Daisy Chaining Precision DACs

1.0 Introduction

It is not uncommon for the system designer to face a quagmire of reconciling the system complexity with the desire to keep the system footprint small. One specific example that often arises in the context of small system footprint is the need for single master controller to communicate with a number of slave devices. This is not much of a problem if the master controller has multiple I/O resources available, and the routing of individual busses to the slave devices can be accommodated by ample board space. Challenge arises when the master controller has only one I/O port available, and the signal routing space is scarce. In these cases Daisy Chaining of slave devices may be the solution to consider.

This application note concerns exclusively the family of Precision DAC devices offered by National Semiconductor Corp. All of the devices mentioned in this note have a unidirectional SPI interface through which they receive data and configuration commands from the master controller. However, since the protocol details of each SPI interface differ slightly, due care must be taken when architecting systems comprising multiple Precision DACs.

The following sections will demonstrate how to achieve an arbitrary number of analog output channels that are controlled by a single 3-wire SPI interface. These expansion schemes will exploit the Daisy Chaining capability of the 8-channel DACs (DACxx8S085), and unique properties of the SPI interface of the Micro Power family of devices (DACxx4S085, DACxx2S085 and DACxx1S101).

2.0 Revisiting Micro Power DACs' Digital Interface: 1, 2 and 4 Channel Devices

The series of Micro Power DACs comprises the following families of devices: DACxx4S085 (4 channel), DACxx2S085

(2 channel), DACxx1S101 (single channel). A typical digital interface connectivity is shown in *Figure 1*. Here, a controller transfers data into a single DAC via the unidirectional SPI bus.

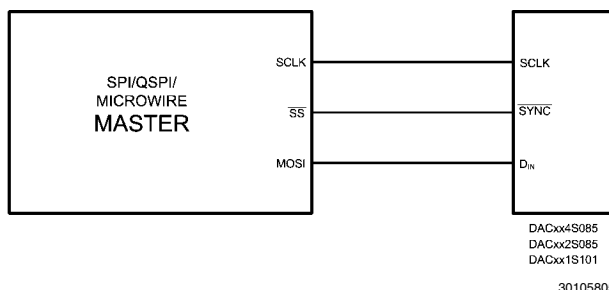DACxx4S085
DACxx2S085
DACxx1S101

30105805

FIGURE 1. Typical Digital Interface Connectivity

A typical bus cycle is shown in *Figure 2*. A bus cycle is initiated with the falling edge of $\overline{SS}$. The DAC shifts bits presented by MOSI into its internal shift register on each falling edge of the SCLK, for 16 clock cycles. After the 16th falling edge the DATA contained in the shift register is interpreted by the DAC's internal controller resulting in output level and/or mode of operation update. Also, on that 16th falling edge of the CLOCK the internal SCLK and D_{IN} busses of the DAC are gated off and therefore any further DATA present at D_{IN} pin, or SCLK pulse, is ignored.

The $\overline{SS}$ signal can be raised again any time after 16th falling edge. Subsequent DATA transfers will commence with the falling edge of the $\overline{SS}$.
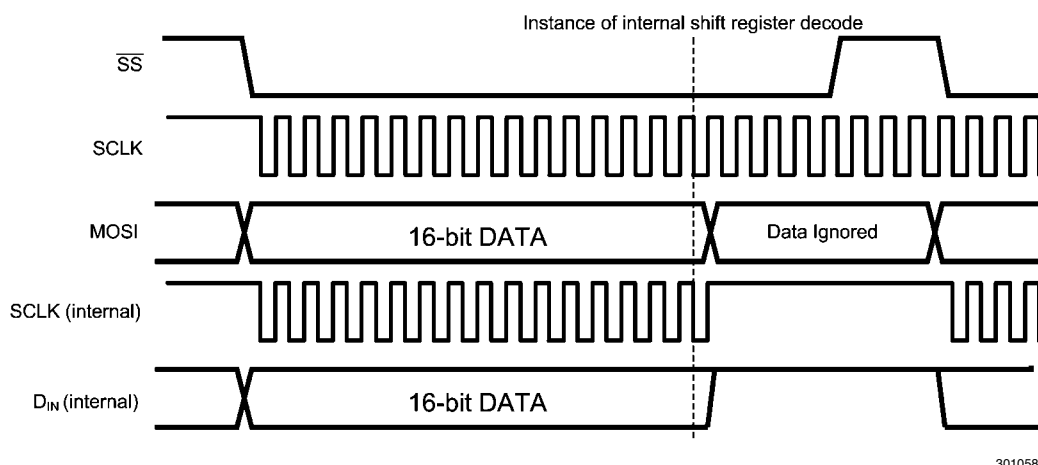

