Lab 9: Interfacing DAC with 8051 using SPI protocol

1 Problem Statement

1. Write a C program to generate a square wave signal using a Digital to Analog Converter (DAC) with SPI protocol and show the waveforms on oscilloscope.

   In the partial program given, update the SPCON register. As per the data sheet provided for the DAC MCP4911, update the write command register for the specified waveform generation.

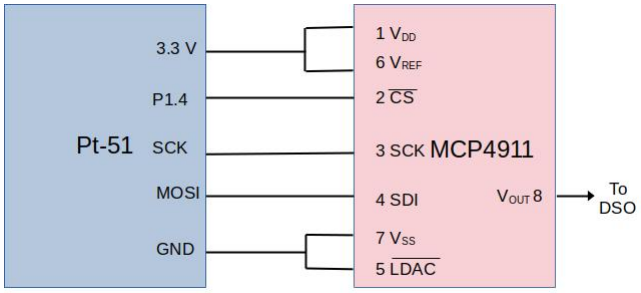

Figure 1: Connection Diagram

2 MCP4911 Digital to Analog Converter

MCP4911 is a 10-bit Digital to Analog Converter with a single voltage output channel, operating in a voltage range of 2.7 to 5.5 V, with SPI protocol.

2.1 Pin description

The pin description of MCP4911 is as follows:

PDIP, MSOP, SOIC	DFN	Symbol	Description
1	1	V_{DD}	Supply Voltage Input (2.7V to 5.5V)
2	2	$\overline{CS}$	Chip Select Input
3	3	SCK	Serial Clock Input
4	4	SDI	Serial Data Input
5	5	$\overline{LDAC}$	DAC Output Synchronization Input. This pin is used to transfer the input register (DAC settings) to the output register (V_{OUT})
6	6	V_{REF}	Voltage Reference Input
7	7	V_{SS}	Ground reference point for all circuitry on the device
8	8	V_{OUT}	DAC Analog Output
—	9	EP	Exposed Thermal Pad. This pad must be connected to V_{SS} in application

2.2 Analog output voltage (V_{out}) calculation

The DAC analog output voltage calculation is given by,

$$V_{out} = \frac{V_{ref} \times D_n}{2^n} G$$

where, V_{ref} - External input voltage reference
D_n - DAC input code
G - Gain selection
n - DAC resolution

3 DAC Serial Interface with SPI

MCP4911 DAC can be directly interfaced with SPI port on Pt-51 board. The communication is unidirectional.

- Analog output is obtained at V_{out} pin.
- Commands and data are send through SDI pin, serially.
- Write command consists of 16 bits, and used to configure the DAC's control and data latches.
- $\overline{CS}$ pin should be held low for duration of write command and raised after write command.

3.1 Write command register

1. All writes to the MCP4911 are 16-bit words, with the *most significant 4 bits being configuration bits and the remaining 10 bits are data bits.*

W-x	W-x	W-x	W-0	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x
0	BUF	$\overline{GA}$	$\overline{SHDN}$	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	x	x
bit 15															bit 0

Figure 2: Write command register for MCP4911 (10-bit ADC)

where, BUF: V_{REF} Input Buffer Control bit (1 = Buffered, 0 = Unbuffered)
$\overline{GA}$: Output Gain Control bit($1 := 1 \times V_{OUT}$, $0 := 2 \times V_{OUT}$)
$\overline{SHDN}$: V_{REF} Output Shutdown Control bit (1= Active mode of operation, i.e. V_{OUT} is available, 0= Shutdown the device i.e. V_{OUT} is not available.)
D_9 to D_0: DAC Input Data bit.

- Here, we set BUF, $\overline{GA}$ and $\overline{SHDN}$ for waveform generation.
- Each 8 bit of this 16 bits is to be transferred sequentially to SPDAT register to send the 16 bit word to DAC.

2. The write command is initiated by driving the $\overline{CS}$ pin low, followed by clocking the four configuration bits and the 10 data bits into the SDI pin. The $\overline{CS}$ pin is then raised, causing the data to be latched into the DAC's input register.

