# THE DATASHEET OF

# AD9522-5BCPZ-REEL7

www.win-source.net 0086-755-83957316

# 12 LVDS/24 CMOS Output Clock Generator

Data Sheet

# AD9522-5

## FEATURES

- Low phase noise, phase-locked loop (PLL)
  - Supports external 3.3 V/5 V voltage controlled oscillator (VCO)/VCXO to 2.4 GHz
  - 1 differential or 2 single-ended reference inputs
  - Accepts CMOS, LVPECL, or LVDS references to 250 MHz
  - Accepts 16.62 MHz to 33.3 MHz crystal for reference input
  - Optional reference clock doubler
  - Reference monitoring capability
  - Revertive automatic and manual reference switchover/holdover modes
  - Glitch-free switchover between references
  - Automatic recovery from holdover
  - Digital or analog lock detect, selectable
  - Optional zero delay operation
- Twelve 800 MHz LVDS outputs divided into 4 groups
  - Each group of 3 has a 1-to-32 divider with phase delay
  - Additive output jitter as low as 242 fs rms
  - Channel-to-channel skew grouped outputs < 60 ps
  - Each LVDS output can be configured as 2 CMOS outputs (for $f_{OUT} \leq 250$ MHz)
- Automatic synchronization of all outputs on power-up
- Manual synchronization of outputs as needed
- SPI- and I²C-compatible serial control port
- 64-lead LFCSP
- Nonvolatile EEPROM stores configuration settings

## APPLICATIONS

- Low jitter, low phase noise clock distribution
- Clock generation and translation for SONET, 10Ge, 10G FC, and other 10 Gbps protocols
- Forward error correction (G.710)
- Clocking high speed ADCs, DACs, DDSs, DDCs, DUCs, MxFEs
- High performance wireless transceivers
- ATE and high performance instrumentation
- Broadband infrastructures

## GENERAL DESCRIPTION

The AD9522-5[1] provides a multioutput clock distribution function with subpicosecond jitter performance, along with an on-chip PLL that can be used with an external VCO.

## FUNCTIONAL BLOCK DIAGRAM

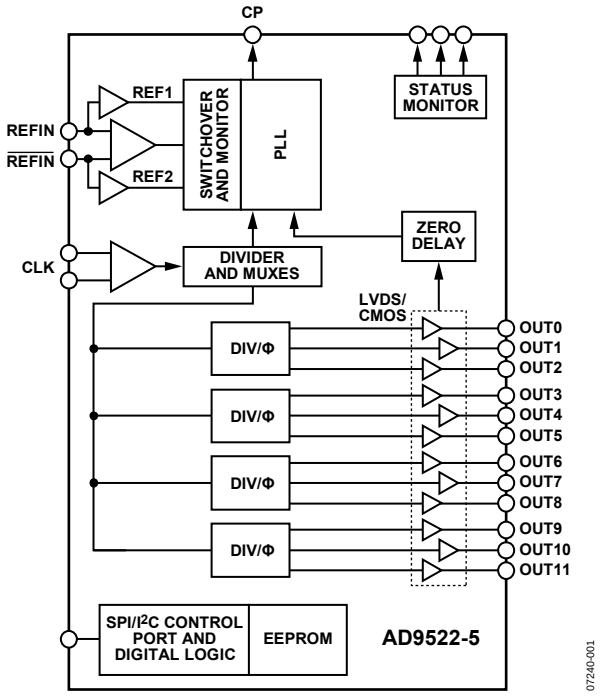


*Figure 1.*

The AD9522 serial interface supports both SPI and I²C® ports. An in-package EEPROM can be programmed through the serial interface and store user-defined register settings for power-up and chip reset.

The AD9522 features 12 LVDS outputs in four groups. Any of the 800 MHz LVDS outputs can be reconfigured as two 250 MHz CMOS outputs.

Each group of outputs has a divider that allows both the divide ratio (from 1 to 32) and the phase (coarse delay) to be set.

The AD9522 is available in a 64-lead LFCSP and can be operated from a single 3.3 V supply. The external VCO can have an operating voltage up to 5.5 V.

The AD9522 is specified for operation over the standard industrial range of −40°C to +85°C.

The AD9520-5 is an equivalent part to the AD9522-5 featuring LVPECL/CMOS drivers instead of LVDS/CMOS drivers.

[1] The AD9522 is used throughout this data sheet to refer to all the members of the AD9522 family. However, when AD9522-5 is used, it is referring to that specific member of the AD9522 family.

Rev. A

Document Feedback

Information furnished by Analog Devices is believed to be accurate and reliable. However, no responsibility is assumed by Analog Devices for its use, nor for any infringements of patents or other rights of third parties that may result from its use. Specifications subject to change without notice. No license is granted by implication or otherwise under any patent or patent rights of Analog Devices. Trademarks and registered trademarks are the property of their respective owners.

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.
Tel: 781.329.4700 ©2008–2015 Analog Devices, Inc. All rights reserved.
Technical Support www.analog.com

# TABLE OF CONTENTS


Features ... 1
Applications ... 1
General Description ... 1
Functional Block Diagram ... 1
Revision History ... 4
Specifications ... 5
Power Supply Requirements ... 5
PLL Characteristics ... 5
Clock Inputs ... 8
Clock Outputs ... 8
Timing Characteristics ... 9
Timing Diagrams ... 9
Clock Output Additive Phase Noise (Distribution Only; VCO Divider Not Used) ... 10
Clock Output Absolute Time Jitter (Clock Generation Using External VCXO) ... 11
Clock Output Additive Time Jitter (VCO Divider Not Used) ... 11
Clock Output Additive Time Jitter (VCO Divider Used) ... 12
Serial Control Port—SPI Mode ... 12
Serial Control Port—I²C Mode ... 13
$\overline{PD}$, $\overline{SYNC}$, and $\overline{RESET}$ Pins ... 14
Serial Port Setup Pins: SP1, SP0 ... 14
LD, STATUS, and REFMON Pins ... 14
Power Dissipation ... 15
Absolute Maximum Ratings ... 16
Thermal Resistance ... 16
ESD Caution ... 16
Pin Configuration and Function Descriptions ... 17
Typical Performance Characteristics ... 20
Terminology ... 24
Detailed Block Diagram ... 25
Theory of Operation ... 26
Operational Configurations ... 26
Mode 1: Clock Distribution or External VCO < 1600 MHz ... 26
Mode 2: High Frequency Clock Distribution—CLK or External VCO > 1600 MHz ... 28
Phase-Locked Loop (PLL) ... 30
Configuration of the PLL ... 30
Phase Frequency Detector (PFD) ... 30
Charge Pump (CP) ... 30
PLL External Loop Filter ... 31
PLL Reference Inputs ... 31
Reference Switchover ... 31
Reference Divider R ... 32
VCO/VCXO Feedback Divider N: P, A, B ... 32
Digital Lock Detect (DLD) ... 33
Analog Lock Detect (ALD) ... 33
Current Source Digital Lock Detect (CSDLD) ... 33
External VCXO/VCO Clock Input (CLK/$\overline{CLK}$) ... 34
Holdover ... 34
External/Manual Holdover Mode ... 34
Automatic/Internal Holdover Mode ... 35
Frequency Status Monitors ... 36
Zero Delay Operation ... 37
Clock Distribution ... 38
Operation Modes ... 38
Clock Frequency Division ... 38
VCO Divider ... 39
Channel Dividers ... 39
Synchronizing the Outputs—$\overline{SYNC}$ Function ... 41
LVDS Output Drivers ... 42
CMOS Output Drivers ... 43
Reset Modes ... 43
Power-On Reset ... 43
Hardware Reset via the $\overline{RESET}$ Pin ... 43
Soft Reset via the Serial Port ... 43
Soft Reset to Settings in EEPROM when EEPROM Pin = 0 via the Serial Port ... 43
Power-Down Modes ... 43
Chip Power-Down via $\overline{PD}$ ... 43
PLL Power-Down ... 44
Distribution Power-Down ... 44
Individual Clock Output Power-Down ... 44
Individual Clock Channel Power-Down ... 44
Serial Control Port ... 45
SPI/I²C Port Selection ... 45
I²C Serial Port Operation ... 45
I²C Bus Characteristics ... 45
Data Transfer Process ... 46

Data Transfer Format ..... 47
I²C Serial Port Timing ..... 47
SPI Serial Port Operation ..... 48
Pin Descriptions ..... 48
SPI Mode Operation ..... 48
Communication Cycle—Instruction Plus Data ..... 48
Write ..... 48
Read ..... 48
SPI Instruction Word (16 Bits) ..... 49
SPI MSB/LSB First Transfers ..... 49
EEPROM Operations ..... 52
Writing to the EEPROM ..... 52
Reading from the EEPROM ..... 52
Programming the EEPROM Buffer Segment ..... 52
Register Section Definition Group ..... 53
IO_UPDATE (Operational Code 0x80) ..... 53
End-of-Data (Operational Code 0xFF) ..... 53
Pseudo-End-of-Data (Operational Code 0xFE) ..... 53
Thermal Performance ..... 54
Register Map ..... 55
Register Map Descriptions ..... 59
Applications Information ..... 72
Frequency Planning Using the AD9522 ..... 72
Using the AD9522 Outputs for ADC Clock Applications ..... 72
LVDS Clock Distribution ..... 72
CMOS Clock Distribution ..... 73
Outline Dimensions ..... 74
Ordering Guide ..... 74

## REVISION HISTORY

**3/15—Rev. 0 to Rev. A**

Changes to Features Section ... 1
Changes to Table 1 and Table 2 ... 5
Change to Input Frequency Parameter, Table 3 ... 8
Changes to Table 4 ... 8
Changes to SDIO, SDO (Outputs) Parameter, Test Conditions/Comments Column, Table 10 ... 12
Changes to Table 14 ... 14
Change to Junction Temperature Parameter, Table 16 ... 16
Changes to Pin 22 Description Column, Table 18 ... 18
Changes to Figure 26 ... 27
Changes to Table 22 ... 28
Changes to Figure 27 ... 29
Changes to Configuration of the PLL Section and Charge Pump (CP) Section ... 30
Changes to PLL Reference Inputs Section and Reference Switchover Section ... 31
Changes to Reference Divider R Section, Prescaler Section, A and B Counters Section, and R and N Divider Delays Section ... 32
Changes to Table 25 and Current Source Digital Lock Detect (CSDLD) Section ... 33
Changes to External VCXO/VCO Clock Input (CLK/$\overline{\text{CLK}}$) and Holdover Section ... 34
Changes to Frequency Status Monitors Section ... 36
Changes to Clock Distribution Section ... 38
Added Channel Divider Maximum Frequency Section ... 39
Changes to Duty Cycle and Duty-Cycle Correction Section ... 39
Changes to Table 31 ... 40
Changes to Synchronizing the Outputs—$\overline{\text{SYNC}}$ Function Section ... 41
Changes to Power-On Reset Section, Hardware Reset via the $\overline{\text{RESET}}$ Pin Section, and Soft Reset via the Serial Port Section ... 43
Changes to Pin Descriptions Section and SPI Mode Operation Section ... 48
Changes to Figure 52, Figure 53 Caption, and Figure 54 ... 50
Changes to EEPROM Operation Section, Writing to EEPROM section, and Reading from EEPROM section ... 52
Changes to Table 43 ... 55
Changes to Table 44 and Table 45 ... 59
Changes to Table 47 ... 61
Changes to Table 49 ... 68
Changes to Table 52 ... 71
Change to Frequency Planning Using the AD9522 Section ... 72
Updated Outline Dimensions ... 74

**12/08—Revision 0: Initial Version**

# SPECIFICATIONS

Typical (typ) is given for VS = 3.3 V ± 5%; VS ≤ VCP ≤ 5.25 V; $T_A$ = 25°C; RSET = 4.12 kΩ; CPRSET = 5.1 kΩ, unless otherwise noted. Minimum (min) and maximum (max) values are given over full VS and $T_A$ (−40°C to +85°C) variation.

## POWER SUPPLY REQUIREMENTS

**Table 1.**

| Parameter | Min | Typ | Max | Unit | Test Conditions/Comments |
|---|---|---|---|---|---|
| VS | 3.135 | 3.3 | 3.465 | V | 3.3 V ± 5% |
| VCP | VS | | 5.25 | V | This supply is usually at the same voltage as VS; set VCP = 5.0 V ± 5% only if connecting a 5 V external VCO/VCXO |
| RSET Pin Resistor | | 4.12 | | kΩ | Sets internal biasing currents; connect to ground |
| CPRSET Pin Resistor | | 5.1 | | kΩ | Sets internal CP current range, nominally 4.8 mA (CP_Isb = 600 µA); actual current can be calculated by CP_Isb = 3.06/CPRSET; connect to ground |

## PLL CHARACTERISTICS

**Table 2.**

| Parameter | Min | Typ | Max | Unit | Test Conditions/Comments |
|---|---|---|---|---|---|
| REFERENCE INPUTS | | | | | |
| Differential Mode (REFIN, $\overline{\text{REFIN}}$) | | | | | Differential mode (can accommodate single-ended input by ac grounding the unused complementary input) |
| Input Frequency | 0 | | 250 | MHz | Frequencies below about 1 MHz must be dc-coupled; be careful to match $V_{CM}$ (self-bias voltage) |
| Input Sensitivity | | 280 | | mV p-p | PLL figure of merit (FOM) increases with increasing slew rate (see Figure 11); the input sensitivity is sufficient for ac-coupled LVDS and LVPECL signals |
| Self-Bias Voltage, REFIN | 1.35 | 1.60 | 1.75 | V | Self-bias voltage of REFIN[1] |
| Self-Bias Voltage, $\overline{\text{REFIN}}$ | 1.30 | 1.50 | 1.60 | V | Self-bias voltage of $\overline{\text{REFIN}}$[1] |
| Input Resistance, REFIN | 4.0 | 4.8 | 5.9 | kΩ | Self-biased[1] |
| Input Resistance, $\overline{\text{REFIN}}$ | 4.4 | 5.3 | 6.4 | kΩ | Self-biased[1] |
| Dual Single-Ended Mode (REF1, REF2) | | | | | Two single-ended CMOS-compatible inputs |
| Input Frequency (AC-Coupled with DC Offset Off) | 10 | | 250 | MHz | Slew rate must be > 50 V/µs |
| Input Frequency (AC-Coupled with DC Offset On) | | | 250 | MHz | Slew rate must be > 50 V/µs, and input amplitude sensitivity specification must be met; see input sensitivity |
| Input Frequency (DC-Coupled) | 0 | | 250 | MHz | Slew rate > 50 V/µs; CMOS levels |
| Input Sensitivity (AC-Coupled with DC Offset Off) | 0.55 | | 3.28 | V p-p | VIH must not exceed VS |
| Input Sensitivity (AC-Coupled with DC Offset On) | 1.5 | | 2.78 | V p-p | VIH must not exceed VS |
| Input Logic High, DC Offset Off | 2.0 | | | V | |
| Input Logic Low, DC Offset Off | | | 0.8 | V | |
| Input Current | −100 | | +100 | µA | |
| Input Capacitance | | 2 | | pF | Each pin, REFIN (REF1)/$\overline{\text{REFIN}}$ (REF2) |
| Pulse Width High/Low | 1.8 | | | ns | Amount of time a square wave is high/low determines the allowable input duty cycle |
| Crystal Oscillator | | | | | |
| Crystal Resonator Frequency Range | 16.62 | | 33.33 | MHz | |
| Maximum Crystal Motional Resistance | | | 30 | Ω | |
| PHASE/FREQUENCY DETECTOR (PFD) | | | | | |
| PFD Input Frequency | | | 100 | MHz | Antibacklash pulse width = 1.3 ns, 2.9 ns |
| | | | 45 | MHz | Antibacklash pulse width = 6.0 ns |
| Reference Input Clock Doubler Frequency | 0.004 | | 50 | MHz | Antibacklash pulse width = 1.3 ns, 2.9 ns |
| Antibacklash Pulse Width | | 1.3 | | ns | Register 0x017[1:0] = 01b |
| | | 2.9 | | ns | Register 0x017[1:0] = 00b; Register 0x017[1:0] = 11b |
| | | 6.0 | | ns | Register 0x017[1:0] = 10b |

| Parameter | Min | Typ | Max | Unit | Test Conditions/Comments |
|---|---|---|---|---|---|
| CHARGE PUMP (CP) | | | | | |
| $I_{CP}$ Sink/Source | | | | | Programmable |
| High Value | | 4.8 | | mA | With CPRSET = 5.1 kΩ; higher $I_{CP}$ is possible by changing CPRSET |
| Low Value | | 0.60 | | mA | With CPRSET = 5.1 kΩ; lower $I_{CP}$ is possible by changing CPRSET |
| Absolute Accuracy | | 2.5 | | % | Charge pump voltage set to $V_{CP}/2$ |
| CPRSET Range | 2.7 | | 10 | kΩ | |
| $I_{CP}$ High Impedance Mode Leakage | | 1 | | nA | |
| Sink-and-Source Current Matching | | 1 | | % | 0.5 V < $V_{CP}$ < VCP − 0.5 V; $V_{CP}$ is the voltage on the CP (charge pump) pin; VCP is the voltage on the VCP power supply pin |
| $I_{CP}$ vs. $V_{CP}$ | | 1.5 | | % | 0.5 V < $V_{CP}$ < VCP − 0.5 V |
| $I_{CP}$ vs. Temperature | | 2 | | % | $V_{CP}$ = VCP/2 V |
| PRESCALER (PART OF N DIVIDER) | | | | | |
| Prescaler Input Frequency | | | | | |
| P = 1 FD | | | 300 | MHz | |
| P = 2 FD | | | 600 | MHz | |
| P = 3 FD | | | 900 | MHz | |
| P = 2 DM (2/3) | | | 200 | MHz | |
| P = 4 DM (4/5) | | | 1000 | MHz | |
| P = 8 DM (8/9) | | | 2400 | MHz | |
| P = 16 DM (16/17) | | | 3000 | MHz | |
| P = 32 DM (32/33) | | | 3000 | MHz | |
| Prescaler Output Frequency | | | 300 | MHz | A, B counter input frequency (prescaler input frequency divided by P) |
| PLL N DIVIDER DELAY | | | | | Register 0x019[2:0]; see Table 47 |
| 000 | | Off | | | |
| 001 | | 385 | | ps | |
| 010 | | 504 | | ps | |
| 011 | | 623 | | ps | |
| 100 | | 743 | | ps | |
| 101 | | 866 | | ps | |
| 110 | | 989 | | ps | |
| 111 | | 1112 | | ps | |
| PLL R DIVIDER DELAY | | | | | Register 0x019[5:3]; see Table 47 |
| 000 | | Off | | | |
| 001 | | 365 | | ps | |
| 010 | | 486 | | ps | |
| 011 | | 608 | | ps | |
| 100 | | 730 | | ps | |
| 101 | | 852 | | ps | |
| 110 | | 976 | | ps | |
| 111 | | 1101 | | ps | |
| PHASE OFFSET IN ZERO DELAY | | | | | REF refers to REFIN (REF1)/$\overline{\text{REFIN}}$ (REF2) |
| Phase Offset (REF-to-LVDS Clock Output Pins) in Zero Delay Mode | 1890 | 2348 | 3026 | ps | When N delay and R delay are bypassed |
| Phase Offset (REF-to-LVDS Clock Output Pins) in Zero Delay Mode | 900 | 1217 | 1695 | ps | When N delay = Setting 111 and R delay is bypassed |

| Parameter | Min | Typ | Max | Unit | Test Conditions/Comments |
|---|---|---|---|---|---|
| NOISE CHARACTERISTICS | | | | | |
| In-Band Phase Noise of the Charge Pump/ Phase Frequency Detector (In-Band Means Within the LBW of the PLL) | | | | | The PLL in-band phase noise floor is estimated by measuring the in-band phase noise at the output of the VCO and subtracting 20 log(N) (where N is the value of the N divider) |
| At 500 kHz PFD Frequency | | −165 | | dBc/Hz | |
| At 1 MHz PFD Frequency | | −162 | | dBc/Hz | |
| At 10 MHz PFD Frequency | | −152 | | dBc/Hz | |
| At 50 MHz PFD Frequency | | −144 | | dBc/Hz | |
| PLL Figure of Merit (FOM) | | −222 | | dBc/Hz | Reference slew rate > 0.5 V/ns; FOM + 10 log($f_{PFD}$) is an approximation of the PFD/CP in-band phase noise (in the flat region) inside the PLL loop bandwidth; when running closed-loop, the phase noise, as observed at the VCO output, is increased by 20 log(N); PLL figure of merit decreases with decreasing slew rate; see Figure 11 |
| PLL DIGITAL LOCK DETECT WINDOW[2] | | | | | Signal available at the LD, STATUS, and REFMON pins when selected by appropriate register settings; lock detect window settings can be varied by changing the CPRSET resistor |
| Lock Threshold (Coincidence of Edges) | | | | | Selected by Register 0x017[1:0] and Register 0x018[4] (this is the threshold to go from unlock to lock) |
| Low Range (ABP 1.3 ns, 2.9 ns) | | 3.5 | | ns | Register 0x017[1:0] = 00b, 01b, 11b; Register 0x018[4] = 1b |
| High Range (ABP 1.3 ns, 2.9 ns) | | 7.5 | | ns | Register 0x017[1:0] = 00b, 01b, 11b; Register 0x018[4] = 0b |
| High Range (ABP 6.0 ns) | | 3.5 | | ns | Register 0x017[1:0] = 10b; Register 0x018[4] = 0b |
| Unlock Threshold (Hysteresis)[2] | | | | | Selected by Register 0x017[1:0] and Register 0x018[4] (this is the threshold to go from lock to unlock) |
| Low Range (ABP 1.3 ns, 2.9 ns) | | 7 | | ns | Register 0x017[1:0] = 00b, 01b, 11b; Register 0x018[4] = 1b |
| High Range (ABP 1.3 ns, 2.9 ns) | | 15 | | ns | Register 0x017[1:0] = 00b, 01b, 11b; Register 0x018[4] = 0b |
| High Range (ABP 6.0 ns) | | 11 | | ns | Register 0x017[1:0] = 10b; Register 0x018[4] = 0b |

[1] The REFIN and $\overline{\text{REFIN}}$ self-bias points are offset slightly to avoid chatter on an open input condition.
[2] For reliable operation of the digital lock detect, the period of the PFD frequency must be greater than the unlock-after-lock time.

## CLOCK INPUTS

**Table 3.**

| Parameter | Min | Typ | Max | Unit | Test Conditions/Comments |
|---|---|---|---|---|---|
| CLOCK INPUTS (CLK, $\overline{CLK}$) | | | | | Differential input |
| Input Frequency | 0[1] | | 2.4 | GHz | High frequency distribution (VCO divider) |
| | 0[1] | | 2 | GHz | Distribution only (VCO divider bypassed); this is the frequency range supported by the channel divider, see the Channel Divider Maximum Frequency section |
| Input Sensitivity, Differential | | 150 | | mV p-p | Measured at 2.4 GHz; jitter performance is improved with slew rates > 1 V/ns |
| Input Level, Differential | | | 2 | V p-p | Larger voltage swings can turn on the protection diodes and can degrade jitter performance |
| Input Common-Mode Voltage, $V_{CM}$ | 1.3 | 1.57 | 1.8 | V | Self-biased; enables ac coupling |
| Input Common-Mode Range, $V_{CMR}$ | 1.3 | | 1.8 | V | With 200 mV p-p signal applied; dc-coupled |
| Input Sensitivity, Single-Ended | | 150 | | mV p-p | CLK ac-coupled; $\overline{CLK}$ ac-bypassed to RF ground |
| Input Resistance | 3.9 | 4.7 | 5.7 | kΩ | Self-biased |
| Input Capacitance | | 2 | | pF | |

[1] Below about 1 MHz, the input must be dc-coupled. Take care to match $V_{CM}$.

## CLOCK OUTPUTS

**Table 4.**

| Parameter | Min | Typ | Max | Unit | Test Conditions/Comments |
|---|---|---|---|---|---|
| LVDS CLOCK OUTPUTS | | | | | Termination = 100 Ω across differential pair |
| OUT0, OUT1, OUT2, OUT3, OUT4, OUT5, OUT6, OUT7, OUT8, OUT9, OUT10, OUT11 | | | | | Differential (OUT, $\overline{OUT}$) |
| Output Frequency | | | 800 | MHz | The AD9522 outputs toggle at higher frequencies, but the output amplitude may not meet the $V_{OD}$ specification |
| Output Differential Voltage, $V_{OD}$ | 247 | 360 | 454 | mV | $V_{OH} - V_{OL}$ for each leg of a differential pair for default amplitude setting with the driver not toggling; the peak-to-peak amplitude measured using a differential probe across the differential pair with the driver toggling is roughly 2× these values (see Figure 17) |
| Delta $V_{OD}$ | | | 25 | mV | Absolute difference between voltage swing of normal pin and inverted pin, output driver static |
| Output Offset Voltage, $V_{OS}$ | 1.125 | 1.25 | 1.375 | V | $(V_{OH} + V_{OL})/2$ across a differential pair |
| Delta $V_{OS}$ | | | 25 | mV | This is the absolute value of the difference between $V_{OS}$ when the normal output is high vs. when the complementary output is high |
| Short-Circuit Current, $I_{SA}$, $I_{SB}$ | | 14 | 24 | mA | Output shorted to GND |
| Tristate Leakage Current per Output | | <1 | | nA | Output in tristate with 100 Ω across differential pair |
| CMOS CLOCK OUTPUTS | | | | | |
| OUT0A, OUT0B, OUT1A, OUT1B, OUT2A, OUT2B, OUT3A, OUT3B, OUT4A, OUT4B, OUT5A, OUT5B, OUT6A, OUT6B, OUT7A, OUT7B, OUT8A, OUT8B, OUT9A, OUT9B, OUT10A, OUT10B, OUT11A, OUT11B | | | | | Single-ended; termination = 10 pF |
| Output Frequency | | | 250 | MHz | See Figure 18 |
| Output Voltage High, $V_{OH}$ | VS − 0.1 | | | V | At 1 mA load |
| Output Voltage Low, $V_{OL}$ | | | 0.1 | V | At 1 mA load |
| Output Voltage High, $V_{OH}$ | 2.7 | | | V | At 10 mA load |
| Output Voltage Low, $V_{OL}$ | | | 0.5 | V | At 10 mA load |

## TIMING CHARACTERISTICS

**Table 5.**

| Parameter | Min | Typ | Max | Unit | Test Conditions/Comments |
|---|---|---|---|---|---|
| LVDS OUTPUT RISE/FALL TIMES | | | | | Termination = 100 Ω across differential pair |
| Output Rise Time, $t_{RP}$ | | 150 | 350 | ps | 20% to 80%, measured differentially |
| Output Fall Time, $t_{FP}$ | | 150 | 350 | ps | 80% to 20%, measured differentially |
| PROPAGATION DELAY, $t_{LVDS}$, CLK-TO-LVDS OUTPUT | | | | | |
| For All Divide Values | 1866 | 2313 | 2812 | ps | High frequency clock distribution configuration |
| | 1808 | 2245 | 2740 | ps | Clock distribution configuration |
| Variation with Temperature | | 1 | | ps/°C | |
| OUTPUT SKEW, LVDS OUTPUTS[1] | | | | | Termination = 100 Ω across differential pair |
| LVDS Outputs That Share the Same Divider | | 7 | 60 | ps | |
| LVDS Outputs on Different Dividers | | 19 | 162 | ps | |
| All LVDS Outputs Across Multiple Parts | | | 432 | ps | |
| CMOS OUTPUT RISE/FALL TIMES | | | | | Termination = open |
| Output Rise Time, $t_{RC}$ | | 625 | 835 | ps | 20% to 80%; $C_{LOAD}$ = 10 pF |
| Output Fall Time, $t_{FC}$ | | 625 | 800 | ps | 80% to 20%; $C_{LOAD}$ = 10 pF |
| PROPAGATION DELAY, $t_{CMOS}$, CLK-TO-CMOS OUTPUT | | | | | Clock distribution configuration |
| For All Divide Values | 1913 | 2400 | 2950 | ps | |
| Variation with Temperature | | 2 | | ps/°C | |
| OUTPUT SKEW, CMOS OUTPUTS[1] | | | | | |
| CMOS Outputs That Share the Same Divider | | 10 | 55 | ps | |
| All CMOS Outputs on Different Dividers | | 27 | 230 | ps | |
| All CMOS Outputs Across Multiple Parts | | | 500 | ps | |
| OUTPUT SKEW, LVDS-TO-CMOS OUTPUT[1] | | | | | All settings identical; different logic type |
| Outputs That Share the Same Divider | −31 | +152 | +495 | ps | LVDS to CMOS on the same part |
| Outputs That Are on Different Dividers | −193 | +160 | +495 | ps | LVDS to CMOS on the same part |

[1] The output skew is the difference between any two similar delay paths while operating at the same voltage and temperature.

### *Timing Diagrams*

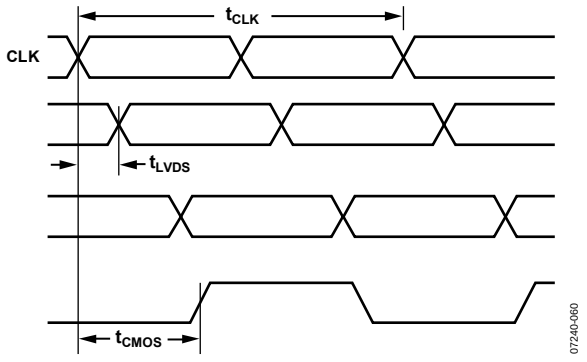


*Figure 2. CLK/$\overline{CLK}$ to Clock Output Timing, DIV = 1*

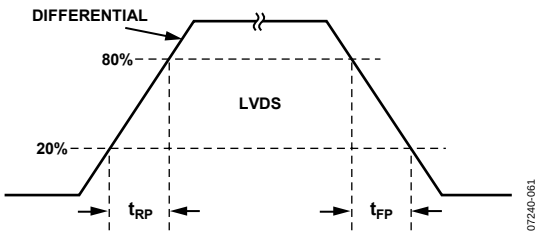


*Figure 3. LVDS Timing, Differential*

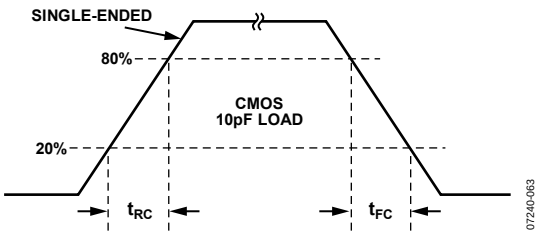


*Figure 4. CMOS Timing, Single-Ended, 10 pF Load*

## CLOCK OUTPUT ADDITIVE PHASE NOISE (DISTRIBUTION ONLY; VCO DIVIDER NOT USED)

**Table 6.**

| Parameter | Min | Typ | Max | Unit | Test Conditions/Comments |
|---|---|---|---|---|---|
| CLK-TO-LVDS ADDITIVE PHASE NOISE | | | | | Distribution section only; does not include PLL and VCO |
| CLK = 1.6 GHz, Output = 800 MHz | | | | | Input slew rate > 1 V/ns |
| Divider = 2 | | | | | |
| At 10 Hz Offset | | −100 | | dBc/Hz | |
| At 100 Hz Offset | | −110 | | dBc/Hz | |
| At 1 kHz Offset | | −117 | | dBc/Hz | |
| At 10 kHz Offset | | −126 | | dBc/Hz | |
| At 100 kHz Offset | | −134 | | dBc/Hz | |
| At 1 MHz Offset | | −137 | | dBc/Hz | |
| At 10 MHz Offset | | −147 | | dBc/Hz | |
| At 100 MHz Offset | | −148 | | dBc/Hz | |
| CLK = 1 GHz, Output = 200 MHz | | | | | Input slew rate > 1 V/ns |
| Divider = 5 | | | | | |
| At 10 Hz Offset | | −111 | | dBc/Hz | |
| At 100 Hz Offset | | −123 | | dBc/Hz | |
| At 1 kHz Offset | | −132 | | dBc/Hz | |
| At 10 kHz Offset | | −141 | | dBc/Hz | |
| At 100 kHz Offset | | −146 | | dBc/Hz | |
| At 1 MHz Offset | | −150 | | dBc/Hz | |
| >10 MHz Offset | | −156 | | dBc/Hz | |
| CLK-TO-CMOS ADDITIVE PHASE NOISE | | | | | Distribution section only; does not include PLL and VCO |
| CLK = 1 GHz, Output = 500 MHz | | | | | Input slew rate > 1 V/ns |
| Divider = 2 | | | | | |
| At 10 Hz Offset | | −102 | | dBc/Hz | |
| At 100 Hz Offset | | −114 | | dBc/Hz | |
| At 1 kHz Offset | | −122 | | dBc/Hz | |
| At 10 kHz Offset | | −129 | | dBc/Hz | |
| At 100 kHz Offset | | −135 | | dBc/Hz | |
| At 1 MHz Offset | | −140 | | dBc/Hz | |
| >10 MHz Offset | | −150 | | dBc/Hz | |
| CLK = 1 GHz, Output = 50 MHz | | | | | Input slew rate > 1 V/ns |
| Divider = 20 | | | | | |
| At 10 Hz Offset | | −125 | | dBc/Hz | |
| At 100 Hz Offset | | −136 | | dBc/Hz | |
| At 1 kHz Offset | | −144 | | dBc/Hz | |
| At 10 kHz Offset | | −152 | | dBc/Hz | |
| At 100 kHz Offset | | −157 | | dBc/Hz | |
| At 1 MHz Offset | | −160 | | dBc/Hz | |
| >10 MHz Offset | | −164 | | dBc/Hz | |

## CLOCK OUTPUT ABSOLUTE TIME JITTER (CLOCK GENERATION USING EXTERNAL VCXO)

**Table 7.**

| Parameter | Min | Typ | Max | Unit | Test Conditions/Comments |
|---|---|---|---|---|---|
| LVDS OUTPUT ABSOLUTE TIME JITTER | | | | | Application example based on a typical setup using an external 245.76 MHz VCXO (Toyocom TCO-2112); reference = 15.36 MHz; R DIV = 1 |
| LVDS = 245.76 MHz; PLL LBW = 125 Hz | | 87 | | fs rms | Integration bandwidth = 200 kHz to 5 MHz |
| | | 108 | | fs rms | Integration bandwidth = 200 kHz to 10 MHz |
| | | 146 | | fs rms | Integration bandwidth = 12 kHz to 20 MHz |
| LVDS = 122.88 MHz; PLL LBW = 125 Hz | | 120 | | fs rms | Integration bandwidth = 200 kHz to 5 MHz |
| | | 151 | | fs rms | Integration bandwidth = 200 kHz to 10 MHz |
| | | 207 | | fs rms | Integration bandwidth = 12 kHz to 20 MHz |
| LVDS = 61.44 MHz; PLL LBW = 125 Hz | | 157 | | fs rms | Integration bandwidth = 200 kHz to 5 MHz |
| | | 210 | | fs rms | Integration bandwidth = 200 kHz to 10 MHz |
| | | 295 | | fs rms | Integration bandwidth = 12 kHz to 20 MHz |

