16-BIT, LOW POWER, VOLTAGE OUTPUT, I²C INTERFACE DIGITAL-TO-ANALOG CONVERTER

FEATURES

- **Micropower Operation: 160 µA @ 5 V**
- **Power-On Reset to Zero**
- **Single Supply: +2.7 V to +5.5 V**
- **16-Bit Monotonic**
- **Settling Time: 10 µs to ±0.003% FSR**
- **I²C™ Interface With High-Speed Mode**
- **Supports Data Receive and Transmit**
- **On-Chip Rail-to-Rail Output Buffer**
- **Double-Buffered Input Register**
- **Supports Synchronous Multichannel Update**
- **Offset Error: ±1 mV max at 25°C**
- **Full-Scale Error: ±3 mV max at 25°C**
- **Small 8 Lead MSOP Package**

APPLICATIONS

- **Process Control**
- **Data Acquisition Systems**
- **Closed-Loop Servo Control**
- **PC Peripherals**
- **Portable Instrumentation**

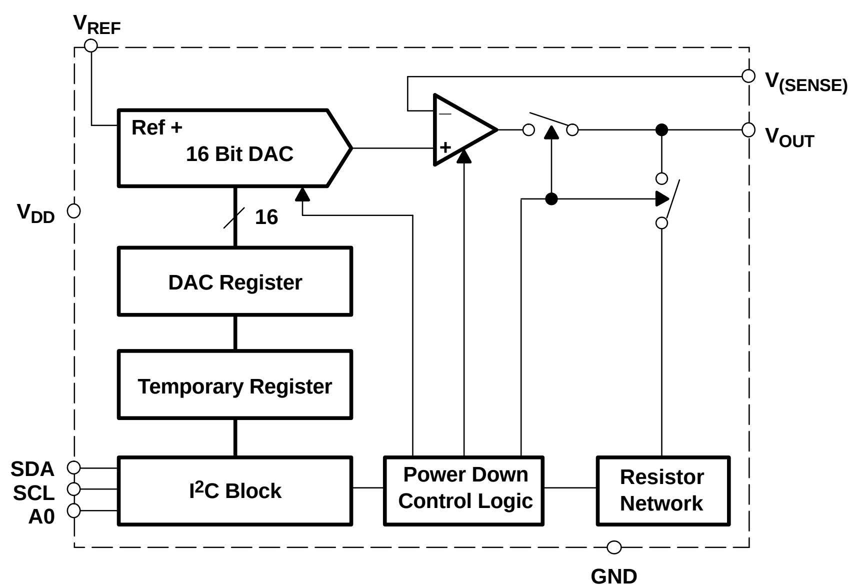
DESCRIPTION

The DAC8571 is a small low-power, 16-bit voltage output DAC with an I²C compatible two-wire serial interface. Its on-chip precision output amplifier allows rail-to-rail output swing and settles within 10 microseconds. The DAC8571 architecture is 16-bit monotonic, and factory trimming typically achieves ±4 mV absolute accuracy at all codes. The DAC8571 requires an external reference voltage to set its output voltage range.

The low power consumption and small size of this part make it ideally suited to portable battery operated equipment. The power consumption is typically 800 µW at V_{DD} = 5 V reducing to 1 µW in power-down mode.

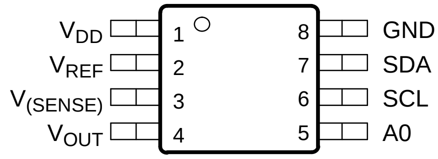
The DAC8571 incorporates a 2-wire I²C interface. Standard, fast, and high-speed modes of I²C operation are all supported up to 3.4 MHz serial clock speeds. Multichannel synchronous data update and power-down operations are supported through the I²C bus. DAC8571 is also capable of transmitting the contents of its serial shift register, a key feature for I²C system verification.

The DAC8571 is available in an 8-lead MSOP package and is specified over -40°C to 105°C.



These devices have limited built-in ESD protection. The leads should be shorted together or the device placed in conductive foam during storage or handling to prevent electrostatic damage to the MOS gates.

PIN CONFIGURATIONS



PIN DESCRIPTION

Pin	Name	Function
1	V_{DD}	Analog voltage supply input
2	V_{REF}	Positive reference voltage input
3	$V_{(SENSE)}$	Analog output sense
4	V_{OUT}	Analog output voltage from DAC
5	A0	Device address select
6	SCL	Serial clock input
7	SDA	Serial data input/output
8	GND	Ground reference point for all circuitry on the part

PACKAGE/ORDERING INFORMATION

Product	Package	Package Designator	Specified Temperature Range	Package Marking	Ordering Number	Transport Media, Quantity
DAC8571	8-MSOP	DGK	-40°C to +105°C	D871	DAC8571IDGK DAC8571IDGKR	Tube, 80 Tape & Reel, 2500

ABSOLUTE MAXIMUM RATINGS(1)

		UNITS
V_{DD} to GND		-0.3 V to +6 V
Digital input voltage to GND		-0.3V to V_{DD} + 0.3V
V_{OUT} to GND		-0.3V to +V_{DD} + 0.3V
Operating temperature range		-40°C to + 105°C
Storage temperature range		-65°C to +150°C
Junction temperature range ($T_J$max)		+ 150°C
Θ_{JA}Thermal impedance		260°C/W
Θ_{JC}Thermal impedance		44°C/W
Lead temperature, soldering	Vapor phase (60s)	215°C
	Infrared (15s)	220°C

(1) Stresses above those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. Exposure to absolute maximum conditions for extended periods may affect device reliability.

ELECTRICAL CHARACTERISTICS

V_{DD} = +2.7 V to +5.5 V; R_L = 2 kΩ to GND; C_L = 200 pF to GND; low power mode; all specifications -40°C to 105°C (unless otherwise noted)

		DAC8571			
PARAMETER	**CONDITIONS**	**MIN**	**TYP**	**MAX**	**UNITS**
STATIC PERFORMANCE (1)					
Resolution		16			Bits
Relative accuracy				±0.098	% of FSR
Differential nonlinearity	Monotonic by design		±0.25	±1	LSB
Offset error	Measured at code 485, 25°C		0.3	±1.0	mV
	Measured at code 485, -40°C...105°C		1.0	±5.0	
Full-scale error	Measured at code 64714, 25°C		0.5	±3.0	mV
	Measured at code 64714, -40°C...105°C		1.0	±5.0	
Gain error	Measured at code 64714, 25°C		1.0	±3.0	mV
	Measured at code 64714, -40°C...105°C		2.0	±5.0	
Zero code error drift	All zeroes loaded to DAC register		-20		µV/°C
Gain temperature coefficient			-5		ppm of FSR/°C
Absolute accuracy	All codes from code 485 to code 64714, 25°C		±2.5		mV
	All codes from code 485 to code 64714, -40°C...105°C		±3.5		
OUTPUT CHARACTERISTICS (2)					
Output voltage range		0		V_{REF}	V
Output voltage settling time (full scale)	R_L = 2 kΩ; C_L < 200 pF, fast settling		8	10	µs
	R_L = 2 kΩ; C_L = 500 pF, fast settling		12		µs
	R_L = 2 kΩ; C_L < 200 pF, low power		13	15	µs
Slew rate	R_L = 2 kΩ; C_L < 200 pF, fast settling		1		V/µs
	R_L = 2 kΩ; C_L < 200 pF, low power		0.5		
Capacitive load stability	R_L = ∞		470		pF
	R_L = 2 kΩ		1000		pF
Digital-to-analog glitch impulse			20		nV-s
Digital feedthrough			0.5		nV-s
DC output impedance			1		Ω
Short circuit current	V_{DD} = +5 V		50		mA
	V_{DD} = +3 V		20		mA
Power-up time	Coming out of power-down mode, V_{DD} = +5 V		2.5		µs
	Coming out of power-down mode, V_{DD} = +3 V		5		µs
PSRR			0.75		mV/V
REFERENCE INPUT					
V_{REFH} input range		0		V_{DD}	V
Reference input impedance			140		kΩ
LOGIC INPUTS (2)					
Input current				±1	µA
V_{IN_L}, Input low voltage	V_{DD} = 2.7-5.5 V			$0.3V_{DD}$	V
V_{IN_H0} , Input high voltage	V_{DD} = 2.7-5.5 V	$0.7V_{DD}$			V
Pin capacitance				3	pF
POWER REQUIREMENTS					
V_{DD}		2.7		5.5	V
I_{DD} (normal operation)	DAC active, I_{ref} included				

(1) Linearity calculated using a reduced code range of 485 to 64714. Output unloaded.
(2) Assured by design and characterization, not production tested.

ELECTRICAL CHARACTERISTICS (continued)

V_{DD} = +2.7 V to +5.5 V; R_L = 2 kΩ to GND; C_L = 200 pF to GND; low power mode; all specifications -40°C to 105°C (unless otherwise noted)

		DAC8571			
PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS
V_{DD} = +4.5 V to +5.5 V	$V_{IH} = V_{DD}$, V_{IL} = GND, fast settling		250	400	µA
	$V_{IH} = V_{DD}$, V_{IL} = GND, low power		160	225	
V_{DD} = +2.7 V to +3.6 V	$V_{IH} = V_{DD}$, V_{IL} = GND, fast settling		240	380	µA
	$V_{IH} = V_{DD}$, V_{IL} = GND, low power		140	200	
I_{DD} (all power-down modes)	DAC active, I_{ref} included				
V_{DD} = +4.5 V to +5.5 V	$V_{IH} = V_{DD}$ and V_{IL} = GND		0.2	1	µA
V_{DD} = +2.7 V to +3.6 V	$V_{IH} = V_{DD}$ and V_{IL} = GND		0.05	1	µA
POWER EFFICIENCY					
I_{OUT}/I_{DD}	I_L = 2 mA, V_{DD} = +5 V		93%		

TIMING CHARACTERISTICS

V_{DD} = +2.7 V to +5.5 V; R_L = 2 kΩ to GND; all specifications -40°C to 105°C (unless otherwise noted)

SYMBOL	PARAMETER	TEST CONDITIONS	MIN	TYP	MAX	UNITS
t_{SCL}	SCL clock frequency	Standard mode			100	kHz
		Fast mode			400	kHz
		High-speed mode, C_B - 100pF max			3.4	MHz
		High-speed mode, C_B - 400pF max			1.7	MHz
t_{BUF}	Bus free time between a STOP and START condition	Standard mode	4.7			µs
		Fast mode	1.3			µs
t_{HO}; t_{STA}	Hold time (repeated) START condition	Standard mode	4.0			\µs
		Fast mode	600			ns
		High-speed mode	160			ns
t_{LOW}	LOW period of the SCL clock	Standard mode	4.7			µs
		Fast mode	1.3			µs
t_{HIGH}	HIGH period of the SCL clock	Standard mode	4.0			µs
		Fast mode	600			ns
		High-speed mode, C_B - 100pF max	60			ns
		High-speed mode, C_B - 400pF max	120			ns
t_{SU}; t_{STA}	Setup time for a repeated START condition	Standard mode	4.7			µs
		Fast mode	600			ns
		High-speed mode	160			ns
t_{SU}; t_{DAT}	Data setup time	Standard mode	250			ns
		Fast mode	100			ns
		High-speed mode	10			ns
t_{HD}; t_{DAT}	Data hold time	Standard mode	0		0.9	µs
		Fast mode	0		0.9	µs
		High-speed mode, C_B - 100pF max	0		70	ns
		High-speed mode, C_B - 400pF max	0		150	ns
t_{RCL}	Rise time of SCL signal	Standard mode	20 + 0.1C_B		1000	ns
		Fast mode	20 + 0.1C_B		300	ns
		High-speed mode, C_B - 100pF max	10		40	ns
		High-speed mode, C_B - 400pF max	20		80	ns

TIMING CHARACTERISTICS (continued)

V_{DD} = +2.7 V to +5.5 V; R_L = 2 kΩ to GND; all specifications -40°C to 105°C (unless otherwise noted)

SYMBOL	PARAMETER	TEST CONDITIONS	MIN	TYP	MAX	UNITS
t_{RCL1}	Rise time of SCL signal after a repeated START condition, and after an acknowledge BIT	Standard mode	20 + 0.1C_B		1000	ns
		Fast mode	20 + 0.1C_B		300	ns
		High-speed mode, C_B - 100pF max	10		80	ns
		High-speed mode, C_B - 400pF max	20		1600	ns
t_{FCL}	Fall time of SCL signal	Standard mode	20 + 0.1C_B		300	ns
		Fast mode	20 + 0.1C_B		300	ns
		High-speed mode, C_B - 100pF max	10		40	ns
		High-speed mode, C_B - 400pF max	20		80	ns
t_{RCA}	Rise time of SDA signal	Standard mode	20 + 0.1C_B		1000	ns
		Fast mode	20 + 0.1C_B		300	ns
		High-speed mode, C_B - 100pF max	10		80	ns
		High-speed mode, C_B - 400pF max	20		160	ns
t_{FDA}	Fall time of SDA signal	Standard mode	20 + 0.1C_B		300	ns
		Fast mode	20 + 0.1C_B		300	ns
		High-speed mode, C_B - 100pF max	10		80	ns
		High-speed mode, C_B - 400pF max	20		160	ns
t_{SU}; t_{STO}	Setup time for STOP condition	Standard mode	4.0			µs
		Fast mode	600			ns
		High-speed mode	160			ns
C_B	Capacitive load for SDA and SCL				400	pF
t_{SP}	Pulse width of spike suppressed	Fast mode			50	ns
		High-speed mode			10	ns
V_{NH}	Noise margin at the HIGH level for each connected device (including hysteresis)	Standard mode	0.2V_{DO}			V
		Fast mode				
		High-speed mode				
V_{NL}	Noise margin at the LOW level for each connected device (including hysteresis)	Standard mode	0.1V_{DO}			V
		Fast mode				
		High-speed mode				

TYPICAL CHARACTERISTICS

At T_A = +25°C, unless otherwise noted.