Instance of internal shift register decode

$\overline{SS}$

SCLK

MOSI — 16-bit DATA — Data Ignored

SCLK (internal)

D_{IN} (internal) — 16-bit DATA

30105808

FIGURE 2. Typical Bus Cycle

Obviously the above described scheme allows the controller to interface with a single device only. In many applications this is sufficient, but if more channels are needed see next section for solutions providing 8 DAC channels, and more.

For more detailed information on the DACxx4S085, DACxx2S085 and DACxx1S101 please refer to the respective device Data Sheet.

3.0 Revisiting Eight-Channel DACs' Digital Interface

This group of DACs has just 3 members: DAC088S085, DAC108S085 and DAC128S085 (8-channel devices with 8, 10 and 12-bit resolutions respectively).These 8-channel de-

vices were designed to interface to the controller individually, or in the group of N devices in the so called Daisy Chain scheme.

A typical digital interface connectivity is shown in *Figure 3*. Here, a controller transfers data into N DACs via the unidirectional SPI bus.

Unlike the DACs in the previous section, DACxx8S085 features D_{OUT} output, which is the output of the DAC's internal shift register. The DATA that was shifted into the D_{IN} input of the device will start appearing at D_{OUT} output after 16 SCLK cycles. Therefore, it is possible to feed the D_{OUT} of the first device in the chain to the D_{IN} of the following device. Arbitrarily long chain of devices can be assembled.
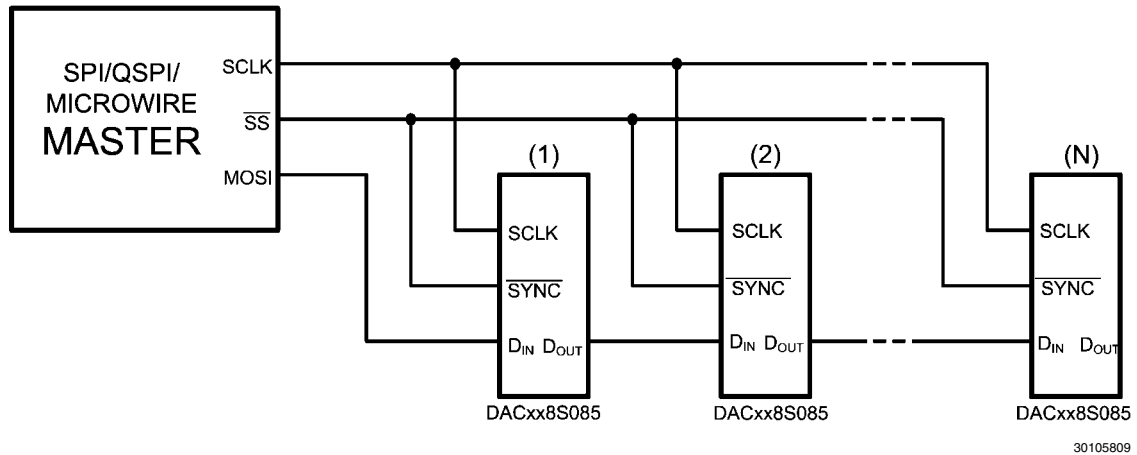

30105809

FIGURE 3. Typical Controller to N 8–Channel DACs Interface: Daisy Chain

The arbitrary length of the chain is possible only because the individual devices do not "count" the shifted DATA bits, rather they wait for the rising edge of $\overline{SS}$ to decode the contents of the internal shift register.

