AD9832/AD9835 Programming Examples
by Liam Riordan

INTRODUCTION

This application note describes how to load a sinusoidal waveform on the output of the AD9832/AD9835 parts. This includes setting up the AD9832/AD9835 from a power-on scenario, setting up the FSELECT control bit, and loading new data to the part while using the FSELECT control bit.

PROGRAMMING THE AD9832/AD9835

When the AD9832/AD9835 is powered up, the part should be reset. This resets the appropriate internal registers to 0 to provide an analog output of full scale. To avoid spurious DAC outputs while the AD9832 is being initialized, the RESET bit should be set to 1 until the part is ready to begin generating an output. The RESET bit does not reset the phase, frequency, or control registers. These registers will contain invalid data, and, therefore, should be set to a known value by the user. The RESET bit should then be set to 0 to begin generating an output. The data appears on the DAC output seven to eight MCLK cycles after RESET is set to 0.
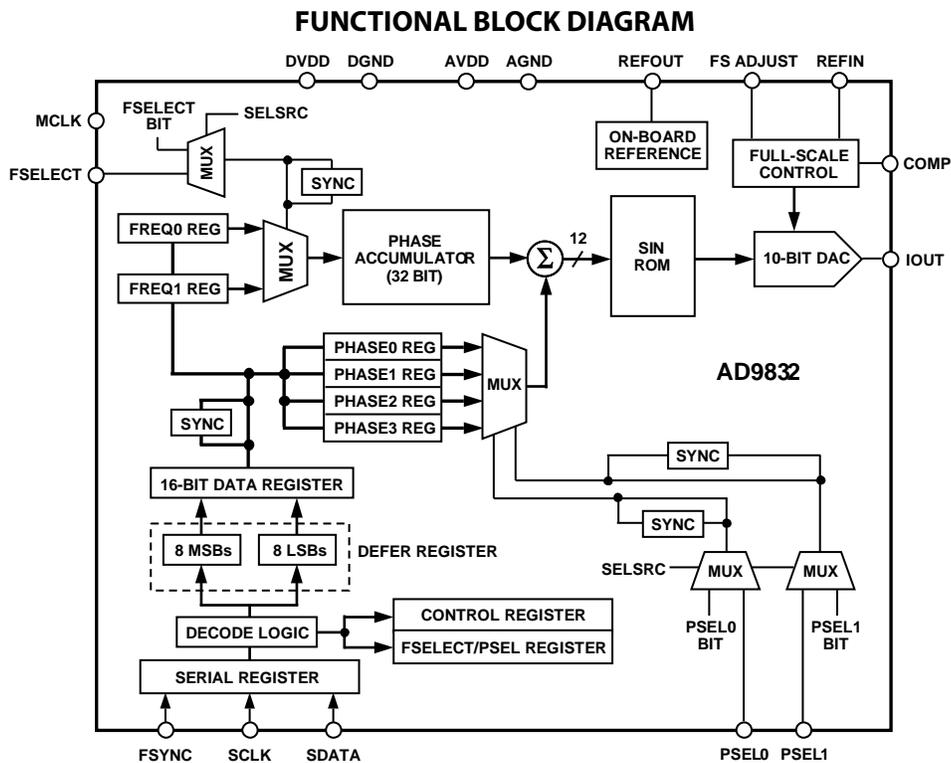
FUNCTIONAL BLOCK DIAGRAM

Figure 1. AD9832 Functional Block Diagram

TABLE OF CONTENTS

REVISION HISTORY

2/11—Revision 0: Initial Version

PROGRAMMING THE AD9832/AD9835

A simple example is best to explain how to program the AD9832 and the AD9835. For further details, refer to the AN-621 Application Note and the AD9832 or the AD9835 data sheet.

Example 1: Setting Up the Part After Power-Up

The aim is to generate two output frequencies, loading 3 kHz to the FREQ0 register, and 10 kHz to the FREQ1 register, with a 25 MHz MCLK.

