SPI Protocol and Bus Configuration of Multiple DCPs

Application Note August 20, 2007 AN1340.0

The Serial Peripheral Interface (SPI) is one of the widely accepted communication interfaces implemented in Intersil's Digitally Controlled Potentiometers (DCP) portfolio. Developed by Motorola, the SPI protocol became a standard de facto, but does not have officially released specification or agreed by any international committee. That gives some flexibility in implementing SPI protocol in the electronic devices but also requires programmable flexibility of the host micro controller. This application note describes SPI bus implementations which utilize Intersil's DCPs.

Basic SPI Bus Information

The SPI is used for a synchronous serial communication of host micro controller and peripherals. SPI requires two control lines (CS and SCK) and two data lines (SDI and SDO) as shown in Figure 1.

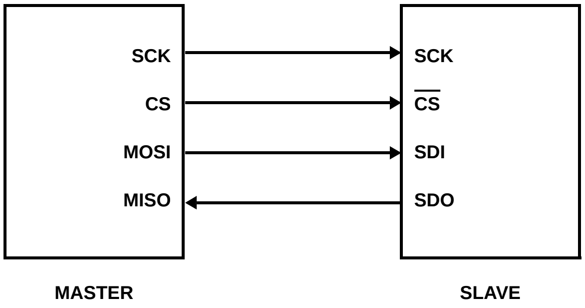

FIGURE 1. SINGLE MASTER-SLAVE SPI BUS IMPLEMENTATION

The SPI bus specifies four logic signals:

- SCK – Serial Clock, provided by master
- CS – Chip Select, allow master to select peripheral (slave) device
- MOSI/SDI – Master Output Slave Input/Serial Data In
- MISO/SDO – Master Input Slave Output/Serial Data Out

With CS active low, the corresponding peripheral device is selected. A master, usually the host micro controller, always provides clock signal to all devices on a bus whether it is selected or not. Only one master must be active on a bus at a time. The SPI protocol operates in full duplex mode, when input and output data transfers on both lines simultaneously. The unselected devices keep the SDO lines in hi-Z state and therefore inactive.

Since the clock serves as synchronization of the data communication, there are four possible modes that can be used in an SPI protocol, based on clock polarity (CPOL) and clock phase (CPHA) as shown in Table 1 and Figure 2.

TABLE 1. SPI MODES

SPI MODE	CPOL	CPHA
0	0	0
1	0	1
2	1	0
3	1	1

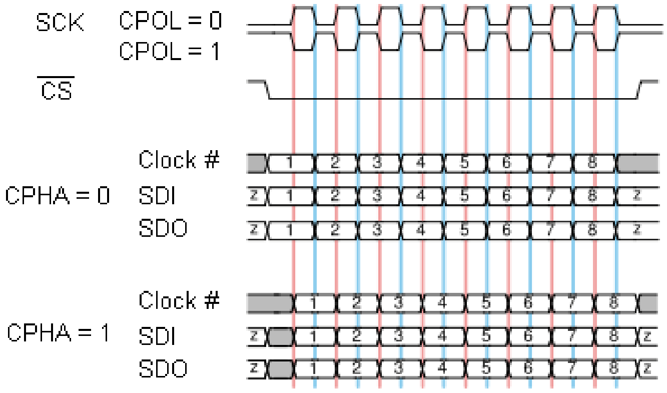

FIGURE 2. SPI PROTOCOL TIMING DIAGRAM

If the phase of the clock is zero (i.e. CPHA = 0) data is latched at the rising edge of the clock with CPOL = 0, and at the falling edge of the clock with CPOL = 1. If CPHA = 1, the polarities are reversed. Data is latched at the falling edge of the clock with CPOL = 0, and at the rising edge with CPOL = 1.

All Intersil's SPI DCPs support Mode 0 (CPOL = 0, CPHA = 0) protocol.

Bus Configuration and SPI Protocol of Multiple DCPs

Multiple slave devices can be connected in parallel or daisy chained utilizing the same SPI bus.

Parallel Configuration

For the parallel connection, each device on the bus should have a separate CS line, while SCK, SDI and SDO lines are connected in parallel as shown in Figure 3.

CAUTION: These devices are sensitive to electrostatic discharge; follow proper IC Handling Procedures.
1-888-INTERSIL or 1-888-468-3774 | Intersil (and design) is a registered trademark of Intersil Americas Inc.