A typical bus cycle for a device in this family of DACs is shown in *Figure 4*. Falling edge of $\overline{SS}$ initiates the DATA transfer into the DAC's shift register. On each falling edge of SCLK a bit is acquired. After 16 CLOCK cycles the DATA starts to appear at the D_{OUT} output of the DAC. Rising edge of $\overline{SS}$ will cause the internal controller to interpret the contents of the shift register resulting in output level and/or mode of operation update.

Caution: It should be noted that the DATA at D_{OUT} is preceded by a "1" bit. Even though this pre-amble bit may be shifted into the following devices in the Daisy Chain, if the DATA protocol is strictly observed, this bit will not

be present in any of the shift registers when the $\overline{SS}$ is raised (moment of shift register content decode) – the DATA protocol dictates that all transfers into the DACxx8S085 consist of series of at least N*16 bits where N is the number of devices in the chain. Refer to the DACxx8S085 Data Sheet for details.

Note that *Figure 4*. shows a bus cycle of just one device in the chain. Also note that the number of SCLK cycles during the interval when $\overline{SS}$ is low exceeds 16. Therefore, after delay, D_{OUT} starts presenting data that was shifted into D_{IN} input of the internal shift register. When the rising edge of $\overline{SS}$ arrives, only 16 bits shifted in immediately before this edge are decoded by the internal controller. The data preceding the 16 decoded bits (marked as "runt" in *Figure 4*), is captured by the next device in the Daisy Chain, or simply discarded.
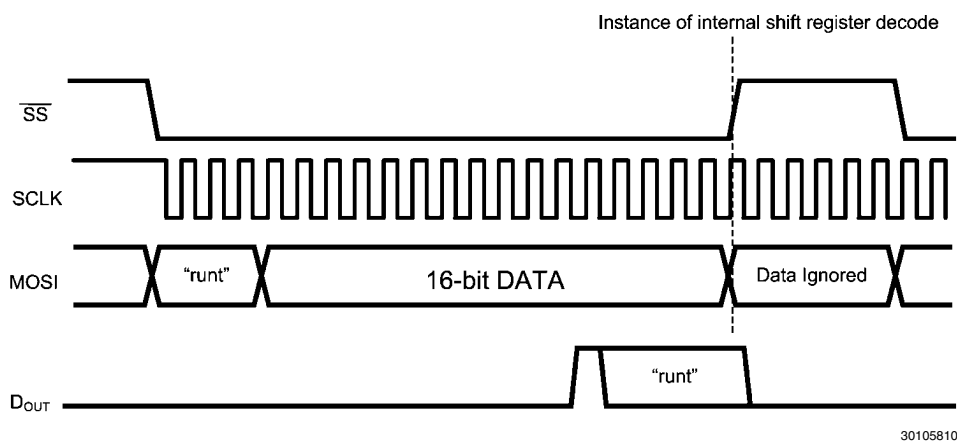
FIGURE 4. 8–Channel DAC Bus Cycle

Daisy Chain scheme shown in *Figure 3* provides (k*8) analog channels, where k=0…N, which can be controlled by a single, unidirectional SPI bus.

The following section describes how to combine both the Micro Power DACs (1, 2 and 4 channel devices), and 8-channel DACs on a single SPI bus, and thus enable creation of arbitrary number of analog output channels.

For more detailed information on the DACxx8S085, please refer to the respective device Data Sheet.

4.0 Flexible DAC Channel Expansion

The previously presented DAC families, in Sections 2 and 3, provide a wide range of choices to the system designer. But still there are some limitations: first scheme, in *Section 2.0 Revisiting Micro Power DACs' Digital Interface: 1, 2 and 4 Channel Devices*, allows for at most 4 channels (1, 2, or 4), the second allows for any multiple of 8-channels to be controlled by a single SPI bus (say 8, 16, 24…..). But what if 10 channels are needed?