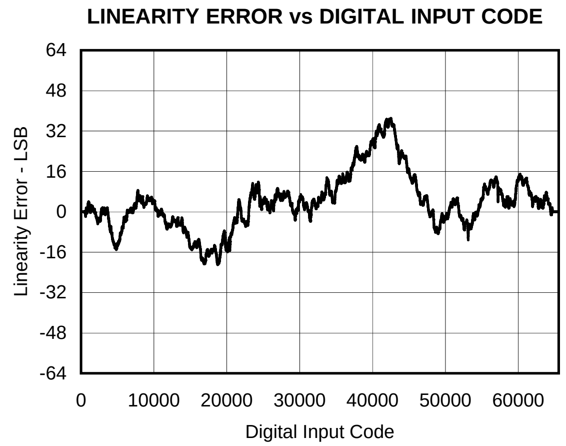


Figure 1.

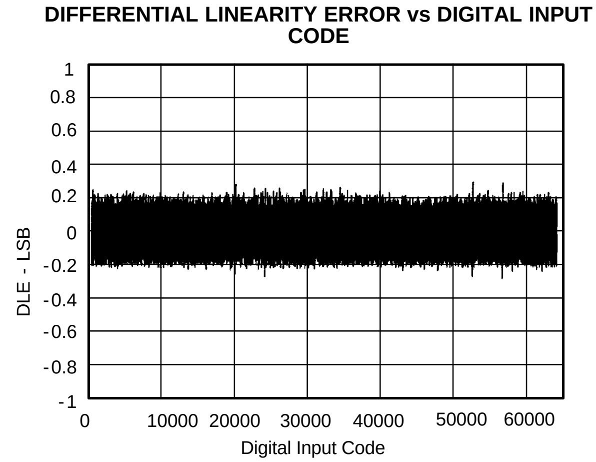


Figure 2.

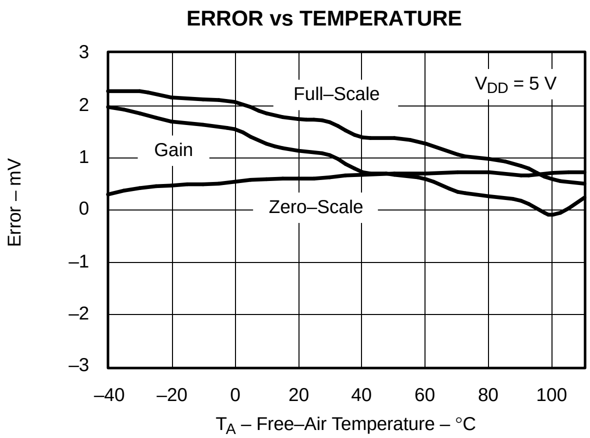


Figure 3.

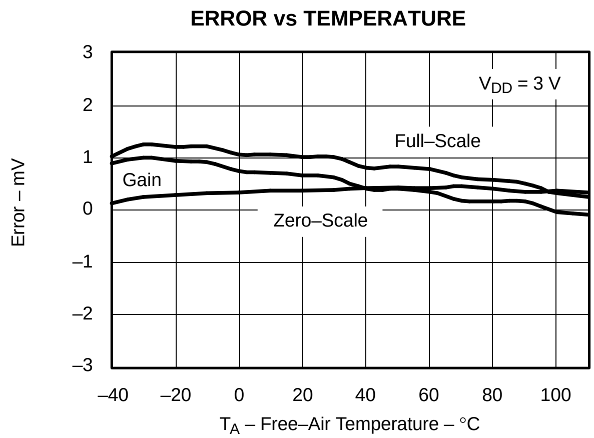


Figure 4.

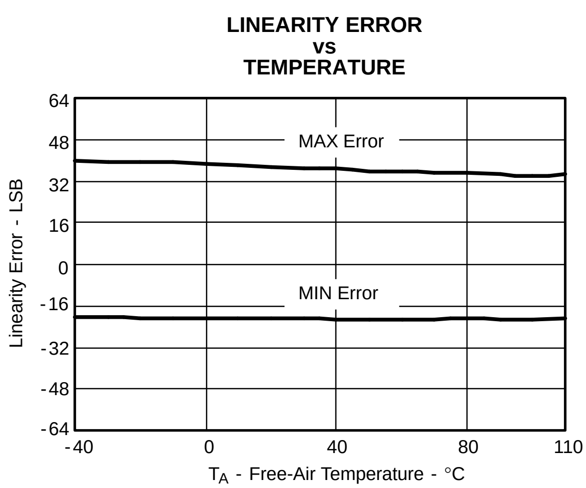


Figure 5.

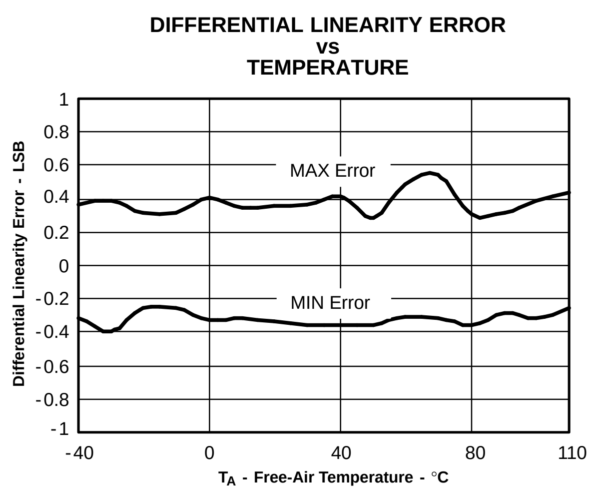


Figure 6.

TYPICAL CHARACTERISTICS (continued)

At T_A = +25°C, unless otherwise noted.

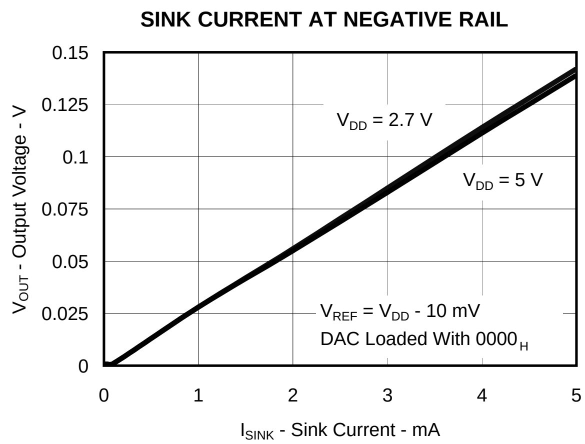


Figure 7.

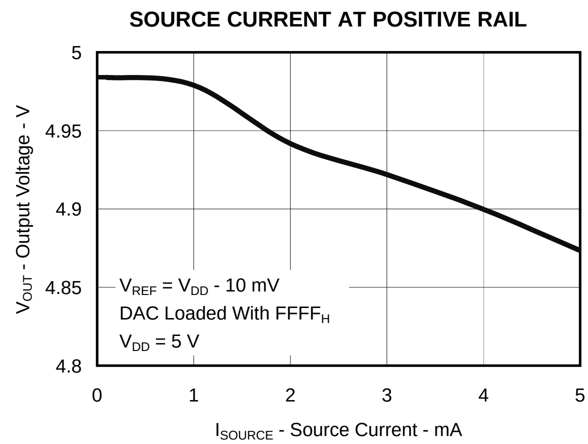


Figure 8.

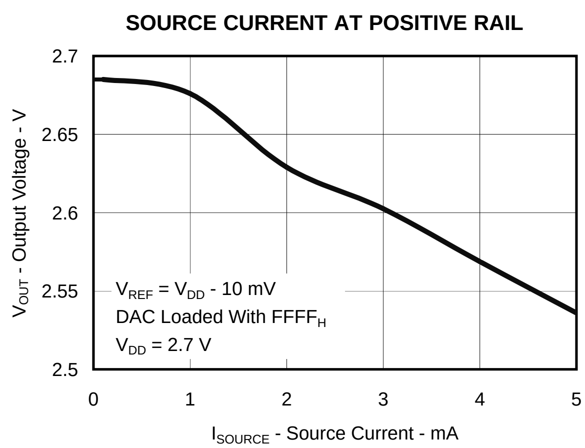


Figure 9.

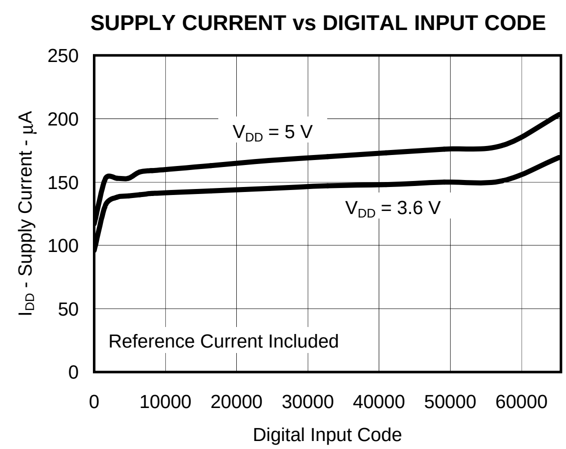


Figure 10.

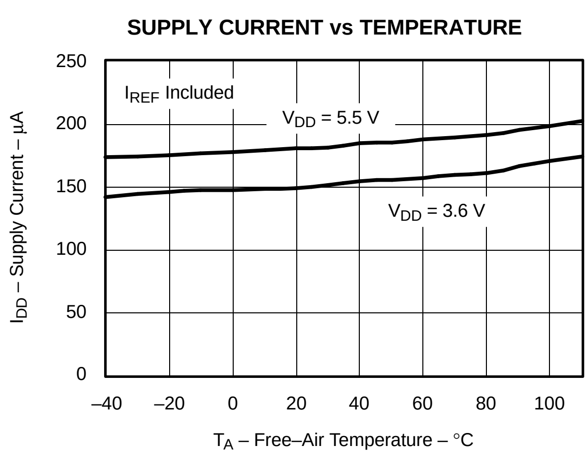


Figure 11.

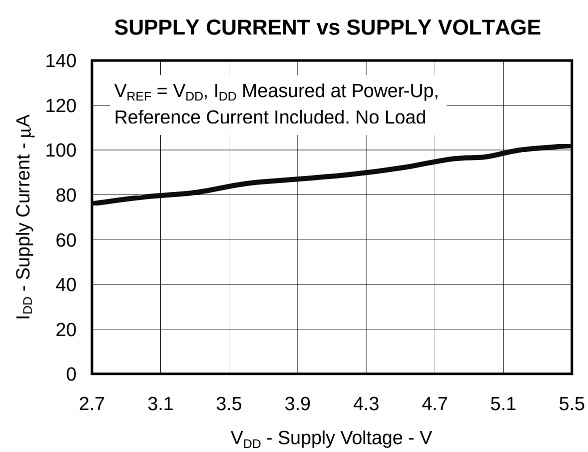


Figure 12.