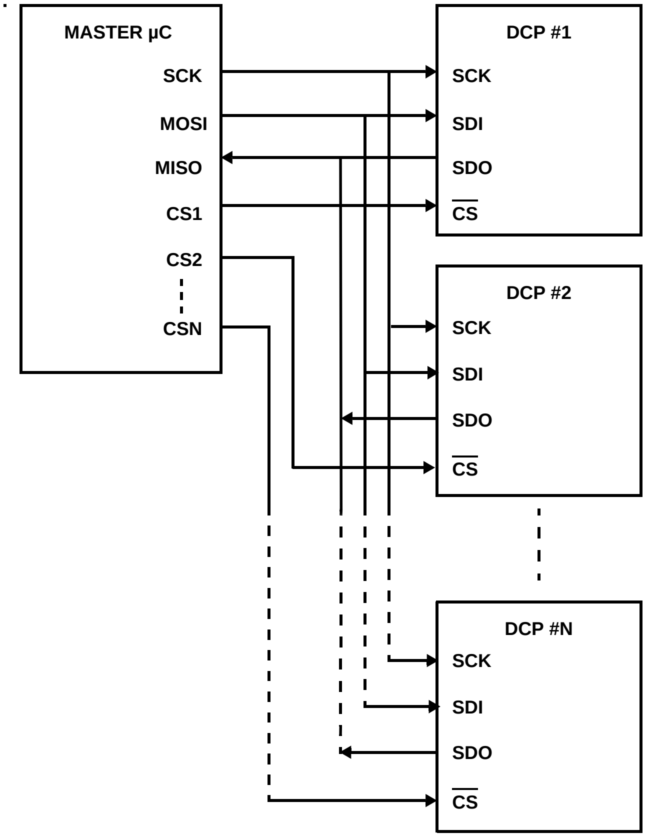

FIGURE 3. PARALLEL CONFIGURATION OF MULTIPLE DCPs

The bus protocol for the parallel configuration remains the same as for single master-slave configuration as in Figure 4 and Figure 5. Figure 4 shows a 2-byte write sequence, whereby the first byte contains instruction code 011b for Access Control Register (ACR) only, or 110b for any other register, followed by the register address where the data will be written. The second byte contains the data itself. Note that a write command with instruction code 011b is used for access to the ACR register only; any address bits will be ignored in the instruction byte.

For example, in order to move wiper RW1 of ISL22424 to 195th tap position the following sequence should be issued (only SDI line is exhibited).

011 00000 11000000 – set VOL and SHDN bit in ACR (ACR[7:6] = 11b)

110 00001 11000011 – write (code 110b) into wiper register WR1 at address 00001b the data 11000011b (195 decimal)

Similarly, in order to read the data from the wiper register WR1, master should issue a 4-byte sequence as shown in Figure 5.

For more information about bit definition, instructions and register maps, refer to the corresponding ISL224x4 datasheet.