## CLOCK OUTPUT ADDITIVE TIME JITTER (VCO DIVIDER NOT USED)

**Table 8.**

| Parameter | Min | Typ | Max | Unit | Test Conditions/Comments |
|---|---|---|---|---|---|
| LVDS OUTPUT ADDITIVE TIME JITTER | | | | | Distribution section only; does not include PLL and VCO; measured at rising edge of clock signal |
| CLK = 622.08 MHz<br>Any LVDS Output = 622.08 MHz<br>Divide Ratio = 1 | | 69 | | fs rms | Integration bandwidth = 12 kHz to 20 MHz |
| CLK = 622.08 MHz<br>Any LVDS Output = 155.52 MHz<br>Divide Ratio = 4 | | 116 | | fs rms | Integration bandwidth = 12 kHz to 20 MHz |
| CLK = 100 MHz<br>Any LVDS Output = 100 MHz<br>Divide Ratio = 1 | | 263 | | fs rms | Calculated from SNR of ADC method<br>Broadband jitter |
| CLK = 500 MHz<br>Any LVDS Output = 100 MHz<br>Divide Ratio = 5 | | 242 | | fs rms | Calculated from SNR of ADC method<br>Broadband jitter |
| CMOS OUTPUT ADDITIVE TIME JITTER | | | | | Distribution section only; does not include PLL and VCO |
| CLK = 200 MHz<br>Any CMOS Output Pair = 100 MHz<br>Divide Ratio = 2 | | 289 | | fs rms | Calculated from SNR of ADC method<br>Broadband jitter |

## CLOCK OUTPUT ADDITIVE TIME JITTER (VCO DIVIDER USED)

**Table 9.**

| Parameter | Min | Typ | Max | Unit | Test Conditions/Comments |
|---|---|---|---|---|---|
| LVDS OUTPUT ADDITIVE TIME JITTER | | | | | Distribution section only; does not include PLL and VCO; uses rising edge of clock signal |
| CLK = 500 MHz; VCO DIV = 5; LVDS = 100 MHz; Bypass Channel Divider; Duty-Cycle Correction = On | | 248 | | fs rms | Calculated from SNR of ADC method (broadband jitter) |
| CMOS OUTPUT ADDITIVE TIME JITTER | | | | | Distribution section only; does not include PLL and VCO; uses rising edge of clock signal |
| CLK = 200 MHz; VCO DIV = 2; CMOS = 100 MHz; Bypass Channel Divider; Duty-Cycle Correction = Off | | 290 | | fs rms | Calculated from SNR of ADC method (broadband jitter) |
| CLK = 200 MHz; VCO DIV = 1; CMOS = 100 MHz; Bypass Channel Divider; Duty-Cycle Correction = Off | | 288 | | fs rms | Calculated from SNR of ADC method (broadband jitter) |

## SERIAL CONTROL PORT—SPI MODE

**Table 10.**

| Parameter | Min | Typ | Max | Unit | Test Conditions/Comments |
|---|---|---|---|---|---|
| $\overline{CS}$ (INPUT) | | | | | $\overline{CS}$ has an internal 30 kΩ pull-up resistor |
| Input Logic 1 Voltage | 2.0 | | | V | |
| Input Logic 0 Voltage | | | 0.8 | V | |
| Input Logic 1 Current | | | 3 | μA | |
| Input Logic 0 Current | | −110 | | μA | The minus sign indicates that current is flowing out of the AD9522, which is due to the internal pull-up resistor |
| Input Capacitance | | 2 | | pF | |
| SCLK (INPUT) IN SPI MODE | | | | | SCLK has an internal 30 kΩ pull-down resistor in SPI mode, but not in $I^2C$ mode |
| Input Logic 1 Voltage | 2.0 | | | V | |
| Input Logic 0 Voltage | | | 0.8 | V | |
| Input Logic 1 Current | | 110 | | μA | |
| Input Logic 0 Current | | | 1 | μA | |
| Input Capacitance | | 2 | | pF | |
| SDIO (WHEN AN INPUT IN BIDIRECTIONAL MODE) | | | | | |
| Input Logic 1 Voltage | 2.0 | | | V | |
| Input Logic 0 Voltage | | | 0.8 | V | |
| Input Logic 1 Current | | 1 | | μA | |
| Input Logic 0 Current | | 1 | | μA | |
| Input Capacitance | | 2 | | pF | |
| SDIO, SDO (OUTPUTS) | | | | | |
| Output Logic 1 Voltage | 2.7 | | | V | At 1 mA current; maximum recommended current: 5 mA |
| Output Logic 0 Voltage | | | 0.4 | V | At 1 mA current |
| TIMING | | | | | |
| Clock Rate (SCLK, $1/t_{SCLK}$) | | | 25 | MHz | |
| Pulse Width High, $t_{HIGH}$ | 16 | | | ns | |
| Pulse Width Low, $t_{LOW}$ | 16 | | | ns | |
| SDIO to SCLK Setup, $t_{DS}$ | 4 | | | ns | |
| SCLK to SDIO Hold, $t_{DH}$ | 0 | | | ns | |
| SCLK to Valid SDIO and SDO, $t_{DV}$ | | | 11 | ns | |
| $\overline{CS}$ to SCLK Setup and Hold, $t_S$, $t_C$ | 2 | | | ns | |
| $\overline{CS}$ Minimum Pulse Width High, $t_{PWH}$ | 3 | | | ns | |

## SERIAL CONTROL PORT—I²C MODE

**Table 11.**

| Parameter | Min | Typ | Max | Unit | Test Conditions/Comments |
|---|---|---|---|---|---|
| SDA, SCL (WHEN INPUTTING DATA) | | | | | |
| Input Logic 1 Voltage | 0.7 × VS | | | V | |
| Input Logic 0 Voltage | | | 0.3 × VS | V | |
| Input Current with an Input Voltage Between 0.1 × VS and 0.9 × VS | −10 | | +10 | µA | |
| Hysteresis of Schmitt Trigger Inputs | 0.015 × VS | | | V | |
| Pulse Width of Spikes That Must Be Suppressed by the Input Filter, $t_{SPIKE}$ | | | 50 | ns | |
| SDA (WHEN OUTPUTTING DATA) | | | | | |
| Output Logic 0 Voltage at 3 mA Sink Current | | | 0.4 | V | |
| Output Fall Time from $VIH_{MIN}$ to $VIL_{MAX}$ with a Bus Capacitance from 10 pF to 400 pF | 20 + 0.1 $C_b$ | | 250 | ns | $C_b$ = capacitance of one bus line in pF |
| TIMING | | | | | Note that all I²C timing values refer to $VIH_{MIN}$ (0.3 × VS) and $VIL_{MAX}$ levels (0.7 × VS) |
| Clock Rate (SCL, $f_{I2C}$) | | | 400 | kHz | |
| Bus Free Time Between a Stop and Start Condition, $t_{IDLE}$ | 1.3 | | | µs | |
| Setup Time for a Repeated Start Condition, $t_{SET; STR}$ | 0.6 | | | µs | |
| Hold Time (Repeated) Start Condition (After This Period, the First Clock Pulse Is Generated), $t_{HLD; STR}$ | 0.6 | | | µs | |
| Setup Time for Stop Condition, $t_{SET; STP}$ | 0.6 | | | µs | |
| Low Period of the SCL Clock, $t_{LOW}$ | 1.3 | | | µs | |
| High Period of the SCL Clock, $t_{HIGH}$ | 0.6 | | | µs | |
| SCL, SDA Rise Time, $t_{RISE}$ | 20 + 0.1 $C_b$ | | 300 | ns | $C_b$ = capacitance of one bus line in pF |
| SCL, SDA Fall Time, $t_{FALL}$ | 20 + 0.1 $C_b$ | | 300 | ns | $C_b$ = capacitance of one bus line in pF |
| Data Setup Time, $t_{SET; DAT}$ | 120 | | | ns | This is a minor deviation from the original I²C specification of 100 ns minimum |
| Data Hold Time, $t_{HLD; DAT}$ | 140 | | 880 | ns | This is a minor deviation from the original I²C specification of 0 ns minimum[1] |
| Capacitive Load for Each Bus Line, $C_b$ | | | 400 | pF | |

[1] According to the original I²C specification, an I²C master must also provide a minimum hold time of 300 ns for the SDA signal to bridge the undefined region of the SCL falling edge.

## $\overline{PD}$, $\overline{SYNC}$, AND $\overline{RESET}$ PINS

**Table 12.**

| Parameter | Min | Typ | Max | Unit | Test Conditions/Comments |
|---|---|---|---|---|---|
| INPUT CHARACTERISTICS | | | | | Each of these pins has an 30 kΩ internal pull-up resistor |
| Logic 1 Voltage | 2.0 | | | V | |
| Logic 0 Voltage | | | 0.8 | V | |
| Logic 1 Current | | | 1 | µA | |
| Logic 0 Current | | −110 | | µA | The minus sign indicates that current is flowing out of the AD9522, which is due to the internal pull-up resistor |
| Capacitance | | 2 | | pF | |
| RESET TIMING | | | | | |
| Pulse Width Low | 50 | | | ns | |
| $\overline{RESET}$ Inactive to Start of Register Programming | 100 | | | ns | |
| $\overline{SYNC}$ TIMING | | | | | |
| Pulse Width Low | 1.3 | | | ns | High speed clock is CLK input signal |

## SERIAL PORT SETUP PINS: SP1, SP0

**Table 13.**

| Parameter | Min | Typ | Max | Unit | Test Conditions/Comments |
|---|---|---|---|---|---|
| SP1, SP0 | | | | | These pins do not have internal pull-up/pull-down resistors |
| Logic Level 0 | | | 0.25 × VS | V | VS is the voltage on the VS pin |
| Logic Level ½ | 0.4 × VS | | 0.65 × VS | V | User can float these pins to obtain Logic Level ½; if floating this pin, connect a capacitor to ground |
| Logic Level 1 | 0.8 × VS | | | V | |

## LD, STATUS, AND REFMON PINS

**Table 14.**

| Parameter | Min | Typ | Max | Unit | Test Conditions/Comments |
|---|---|---|---|---|---|
| OUTPUT CHARACTERISTICS | | | | | When selected as a digital output (CMOS); there are other modes in which these pins are not CMOS digital outputs; see Table 47, Register 0x017, Register 0x01A, and Register 0x01B |
| Output Voltage High, $V_{OH}$ | 2.7 | | | V | At 1 mA current; maximum recommended current: 5 mA |
| Output Voltage Low, $V_{OL}$ | | | 0.4 | V | At 1 mA current |
| MAXIMUM TOGGLE RATE | | 100 | | MHz | Applies when mux is set to any divider or counter output, or PFD up/down pulse; also applies in analog lock detect mode; usually debug mode only; note that spurs can couple to output when any of these pins are toggling |
| ANALOG LOCK DETECT | | | | | |
| Capacitance | | 3 | | pF | On-chip capacitance; used to calculate RC time constant for analog lock detect read back; use a pull-up resistor |
| REF1, REF2, AND CLK FREQUENCY STATUS MONITOR | | | | | |
| Normal Range | 1.02 | | | MHz | Frequency above which the monitor indicates the presence of the reference |
| Extended Range | 8 | | | kHz | Frequency above which the monitor indicates the presence of the reference |
| LD PIN COMPARATOR | | | | | |
| Trip Point | | 1.6 | | V | |
| Hysteresis | | 260 | | mV | |

## POWER DISSIPATION

**Table 15.**

| Parameter | Min | Typ | Max | Unit | Test Conditions/Comments |
|---|---|---|---|---|---|
| POWER DISSIPATION, CHIP | | | | | Does not include power dissipated in external resistors; all LVDS outputs terminated with 100 Ω across differential pair; all CMOS outputs have 10 pF capacitive loading |
| Power-On Default | | 0.88 | 1.0 | W | No clock; no programming; default register values |
| Distribution Only Mode; VCO Divider On; One LVDS Output Enabled | | 0.36 | 0.43 | W | $f_{CLK}$ = 2.4 GHz; $f_{OUT}$ = 200 MHz; VCO divider = 2; one LVDS output and output divider enabled; zero delay off |
| Distribution Only Mode; VCO Divider Off; One LVDS Output Enabled | | 0.33 | 0.4 | W | $f_{CLK}$ = 2.4 GHz; $f_{OUT}$ = 200 MHz; VCO divider bypassed; one LVDS output and output divider enabled; zero delay off |
| Maximum Power, Full Operation | | 1.1 | 1.3 | W | PLL on; VCO divider = 3; all channel dividers on; 12 LVDS outputs at 125 MHz; zero delay on |
| $\overline{PD}$ Power-Down | | 35 | 50 | mW | $\overline{PD}$ pin pulled low; does not include power dissipated in termination resistors |
| $\overline{PD}$ Power-Down, Maximum Sleep | | 27 | 43 | mW | $\overline{PD}$ pin pulled low; PLL power-down, Register 0x010[1:0] = 01b; power-down SYNC, Register 0x230[2] = 1b; power-down distribution reference, Register 0x230[1] = 1b |
| VCP Supply | | 2.3 | 8 | mW | PLL operating; typical closed-loop configuration |
| POWER DELTAS, INDIVIDUAL FUNCTIONS | | | | | Power delta when a function is enabled/disabled |
| VCO Divider On/Off | | 33 | 43 | mW | VCO divider not used |
| REFIN (Differential) Off | | 25 | 31 | mW | Delta between reference input off and differential reference input mode |
| REF1, REF2 (Single-Ended) On/Off | | 16 | 22 | mW | Delta between reference inputs off and one single-ended reference enabled; double this number if both REF1 and REF2 are powered up |
| PLL Dividers and Phase Detector On/Off | | 54 | 67 | mW | PLL off to PLL on, normal operation; no reference enabled |
| LVDS Channel | | 118 | 146 | mW | No LVDS output on to one LVDS output on; channel divider set to 1 |
| LVDS Driver | | 11 | 15 | mW | Second LVDS output turned on, same channel |
| CMOS Channel | | 120 | 154 | mW | No CMOS output on to one CMOS output on; channel divider set to 1; $f_{OUT}$ = 62.5 MHz and 10 pF of capacitive loading |
| CMOS Driver On/Off | | 16 | 30 | mW | Additional CMOS outputs within the same channel turned on |
| Channel Divider Enabled | | 33 | 40 | mW | Delta between divider bypassed (divide-by-1) and divide-by-2 to divide-by-32 |
| Zero Delay Block On/Off | | 30 | 35 | mW | |

# ABSOLUTE MAXIMUM RATINGS

**Table 16.**

| Parameter or Pin | With Respect to | Rating |
|---|---|---|
| VS | GND | −0.3 V to +3.6 V |
| VCP, CP | GND | −0.3 V to +5.8 V |
| REFIN, $\overline{\text{REFIN}}$ | GND | −0.3 V to VS + 0.3 V |
| RSET | GND | −0.3 V to VS + 0.3 V |
| CPRSET | GND | −0.3 V to VS + 0.3 V |
| CLK, $\overline{\text{CLK}}$ | GND | −0.3 V to VS + 0.3 V |
| CLK | $\overline{\text{CLK}}$ | −1.2 V to +1.2 V |
| SCLK/SCL, SDIO/SDA, SDO, $\overline{\text{CS}}$ | GND | −0.3 V to VS + 0.3 V |
| OUT0, $\overline{\text{OUT0}}$, OUT1, $\overline{\text{OUT1}}$, OUT2, $\overline{\text{OUT2}}$, OUT3, $\overline{\text{OUT3}}$, OUT4, $\overline{\text{OUT4}}$, OUT5, $\overline{\text{OUT5}}$, OUT6, $\overline{\text{OUT6}}$, OUT7, $\overline{\text{OUT7}}$, OUT8, $\overline{\text{OUT8}}$, OUT9, $\overline{\text{OUT9}}$, OUT10, $\overline{\text{OUT10}}$, OUT11, $\overline{\text{OUT11}}$ | GND | −0.3 V to VS + 0.3 V |
| $\overline{\text{SYNC}}$, $\overline{\text{RESET}}$, $\overline{\text{PD}}$ | GND | −0.3 V to VS + 0.3 V |
| REFMON, STATUS, LD | GND | −0.3 V to VS + 0.3 V |
| SP0, SP1, EEPROM | GND | −0.3 V to VS + 0.3 V |
| Junction Temperature[1] | | 125°C |
| Storage Temperature Range | | −65°C to +150°C |
| Lead Temperature (10 sec) | | 300°C |

[1] See the Specifications section for operating temperature range ($T_A$).

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 impedance measurements were taken on a JEDEC JESD51-5 2S2P test board in still air in accordance with JEDEC JESD51-2. See the Thermal Performance section for more details.

**Table 17.**

| Package Type | $\theta_{JA}$ | Unit |
|---|---|---|
| 64-Lead LFCSP (CP-64-4) | 22 | °C/W |

## 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 CONFIGURATION AND FUNCTION DESCRIPTIONS

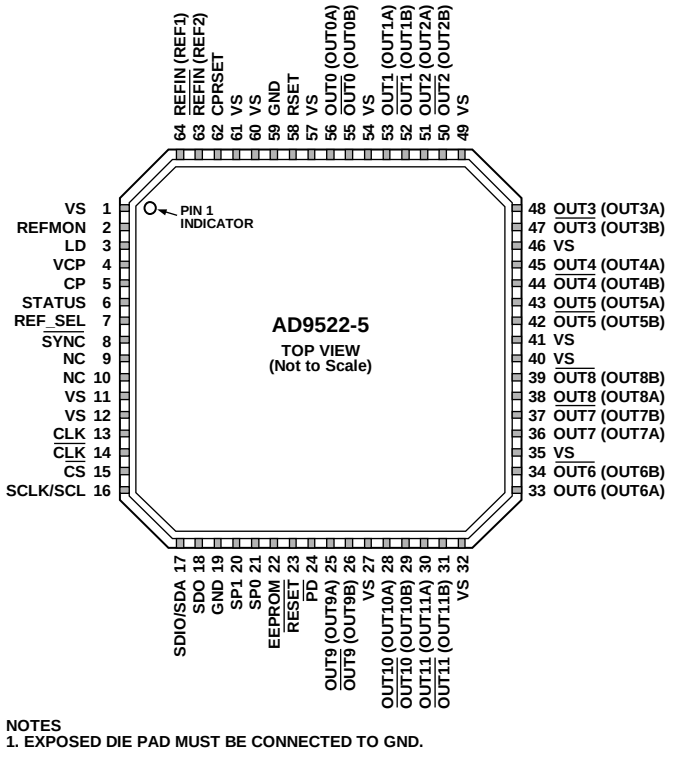


*Figure 5. Pin Configuration*

**Table 18. Pin Function Descriptions**

| Pin No. | Input/ Output | Pin Type | Mnemonic | Description |
|---|---|---|---|---|
| 1, 11, 12, 27, 32, 35, 40, 41, 46, 49, 54, 57, 60, 61 | I | Power | VS | 3.3 V Power Pins. |
| 2 | O | 3.3 V CMOS | REFMON | Reference Monitor (Output). This pin has multiple selectable outputs. |
| 3 | O | 3.3 V CMOS | LD | Lock Detect (Output). This pin has multiple selectable outputs. |
| 4 | I | Power | VCP | Power Supply for Charge Pump (CP); VS ≤ VCP ≤ 5.25 V. VCP must still be connected to 3.3 V if the PLL is not used. |
| 5 | O | Loop filter | CP | Charge Pump (Output). This pin connects to an external loop filter. This pin can be left unconnected if the PLL is not used. |
| 6 | O | 3.3 V CMOS | STATUS | Programmable Status Output. |
| 7 | I | 3.3 V CMOS | REF_SEL | Reference Select. It selects REF1 (low) or REF2 (high). This pin has an internal 30 kΩ pull-down resistor. |
| 8 | I | 3.3 V CMOS | $\overline{\text{SYNC}}$ | Manual Synchronizations and Manual Holdover. This pin initiates a manual synchronization and is used for manual holdover. Active low. This pin has an internal 30 kΩ pull-up resistor. |
| 9, 10 | | | NC | No Connect. These pins can be left floating. |
| 13 | I | Differential clock input | CLK | Along with $\overline{\text{CLK}}$, this pin is the differential input for the clock distribution section. |
| 14 | I | Differential clock input | $\overline{\text{CLK}}$ | Along with CLK, this pin is the differential input for the clock distribution section. If a single-ended input is connected to the CLK pin, connect a 0.1 µF bypass capacitor from this pin to ground. |

| Pin No. | Input/ Output | Pin Type | Mnemonic | Description |
|---|---|---|---|---|
| 15 | I | 3.3 V CMOS | $\overline{CS}$ | Serial Control Port Chip Select; Active Low. This pin has an internal 30 kΩ pull-up resistor. |
| 16 | I | 3.3 V CMOS | SCLK/SCL | Serial Control Port Clock Signal. This pin has an internal 30 kΩ pull-down resistor in SPI mode but is high impedance in $I^2C$ mode. |
| 17 | I/O | 3.3 V CMOS | SDIO/SDA | Serial Control Port Bidirectional Serial Data In/Out. |
| 18 | O | 3.3 V CMOS | SDO | Serial Control Port Unidirectional Serial Data Out. |
| 19, 59 | I | GND | GND | Ground Pins. |
| 20 | I | Three-level logic | SP1 | Select SPI or $I^2C$ as the serial interface port and select the $I^2C$ slave address in $I^2C$ mode. Three-level logic. This pin is internally biased for the open logic level. |
| 21 | I | Three-level logic | SP0 | Select SPI or $I^2C$ as the serial interface port and select the $I^2C$ slave address in $I^2C$ mode. Three-level logic. This pin is internally biased for the open logic level. |
| 22 | I | 3.3 V CMOS | EEPROM | Setting this pin high selects the register values stored in the internal EEPROM to load at reset and/or power-up. Setting this pin low causes the AD9522 to load the hard-coded default register values at power-up/reset. This pin has an internal 30 kΩ pull-down resistor. Note that to guarantee the proper loading of the EEPROM during startup, a high-low-high pulse on the $\overline{RESET}$ pin occurs after the power supply stabilizes. |
| 23 | I | 3.3 V CMOS | $\overline{RESET}$ | Chip Reset, Active Low. This pin has an internal 30 kΩ pull-up resistor. |
| 24 | I | 3.3 V CMOS | $\overline{PD}$ | Chip Power-Down, Active Low. This pin has an internal 30 kΩ pull-up resistor. |
| 25 | O | LVDS or CMOS | OUT9 (OUT9A) | Clock Output. This pin can be configured as one side of a differential LVDS output or as a single-ended CMOS output. |
| 26 | O | LVDS or CMOS | $\overline{OUT9}$ (OUT9B) | Clock Output. This pin can be configured as one side of a differential LVDS output or as a single-ended CMOS output. |
| 28 | O | LVDS or CMOS | OUT10 (OUT10A) | Clock Output. This pin can be configured as one side of a differential LVDS output or as a single-ended CMOS output. |
| 29 | O | LVDS or CMOS | $\overline{OUT10}$ (OUT10B) | Clock Output. This pin can be configured as one side of a differential LVDS output or as a single-ended CMOS output. |
| 30 | O | LVDS or CMOS | OUT11 (OUT11A) | Clock Output. This pin can be configured as one side of a differential LVDS output or as a single-ended CMOS output. |
| 31 | O | LVDS or CMOS | $\overline{OUT11}$ (OUT11B) | Clock Output. This pin can be configured as one side of a differential LVDS output or as a single-ended CMOS output. |
| 33 | O | LVDS or CMOS | OUT6 (OUT6A) | Clock Output. This pin can be configured as one side of a differential LVDS output or as a single-ended CMOS output. |
| 34 | O | LVDS or CMOS | $\overline{OUT6}$ (OUT6B) | Clock Output. This pin can be configured as one side of a differential LVDS output or as a single-ended CMOS output. |
| 36 | O | LVDS or CMOS | OUT7 (OUT7A) | Clock Output. This pin can be configured as one side of a differential LVDS output or as a single-ended CMOS output. |
| 37 | O | LVDS or CMOS | $\overline{OUT7}$ (OUT7B) | Clock Output. This pin can be configured as one side of a differential LVDS output or as a single-ended CMOS output. |
| 38 | O | LVDS or CMOS | OUT8 (OUT8A) | Clock Output. This pin can be configured as one side of a differential LVDS output or as a single-ended CMOS output. |
| 39 | O | LVDS or CMOS | $\overline{OUT8}$ (OUT8B) | Clock Output. This pin can be configured as one side of a differential LVDS output or as a single-ended CMOS output. |
| 42 | O | LVDS or CMOS | $\overline{OUT5}$ (OUT5B) | Clock Output. This pin can be configured as one side of a differential LVDS output or as a single-ended CMOS output. |
| 43 | O | LVDS or CMOS | OUT5 (OUT5A) | Clock Output. This pin can be configured as one side of a differential LVDS output or as a single-ended CMOS output. |
| 44 | O | LVDS or CMOS | $\overline{OUT4}$ (OUT4B) | Clock Output. This pin can be configured as one side of a differential LVDS output or as a single-ended CMOS output. |
| 45 | O | LVDS or CMOS | OUT4 (OUT4A) | Clock Output. This pin can be configured as one side of a differential LVDS output or as a single-ended CMOS output. |

| Pin No. | Input/ Output | Pin Type | Mnemonic | Description |
|---|---|---|---|---|
| 47 | O | LVDS or CMOS | $\overline{\text{OUT3}}$ (OUT3B) | Clock Output. This pin can be configured as one side of a differential LVDS output or as a single-ended CMOS output. |
| 48 | O | LVDS or CMOS | OUT3 (OUT3A) | Clock Output. This pin can be configured as one side of a differential LVDS output or as a single-ended CMOS output. |
| 50 | O | LVDS or CMOS | $\overline{\text{OUT2}}$ (OUT2B) | Clock Output. This pin can be configured as one side of a differential LVDS output or as a single-ended CMOS output. |
| 51 | O | LVDS or CMOS | OUT2 (OUT2A) | Clock Output. This pin can be configured as one side of a differential LVDS output or as a single-ended CMOS output. |
| 52 | O | LVDS or CMOS | $\overline{\text{OUT1}}$ (OUT1B) | Clock Output. This pin can be configured as one side of a differential LVDS output or as a single-ended CMOS output. |
| 53 | O | LVDS or CMOS | OUT1 (OUT1A) | Clock Output. This pin can be configured as one side of a differential LVDS output or as a single-ended CMOS output. |
| 55 | O | LVDS or CMOS | $\overline{\text{OUT0}}$ (OUT0B) | Clock Output. This pin can be configured as one side of a differential LVDS output or as a single-ended CMOS output. |
| 56 | O | LVDS or CMOS | OUT0 (OUT0A) | Clock Output. This pin can be configured as one side of a differential LVDS output or as a single-ended CMOS output. |
| 58 | O | Current set resistor | RSET | Clock Distribution Current Set Resistor. Connect a 4.12 kΩ resistor from this pin to GND. |
| 62 | O | Current set resistor | CPRSET | Charge Pump Current Set Resistor. Connect a 5.1 kΩ resistor from this pin to GND. This resistor can be omitted if the PLL is not used. |
| 63 | I | Reference input | $\overline{\text{REFIN}}$ (REF2) | Along with REFIN, this is the differential input for the PLL reference. Alternatively, this pin is a single-ended input for REF2. |
| 64 | I | Reference input | REFIN (REF1) | Along with $\overline{\text{REFIN}}$, this is the differential input for the PLL reference. Alternatively, this pin is a single-ended input for REF1. |
| EPAD | | GND | GND | The exposed die pad must be connected to GND. |

# TYPICAL PERFORMANCE CHARACTERISTICS

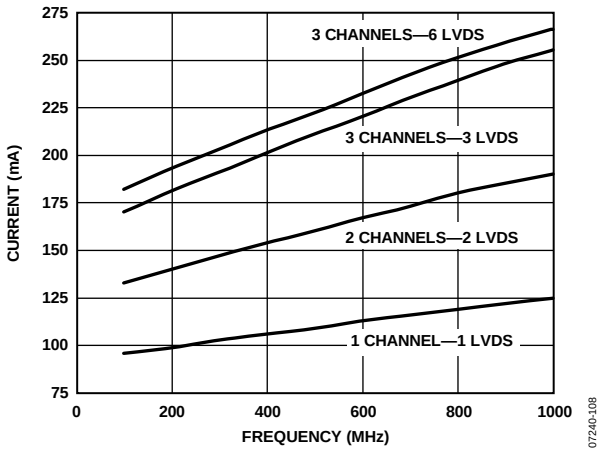


*Figure 6. Total Current vs. Frequency, CLK-to-Output (PLL Off), Channel and VCO Divider Bypassed, LVDS Outputs Terminated 100 Ω Across Differential Pair*

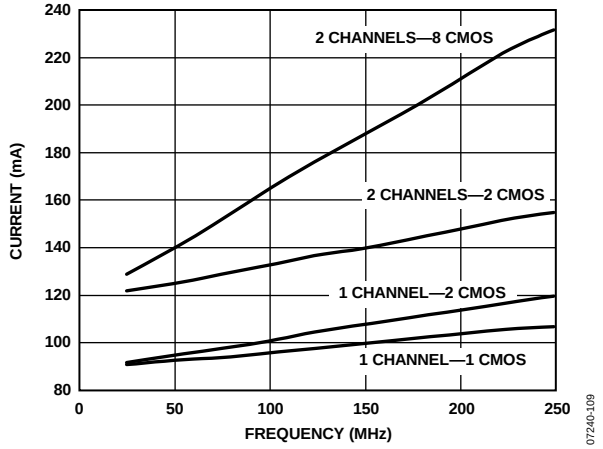


*Figure 7. Total Current vs. Frequency, CLK-to-Output (PLL Off), Channel and VCO Divider Bypassed, CMOS Outputs with 10 pF Load*

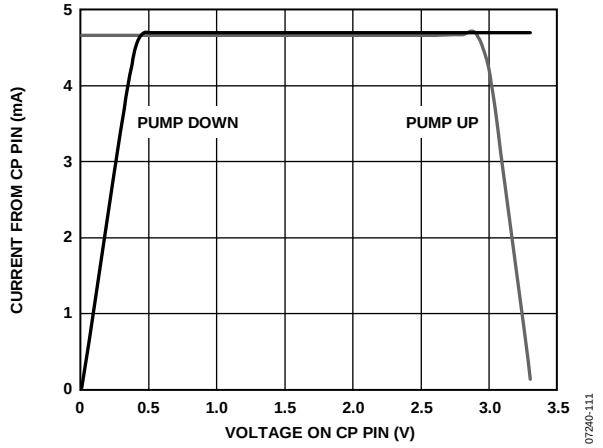


*Figure 8. Charge Pump Characteristics at VCP = 3.3 V*

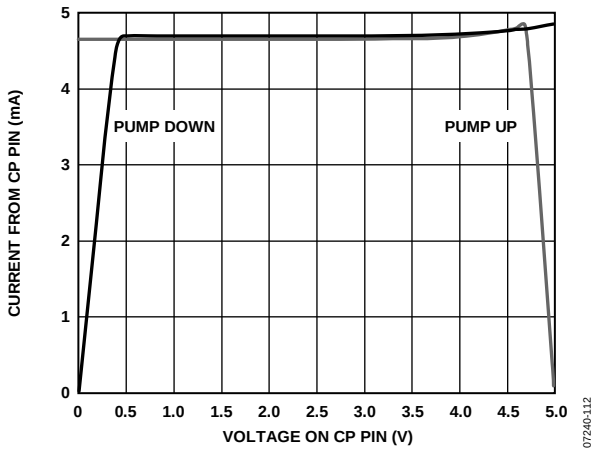


*Figure 9. Charge Pump Characteristics at VCP = 5.0 V*

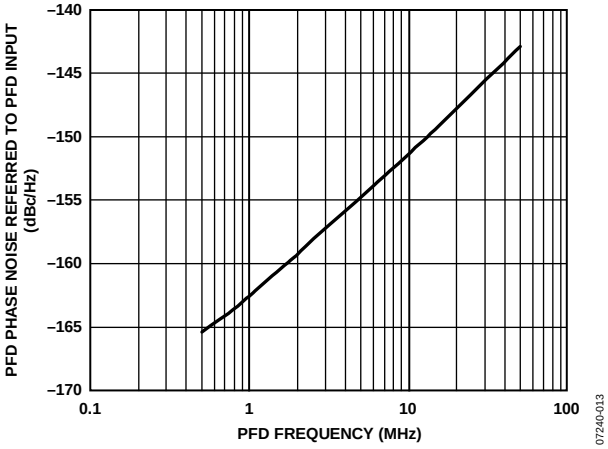


*Figure 10. PFD Phase Noise Referred to PFD Input vs. PFD Frequency*

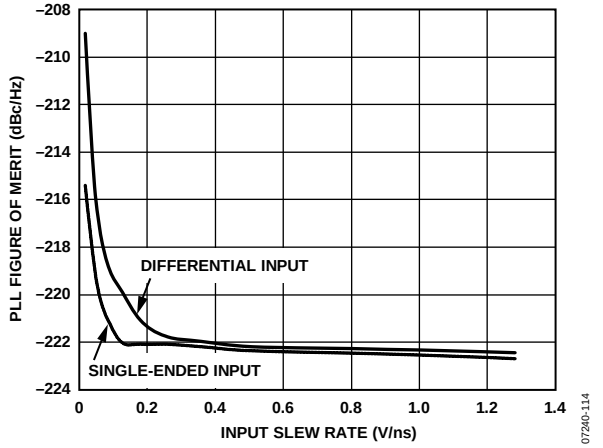


*Figure 11. PLL Figure of Merit (FOM) vs. Slew Rate at REFIN/$\overline{\text{REFIN}}$*

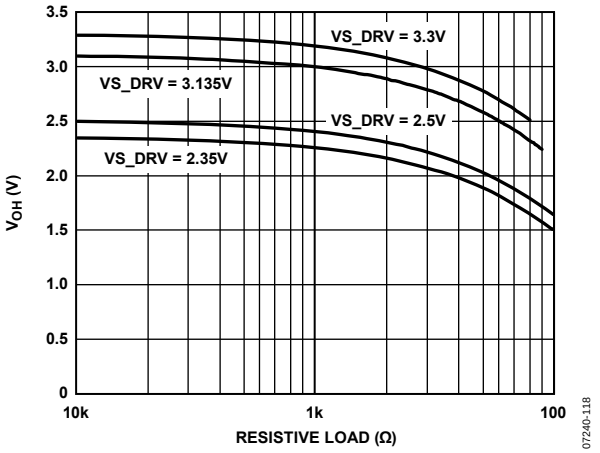


Figure 12. CMOS Output $V_{OH}$ (Static) vs. $R_{LOAD}$ (to Ground)

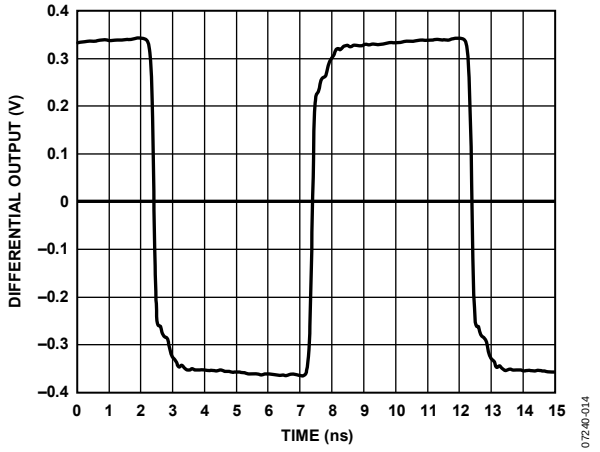


Figure 13. LVDS Output (Differential) at 100 MHz, Output Terminated 100 Ω Across Differential Pair