TYPICAL CHARACTERISTICS (continued)

At T_A = +25°C, unless otherwise noted.

SUPPLY CURRENT vs LOGIC INPUT VOLTAGE

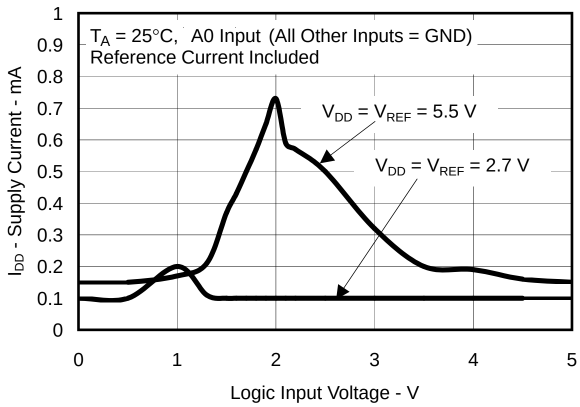


Figure 13.

HISTOGRAM OF CURRENT CONSUMPTION

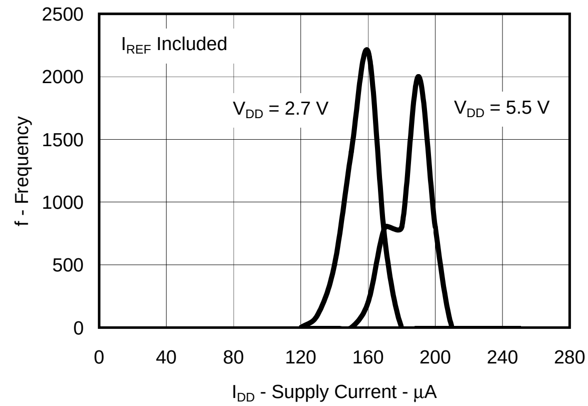


Figure 14.

EXITING POWER-DOWN MODE

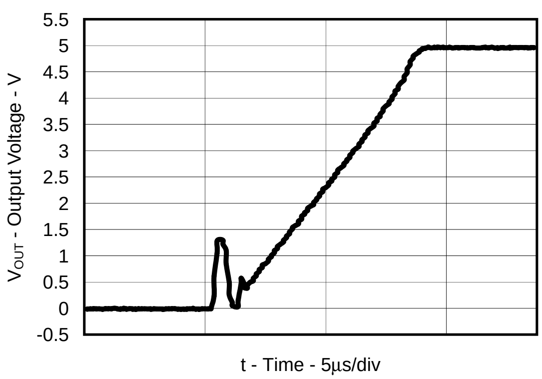


Figure 15.

OUTPUT GLITCH (Mid-Scale)

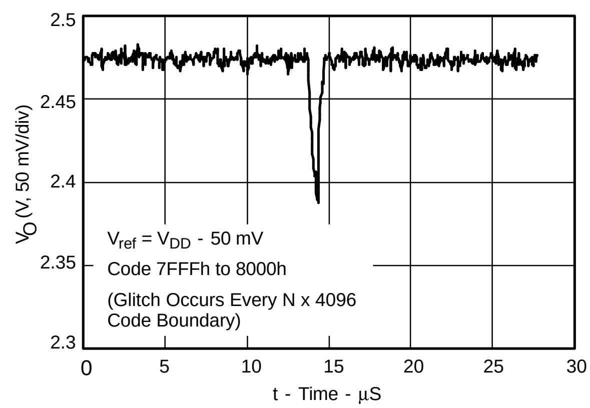


Figure 16.

ABSOLUTE ERROR

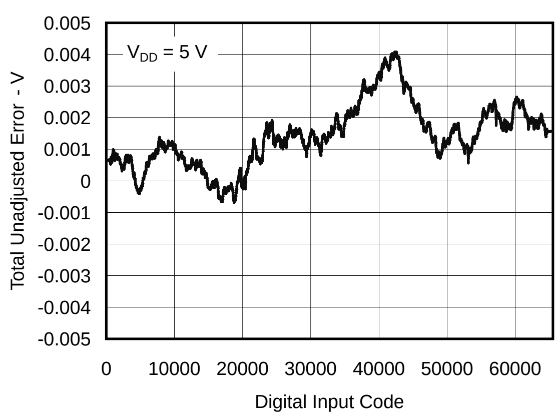


Figure 17.

FULL-SCALE SETTLING TIME (Large Signal)

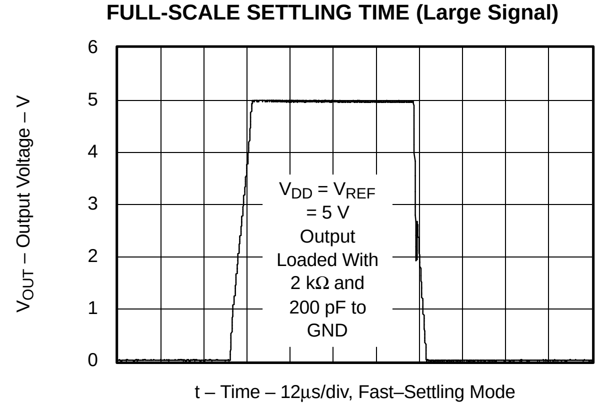


Figure 18.

TYPICAL CHARACTERISTICS (continued)

At T_A = +25°C, unless otherwise noted.

HALF-SCALE SETTLING TIME (Large Signal)

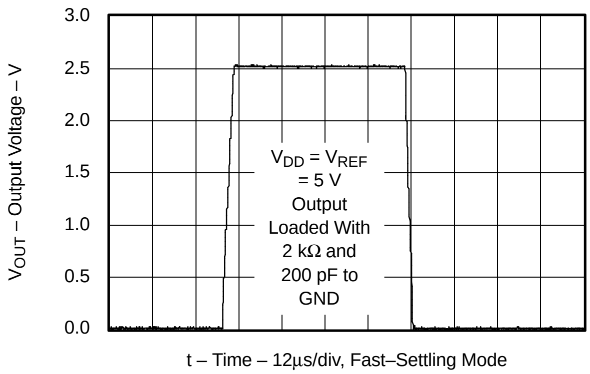


Figure 19.

FULL-SCALE SETTLING TIME (Large Signal)

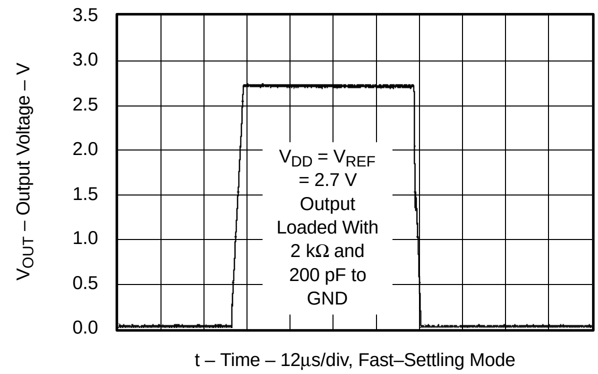


Figure 20.

HALF-SCALE SETTLING TIME

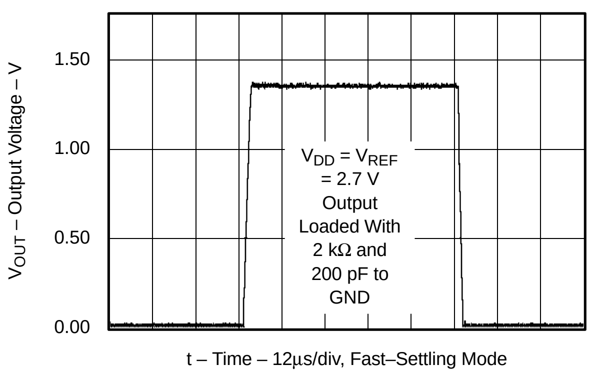


Figure 21.

SIGNAL-TO-NOISE RATIO vs OUTPUT FREQUENCY

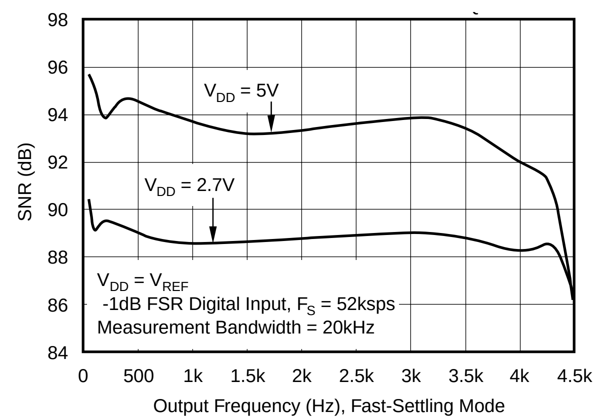


Figure 22.

TOTAL HARMONIC DISTORTION vs OUTPUT FREQUENCY

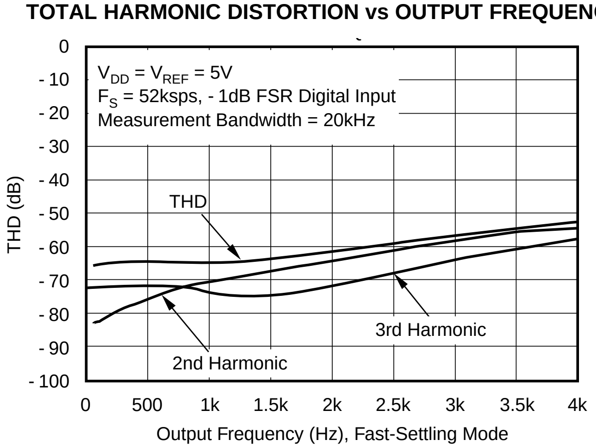


Figure 23.

TOTAL HARMONIC DISTORTION vs OUTPUT FREQUENCY

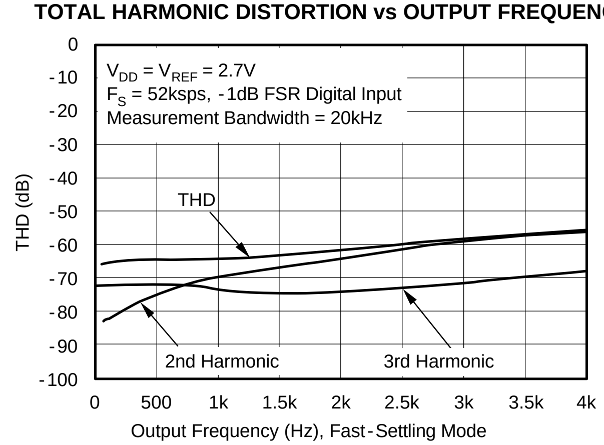


Figure 24.

TYPICAL CHARACTERISTICS (continued)

At T_A = +25°C, unless otherwise noted.

FULL-SCALE SETTLING TIME (Small-Signal-Positive Going Step)

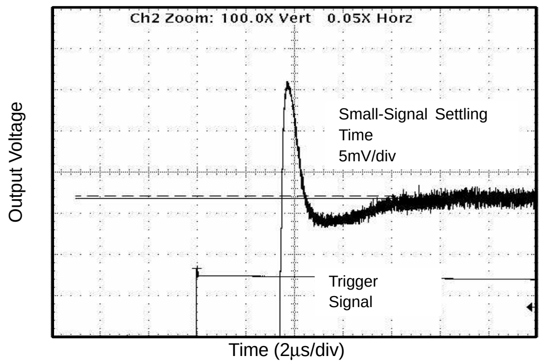


Figure 25.

FULL-SCALE SETTLING TIME (Small-Signal-Negative Going Step)

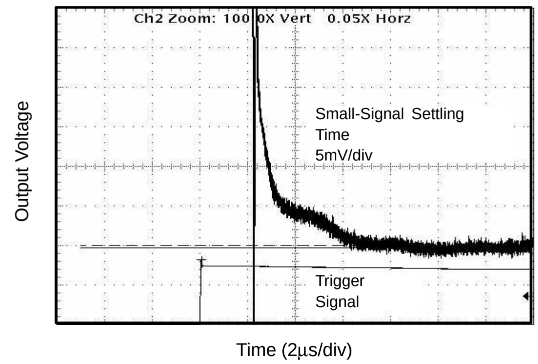


Figure 26.

THEORY OF OPERATION

D/A SECTION

The architecture of the DAC8571 consists of a string DAC followed by an output buffer amplifier. Figure 27 shows a block diagram of the DAC architecture.

