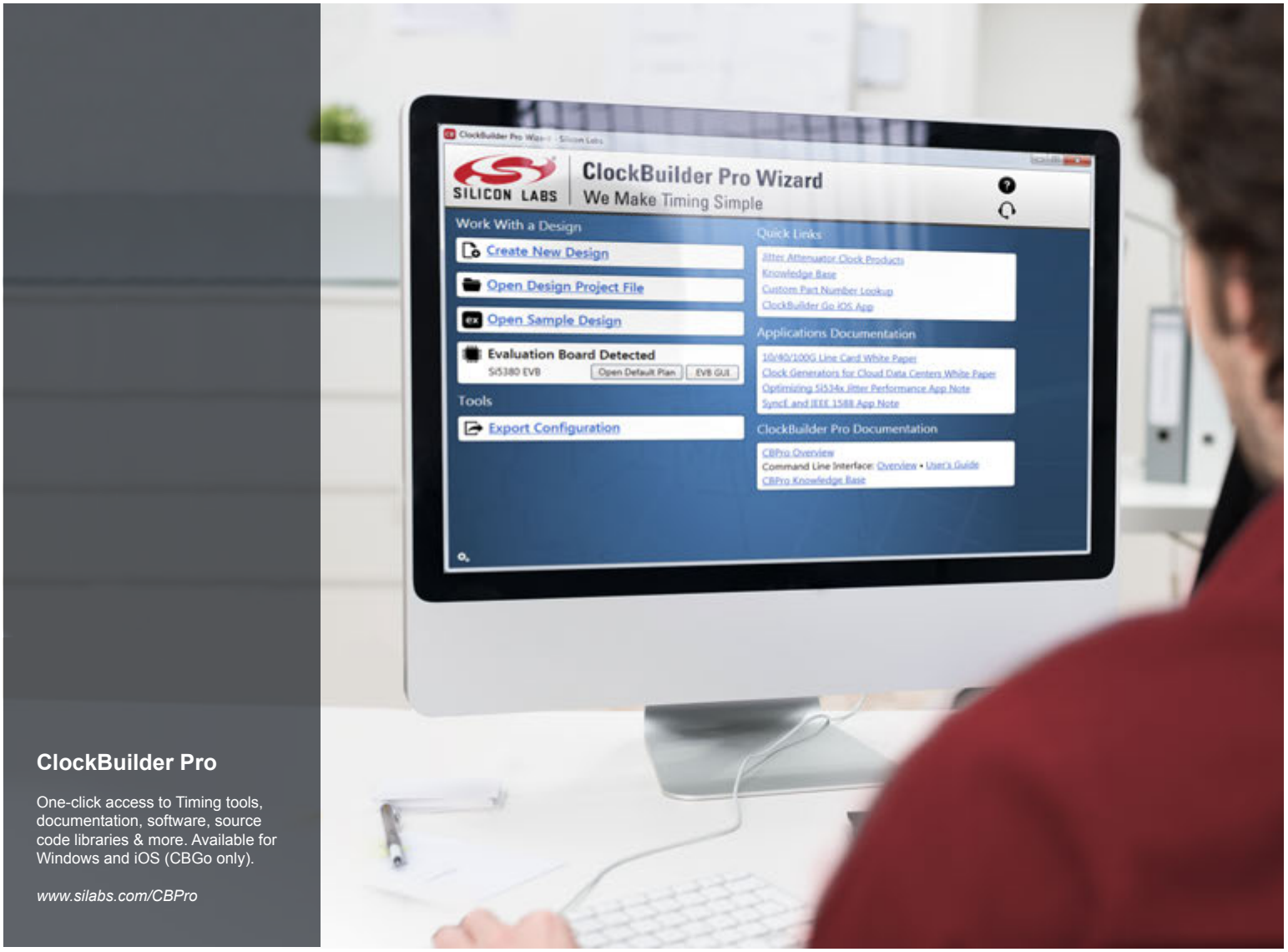
September, 2018

- Updated Electrical Specifications table to include high drive LVDS swing.
- Added section 5.8 Configuring High Drive LVDS Swing.

### Revision 1.0

June, 2018

- Initial release.




## ClockBuilder Pro

One-click access to Timing tools, documentation, software, source code libraries & more. Available for Windows and iOS (CBGo only).

*www.silabs.com/CBPro*

**Timing Portfolio**
*www.silabs.com/timing*

**SW/HW**
*www.silabs.com/CBPro*

**Quality**
*www.silabs.com/quality*

**Support and Community**
*community.silabs.com*

**Disclaimer**
Silicon Labs intends to provide customers with the latest, accurate, and in-depth documentation of all peripherals and modules available for system and software implementers using or intending to use the Silicon Labs products. Characterization data, available modules and peripherals, memory sizes and memory addresses refer to each specific device, and "Typical" parameters provided can and do vary in different applications. Application examples described herein are for illustrative purposes only. Silicon Labs reserves the right to make changes without further notice and limitation to product information, specifications, and descriptions herein, and does not give warranties as to the accuracy or completeness of the included information. Silicon Labs shall have no liability for the consequences of use of the information supplied herein. This document does not imply or express copyright licenses granted hereunder to design or fabricate any integrated circuits. The products are not designed or authorized to be used within any Life Support System without the specific written consent of Silicon Labs. A "Life Support System" is any product or system intended to support or sustain life and/or health, which, if it fails, can be reasonably expected to result in significant personal injury or death. Silicon Labs products are not designed or authorized for military applications. Silicon Labs products shall under no circumstances be used in weapons of mass destruction including (but not limited to) nuclear, biological or chemical weapons, or missiles capable of delivering such weapons.

**Trademark Information**
Silicon Laboratories Inc.® , Silicon Laboratories®, Silicon Labs®, SiLabs® and the Silicon Labs logo®, Bluegiga®, Bluegiga Logo®, Clockbuilder®, CMEMS®, DSPLL®, EFM®, EFM32®, EFR, Ember®, Energy Micro, Energy Micro logo and combinations thereof, "the world's most energy friendly microcontrollers", Ember®, EZLink®, EZRadio®, EZRadioPRO®, Gecko®, ISOmodem®, Micrium, Precision32®, ProSLIC®, Simplicity Studio®, SiPHY®, Telegesis, the Telegesis Logo®, USBXpress®, Zentri, Z-Wave, and others are trademarks or registered trademarks of Silicon Labs. ARM, CORTEX, Cortex-M3 and THUMB are trademarks or registered trademarks of ARM Holdings. Keil is a registered trademark of ARM Limited. All other products or brand names mentioned herein are trademarks of their respective holders.

**Silicon Laboratories Inc.**
**400 West Cesar Chavez**
**Austin, TX 78701**
**USA**

**http://www.silabs.com**