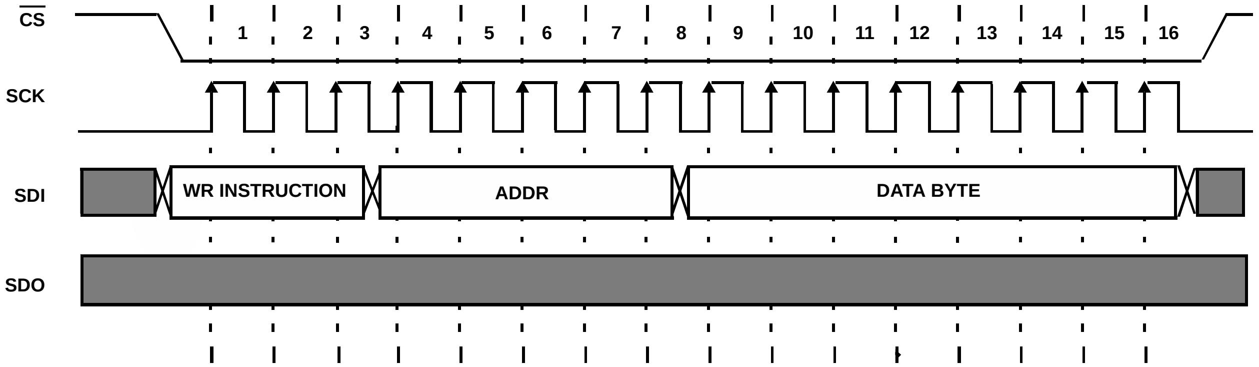

FIGURE 4. 2-BYTE WRITE SEQUENCE

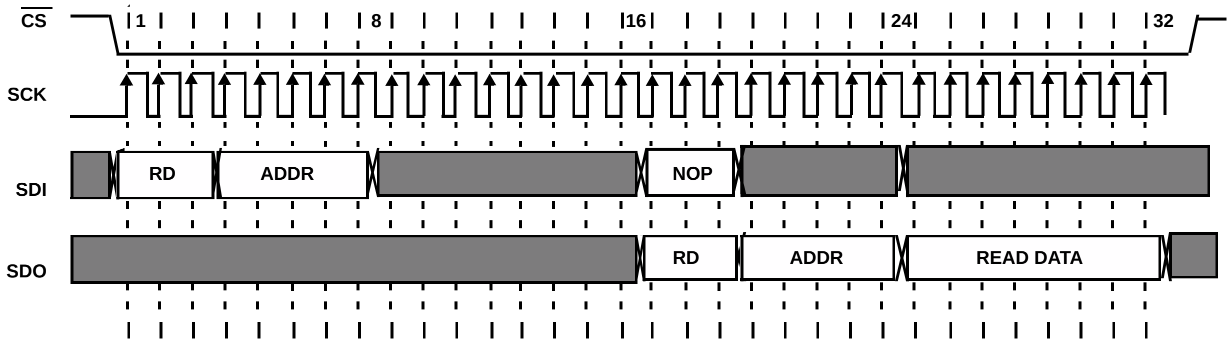

FIGURE 5. 4-BYTE READ SEQUENCE

The first byte must contain read instruction code 001b for ACR read only or 100b for read instruction from any other register plus register address from which the data will be read, followed by the dummy byte (i.e. 00000000b). The third byte must have a NOP instruction code of 00000000b, followed by another dummy byte (00000000b). The valid data will be read on SDO line during fourth byte. Note that during the third byte the read instruction and register address will be repeated on SDO line.

Continuing from the previous example, the read sequence is shown as:

MOSI line (binary): 10000001 00000000 00000000 00000000 (read instruction from address 00001b, wiper register WR1)

MISO line (binary): xxxxxxxx xxxxxxxx 10000001 11000011 (data out is second byte out, 195d).

Any write or read sequence must start with CS line transition from HIGH to LOW and finish with transition from LOW to HIGH. During write or read operation CS line must be kept LOW.

Daisy Chain Configuration

DCPs that support the daisy chain configuration (i.e. ISL224x4 family) can be connected as shown in Figure 6. In this configuration CS and SCK lines connected in parallel, and each SDO pin of previous chip is connected to SDI pin.

Daisy Chaining simplifies the connection by reducing the length and connections of the data lines, but restricts access to a single device in chain. In other words, all the devices in chain will be involved in a write or read operation. We can consider that every device in the chain is a portion of one big shift register, where serial data is shifted out on each clock going through all the DCPs, from DCP0 to DCP1 and all the way down to the last DCP(N-1) in this chain.

A chain write operation should be formed from N number of 2-byte write operations as shown in Figure 7.

