



# 12 Channel, 8-Bit TrimDACs with Power Shutdown

## AD8802/AD8804

### FEATURES

- Low Cost
- Replaces 12 Potentiometers
- Individually Programmable Outputs
- 3-Wire SPI Compatible Serial Input
- Power Shutdown  $<55 \mu\text{Watts}$  Including  $I_{DD}$  &  $I_{REF}$
- Midscale Preset, AD8802
- Separate  $V_{REFL}$  Range Setting, AD8804
- +3 V to +5 V Single Supply Operation

### APPLICATIONS

- Automatic Adjustment
- Trimmer Replacement
- Video and Audio Equipment Gain and Offset Adjustment
- Portable and Battery Operated Equipment

### GENERAL DESCRIPTION

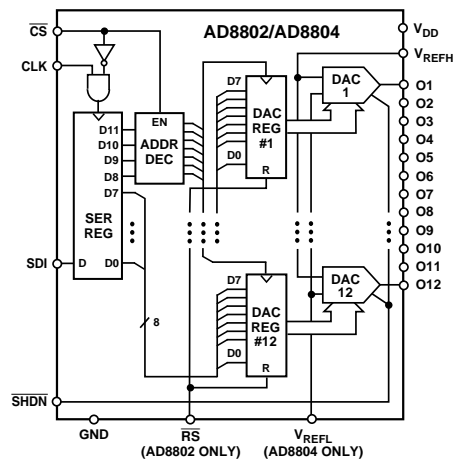
The 12-channel AD8802/AD8804 provides independent digitally-controllable voltage outputs in a compact 20-lead package. This potentiometer divider TrimDAC® allows replacement of the mechanical trimmer function in new designs. The AD8802/AD8804 is ideal for dc voltage adjustment applications.

Easily programmed by serial interfaced microcontroller ports, the AD8802 with its midscale preset is ideal for potentiometer replacement where adjustments start at a nominal value. Applications such as gain control of video amplifiers, voltage controlled frequencies and bandwidths in video equipment, geometric correction and automatic adjustment in CRT computer graphic displays are a few of the many applications ideally suited for these parts. The AD8804 provides independent control of both the top and bottom end of the potentiometer divider allowing a separate zero-scale voltage setting determined by the  $V_{REFL}$  pin. This is helpful for maximizing the resolution of devices with a limited allowable voltage control range.

Internally the AD8802/AD8804 contains 12 voltage-output digital-to-analog converters, sharing a common reference-voltage input.

TrimDAC is a registered trademark of Analog Devices, Inc.

### FUNCTIONAL BLOCK DIAGRAM



Each DAC has its own DAC latch that holds its output state. These DAC latches are updated from an internal serial-to-parallel shift register that is loaded from a standard 3-wire serial input digital interface. The serial-data-input word is decoded where the first 4 bits determine the address of the DAC latches to be loaded with the last 8 bits of data. The AD8802/AD8804 consumes only  $10 \mu\text{A}$  from 5 V power supplies. In addition, in shutdown mode reference input current consumption is also reduced to  $10 \mu\text{A}$  while saving the DAC latch settings for use after return to normal operation.

The AD8802/AD8804 is available in the 20-pin plastic DIP, the SOIC-20 surface mount package, and the 1 mm thin TSSOP-20 package.

REV. 0

Information furnished by Analog Devices is believed to be accurate and reliable. However, no responsibility is assumed by Analog Devices for its use, nor for any infringements of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of Analog Devices.

© Analog Devices, Inc., 1995

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.  
Tel: 617/329-4700 Fax: 617/326-8703

# AD8802/AD8804—SPECIFICATIONS $(V_{DD} = +3\text{ V} \pm 10\%$ or $+5\text{ V} \pm 10\%$ , $V_{REFH} = +V_{DD}$ , $V_{REFL} = 0\text{ V}$ , $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ unless otherwise noted)