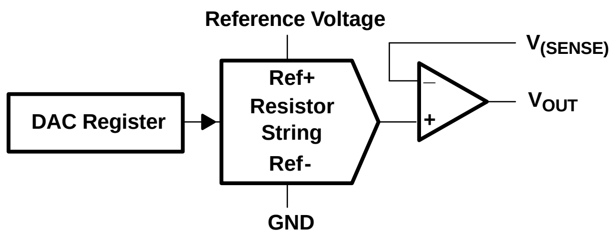


Figure 27. DAC8571 Architecture

The input coding to the DAC8571 is unsigned binary, which gives the ideal output voltage as:

$$V_{OUT} = V_{REF} \times \frac{D}{65536}$$

where D = decimal equivalent of the binary code that is loaded to the DAC register; it can range from 0 to 65535.

RESISTOR STRING

The resistor string section is shown in Figure 28. It is simply a divide-by-two resistor, followed by a string of resistors, each of value R. The code loaded into the DAC register determines at which node on the string the voltage is tapped off to be fed into the output amplifier by closing one of the switches connecting the string to the amplifier. Because it is a string of resistors, it is assured monotonic.

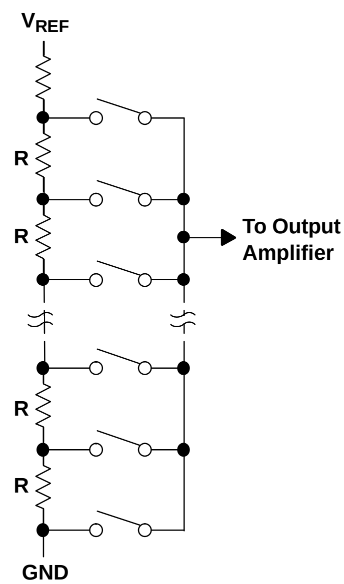


Figure 28. Resistor String.

THEORY OF OPERATION (continued)

Output Amplifier

The output buffer is a gain-of-2 noninverting amplifier capable of generating rail-to-rail voltages at its output, which gives an output range of 0 V to V_{DD}. It is capable of driving a load of 2 kΩ in parallel with 1000 pF to GND. The source and sink capabilities (fast settling) of the output amplifier can be seen in the typical curves. The slew rate is 1 V/µs with a full-scale settling time of 10 µs with the output loaded. The feedback and gain setting resistors of the amplifier are in the order of 50 kΩ. Their absolute value can be off significantly, but they are matched to within 0.1%.

The inverting input of the output amplifier is brought out to the VSENSE pin, through the feedback resistor. This allows for better accuracy in critical applications by tying the VSENSE point and the amplifier output together directly at the load. Other signal conditioning circuitry may also be connected between these points for specific applications including current sourcing.

I^2C Interface

The DAC8571 uses the I^2C interface (see I^2C-Bus Specification Version 2.1, January 2000, Philips Semiconductor) to receive and transmit digital data. I^2C is a 2-wire serial interface that allows multiple devices on the same bus to communicate with each other. The serial bus consists of the serial data (SDA) and serial clock (SCL) lines. Connections to the SDA and SCL lines of the bus are made through open drain IO pins of each device on the bus. Since the devices that connect to the bus have open drain outputs, the bus should include pullup structures. When the bus is not active, both SCL and SDA lines are pulled high by these pullup devices.

The DAC8571 supports the I^2C serial bus and data transmission protocol, in all three defined modes: standard (100 Kbps), fast (400 kBps), and high speed (3.4 Mbps).

I^2C specification states that the device that controls the message is called a *master*, and the devices that are controlled by the master are *slaves*. The master device generates the SCL signal. A master device also generates special timing conditions (start condition, repeated start condition, and stop condition) on the bus to indicate the start or stop of a data transfer. Device addressing is also done by the master. The master device on an I^2C bus is usually a microcontroller or a digital signal processor (DSP). The DAC8571 on the other hand, operates as a slave device on the I^2C bus. A slave device acknowledges master's commands and upon master's control, either receives or transmits data.

I^2C specification states that a device that sends data onto the bus is defined as a transmitter, and a device receiving data from the bus is defined as a *receiver*. DAC8571 normally operates as a slave receiver. A master device *writes* to DAC8571, a slave receiver. However, if a master device inquires DAC8571 internal register data, DAC8571, operates as a slave transmitter. In this case, the master device *reads* from the DAC8571, a slave transmitter. According to I^2C terminology, *read* and *write* are with respect to the master device.

Other than specific timing signals, I^2C interface works with serial bytes. At the end of each byte, a 9^{th} clock cycle is used to generate/detect an acknowledge signal. An acknowledge is when the SDA line is pulled low during the high period of 9^{th} clock cycle. A *not-acknowledge* is when SDA line is left high during the high period of the 9^{th} clock cycle.

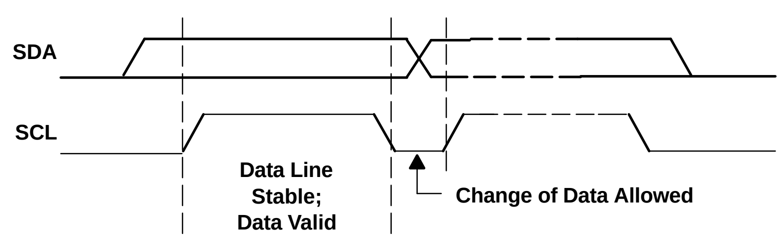


Figure 29. Valid Data

THEORY OF OPERATION (continued)

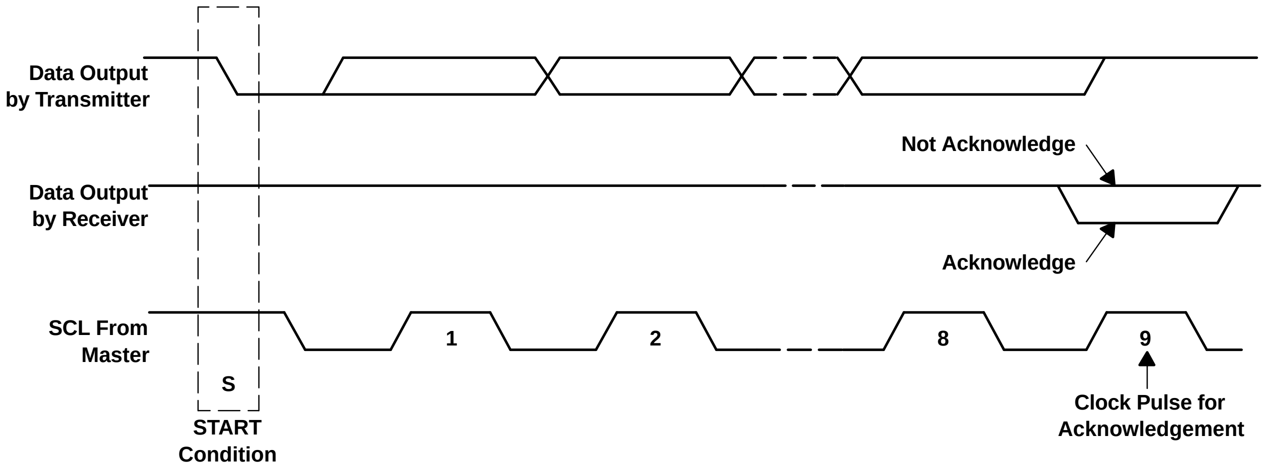


Figure 30. Acknowledge on the I²C Bus

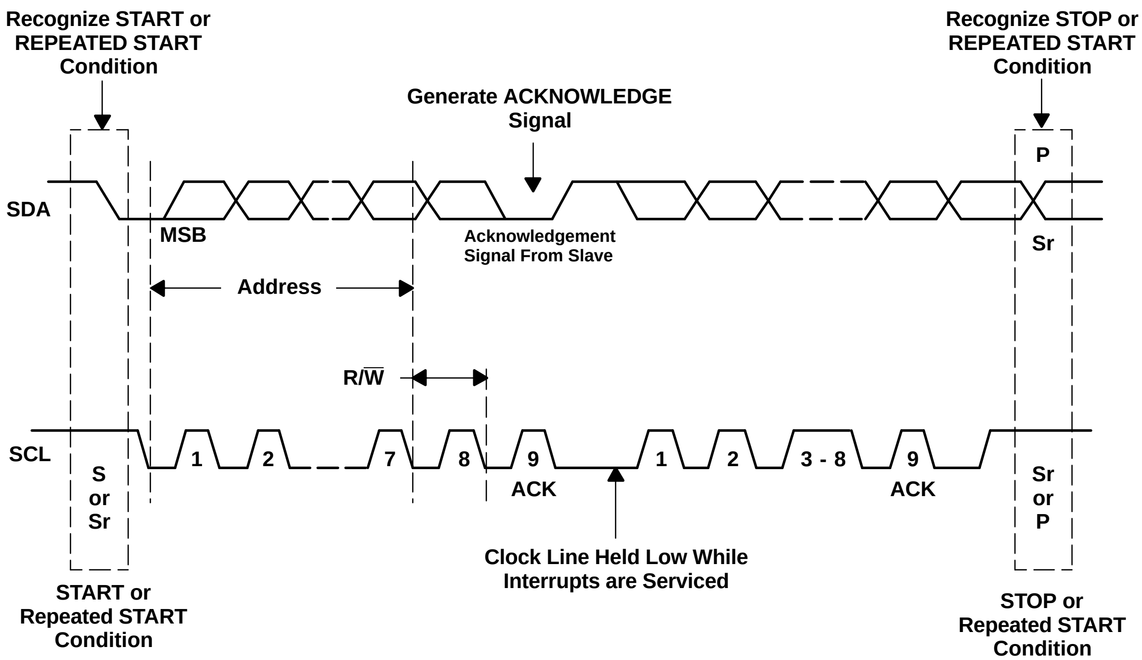


Figure 31. Bus Protocol

Master Writing to a Slave Receiver (Standard/Fast Modes)

I^2C protocol starts when the bus is idle, that is, when SDA and SCL lines are stable high. The master then pulls the SDA line low while SCL is still high indicating that serial data transfer has started. This is called a *start condition*, and can only be asserted by the master. After the start condition, the master generates the serial clock pulses and puts out an address byte, ADDRESS<7:0>. While generating the bit stream, the master ensures the timing for valid data. For each valid I^2C bit, SDA line should remain stable during the entire high period of the SCL line. The address byte consists of 7 address bits (1001100, assuming A0=0) and a direction bit ($R/\overline{W}$=0). After sending the address byte, the master generates a 9^{th} SCL pulse and monitors the state of the SDA line during the high period of this 9^{th} clock cycle. The SDA line being pulled low by a receiver during the high period of 9^{th} clock cycle is called an *acknowledge* signal. If the master receives an acknowledge signal, it knows that a DAC8571 successfully matched the address the master sent. Upon the receipt of this acknowledge, the master knows that the communication link with a DAC8571 has been established and more data could be sent. The master continues by sending a control byte C<7:0>, which sets DAC8571's operation mode. After sending the control byte, the master expects an acknowledge signal. Upon receipt of the acknowledge, the master sends a *most significant byte* M<7:0> that represents the eight most significant bits of DAC8571's 16-bit digital-to-analog conversion data. Upon receipt of the M<7:0>, DAC8571 sends an acknowledge. After receiving the acknowledge, the master sends a *least significant byte* L<7:0> that represents the eight least significant bits of DAC8571's 16-bit conversion data. After receiving the L<7:0>, the DAC8571 sends an acknowledge. At the falling edge of the acknowledge signal following the L<0>, DAC8571 performs a digital to analog conversion. For further DAC updates, the master can keep repeating M<7:0> and L<7:0> sequences, expecting an acknowledge after each byte. After the required number of digital-to-analog conversions is complete, the master can break the communication link with DAC8571 by pulling the SDA line from low to high while SCL line is high. This is called a *stop condition*. A stop condition brings the bus back to idle (SDA and SCL both high). A stop condition indicates that communication with DAC8571 has ended. All devices on the bus including DAC8571 then await a new start condition followed by a matching address byte. DAC8571 stays at its current state upon receipt of a stop condition. Table 1 demonstrates the sequence of events that should occur while a master transmitter is writing to DAC8571.