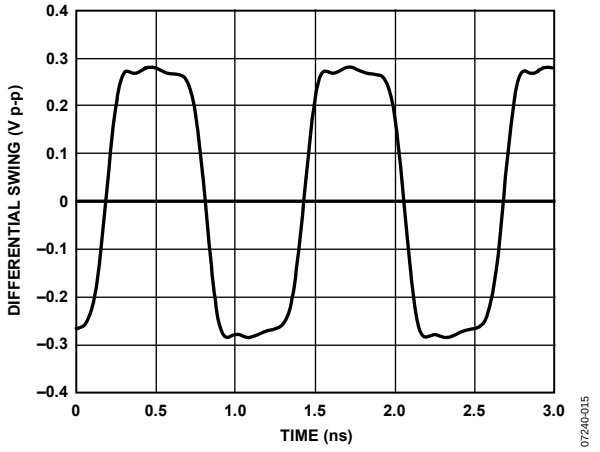


Figure 14. LVDS Differential Voltage Swing at 800 MHz, Output Terminated 100 Ω Across Differential Pair

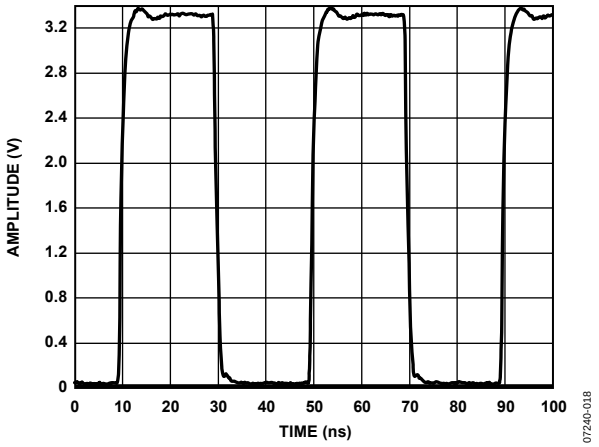


Figure 15. CMOS Output with 10 pF Load at 25 MHz

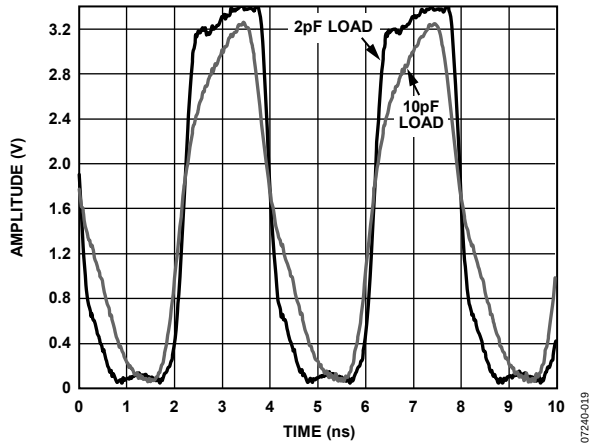


Figure 16. CMOS Output with 2 pF and 10 pF Load at 250 MHz

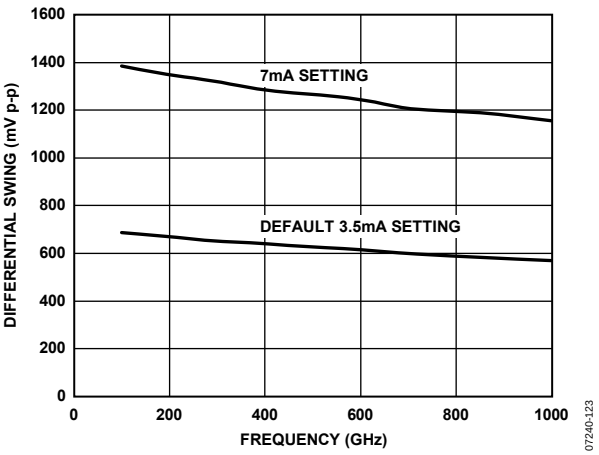


Figure 17. LVDS Differential Voltage Swing vs. Frequency, Output Terminated 100 Ω Across Differential Pair

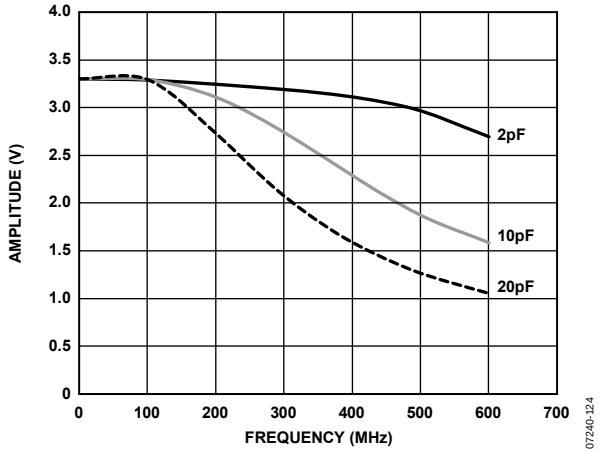


Figure 18. CMOS Output Swing vs. Frequency and Capacitive Load

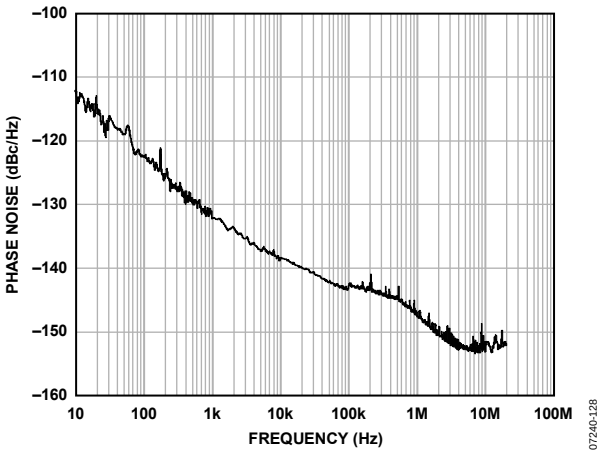


Figure 19. Additive (Residual) Phase Noise, CLK-to-LVDS at 245.76 MHz, Divide-by-1

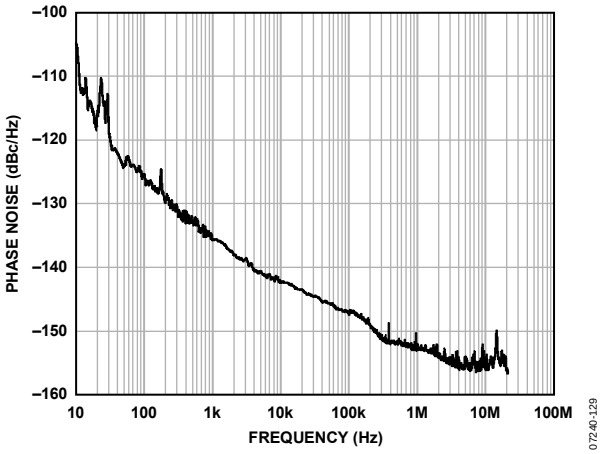


Figure 20. Additive (Residual) Phase Noise, CLK-to-LVDS at 200 MHz, Divide-by-5

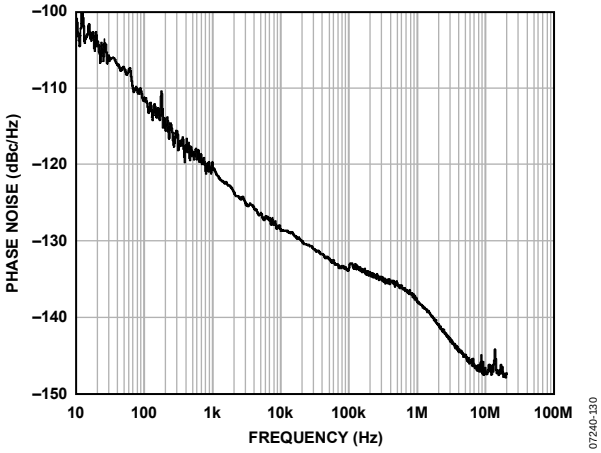


Figure 21. Additive (Residual) Phase Noise, CLK-to-LVDS at 800 MHz, Divide-by-1

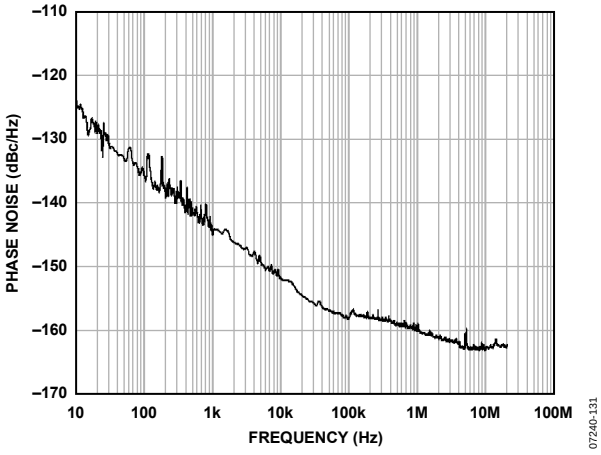


Figure 22. Additive (Residual) Phase Noise, CLK-to-CMOS at 50 MHz, Divide-by-20

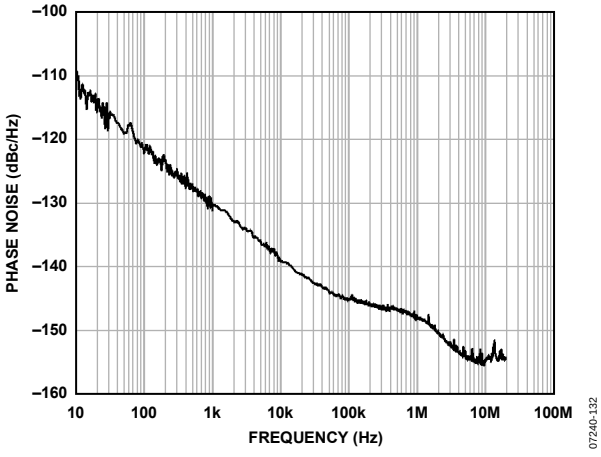


Figure 23. Additive (Residual) Phase Noise, CLK-to-CMOS at 250 MHz, Divide-by-4

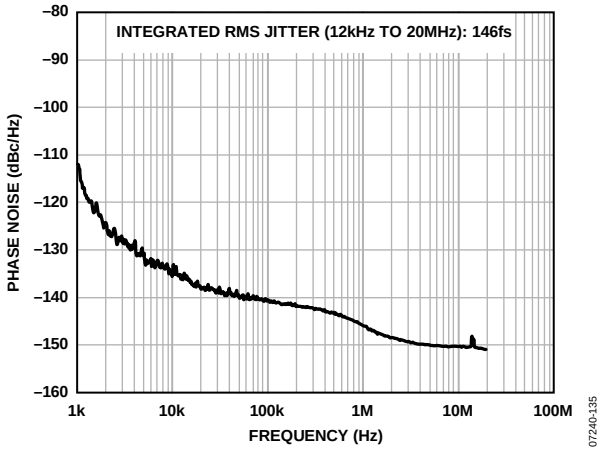


*Figure 24. Phase Noise (Absolute), External VCXO (Toyocom TCO-2112) at 245.76 MHz; PFD = 15.36 MHz; LBW = 250 Hz; LVDS Output = 245.76 MHz*

# TERMINOLOGY

**Phase Jitter and Phase Noise**
An ideal sine wave can be thought of as having a continuous and even progression of phase with time from 0° to 360° for each cycle. Actual signals, however, display a certain amount of variation from ideal phase progression over time. This phenomenon is called phase jitter. Although many causes can contribute to phase jitter, one major cause is random noise, which is characterized statistically as Gaussian (normal) in distribution.

This phase jitter leads to a spreading out of the energy of the sine wave in the frequency domain, producing a continuous power spectrum. This power spectrum is usually reported as a series of values whose units are dBc/Hz at a given offset in frequency from the sine wave (carrier). The value is a ratio (expressed in decibels) of the power contained within a 1 Hz bandwidth with respect to the power at the carrier frequency. For each measurement, the offset from the carrier frequency is also given.

It is meaningful to integrate the total power contained within some interval of offset frequencies (for example, 10 kHz to 10 MHz). This is called the integrated phase noise over that frequency offset interval and can be readily related to the time jitter due to the phase noise within that offset frequency interval.

Phase noise has a detrimental effect on the performance of ADCs, DACs, and RF mixers. It lowers the achievable dynamic range of the converters and mixers, although they are affected in somewhat different ways.

**Time Jitter**
Phase noise is a frequency domain phenomenon. In the time domain, the same effect is exhibited as time jitter. When observing a sine wave, the time of successive zero crossings varies. In a square wave, the time jitter is a displacement of the edges from their ideal (regular) times of occurrence. In both cases, the variations in timing from the ideal are the time jitter. Because these variations are random in nature, the time jitter is specified in seconds root mean square (rms) or 1 sigma of the Gaussian distribution.

Time jitter that occurs on a sampling clock for a DAC or an ADC decreases the signal-to-noise ratio (SNR) and dynamic range of the converter. A sampling clock with the lowest possible jitter provides the highest performance from a given converter.

**Additive Phase Noise**
Additive phase noise is the amount of phase noise that is attributable to the device or subsystem being measured. The phase noise of any external oscillators or clock sources is subtracted. This makes it possible to predict the degree to which the device impacts the total system phase noise when used in conjunction with the various oscillators and clock sources, each of which contributes its own phase noise to the total. In many cases, the phase noise of one element dominates the system phase noise. When there are multiple contributors to phase noise, the total is the square root of the sum of squares of the individual contributors.

**Additive Time Jitter**
Additive time jitter is the amount of time jitter that is attributable to the device or subsystem being measured. The time jitter of any external oscillators or clock sources is subtracted. This makes it possible to predict the degree to which the device impacts the total system time jitter when used in conjunction with the various oscillators and clock sources, each of which contributes its own time jitter to the total. In many cases, the time jitter of the external oscillators and clock sources dominates the system time jitter.

# DETAILED BLOCK DIAGRAM

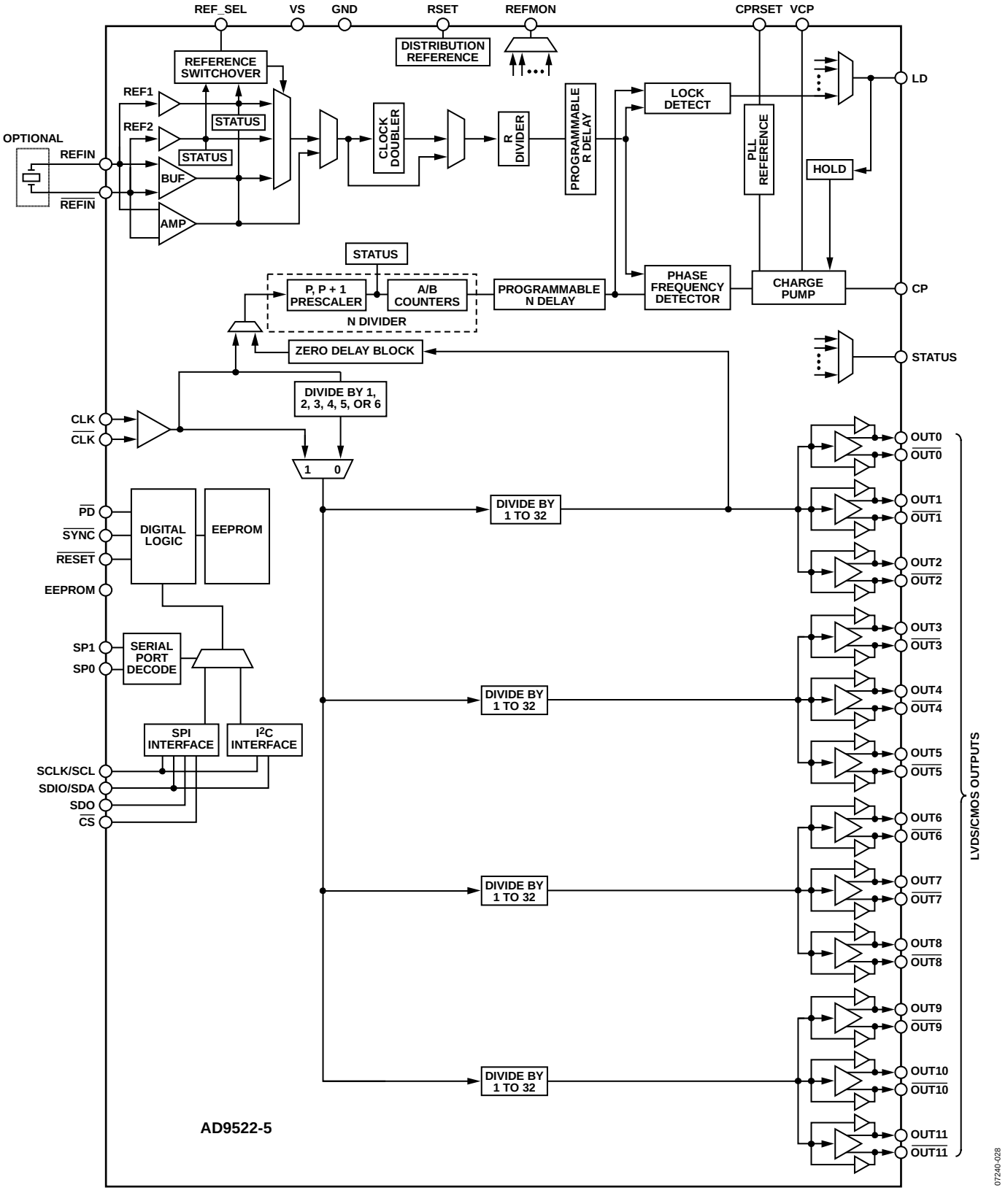


*Figure 25.*

# THEORY OF OPERATION

## OPERATIONAL CONFIGURATIONS

The AD9522 can be configured in several ways. These configurations must be set up by loading the control registers (see Table 43 to Table 54). Each section or function must be individually programmed by setting the appropriate bits in the corresponding control register or registers. After the desired configuration is programmed, the user can store these values in the on-board EEPROM to allow the part to power up in the desired configuration without user intervention.

### *Mode 1: Clock Distribution or External VCO < 1600 MHz*

When the external clock source to be distributed or the external VCO/VCXO is <1600 MHz, a configuration that bypasses the VCO divider can be used. This is the only difference from Mode 2. Bypassing the VCO divider limits the frequency of the clock source to <1600 MHz (due to the maximum input frequency allowed at the channel dividers).

For clock distribution applications where the external clock is <1600 MHz, use the register settings shown in Table 19.

**Table 19. Settings for Clock Distribution < 1600 MHz**

| Register | Description |
|---|---|
| 0x010[1:0] = 01b | PLL asynchronous power-down (PLL off) |
| 0x1E1[0] = 1b | Bypass the VCO divider as the source for the distribution section |

When using the PLL with an external VCO < 1600 MHz, the PLL must be turned on.

**Table 20. Settings for Using the PLL with External VCO < 1600 MHz**

| Register | Description |
|---|---|
| 0x1E1[0] = 1b | Bypass the VCO divider as the source for the distribution section |
| 0x010[1:0] = 00b | PLL normal operation (PLL on) along with other appropriate PLL settings in 0x010 to 0x01F |

An external VCO/VCXO requires an external loop filter that must be connected between CP and the tuning pin of the VCO/VCXO. This loop filter determines the loop bandwidth and stability of the PLL. Make sure to select the proper PFD polarity for the VCO/VCXO being used.

**Table 21. Setting the PFD Polarity**

| Register | Description |
|---|---|
| 0x010[7] = 0b | PFD polarity positive (higher control voltage produces higher frequency) |
| 0x010[7] = 1b | PFD polarity negative (higher control voltage produces lower frequency) |

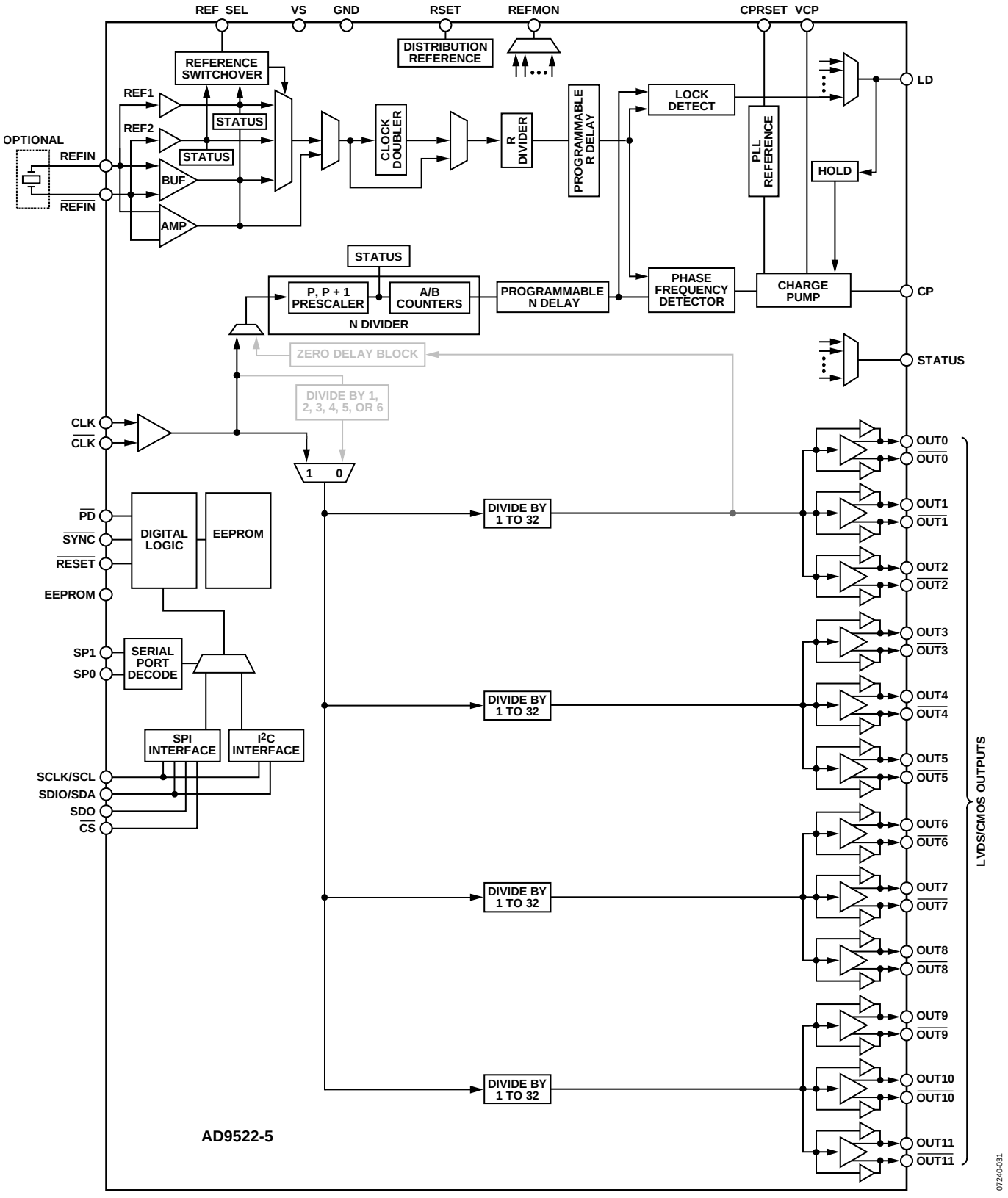


*Figure 26. Clock Distribution or External VCO < 1600 MHz (Mode 1)*

### *Mode 2: High Frequency Clock Distribution—CLK or External VCO > 1600 MHz*

The AD9522 power-up default configuration has the PLL powered off and the routing of the input set so that the CLK/$\overline{\text{CLK}}$ input is connected to the distribution section through the VCO divider (divide-by-1/divide-by-2/divide-by-3/divide-by-4/divide-by-5/divide-by-6). This is a distribution-only mode that allows for an external input up to 2400 MHz (see Table 3). The maximum frequency that can be applied to the channel dividers is 1600 MHz; therefore, higher input frequencies must be divided down before reaching the channel dividers.

When the PLL is enabled, this routing also allows the use of the PLL with an external VCO or VCXO with a frequency <2400 MHz. In this configuration, the external VCO/VCXO feeds directly into the prescaler.

The register settings shown in Table 22 are the default values of these registers at power-up or after a reset operation.

**Table 22. Default Register Settings for Clock Distribution Mode**

| Register | Description |
|---|---|
| 0x010[1:0] = 01b | PLL asynchronous power-down (PLL off) |
| 0x1E0[2:0] = 000b | Set VCO divider = 2 |
| 0x1E1[0] = 0b | CLK selected as the source |
| 0x1E1[1] = 0b | Use the VCO divider |

When using the PLL with an external VCO, the PLL must be turned on.

**Table 23. Settings When Using an External VCO**

| Register | Description |
|---|---|
| 0x010[1:0] = 00b | PLL normal operation (PLL on) |
| 0x010 to 0x01F | PLL settings; select and enable a reference input; set R, N (P, A, B), PFD polarity, and $I_{CP}$ according to the intended loop configuration |

An external VCO requires an external loop filter that must be connected between CP and the tuning pin of the VCO. This loop filter determines the loop bandwidth and stability of the PLL. Make sure to select the proper PFD polarity for the VCO being used.

**Table 24. Setting the PFD Polarity**

| Register | Description |
|---|---|
| 0x010[7] = 0b | PFD polarity positive (higher control voltage produces higher frequency) |
| 0x010[7] = 1b | PFD polarity negative (higher control voltage produces lower frequency) |

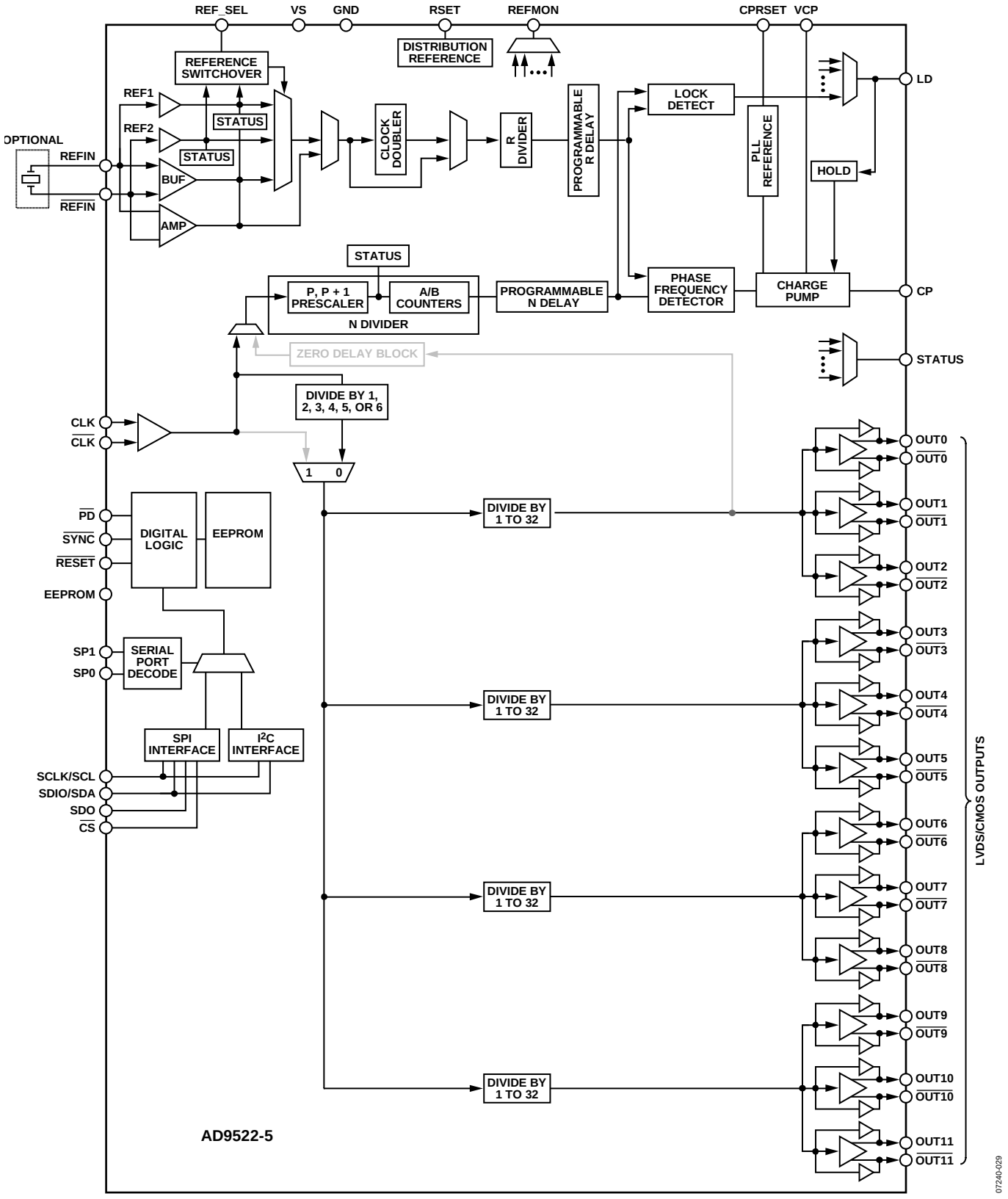


*Figure 27. High Frequency Clock Distribution—CLK or External VCO > 1600 MHz (Mode 2)*

### Phase-Locked Loop (PLL)

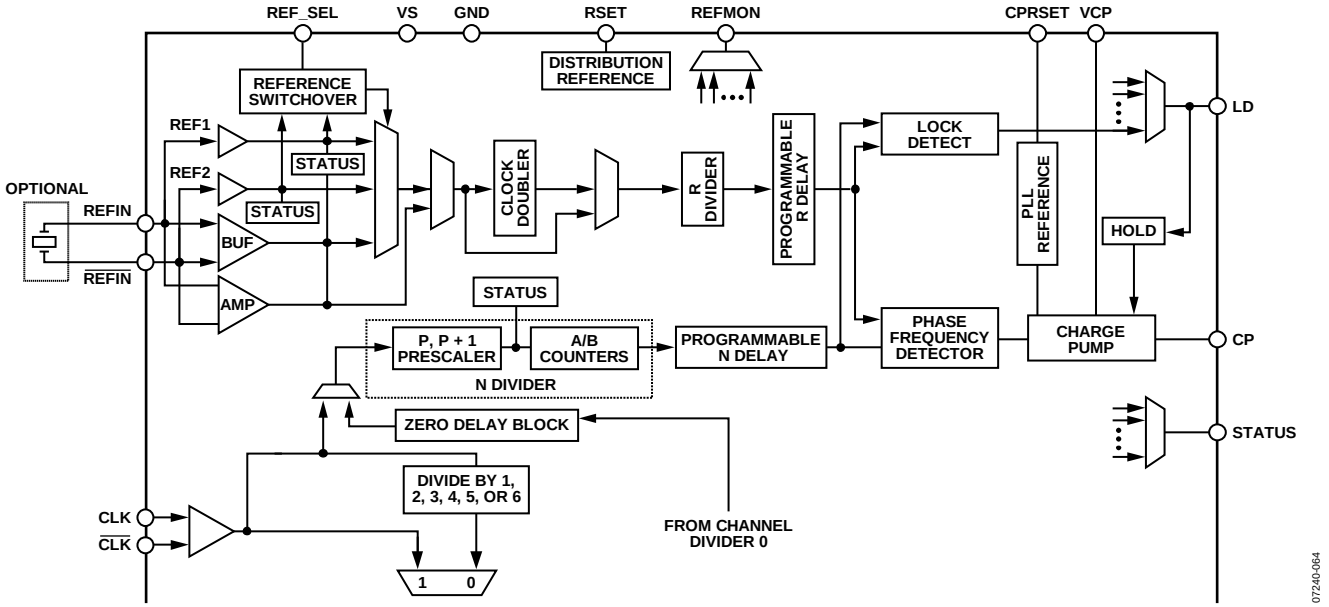


*Figure 28. PLL Functional Block Diagram*

The AD9522 includes on-chip PLL blocks that can be used with an external VCO or VCXO to create a complete phase-locked loop. The PLL requires an external loop filter, which usually consists of a small number of capacitors and resistors. The configuration and components of the loop filter help to establish the loop bandwidth and stability of the PLL.

The AD9522 PLL is useful for generating clock frequencies from a supplied reference frequency. This includes conversion of reference frequencies to much higher frequencies for subsequent division and distribution. In addition, the PLL can be used to clean up jitter and phase noise on a noisy reference. The exact choice of PLL parameters and loop dynamics is application specific. The flexibility and depth of the AD9522 PLL allow the part to be tailored to function in many different applications and signal environments.

#### Configuration of the PLL

Configuration of the PLL is accomplished by programming the various settings for the R divider, N divider, PFD polarity, and charge pump current. The combination of these settings determines the PLL loop bandwidth. These are managed through programmable register settings (see Table 43 and Table 47) and by the design of the external loop filter.

Successful PLL operation and satisfactory PLL loop performance are highly dependent on proper configuration of the PLL settings, and the design of the external loop filter is crucial to the proper operation of the PLL.

ADIsimCLK™ is a free program that can help with the design and exploration of the capabilities and features of the AD9522, including the design of the PLL loop filter. The UG-077 is the AD9522 evaluation software user guide that allows users to easily set the correct register values when the desired configuration is determined. Both are available at www.analog.com/clocks.

#### Phase Frequency Detector (PFD)

The PFD takes inputs from the R divider and the N divider and produces an output proportional to the phase and frequency difference between them. The PFD includes a programmable delay element that controls the width of the antibacklash pulse. This pulse ensures that there is no dead zone in the PFD transfer function and minimizes phase noise and reference spurs. The antibacklash pulse width is set by Register 0x017[1:0].

An important limit to keep in mind is the maximum frequency allowed into the PFD. The maximum input frequency into the PFD is a function of the antibacklash pulse setting, as specified in the phase/frequency detector (PFD) parameter in Table 2.

#### Charge Pump (CP)

The charge pump is controlled by the PFD. The PFD monitors the phase and frequency relationship between its two inputs and tells the CP to pump up or pump down to charge or discharge the integrating node (part of the loop filter). The integrated and filtered CP current is transformed into a voltage that drives the tuning node of the external VCO to move the VCO frequency up or down. The CP can be set (0x010[3:2]) for high impedance (allows holdover operation), for normal operation (attempts to lock the PLL loop), for pump-up, or for pump-down (test modes). The CP current is programmable in eight steps from (nominally) 0.6 mA to 4.8 mA. The exact value of the CP current LSB is set by the CPRSET resistor and is calculated by

$$I_{CP} = \frac{3.06}{CPRSET}$$

### PLL External Loop Filter

An example of an external loop filter for a PLL is shown in Figure 29. A loop filter must be calculated for each desired PLL configuration. The values of the components depend upon the VCO frequency, the $K_{VCO}$, the PFD frequency, the CP current, the desired loop bandwidth, and the desired phase margin. The loop filter affects the phase noise, the loop settling time, and the loop stability. A basic knowledge of PLL theory is helpful for understanding loop filter design. ADIsimCLK can help with the calculation of a loop filter according to the application requirements.