Parameter	Symbol	Conditions	Min	Typ <sup>1</sup>	Max	Units
<b>STATIC ACCURACY</b>						
Specifications apply to all DACs						
Resolution	N	Guaranteed Monotonic	8			Bits
Differential Nonlinearity Error	DNL		-1	$\pm 1/4$	+1	LSB
Integral Nonlinearity Error	INL		-1.5	$\pm 1/2$	+1.5	LSB
Full-Scale Error	$G_{FSE}$		-1	1/2	+1	LSB
Zero Code Error	$V_{ZSE}$		-1	1/4	+1	LSB
DAC Output Resistance	$R_{OUT}$		3	5	8	k $\Omega$
Output Resistance Match	$\Delta R/R_0$			1.5		%
<b>REFERENCE INPUT</b>						
Voltage Range <sup>2</sup>	$V_{REFH}$ $V_{REFL}$	Pin Available on AD8804 Only	0		$V_{DD}$	V
REFH Input Resistance	$R_{REFH}$	Digital Inputs = 55 <sub>H</sub> , $V_{REFH} = V_{DD}$		1.2		k $\Omega$
REFL Input Resistance <sup>3</sup>	$R_{REFL}$	Digital Inputs = 55 <sub>H</sub> , $V_{REFL} = V_{DD}$		1.2		k $\Omega$
Reference Input Capacitance <sup>3</sup>	$C_{REF0}$ $C_{REF1}$	Digital Inputs all Zeros Digital Inputs all Ones		32 32		pF pF
<b>DIGITAL INPUTS</b>						
Logic High	$V_{IH}$	$V_{DD} = +5\text{ V}$	2.4			V
Logic Low	$V_{IL}$	$V_{DD} = +5\text{ V}$			0.8	V
Logic High	$V_{IH}$	$V_{DD} = +3\text{ V}$	2.1			V
Logic Low	$V_{IL}$	$V_{DD} = +3\text{ V}$			0.6	V
Input Current	$I_{IL}$	$V_{IN} = 0\text{ V}$ or $+5\text{ V}$			$\pm 1$	$\mu\text{A}$
Input Capacitance <sup>3</sup>	$C_{IL}$			5		pF
<b>POWER SUPPLIES<sup>4</sup></b>						
Power Supply Range	$V_{DD}$ Range		2.7		5.5	V
Supply Current (CMOS)	$I_{DD}$	$V_{IH} = V_{DD}$ or $V_{IL} = 0\text{ V}$		0.01	10	$\mu\text{A}$
Supply Current (TTL)	$I_{DD}$	$V_{IH} = 2.4\text{ V}$ or $V_{IL} = 0.8\text{ V}$ , $V_{DD} = +5.5\text{ V}$		1	4	mA
Shutdown Current	$I_{REFH}$	$\overline{\text{SHDN}} = 0$		0.2	10	$\mu\text{A}$
Power Dissipation	$P_{DISS}$	$V_{IH} = V_{DD}$ or $V_{IL} = 0\text{ V}$ , $V_{DD} = +5.5\text{ V}$			55	$\mu\text{W}$
Power Supply Sensitivity	PSRR	$V_{DD} = +5\text{ V} \pm 10\%$		0.001	0.002	%/%
<b>DYNAMIC PERFORMANCE<sup>3</sup></b>						
$V_{OUT}$ Settling Time	$t_S$	$\pm 1/2$ LSB Error Band		0.6		$\mu\text{s}$
Crosstalk	CT	Between Adjacent Outputs <sup>5</sup>		50		dB
<b>SWITCHING CHARACTERISTICS<sup>3, 6</sup></b>						
Input Clock Pulse Width	$t_{CH}$ , $t_{CL}$	Clock Level High or Low	15			ns
Data Setup Time	$t_{DS}$		5			ns
Data Hold Time	$t_{DH}$		5			ns
$\overline{\text{CS}}$ Setup Time	$t_{CSS}$		10			ns
CS High Pulse Width	$t_{CSW}$		10			ns
Reset Pulse Width	$t_{RS}$		90			ns
CLK Rise to $\overline{\text{CS}}$ Rise Hold Time	$t_{CSH}$		20			ns
$\overline{\text{CS}}$ Rise to Clock Rise Setup	$t_{CS1}$		10			ns

## NOTES

<sup>1</sup>Typicals represent average readings at  $+25^{\circ}\text{C}$ .

<sup>2</sup> $V_{REFH}$  can be any value between GND and  $V_{DD}$ , for the AD8804  $V_{REFL}$  can be any value between GND and  $V_{DD}$ .

<sup>3</sup>Guaranteed by design and not subject to production test.

<sup>4</sup>Digital Input voltages  $V_{IN} = 0\text{ V}$  or  $V_{DD}$  for CMOS condition. DAC outputs unloaded.  $P_{DISS}$  is calculated from  $(I_{DD} \times V_{DD})$ .

<sup>5</sup>Measured at a  $V_{OUT}$  pin where an adjacent  $V_{OUT}$  pin is making a full-scale voltage change ( $f = 100\text{ kHz}$ ).

<sup>6</sup>See timing diagram for location of measured values. All input control voltages are specified with  $t_R = t_F = 2\text{ ns}$  (10% to 90% of  $V_{DD}$ ) and timed from a voltage level of 1.6 V.