The dial-up code for this is defined by the equation

$$FreqReg = \frac{f_{OUT} \times 2^{28}}{f_{MCLK}}$$

For *FREQ0* = 3000 Hz

$$FREQ0 = \frac{3000 \text{ Hz} \times 2^{32}}{25 \text{ MHz}}$$

$$= 515{,}396 \text{ decimal} = 0x0007DD44$$

For *FREQ1* = 10 KHz

$$FREQ1 = \frac{10{,}000 \text{ Hz} \times 2^{32}}{25 \text{ MHz}}$$

$$= 1{,}717{,}987 \text{ decimal} = 0x001A363E$$

Note that the ADIsimDDS tool offers a quick and easy method of generating the desired code in hexadecimal, decimal or binary.

The required initialization sequence is shown in Table 1.

Table 1. Initialization Sequence

Hexadecimal	Binary
0xD000	1101 0000 0000 0000
0x3044	0011 0000 0100 0100
0x21DD	0010 0001 1101 1101
0x3207	0011 0010 0000 0111
0x2300	0010 0011 0000 0000
0x343E	0011 0100 0011 1110
0x2536	0010 0101 0011 0110
0x361A	0011 0110 0001 1010
0x2700	0010 0111 0000 0000
0x9000	1001 0000 0000 0000
0XC000	1100 0000 0000 0000

Command Sequence Explained

Use the register tables in the AD9832 or AD9835 data sheet for reference.

0xD000—Resets AD9832

RESET Bit D12 is set to 1. This resets the internal registers to 0, which corresponds to an analog output of full scale (see Figure 2). Ch1 is the spurious output set to reset. Ch2 is FSYNC, Ch3 is SDATA, and Ch4 is SCLK. Note that data is valid on the falling edge of SCLK. Ignore the SDATA high after

eight SCLK cycles; this is an inherent feature of the particular SPI driver used and does not affect the part.
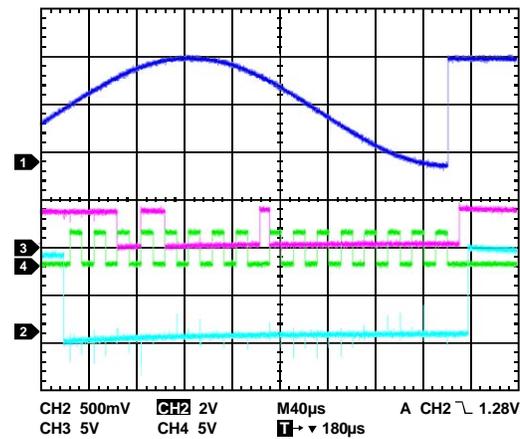

Figure 2. Put AD9832/AD9835 in Reset–0xD000

0x3044

- Eight bits to defer.

- Frequency Register 0, L LSB data.

- 8-bit data = 0x44.

0x21DD

- Present eight bits plus the deferred eight bits to the select frequency register.

- Frequency Register 0, H LSB data.
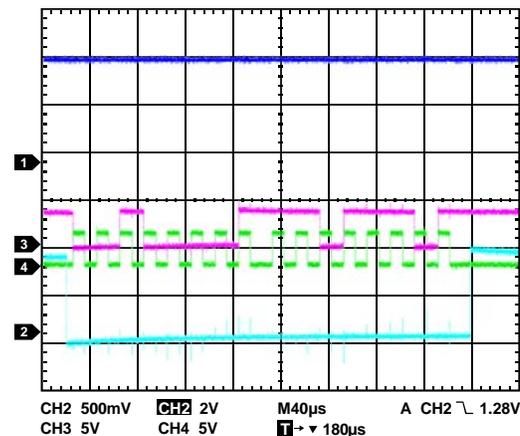
- 8-bit data (0xDD) + deferred data = 0xDD44.

Figure 3. Load 0x21DD

0x3207—Frequency Register 0, L MSB data

0x2300—Frequency Register 0, H MSB data

0x343E—Frequency Register 1, L LSB data

0x2536—Frequency Register 1, H LSB data

0x361A—Frequency Register 1, L MSB data

0x2700—Frequency Register 1, H MSB data

0x9000—Set SELSRC to 1 using Command Bits 1:0 for C15 to C14.

- Setting SELSRC to 1—FSELECT is controlled by data bits. Latency is increased by two MCLK cycles.

0xC000—Frequency Register 1, L LSB data.