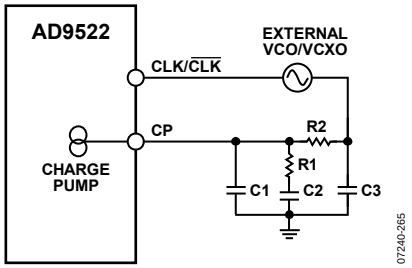


*Figure 29. Example of External Loop Filter for PLL*

### PLL Reference Inputs

The AD9522 features a flexible PLL reference input circuit that allows a fully differential input, two separate single-ended inputs, or a 16.62 MHz to 33.33 MHz crystal oscillator with an on-chip maintaining amplifier. An optional reference clock doubler can be used to double the PLL reference frequency. The input frequency range for the reference inputs is specified in Table 2. Both the differential and the single-ended inputs are self-biased, allowing for easy ac coupling of input signals.

Either a differential or a single-ended reference must be specifically enabled. All PLL reference inputs are off by default.

The differential input and the single-ended inputs share two pins, REFIN (REF1) and $\overline{\text{REFIN}}$ (REF2). The desired reference input type is selected and controlled by Register 0x01C (see Table 43 and Table 47).

When the differential reference input is selected, the self-bias level of the two sides is offset slightly (~100 mV, see Table 2) to prevent chattering of the input buffer when the reference is slow or missing. This increases the voltage swing that is required of the driver and overcomes the offset. The differential reference input can be driven by either ac-coupled LVDS or ac-coupled LVPECL signals.

The single-ended inputs can be driven by either a dc-coupled CMOS level signal or an ac-coupled sine wave or square wave. To avoid input buffer chatter when a single-ended, ac-coupled input signal stops toggling, set Register 0x018[7] to 1b. This setting shifts the dc offset bias point down 140 mV. To increase isolation and reduce power, each single-ended input can be independently powered down.

The differential reference input receiver is powered down when it is not selected or when the PLL is powered down. The single-ended buffers power down when the PLL is powered down or when their respective individual power-down registers are set. When the differential mode is selected, the single-ended inputs are powered down.

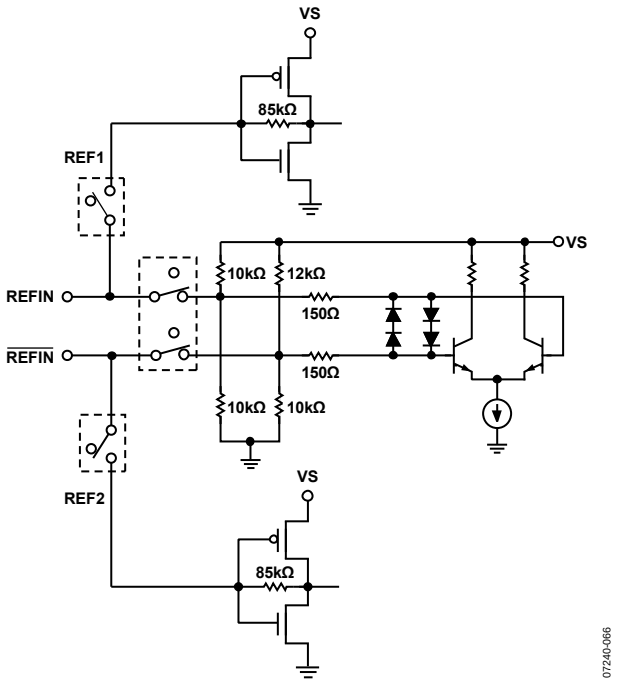


*Figure 30. REFIN Equivalent Circuit for Non-XTAL Mode*

In differential mode, the reference input pins are internally self-biased so that they can be ac-coupled via capacitors. It is possible to dc couple to these inputs. If the differential REFIN is driven by a single-ended signal, decouple the unused side ($\overline{\text{REFIN}}$) via a suitable capacitor to a quiet ground. Figure 30 shows the equivalent circuit of REFIN.

Crystal mode is nearly identical to differential mode. The user enables a maintaining amplifier by setting the enable XTAL OSC bit, and putting a series resonant, AT fundamental cut crystal across the REFIN/$\overline{\text{REFIN}}$ pins.

### Reference Switchover

The AD9522 supports dual single-ended CMOS inputs, as well as a single differential reference input. In the dual single-ended reference mode, the AD9522 supports automatic revertive and manual PLL reference clock switching between REF1 (on Pin REFIN) and REF2 (on Pin $\overline{\text{REFIN}}$). This feature supports networking and other applications that require redundant references. When used in conjunction with the automatic holdover function, the AD9522 can achieve a worst-case reference input switchover with an output frequency disturbance as low as 10 ppm.

The AD9522 features a dc offset option in single-ended mode. This option is designed to eliminate the risk of the reference inputs chattering when they are ac-coupled and the reference clock disappears. When using the reference switchover, the single-

ended reference inputs must be dc-coupled CMOS levels (with the AD9522 dc offset feature disabled). Alternatively, the inputs can be ac-coupled and dc offset feature enabled. Keep in mind, however, that the minimum input amplitude for the reference inputs is greater when the dc offset is turned on.

Reference switchover can be performed manually or automatically. Manual switchover is performed either through Register 0x01C or by using the REF_SEL pin. Manual switchover requires the presence of a clock on the reference input that is being switched to; otherwise, the deglitching feature must be disabled in Bit 7 of Register 0x01C. The reference switching logic fails if this condition is not met, and the PLL does not reacquire.

Automatic revertive switchover relies on the REFMON pin to indicate when REF1 disappears. By programming Register 0x01B = 0xF7 and Register 0x01C = 0x26, the REFMON pin is programmed high when REF1 is invalid, which commands the switch to REF2. When REF1 is valid again, the REFMON pin goes low, and the device again locks to REF1. The STATUS pin can also be used for this function, and REF2 can be used as the preferred reference.

A switchover deglitch feature ensures that the PLL does not receive rising edges that are far out of alignment with the newly selected reference. For the switchover deglitch feature to work correctly, the presence of a clock is required on the reference input that is being switched to. The deglitching feature can also be disabled (Register 0x01C[7]).

Automatic nonrevertive switching is not supported.

### *Reference Divider R*

The reference inputs are routed to the reference divider, R. R (a 14-bit counter) can be set to any value from 0 to 16,383 by writing to Register 0x011 and Register 0x012. (Both R = 0 and R = 1 give divide-by-1.) The output of the R divider goes to one of the PFD inputs to be compared with the VCO frequency divided by the N divider. The frequency applied to the PFD must not exceed the maximum allowable frequency, which depends on the antibacklash pulse setting (see Table 2).

The R divider has its own reset. The R divider can be reset using the shared reset bit of the R, A, and B counters. It can also be reset by a $\overline{\text{SYNC}}$ operation.

### *VCO/VCXO Feedback Divider N: P, A, B*

The N divider is a combination of a prescaler (P) and two counters, A and B. The total divider value is

$$N = (P \times B) + A$$

where $P$ can be 2, 4, 8, 16, or 32.

#### Prescaler

The prescaler of the AD9522 allows for two modes of operation: a fixed divide (FD) mode of 1, 2, or 3, and a dual modulus (DM) mode where the prescaler divides by P and (P + 1) {2 and 3, 4 and 5, 8 and 9, 16 and 17, or 32 and 33}. The prescaler modes of operation are given in Table 47, Register 0x016[2:0]. Not all modes are available at all frequencies (see Table 2). It is common to use a prescaler of 8 for VC frequencies <2400 MHz, and P = 32 is usually used for very large feedback divider values only.

When operating the AD9522 in dual modulus mode, P/(P + 1), the equation used to relate the input reference frequency to the VCO output frequency is

$$f_{VCO} = (f_{REF}/R) \times (P \times B + A) = f_{REF} \times N/R$$

However, when operating the prescaler in FD Mode 1, FD Mode 2, or FD Mode 3, the A counter is not used (A = 0; the divide is a fixed divide of P = 2, 4, 8, 16, or 32) and the equation simplifies to

$$f_{VCO} = (f_{REF}/R) \times (P \times B) = f_{REF} \times N/R$$

By using combinations of DM and FD modes, the AD9522 can achieve values of N from 1 to 262,175.

Table 25 shows how a 10 MHz reference input can be locked to any integer multiple of N. Note that the same value of N can be derived in different ways, as illustrated by the case of N = 12. The user can choose a fixed divide mode of P = 2 with B = 6, use the dual modulus mode of 2/3 with A = 0, B = 6, or use the dual modulus mode of 4/5 with A = 0, B = 3.

#### A and B Counters

The B counter must be ≥3 or bypassed, and unlike the R counter, A = 0 is actually zero. The B counter must always be greater than or equal to the A counter.

The maximum input frequency to the A/B counter is reflected in the maximum prescaler output frequency (~300 MHz) specified in Table 2. This is the prescaler input frequency (external VCO or CLK) divided by P. For example, dual modulus P = 8/9 mode is not allowed if the VCO frequency is greater than 2400 MHz because the frequency going to the A/B counter is too high.

When the AD9522 B counter is bypassed (B = 1), the A counter must be set to zero, and the overall resulting divide is equal to the prescaler setting, P. The possible divide ratios in this mode are 1, 2, 3, 4, 8, 16, and 32.

Although manual reset is not normally required, the A/B counters have their own reset bit. Alternatively, the A and B counters can be reset using the shared reset bit of the R, A, and B counters. Note that these reset bits are not self-clearing.

#### R, A, and B Counters: $\overline{\text{SYNC}}$ Pin Reset

The R, A, and B counters can be reset simultaneously through the $\overline{\text{SYNC}}$ pin. This function is controlled by Register 0x019[7:6] (see Table 47). The $\overline{\text{SYNC}}$ pin reset is disabled by default.

#### R and N Divider Delays

Both the R and N dividers feature a programmable delay cell. These delays can be enabled to allow adjustment of the phase relationship between the PLL reference clock and the VCO or CLK, and are useful for controlling the input/output phase relationship in zero delay mode. Each delay is controlled by three bits. The total delay range is about 1 ns. See Register 0x019 in Table 2 and Table 47.

**Table 25. How a 10 MHz Reference Input Can Be Locked to Any Integer Multiple of N (X= Don't Care)**

| $f_{REF}$ (MHz) | R | P | A | B | N | $f_{VCO}$ (MHz) | Mode | Notes |
|---|---|---|---|---|---|---|---|---|
| 10 | 1 | 1 | X | 1 | 1 | 10 | FD | P = 1, B = 1 (A and B counters are bypassed). |
| 10 | 1 | 2 | X | 1 | 2 | 20 | FD | P = 2, B = 1 (A and B counters are bypassed). |
| 10 | 1 | 1 | X | 3 | 3 | 30 | FD | A counter is bypassed. |
| 10 | 1 | 1 | X | 4 | 4 | 40 | FD | A counter is bypassed. |
| 10 | 1 | 1 | X | 5 | 5 | 50 | FD | A counter is bypassed. |
| 10 | 1 | 2 | X | 3 | 6 | 60 | FD | A counter is bypassed. |
| 10 | 1 | 2 | 0 | 3 | 6 | 60 | DM | |
| 10 | 1 | 2 | 1 | 3 | 7 | 70 | DM | Maximum frequency into prescaler in P 2/3 mode is 200 MHz. If N = 7 or N = 11 is desired for prescaler input frequency of 200 MHz to 300 MHz, use P = 1 and N = 7, or 11, respectively. |
| 10 | 1 | 2 | 2 | 3 | 8 | 80 | DM | |
| 10 | 1 | 2 | 1 | 4 | 9 | 90 | DM | |
| 10 | 1 | 8 | 6 | 18 | 150 | 1500 | DM | |
| 10 | 1 | 8 | 7 | 18 | 151 | 1510 | DM | |
| 10 | 1 | 16 | 7 | 9 | 151 | 1510 | DM | |
| 10 | 10 | 32 | 6 | 47 | 151 | 1510 | DM | |
| 10 | 1 | 8 | 0 | 25 | 200 | 2000 | DM | |
| 10 | 1 | 16 | 0 | 15 | 240 | 2400 | DM | |
| 10 | 10 | 32 | 0 | 75 | 2400 | 2400 | DM | |

### *Digital Lock Detect (DLD)*

By selecting the proper output through the mux on each pin, the DLD function is available at the LD, STATUS, and REFMON pins. The digital lock detect circuit indicates a lock when the time difference of the rising edges at the PFD inputs is less than a specified value (the lock threshold). The loss of a lock is indicated when the time difference exceeds a specified value (the unlock threshold). Note that the unlock threshold is wider than the lock threshold, which allows some phase error in excess of the lock window to occur without chattering on the lock indicator.

The lock detect window timing depends on the value of the CPRSET resistor, as well as three settings: the digital lock detect window bit (Register 0x018[4]), the antibacklash pulse width bit (Register 0x017[1:0], see Table 2), and the lock detect counter (Register 0x018[6:5]). The lock and unlock detection values in Table 2 are for the nominal value of CPRSET = 5.1 kΩ. Doubling the CPRSET value to 10 kΩ doubles the values in Table 2.

A lock is not indicated until there is a programmable number of consecutive PFD cycles with a time difference less than the lock detect threshold. The lock detect circuit continues to indicate a lock until a time difference greater than the unlock threshold occurs on a single subsequent cycle. For the lock detect to work properly, the period of the PFD frequency must be greater than the unlock threshold. The number of consecutive PFD cycles required for lock is programmable (Register 0x018[6:5]).

Note that it is possible in certain low (<500 Hz) loop bandwidth, high phase margin cases that the DLD can chatter during acquisition, which can cause the AD9522 to automatically enter and exit holdover. To avoid this problem, it is recommended that the user make provisions for a capacitor to ground on the LD pin so that current source digital lock detect (CSDLD) mode can be used.

### *Analog Lock Detect (ALD)*

The AD9522 provides an ALD function that can be selected for use at the LD pin. There are two operating modes for ALD.

- N-channel open-drain lock detect. This signal requires a pull-up resistor to the positive supply, VS. The output is normally high with short, low going pulses. Lock is indicated by the minimum duty cycle of the low going pulses.
- P-channel open-drain lock detect. This signal requires a pull-down resistor to GND. The output is normally low with short, high going pulses. Lock is indicated by the minimum duty cycle of the high going pulses.

The analog lock detect function requires an RC filter to provide a logic level indicating lock/unlock. The ADIsimCLK tool can be used to help the user select the right passive component values for ALD to ensure its correct operation.

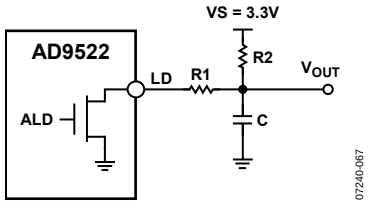


*Figure 31. Example of Analog Lock Detect Filter Using N-Channel Open-Drain Driver*

### *Current Source Digital Lock Detect (CSDLD)*

During the PLL locking sequence, it is normal for the DLD signal to toggle a number of times before remaining steady. There may be applications where it is necessary to have DLD asserted without chattering and only after the PLL is solidly locked. This is possible by using the current source digital lock detect function.

The current source lock detect provides a current of 110 µA when DLD is true and shorts to ground when DLD is false. If a capacitor is connected to the LD pin, it charges at a rate determined by the current source during the DLD true time but is discharged nearly instantly when DLD is false. By monitoring the voltage at the LD pin (top of the capacitor), LD = high happens only after the DLD is true for a sufficiently long time. Any momentary DLD false resets the charging. By selecting a properly sized capacitor, it is possible to delay a lock detect indication until the PLL is stably locked and the lock detect does not chatter.

To use current source digital lock detect, do the following:

- Place a capacitor to ground on the LD pin
- Set Register 0x01A[5:0] = 0x04
- Enable the LD pin comparator (0x01D[3] = 1)

The LD pin comparator senses the voltage on the LD pin, and the comparator output can be made available at the REFMON pin control (Register 0x01B[4:0]) or the STATUS pin control (Register 0x017[7:2]). The internal LD pin comparator trip point and hysteresis are given in Table 14. The voltage on the capacitor can also be sensed by an external comparator connected to the LD pin. In this case, enabling the on-board LD pin comparator is not necessary.

The user can asynchronously enable individual clock outputs only when CSDLD is high. To enable this feature, set the appropriate bits in the enable output on the CSDLD registers (Register 0x0FC and Register 0x0FD).

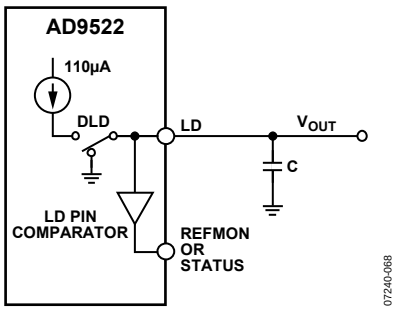


*Figure 32. Current Source Digital Lock Detect*

### External VCXO/VCO Clock Input (CLK/$\overline{\text{CLK}}$)

This differential input is used to drive the AD9522 clock distribution section. This input can receive up to 2.4 GHz. The pins are internally self-biased, and the input signal must be ac-coupled via capacitors.

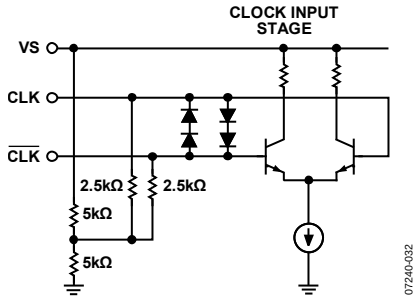


*Figure 33. CLK Equivalent Input Circuit*

The self-biased CLK/$\overline{\text{CLK}}$ input can be used either as a distribution only input (with the PLL off) or as a feedback input for an external VCO/VCXO using the internal PLL when the internal VCO is not used. These inputs are also used as a feedback path for the external zero delay mode.

### Holdover

The AD9522 PLL has a holdover function. Holdover mode allows the VCO to maintain a relatively constant frequency even though there is no reference clock. This function is useful when the PLL reference clock is lost. Holdover is implemented by placing the charge pump in a high impedance state. Without this function, the charge pump is placed into a constant pump-up or pump-down state, resulting in a significant VCO frequency shift. Because the charge pump is placed in a high impedance state, any leakage that occurs at the charge pump output or the VCO tuning node causes a drift of the VCO frequency. This drift can be mitigated by using a loop filter that contains a large capacitive component because this drift is limited by the current leakage induced slew rate ($I_{LEAK}$/C) of the VCO control voltage.

Both a manual holdover mode, using the $\overline{\text{SYNC}}$ pin, and an automatic holdover mode are provided. To use either function, the holdover function must be enabled (Register 0x01D[0]).

### External/Manual Holdover Mode

A manual holdover mode can be enabled that allows the user to place the charge pump into a high impedance state when the $\overline{\text{SYNC}}$ pin is asserted low. This operation is edge sensitive, not level sensitive. The charge pump enters a high impedance state immediately. To take the charge pump out of a high impedance state, take the $\overline{\text{SYNC}}$ pin high. The charge pump then leaves the high impedance state synchronously with the next PFD rising edge from the reference clock. This prevents extraneous charge pump events from occurring during the time between $\overline{\text{SYNC}}$ going high and the next PFD event. This also means that the charge pump stays in a high impedance state if there is no reference clock present.

The B counter (in the N divider) is reset synchronously with the charge pump leaving the high impedance state on the reference path PFD event. This helps align the edges out of the R and N dividers for faster settling of the PLL. Because the prescaler is not reset, this feature works best when the B and R numbers are close because this results in a smaller phase difference for the loop to settle out.

When using this mode, the channel dividers must be set to ignore the $\overline{\text{SYNC}}$ pin (at least after an initial $\overline{\text{SYNC}}$ event). If the dividers are not set to ignore the $\overline{\text{SYNC}}$ pin, any time $\overline{\text{SYNC}}$ is taken low to put the part into holdover, the distribution outputs turn off. The channel divider ignore SYNC function is found in Register 0x191[6], Register 0x194[6], Register 0x197[6], and Register 0x19A[6] for Channel Divider 0, Channel Divider 1, Channel Divider 2, and Channel Divider 3, respectively.

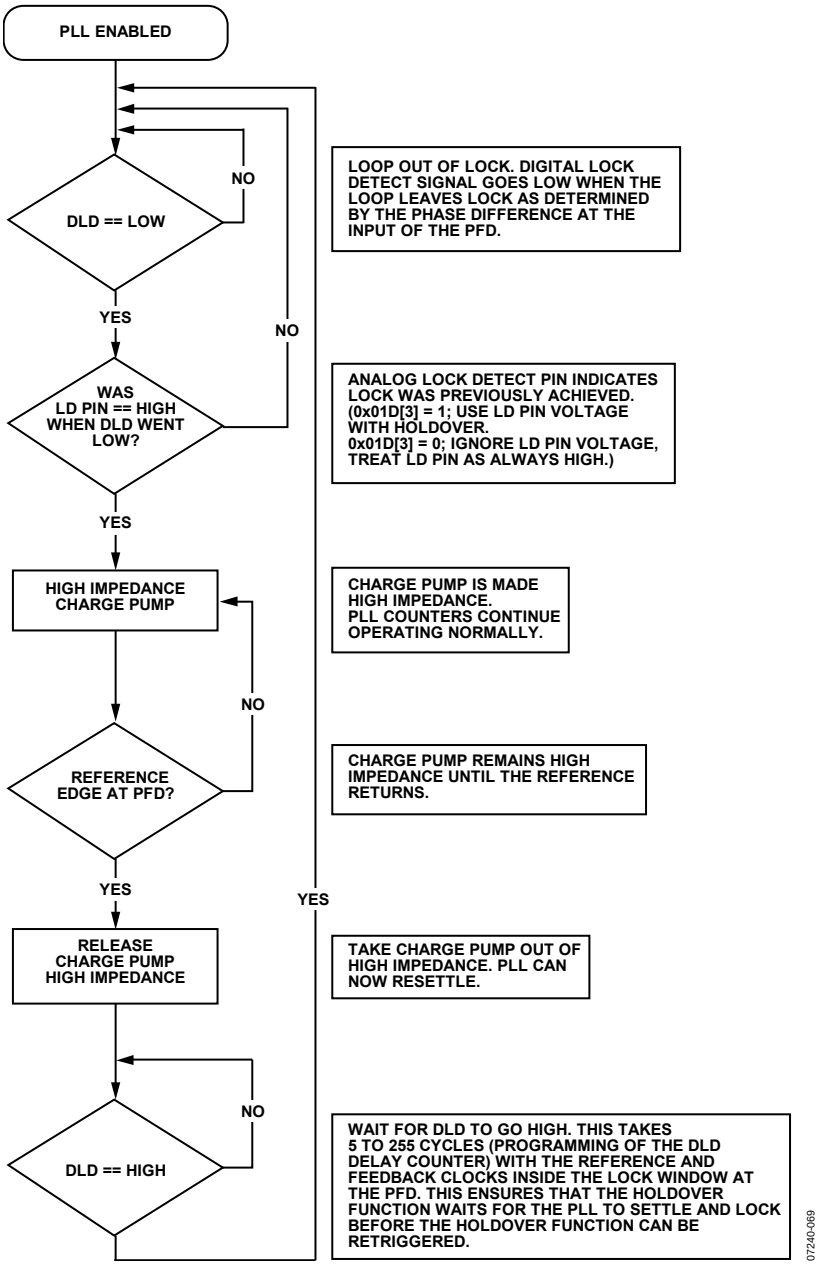


*Figure 34. Flowchart of Automatic/Internal Holdover Mode*

### *Automatic/Internal Holdover Mode*

When enabled, this function automatically puts the charge pump into a high impedance state when the loop loses lock. The assumption is that the only reason the loop loses lock is due to the PLL losing the reference clock; therefore, the holdover function puts the charge pump into a high impedance state to maintain the VCO frequency as close as possible to the original frequency before the reference clock disappeared.

A flowchart of the automatic/internal holdover function operation is shown in Figure 34.

The holdover function senses the logic level of the LD pin as a condition to enter holdover. The signal at LD can be from the DLD, ALD, or current source LD (CSDLD) mode. It is possible to disable the LD comparator (0x01D[3]), which causes the holdover function to always sense LD as high. If DLD is used, it is possible for the DLD signal to chatter while the PLL is reacquiring lock. The holdover function may retrigger, thereby preventing the holdover mode from terminating. Use of the current source lock detect mode is recommended to avoid this situation (see the Current Source Digital Lock Detect (CSDLD) section).

When in holdover mode, the charge pump stays in a high impedance state as long as there is no reference clock present.

As in the external holdover mode, the B counter (in the N divider) is reset synchronously with the charge pump leaving the high impedance state on the reference path PFD event. This helps align the edges out of the R and N dividers for faster settling of the PLL and reduces frequency errors during settling. Because

the prescaler is not reset, this feature works best when the B and R numbers are close because this results in a smaller phase difference for the loop to settle out.

After leaving holdover, the loop then reacquires lock and the LD pin must go high (if Register 0x01D[3] = 1) before it can reenter holdover.

The holdover function always responds to the state of the currently selected reference (Register 0x01C). If the loop loses lock during a reference switchover (see the Reference Switchover section), holdover is triggered briefly until the next reference clock edge at the PFD.

The following registers affect the automatic/internal holdover function:

- Register 0x018[6:5]—lock detect counter. This changes how many consecutive PFD cycles with edges inside the lock detect window are required for the DLD indicator to indicate lock. This impacts the time required before the LD pin can begin to charge, as well as the delay from the end of a holdover event until the holdover function can be reengaged.
- Register 0x018[3]—disable digital lock detect. This bit must be set to 0 to enable the DLD circuit. Internal/automatic holdover does not operate correctly without the DLD function enabled.
- Register 0x01A[5:0]—lock detect pin control. Set this to 000100b to put it in the current source lock detect mode if using the LD pin comparator. Load the LD pin with a capacitor of an appropriate value.
- Register 0x01D[3]—enable LD pin comparator. 1 = enable; 0 = disable. When disabled, the holdover function always senses the LD pin as high.
- Register 0x01D[1]—external holdover control.
- Register 0x01D[0]—enable holdover. If holdover is disabled, both external and automatic/internal holdover are disabled.

In the following example, automatic holdover is configured with

- Automatic reference switchover, prefer REF1.
- Digital lock detect: five PFD cycles, high range window.
- Automatic holdover using the LD pin comparator.

The following registers are set (in addition to the normal PLL registers):

- Register 0x018[6:5] = 00b; lock detect counter = five cycles.
- Register 0x018[4] = 0b; digital lock detect window = high range.
- Register 0x018[3] = 1b; disable DLD normal operation.
- Register 0x01A[5:0] = 000100b; program LD pin control to current source lock detect mode.
- Register 0x01C[4] = 1b; enable automatic switchover.
- Register 0x01C[3] = 0b; prefer REF1.
- Register 0x01C[2:1] = 11b; enable REF1 and REF2 input buffers.
- Register 0x01D[3] = 1b; enable LD pin comparator.
- Register 0x01D[1] = 0b; disable external holdover mode and use automatic/internal holdover mode.
- Register 0x01D[0] = 1b; enable holdover.

### *Frequency Status Monitors*

The AD9522 contains three frequency status monitors that are used to indicate if the PLL reference (or references in the case of single-ended mode) and the external VCO/CLK input fall below a threshold frequency. A diagram showing their location in the PLL is shown in Figure 35.

The PLL reference monitors have two threshold frequencies: normal and extended (see Table 14). The reference frequency monitor thresholds are selected in Register 0x01F.

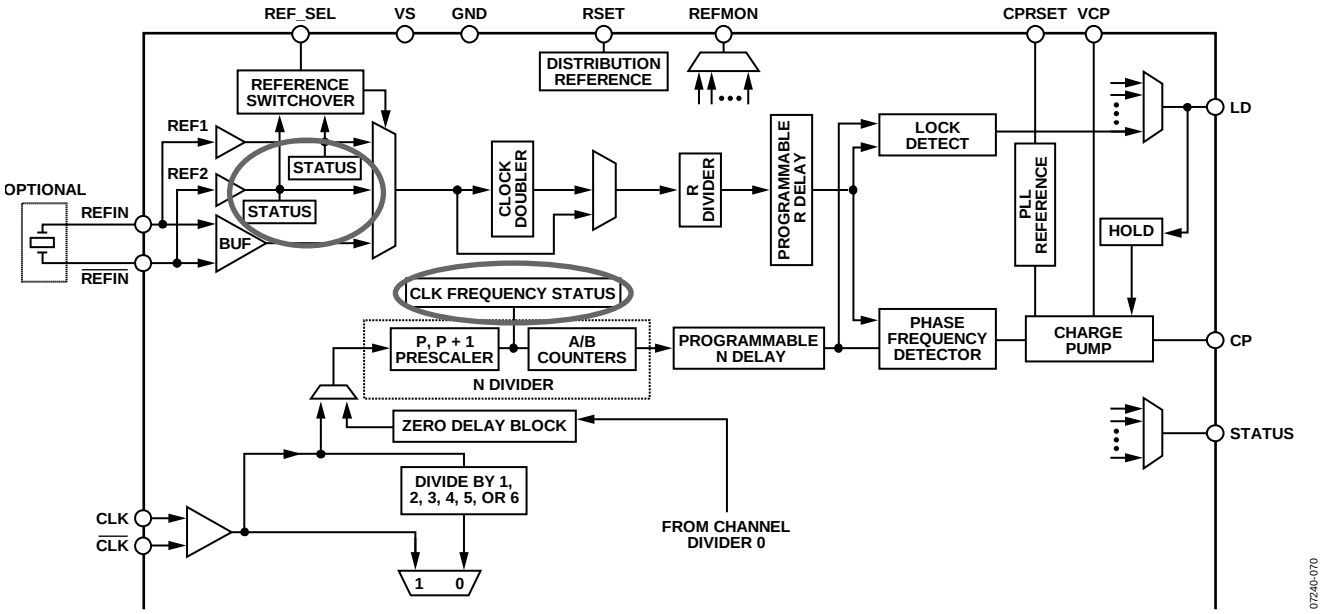


*Figure 35. Reference and CLK Frequency Status Monitors*

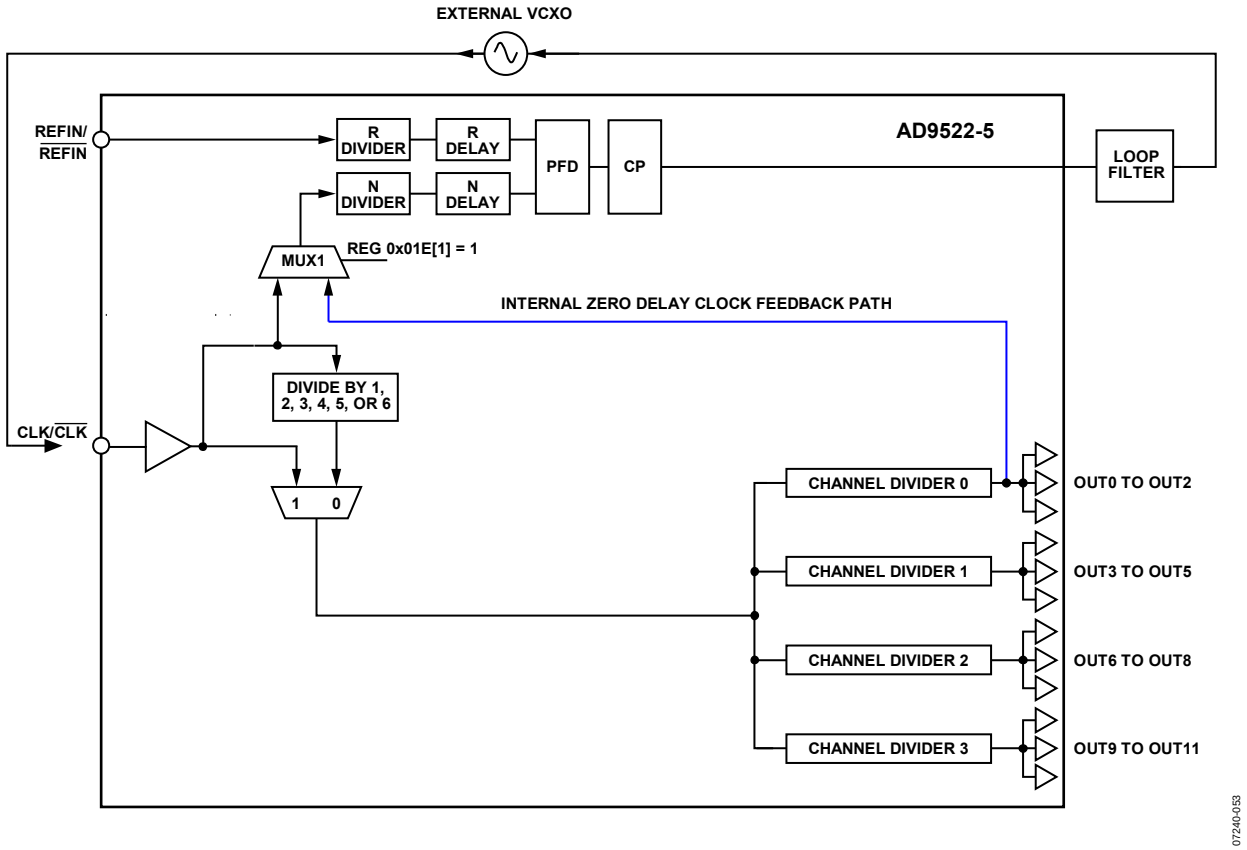


*Figure 36. Zero Delay Function*

## ZERO DELAY OPERATION

Zero delay operation aligns the phase of the output clocks with the phase of the external PLL reference input.

The zero delay function of the AD9522 is achieved by feeding the output of Channel Divider 0 back to the PLL N divider. In Figure 36, the change in signal routing for zero delay mode is shown in blue.

Set Register 0x01E[1] = 1b to select the zero delay mode. In the zero delay mode, the output of Channel Divider 0 is routed back to the PLL (N divider) through MUX1 (feedback path shown in blue in Figure 36). The PLL synchronizes the phase/edge of the output of Channel Divider 0 with the phase/edge of the reference input.

Because the channel dividers are synchronized to each other, the outputs of the channel dividers are synchronous with the reference input. Both the R delay and the N delay inside the PLL can be programmed to compensate for the propagation delay from the output drivers and PLL components to minimize the phase offset between the clock output and the reference input to achieve zero delay.

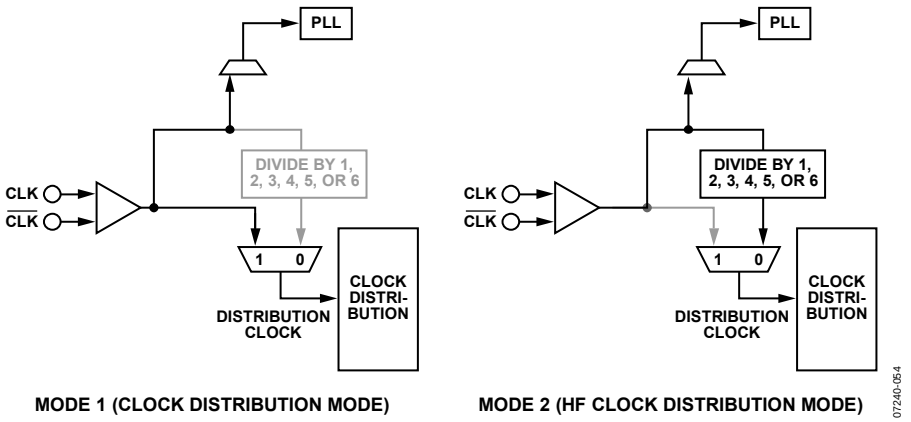


*Figure 37. Simplified Diagram of the Two Clock Distribution Operation Modes*

## CLOCK DISTRIBUTION

A clock channel consists of three LVDS clock outputs or six CMOS clock outputs that share a common divider. A clock output consists of the drivers that connect to the output pins. The clock outputs have either LVDS or CMOS at the pins.