Specifications subject to change without notice.

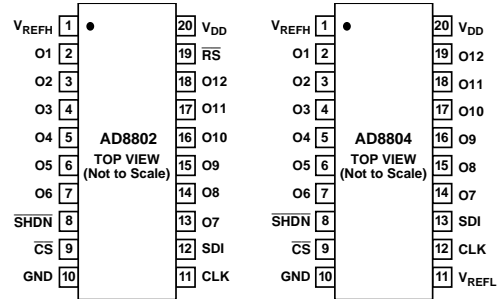
# AD8802/AD8804

## ABSOLUTE MAXIMUM RATINGS

(T<sub>A</sub> = +25°C, unless otherwise noted)

V <sub>DD</sub> to GND	-0.3, +8 V
V <sub>REFX</sub> to GND	0 V, V <sub>DD</sub>
Outputs (O <sub>x</sub> ) to GND	0 V, V <sub>DD</sub>
Digital Input Voltage to GND	0 V, +8 V
Operating Temperature Range	-40°C to +85°C
Maximum Junction Temperature (T <sub>J</sub> MAX)	+150°C
Storage Temperature	-65°C to +150°C
Lead Temperature (Soldering, 10 sec)	+300°C
Package Power Dissipation	(T <sub>J</sub> MAX - T <sub>A</sub> )/θ <sub>JA</sub>
Thermal Resistance θ <sub>JA</sub> ,	
SOIC (SOL-20)	60°C/W
P-DIP (N-20)	57°C/W
TSSOP-20 (RU-20)	155°C/W

## PIN CONFIGURATIONS



### AD8802 PIN DESCRIPTIONS

Pin	Name	Description
1	V <sub>REF</sub>	Common DAC Reference Input
2	O1	DAC Output #1, addr = 0000 <sub>2</sub>
3	O2	DAC Output #2, addr = 0001 <sub>2</sub>
4	O3	DAC Output #3, addr = 0010 <sub>2</sub>
5	O4	DAC Output #4, addr = 0011 <sub>2</sub>
6	O5	DAC Output #5, addr = 0100 <sub>2</sub>
7	O6	DAC Output #6, addr = 0101 <sub>2</sub>
8	SHDN	Reference input current goes to zero. DAC latch settings maintained
9	CS	Chip Select Input, Active Low. When CS returns high, data in the serial input register is decoded based on the address bits and loaded into the target DAC register
10	GND	Ground
11	CLK	Serial Clock Input, Positive Edge Triggered
12	SDI	Serial Data Input
13	O7	DAC Output #7, addr = 0110 <sub>2</sub>
14	O8	DAC Output #8, addr = 0111 <sub>2</sub>
15	O9	DAC Output #9, addr = 1000 <sub>2</sub>
16	O10	DAC Output #10, addr = 1001 <sub>2</sub>
17	O11	DAC Output #11, addr = 1010 <sub>2</sub>
18	O12	DAC Output #12, addr = 1011 <sub>2</sub>
19	RS	Asynchronous Preset to Midscale Output Setting. Loads all DAC Registers with 80 <sub>H</sub>
20	V <sub>DD</sub>	Positive Power Supply, Specified for Operation at Both +3 V and +5 V

### AD8804 PIN DESCRIPTIONS

Pin	Name	Description
1	V <sub>REFH</sub>	Common High-Side DAC Reference Input
2	O1	DAC Output #1, addr = 0000 <sub>2</sub>
3	O2	DAC Output #2, addr = 0001 <sub>2</sub>
4	O3	DAC Output #3, addr = 0010 <sub>2</sub>
5	O4	DAC Output #4, addr = 0011 <sub>2</sub>
6	O5	DAC Output #5, addr = 0100 <sub>2</sub>
7	O6	DAC Output #6, addr = 0101 <sub>2</sub>
8	SHDN	Reference input current goes to zero DAC latch settings maintained
9	CS	Chip Select Input, Active Low. When CS returns high, data in the serial input register is decoded based on the address bits and loaded input the target DAC register
10	GND	Ground
11	V <sub>REFL</sub>	Common Low-Side DAC Reference Input
12	CLK	Serial Clock Input, Positive Edge Triggered
13	SDI	Serial Data Input
14	O7	DAC Output #7, addr = 0110 <sub>2</sub>
15	O8	DAC Output #8, addr = 0111 <sub>2</sub>
16	O9	DAC Output #9, addr = 1000 <sub>2</sub>
17	O10	DAC Output #10, addr = 1001 <sub>2</sub>
18	O11	DAC Output #11, addr = 1010 <sub>2</sub>
19	O12	DAC Output #12, addr = 1011 <sub>2</sub>
20	V <sub>DD</sub>	Positive power supply, specified for operation at both +3 V and +5 V