The technique presented in *Figure 5* creates a DAC solution with a number of channels = (0|1|2|4)+k*8, where k=0…N (e.g. 1, 2, 4, 8, 9, 10, 12, 16, 17… are some of the possible DAC channel multiples that can be created).
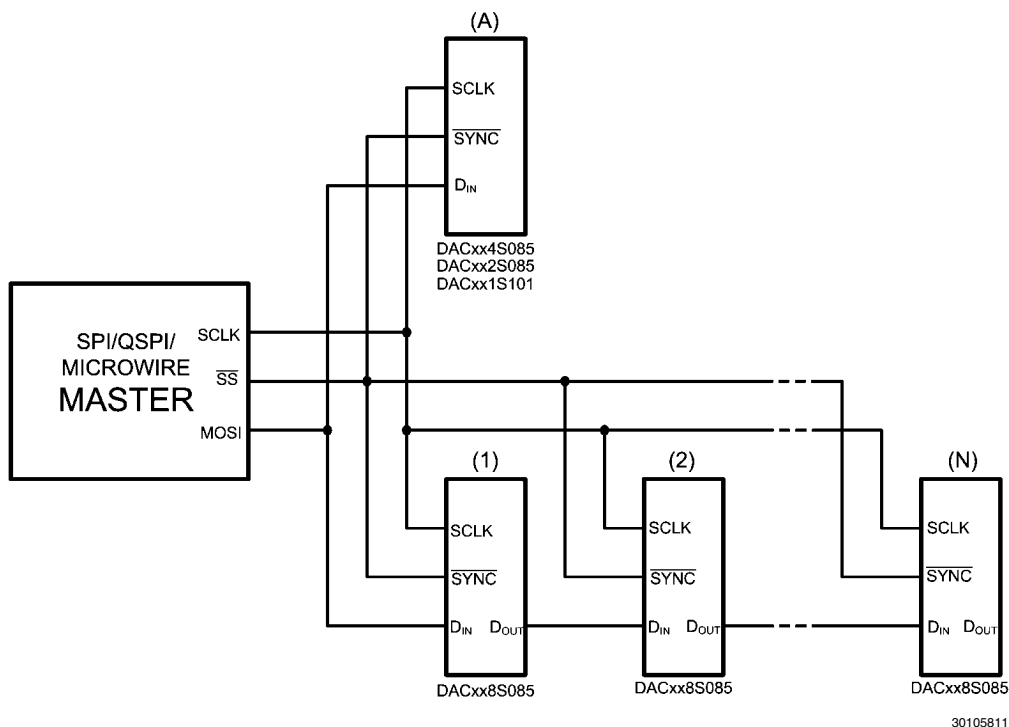

FIGURE 5. Flexible DAC Channel Expansion

A bus cycle of this scheme is shown in *Figure 6*. As in the previous sections the DATA transfer is initiated by the falling edge of $\overline{SS}$. And as in the previous sections the first 16 bits of DATA are shifted into both the device (A) and device (1) in FIGURE 5. But after the 16th falling edge of the SCLK attention needs to be paid to the differences in SPI protocol between device (A) and devices (1)… (N).

Consider device (A) first. After the 16th falling edge of SCLK device (A) will stop the shifting operation, the contents of its shift register will be decoded, and the state of this device will update. Internal SCLK and D_{IN} buses will be gated off, and device (A) will ignore any further DATA. It will stay in this state until next appearance of falling edge of $\overline{SS}$.

Now, consider devices in the Daisy Chain (1) through (N). As $\overline{SS}$ signal stays low, and SCLK transitions continue, so does the shift operation of the chain of devices (1)… (N). If the transitions continue for at least N*16 additional SCLK cycles beyond DATA(A) interval, the initial DATA(A) will pass through the (1)…(N) chain and will never be captured (decoded) by these devices.

Therefore, to program the (1)…(N) devices the 16 bit DATA (A) word has to be followed by N*16 data words (Nth word first) which are immediately followed by rising edge of $\overline{SS}$.