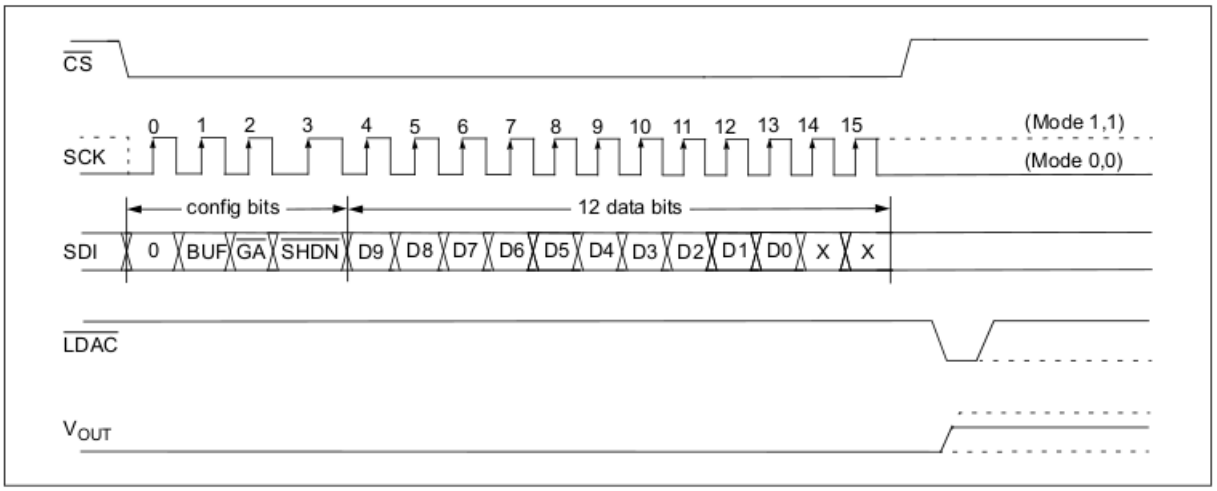

Figure 3: Write command for MCP4911 (10-bit ADC)

3. By bringing the $\overline{LDAC}$ pin down to a low state, the content stored in the DAC's input register is transferred into the DAC's output register (V_{out}), and (V_{out}) is updated.

3.2 SPCON register

SPI control register (SPCON) is used to control serial data transfer by setting the SPI as master, setting the baud rate and transmission mode and further running the SPI.

Steps to configure SPCON register to initialize it for SPI communication with DAC:

1. Configure the SPI module as Master (Set SSDIS and MSTR)
2. Select serial clock polarity and phase (Here, take it as CPHA=0 and CPOL=0)
3. Select appropriate baud rate (Here, take it as $f_{clk}/8$ by selecting SPR2, SPR1 and SPR0 with appropriate values)
4. Run SPI (Set SPEN)

7	6	5	4	3	2	1	0
SPR2	SPEN	SSDIS	MSTR	CPOL	CPHA	SPR1	SPR0

Bit Number	Bit Mnemonic	Description
7	SPR2	**Serial Peripheral Rate 2** Bit with SPR1 and SPR0 define the clock rate.
6	SPEN	**Serial Peripheral Enable** Cleared to disable the SPI interface. Set to enable the SPI interface.
5	SSDIS	**$\overline{SS}$ Disable** Cleared to enable $\overline{SS}$ in both Master and Slave modes. Set to disable $\overline{SS}$ in both Master and Slave modes. In Slave mode, this bit has no effect if CPHA = "0".
5	MSTR	**Serial Peripheral Master** Cleared to configure the SPI as a Slave. Set to configure the SPI as a Master.
4	CPOL	**Clock Polarity** Cleared to have the SCK set to "0" in idle state. Set to have the SCK set to "1" in idle low.
3	CPHA	**Clock Phase** Cleared to have the data sampled when the SCK leaves the idle state (see CPOL). Set to have the data sampled when the SCK returns to idle state (see CPOL).
2	SPR1	**SPR2 SPR1 SPR0 Serial Peripheral Rate** 0 0 0 Invalid 0 0 1 $F_{CLK\ PERIPH}/4$ 0 1 0 $F_{CLK\ PERIPH}/8$ 0 1 1 $F_{CLK\ PERIPH}/16$ 1 0 0 $F_{CLK\ PERIPH}/32$ 1 0 1 $F_{CLK\ PERIPH}/64$ 1 1 0 $F_{CLK\ PERIPH}/128$ 1 1 1 Invalid
1	SPR0	

Reset Value = 0001 0100b

Figure 4: SPCON register