The AD9522 has four clock channels. Each channel has its own programmable divider that divides the clock frequency applied to its input. The channel dividers can divide by any integer from 1 to 32. Divide by 1 is achieved by setting the Divider n bypass bit (where n = 0 through 3) in the appropriate channel divider register.

The AD9522 features a VCO divider that divides the VCO output by 1, 2, 3, 4, 5, or 6 before going to the individual channel dividers. The VCO divider has two purposes. The first is to limit the maximum input frequency of the channel dividers to 1.6 GHz. See the VCO Divider section for discussion of special cases where the channel divider maximum input frequency is >1.6 GHz. The other is to allow the AD9522 to generate lower output frequencies than is normally possible with only a simple post divider. External clock signals connected to the CLK input can also use the VCO divider.

The channel dividers allow for a selection of various duty cycles, depending on the currently set division. That is, for any specific division, D, the output of the divider can be set to high for N + 1 input clock cycles and low for M + 1 input clock cycles (where D = N + M + 2). For example, a divide-by-5 can be high for one divider input cycle and low for four cycles, or a divide-by-5 can be high for three divider input cycles and low for two cycles. Other combinations are also possible.

The channel dividers include a duty-cycle correction function that can be disabled. In contrast to the selectable duty cycle just described, this function can correct a non-50% duty cycle caused by an odd division. However, this requires that the division be set by M = N + 1. Note that the duty cycle correction feature is not available when the VCO divider is set to 1.

In addition, the channel dividers allow a coarse phase offset or delay to be set. Depending on the division selected, the output can be delayed by up to 15 input clock cycles. For example, if the frequency at the input of the channel divider is 1 GHz, the channel divider output can be delayed by up to 15 ns. The divider outputs can also be set to start high or to start low.

### *Operation Modes*

The AD9522 has two clock distribution operating modes that are shown in Figure 37.

It is not necessary to use the VCO divider if the CLK frequency is less than the maximum channel divider input frequency (1600 MHz); otherwise, the VCO divider must be used to reduce the frequency going to the channel dividers.

Table 26 shows how the operation modes are selected. 0x1E1[0] selects the channel divider source.

**Table 26. Operation Modes**

| Mode | 0x1E1[0] | VCO Divider |
|---|---|---|
| 2 | 0 | Used |
| 1 | 1 | Not used |

### *Clock Frequency Division*

The total frequency division is a combination of the VCO divider (when used) and the channel divider. When the VCO divider is used, the total division from the CLK input to the output is the product of the VCO divider (1, 2, 3, 4, 5, or 6) and the division of the channel divider. Table 27 indicates how the frequency division for a channel is set.

**Table 27. Frequency Division**

| VCO Divider Setting[1] | Channel Divider Setting | Resulting Frequency Division |
|---|---|---|
| 1 to 6 | 2 to 32 | (1 to 6) × (2 to 32) |
| 2 to 6 | Bypass | (2 to 6) × (1) |
| 1 | Bypass | Output static (illegal state) |
| VCO divider bypassed | Bypass | 1 |
| VCO divider bypassed | 2 to 32 | 2 to 32 |

[1] The bypass VCO divider (0x1E1[0] = 1) is not the same as VCO divider = 1.

The channel dividers feeding the output drivers contain one 2-to-32 frequency divider. This divider provides for division-by-1 to division-by-32. Division-by-1 is accomplished by bypassing the divider. The dividers also provide for a programmable duty cycle, with optional duty-cycle correction when the divide ratio is odd. A phase offset or delay in increments of the input clock cycle is selectable. The channel dividers operate with a signal at their inputs up to 1600 MHz. The features and settings of the dividers are selected by programming the appropriate setup and control registers (see Table 43 through Table 54).

### *VCO Divider*

The VCO divider provides frequency division between the CLK input and the clock distribution channel dividers. The VCO divider can be set to divide by 1, 2, 3, 4, 5, or 6 (see Table 50, 0x1E0[2:0]). However, when the VCO divider is set to 1, none of the channel output dividers can be bypassed.

The VCO divider can also be set to static, which is useful for applications where the only desired output frequency is the CLK input frequency. Making the VCO divider static increases the wideband spurious-free dynamic range (SFDR). The same improvement in SFDR performance can also be achieved by setting the VCO divider to 1.

### *Channel Dividers*

A channel divider drives each group of three LVDS outputs. There are four channel dividers (0, 1, 2, and 3) driving 12 LVDS outputs (OUT0 to OUT11). Table 28 gives the register locations used for setting the division and other functions of these dividers. The division is set by the values of M and N. The divider can be bypassed (equivalent to divide-by-1, divider circuit is powered down) by setting the bypass bit. The duty-cycle correction can be enabled or disabled according to the setting of the disable divider DCC bits.

**Table 28. Setting $D_X$ for the Output Dividers**

| Divider | Low Cycles M | High Cycles N | Bypass | Disable Div DCC |
|---|---|---|---|---|
| 0 | 0x190[7:4] | 0x190[3:0] | 0x191[7] | 0x192[0] |
| 1 | 0x193[7:4] | 0x193[3:0] | 0x194[7] | 0x195[0] |
| 2 | 0x196[7:4] | 0x196[3:0] | 0x197[7] | 0x198[0] |
| 3 | 0x199[7:4] | 0x199[3:0] | 0x19A[7] | 0x19B[0] |

#### Channel Divider Maximum Frequency

The maximum frequency at which all features of the channel divider are guaranteed to work is 1.6 GHz; this is the number that appears elsewhere in the data sheet. However, if the divide-by-3 and divide-by-17 settings are avoided, the maximum channel divider input frequency is 2 GHz.

#### Channel Frequency Division (0, 1, 2, and 3)

For each channel (where the channel number x is 0, 1, 2, or 3), the frequency division, $D_X$, is set by the values of M and N (four bits each, representing Decimal 0 to Decimal 15), where

*Number of Low Cycles* = *M* + 1

*Number of High Cycles* = *N* + 1

The high and low cycles are cycles of the clock signal currently routed to the input of the channel dividers (VCO divider out or CLK).

When a divider is bypassed, $D_X = 1$.

Otherwise, $D_X = (N + 1) + (M + 1) = N + M + 2$. This allows each channel divider to divide by any integer from 1 to 32.

#### Duty Cycle and Duty-Cycle Correction

The duty cycle of the clock signal at the output of a channel is a result of some or all of the following conditions:

- The M and N values for the channel
- DCC enabled/disabled
- VCO divider enabled/bypassed
- The CLK input duty cycle

The DCC function is enabled by default for each channel divider. However, the DCC function can be disabled individually for each channel divider by setting the disable divider DCC bit for that channel.

Certain M and N values for a channel divider result in a non-50% duty cycle. A non-50% duty cycle can also result with an even division, if M ≠ N. The duty-cycle correction function automatically corrects non-50% duty cycles at the channel divider output to 50% duty cycle. Note that the duty-cycle correction feature is not available when the VCO divider is set to 1.

Duty-cycle correction requires the following channel divider conditions:

- An even division must be set as M = N.
- An odd division must be set as M = N + 1.

When not bypassed or corrected by the DCC function, the duty cycle of each channel divider output is the numerical value of (N + 1)/(N + M + 2) expressed as a percent.

The duty cycle at the output of the channel divider for various configurations is shown in Table 29 to Table 32.

**Table 29. Channel Divider Output Duty Cycle with VCO Divider ≠ 1, Input Duty Cycle Is 50%**

| | $D_X$ | Output Duty Cycle | |
|---|---|---|---|
| VCO Divider | N + M + 2 | Disable Div DCC = 1 | Disable Div DCC = 0 |
| Even | Channel divider bypassed | 50% | 50% |
| Odd = 3 | Channel divider bypassed | 33.3% | 50% |
| Odd = 5 | Channel divider bypassed | 40% | 50% |
| Even, odd | Even | (N + 1)/(N + M + 2) | 50%, requires M = N |
| Even, odd | Odd | (N + 1)/(N + M + 2) | 50%, requires M = N + 1 |

**Table 30. Channel Divider Output Duty Cycle with VCO Divider ≠ 1, Input Duty Cycle Is X%**

| VCO Divider | $D_X$ N + M + 2 | Output Duty Cycle: Disable Div DCC = 1 | Output Duty Cycle: Disable Div DCC = 0 |
|---|---|---|---|
| Even | Channel divider bypassed | 50% | 50% |
| Odd = 3 | Channel divider bypassed | 33.3% | (1 + X%)/3 |
| Odd = 5 | Channel divider bypassed | 40% | (2 + X%)/5 |
| Even | Even | (N + 1)/(N + M + 2) | 50%, requires M = N |
| Even | Odd | (N + 1)/(N + M + 2) | 50%, requires M = N + 1 |
| Odd = 3 | Even | (N + 1)/(N + M + 2) | 50%, requires M = N |
| Odd = 3 | Odd | (N + 1)/(N + M + 2) | (3N + 4 + X%)/(6N + 9), requires M = N + 1 |
| Odd = 5 | Even | (N + 1)/(N + M + 2) | 50%, requires M = N |
| Odd = 5 | Odd | (N + 1)/(N + M + 2) | (5N + 7 + X%)/(10N + 15), requires M = N + 1 |

**Table 31. Channel Divider Output Duty Cycle When the VCO Divider Is Enabled and Set to 1**

| Input Clock Duty Cycle | $D_X$ N + M + 2 | Output Duty Cycle: Disable Div DCC = 1 | Output Duty Cycle: Disable Div DCC = 0 |
|---|---|---|---|
| Any | Even | (N + 1)/(M + N + 2) | 50%, requires M = N |
| 50% | Odd | (N + 1)/(M + N + 2) | (N + 1)/(M + N + 2) |
| X% | Odd | (N + 1)/(M + N + 2) | (N + 1 + X%)/(2 × N + 3), requires M = N + 1 |

Note that the channel divider must be enabled when the VCO divider = 1.

**Table 32. Channel Divider Output Duty Cycle When the VCO Divider Is Bypassed**

| Input Clock Duty Cycle | $D_X$ N + M + 2 | Output Duty Cycle: Disable Div DCC = 1 | Output Duty Cycle: Disable Div DCC = 0 |
|---|---|---|---|
| Any | Channel divider bypassed | Same as input duty cycle | Same as input duty cycle |
| Any | Even | (N + 1)/(M + N + 2) | 50%, requires M = N |
| 50% | Odd | (N + 1)/(M + N + 2) | 50%, requires M = N + 1 |
| X% | Odd | (N + 1)/(M + N + 2) | (N + 1 + X%)/(2 × N + 3), requires M = N + 1 |

If the CLK input is routed directly to the output, the duty cycle of the output is the same as the CLK input.

### Phase Offset or Coarse Time Delay

Each channel divider allows for a phase offset, or a coarse time delay, to be programmed by setting register bits (see Table 33). These settings determine the number of cycles (successive rising edges) of the channel divider input frequency by which to offset, or delay, the rising edge of the output of the divider. This delay is with respect to a nondelayed output (that is, with a phase offset of zero). The amount of the delay is set by five bits loaded into the phase offset (PO) register plus the start high (SH) bit for each channel divider. When the start high bit is set, the delay is also affected by the number of low cycles (M) programmed for the divider.

It is necessary to use the SYNC function to make phase offsets effective (see the Synchronizing the Outputs— Function section).

**Table 33. Setting Phase Offset and Division**

| Divider | Start High (SH) | Phase Offset (PO) | Low Cycles M | High Cycles N |
|---|---|---|---|---|
| 0 | 0x191[4] | 0x191[3:0] | 0x190[7:4] | 0x190[3:0] |
| 1 | 0x194[4] | 0x194[3:0] | 0x193[7:4] | 0x193[3:0] |
| 2 | 0x197[4] | 0x197[3:0] | 0x196[7:4] | 0x196[3:0] |
| 3 | 0x19A[4] | 0x19A[3:0] | 0x199[7:4] | 0x199[3:0] |

Let

$\Delta_t$ = delay (in seconds).

$\Delta_c$ = delay (in cycles of clock signal at input to $D_X$).

$T_X$ = period of the clock signal at the input of the divider, $D_X$ (in seconds).

$\Phi$ =

$16 \times SH[4] + 8 \times PO[3] + 4 \times PO[2] + 2 \times PO[1] + 1 \times PO[0]$

The channel divide-by is set as N = high cycles and M = low cycles.

**Case 1**

For $\Phi \leq 15$,

$\Delta_t = \Phi \times T_X$

$\Delta_c = \Delta_t / T_X = \Phi$

**Case 2**

For $\Phi \geq 16$,

$\Delta_t = (\Phi - 16 + M + 1) \times T_X$

$\Delta_c = \Delta_t / T_X$

By giving each divider a different phase offset, output-to-output delays can be set in increments of the channel divider input clock cycle. Figure 38 shows the results of setting such a coarse offset between outputs.

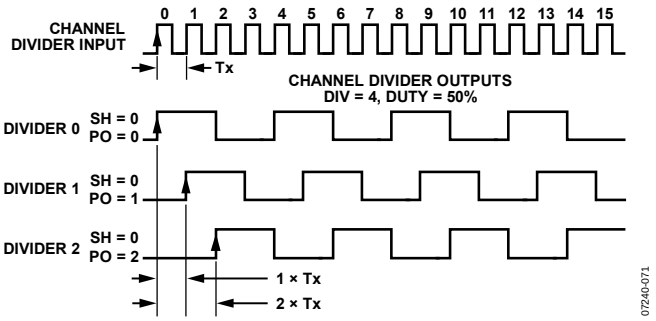


*Figure 38. Effect of Coarse Phase Offset (or Delay)*

### *Synchronizing the Outputs—$\overline{\text{SYNC}}$ Function*

The AD9522 clock outputs can be synchronized to each other. Outputs can be individually excluded from synchronization. Synchronization consists of setting the nonexcluded outputs to a preset set of static conditions. These conditions include the divider ratio and phase offsets for a given channel divider. This allows the user to specify different divide ratios and phase offsets for each of the four channel dividers. Releasing the $\overline{\text{SYNC}}$ pin allows the outputs to continue clocking with the preset conditions applied.

Synchronization of the outputs is executed in the following ways:

- The $\overline{\text{SYNC}}$ pin is forced low and then released (manual sync).
- By setting and then resetting any one of the following three bits: the soft SYNC bit (Register 0x230[0]), the soft reset bit (0x000[5] [mirrored]), and the power-down distribution reference bit (Register 0x230[1]).
- Synchronization of the outputs can be executed as part of the chip power-up sequence.
- The $\overline{\text{RESET}}$ pin is forced low and then released (chip reset).
- The $\overline{\text{PD}}$ pin is forced low and then released (chip power-down).

The most common way to execute the SYNC function is to use the $\overline{\text{SYNC}}$ pin to perform a manual synchronization of the outputs. This requires a low going signal on the $\overline{\text{SYNC}}$ pin, which is held low and then released when synchronization is desired. The timing of the SYNC operation is shown in Figure 39 (using the VCO divider) and in Figure 40 (the VCO divider is not used). There is an uncertainty of up to one cycle of the clock at the input to the channel divider due to the asynchronous nature of the SYNC signal with respect to the clock edges inside the AD9522.

The pipeline delay from the $\overline{\text{SYNC}}$ rising edge to the beginning of the synchronized output clocking is between 14 cycles and 15 cycles of clock at the channel divider input, plus one cycle of the VCO divider input (see Figure 39) or one cycle of the channel divider input (see Figure 40), depending on whether the VCO divider is used. Cycles are counted from the rising edge of the signal. In addition, there is an additional 1.2 ns (typical) delay from the SYNC signal to the internal synchronization logic, as well as the propagation delay of the output driver. The driver propagation delay is approximately 100 ps for the LVDS driver and approximately 1.5 ns for the CMOS driver.

Another common way to execute the SYNC function is by setting and resetting the soft SYNC bit at Register 0x230[0]. Both setting and resetting of the soft SYNC bit require an update all registers (Register 0x232[0] = 1b) operation to take effect.

A SYNC operation brings all outputs that are not excluded (by the ignore SYNC bit) to a preset condition before allowing the outputs to begin clocking in synchronicity. The preset condition takes into account the settings in each of the channel's start high bit and its phase offset. These settings govern both the static state of each output when the SYNC operation is happening and the state and relative phase of the outputs when they begin clocking again upon completion of the SYNC operation. A SYNC operation must take place for the phase offsets setting to take effect.

The AD9522 differential LVDS outputs are four groups of three, sharing a channel divider per triplet. In the case of CMOS, each LVDS differential pair can be configured as two single-ended CMOS outputs. The synchronization conditions apply to all of the drivers that belong to that channel divider.

Each channel (a divider and its outputs) can be excluded from any SYNC operation by setting the ignore SYNC bit of the channel. Channels that are set to ignore SYNC (excluded channels) do not set their outputs static during a SYNC operation, and their outputs are not synchronized with those of the included channels.

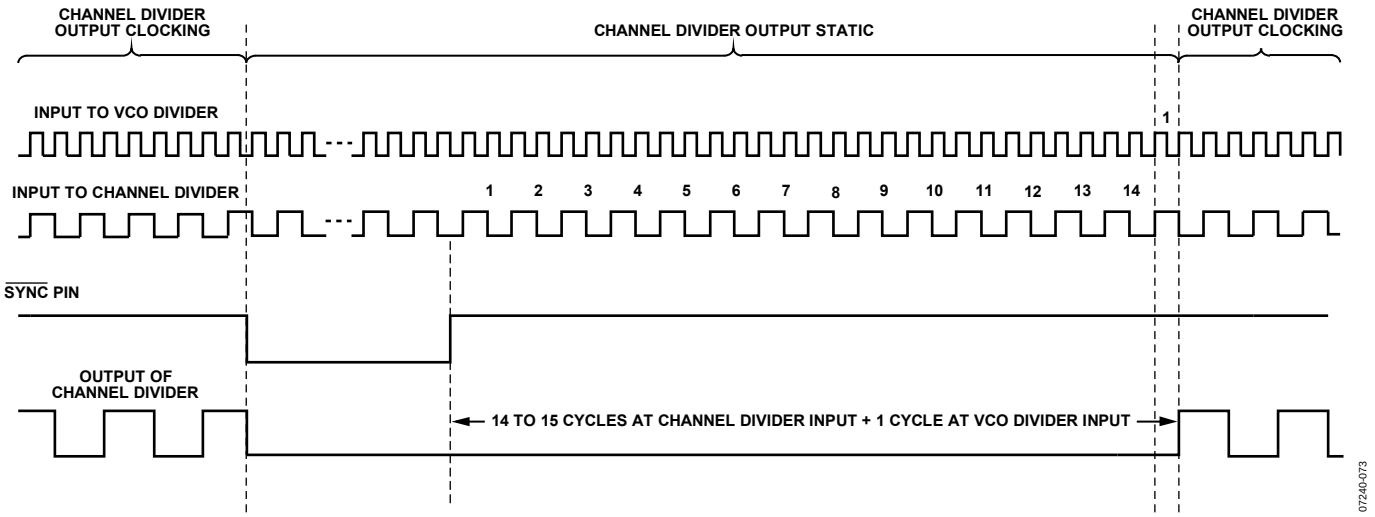


*Figure 39. SYNC Timing Pipeline Delay When the VCO Divider Is Used—CLK or VCO Is Input*

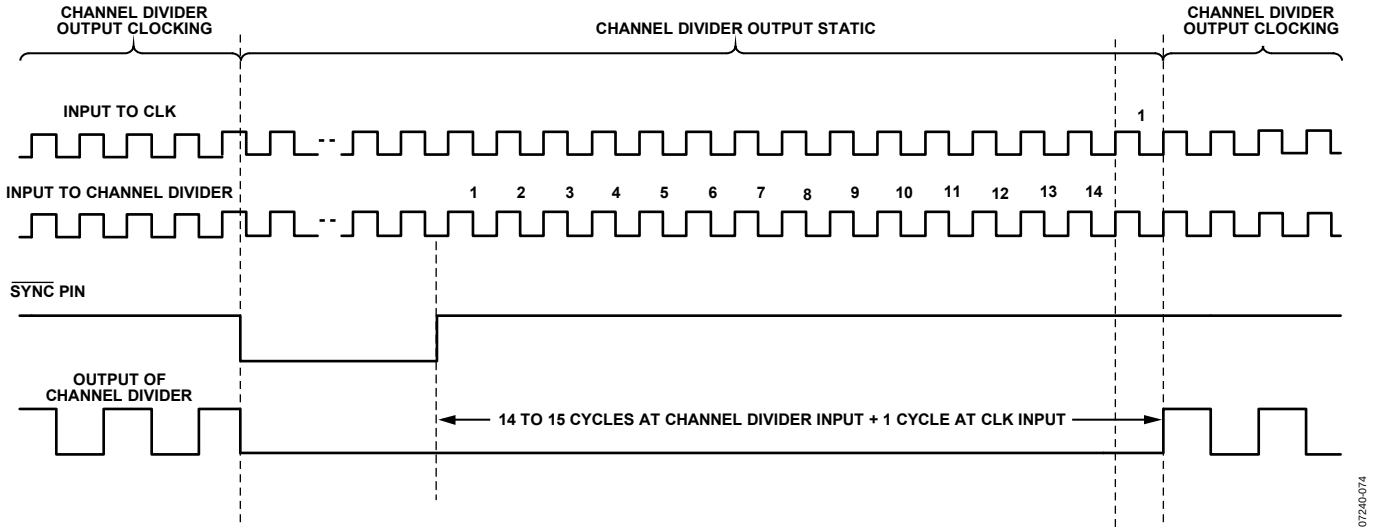


*Figure 40. SYNC Timing Pipeline Delay When the VCO Divider Is Not Used*

### *LVDS Output Drivers*

The AD9522 output drivers can be configured as either an LVDS differential output or as a pair of CMOS single-ended outputs. The LVDS outputs allow for selectable output current from ~1.75 mA to ~7 mA.

The LVDS output polarity can be set as noninverting or inverting, which allows for the adjustment of the relative polarity of outputs within an application without requiring a board layout change. Each LVDS output can be individually powered down to save power.

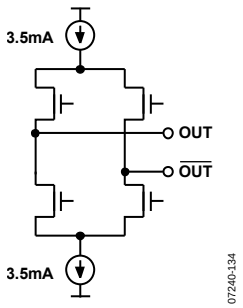


*Figure 41. LVDS Output Simplified Equivalent Circuit with 3.5 mA Typical Current Source*

### CMOS Output Drivers

The user can also individually configure each LVDS output as a pair of CMOS outputs, which provides up to 24 CMOS outputs. When an output is configured as CMOS, CMOS Output A and CMOS Output B are automatically turned on. For a given differential pair, either CMOS Output A or Output B can be turned on or off independently. The user can also select the relative polarity of the CMOS outputs for any combination of inverting and noninverting (see Register 0x0F0 to Register 0x0FB).

The user can power down each CMOS output as needed to save power. The CMOS output power-down is individually controlled by the enable CMOS output register (0x0F0[6:5] to 0x0FB[6:5]). The CMOS driver is in tristate when it is powered down.

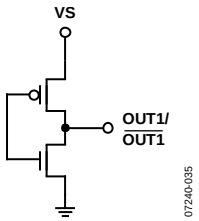


*Figure 42. CMOS Equivalent Output Circuit*

## RESET MODES

The AD9522 has a power-on reset (POR) and several other ways to apply a reset condition to the chip.

### Power-On Reset

During chip power-up, a power-on reset pulse is issued when VS reaches ~2.6 V (<2.8 V) and restores the chip either to the setting stored in EEPROM (with the EEPROM pin = 1) or to the on-chip setting (with the EEPROM pin = 0). At power-on, the AD9522 also executes a SYNC operation, ~50 ms after the supply reaches ~2.4 V, which brings the outputs into phase alignment according to the default settings. It takes ~70 ms for the outputs to begin toggling after the power-on reset pulse signal is internally generated.

### Hardware Reset via the $\overline{\text{RESET}}$ Pin

$\overline{\text{RESET}}$, a hard reset (an asynchronous hard reset is executed by briefly pulling $\overline{\text{RESET}}$ low), restores the chip either to the setting stored in EEPROM (the EEPROM pin = 1b) or to the on-chip setting (the EEPROM pin = 0b). A hard reset also executes a SYNC operation, bringing the outputs into phase alignment according to the default settings. When EEPROM is inactive (the EEPROM pin = 0b), it takes ~2 µs for the outputs to begin toggling after $\overline{\text{RESET}}$ is issued. When EEPROM is active (the EEPROM pin = 1b), it takes ~20 ms for the outputs to toggle after $\overline{\text{RESET}}$ is brought high.

### Soft Reset via the Serial Port

The serial port control register allows for a soft reset by setting Bit 2 and Bit 5 in Register 0x000. The function of this register is determined by the state of the EEPROM pin.

When Bit 2 and Bit 5 are set and the EEPROM pin is high, the chip is restored to the settings saved in the EEPROM. When Bit 2 and Bit 5 are set and the EEPROM pin is low, the chip is restored to the on-chip defaults.

Except for the self-clearing bits, Bit 2 and Bit 5, Register 0x000 retains its previous value prior to reset. During the internal reset, the outputs hold static. However, the self-clearing operation does not complete until an additional serial port SCLK cycle occurs, and the AD9520 is held in reset until that happens.

### Soft Reset to Settings in EEPROM when EEPROM Pin = 0 via the Serial Port

The serial port control register allows the chip to be reset to settings in EEPROM when the EEPROM pin = 1 via Register 0xB02[1]. This bit is self-clearing. This bit does not have any effect when the EEPROM pin = 0. It takes ~20 ms for the outputs to begin toggling after the SOFT_EEPROM register is cleared.

## POWER-DOWN MODES

### Chip Power-Down via $\overline{\text{PD}}$

The AD9522 can be put into a power-down condition by pulling the $\overline{\text{PD}}$ pin low. Power-down turns off most of the functions and currents inside the AD9522. The chip remains in this power-down state until $\overline{\text{PD}}$ is brought back to logic high. When taken out of power-down mode, the AD9522 returns to the settings programmed into its registers prior to the power-down, unless the registers are changed by new programming while the $\overline{\text{PD}}$ pin is held low.

Powering down the chip shuts down the currents on the chip. Because this is not a complete power-down, it can be called sleep mode. The AD9522 contains special circuitry to prevent runt pulses on the outputs when the chip is entering or exiting sleep mode.

When the AD9522 is in a $\overline{\text{PD}}$ power-down, the chip is in the following state:

- The PLL is off (asynchronous power-down).
- The CLK input buffer is off, but the CLK input dc bias circuit is on.
- In differential mode, the reference input buffer is off, but the dc bias circuit is still on.
- In singled-ended mode, the reference input buffer is off, and the dc bias circuit is off.
- All dividers are off.
- All CMOS outputs are tristated.
- All LVDS outputs are in power-down (high impedance) mode.
- The serial control port is active, and the chip responds to commands.

### PLL Power-Down

The PLL section of the AD9522 can be selectively powered down. There are two PLL power-down modes set by Register 0x010[1:0]: asynchronous and synchronous.

In asynchronous power-down mode, the device powers down as soon as the registers are updated. In synchronous power-down mode, the PLL power-down is gated by the charge pump to prevent unwanted frequency jumps. The device goes into power-down on the occurrence of the next charge pump event after the registers are updated.

### Distribution Power-Down

The distribution section can be powered down by writing Register 0x230[1] = 1b, which turns off the bias to the distribution section. If the LVDS power-down mode is normal operation (0b), it is possible for a low impedance load on that LVDS output to draw significant current during this power-down. If the LVDS power-down mode is set to 1b, the LVDS output is not protected from reverse bias and can be damaged under certain termination conditions.

### Individual Clock Output Power-Down

Any of the clock distribution outputs can be put into power-down mode by individually writing to the appropriate registers. The register map details the individual power-down settings for each output. These settings are found in Register 0x0F0[0] to Register 0x0FB[0].

### Individual Clock Channel Power-Down

Any of the clock distribution channels can be powered down individually by writing to the appropriate registers. Powering down a clock channel is similar to powering down an individual driver, but it saves more power because the dividers are also powered down. Powering down a clock channel also automatically powers down the drivers connected to it. The register map details the individual power-down settings for each output channel. These settings are found in Register 0x192[2], Register 0x195[2], Register 0x198[2], and Register 0x19B[2].

# SERIAL CONTROL PORT

The AD9522 serial control port is a flexible, synchronous serial communications port that allows an easy interface with many industry-standard microcontrollers and microprocessors. The AD9522 serial control port is compatible with most synchronous transfer formats, including Philips I²C, Motorola® SPI®, and Intel® SSR protocols. The AD9522 I²C implementation deviates from the classic I²C specification on two specifications; these deviations are documented in Table 11 of this data sheet. The serial control port allows read/write access to all registers that configure the AD9522.

## SPI/I²C PORT SELECTION

The AD9522 has two serial interfaces, SPI and I²C. Users can select either SPI or I²C depending on the states of the three logic level (high, open, low) input pins, SP1 and SP0. When both SP1 and SP0 are high, the SPI interface is active. Otherwise, I²C is active with eight different I²C slave address (seven bits wide) settings, see Table 34. The four MSBs of the slave address are hardware coded as 1011, and the three LSBs are programmed by SP1 and SP0.

**Table 34. Serial Port Mode Selection**

| SP1 | SP0 | Address |
|---|---|---|
| Low | Low | I²C, 1011000 |
| Low | Open | I²C, 1011001 |
| Low | High | I²C, 1011010 |
| Open | Low | I²C, 1011011 |
| Open | Open | I²C, 1011100 |
| Open | High | I²C, 1011101 |
| High | Low | I²C, 1011110 |
| High | Open | I²C, 1011111 |
| High | High | SPI |

## I²C SERIAL PORT OPERATION

The AD9522 I²C port is based on the I²C fast mode standard. The AD9522 supports both I²C protocols: standard mode (100 kHz) and fast mode (400 kHz).

The AD9522 I²C port has a 2-wire interface consisting of a serial data line (SDA) and a serial clock line (SCL). In an I²C bus system, the AD9522 is connected to the serial bus (data bus SDA and clock bus SCL) as a slave device, meaning that no clock is generated by the AD9522. The AD9522 uses direct 16-bit (two bytes) memory addressing instead of traditional 8-bit (one byte) memory addressing.

### *I²C Bus Characteristics*

**Table 35. I²C Bus Definitions**

| Abbreviation | Definition |
|---|---|
| S | Start |
| Sr | Repeated start |
| P | Stop |
| A | Acknowledge |
| $\overline{A}$ | No acknowledge |
| W | Write |
| R | Read |

One pulse on the SCL clock line is generated for each data bit transferred.

The data on the SDA line must not change during the high period of the clock. The state of the data line can change only when the clock on the SCL line is low.

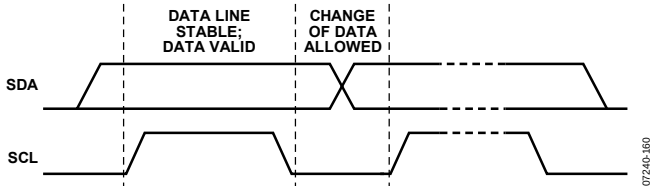


*Figure 43. Valid Bit Transfer*

A start condition is a transition from high to low on the SDA line while SCL is high. The start condition is always generated by the master to initialize the data transfer.

A stop condition is a transition from low to high on the SDA line while SCL is high. The stop condition is always generated by the master to end the data transfer.

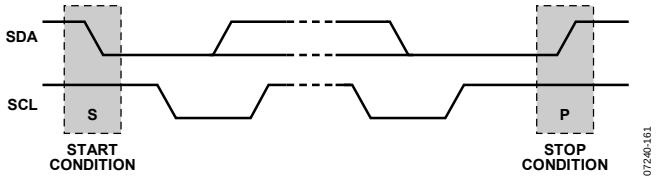


*Figure 44. Start and Stop Conditions*

A byte on the SDA line is always eight bits long. An acknowledge bit must follow every byte. Bytes are sent MSB first.

The acknowledge bit is the ninth bit attached to any 8-bit data byte. An acknowledge bit is always generated by the receiving device (receiver) to inform the transmitter that the byte has been received. It is accomplished by pulling the SDA line low during the ninth clock pulse after each 8-bit data byte.

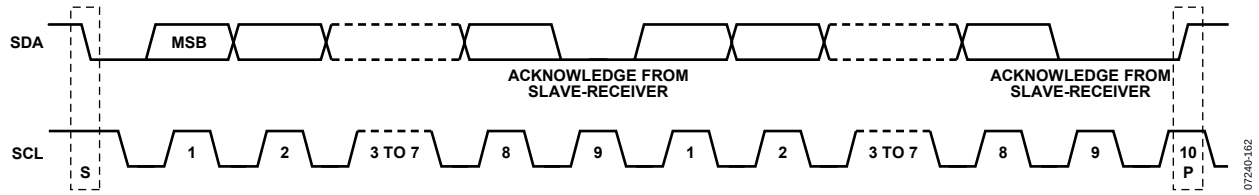


*Figure 45. Acknowledge Bit*

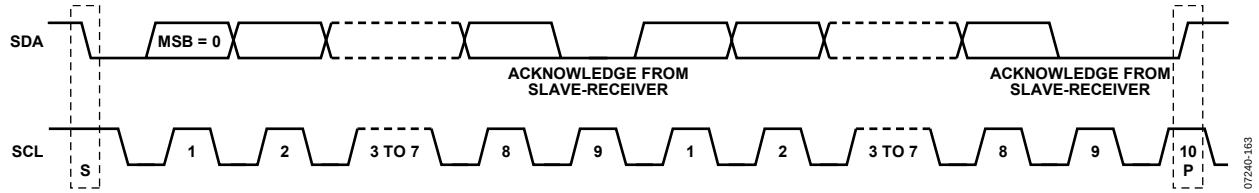


*Figure 46. Data Transfer Process (Master Write Mode, 2-Byte Transfer Used for Illustration)*

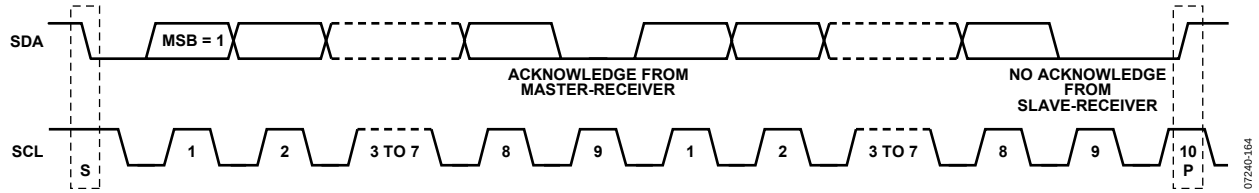


*Figure 47. Data Transfer Process (Master Read Mode, 2-Byte Transfer Used for Illustration)*

The no acknowledge bit is the ninth bit attached to any 8-bit data byte. A no acknowledge bit is always generated by the receiving device (receiver) to inform the transmitter that the byte has not been received. It is done by leaving the SDA line high during the ninth clock pulse after each 8-bit data byte.

### *Data Transfer Process*

The master initiates data transfer by asserting a start condition. This indicates that a data stream follows. All I²C slave devices connected to the serial bus respond to the start condition.