For example, if device (A) is present and N=2, then the data transfer must consist of at least 16+2*16=48 bits: the first 16 bits will be decoded by device (A), next 16 bits will be decoded by device (2), and the last 16 bits will be decoded by device (1). If there are any extra SCLK cycles present, these will shift in the "runt" data.
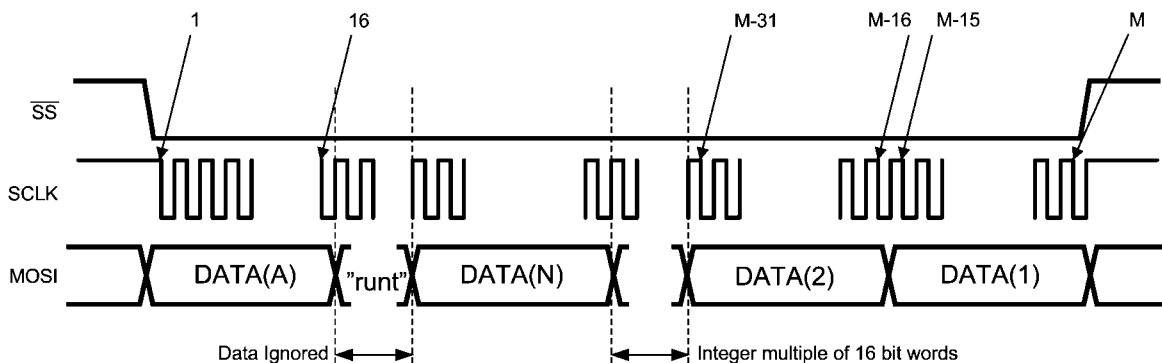

FIGURE 6. Flexible DAC Bus Cycle

Another way of looking at this data protocol is to consider interval when $\overline{SS}$ is low as a "frame" of data. The data for device (A) is left-justified within the "frame": the SCLK falling edges 1 through 16 immediately following falling edge of $\overline{SS}$ are of interest. Whereas the N*16 bits of data destined for devices (1)… (N) are right-justified within the "frame": if total data transfer consists of M bits, then bits of interest are M, (M-1)… (M-N*16+1). The excess SCLK pulses between the DATA(A) and DATA(1..N) will result in "runt", or discarded, data.

CAUTION: Observe that the data destined for the Daisy Chained devices (1)… (N) appears in reverse order from the actual physical device arrangement – compare *Figure 5* and *Figure 6*. This simply arises from the fact that the first data word has to travel the farthest down the chain. However, the bit order within each data word is not reversed: it is still MSB first.

Figure 7 shows the internal shift register contents of each device following the execution of the bus cycle shown in **Figure 6**. Here, device (A) contains DATA(A), while that same DATA (A) and the "runt" data have been pushed through, and discarded by the Daisy Chain of devices (1)…(N). Devices in the Daisy Chain contain the DATA that followed DATA(A) and "runt" data.