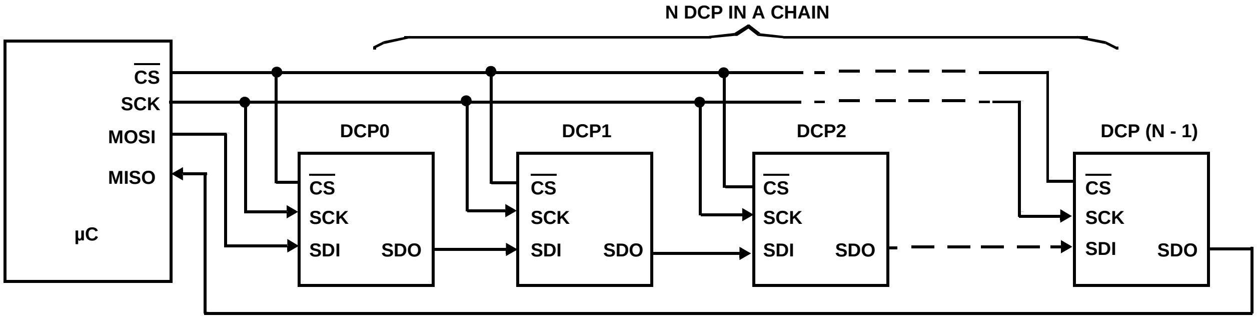

FIGURE 6. DAISY CHAIN CONFIGURATION OF MULTIPLE DCPs

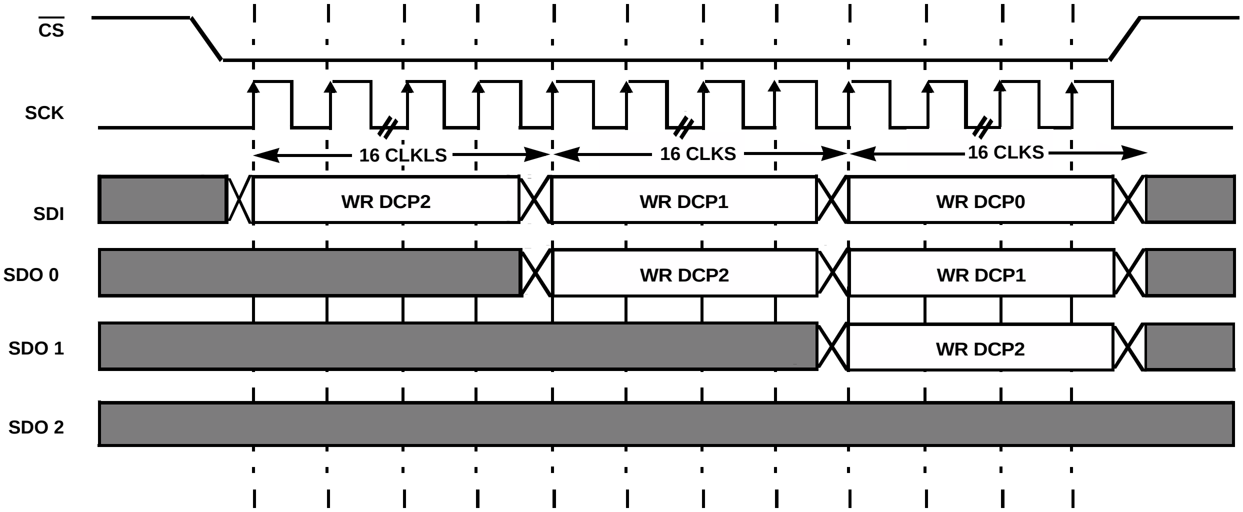

FIGURE 7. DAISY CHAIN WRITE SEQUENCE OF N = 3 DCP

The SPI master device shifted out serial data starting with the data going to the last DCP in the chain (instruction byte + data byte for DCP2, first 16 clocks), followed by two bytes for DCP1 (instruction byte + data byte, next 16 clocks) and tailing two bytes for the first DCP in the chain - DCP0 (instruction byte + data byte, last 16 clocks).

For example, we daisy chained three ISL22424 (Dual DCPs) and need to move wiper of DCP0 of device #1 to tap 31d, wiper of DCP1 of device #2 at tap 207d and wiper of DCP0 of device #3 at tap 126d. The sequence in Table 2 should be issued by the master device.

These commands will be executed by each device simultaneously on the rising edge of CS. Thus, Daisy Chain configuration is well suited for synchronous setting of different DCPs.

The read sequence of daisy chained ISL22424 DCPs, per previous example, consist of two parts:

1. The master sends three 2-byte read instructions, 48 clocks, (where the first byte is read instruction with address followed by dummy data byte)
2. The master reads the requested data on MISO in sequence shown on Figure 8 while sending three 2-byte NOP operations (48 clocks). Note, that CS has a positive pulse after first portion of the sequence (or after first 48 clocks).

A diagram of the 2-byte operation is shown in Figure 9.

For our example of three daisy chained ISL22424 devices, the read sequence to obtain the wiper positions for DCP0 of device #1, for DCP1 of device #2, for DCP0 of device #3 is shown in Table 3. The resulting data is: device#1 = 7Eh or 126d, device #2 = CFh or 207d, and device #3 = 1Fh or 31d.

TABLE 2.