Table 1. Master Transmitter Writing to Slave Receiver (DAC8571)

Standard/Fast Mode Write Sequence - Data Input									
Transmitter	**MSB**	**6**	**5**	**4**	**3**	**2**	**1**	**LSB**	**Comment**
Master	Start								Begin sequence
Master	1	0	0	1	1	A0	0	R/$\overline{W}$	Write addressing (LSB=0)
DAC8571	DAC8571 Acknowledges								
Master	0	0	Load 1	Load 0	0	Brcsel	0	PD0	Control byte (PD0=0)
DAC8571	DAC8571 Acknowledges								
Master	D15	D14	D13	D12	D11	D10	D9	D8	Writing dataword, high byte
DAC8571	DAC8571 Acknowledges								
Master	D7	D6	D5	D4	D3	D2	D1	D0	Writing dataword, low byte
DAC8571	DAC8571 Acknowledges								
Master	Stop or Repeated Start(1)(2)								Done

(1) High byte, low byte sequence can repeat.
(2) Use repeated start to secure bus operation and loop back to the stage of write addressing for next Write.

Standard/Fast Mode Write Sequence-Power Down Input									
Transmitter	**MSB**	**6**	**5**	**4**	**3**	**2**	**1**	**LSB**	**Comment**
Master	Start								Begin sequence
Master	1	0	0	1	1	A0	0	R/$\overline{W}$	Write addressing (LSB=0)
DAC8571	DAC8571 Acknowledges								
Master	0	0	Load 1	Load 0	0	Brcsel	0	PD0	Control byte (PD0=1)
DAC8571	DAC8571 Acknowledges								
Master	PD1	PD2	PD3	0	0	0	0	0	Writing dataword, high byte
DAC8571	DAC8571 Acknowledges								
Master	0	0	0	0	0	0	0	0	Writing dataword, low byte
DAC8571	DAC8571 Acknowledges								
Master	Stop or Repeated Start(1)(2)								Done

(1) High byte, low byte sequence can repeat.
(2) Use repeated start to secure bus operation and loop back to the stage of write addressing for next Write.

Master Reading From a Slave Transmitter (Standard/Fast Modes)

I^2C protocol starts when the bus is idle, that is, when SDA and SCL lines are stable high. The master then pulls the SDA line low while SCL is still high indicating that serial data transfer has started. This is called a *start condition*, and can only be asserted by the master. After the start condition, the master generates the serial clock pulses and puts out an address byte, ADDRESS<7:0>. While generating the bit stream, the master ensures the timing for valid data. For each valid I^2C bit, SDA line should remain stable during the entire high period of the SCL line. The address byte consists of seven address bits (1001100, assuming A0=0) and a direction bit ($R/\overline{W}$=1). After sending the address byte, the master generates a 9th SCL pulse and monitors the state of the SDA line during the high period of this 9th clock cycle (master leaves the SDA line high). The SDA line being pulled low by a receiver during the high period of 9th clock cycle is called an *acknowledge* signal. If the master receives an acknowledge signal, it knows that a DAC8571 successfully matched the address the master sent. Since the $R/\overline{W}$ bit in the address byte was set, master also knows that DAC8571 is ready to transmit data. Upon the receipt of this acknowledge, the master knows that the communication link with a DAC8571 has been established and more data could be received. The master continues by sending eight clock cycles during which DAC8571 transmits a most significant byte, M<7:0>. If the master detects all bits of the M<7:0> as valid data, it sends an acknowledge signal in the 9th cycle. DAC8571 detects this acknowledge signal and prepares to send more data. Upon the receipt of eight clock cycles from the master, DAC8571 transmits the least significant byte L<7:0>. If the master detects all bits of the L<7:0> as valid data, it sends an acknowledge signal to DAC8571 during the 9th clock cycle. DAC8571 detects this acknowledge signal and prepares to send more data. Upon the receipt of 8 more clock cycles from the master, DAC8571 transmits the control byte C<7:0>. During the 9th clock cycle, the master transmits a not-acknowledge signal to DAC8571 and terminates the sequence with a stop condition, by pulling the SDA line from low to high while clock is high. M<7:0> and L<7:0> data could be either DAC data or could be the data stored in the temporary register. Bits in the C<7:0> reveal this information. Table 2 demonstrates the sequence of events that should occur while a master receiver is reading from DAC8571.

Table 2. Master Receiver Reads From Slave Transmitter (DAC8571)

Standard/Fast Mode Read Sequence-Data Transmit									
Transmitter	**MSB**	**6**	**5**	**4**	**3**	**2**	**1**	**LSB**	**Comment**
Master	Start								Begin sequence
Master	1	0	0	1	1	A0	0	$R/\overline{W}$	Read addressing ($R/\overline{W}$ = 1)
DAC8571	DAC8571 Acknowledges								
DAC8571	D15	D14	D13	D12	D11	D10	D9	D8	High byte
Master	Master Acknowledges								
DAC8571	D7	D6	D5	D4	D3	D2	D1	D0	Low byte
Master	Master Acknowledges								
DAC8571	C7	C6	C5	C4	C3	C2	C1	C0	Control byte
Master	Master Not Acknowledges								Master signal end of read
Master	Stop or Repeated Start								Done

Master Writing to a Slave Receiver (High-Speed Mode)

All devices must start operation in standard/fast mode and switch to high-speed mode using a well defined protocol. This is required because high-speed mode requires the on chip filter settings of each I^2C device (for SDA and SCL lines) to be switched to support 3.4 Mbps operation. A stop condition always ends the high speed mode and puts all devices back to standard/fast mode.

I^2C protocol starts when the bus is idle, that is, when SDA and SCL lines are stable high. The master then pulls the SDA line low while SCL is still high indicating that serial data transfer has started. This is called a *start condition*, and can only be asserted by the master. After the start condition, the master device puts out the high-speed master code 0000 1xxx. No device is allowed to acknowledge the master code, but the devices are required to switch their internal settings to support 3.4 Mbps operation upon the receipt of this code. After the not-acknowledge signal, the master is allowed to operate at high speed. Now at much higher speed, the master generates a repeated start condition. After the start condition, master generates the serial clock pulses and puts out an address byte, ADDRESS<7:0>. While generating the bit stream, the master ensures the timing for valid data. For each valid I^2C bit, SDA line should remain stable during the entire high period of the SCL line. The address byte consists of seven address bits and a direction bit ($R/\overline{W}$=0). After sending the address byte, the master generates a 9th SCL pulse and monitors the state of the SDA line during the high period of this 9th clock cycle (master leaves the SDA line high). The SDA line being pulled low by the receiver during the high period of 9th clock cycle is called an *acknowledge* signal. If the master receives an acknowledge signal, it knows that a DAC8571 successfully matched the address the master sent. Upon the receipt of this acknowledge, the master knows that the high-speed communication link with a DAC8571 has been established and more data could be sent. The master continues by sending a control byte, C<7:0>, which sets DAC8571 operation mode. After sending the control byte, master expects an acknowledge. Upon the receipt of an acknowledge, the master sends a *most significant byte*, M<7:0> that represents the eight most significant bits of DAC8571's 16-bit digital-to-analog conversion data. Upon the receipt of the M<7:0>, DAC8571 sends an acknowledge. After receiving the acknowledge, the master sends a *least significant byte,* L<7:0>, that represents the eight least significant bits of DAC8571's 16-bit conversion data. After receiving the L<7:0>, the DAC8571 sends an acknowledge. At the falling edge of the acknowledge signal following the L<0>, DAC8571 performs a digital to analog conversion, depending on the operational mode. For further DAC updates, the master can keep repeating M<7:0> and L<7:0> sequences, expecting an acknowledge after each byte. After the required number of digital to analog conversions is complete, the master can break the communication link with DAC8571 by pulling the SDA line from low to high while SCL line is high. This is called a *stop condition*. A stop condition brings the bus back to idle (SDA and SCL both high). A stop condition indicates that communication with a device (DAC8571) has ended. All devices on the bus including DAC8571 then await a new start condition followed by a matching address byte. DAC8571 stays at its current state upon the receipt of a stop condition. A stop condition during the high-speed mode also indicates the end of the high-speed mode. Table 3 demonstrates the sequence of events that should occur while a master transmitter is writing to DAC8571 in I^2C high-speed mode.

Table 3. Master Transmitter Writes to Slave Receiver in High-Speed Mode

HS Mode Write Sequence-Data Input									
Transmitter	**MSB**	**6**	**5**	**4**	**3**	**2**	**1**	**LSB**	**Comment**
Master	Start								Begin sequence(1)
Master	0	0	0	0	1	X	X	X	HS mode master code
NONE	Not Acknowledge								No device may acknowledge HS master code
Master	Repeated Start								
Master	1	0	0	1	1	A0	0	$R/\overline{W}$	Write addressing (LSB = 0)
DAC8571	DAC8571 Acknowledges								
Master	0	0	Load 1	Load 0	0	Brcsel	0	PD0	Control byte (PD0=0)
DAC8571	DAC8571 Acknowledges								
Master	D15	D14	D13	D12	D11	D10	D9	D8	Writing dataword, high byte
DAC8571	DAC8571 Acknowledges								
Master	D7	D6	D5	D4	D3	D2	D1	D0	Writing dataword, low byte
DAC8571	DAC8571 Acknowledges								
Master	Stop or Repeated Start(2)								Done

(1) High-byte, low-byte sequences can repeat
(2) Use repeated start to secure bus operation and loop back to the stage of write addressing for next Write.

Master Receiver Reading From a Slave Transmitter (High-Speed Mode)

I^2C protocol starts when the bus is idle, that is, when SDA and SCL lines are stable high. The master then pulls the SDA line low while SCL is still high indicating that serial data transfer has started. This is called a *start condition*, and can only be asserted by the master. After the start condition, the master device puts out the high-speed master code 0000 1xxx. No device is allowed to acknowledge the master code, but the devices are required to switch their internal settings to support 3.4 Mbps operation upon the receipt of this code. After the not-acknowledge signal, the master is allowed to operate at high speed. Now at much higher speed, the master generates a repeated start condition. After the start condition, the master generates the serial clock pulses and puts out an address byte, ADDRESS<7:0>. While generating the bit stream, the master ensures the timing for valid data. For each valid I^2C bit, SDA line should remain stable during the entire high period of the SCL line. The address byte consists of seven address bits and a direction bit ($R/\overline{W}$=1). After sending the address byte, the master generates a 9^{th} SCL pulse and monitors the state of the SDA line during the high period of this 9^{th} clock cycle (master leaves the SDA line high). The SDA line being pulled low by the receiver during the high period of 9^{th} clock cycle is called an *acknowledge* signal. If the master receives an acknowledge signal, it knows that a DAC8571 successfully matched the address the master sent. Since the R/W bit in the address byte was set, master also knows that DAC8571 is ready to transmit data. Upon the receipt of this acknowledge, the master knows that the communication link with a DAC8571 has been established and more data could be received. The master continues by sending eight clock cycles during which DAC8571 transmits an M<7:0>. If the master detects all bits of the M<7:0> as valid data, it sends an acknowledge signal in the 9^{th} cycle. DAC8571 detects this acknowledge signal and prepares to send more data. Upon the receipt of eight more clock cycles from the master, DAC8571 transmits L<7:0>. If the master detects all bits of the L<7:0> as valid data, it sends an acknowledge signal to DAC8571 during the 9th clock cycle. DAC8571 detects this acknowledge signal and prepares to send more data. Upon the receipt of eight more clock cycles from the master, DAC8571 transmits the control byte, C<7:0>. In the 9th clock cycle the master transmits a not-acknowledge signal to DAC8571 and terminates the sequence with a stop condition, by pulling the SDA line from low to high while clock is high. M<7:0> and L<7:0> data could be either DAC data or could be the data stored in the temporary register. Bits in the C<7:0> reveal this information. A stop condition during the high-speed mode also indicates the end of the high-speed mode. Table 4 demonstrates the sequence of events that should occur while a master receiver is reading from DAC8571 in I^2C high-speed mode.

Table 4. Master Receiver Reads Data From Slave Transmitter in High-Speed Mode

HS Mode Read Sequence-Data Transmit									
Transmitter	**MSB**	**6**	**5**	**4**	**3**	**2**	**1**	**LSB**	**Comment**
Master	Start								Begin sequence
Master	0	0	0	0	1	X	X	X	HS Mode master code
NONE	Not Acknowledge								No device may acknowledge HS master code
Master	Repeated Start								
Master	1	0	0	1	1	A0	0	$R/\overline{W}$	Read addressing ($R/\overline{W}$=1)
DAC8571	DAC8571 Acknowledges								
DAC8571	D15	D14	D13	D12	D11	D10	D9	D8	High byte
Master	Master Acknowledges								
DAC8571	D7	D6	D5	D4	D3	D2	D1	D0	Low byte
Master	Master Acknowledges								
DAC8571	C7	C6	C5	C4	C3	C2	C1	C0	Control byte
Master	Master Not Acknowledges								Master signal end of read
Master	Stop or Repeated Start(1)								Done

(1) Use repeated start to secure bus operation and loop back to the stage of write addressing for next Write.