Thus every DAC in the diagram shown in *Figure 5* has acquired unique data, which is requisite for independent control of each device by a single master.
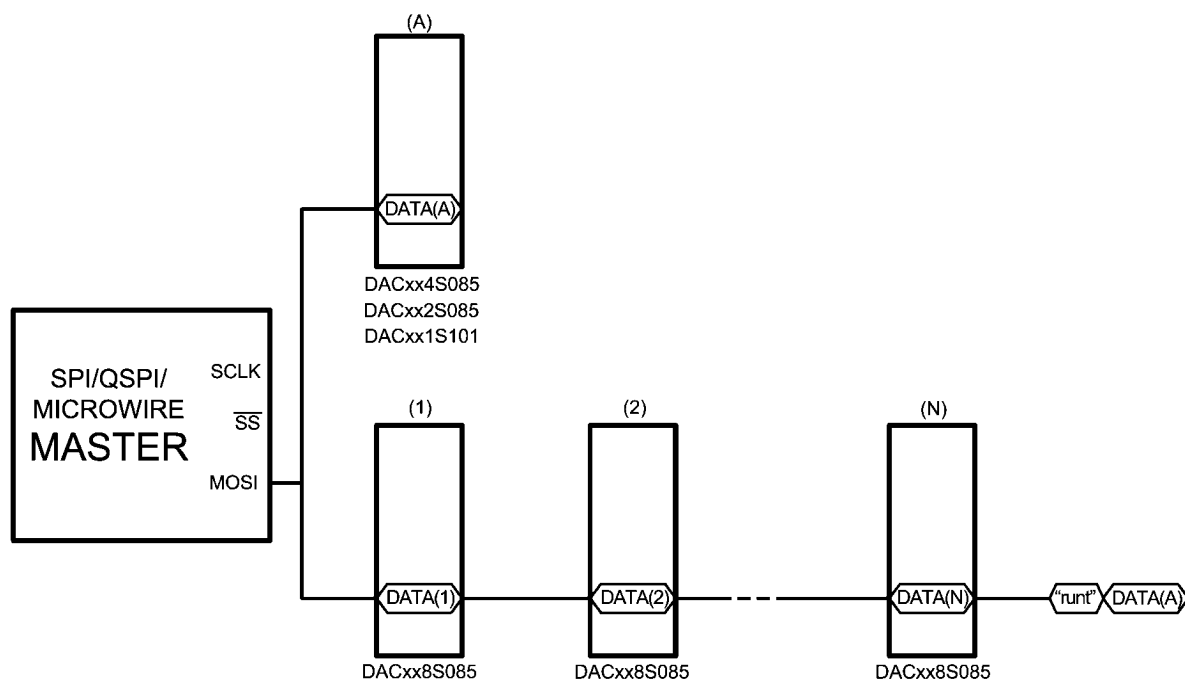
30105813

FIGURE 7. Data Register Contents Resulting From Bus Cycle Shown in Figure 6

5.0 Conclusion

This application note presented a summary of interface methods available in the Precision DAC products from National Semiconductor Corp. In particular, the architecture shown in *Section 4.0 Flexible DAC Channel Expansion* described the methods of interfacing DACs from 2 distinct families of DACs: Micro Power (1, 2 and 4 channel), and 8 channel. This Flexible DAC Expansion technique allows the system designer ultimate freedom in mixing and matching DACs in order to achieve desired number of analog output channels controllable by a single SPI master.

Note: This Application Note applies to:
- DAC081S101, DAC101S101, DAC121S101;
- DAC082S085, DAC102S085, DAC122S085;
- DAC084S085, DAC104S085, DAC124S085; and
- DAC088S085, DAC108S085, DAC128085

Notes

For more National Semiconductor product information and proven design tools, visit the following Web sites at:

Products		Design Support	
Amplifiers	www.national.com/amplifiers	WEBENCH® Tools	www.national.com/webench
Audio	www.national.com/audio	App Notes	www.national.com/appnotes
Clock and Timing	www.national.com/timing	Reference Designs	www.national.com/refdesigns
Data Converters	www.national.com/adc	Samples	www.national.com/samples
Interface	www.national.com/interface	Eval Boards	www.national.com/evalboards
LVDS	www.national.com/lvds	Packaging	www.national.com/packaging
Power Management	www.national.com/power	Green Compliance	www.national.com/quality/green
Switching Regulators	www.national.com/switchers	Distributors	www.national.com/contacts
LDOs	www.national.com/ldo	Quality and Reliability	www.national.com/quality
LED Lighting	www.national.com/led	Feedback/Support	www.national.com/feedback
Voltage Reference	www.national.com/vref	Design Made Easy	www.national.com/easy
PowerWise® Solutions	www.national.com/powerwise	Solutions	www.national.com/solutions
Serial Digital Interface (SDI)	www.national.com/sdi	Mil/Aero	www.national.com/milaero
Temperature Sensors	www.national.com/tempsensors	SolarMagic™	www.national.com/solarmagic
Wireless (PLL/VCO)	www.national.com/wireless	PowerWise® Design University	www.national.com/training

National Semiconductor Americas Technical Support Center Email: support@nsc.com Tel: 1-800-272-9959	**National Semiconductor Europe Technical Support Center** Email: europe.support@nsc.com	**National Semiconductor Asia Pacific Technical Support Center** Email: ap.support@nsc.com	**National Semiconductor Japan Technical Support Center** Email: jpn.feedback@nsc.com