MOSI Line	11000000	01111110	11000001	11001111	11000000	00011111
	WR DCP0	126	WR DCP1	207	WR DCP0	31

TABLE 3.

MOSI Line	10000000	00000000	10000001	00000000	10000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
MISO Line	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx	01111110	xxxxxxxx	11001111	xxxxxxxx	00011111

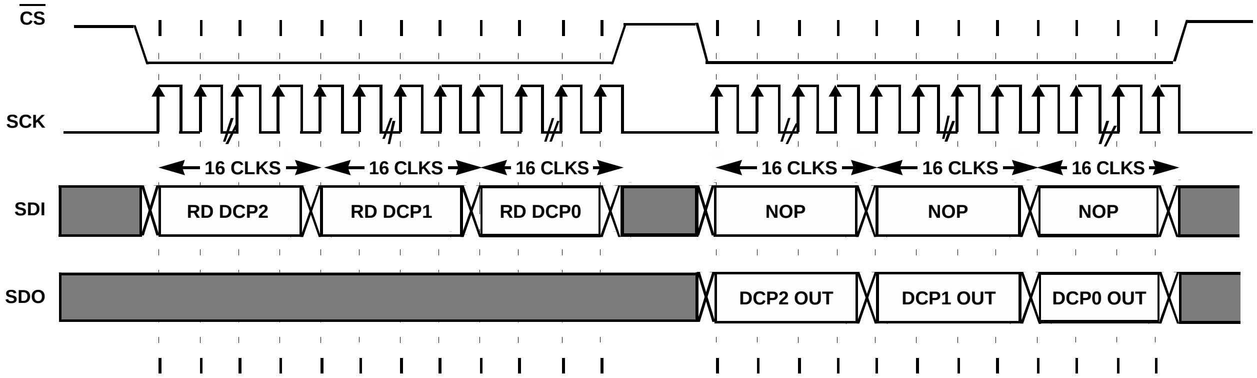

FIGURE 8. DAISY CHAIN READ SEQUENCE OF N = 3 DCP

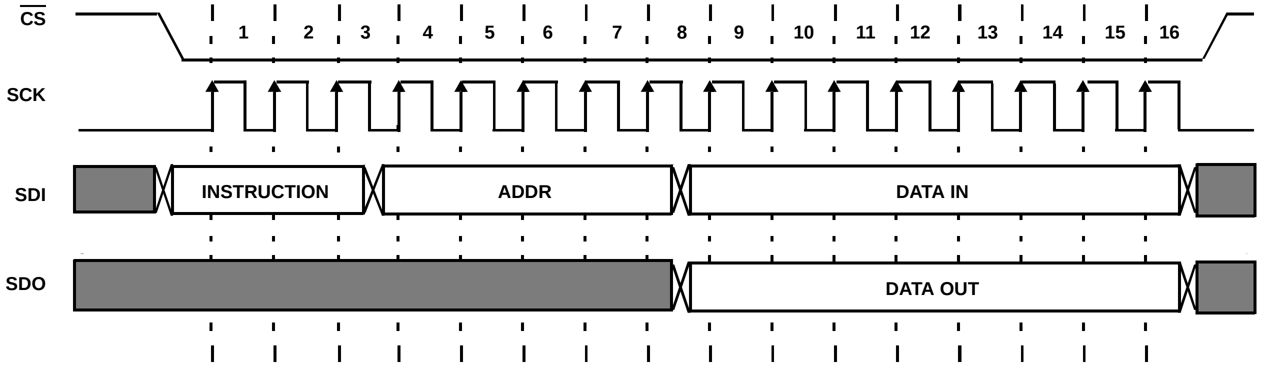

FIGURE 9. 2-BYTE OPERATION

Conclusion

The SPI bus allows for either parallel or daisy chain configurations of Intersil DCPs, with clock bus loading being the only practical limit to the number of devices that can be connected. The parallel configuration allows simplified access to individual DCPs and straightforward firmware code but a more complicated interconnect. The daisy chain configuration allows simplified interconnect and synchronous loading of DCPs at the expense of more complex code.

Intersil Corporation reserves the right to make changes in circuit design, software and/or specifications at any time without notice. Accordingly, the reader is cautioned to verify that the Application Note or Technical Brief is current before proceeding.

For information regarding Intersil Corporation and its products, see www.intersil.com