DAC8571 Update Sequence

DAC8571 requires a start condition, a valid I^2C address, a control byte, an MS byte and an LS byte for an update. The control byte sets the operational mode of the DAC8571. After the receipt of the control byte, DAC8571 expects an MS byte and an LS byte. After the receipt of each byte, DAC8571 acknowledges by pulling the SDA line low. At the falling edge of the acknowledge signal that follows the LS byte, DAC8571 performs an update.

After the first update, further data can be sent as MS byte and LS byte sequences and DAC8571 keeps updating at the falling edge of the acknowledge signal that follows each LS byte. The bits of the last control byte determine the type of update being performed. Thus, for the first update, DAC8571 requires a start condition, a valid I^2C address, a control byte, an MS byte and an LS byte. For all consecutive updates, DAC8571 needs an MS byte and an LS byte.

Using the I^2C high-speed mode, the clock running a 3.4 MHz, each 16-bit DAC update can be done within 18-clock cycles (MS byte, acknowledge bit, LS byte, acknowledge bit), at 188.88 KSPS. Using the fast mode, clock running at 400 kHz, maximum DAC update rate is limited to 22.22 KSPS.

DAC8571 Address Byte

MSB							LSB
1	0	0	1	1	A0	0	$R/\overline{W}$

The address byte is the first byte received following a START condition from the master device. The first five bits (MSBs) of the slave address are factory preset to 10011. The next bit of the address byte is the device select bit A0, followed by a fixed 0 and the read/write direction bit $R/\overline{W}$. In order for DAC8571 to respond, the 7-bit address should be 10011A00, where the state of the A0 bit matches the state of the A0 pin. A maximum of two DAC8571 devices with the same preset code can therefore be connected on the same bus at one time. The A0 Address inputs can be permanently connected to VDD or digital ground, or can be actively driven by TTL or CMOS logic levels. The device address is set by the state of these pins upon power up of the DAC8571. The last bit of the address byte ($R/\overline{W}$) defines the direction of the data flow. When set to a 1, a read operation is selected (master device reads from DAC8571); when set to a 0, a write operation is selected (master device writes to DAC8571). Following the START condition, the DAC8571 monitors the SDA bus, checking the device address being transmitted. Upon receiving the 10011A00 code, and the R/W bit, the DAC8571 outputs an acknowledge signal on the SDA line.

Broadcast addressing is also supported by DAC8571. Broadcast addressing can be used for synchronously updating or powering down multiple DAC8571 devices on the same bus. DAC8571 is designed to work with other members of DAC857x, DAC757x families to support multichannel synchronous update. When broadcast addressing is used, DAC8571 responds regardless of the state of the A0 pin. Broadcast address is only valid for write operation and cannot be used for read operation. Broadcast address is as follows.

MSB							LSB
1	0	0	1	0	0	0	0

Control Byte

After transmitting an acknowledge pulse following a valid address, DAC8571 expects a control byte C<7:0>. Control byte functionality is shown in Table 5.

The first two MSBs C<7> and C<6> of the control byte must be zeroes for DAC8571 to update. If these two bits are not assigned to zero, DAC8571 ignores all update commands, but still generates an acknowledge signal.

C<5> and C<4> are used for setting the update mode. Some of these modes are designed to support multichannel synchronous operation between multiple devices.

- C<5>=0, C<4>=0: Store I^2C data. The contents of MS byte and LS byte data (or power-down information) are stored into the temporary register. This mode does not change the DAC output.
- C<5>=0, C<4>=1: Update DAC with I^2C data. Most common mode. The contents of MS byte and LS byte data (or power-down information) are stored into the temporary data register and into the DAC register. This mode changes the DAC output with the contents of I^2C MS byte and LS byte data.

- C<5>=1, C<4>=0: Update with previously stored data. The contents of MS byte and LS byte data (or power-down information) are ignored. The DAC is updated with the contents of the data previously stored in the temporary register. This mode changes the DAC output.
- C<5>=1, C<4>=1: Broadcast update, If C<2>=0, DAC is updated with the contents of its temporary register. If C<2>=1, DAC is updated with I^2C MS byte and LS byte data. C<7> and C<6> do not have to be zeroes in order for DAC8571 to update. This mode is intended to help DAC8571 work with other DAC857x and DAC757x devices for multichannel synchronous update applications.

C<3> should always be zero.

C<2> is utilized only when C<5>=C<4>=1. Otherwise, C<2> must be assigned to zero.

C<1> should always be zero.

C<0> should be zero during normal DAC operation. C<0>=1 is a power-down flag. If C<0>=1, M<7>, M<6>, and M<5> indicate a powerdown operation as shown in Table 6.

Table 5. Control Byte Functionality

C<7>	C<6>	C<5>	C<4>	C<3>	C<2>	C<1>	C<0>	M<7>	M<6>	M<5>	
		Load1	Load0		Brcsel		PD0	MSB	MSB-1	MSB-2...LSB	**DAC8571 FUNCTION**
0	0	0	0	0	0	0	0	Data			Write temporary register with data
0	0	0	0	0	0	0	1	See Table 6			Write temporary register with power down command
0	0	0	1	0	0	0	0	Data			Write temporary register and load DAC with data
0	0	0	1	0	0	0	1	See Table 6			Power down DAC
0	0	1	0	0	0	0	0	x			Update DAC with temporary register data or power down
Broadcast Commands											
x	x	1	1	x	0	x	x	x			Load all DACs, all devices with temporary register data
x	x	1	1	x	1	x	0	Data			Load all DACs, all devices with data
x	x	1	1	x	1	x	1	See Table 6			Power down all DACs, all devices

Most Significant Byte

Most Significant Byte M<7:0> consists of 8 most significant bits of D/A conversion data. When C<0>=1. M<7>, M<6>, M<5> indicate a powerdown operation as shown in Table 6.

Least Significant Byte

Least Significant Byte L<7:0> consists of the 8 least significant bits of D/A conversion data. DAC8571 updates at the falling edge of the acknowledge signal that follows the L<0> bit.

Data Transmit and Read-Back

I^2C bus can be noisy and data integrity and can be a problem in a system of many I^2C devices. To enable I^2C system verification, DAC8571 provides read back capability for the user. During read back operation, the contents of the control byte, MS byte and the LS byte can be sent back to the master device using the I^2C bus. This read-back function is also useful if a device on the I^2C bus inquires DAC8571 data.

For read-back operation, the master device sends the I^2C address and sets the R/$\overline{W}$ bit. DAC8571 acknowledges. Then, upon the receipt of clock pulses from the master, DAC8571 sends the MS byte. If the master acknowledges, DAC8571 sends the LS byte. If the master acknowledges, DAC8571 sends the control byte. This sequence is interrupted by the master sending a not acknowledge signal.

Depending on the contents of the control byte transmitted by the DAC8571, the MS byte and LS byte information (transmitted by the DAC8571) is interpreted as follows:

C<5>	C<4>	C<2>	
0	0	0	MS and LS bytes represent temporary register data

C<5>	C<4>	C<2>	
0	1	0	MS and LS bytes represent temporary and DAC register data
1	0	0	MS and LS bytes represent I²C data that is discarded
1	1	0	MS and LS bytes represent I²C data that is discarded
1	1	1	MS and LS bytes represent temporary and DAC register data

EXAMPLES (A0 TIED TO GND, VDD = 5 V)

EXAMPLE 1: Write 1/4 scale to DAC8571									
	ADDRESS <7...0>		C<7...0>		M<7...0>		L<7...0>		
START	1001 1000	ACK	0001 0000	ACK	0100 0000	ACK	0000 0000	ACK	STOP
Previous output voltage is valid								Vout = 1.25 V	
EXAMPLE 2: Switch DAC8571 to fast settling mode									
	ADDRESS <7...0>		C<7...0>		M<7...0>		L<7...0>		
START	1001 1000	ACK	0001 0001	ACK	0010 0000	ACK	0000 0000	ACK	STOP
Previous output voltage is valid								Vout = 0 V	
EXAMPLE 3: Switch DAC8571 back to low power mode									
	ADDRESS <7...0>		C<7...0>		M<7...0>		L<7...0>		
START	1001 1000	ACK	0001 0001	ACK	0000 0000	ACK	0000 0000	ACK	STOP
Previous output voltage is valid								Vout = 0 V	
EXAMPLE 4: Power-down DAC8571 with Hi-Z output									
	ADDRESS <7...0>		C<7...0>		M<7...0>		L<7...0>		
START	1001 1000	ACK	0001 0001	ACK	1100 0000	ACK	0000 0000	ACK	STOP
Previous output voltage is valid								Vout = Hi-Z	
EXAMPLE 5: Power-down DAC8571 with 1K output impedance to ground									
	ADDRESS <7...0>		C<7...0>		M<7...0>		L<7...0>		
START	1001 1000	ACK	0001 0001	ACK	0100 0000	ACK	0000 0000	ACK	STOP
Previous output voltage is valid								Vout = 0 V	
EXAMPLE 6: Power-down DAC8571 with 100K output impedance to ground									
	ADDRESS <7...0>		C<7...0>		M<7...0>		L<7...0>		
START	1001 1000	ACK	0001 0001	ACK	1000 0000	ACK	0000 0000	ACK	STOP
Previous output voltage is valid								Vout = 0 V	
EXAMPLE 7: Store full scale data in temporary register									
	ADDRESS <7...0>		C<7...0>		M<7...0>		L<7...0>		
START	1001 1000	ACK	0000 0000	ACK	1111 1111	ACK	1111 1111	ACK	STOP
Previous output voltage is valid									
EXAMPLE 8: Update DAC8571 with the data previously stored in the temporary register									
	ADDRESS <7...0>		C<7...0>		M<7...0>		L<7...0>		
START	1001 1000	ACK	0010 0000	ACK	XXXX XXXX	ACK	XXXX XXXX	ACK	STOP
Previous output voltage is valid								New Vout valid	
EXAMPLE 9: Broadcast a powerdown command to all DAC8571s on the I²C bus									
	ADDRESS <7...0>		C<7...0>		M<7...0>		L<7...0>		
START	1001 0000	ACK	0011 0101	ACK	1100 0000	ACK	0000 0000	ACK	STOP
Previous output voltage is valid								Vout = Hi-Z	
EXAMPLE 10: Broadcast update. All DAC8571s on the I²C bus update synchronously with the contents of their temporary registers									
	ADDRESS <7...0>		C<7...0>		M<7...0>		L<7...0>		
START	1001 0000	ACK	0011 0000	ACK	XXXX XXXX	ACK	XXXX XXXX	ACK	STOP
Previous output voltage is valid								New Vout valid	

EXAMPLE 11: Read back DAC8571 internal data. V denotes valid logic.									
START	ADDRESS<7...0> 1001 1001	ACK	M<7...0> VVVV VVVV	MASTER ACK	L<7...0> VVVV VVVV	MASTER ACK	C<7...0> VVVV VVVV	MASTER NOT ACK	STOP

EXAMPLE 12: Ramp generation in high speed mode (up to code 7 is shown)							
START	HS Master Code 0000 1000	NOT ACK	REPEATED START	ADDRESS 1001 1000	ACK	C<7...0> 0001 0000	ACK
Previous Vout voltage valid							
MSB<7...0> 0000 0000	ACK	LSB<7...0> 0000 0000	ACK	MSB<7...0> 0000 0000	ACK	LSB<7...0> 0000 0001	ACK
Previous Vout voltage valid			Vout = 0 V				Vout = 76 µV
MSB<7...0> 0000 0000	ACK	LSB<7...0> 0000 0010	ACK	MSB<7...0> 0000 0000	ACK	LSB<7...0> 0000 0011	ACK
Vout = 76 µV			Vout = 2 ×76 µV				Vout = 3 ×76 µV
MSB<7...0> 0000 0000	ACK	LSB<7...0> 0000 0100	ACK	MSB<7...0> 0000 0000	ACK	LSB<7...0> 0000 0101	ACK
Vout = 3 ×76 µV			Vout = 4 ×76 µV				Vout = 5 ×76 µV
MSB<7...0> 0000 0000	ACK	LSB<7...0> 0000 0110	ACK	MSB<7...0> 0000 0000	ACK	LSB<7...0> 0000 0111	ACK
Vout = 5 ×76 µV			Vout = 6 ×76 µV				Vout = 7 ×76 µV

Power-On Reset

The DAC8571 contains a power-on-reset circuit that controls the output voltage during power-up. On power-up, the DAC register is filled with zeros and the output voltage is 0V; it remains there until a valid write sequence is made to the DAC. This is useful in applications where it is important to know the state of the output of the DAC while it is in the process of powering up. No input is brought high before the power is applied.