- Exit reset mode. A 3 kHz signal, from the FREQ0 register, appears at the output of the DAC eight MCLK cycles after the RESET bit is set to 0.
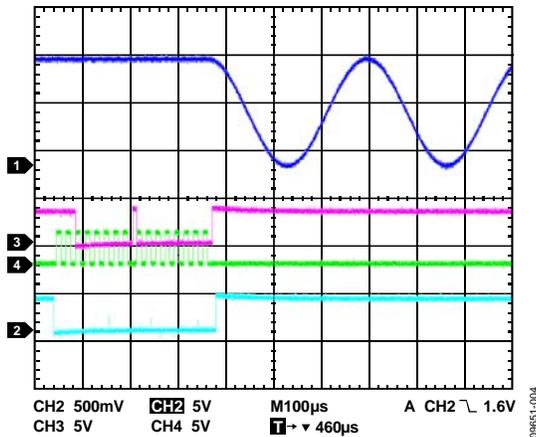

Figure 4. Take Part Out of Reset–0xC000

Example 2: Changing the FSELECT Bit

The part is now set up with a 3 kHz signal on the output and is controllable by the FSELECT bit, as opposed to the FSELECT pin. To change this frequency to 10 KHz requires only a write to the control register to change the FSELECT bit.

0x5800—Selects the FREQ1 register.
- That is, 0x5 written to the Command Bits C3:C0.
- A 1 written to Bit D11 to select the FREQ1 register (see Figure 5).
To return to the 3 kHz FREQ0 output, reset the FSELECT bit to 0.
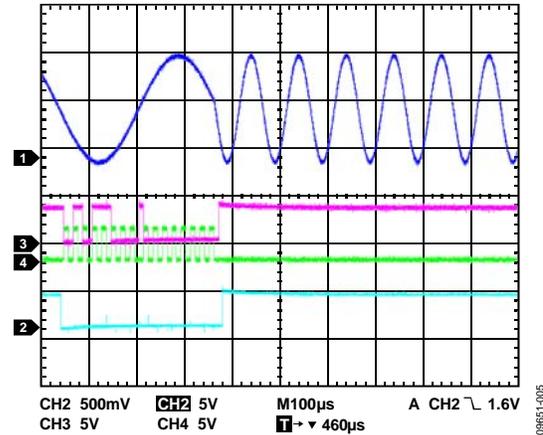
0x5800—Selects the FREQ0 register.

Figure 5. Change FSELECT Bit–0x5800

Example 3: Updating the Output as Needed

The next step is to change the FREQ1 register from 10 kHz to 40 kHz without entering RESET mode.

- For *FREQ1* = 40 kHz

$$FREQ1 = \frac{40{,}000 \text{ Hz} \times 2^{32}}{25 \text{ MHz}}$$

= 6,871,948 decimal = 0x0068DB8C

Load the following command sequence:

- 0x348C—Frequency Register 1, L LSB data.
- 0x25DB—Frequency Register 1, H LSB data.
- 0x3668—Frequency Register 1, L MSB data.
- 0x2700—Frequency Register 1, H MSB data.
- 0x9000—Latch to output by synchonizing data. In this case, the SELSRC bit is again set to 1 using Command Bits [1:0] for C15 and C14. The part simply needs to see a falling FSYNC edge to latch data (see Figure 6).
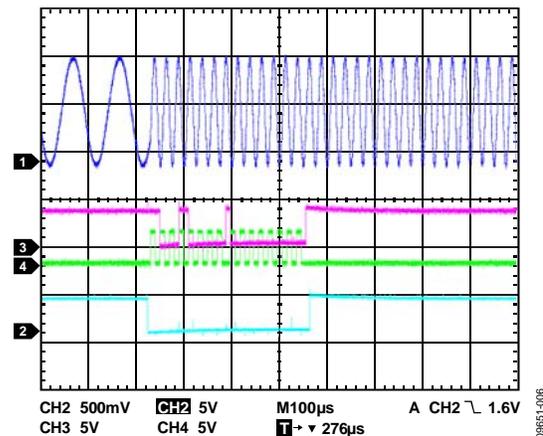

Figure 6. Changing FREQ1 Output from 10 kHz to 40 kHz

www.analog.com