The master then sends an 8-bit address byte over the SDA line, consisting of a 7-bit slave address (MSB first) plus an R/$\overline{W}$ bit. This bit determines the direction of the data transfer, that is, whether data is written to or read from the slave device (0 = write, 1 = read).

The peripheral whose address corresponds to the transmitted address responds by sending an acknowledge bit. All other devices on the bus remain idle while the selected device waits for data to be read from or written to it. If the R/$\overline{W}$ bit is 0, the master (transmitter) writes to the slave device (receiver). If the R/$\overline{W}$ bit is 1, the master (receiver) reads from the slave device (transmitter).

The format for these commands is described in the Data Transfer Format section.

Data is then sent over the serial bus in the format of nine clock pulses, one data byte (8-bit) from either master (write mode) or slave (read mode) followed by an acknowledge bit from the receiving device. The number of bytes that can be transmitted per transfer is unrestricted. In write mode, the first two data bytes immediately after the slave address byte are the internal memory (control registers) address bytes with the high address byte first. This addressing scheme gives a memory address up to $2^{16} - 1 = 65{,}535$. The data bytes after these two memory address bytes are register data written into the control registers. In read mode, the data bytes after the slave address byte are register data read from the control registers.

When all data bytes are read or written, stop conditions are established. In write mode, the master (transmitter) asserts a stop condition to end data transfer during the (10th) clock pulse following the acknowledge bit for the last data byte from the slave device (receiver). In read mode, the master device (receiver) receives the last data byte from the slave device (transmitter) but does not pull it low during the ninth clock pulse. This is known as a no acknowledge bit. By receiving the no acknowledge bit, the slave device knows that the data transfer is finished and releases the SDA line. The master then takes the data line low during the low period before the 10th clock pulse and high during the 10th clock pulse to assert a stop condition.

A repeated start (Sr) condition can be used in place of a stop condition. Furthermore, a start or stop condition can occur at any time, and partially transferred bytes are discarded.

### *Data Transfer Format*

Send byte format—the send byte protocol is used to set up the register address for subsequent commands.

| S | Slave Address | W | A | RAM Address High Byte | A | RAM Address Low Byte | A | P |
|---|---|---|---|---|---|---|---|---|

Write byte format—the write byte protocol is used to write a register address to the RAM starting from the specified RAM address.

| S | Slave Address | W | A | RAM Address High Byte | A | RAM Address Low Byte | A | RAM Data 0 | A | RAM Data 1 | A | RAM Data 2 | A | P |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|

Receive byte format—the receive byte protocol is used to read the data byte(s) from RAM starting from the current address.

| S | Slave Address | R | A | RAM Data 0 | A | RAM Data 1 | A | RAM Data 2 | $\overline{A}$ | P |
|---|---|---|---|---|---|---|---|---|---|---|

Read byte format—the combined format of the send byte and the receive byte.

| S | Slave Address | W | A | RAM Address High Byte | A | RAM Address Low Byte | A | Sr | Slave Address | R | A | RAM Data 0 | A | RAM Data 1 | A | RAM Data 2 | $\overline{A}$ | P |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|

### *I²C Serial Port Timing*

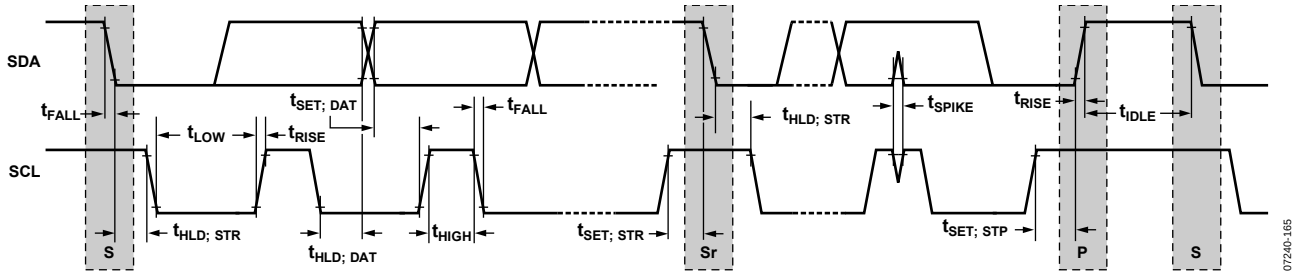


*Figure 48. I²C Serial Port Timing*

**Table 36. I²C Timing Definitions**

| Parameter | Description |
|---|---|
| $f_{I2C}$ | I²C clock frequency |
| $t_{IDLE}$ | Bus idle time between stop and start conditions |
| $t_{HLD;\ STR}$ | Hold time for repeated start condition |
| $t_{SET;\ STR}$ | Setup time for repeated start condition |
| $t_{SET;\ STP}$ | Setup time for stop condition |
| $t_{HLD;\ DAT}$ | Hold time for data |
| $t_{SET;\ DAT}$ | Setup time for data |
| $t_{LOW}$ | Duration of SCL clock low |
| $t_{HIGH}$ | Duration of SCL clock high |
| $t_{RISE}$ | SCL/SDA rise time |
| $t_{FALL}$ | SCL/SDA fall time |
| $t_{SPIKE}$ | Voltage spike pulse width that must be suppressed by the input filter |

## SPI SERIAL PORT OPERATION

### *Pin Descriptions*

SCLK (serial clock) is the serial shift clock. This pin is an input. SCLK is used to synchronize serial control port reads and writes. Write data bits are registered on the rising edge of this clock. The read data bits transition on the falling edge of SCLK. This pin is internally pulled down by a 30 kΩ resistor to ground.

SDIO (serial data input/output) is a dual-purpose pin and acts either as an input only (unidirectional mode) or as an input/output (bidirectional mode). The AD9522 defaults to the bidirectional I/O mode (Register 0x000[7] = 0b).

SDO (serial data out) is used only in the unidirectional I/O mode (Register 0x000[7] = 1b) as a separate output pin for reading back data.

$\overline{CS}$ (chip select bar) is an active low control that gates the read and write cycles. When $\overline{CS}$ is high, SDO and SDIO are in a high impedance state. This pin is internally pulled up by a 30 kΩ resistor to VS.

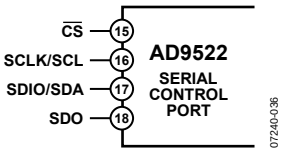


*Figure 49. Serial Control Port*

### *SPI Mode Operation*

In SPI mode, single or multiple byte transfers are supported, as well as MSB first or LSB first transfer formats. The AD9522 serial control port can be configured for a single bidirectional I/O pin (SDIO only) or for two unidirectional I/O pins (SDIO/SDO). By default, the AD9522 is in bidirectional mode. Short instruction mode (8-bit instructions) is not supported. Only long (16-bit) instruction mode is supported. It is possible that serial activity on the SDIO/SDO pins may induce jitter on the PLL while data is transmitted.

A write or a read operation to the AD9522 is initiated by pulling $\overline{CS}$ low.

The $\overline{CS}$ stalled high mode is supported in data transfers where three or fewer bytes of data (plus instruction data) are transferred (see Table 37). In this mode, the $\overline{CS}$ pin can temporarily return high on any byte boundary, allowing time for the system controller to process the next byte. $\overline{CS}$ can go high on byte boundaries only and can go high during either part (instruction or data) of the transfer.

During this period, the serial control port state machine enters a wait state until all data is sent. If the system controller decides to abort the transfer before all of the data is sent, the state machine must be reset by either completing the remaining transfers or by returning $\overline{CS}$ low for at least one complete SCLK cycle (but fewer than eight SCLK cycles). Raising the $\overline{CS}$ pin on a nonbyte boundary terminates the serial transfer and flushes the buffer.

In the streaming mode (see Table 37), any number of data bytes can be transferred in a continuous stream. The register address is automatically incremented or decremented (see the SPI MSB/LSB First Transfers section). $\overline{CS}$ must be raised at the end of the last byte to be transferred, thereby ending streaming mode.

### *Communication Cycle—Instruction Plus Data*

There are two parts to a communication cycle with the AD9522. The first part writes a 16-bit instruction word into the AD9522, coincident with the first 16 SCLK rising edges. The instruction word provides the AD9522 serial control port with information regarding the data transfer, which is the second part of the communication cycle. The instruction word defines whether the upcoming data transfer is a read or a write, the number of bytes in the data transfer, and the starting register address for the first byte of the data transfer.

### *Write*

If the instruction word is for a write operation, the second part is the transfer of data into the serial control port buffer of the AD9522. Data bits are registered on the rising edge of SCLK.

The length of the transfer (one, two, or three bytes, or streaming mode) is indicated by two bits (W1:W0) in the instruction byte. When the transfer is one, two, or three bytes, but not streaming, $\overline{CS}$ can be raised after each sequence of eight bits to stall the bus (except after the last byte, where it ends the cycle). When the bus is stalled, the serial transfer resumes when $\overline{CS}$ is lowered. Raising the $\overline{CS}$ pin on a nonbyte boundary resets the serial control port. During a write, streaming mode does not skip over reserved or blank registers, and the user can write 0x00 to the reserved register addresses.

Because data is written into a serial control port buffer area, not directly into the actual control registers of the AD9522, an additional operation is needed to transfer the serial control port buffer contents to the actual control registers of the AD9522, thereby causing them to become active. The update registers operation consists of setting Register 0x232[0] = 1b (this bit is self-clearing). Any number of bytes of data can be changed before executing an update registers operation. The update registers operation simultaneously actuates all register changes that have been written to the buffer since any previous update.

### *Read*

The AD9522 supports only the long instruction mode. If the instruction word is for a read operation, the next N × 8 SCLK cycles clock out the data from the address specified in the instruction word, where N is 1 to 3 as determined by W1:W0. If N = 4, the read operation is in streaming mode, continuing until $\overline{CS}$ is raised. Streaming mode does not skip over reserved or blank registers. The readback data is valid on the falling edge of SCLK.

The default mode of the AD9522 serial control port is the bidirectional mode. In bidirectional mode, both the sent data and the readback data appear on the SDIO pin. It is also possible to set the AD9522 to unidirectional mode (Register 0x000[7] = 1 and Register 0x000[0] = 1). In unidirectional mode, the readback data appears on the SDO pin.

A readback request reads the data that is in the serial control port buffer area or the data that is in the active registers (see Figure 50). Readback of the buffer or active registers is controlled by 0x004[0].

The AD9522 uses Register Address 0x000 to Register Address 0xB03.

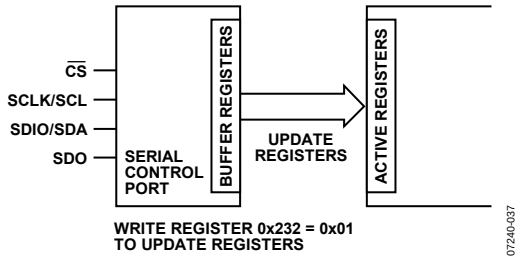


*Figure 50. Relationship Between Serial Control Port Buffer Registers and Active Registers of the AD9522*

## SPI INSTRUCTION WORD (16 BITS)

The MSB of the instruction word is R/$\overline{W}$, which indicates whether the instruction is a read or a write. The next two bits (W1:W0) indicate the length of the transfer in bytes. The final 13 bits are the address (A12:A0) at which to begin the read or write operation, see Table 39.

For a write, the instruction word is followed by the number of bytes of data indicated by Bits[W1:W0], see Table 37.

**Table 37. Byte Transfer Count**

| W1 | W0 | Bytes to Transfer |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 2 |
| 1 | 0 | 3 |
| 1 | 1 | Streaming mode |

Bits[A12:A0] select the address within the register map that is written to or read from during the data transfer portion of the communications cycle. Only Bits[A9:A0] are needed to cover the range of the 0x232 registers used by the AD9522. Bits[A12:A10] must always be 0b. For multibyte transfers, this address is the starting byte address. In MSB first mode, subsequent bytes increment the address.

## SPI MSB/LSB FIRST TRANSFERS

The AD9522 instruction word and byte data can be MSB first or LSB first. Any data written to 0x000 must be mirrored; the upper four bits ([7:4]) must mirror the lower four bits ([3:0]). This makes it irrelevant whether LSB first or MSB first is in effect. As an example of this mirroring, see the default setting for 0x000, which mirrors Bit 4 and Bit 3. This sets the long instruction mode, which is the default and the only mode supported.

The default for the AD9522 is MSB first.

When LSB first is set by 0x000[1] and 0x000[6], it takes effect immediately because it affects only the operation of the serial control port and does not require that an update be executed.

When MSB first mode is active, the instruction and data bytes must be written from MSB to LSB. Multibyte data transfers in MSB first format start with an instruction byte that includes the register address of the most significant data byte. Subsequent data bytes must follow in order from the high address to the low address. In MSB first mode, the serial control port internal address generator decrements for each data byte of the multibyte transfer cycle.

When LSB first is active, the instruction and data bytes must be written from LSB to MSB. Multibyte data transfers in LSB first format start with an instruction byte that includes the register address of the least significant data byte followed by multiple data bytes. In a multibyte transfer cycle, the internal byte address generator of the serial port increments for each byte.

The AD9522 serial control port register address decrements from the register address just written toward 0x000 for multibyte I/O operations if the MSB first mode is active (default). If the LSB first mode is active, the register address of the serial control port increments from the address just written toward 0x232 for multibyte I/O operations.

Streaming mode always terminates when it reaches 0x232. Note that unused addresses are not skipped during multibyte I/O operations.

**Table 38. Streaming Mode (No Addresses Are Skipped)**

| Write Mode | Address Direction | Stop Sequence |
|---|---|---|
| LSB first | Increment | 0x230, 0x231, 0x232, stop |
| MSB first | Decrement | 0x001, 0x000, 0x232, stop |

**Table 39. Serial Control Port, 16-Bit Instruction Word, MSB First**

| MSB | | | | | | | | | | | | | | | LSB |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| I15 | I14 | I13 | I12 | I11 | I10 | I9 | I8 | I7 | I6 | I5 | I4 | I3 | I2 | I1 | I0 |
| R/$\overline{W}$ | W1 | W0 | A12 = 0 | A11 = 0 | A10 = 0 | A9 | A8 | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 |

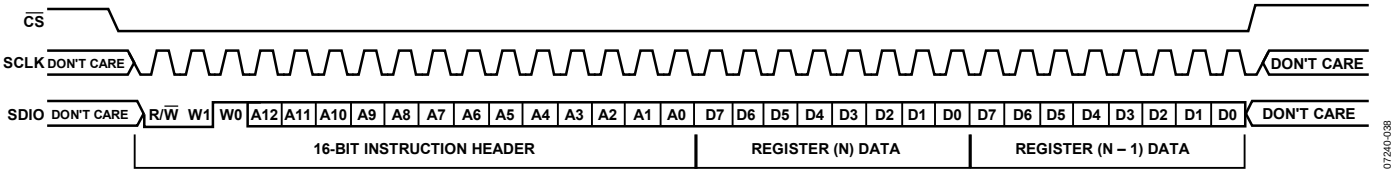


*Figure 51. Serial Control Port Write—MSB First, 16-Bit Instruction, Two Bytes of Data*

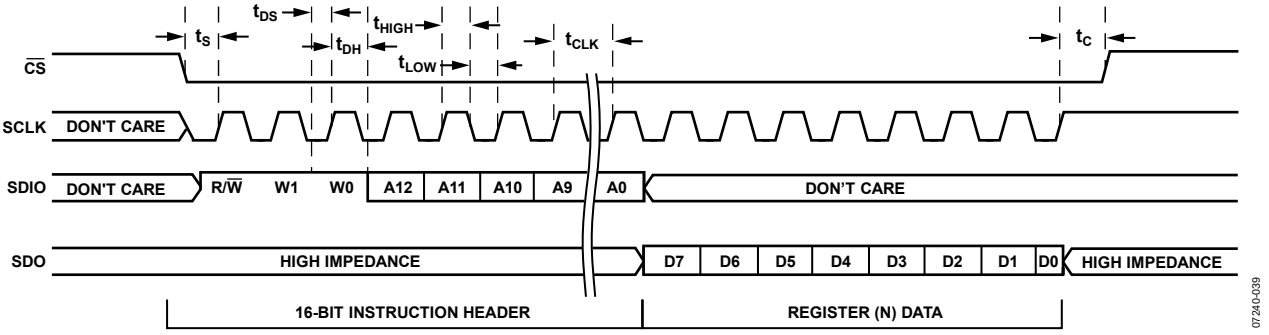


*Figure 52. Serial Control Port Read—MSB First, 16-Bit Instruction, Four Bytes of Data*

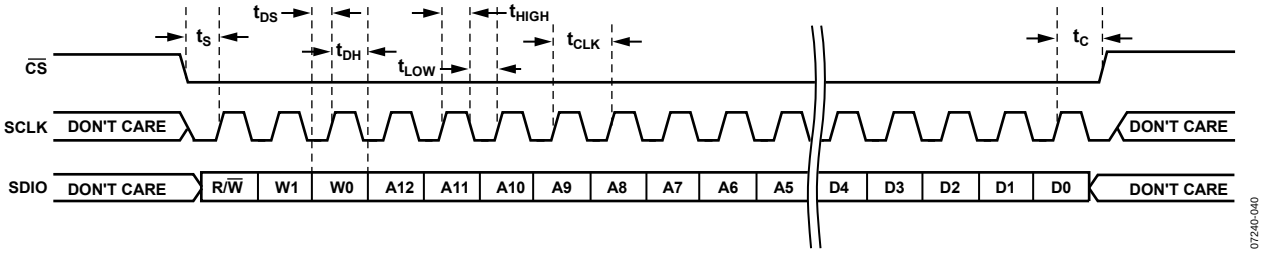


*Figure 53. Serial Control Port Read—MSB First, 16-Bit Instruction, Timing Measurements*

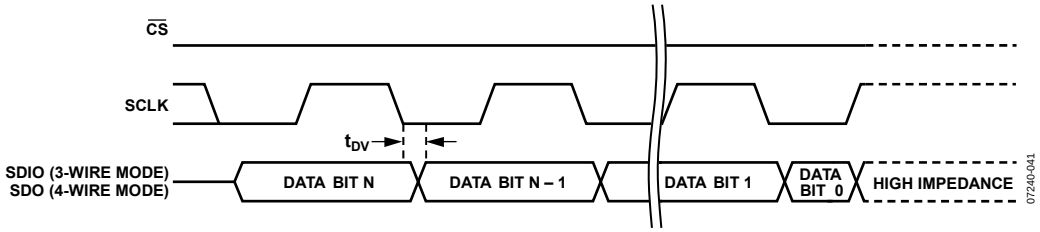


*Figure 54. Timing Diagram for Serial Control Port Register Read*

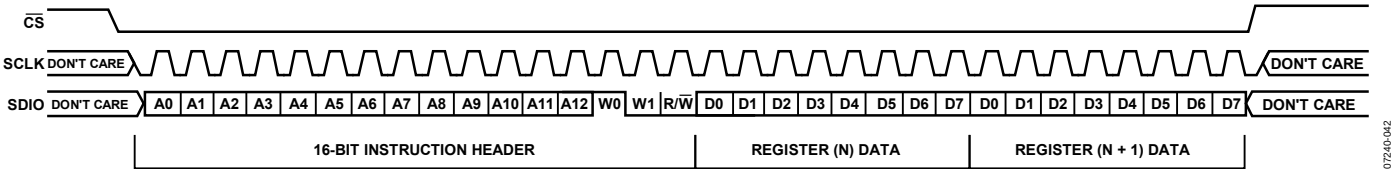


*Figure 55. Serial Control Port Write—LSB First, 16-Bit Instruction, Two Bytes of Data*

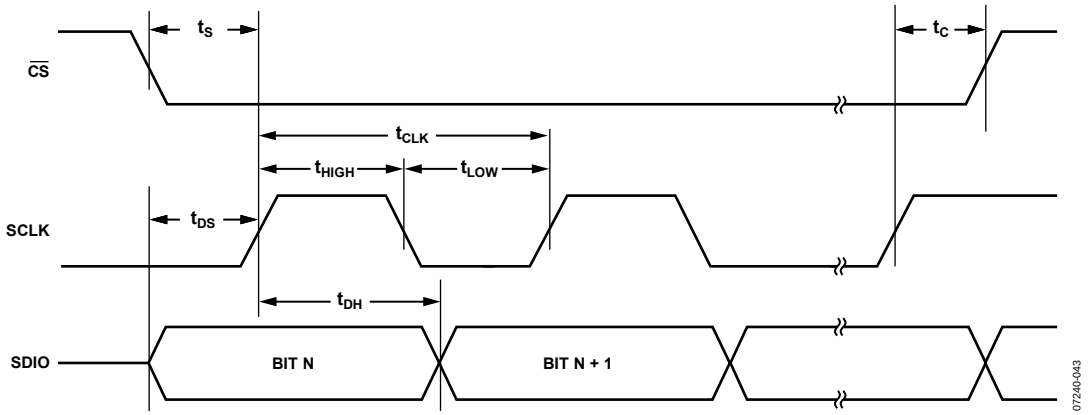


*Figure 56. Serial Control Port Timing—Write*

**Table 40. Serial Control Port Timing**

| Parameter | Description |
|---|---|
| $t_{DS}$ | Setup time between data and rising edge of SCLK |
| $t_{DH}$ | Hold time between data and rising edge of SCLK |
| $t_{CLK}$ | Period of the clock |
| $t_S$ | Setup time between the $\overline{CS}$ falling edge and SCLK rising edge (start of communication cycle) |
| $t_C$ | Setup time between SCLK rising edge and the $\overline{CS}$ rising edge (end of communication cycle) |
| $t_{HIGH}$ | Minimum period that SCLK should be in a logic high state |
| $t_{LOW}$ | Minimum period that SCLK should be in a logic low state |
| $t_{DV}$ | SCLK to valid SDIO and SDO (see Figure 54) |

# EEPROM OPERATIONS

The AD9522 contains an internal EEPROM (nonvolatile memory). The EEPROM can be programmed by users to create and store a user-defined register setting file when the power is off. This setting file can be used for power-up and chip reset as a default setting. The EEPROM size is 512 bytes.

To guarantee proper loading of the EEPROM during startup, a high-low-high pulse on the $\overline{\text{RESET}}$ pin must occur after the power supply stabilizes.

During the data transfer process, the write and read registers via the serial port are generally not available except for one readback register, STATUS_EEPROM.

To determine the data transfer state through the serial port in SPI mode, users can read the value of STATUS_EEPROM (1 = in process and 0 = completed).

In $I^2C$ mode, the user can address the AD9522 slave port with the external $I^2C$ master (send an address byte to the AD9522). If the AD9522 responds with a no acknowledge bit, the data transfer process is not done. If the AD9522 responds with an acknowledge bit, the data transfer process is completed. The user can monitor the STATUS_EEPROM register or program the STATUS pin to monitor the status of the data transfer.

## WRITING TO THE EEPROM

The EEPROM cannot be programmed directly through the serial port interface. To program the EEPROM and store a register setting file, do the following:

1. Program the AD9522 registers to the desired circuit state. If the user wants the PLL to lock automatically after power-up, the VCO calibration now bit (Register 0x018[0]) must be set to 1. This allows VCO calibration to start automatically after register loading. Note that a valid input reference signal must be present during VCO calibration.
2. Program the EEPROM buffer registers, if necessary (see the Programming the EEPROM Buffer Segment section). This is only necessary if users want to use the EEPROM to control the default setting of some (but not all) of the AD9522 registers, or if they want to control the register setting update sequence during power-up or chip reset.
3. Set the enable EEPROM write bit (Register 0xB02[0]) to 1 to enable the EEPROM.
4. Set the REG2EEPROM bit (Register 0xB03[0]) to 1.
5. Set the IO_UPDATE bit (Register 0x232[0]) to 1, which starts the process of writing data into the EEPROM to create the EEPROM setting file. This enables the AD9522 EEPROM controller to transfer the current register values, as well as the memory address and instruction bytes from the EEPROM buffer segment, into the EEPROM. After the write process is completed, the internal controller sets Register 0xB03[0] (REG2EEPROM) back to 0.

   The readback register, STATUS_EEPROM (0xB00[0]), is used to indicate the data transfer status between the EEPROM and the control registers (0 = done/inactive; 1 = in process/active). At the beginning of the data transfer, STATUS_EEPROM is set to 1 by the EEPROM controller and cleared to 0 at the end of the data transfer. The user can access STATUS_EEPROM through the STATUS pin when the STATUS pin is programmed to monitor STATUS_EEPROM. Alternatively, the user can monitor the STATUS_EEPROM bit.
6. After the data transfer process is done (Register 0xB00[0] = 0), set the enable EEPROM write register (Register 0xB02[0]) to 0 to disable writing to the EEPROM.

To verify that the data transfer has completed correctly, the user can verify that 0xB01[0] = 0. A value of 1 in this register indicates a data transfer error.

## READING FROM THE EEPROM

The following reset-related events can start the process of restoring the settings stored in EEPROM to the control registers.

When the EEPROM pin is set high, do any of the following:

- Power up the AD9522.
- Perform a hardware chip reset by pulling the $\overline{\text{RESET}}$ pin low and then releasing $\overline{\text{RESET}}$.
- Set the self-clearing soft reset bit (Register 0x000[5]) to 1.

When the EEPROM pin is set low, set the self-clearing SOFT_EEPROM bit (Register 0xB02[1]) to 1. The AD9522 then starts to read the EEPROM and loads the values into the AD9522.

If the EEPROM pin is low during reset or power-up, the EEPROM is not active, and the AD9522 default values are loaded instead.

When using the EEPROM to automatically load the AD9522 register values and lock the PLL, the VCO calibration now bit (Register 0x018[0]) must be set to 1 when the register values are written to the EEPROM. This allows VCO calibration to start automatically after register loading. A valid input reference signal must be present during VCO calibration.

To verify that the data transfer has completed correctly, the user can verify that Register 0xB01[0] = 0. A value of 1 in this register indicates a data transfer error.

## PROGRAMMING THE EEPROM BUFFER SEGMENT

The EEPROM buffer segment is a register space on the AD9522 that allows the user to specify which groups of registers are stored to the EEPROM during EEPROM programming. Normally, this segment does not need to be programmed by the user. Instead, the default power-up values for the EEPROM buffer segment allow the user to store all of the AD9522 register values from Register 0x000 to Register 0x231 to the EEPROM.

For example, if users want to load only the output driver settings from the EEPROM without disturbing the PLL register settings currently stored in the AD9522, they can alter the EEPROM buffer segment to include only the registers that apply to the output drivers and exclude the registers that apply to the PLL configuration.

There are two parts to the EEPROM buffer segment: register section definition groups and operational codes. Each register section definition group contains the starting address and number of bytes to be written to the EEPROM.

If the AD9522 register map were continuous from Address 0x000 to Address 0x232, only one register section definition group would consist of a starting address of 0x000 and a length of 563 bytes. However, this is not the case. The AD9522 register map is noncontiguous, and the EEPROM is only 512 bytes long. Therefore, the register section definition group tells the EEPROM controller how the AD9522 register map is segmented.

There are three operational codes: IO_UPDATE, end-of-data, and pseudo-end-of-data. It is important that the EEPROM buffer segment always have either an end-of-data or a pseudo-end-of-data operational code and that an IO_UPDATE operation code appear at least once before the end-of-data op code.

### *Register Section Definition Group*

The register section definition group is used to define a continuous register section for the EEPROM profile. It consists of three bytes. The first byte defines how many continuous register bytes are in this group. If the user puts 0x000 in the first byte, it means there is only one byte in this group. If the user puts 0x001, it means there are two bytes in this group. The maximum number of registers in one group is 128.

The next two bytes are the low byte and high byte of the memory address (16-bit) of the first register in this group.

### *IO_UPDATE (Operational Code 0x80)*

The EEPROM controller uses Operational Code 0x80 to generate an IO_UPDATE signal to update the active control register bank from the buffer register bank during the download process.

At a minimum, there must be at least one IO_UPDATE operational code after the end of the final register section definition group. This is needed so that at least one IO_UPDATE occurs after all of the AD9522 registers are loaded when the EEPROM is read. If this operational code is absent during a write to the EEPROM, the register values loaded from the EEPROM are not transferred to the active register space, and these values do not take effect after they are loaded from the EEPROM to the AD9522.

### *End-of-Data (Operational Code 0xFF)*

The EEPROM controller uses Operational Code 0xFF to terminate the data transfer process between EEPROM and the control register during the upload and download process. The last item appearing in the EEPROM buffer segment must be either this operational code or the pseudo-end-of-data operational code.

### *Pseudo-End-of-Data (Operational Code 0xFE)*

The AD9522 EEPROM buffer segment has 23 bytes that can contain up to seven register section definition groups. If users want to define more than seven register section definition groups, the pseudo-end-of-data operational code (0xFE) can be used. During the upload process, when the EEPROM controller receives the pseudo-end-of-data operational code, it halts the data transfer process, clears the REG2EEPROM bit, and enables the AD9522 serial port. Users can then program the EEPROM buffer segment again and reinitiate the data transfer process by setting the REG2EEPROM bit (0xB03[0]) to 1 and the IO_UPDATE bit (0x232[0]) to 1. The internal $I^2C$ master then begins writing to the EEPROM starting from the EEPROM address held from the last writing.

This sequence enables more discrete instructions to be written to the EEPROM than would otherwise be possible due to the limited size of the EEPROM buffer segment. It also permits the user to write to the same register multiple times with a different value each time.

**Table 41. Example of EEPROM Buffer Segment**

| Reg Addr (Hex) | Bit 7 (MSB) | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 (LSB) |
|---|---|---|---|---|---|---|---|---|
| **Start EEPROM Buffer Segment** | | | | | | | | |
| 0xA00 | 0 | Number of bytes [6:0] of the first group of registers | | | | | | |
| 0xA01 | Address [15:8] of the first group of registers | | | | | | | |
| 0xA02 | Address [7:0] of the first group of registers | | | | | | | |
| 0xA03 | 0 | Number of bytes [6:0] of the second group of registers | | | | | | |
| 0xA04 | Address [15:8] of the second group of registers | | | | | | | |
| 0xA05 | Address [7:0] of the second group of registers | | | | | | | |
| 0xA06 | 0 | Number of bytes [6:0] of the third group of registers | | | | | | |
| 0xA07 | Address [15:8] of the third group of registers | | | | | | | |
| 0xA08 | Address [7:0] of the third group of registers | | | | | | | |
| 0xA09 | IO_UPDATE operational code (0x80) | | | | | | | |
| 0xA0A | End-of-data operational code (0xFF) | | | | | | | |

# THERMAL PERFORMANCE

**Table 42. Thermal Parameters for the 64-Lead LFCSP**

| Symbol | Thermal Characteristic Using a JEDEC JESD51-7 Plus JEDEC JESD51-5 2S2P Test Board | Value (°C/W) |
|---|---|---|
| $\theta_{JA}$ | Junction-to-ambient thermal resistance, 0.0 m/sec airflow per JEDEC JESD51-2 (still air) | 22.0 |
| $\theta_{JMA}$ | Junction-to-ambient thermal resistance, 1.0 m/sec airflow per JEDEC JESD51-6 (moving air) | 19.2 |
| $\theta_{JMA}$ | Junction-to-ambient thermal resistance, 2.0 m/sec airflow per JEDEC JESD51-6 (moving air) | 17.2 |
| $\Psi_{JB}$ | Junction-to-board characterization parameter, 1.0 m/sec airflow per JEDEC JESD51-6 (moving air) and JEDEC JESD51-8 | 11.6 |
| $\theta_{JC}$ | Junction-to-case thermal resistance (die-to-heat sink) per MIL-STD-883, Method 1012.1 | 1.3 |
| $\Psi_{JT}$ | Junction-to-top-of-package characterization parameter, 0 m/sec airflow per JEDEC JESD51-2 (still air) | 0.1 |

The AD9522 is specified for a case temperature ($T_{CASE}$). To ensure that $T_{CASE}$ is not exceeded, an airflow source can be used.

Use the following equation to determine the junction temperature on the application PCB:

$$T_J = T_{CASE} + (\Psi_{JT} \times PD)$$

where:
$T_J$ is the junction temperature (°C).
$T_{CASE}$ is the case temperature (°C) measured by the user at the top center of the package.
$\Psi_{JT}$ is the value from Table 42.

*PD* is the power dissipation (see the total power dissipation in Table 15).

Values of $\theta_{JA}$ are provided for package comparison and PCB design considerations. $\theta_{JA}$ can be used for a first-order approximation of $T_J$ by the equation

$$T_J = T_A + (\theta_{JA} \times PD)$$

where $T_A$ is the ambient temperature (°C).

Values of $\theta_{JC}$ are provided for package comparison and PCB design considerations when an external heat sink is required.

Values of $\Psi_{JB}$ are provided for package comparison and PCB design considerations.

# REGISTER MAP

Register addresses that are not listed in Table 43 are not used, and writing to those registers has no effect. Writing to register addresses marked unused must have 00h written to them, unless otherwise noted. N/A is not applicable.