Power-Down Modes

The DAC8571 contains five separate power settings. These modes are programmable when C<0>=1. When C<0>=1, M<7>, M<6>, and M<5> bits represent power setting control bits, and M<4...0> and L<7...0> are assigned to zeroes. Power setting of DAC8571 is updated at the falling edge of the acknowledge signal that follows the least significant byte. To set the power consumption of the device, following I^2C sequence is used.

Start_condition ->	
Valid_address	(1001 1000) -> ack
C<7:0>	(0001 0001) -> ack
M<7:0>	(vvv0 0000) -> ack
L<7:0>	(0000 0000) -> ack
Stop_condition	

Table 6. Power Settings for the DAC8571 (C<0>=1)

M<7>	M<6>	M<5>	Operating Mode
0	0	0	Low power mode, default
0	0	1	Fast settling mode
0	1	X	PWD. 1kΩ to GND
1	0	X	PWD. 100 kΩ to GND
1	1	X	PWD. Output Hi-Z

After power-up, the device works in low power mode with its normal power consumption of 170 µA at 5 V. At fast settling mode, device consumes 250 µA nominally, but settles in 10 µs. For the three power-down modes, the supply current falls to 200 nA at 5 V (50 nA at 3 V). Not only does the supply current fall but the output stage is also internally switched from the output of the amplifier to a resistor network of known values. This has the advantage that the output impedance of the device is known while in power-down mode. There are three different options: The output is connected internally to GND through a 1-kΩ resistor, a 100-kΩ resistor or it is left open-circuit (high impedance). The output stage is illustrated in Figure 32.

A power on reset starts the DAC8571 in the low power mode. Low power mode and fast-settling mode settings stay unchanged during DAC8571 data updates, unless they are specifically overwritten as explained in Table 6. On the other hand, each new data sequence requiring a DAC update brings the DAC8571 out of the three power-down conditions.

DAC8571 power settings can be stored in the temporary register, just like data (use C<7:0> = 0000 0001). This allows simultaneous powerdown capability for multichannel applications.

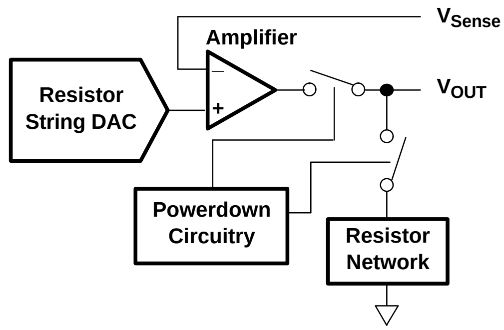


Figure 32. Output Stage During Power-Down

All linear circuitry is shut down when the power-down mode is activated. However, the contents of the DAC register are unaffected when in power-down. The time to exit power-down is typically 2.5 µs for V_{DD} = 5 V and 5 µs for V_{DD} = 3 V. (See the Typical Characteristics section for additional information.)

CURRENT CONSUMPTION

In the low power mode, the DAC8571 typically consumes 170 µA at V_{DD} = 5 V and 150 µA at V_{DD} = 3 V including reference current consumption. Fast settling mode adds 80 µA of current consumption, but ensures 10-µs settling. Additional current consumption can occur at the digital inputs if VIH<<VDD. For most efficient power operation, CMOS logic levels are recommended at the digital inputs to the DAC. In power-down mode, typical current consumption is 200 nA.

DRIVING RESISTIVE AND CAPACITIVE LOADS

The DAC8571 output stage is capable of driving loads of up to 1000 pF while remaining stable. Within the offset and gain error margins, the DAC8571 can operate rail-to-rail when driving a capacitive load. Resistive loads of 2 kΩ can be driven by the DAC8571 while achieving a very good load regulation. Load regulation error increases when the DAC output voltage is close to supply rails. When the outputs of the DAC are driven to the positive rail under resistive loading, the PMOS transistor of each Class-AB output stage can enter into the linear region. When this occurs, the added IR voltage drop deteriorates the linearity performance of the DAC. This only occurs within approximately the top 20 mV of the DAC's digital input-to-voltage output transfer characteristic. The reference voltage applied to the DAC8571 may be reduced below the supply voltage applied to VDD in order to eliminate this condition if good linearity is a requirement at full scale (under resistive loading conditions).

AC PERFORMANCE

DAC8571 can achieve typical ac performance of 96-dB signal-to-noise ratio (SNR) and 65-dB total harmonic distortion (THD), making the DAC8571 a solid choice for applications requiring low SNR at output frequencies at or below 4 kHz.

OUTPUT VOLTAGE STABILITY

The DAC8571 exhibits excellent temperature stability of 5 ppm/°C typical output voltage drift over the specified temperature range of the device. This enables the output voltage of each channel to stay within a ±25 µV window for a ±1°C ambient temperature change. Good power supply rejection ratio (PSRR) performance reduces supply noise present on V_{DD} from appearing at the outputs to well below 10 µV. Combined with good dc noise performance and true 16-bit differential linearity, the DAC8571 becomes a perfect choice for closed-loop control applications.

SETTLING TIME AND OUTPUT GLITCH PERFORMANCE

Settling time to within the 16-bit accurate range of the DAC8571 is achievable within 10 µs for a full-scale code change at the input. Worst case settling times between consecutive code changes is typically less than 2 µs, therefore, the update rate is limited by the I^2C interface for digital input signals changing code-to-code. For full-scale output swings, the output stage of each DAC8571 channel typically exhibits less than 100-mV overshoot and undershoot when driving a 200-pF capacitive load. Code-to-code change glitches are extremely low (~10µV) given that the code-to-code transition does not cross an Nx4096 code boundary. Due to internal segmentation of the DAC8571, code-to-code glitches occur at each crossing of an Nx4096 code boundary. These glitches can approach 100 mVs for N = 15, but settle out within ~2 µs.

USING REF02 AS A POWER SUPPLY FOR DAC8571

Due to the extremely low supply current required by the DAC8571, a possible configuration is to use a REF02 5-V precision voltage reference to supply the required voltage to the DAC8571's supply input as well as the reference input, as shown in Figure 33. This is especially useful if the power supply is quite noisy or if the system supply voltages are at some value other than 5 V. The REF02 outputs a steady supply voltage for the DAC8571. If the REF02 is used, the current it needs to supply to the DAC8571 is 160-µA typical and 225-µA max for V_{DD} = 5 V. When a DAC output is loaded, the REF02 also needs to supply the current to the load. The total typical current required (with a 5-kΩ load on a given DAC output) is:

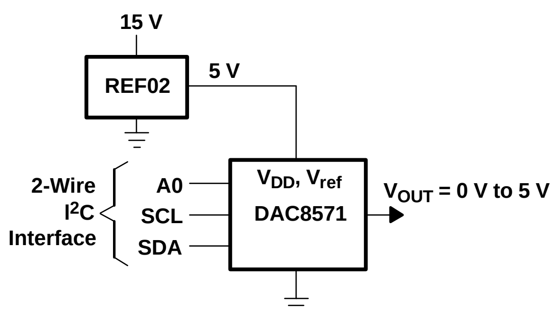


Figure 33. REF02 as a Power Supply

$$160\ \mu A + \frac{5\ V}{5\ k\Omega} = 1.16\ mA$$

The load regulation of the REF02 is typically 0.005%/mA, which results in an error of 290 µV for a 1.16-mA current drawn. This corresponds to a 3.82 LSB error for a 0-V to 5-V output range.

LAYOUT

A precision analog component requires careful layout, adequate bypassing, and clean, well-regulated power supplies.

The power applied to V_{DD} and V_{REF} should be well regulated and low noise. Switching power supplies and dc/dc converters often has high-frequency glitches or spikes riding on the output voltage. In addition, digital components can create similar high-frequency spikes as their internal logic switches states. This noise easily couples into the DAC output voltage through various paths between the power connections and analog output.

As with the GND connection, V_{DD} is connected to a +5-V power supply plane or trace that is separate from the connection for digital logic until they are connected at the power entry point. In addition, the 1-μF to 10-μF, and 0.1-μF bypass capacitors are strongly recommended. In some situations, additional bypassing may be required, such as a 100-μF electrolytic capacitor or even a Pi filter made up of inductors and capacitors—all designed to essentially lowpass filter the 5-V supply, removing the high frequency noise.

PACKAGING INFORMATION

Orderable Device	Status (1)	Package Type	Package Drawing	Pins	Package Qty	Eco Plan (2)	Lead finish/ Ball material (6)	MSL Peak Temp (3)	Op Temp (°C)	Device Marking (4/5)	Samples
DAC8571IDGK	ACTIVE	VSSOP	DGK	8	80	RoHS & Green	Call TI | NIPDAUAG	Level-2-260C-1 YEAR	-40 to 105	D871	Samples
DAC8571IDGKG4	ACTIVE	VSSOP	DGK	8	80	RoHS & Green	Call TI	Level-2-260C-1 YEAR	-40 to 105	D871	Samples
DAC8571IDGKR	ACTIVE	VSSOP	DGK	8	2500	RoHS & Green	Call TI | NIPDAUAG	Level-2-260C-1 YEAR	-40 to 105	D871	Samples
DAC8571IDGKRG4	ACTIVE	VSSOP	DGK	8	2500	RoHS & Green	Call TI	Level-2-260C-1 YEAR	-40 to 105	D871	Samples

(1) The marketing status values are defined as follows:
ACTIVE: Product device recommended for new designs.
LIFEBUY: TI has announced that the device will be discontinued, and a lifetime-buy period is in effect.
NRND: Not recommended for new designs. Device is in production to support existing customers, but TI does not recommend using this part in a new design.
PREVIEW: Device has been announced but is not in production. Samples may or may not be available.
OBSOLETE: TI has discontinued the production of the device.

(2) **RoHS:** TI defines "RoHS" to mean semiconductor products that are compliant with the current EU RoHS requirements for all 10 RoHS substances, including the requirement that RoHS substance do not exceed 0.1% by weight in homogeneous materials. Where designed to be soldered at high temperatures, "RoHS" products are suitable for use in specified lead-free processes. TI may reference these types of products as "Pb-Free".
RoHS Exempt: TI defines "RoHS Exempt" to mean products that contain lead but are compliant with EU RoHS pursuant to a specific EU RoHS exemption.
Green: TI defines "Green" to mean the content of Chlorine (Cl) and Bromine (Br) based flame retardants meet JS709B low halogen requirements of <=1000ppm threshold. Antimony trioxide based flame retardants must also meet the <=1000ppm threshold requirement.

(3) MSL, Peak Temp. - The Moisture Sensitivity Level rating according to the JEDEC industry standard classifications, and peak solder temperature.

(4) There may be additional marking, which relates to the logo, the lot trace code information, or the environmental category on the device.

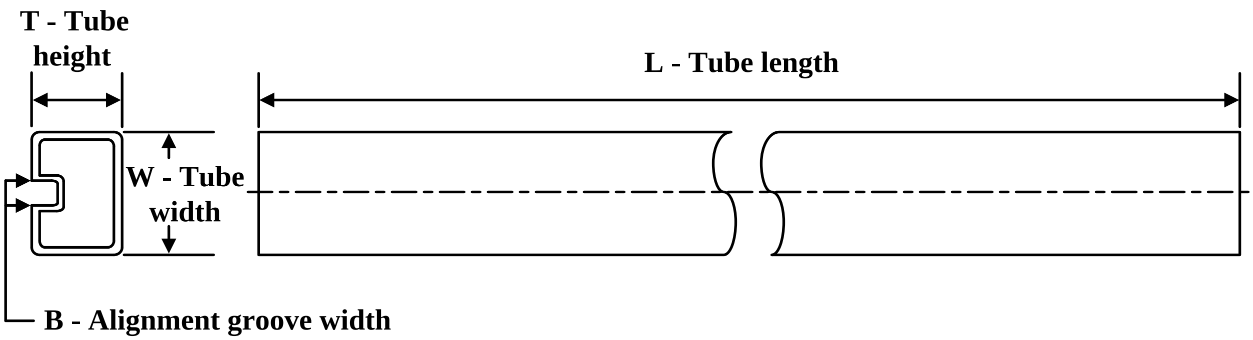
(5) Multiple Device Markings will be inside parentheses. Only one Device Marking contained in parentheses and separated by a "~" will appear on a device. If a line is indented then it is a continuation of the previous line and the two combined represent the entire Device Marking for that device.

(6) Lead finish/Ball material - Orderable Devices may have multiple material finish options. Finish options are separated by a vertical ruled line. Lead finish/Ball material values may wrap to two lines if the finish value exceeds the maximum column width.

Important Information and Disclaimer: The information provided on this page represents TI's knowledge and belief as of the date that it is provided. TI bases its knowledge and belief on information provided by third parties, and makes no representation or warranty as to the accuracy of such information. Efforts are underway to better integrate information from third parties. TI has taken and continues to take reasonable steps to provide representative and accurate information but may not have conducted destructive testing or chemical analysis on incoming materials and chemicals. TI and TI suppliers consider certain information to be proprietary, and thus CAS numbers and other limited information may not be available for release.

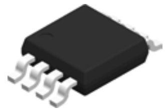
In no event shall TI's liability arising out of such information exceed the total purchase price of the TI part(s) at issue in this document sold by TI to Customer on an annual basis.

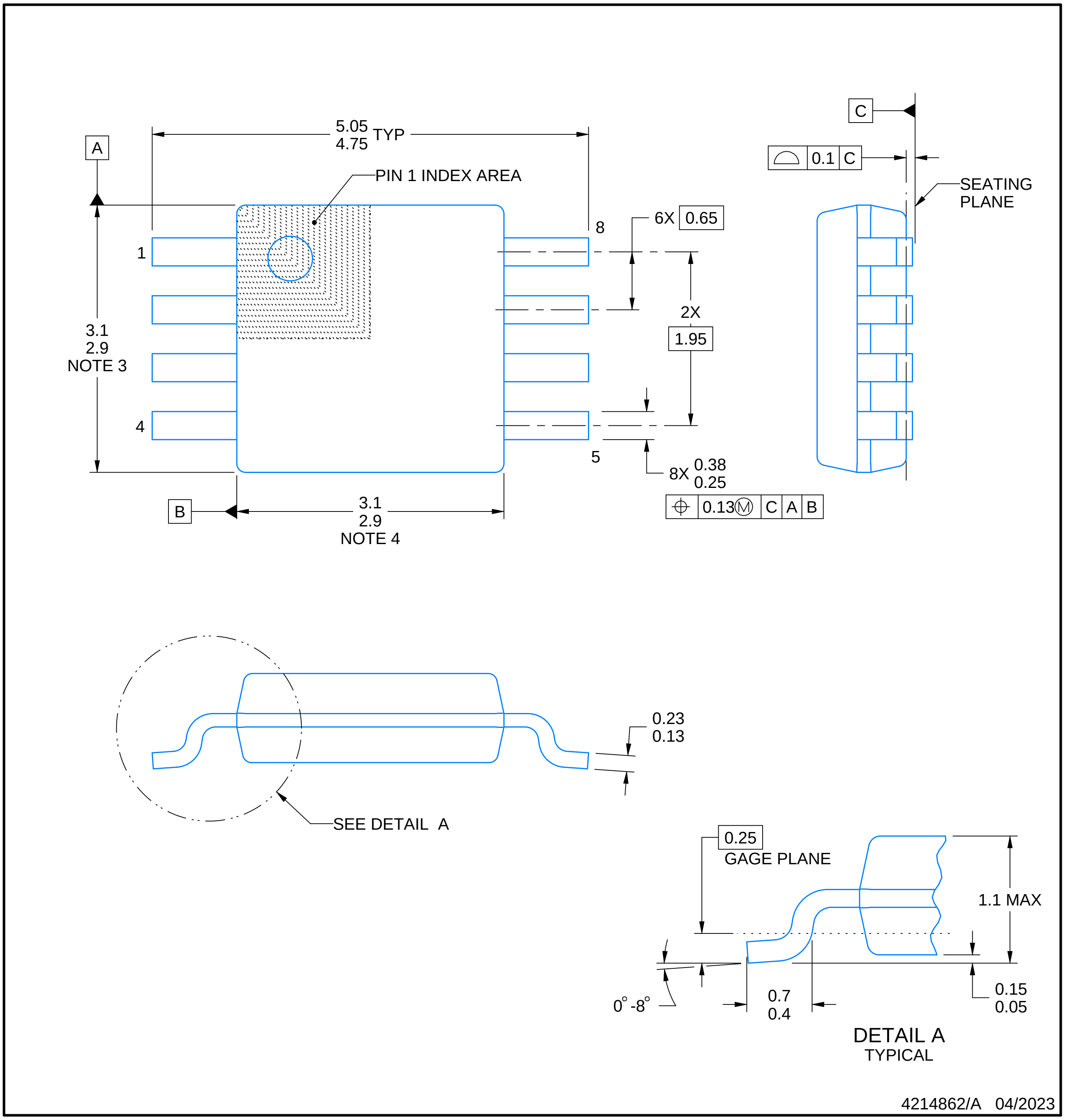
TUBE



*All dimensions are nominal

Device	Package Name	Package Type	Pins	SPQ	L (mm)	W (mm)	T (µm)	B (mm)
DAC8571IDGK	DGK	VSSOP	8	80	331.47	6.55	3000	2.88
DAC8571IDGKG4	DGK	VSSOP	8	80	331.47	6.55	3000	2.88

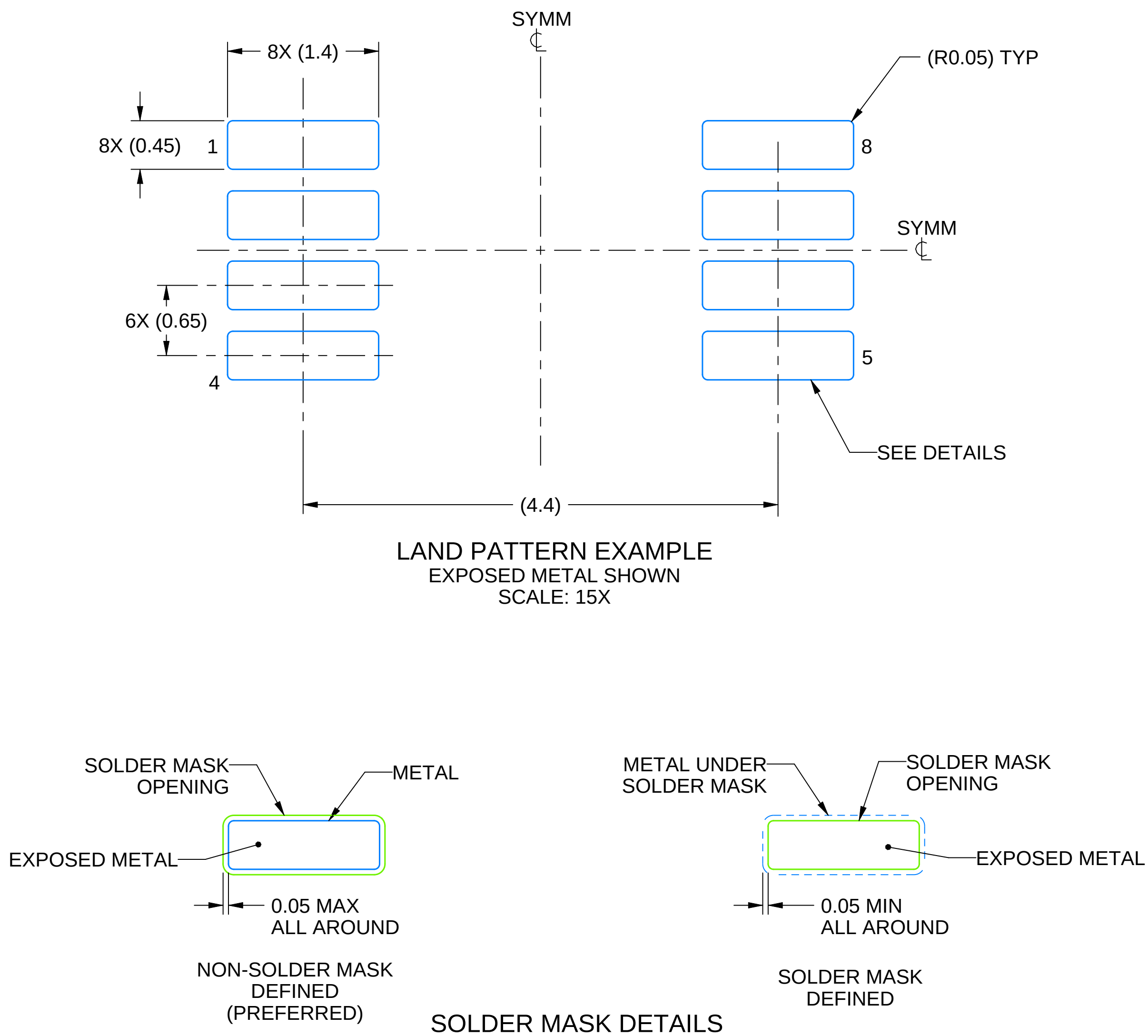




PowerPAD is a trademark of Texas Instruments.

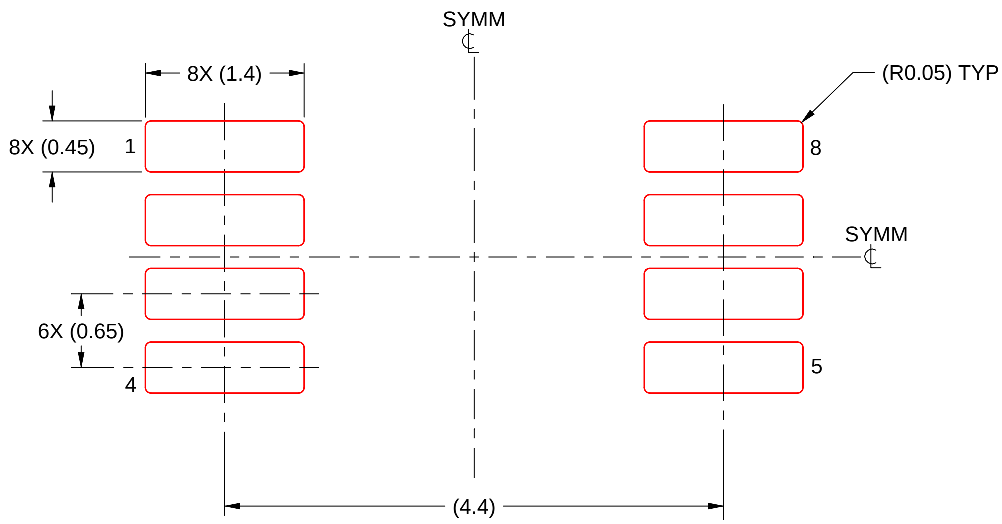
NOTES:

1. All linear dimensions are in millimeters. Any dimensions in parenthesis are for reference only. Dimensioning and tolerancing per ASME Y14.5M.
2. This drawing is subject to change without notice.
3. This dimension does not include mold flash, protrusions, or gate burrs. Mold flash, protrusions, or gate burrs shall not exceed 0.15 mm per side.
4. This dimension does not include interlead flash. Interlead flash shall not exceed 0.25 mm per side.
5. Reference JEDEC registration MO-187.



NOTES: (continued)

6. Publication IPC-7351 may have alternate designs.
7. Solder mask tolerances between and around signal pads can vary based on board fabrication site.
8. Vias are optional depending on application, refer to device data sheet. If any vias are implemented, refer to their locations shown on this view. It is recommended that vias under paste be filled, plugged or tented.
9. Size of metal pad may vary due to creepage requirement.



SOLDER PASTE EXAMPLE
SCALE: 15X

4214862/A 04/2023

NOTES: (continued)

11. Laser cutting apertures with trapezoidal walls and rounded corners may offer better paste release. IPC-7525 may have alternate design recommendations.
12. Board assembly site may have different recommendations for stencil design.

IMPORTANT NOTICE AND DISCLAIMER

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265