**Table 43. Register Map Overview**

| Addr (Hex) | Parameter | Bit 7 (MSB) | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 (LSB) | Default Value (Hex) |
|---|---|---|---|---|---|---|---|---|---|---|
| Serial Port Configuration | | | | | | | | | | |
| 000 | Serial port config (SPI mode) | SDO active | LSB first/ addr incr | Soft reset (self-clearing) | Unused | Unused | Soft reset (self-clearing) | LSB first/ addr incr | SDO active | 00 |
| | Serial port config ($I^2C$ mode) | Unused | | Soft reset (self-clearing) | Unused | Unused | Soft reset (self-clearing) | Unused | | 00 |
| 001 | | Unused | | | | | | | | N/A |
| 002 | Silicon Revision | Silicon Revision (read only) | | | | | | | | 11 |
| 003 | Part ID | Part ID (read only) | | | | | | | | E0 |
| 004 | Readback control | Unused | | | | | | | Readback active regs | 00 |
| EEPROM ID | | | | | | | | | | |
| 005 | EEPROM customer version ID | EEPROM customer version ID (LSB) | | | | | | | | 00 |
| 006 | | EEPROM customer version ID (MSB) | | | | | | | | 00 |
| 007 to 00F | | Unused | | | | | | | | 00 |
| PLL | | | | | | | | | | |
| 010 | PFD charge pump | PFD polarity | Charge pump current | | | Charge pump mode | | PLL power-down | | 7D |
| 011 | R counter | 14-bit R counter, Bits[7:0] (LSB) | | | | | | | | 01 |
| 012 | | Unused | | 14-bit R counter, Bits[13:8] (MSB) | | | | | | 00 |
| 013 | A counter | Unused | | 6-bit A counter | | | | | | 00 |
| 014 | B counter | 13-bit B counter, Bits[7:0] (LSB) | | | | | | | | 03 |
| 015 | | Unused | | | 13-bit B counter, Bits[12:8] (MSB) | | | | | 00 |
| 016 | PLL_CTRL_1 | Set CP pin to VCP/2 | Reset R counter | Reset A and B counters | Reset all counters | B counter bypass | Prescaler P | | | 06 |
| 017 | PLL_CTRL_2 | STATUS pin control | | | | | | Antibacklash pulse width | | 00 |
| 018 | PLL_CTRL_3 | Enable CMOS reference input dc offset | Lock detect counter | | Digital lock detect window | Disable digital lock detect | Unused | | | 06 |
| 019 | PLL_CTRL_4 | R, A, B counters $\overline{\text{SYNC}}$ pin reset | | R path delay | | | N path delay | | | 00 |
| 01A | PLL_CTRL_5 | Enable STATUS pin divider | Ref freq monitor threshold | LD pin control | | | | | | 00 |
| 01B | PLL_CTRL_6 | Enable CLK frequency monitor | Enable REF2 frequency monitor | Enable REF1 (REFIN) frequency monitor | REFMON pin control | | | | | 00 |
| 01C | PLL_CTRL_7 | Disable switchover deglitch | Select REF2 | Use REF_SEL pin | Reserved | | Enable REF2 | Enable REF1 | Enable differential reference | 00 |
| 01D | PLL_CTRL_8 | Enable STATUS_EEPROM at STATUS pin | Enable XTAL OSC | Enable clock doubler | Disable PLL status register | Enable LD pin comparator | Unused | Enable external holdover | Enable holdover | 80 |

| Addr (Hex) | Parameter | Bit 7 (MSB) | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 (LSB) | Default Value (Hex) |
|---|---|---|---|---|---|---|---|---|---|---|
| 01E | PLL_CTRL_9 | Unused | | | | | | Enable zero delay | Unused | 00 |
| 01F | PLL_READBACK (read-only) | Unused | | Holdover active | REF2 selected | CLK freq > threshold | REF2 freq > threshold | REF1 freq > threshold | Digital lock detect | N/A |
| Output Driver Control | | | | | | | | | | |
| 0F0 | OUT0 control | OUT0 format | OUT0 CMOS configuration | | OUT0 polarity | | OUT0 LVDS differential voltage | | OUT0 LVDS power-down | 62 |
| 0F1 | OUT1 control | OUT1 format | OUT1 CMOS configuration | | OUT1 polarity | | OUT1 LVDS differential voltage | | OUT1 LVDS power-down | 62 |
| 0F2 | OUT2 control | OUT2 format | OUT2 CMOS configuration | | OUT2 polarity | | OUT2 LVDS differential voltage | | OUT2 LVDS power-down | 62 |
| 0F3 | OUT3 control | OUT3 format | OUT3 CMOS configuration | | OUT3 polarity | | OUT3 LVDS differential voltage | | OUT3 LVDS power-down | 62 |
| 0F4 | OUT4 control | OUT4 format | OUT4 CMOS configuration | | OUT4 polarity | | OUT4 LVDS differential voltage | | OUT4 LVDS power-down | 62 |
| 0F5 | OUT5 control | OUT5 format | OUT5 CMOS configuration | | OUT5 polarity | | OUT5 LVDS differential voltage | | OUT5 LVDS power-down | 62 |
| 0F6 | OUT6 control | OUT6 format | OUT6 CMOS configuration | | OUT6 polarity | | OUT6 LVDS differential voltage | | OUT6 LVDS power-down | 62 |
| 0F7 | OUT7 control | OUT7 format | OUT7 CMOS configuration | | OUT7 polarity | | OUT7 LVDS differential voltage | | OUT7 LVDS power-down | 62 |
| 0F8 | OUT8 control | OUT8 format | OUT8 CMOS configuration | | OUT8 polarity | | OUT8 LVDS differential voltage | | OUT8 LVDS power-down | 62 |
| 0F9 | OUT9 control | OUT9 format | OUT9 CMOS configuration | | OUT9 polarity | | OUT9 LVDS differential voltage | | OUT9 LVDS power-down | 62 |
| 0FA | OUT10 control | OUT10 format | OUT10 CMOS configuration | | OUT10 polarity | | OUT10 LVDS differential voltage | | OUT10 LVDS power-down | 62 |
| 0FB | OUT11 control | OUT11 format | OUT11 CMOS configuration | | OUT11 polarity | | OUT11 LVDS differential voltage | | OUT11 LVDS power-down | 62 |
| 0FC | Enable output on CSDLD | CSDLD En OUT7 | CSDLD En OUT6 | CSDLD En OUT5 | CSDLD En OUT4 | CSDLD En OUT3 | CSDLD En OUT2 | CSDLD En OUT1 | CSDLD En OUT0 | 00 |
| 0FD | Enable output on CSDLD | Unused | Unused | Unused | Unused | CSDLD En OUT11 | CSDLD En OUT10 | CSDLD En OUT9 | CSDLD En OUT8 | 00 |
| 0FE to 18F | | Unused | | | | | | | | 00 |
| LVDS Channel Dividers | | | | | | | | | | |
| 190 | Divider 0 | Divider 0 low cycles | | | | Divider 0 high cycles | | | | 77 |
| 191 | | Divider 0 bypass | Divider 0 ignore SYNC | Divider 0 force high | Divider 0 start high | Divider 0 phase offset | | | | 00 |
| 192 | | Unused | | | | Unused | Channel 0 power-down | Reserved | Disable Divider 0 DCC | 00 |
| 193 | Divider 1 | Divider 1 low cycles | | | | Divider 1 high cycles | | | | 33 |
| 194 | | Divider 1 bypass | Divider 1 ignore SYNC | Divider 1 force high | Divider 1 start high | Divider 1 phase offset | | | | 00 |
| 195 | | Unused | | | | Unused | Channel 1 power-down | Reserved | Disable Divider 1 DCC | 00 |
| 196 | Divider 2 | Divider 2 low cycles | | | | Divider 2 high cycles | | | | 11 |
| 197 | | Divider 2 bypass | Divider 2 ignore SYNC | Divider 2 force high | Divider 2 start high | Divider 2 phase offset | | | | 00 |
| 198 | | Unused | | | | Unused | Channel 2 power-down | Reserved | Disable Divider 2 DCC | 00 |

| Addr (Hex) | Parameter | Bit 7 (MSB) | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 (LSB) | Default Value (Hex) |
|---|---|---|---|---|---|---|---|---|---|---|
| 199 | Divider 3 | Divider 3 low cycles | | | | Divider 3 high cycles | | | | 00 |
| 19A | | Divider 3 bypass | Divider 3 ignore SYNC | Divider 3 force high | Divider 3 start high | Divider 3 phase offset | | | | 00 |
| 19B | | Unused | | | | Unused | Channel 3 power-down | Reserved | Disable Divider 3 DCC | 00 |
| 19C to 1DF | | Unused | | | | | | | | 00 |
| VCO Divider and CLK Input | | | | | | | | | | |
| 1E0 | VCO divider | Unused | | | | | VCO divider | | | 00 |
| 1E1 | Input CLKs | Unused | | | Power-down clock input section | Reserved | | | Bypass VCO divider | 00 |
| 1E2 to 22A | | Unused | | | | | | | | 00 |
| System | | | | | | | | | | |
| 230 | Power-down and SYNC | Unused | | | | Disable power-on SYNC | Power-down SYNC | Power-down distribution reference | Soft SYNC | 00 |
| 231 | | Unused | | | | | | | | 00 |
| Update All Registers | | | | | | | | | | |
| 232 | IO_UPDATE | Unused | | | | | | | IO_UPDATE (self-clearing) | 00 |
| 233 to 9FF | | Unused | | | | | | | | 00 |
| EEPROM Buffer Segment | | | | | | | | | | |
| A00 | Serial port configuration | Data transfer: one byte | | | | | | | | 00 |
| A01 | | Starting address: Address 0x000 | | | | | | | | 00 |
| A02 | | | | | | | | | | 00 |
| A03 | EEPROM Customer Version ID | Data transfer: three bytes | | | | | | | | 02 |
| A04 | | Starting address: Address 0x004 | | | | | | | | 00 |
| A05 | | | | | | | | | | 04 |
| A06 | PLL settings | Data transfer: 16 bytes | | | | | | | | 0E |
| A07 | | Starting address: Address 0x010 | | | | | | | | 00 |
| A08 | | | | | | | | | | 10 |
| A09 | Output driver control | Data transfer: 16 bytes | | | | | | | | 0E |
| A0A | | Starting address: Address 0x0F0 | | | | | | | | 00 |
| A0B | | | | | | | | | | F0 |
| A0C | LVDS channel dividers | Data transfer: 12 bytes | | | | | | | | 0B |
| A0D | | Starting address: Address 0x190 | | | | | | | | 01 |
| A0E | | | | | | | | | | 90 |
| A0F | VDO divider and CLK input | Data transfer: two bytes | | | | | | | | 01 |
| A10 | | Starting address: Address 0x1E0 | | | | | | | | 01 |
| A11 | | | | | | | | | | E0 |
| A12 | Power-down and SYNC | Data transfer: two bytes | | | | | | | | 01 |
| A13 | | Starting address: Address 0x230 | | | | | | | | 02 |
| A14 | | | | | | | | | | 30 |
| A15 | I/O update | Action: IO_UPDATE | | | | | | | | 80 |
| A16 | End of data | Action: end of data | | | | | | | | FF |
| A17 to AFF | Unused | Unused (available for additional EEPROM instructions) | | | | | | | | 00 |

| Addr (Hex) | Parameter | Bit 7 (MSB) | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 (LSB) | Default Value (Hex) |
|---|---|---|---|---|---|---|---|---|---|---|
| EEPROM Control | | | | | | | | | | |
| B00 | EEPROM status (read only) | Unused | | | | | | Unused | STATUS_ EEPROM | 00 |
| B01 | EEPROM error checking (read only) | Unused | | | | | | Unused | EEPROM data error | 00 |
| B02 | EEPROM Control 1 | Unused | | | | | | SOFT_ EEPROM (self-clearing) | Enable EEPROM write | 00 |
| B03 | EEPROM Control 2 | Unused | | | | | | Unused | REG2EEPROM (self-clearing) | 00 |

# REGISTER MAP DESCRIPTIONS

Table 44 through Table 54 provide a detailed description of each of the control register functions. The registers are listed by hexadecimal address. Reference to a specific bit or range of bits within a register is indicated by squared brackets. For example, [3] refers to Bit 3 and [5:2] refers to the range of bits from Bit 5 through Bit 2.

**Table 44. SPI Mode Serial Port Configuration**

| Reg Addr (Hex) | Bit(s) | Name | Description |
|---|---|---|---|
| 000 | [7] | SDO active | Selects unidirectional or bidirectional data transfer mode.<br>[7] = 0; SDIO pin used for write and read; SDO is high impedance (default).<br>[7] = 1; SDO used for read; SDIO used for write; unidirectional mode. |
| 000 | [6] | LSB first/addr incr | SPI MSB or LSB data orientation. (This register is ignored in I²C mode.)<br>[6] = 0; data-oriented MSB first; addressing decrements (default).<br>[6] = 1; data-oriented LSB first; addressing increments. |
| 000 | [5] | Soft reset | Soft reset.<br>[5] = 1 (self-clearing). Soft reset; restores default values to internal registers. This bit self-clears on the next SCLK cycle after the completion of writing to this register. |
| 000 | [4] | Unused | |
| 000 | [3:0] | Mirror[7:4] | Bits[3:0] must always mirror Bits[7:4] so that it does not matter whether the part is in MSB or LSB first mode (see Register 0x000[6]). Set bits as follows:<br>[0] = [7].<br>[1] = [6].<br>[2] = [5].<br>[3] = [4]. |
| 002 | [7:0] | Silicon revision | This read only register identifies the revision level of the AD9522. |
| 003 | [7:0] | Part ID (read only) | Uniquely identifies the dash version (AD9522-0 through AD9522-5) of the AD9522.<br>AD9522-0: 0x20.<br>AD9522-1: 0x60.<br>AD9522-2: 0xA0.<br>AD9522-3: 0x61.<br>AD9522-4: 0xE1.<br>AD9522-5: 0xE0. |
| 004 | [0] | Readback active registers | Select register bank used for a readback.<br>[0] = 0; read back buffer registers (default).<br>[0] = 1; read back active registers. |

**Table 45. I²C Mode Serial Port Configuration**

| Reg Addr (Hex) | Bit(s) | Name | Description |
|---|---|---|---|
| 000 | [7:6] | Unused | |
| 000 | [5] | Soft reset | Soft reset.<br>[5] = 1 (self-clearing). Soft reset; restores default values to internal registers. This bit self-clears on the next SCL cycle after the completion of writing to this register. |
| 000 | [4] | Unused | |
| 000 | [3:0] | Mirror[7:4] | Bits[3:0] should always mirror Bits[7:4] so that it does not matter whether the part is in MSB or LSB first mode (see Register 0x000[6]). Set bits as follows:<br>[0] = [7].<br>[1] = [6].<br>[2] = [5].<br>[3] = [4]. |
| 002 | [7:0] | Silicon revision | This read only register identifies the revision level of the AD9522. |

| Reg Addr (Hex) | Bit(s) | Name | Description |
|---|---|---|---|
| 003 | [7:0] | Part ID (read only) | Uniquely identifies the dash version (AD9522-0 through AD9522-5) of the AD9522.<br>AD9522-0: 0x20<br>AD9522-1: 0x60<br>AD9522-2: 0xA0<br>AD9522-3: 0x61<br>AD9522-4: 0xE1<br>AD9522-5: 0xE0 |
| 004 | [0] | Readback active registers | Select register bank used for a readback.<br>[0] = 0; read back buffer registers (default).<br>[0] = 1; read back active registers. |

**Table 46. EEPROM ID**

| Reg Addr (Hex) | Bit(s) | Name | Description |
|---|---|---|---|
| 005 | [7:0] | EEPROM customer version ID (LSB) | 16-bit EEPROM ID[7:0]. This register, along with Register 0x006, allows the user to store a unique ID to identify which version of the AD9522 register settings is stored in the EEPROM. It does not affect AD9522 operation in any way (default: 0x00). |
| 006 | [7:0] | EEPROM customer version ID (MSB) | 16-bit EEPROM ID[15:8]. This register, along with Register 0x005, allows the user to store a unique ID to identify which version of the AD9522 register settings is stored in the EEPROM. It does not affect AD9522 operation in any way (default: 0x00). |

**Table 47. PLL**

| Reg. Addr (Hex) | Bit(s) | Name | Description | | | |
|---|---|---|---|---|---|---|
| 010 | [7] | PFD polarity | Sets the PFD polarity. Negative polarity is for use (if needed) with external VCO/VCXO only.<br>[7] = 0; positive (higher control voltage produces higher frequency) (default).<br>[7] = 1; negative (higher control voltage produces lower frequency). | | | |
| 010 | [6:4] | CP current | Charge pump current (with CPRSET = 5.1 kΩ). | | | |
| | | | **[6]** | **[5]** | **[4]** | **$I_{CP}$ (mA)** |
| | | | 0 | 0 | 0 | 0.6 |
| | | | 0 | 0 | 1 | 1.2 |
| | | | 0 | 1 | 0 | 1.8 |
| | | | 0 | 1 | 1 | 2.4 |
| | | | 1 | 0 | 0 | 3.0 |
| | | | 1 | 0 | 1 | 3.6 |
| | | | 1 | 1 | 0 | 4.2 |
| | | | 1 | 1 | 1 | 4.8 (default) |
| 010 | [3:2] | CP mode | Charge pump operating mode. | | | |
| | | | **[3]** | **[2]** | **Charge Pump Mode** | |
| | | | 0 | 0 | High impedance state. | |
| | | | 0 | 1 | Force source current (pump up). | |
| | | | 1 | 0 | Force sink current (pump down). | |
| | | | 1 | 1 | Normal operation (default). | |
| 010 | [1:0] | PLL power-down | PLL operating mode. | | | |
| | | | **[1]** | **[0]** | **Mode** | |
| | | | 0 | 0 | Normal operation; this mode must be selected to use the PLL. | |
| | | | 0 | 1 | Asynchronous power-down (default). | |
| | | | 1 | 0 | Unused. | |
| | | | 1 | 1 | Synchronous power-down. | |
| 011 | [7:0] | 14-bit R counter, Bits[7:0] (LSB) | Reference divider LSBs—lower eight bits. The reference divider (also called the R divider or R counter) is 14 bits long. The lower eight bits are in this register (default: 0x01). | | | |
| 012 | [5:0] | 14-bit R counter, Bits[13:8] (MSB) | Reference divider MSBs—upper six bits. The reference divider (also called the R divider or R counter) is 14 bits long. The upper six bits are in this register (default: 0x00). | | | |

| Reg. Addr (Hex) | Bit(s) | Name | Description |
|---|---|---|---|
| 013 | [5:0] | 6-bit A counter | A counter (part of N divider). The N divider is also called the feedback divider (default: 0x00). |
| 014 | [7:0] | 13-bit B counter, Bits[7:0] (LSB) | B counter (part of N divider)—lower eight bits. The N divider is also called the feedback divider (default: 0x03). |
| 015 | [4:0] | 13-bit B counter, Bits[12:8] (MSB) | B counter (part of N divider)—upper five bits. The N divider is also called the feedback divider (default: 0x00). |
| 016 | [7] | Set CP pin to VCP/2 | Sets the CP pin to one-half of the VCP supply voltage.<br>[7] = 0; CP normal operation (default).<br>[7] = 1; CP pin set to VCP/2. |
| 016 | [6] | Reset R counter | Reset R counter (R divider).<br>[6] = 0; normal (default).<br>[6] = 1; hold R counter in reset. |
| 016 | [5] | Reset A and B counters | Reset A and B counters (part of N divider).<br>[5] = 0; normal (default).<br>[5] = 1; hold A and B counters in reset. |
| 016 | [4] | Reset all counters | Reset R, A, and B counters.<br>[4] = 0; normal (default).<br>[4] = 1; hold R, A, and B counters in reset. |
| 016 | [3] | B counter bypass | B counter bypass. When this bit is 1, set the A counter in Register 0x013 to 0.<br>[3] = 0; normal (default).<br>[3] = 1; B counter is set to divide-by-1. This allows the prescaler setting to determine the divide for the N divider. |
| 016 | [2:0] | Prescaler P | Prescaler: DM = dual modulus and FD = fixed divide. The Prescaler P is part of the feedback divider. |

| [2] | [1] | [0] | Mode | Prescaler |
|---|---|---|---|---|
| 0 | 0 | 0 | FD | Divide-by-1. |
| 0 | 0 | 1 | FD | Divide-by-2. |
| 0 | 1 | 0 | DM | Divide-by-2 and divide-by-3 when A ≠ 0; divide-by-2 when A = 0. |
| 0 | 1 | 1 | DM | Divide-by-4 and divide-by-5 when A ≠ 0; divide-by-4 when A = 0. |
| 1 | 0 | 0 | DM | Divide-by-8 and divide-by-9 when A ≠ 0; divide-by-8 when A = 0. |
| 1 | 0 | 1 | DM | Divide-by-16 and divide-by-17 when A ≠ 0; divide-by-16 when A = 0. |
| 1 | 1 | 0 | DM | Divide-by-32 and divide-by-33 when A ≠ 0; divide-by-32 when A = 0 (default). |
| 1 | 1 | 1 | FD | Divide-by-3. |

| 017 | [7:2] | STATUS pin control | Selects the signal that appears at the STATUS pin. Register 0x01D[7] must be 0 to reprogram the STATUS pin. |
|---|---|---|---|

| [7] | [6] | [5] | [4] | [3] | [2] | Level or Dynamic Signal | Signal at STATUS Pin |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | LVL | Ground, dc (default). |
| 0 | 0 | 0 | 0 | 0 | 1 | DYN | N divider output (after the delay). |
| 0 | 0 | 0 | 0 | 1 | 0 | DYN | R divider output (after the delay). |
| 0 | 0 | 0 | 0 | 1 | 1 | DYN | A divider output. |
| 0 | 0 | 0 | 1 | 0 | 0 | DYN | Prescaler output. |
| 0 | 0 | 0 | 1 | 0 | 1 | DYN | PFD up pulse. |
| 0 | 0 | 0 | 1 | 1 | 0 | DYN | PFD down pulse. |
| 0 | X | X | X | X | X | LVL | Ground (dc); for all other cases of 0XXXXX not specified. |
| | | | | | | | The selections that follow are the same as for REFMON. |
| 1 | 0 | 0 | 0 | 0 | 0 | LVL | Ground (dc). |
| 1 | 0 | 0 | 0 | 0 | 1 | DYN | REF1 clock (differential reference when in differential mode). |
| 1 | 0 | 0 | 0 | 1 | 0 | DYN | REF2 clock (not applicable in differential mode). |
| 1 | 0 | 0 | 0 | 1 | 1 | DYN | Selected reference to PLL (differential reference when in differential mode). |
| 1 | 0 | 0 | 1 | 0 | 0 | DYN | Unselected reference to PLL (not available in differential mode). |

| Reg. Addr (Hex) | Bit(s) | Name | Description | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | [7] | [6] | [5] | [4] | [3] | [2] | Level or Dynamic Signal | Signal at STATUS Pin |
| | | | 1 | 0 | 0 | 1 | 0 | 1 | LVL | Status of selected reference (status of differential reference); active high. |
| | | | 1 | 0 | 0 | 1 | 1 | 0 | LVL | Status of unselected reference (not available in differential mode); active high. |
| | | | 1 | 0 | 0 | 1 | 1 | 1 | LVL | Status of REF1 frequency (active high). |
| | | | 1 | 0 | 1 | 0 | 0 | 0 | LVL | Status of REF2 frequency (active high). |
| | | | 1 | 0 | 1 | 0 | 0 | 1 | LVL | (Status of REF1 frequency) AND (status of REF2 frequency). |
| | | | 1 | 0 | 1 | 0 | 1 | 0 | LVL | (DLD) AND (status of selected reference) AND (status of CLK). |
| | | | 1 | 0 | 1 | 0 | 1 | 1 | LVL | Status of CLK frequency (active high). |
| | | | 1 | 0 | 1 | 1 | 0 | 0 | LVL | Selected reference (low = REF1, high = REF2). |
| | | | 1 | 0 | 1 | 1 | 0 | 1 | LVL | DLD; active high. |
| | | | 1 | 0 | 1 | 1 | 1 | 0 | LVL | Holdover active (active high). |
| | | | 1 | 0 | 1 | 1 | 1 | 1 | LVL | LD pin comparator output (active high). |
| | | | 1 | 1 | 0 | 0 | 0 | 0 | LVL | VS (PLL power supply). |
| | | | 1 | 1 | 0 | 0 | 0 | 1 | DYN | $\overline{\text{REF1 clock}}$ (differential reference when in differential mode). |
| | | | 1 | 1 | 0 | 0 | 1 | 0 | DYN | $\overline{\text{REF2 clock}}$ (not available in differential mode). |
| | | | 1 | 1 | 0 | 0 | 1 | 1 | DYN | $\overline{\text{Selected reference to PLL}}$ (differential reference when in differential mode). |
| | | | 1 | 1 | 0 | 1 | 0 | 0 | DYN | $\overline{\text{Unselected reference to PLL}}$ (not available when in differential mode). |
| | | | 1 | 1 | 0 | 1 | 0 | 1 | LVL | Status of selected reference (status of differential reference); active low. |
| | | | 1 | 1 | 0 | 1 | 1 | 0 | LVL | Status of unselected reference (not available in differential mode); active low. |
| | | | 1 | 1 | 0 | 1 | 1 | 1 | LVL | Status of REF1 frequency (active low). |
| | | | 1 | 1 | 1 | 0 | 0 | 0 | LVL | Status of REF2 frequency (active low). |
| | | | 1 | 1 | 1 | 0 | 0 | 1 | LVL | $\overline{\text{(Status of REF1 frequency) AND (status of REF2 frequency)}}$. |
| | | | 1 | 1 | 1 | 0 | 1 | 0 | LVL | $\overline{\text{(DLD) AND (Status of selected reference) AND (status of VCO)}}$. |
| | | | 1 | 1 | 1 | 0 | 1 | 1 | LVL | Status of CLK frequency (active low). |
| | | | 1 | 1 | 1 | 1 | 0 | 0 | LVL | Selected reference (low = REF2, high = REF1). |
| | | | 1 | 1 | 1 | 1 | 0 | 1 | LVL | DLD (active low). |
| | | | 1 | 1 | 1 | 1 | 1 | 0 | LVL | Holdover active (active low). |
| | | | 1 | 1 | 1 | 1 | 1 | 1 | LVL | LD pin comparator output (active low). |

| | | | | | |
|---|---|---|---|---|---|
| 017 | [1:0] | Antibacklash pulse width | **[1]** | **[0]** | **Antibacklash Pulse Width (ns)** |
| | | | 0 | 0 | 2.9 (default) |
| | | | 0 | 1 | 1.3 |
| | | | 1 | 0 | 6.0 |
| | | | 1 | 1 | 2.9 |
| 018 | [7] | Enable CMOS reference input dc offset | Enables dc offset in single-ended CMOS input mode to prevent chattering when ac-coupled and input is lost.<br>[7] = 0; disable dc offset (default).<br>[7] = 1; enable dc offset. | | |
| 018 | [6:5] | Lock detect counter | Required consecutive number of PFD cycles with edges inside lock detect window before the DLD indicates a locked condition. | | |
| | | | **[6]** | **[5]** | **PFD Cycles to Determine Lock** |
| | | | 0 | 0 | 5 (default) |
| | | | 0 | 1 | 16 |
| | | | 1 | 0 | 64 |
| | | | 1 | 1 | 255 |

| Reg. Addr (Hex) | Bit(s) | Name | Description |
|---|---|---|---|
| 018 | [4] | Digital lock detect window | If the time difference of the rising edges at the inputs to the PFD are less than the lock detect window time, the digital lock detect flag is set. The flag remains set until the time difference is greater than the loss-of-lock threshold.<br>[4] = 0; high range (default). The default setting is 3.5 ns.<br>[4] = 1; low range. |
| 018 | [3] | Disable digital lock detect | Digital lock detect operation.<br>[3] = 0; normal lock detect operation (default).<br>[3] = 1; disable lock detect. |
| 019 | [7:6] | R, A, B counters $\overline{\text{SYNC}}$ pin reset | **[7]** / **[6]** / **Action**<br>0 / 0 / Do nothing on $\overline{\text{SYNC}}$ (default).<br>0 / 1 / Asynchronous reset.<br>1 / 0 / Synchronous reset.<br>1 / 1 / Do nothing on $\overline{\text{SYNC}}$. |
| 019 | [5:3] | R path delay | R path delay, see Table 2 (default: 0x0). |
| 019 | [2:0] | N path delay | N path delay, see Table 2 (default: 0x0). |
| 01A | [7] | Enable STATUS pin divider | Enables a divide-by-4 on the STATUS pin. This makes it easier to look at low duty-cycle signals out of the R and N dividers.<br>[7] = 0; divide-by-4 disabled on STATUS pin (default).<br>[7] = 1; divide-by-4 enabled on STATUS pin. |
| 01A | [6] | Ref freq monitor threshold | Sets the reference (REF1/REF2) frequency monitor's detection threshold frequency. This does not affect the CLK frequency monitor's detection threshold (see Table 14, REF1, REF2, and CLK frequency status monitor parameter).<br>[6] = 0; frequency valid if frequency is above 1.02 MHz (default).<br>[6] = 1; frequency valid if frequency is above 8 kHz. |
| 01A | [5:0] | LD pin control | Selects the signal that is connected to the LD pin. |

| [5] | [4] | [3] | [2] | [1] | [0] | Level or Dynamic Signal | Signal at LD Pin |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | LVL | Digital lock detect (high = lock; low = unlock, default). |
| 0 | 0 | 0 | 0 | 0 | 1 | DYN | P-channel, open-drain lock detect (analog lock detect). |
| 0 | 0 | 0 | 0 | 1 | 0 | DYN | N-channel, open-drain lock detect (analog lock detect). |
| 0 | 0 | 0 | 0 | 1 | 1 | HIZ | Tristate (high-Z) LD pin. |
| 0 | 0 | 0 | 1 | 0 | 0 | CUR | Current source lock detect (110 µA when DLD is true). |
| 0 | X | X | X | X | X | LVL | Ground (dc); for all other cases of 0XXXXX not specified. |
| | | | | | | | The selections that follow are the same as for REFMON. |
| 1 | 0 | 0 | 0 | 0 | 0 | LVL | Ground (dc). |
| 1 | 0 | 0 | 0 | 0 | 1 | DYN | REF1 clock (differential reference when in differential mode). |
| 1 | 0 | 0 | 0 | 1 | 0 | DYN | REF2 clock (not applicable in differential mode). |
| 1 | 0 | 0 | 0 | 1 | 1 | DYN | Selected reference to PLL (differential reference when in differential mode). |
| 1 | 0 | 0 | 1 | 0 | 0 | DYN | Unselected reference to PLL (not available in differential mode). |
| 1 | 0 | 0 | 1 | 0 | 1 | LVL | Status of selected reference (status of differential reference); active high. |
| 1 | 0 | 0 | 1 | 1 | 0 | LVL | Status of unselected reference (not available in differential mode); active high. |
| 1 | 0 | 0 | 1 | 1 | 1 | LVL | Status of REF1 frequency (active high). |
| 1 | 0 | 1 | 0 | 0 | 0 | LVL | Status of REF2 frequency (active high). |
| 1 | 0 | 1 | 0 | 0 | 1 | LVL | (Status of REF1 frequency) AND (status of REF2 frequency). |
| 1 | 0 | 1 | 0 | 1 | 0 | LVL | (DLD) AND (status of selected reference) AND (status of CLK). |
| 1 | 0 | 1 | 0 | 1 | 1 | LVL | Status of CLK frequency (active high). |
| 1 | 0 | 1 | 1 | 0 | 0 | LVL | Selected reference (low = REF1, high = REF2). |
| 1 | 0 | 1 | 1 | 0 | 1 | LVL | DLD; active high. |
| 1 | 0 | 1 | 1 | 1 | 0 | LVL | Holdover active (active high). |

| Reg. Addr (Hex) | Bit(s) | Name | Description | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | [5] | [4] | [3] | [2] | [1] | [0] | Level or Dynamic Signal | Signal at LD Pin |
| | | | 1 | 0 | 1 | 1 | 1 | 1 | LVL | Not applicable, do not use. |
| | | | 1 | 1 | 0 | 0 | 0 | 0 | LVL | VS (PLL supply). |
| | | | 1 | 1 | 0 | 0 | 0 | 1 | DYN | $\overline{\text{REF1 clock}}$ (differential reference when in differential mode). |
| | | | 1 | 1 | 0 | 0 | 1 | 0 | DYN | $\overline{\text{REF2 clock}}$ (not applicable in differential mode). |
| | | | 1 | 1 | 0 | 0 | 1 | 1 | DYN | $\overline{\text{Selected reference to PLL}}$ (differential reference when in differential mode). |
| | | | 1 | 1 | 0 | 1 | 0 | 0 | DYN | $\overline{\text{Unselected reference to PLL}}$ (not available when in differential mode). |
| | | | 1 | 1 | 0 | 1 | 0 | 1 | LVL | Status of selected reference (status of differential reference); active low. |
| | | | 1 | 1 | 0 | 1 | 1 | 0 | LVL | Status of unselected reference (not available in differential mode); active low. |
| | | | 1 | 1 | 0 | 1 | 1 | 1 | LVL | Status of REF1 frequency (active low). |
| | | | 1 | 1 | 1 | 0 | 0 | 0 | LVL | Status of REF2 frequency (active low). |
| | | | 1 | 1 | 1 | 0 | 0 | 1 | LVL | $\overline{\text{(Status of REF1 frequency) AND (status of REF2 frequency).}}$ |
| | | | 1 | 1 | 1 | 0 | 1 | 0 | LVL | $\overline{\text{(DLD) AND (Status of selected reference) AND (status of VCO).}}$ |
| | | | 1 | 1 | 1 | 0 | 1 | 1 | LVL | Status of CLK frequency (active low). |
| | | | 1 | 1 | 1 | 1 | 0 | 0 | LVL | Selected reference (low = REF2, high = REF1). |
| | | | 1 | 1 | 1 | 1 | 0 | 1 | LVL | DLD; active low. |
| | | | 1 | 1 | 1 | 1 | 1 | 0 | LVL | Holdover active (active low). |
| | | | 1 | 1 | 1 | 1 | 1 | 1 | LVL | Not applicable, do not use. |
| 01B | [7] | Enable CLK frequency monitor | Enables or disables the CLK frequency monitor.<br>[7] = 0; disable the CLK frequency monitor (default).<br>[7] = 1; enable the CLK frequency monitor. | | | | | | | |
| 01B | [6] | Enable REF2 ($\overline{\text{REFIN}}$) frequency monitor | Enables or disables the REF2 frequency monitor.<br>[6] = 0; disable the REF2 ($\overline{\text{REFIN}}$) frequency monitor (default).<br>[6] = 1; enable the REF2 ($\overline{\text{REFIN}}$) frequency monitor. | | | | | | | |
| 01B | [5] | Enable REF1 (REFIN) frequency monitor | REF1 (REFIN) frequency monitor enabled; this is for both REF1 (single-ended) and REFIN (differential) inputs (as selected by differential reference mode).<br>[5] = 0; disable the REF1 (REFIN) frequency monitor (default).<br>[5] = 1; enable the REF1 (REFIN) frequency monitor. | | | | | | | |
| 01B | [4:0] | REFMON pin control | Selects the signal that is connected to the REFMON pin. | | | | | | | |
| | | | [4] | [3] | [2] | [1] | [0] | | Level or Dynamic Signal | Signal at REFMON Pin |
| | | | 0 | 0 | 0 | 0 | 0 | | LVL | Ground, dc (default). |
| | | | 0 | 0 | 0 | 0 | 1 | | DYN | REF1 clock (differential reference when in differential mode). |
| | | | 0 | 0 | 0 | 1 | 0 | | DYN | REF2 clock (not applicable in differential mode). |
| | | | 0 | 0 | 0 | 1 | 1 | | DYN | Selected reference to PLL (differential reference when in differential mode). |
| | | | 0 | 0 | 1 | 0 | 0 | | DYN | Unselected reference to PLL (not available in differential mode). |
| | | | 0 | 0 | 1 | 0 | 1 | | LVL | Status of selected reference (status of differential reference); active high. |
| | | | 0 | 0 | 1 | 1 | 0 | | LVL | Status of unselected reference (not available in differential mode); active high. |
| | | | 0 | 0 | 1 | 1 | 1 | | LVL | Status REF1 frequency (active high). |
| | | | 0 | 1 | 0 | 0 | 0 | | LVL | Status REF2 frequency (active high). |
| | | | 0 | 1 | 0 | 0 | 1 | | LVL | (Status REF1 frequency) AND (status REF2 frequency). |
| | | | 0 | 1 | 0 | 1 | 0 | | LVL | (DLD) AND (status of selected reference) AND (status of CLK). |

| Reg. Addr (Hex) | Bit(s) | Name | Description | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | [4] | [3] | [2] | [1] | [0] | Level or Dynamic Signal | Signal at REFMON Pin |
| | | | 0 | 1 | 0 | 1 | 1 | LVL | Status of CLK frequency (active high). |
| | | | 0 | 1 | 1 | 0 | 0 | LVL | Selected reference (low = REF1, high = REF2). |
| | | | 0 | 1 | 1 | 0 | 1 | LVL | DLD; active low. |
| | | | 0 | 1 | 1 | 1 | 0 | LVL | Holdover active (active high). |
| | | | 0 | 1 | 1 | 1 | 1 | LVL | LD pin comparator output (active high). |
| | | | 1 | 0 | 0 | 0 | 0 | LVL | VS (PLL supply). |
| | | | 1 | 0 | 0 | 0 | 1 | DYN | $\overline{\text{REF1 clock}}$ (differential reference when in differential mode). |
| | | | 1 | 0 | 0 | 1 | 0 | DYN | $\overline{\text{REF2 clock}}$ (not available in differential mode). |
| | | | 1 | 0 | 0 | 1 | 1 | DYN | $\overline{\text{Selected reference to PLL}}$ (differential reference when in differential mode). |
| | | | 1 | 0 | 1 | 0 | 0 | DYN | $\overline{\text{Unselected reference to PLL}}$ (not available when in differential mode). |
| | | | 1 | 0 | 1 | 0 | 1 | LVL | Status of selected reference (status of differential reference); active low. |
| | | | 1 | 0 | 1 | 1 | 0 | LVL | Status of unselected reference (not available in differential mode); active low. |
| | | | 1 | 0 | 1 | 1 | 1 | LVL | Status of REF1 frequency (active low). |
| | | | 1 | 1 | 0 | 0 | 0 | LVL | Status of REF2 frequency (active low). |
| | | | 1 | 1 | 0 | 0 | 1 | LVL | $\overline{\text{(Status of REF1 frequency) AND (status of REF2 frequency)}}$. |
| | | | 1 | 1 | 0 | 1 | 0 | LVL | $\overline{\text{(DLD) AND (status of selected reference) AND (status of CLK)}}$. |
| | | | 1 | 1 | 0 | 1 | 1 | LVL | Status of CLK frequency (active low). |
| | | | 1 | 1 | 1 | 0 | 0 | LVL | Selected reference (low = REF2, high = REF1). |
| | | | 1 | 1 | 1 | 0 | 1 | LVL | DLD; active low. |
| | | | 1 | 1 | 1 | 1 | 0 | LVL | Holdover active (active low). |
| | | | 1 | 1 | 1 | 1 | 1 | LVL | LD pin comparator output (active low). |
| 01C | [7] | Disable switchover deglitch | Disables or enables the switchover deglitch circuit.<br>[7] = 0; enable the switchover deglitch circuit (default).<br>[7] = 1; disable the switchover deglitch circuit. | | | | | | |
| 01C | [6] | Select REF2 | If Register 0x01C[5] = 0, selects the reference for PLL when in manual; register selected reference control.<br>[6] = 0; select REF1 (default).<br>[6] = 1; select REF2. | | | | | | |
| 01C | [5] | Use REF_SEL pin | If Register 0x01C[4] = 0 (manual), sets the method of PLL reference selection.<br>[5] = 0; use Register 0x01C[6] (default).<br>[5] = 1; use REF_SEL pin. | | | | | | |
| 01C | [4:3] | Reserved | Default: 00b | | | | | | |
| 01C | [2] | Enable REF2 | This bit turns the REF2 power on.<br>[2] = 0; REF2 power off (default).<br>[2] = 1; REF2 power on. | | | | | | |
| 01C | [1] | Enable REF1 | This bit turns the REF1 power on.<br>[1] = 0; REF1 power off (default).<br>[1] = 1; REF1 power on. | | | | | | |
| 01C | [0] | Enable differential reference | Selects the PLL reference mode, differential or single-ended. Register 0x01C[2:1] must be cleared when this bit is set.<br>[0] = 0; single-ended reference mode (default).<br>[0] = 1; differential reference mode. | | | | | | |
| 01D | [7] | Enable STATUS_EEPROM at STATUS pin | Enables the STATUS_EEPROM signal at the STATUS pin.<br>[7] = 0; the STATUS pin is controlled by the Register 0x017[7:2] selection.<br>[7] = 1; select the STATUS_EEPROM signal at STATUS pin. This bit overrides Register 0x017[7:2] (default). | | | | | | |

| Reg. Addr (Hex) | Bit(s) | Name | Description |
|---|---|---|---|
| 01D | [6] | Enable XTAL OSC | Enables the maintaining amplifier needed by a crystal oscillator at the PLL reference input.<br>[6] = 0; crystal oscillator maintaining amplifier disabled (default).<br>[6] = 1; crystal oscillator maintaining amplifier enabled. |
| 01D | [5] | Enable clock doubler | Enable PLL reference input clock doubler.<br>[5] = 0; doubler disabled (default).<br>[5] = 1; doubler enabled. |
| 01D | [4] | Disable PLL status register | Disables the PLL status register readback.<br>[4] = 0; PLL status register enabled (default).<br>[4] = 1; PLL status register disabled. If this bit is set, Register 0x01F is not automatically updated. |
| 01D | [3] | Enable LD pin comparator | Enables the LD pin voltage comparator. This is used with the LD pin current source lock detect mode. When the AD9522 is in internal (automatic) holdover mode, this enables the use of the voltage on the LD pin to determine if the PLL was previously in a locked state (see Figure 34). Otherwise, this can be used with the REFMON and STATUS pins to monitor the voltage on the LD pin.<br>[3] = 0; disable LD pin comparator and ignore the LD pin voltage; internal/automatic holdover controller treats this pin as true (high, default).<br>[3] = 1; enable LD pin comparator (use LD pin voltage to determine if the PLL was previously locked). |
| 01D | [1] | Enable external holdover | Enables the external hold control through the $\overline{\text{SYNC}}$ pin. (This disables the internal holdover mode.)<br>[1] = 0; automatic holdover mode, holdover controlled by automatic holdover circuit (default).<br>[1] = 1; external holdover mode, holdover controlled by $\overline{\text{SYNC}}$ pin. |
| 01D | [0] | Enable holdover | Enables the internally controlled holdover function.<br>[0] = 0; holdover disabled (default).<br>[0] = 1; holdover enabled. |
| 01E | [1] | Enable zero delay | Enables zero delay function.<br>[1] = 0; disables zero delay function (default).<br>[1] = 1; enables zero delay function. |
| 01F | [5] | Holdover active (read-only) | Readback register. Indicates if the part is in the holdover state (see Figure 34). This is not the same as holdover enabled.<br>[5] = 0; not in holdover.<br>[5] = 1; holdover state active. |
| 01F | [4] | REF2 selected (read-only) | Readback register. Indicates which PLL reference is selected as the input to the PLL.<br>[4] = 0; REF1 selected (or differential reference if in differential mode).<br>[4] = 1; REF2 selected. |
| 01F | [3] | CLK frequency > threshold (read-only) | Readback register. Indicates if the external CLK input frequency is greater than the threshold (see Table 14, REF1, REF2, and external CLK frequency status monitor parameter).<br>[3] = 0; CLK frequency is less than the threshold.<br>[3] = 1; CLK frequency is greater than the threshold. |
| 01F | [2] | REF2 frequency > threshold (read-only) | Readback register. Indicates if the frequency of the signal at REF2 is greater than the threshold frequency set by Register 0x01A[6].<br>[2] = 0; REF2 frequency is less than the threshold frequency.<br>[2] = 1; REF2 frequency is greater than the threshold frequency. |
| 01F | [1] | REF1 frequency > threshold (read-only) | Readback register. Indicates if the frequency of the signal at REF1 is greater than the threshold frequency set by Register 0x01A[6].<br>[1] = 0; REF1 frequency is less than the threshold frequency.<br>[1] = 1; REF1 frequency is greater than the threshold frequency. |
| 01F | [0] | Digital lock detect (read-only) | Readback register. Digital lock detect.<br>[0] = 0; PLL is not locked.<br>[0] = 1; PLL is locked. |

**Table 48. Output Driver Control**

| Reg. Addr (Hex) | Bit(s) | Name | Description |
|---|---|---|---|
| 0F0 | [7] | OUT0 format | Selects the output type for OUT0.<br>[7] = 0; LVDS (default).<br>[7] = 1; CMOS. |
| 0F0 | [6:5] | OUT0 CMOS configuration | Sets the CMOS output configuration for OUT0 when 0x0F0[7] = 1. |

| [6:5] | OUT0A | OUT0B |
|---|---|---|
| 00 | Tristate | Tristate |
| 01 | On | Tristate |
| 10 | Tristate | On |
| 11 (default) | On | On |

| 0F0 | [4:3] | OUT0 polarity | Sets the output polarity for OUT0. |
|---|---|---|---|

| [7] | [4] | [3] | Output Type | OUT0A | OUT0B |
|---|---|---|---|---|---|
| 0 (default) | X | 0 | LVDS | Noninverting | Inverting |
| 0 | X | 1 | LVDS | Inverting | Noninverting |
| 1 | 0 (default) | 0 (default) | CMOS | Noninverting | Noninverting |
| 1 | 0 | 1 | CMOS | Inverting | Inverting |
| 1 | 1 | 0 | CMOS | Noninverting | Inverting |
| 1 | 1 | 1 | CMOS | Inverting | Noninverting |

| 0F0 | [2:1] | OUT0 LVDS differential voltage | Sets the LVDS output differential voltage ($V_{OD}$). |
|---|---|---|---|

| [2] | [1] | $I_{OD}$ (mA) |
|---|---|---|
| 0 | 0 | 1.75 ($V_{OD}$ = 175 mV for 100 Ω termination across differential pair) |
| 0 (default) | 1 (default) | 3.5 ($V_{OD}$ = 350 mV for 100 Ω termination across differential pair) |
| 1 | 0 | 5.25 ($V_{OD}$ = 525 mV for 100 Ω termination across differential pair) |
| 1 | 1 | 7.0 ($V_{OD}$ = 700 mV for 100 Ω termination across differential pair) |

| 0F0 | [0] | OUT0 LVDS power-down | LVDS power-down.<br>[0] = 0; normal operation (default).<br>[0] = 1; power-down. Output driver is in a high impedance state. |
|---|---|---|---|
| 0F1 | [7:0] | OUT1 control | This register controls OUT1, and the bit assignments for this register are identical to Register 0x0F0. |
| 0F2 | [7:0] | OUT2 control | This register controls OUT2, and the bit assignments for this register are identical to Register 0x0F0. |
| 0F3 | [7:0] | OUT3 control | This register controls OUT3, and the bit assignments for this register are identical to Register 0x0F0. |
| 0F4 | [7:0] | OUT4 control | This register controls OUT4, and the bit assignments for this register are identical to Register 0x0F0. |
| 0F5 | [7:0] | OUT5 control | This register controls OUT5, and the bit assignments for this register are identical to Register 0x0F0. |
| 0F6 | [7:0] | OUT6 control | This register controls OUT6, and the bit assignments for this register are identical to Register 0x0F0. |
| 0F7 | [7:0] | OUT7 control | This register controls OUT7, and the bit assignments for this register are identical to Register 0x0F0. |
| 0F8 | [7:0] | OUT8 control | This register controls OUT8, and the bit assignments for this register are identical to Register 0x0F0. |
| 0F9 | [7:0] | OUT9 control | This register controls OUT9, and the bit assignments for this register are identical to Register 0x0F0. |
| 0FA | [7:0] | OUT10 control | This register controls OUT10, and the bit assignments for this register are identical to Register 0x0F0. |
| 0FB | [7:0] | OUT11 control | This register controls OUT11, and the bit assignments for this register are identical to Register 0x0F0. |
| 0FC | [7] | CSDLD En OUT7 | OUT7 is enabled only if CSDLD is high. |

| [7] | CSDLD Signal | OUT7 Enable Status |
|---|---|---|
| 0 | 0 | Not affected by CSDLD signal (default). |
| 1 | 0 | Asynchronous power-down. |
| 1 | 1 | Asynchronously enable OUT7 if not powered down by other settings. To use this feature, the user must use current source digital lock detect, and set the enable LD pin comparator bit (0x01D[3]). |

| 0FC | [6] | CSDLD En OUT6 | OUT6 is enabled only if CSDLD is high. Setting is identical to Register 0x0FC[7]. |
|---|---|---|---|
| 0FC | [5] | CSDLD En OUT5 | OUT5 is enabled only if CSDLD is high. Setting is identical to Register 0x0FC[7]. |
| 0FC | [4] | CSDLD En OUT4 | OUT4 is enabled only if CSDLD is high. Setting is identical to Register 0x0FC[7]. |
| 0FC | [3] | CSDLD En OUT3 | OUT3 is enabled only if CSDLD is high. Setting is identical to Register 0x0FC[7]. |

| Reg. Addr (Hex) | Bit(s) | Name | Description |
|---|---|---|---|
| 0FC | [2] | CSDLD En OUT2 | OUT2 is enabled only if CSDLD is high. Setting is identical to Register 0x0FC[7]. |
| 0FC | [1] | CSDLD En OUT1 | OUT1 is enabled only if CSDLD is high. Setting is identical to Register 0x0FC[7]. |
| 0FC | [0] | CSDLD En OUT0 | OUT0 is enabled only if CSDLD is high. Setting is identical to Register 0x0FC[7]. |
| 0FD | [3] | CSDLD En OUT11 | OUT11 is enabled only if CSDLD is high. Setting is identical to Register 0x0FC[7]. |
| 0FD | [2] | CSDLD En OUT10 | OUT10 is enabled only if CSDLD is high. Setting is identical to Register 0x0FC[7]. |
| 0FD | [1] | CSDLD En OUT9 | OUT9 is enabled only if CSDLD is high. Setting is identical to Register 0x0FC[7]. |
| 0FD | [0] | CSDLD En OUT8 | OUT8 is enabled only if CSDLD is high. Setting is identical to Register 0x0FC[7]. |

**Table 49. LVDS Channel Dividers**

| Reg. Addr (Hex) | Bit(s) | Name | Description |
|---|---|---|---|
| 190 | [7:4] | Divider 0 low cycles | Number of clock cycles (minus 1) of the divider input during which the divider output stays low. A value of 0x7 means the divider is low for eight input clock cycles (default: 0x7). |
| 190 | [3:0] | Divider 0 high cycles | Number of clock cycles (minus 1) of the divider input during which the divider output stays high. A value of 0x7 means the divider is high for eight input clock cycles (default: 0x7). |
| 191 | [7] | Divider 0 bypass | Bypasses and powers down the divider; routes input to divider output.<br>[7] = 0; use divider (default).<br>[7] = 1; bypass divider. |
| 191 | [6] | Divider 0 ignore SYNC | Ignore SYNC.<br>[6] = 0; obey chip-level SYNC signal (default).<br>[6] = 1; ignore chip-level SYNC signal. |
| 191 | [5] | Divider 0 force high | Forces divider output to a specific state. This requires that ignore SYNC also be set. Note that this bit has no effect if the channel divider is bypassed.<br>[5] = 0; divider output forced to low (default).<br>[5] = 1; divider output forced to high. |
| 191 | [4] | Divider 0 start high | Selects clock output to start high or start low.<br>[4] = 0; start low (default).<br>[4] = 1; start high. |
| 191 | [3:0] | Divider 0 phase offset | Phase offset (default: 0x0). |
| 192 | [2] | Channel 0 power-down | Channel 0 powers down.<br>[2] = 0; normal operation (default).<br>[2] = 1; powered down. (OUT0/$\overline{\text{OUT0}}$, OUT1/$\overline{\text{OUT1}}$, and OUT2/$\overline{\text{OUT2}}$ are put into the high impedance power-down mode by setting this bit.) |
| 192 | [0] | Disable Divider 0 DCC | Duty-cycle correction function.<br>[0] = 0; enable duty-cycle correction (default).<br>[0] = 1; disable duty-cycle correction. |
| 193 | [7:4] | Divider 1 low cycles | Number of clock cycles (minus 1) of the divider input during which the divider output stays low. A value of 0x3 means the divider is low for four input clock cycles (default: 0x3). |
| 193 | [3:0] | Divider 1 high cycles | Number of clock cycles (minus 1) of the divider input during which the divider output stays high. A value of 0x3 means the divider is high for four input clock cycles (default: 0x3). |
| 194 | [7] | Divider 1 bypass | Bypasses and powers down the divider; routes input to divider output.<br>[7] = 0; use divider (default).<br>[7] = 1; bypass divider. |
| 194 | [6] | Divider 1 ignore SYNC | Ignore SYNC.<br>[6] = 0; obey chip-level SYNC signal (default).<br>[6] = 1; ignore chip-level SYNC signal. |

| Reg. Addr (Hex) | Bit(s) | Name | Description |
|---|---|---|---|
| 194 | [5] | Divider 1 force high | Forces divider output to a specific state. This requires that ignore SYNC also be set. Note that this bit has no effect if the channel divider is bypassed.<br>[5] = 0; divider output forced to low (default).<br>[5] = 1; divider output forced to high. |
| 194 | [4] | Divider 1 start high | Selects clock output to start high or start low.<br>[4] = 0; start low (default).<br>[4] = 1; start high. |
| 194 | [3:0] | Divider 1 phase offset | Phase offset (default: 0x0). |
| 195 | [2] | Channel 1 power-down | Channel 1 powers down.<br>[2] = 0; normal operation (default).<br>[2] = 1; powered down. (OUT3/$\overline{\text{OUT3}}$, OUT4/$\overline{\text{OUT4}}$, and OUT5/$\overline{\text{OUT5}}$ are put into the high impedance power-down mode by setting this bit.) |
| 195 | [0] | Disable Divider 1 DCC | Duty-cycle correction function.<br>[0] = 0; enable duty-cycle correction (default).<br>[0] = 1; disable duty-cycle correction. |
| 196 | [7:4] | Divider 2 low cycles | Number of clock cycles (minus 1) of the divider input during which the divider output stays low. A value of 0x1 means the divider is low for two input clock cycles (default: 0x1). |
| 196 | [3:0] | Divider 2 high cycles | Number of clock cycles (minus 1) of the divider input during which the divider output stays high. A value of 0x1 means the divider is high for two input clock cycles (default: 0x1). |
| 197 | [7] | Divider 2 bypass | Bypasses and powers down the divider; routes input to divider output.<br>[7] = 0; use divider (default).<br>[7] = 1; bypass divider. |
| 197 | [6] | Divider 2 ignore SYNC | Ignore SYNC.<br>[6] = 0; obey chip-level SYNC signal (default).<br>[6] = 1; ignore chip-level SYNC signal. |
| 197 | [5] | Divider 2 force high | Forces divider output to a specific state. This requires that ignore SYNC also be set. Note that this bit has no effect if the channel divider is bypassed.<br>[5] = 0; divider output forced to low (default).<br>[5] = 1; divider output forced to high. |
| 197 | [4] | Divider 2 start high | Selects clock output to start high or start low.<br>[4] = 0; start low (default).<br>[4] = 1; start high. |
| 197 | [3:0] | Divider 2 phase offset | Phase offset (default: 0x0). |
| 198 | [2] | Channel 2 power-down | Channel 2 powers down.<br>[2] = 0; normal operation (default).<br>[2] = 1; powered down. (OUT6/$\overline{\text{OUT6}}$, OUT7/$\overline{\text{OUT7}}$, and OUT8/$\overline{\text{OUT8}}$ are put into the high impedance power-down mode by setting this bit.) |
| 198 | [0] | Disable Divider 2 DCC | Duty-cycle correction function.<br>[0] = 0; enable duty-cycle correction (default).<br>[0] = 1; disable duty-cycle correction. |
| 199 | [7:4] | Divider 3 low cycles | Number of clock cycles (minus 1) of the divider input during which the divider output stays low. A value of 0x0 means the divider is low for one input clock cycle (default: 0x0). |
| 199 | [3:0] | Divider 3 high cycles | Number of clock cycles (minus 1) of the divider input during which the divider output stays high. A value of 0x0 means the divider is high for one input clock cycle (default: 0x0). |
| 19A | [7] | Divider 3 bypass | Bypasses and powers down the divider; routes input to divider output.<br>[7] = 0; use divider (default).<br>[7] = 1; bypass divider. |
| 19A | [6] | Divider 3 ignore SYNC | Ignore SYNC.<br>[6] = 0; obey chip-level SYNC signal (default).<br>[6] = 1; ignore chip-level SYNC signal. |

| Reg. Addr (Hex) | Bit(s) | Name | Description |
|---|---|---|---|
| 19A | [5] | Divider 3 force high | Forces divider output to a specific state. This requires that ignore SYNC also be set. Note that this bit has no effect if the channel divider is bypassed.<br>[5] = 0; divider output forced to low (default).<br>[5] = 1; divider output forced to high. |
| 19A | [4] | Divider 3 start high | Selects clock output to start high or start low.<br>[4] = 0; start low (default).<br>[4] = 1; start high. |
| 19A | [3:0] | Divider 3 phase offset | Phase offset (default: 0x0). |
| 19B | [2] | Channel 3 power-down | Channel 3 powers down.<br>[2] = 0; normal operation (default).<br>[2] = 1; powered down. (OUT9/$\overline{\text{OUT9}}$, OUT10/$\overline{\text{OUT10}}$, and OUT11/$\overline{\text{OUT11}}$ are put into the high impedance power-down mode by setting this bit.) |
| 19B | [0] | Disable Divider 3 DCC | Duty-cycle correction function.<br>[0] = 0; enable duty-cycle correction (default).<br>[0] = 1; disable duty-cycle correction. |

**Table 50. VCO Divider and CLK Input**

| Reg. Addr (Hex) | Bit(s) | Name | Description | | | |
|---|---|---|---|---|---|---|
| 1E0 | [2:0] | VCO divider | **[2]** | **[1]** | **[0]** | **Divide** |
| | | | 0 | 0 | 0 | 2 (default) |
| | | | 0 | 0 | 1 | 3 |
| | | | 0 | 1 | 0 | 4 |
| | | | 0 | 1 | 1 | 5 |
| | | | 1 | 0 | 0 | 6 |
| | | | 1 | 0 | 1 | Output static |
| | | | 1 | 1 | 0 | 1 (bypass) |
| | | | 1 | 1 | 1 | Output static |
| 1E1 | [4] | Power-down clock input section | Powers down the clock input section (including CLK buffer, VCO divider, and CLK tree).<br>[4] = 0; normal operation (default).<br>[4] = 1; power down. | | | |
| 1E1 | [0] | Bypass VCO divider | Bypasses or uses the VCO divider.<br>[0] = 0; use VCO divider (default).<br>[0] = 1; bypass VCO divider. | | | |

**Table 51. System**

| Reg. Addr (Hex) | Bit(s) | Name | Description |
|---|---|---|---|
| 230 | [3] | Disable power-on SYNC | Power-on SYNC mode. Used to disable the antiruntpulse circuitry.<br>[3] = 0; enable the antiruntpulse circuitry (default).<br>[3] = 1; disable the antiruntpulse circuitry. |
| 230 | [2] | Power-down SYNC | Powers down the SYNC function.<br>[2] = 0; normal operation of the SYNC function (default).<br>[2] = 1; power-down the SYNC circuitry. |
| 230 | [1] | Power-down distribution reference | Powers down the reference for the distribution section.<br>[1] = 0; normal operation of the reference for the distribution section (default).<br>[1] = 1; powers down the reference for the distribution section. |

| Reg. Addr (Hex) | Bit(s) | Name | Description |
|---|---|---|---|
| 230 | [0] | Soft SYNC | The soft SYNC bit works the same as the $\overline{\text{SYNC}}$ pin, except that the polarity of the bit is reversed; that is, a high level forces the selected channels into a predetermined static state, and a 1-to-0 transition triggers a SYNC.<br>[0] = 0; same as $\overline{\text{SYNC}}$ high.<br>[0] = 1; same as $\overline{\text{SYNC}}$ low. |

**Table 52. Update All Registers**

| Reg. Addr (Hex) | Bit(s) | Name | Description |
|---|---|---|---|
| 232 | [0] | IO_UPDATE | This bit must be set to 1 to transfer the contents of the buffer registers into the active registers. This bit is self-clearing; that is, it does not have to be set back to 0.<br>[0] = 1 (self-clearing); update all active registers to the contents of the buffer registers. |

**Table 53. EEPROM Buffer Segment**

| Reg. Addr (Hex) | Bit(s) | Name | Description |
|---|---|---|---|
| A00 to A16 | [7:0] | EEPROM Buffer Segment Register 1 to EEPROM Buffer Segment Register 23 | The EEPROM buffer segment section stores the starting address and number of bytes that are to be stored and read back to and from the EEPROM. Because the AD9522 register space is noncontiguous, the EEPROM controller needs to know the starting address and number of bytes in the AD9522 register space to store and retrieve from the EEPROM. In addition, there are special instructions for the EEPROM controller, operational codes (that is, IO_UPDATE and end-of-data) that are also stored in the EEPROM buffer segment. The on-chip default setting of the EEPROM buffer segment registers is designed such that all registers are transferred to/from the EEPROM, and an IO_UPDATE is issued after transfer. See the Programming the EEPROM Buffer Segment section for more information. |

**Table 54. EEPROM Control**

| Reg. Addr (Hex) | Bit(s) | Name | Description |
|---|---|---|---|
| B00 | [0] | STATUS_EEPROM (read-only) | This read-only register indicates the status of the data transferred between the EEPROM and the buffer register bank during the writing and reading of the EEPROM. This signal is also available at the STATUS pin when Register 0x01D[7] is set.<br>[0] = 0; data transfer is done.<br>[0] = 1; data transfer is not done. |
| B01 | [0] | EEPROM data error (read-only) | This read-only register indicates an error during the data transferred between the EEPROM and the buffer.<br>[0] = 0; no error. Data is correct.<br>[0] = 1; incorrect data detected. |
| B02 | [1] | SOFT_EEPROM | When the EEPROM pin is tied low, setting SOFT_EEPROM resets the AD9522 using the settings saved in EEPROM.<br>[1] = 1; soft reset with EEPROM settings (self-clearing). This bit self-clears on the next serial port clock cycle after the completion of writing to this register. |
| B02 | [0] | Enable EEPROM write | Enables the user to write to the EEPROM.<br>[0] = 0; EEPROM write protection is enabled. User cannot write to EEPROM (default).<br>[0] = 1; EEPROM write protection is disabled. User can write to EEPROM. |
| B03 | [0] | REG2EEPROM | Transfers data from the buffer register to the EEPROM (self-clearing).<br>[0] = 1; setting this bit initiates the data transfer from the buffer register to the EEPROM (writing process); it is reset by the $I^2C$ master after the data transfer is done. |

# APPLICATIONS INFORMATION

## FREQUENCY PLANNING USING THE AD9522

The AD9522 is a highly flexible PLL. When choosing the PLL settings and version of the AD9522, keep the following guidelines in mind.

The AD9522 has four frequency dividers: the reference (or R) divider, the feedback (or N) divider, the VCO divider, and the channel divider. When trying to achieve a particularly difficult frequency divide ratio requiring a large amount of frequency division, some of the frequency division can be done by either the VCO divider or the channel divider, thus allowing a higher phase detector frequency and more flexibility in choosing the loop bandwidth.

Choosing a nominal charge pump current in the middle of the allowable range as a starting point allows the designer to increase or decrease the charge pump current, and thus allows the designer to fine-tune the PLL loop bandwidth in either direction.

ADIsimCLK is a powerful PLL modeling tool that is a very accurate tool for determining the optimal loop filter for a given application.

## USING THE AD9522 OUTPUTS FOR ADC CLOCK APPLICATIONS

Any high speed ADC is extremely sensitive to the quality of the sampling clock of the AD9522. An ADC can be thought of as a sampling mixer, and any noise, distortion, or time jitter on the clock is combined with the desired signal at the analog-to-digital output. Clock integrity requirements scale with the analog input frequency and resolution, with higher analog input frequency applications at ≥14-bit resolution being the most stringent. The theoretical SNR of an ADC is limited by the ADC resolution and the jitter on the sampling clock. Considering an ideal ADC of infinite resolution where the step size and quantization error can be ignored, the available SNR can be expressed approximately by

$$SNR(\text{dB}) = 20\log\left(\frac{1}{2\pi f_A t_J}\right)$$

where:
$f_A$ is the highest analog frequency being digitized.
$t_J$ is the rms jitter on the sampling clock.

Figure 57 shows the required sampling clock jitter as a function of the analog frequency and effective number of bits (ENOB).

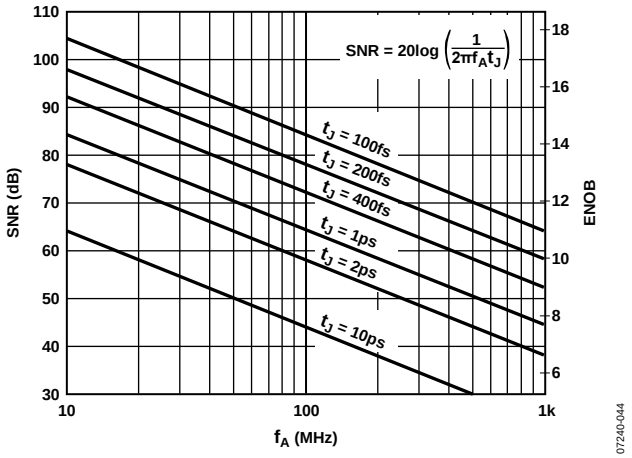


*Figure 57. SNR and ENOB vs. Analog Input Frequency*

See the AN-756 Application Note, *Sampled Systems and the Effects of Clock Phase Noise and Jitter*, and the AN-501 Application Note, *Aperture Uncertainty and ADC System Performance*.

Many high performance ADCs feature differential clock inputs to simplify the task of providing the required low jitter clock on a noisy PCB. Distributing a single-ended clock on a noisy PCB can result in coupled noise on the sampling clock. Differential distribution has inherent common-mode rejection that can provide superior clock performance in a noisy environment. The differential LVDS outputs of the AD9522 enable clock solutions that maximize converter SNR performance.

Consider the input requirements of the ADC (differential or single-ended, logic level termination) when selecting the best clocking/converter solution. In some cases, the LVPECL outputs of the AD9520 may be desirable for clocking a converter instead of the LVDS outputs of the AD9522.

## LVDS CLOCK DISTRIBUTION

The AD9522 provides clock outputs that are selectable as either CMOS or LVDS level outputs. LVDS is a differential output option that uses a current mode output stage. The nominal current is 3.5 mA, which yields 350 mV output swing across a 100 Ω resistor. An output current of 7 mA is also available in cases where a larger output swing is required. The LVDS output meets or exceeds all ANSI/TIA/EIA-644 specifications.

A recommended termination circuit for the LVDS outputs is shown in Figure 58. If ac coupling is necessary, place decoupling capacitors either before or after the 100 Ω termination resistor.

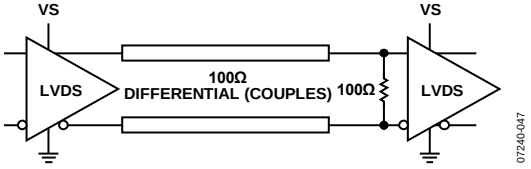


*Figure 58. LVDS Output Termination*

See the AN-586 Application Note for more information on LVDS.

## CMOS CLOCK DISTRIBUTION

The output drivers of the AD9522 can be configured as CMOS drivers. When selected as a CMOS driver, each output becomes a pair of CMOS outputs, each of which can be individually turned on or off and set as inverting or noninverting. These outputs are 3.3 V CMOS compatible.

When single-ended CMOS clocking is used, some of the following guidelines apply.

Point-to-point connections should be designed such that each driver has only one receiver, if possible. Connecting outputs in this manner allows for simple termination schemes and minimizes ringing due to possible mismatched impedances on the output trace. Series termination at the source is generally required to provide transmission line matching and/or to reduce current transients at the driver.

The value of the resistor is dependent on the board design and timing requirements (typically 10 Ω to 100 Ω is used). CMOS outputs are also limited in terms of the capacitive load or trace length that they can drive. Typically, trace lengths less than 3 inches are recommended to preserve signal rise/fall times and signal integrity.

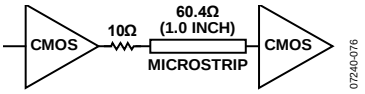


*Figure 59. Series Termination of CMOS Output*

Termination at the far end of the PCB trace is a second option. The CMOS outputs of the AD9522 do not supply enough current to provide a full voltage swing with a low impedance resistive, far-end termination, as shown in Figure 60. The far-end termination network should match the PCB trace impedance and provide the desired switching point. The reduced signal swing may still meet receiver input requirements in some applications. This can be useful when driving long trace lengths on less critical nets.

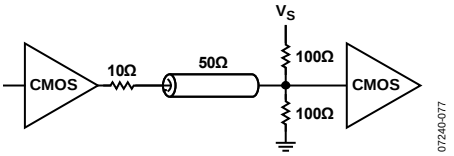


*Figure 60. CMOS Output with Far-End Termination*

Because of the limitations of single-ended CMOS clocking, consider using differential outputs when driving high speed signals over long traces. The AD9522 offers LVDS outputs that are better suited for driving long traces where the inherent noise immunity of differential signaling provides superior performance for clocking converters.

# OUTLINE DIMENSIONS

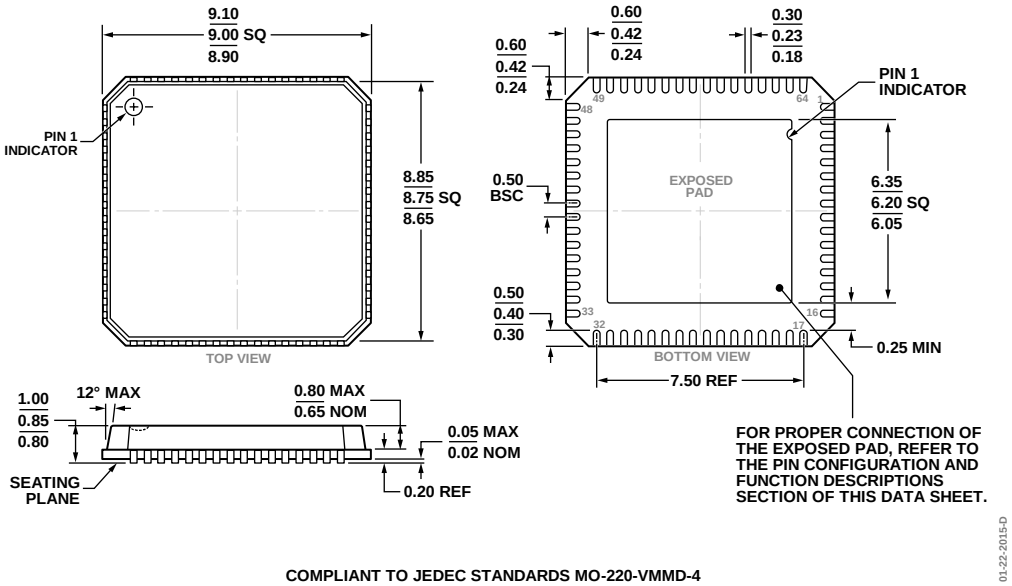


*Figure 61. 64-Lead Lead Frame Chip Scale Package [LFCSP_VQ]*
*9 mm × 9 mm Body, Very Thin Quad*
*CP-64-4*
*Dimensions shown in millimeters*

## ORDERING GUIDE

| Model[1] | Temperature Range | Package Description | Package Option |
|---|---|---|---|
| AD9522-5BCPZ | −40°C to +85°C | 64-Lead Lead Frame Chip Scale Package [LFCSP_VQ] | CP-64-4 |
| AD9522-5BCPZ-REEL7 | −40°C to +85°C | 64-Lead Lead Frame Chip Scale Package [LFCSP_VQ] | CP-64-4 |
| AD9522-5/PCBZ | | Evaluation Board | |

[1] Z = RoHS Compliant Part.

©2008–2015 Analog Devices, Inc. All rights reserved. Trademarks and registered trademarks are the property of their respective owners.
D07240-0-3/15(A)

## Looking for pricing, stock, or lifecycle information?

Click below to explore more details on WIN SOURCE:

- View AD9522-5BCPZ-REEL7 on WIN SOURCE
- Analog Devices Inc. Information

## Optimize Your Supply Chain with WIN SOURCE Solutions

- Global Sourcing Solution
- Obsolete Management
- Cost Control Management
- Shortage Management
- Alternative Solution
- Excess Inventory Management