### ORDERING GUIDE

Model	FTN	Temperature Range	Package Description	Package Option
AD8802AN	RS	-40°C/+85°C	PDIP-20	N-20
AD8802AR	RS	-40°C/+85°C	SOL-20	R-20
AD8802ARU	RS	-40°C/+85°C	TSSOP-20	RU-20
AD8804AN	REFL	-40°C/+85°C	PDIP-20	N-20
AD8804AR	REFL	-40°C/+85°C	SOL-20	R-20
AD8804ARU	REFL	-40°C/+85°C	TSSOP-20	RU-20

### CAUTION

ESD (electrostatic discharge) sensitive device. Electrostatic charges as high as 4000 V readily accumulate on the human body and test equipment and can discharge without detection. Although these devices feature proprietary ESD protection circuitry, permanent damage may occur on devices subjected to high energy electrostatic discharges. Therefore, proper ESD precautions are recommended to avoid performance degradation or loss of functionality.



# AD8802/AD8804—Typical Performance Characteristics

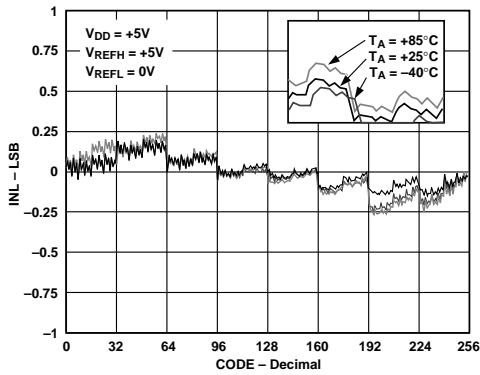


Figure 1. INL vs. Code

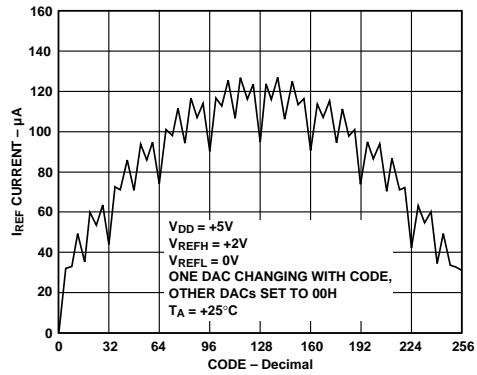


Figure 4. Input Reference Current vs. Code

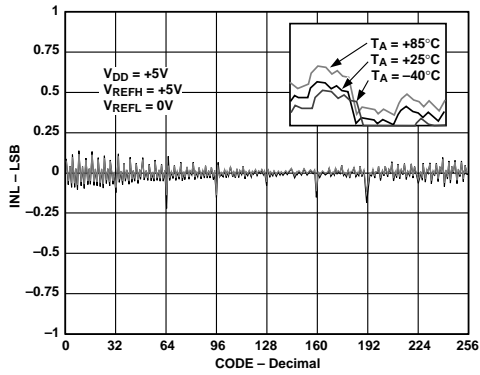


Figure 2. Differential Nonlinearity Error vs. Code

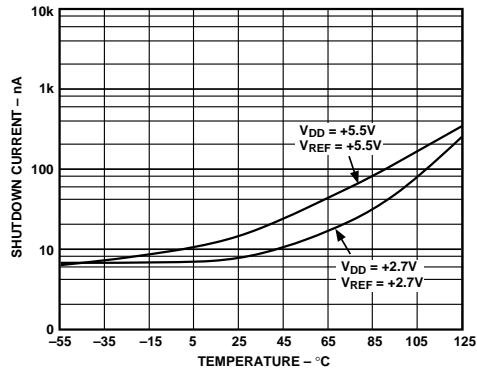


Figure 5. Shutdown Current vs. Temperature

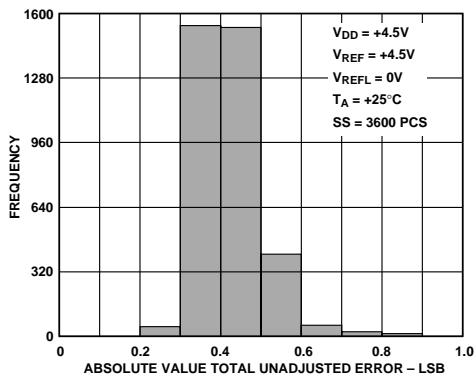


Figure 3. Total Unadjusted Error Histogram

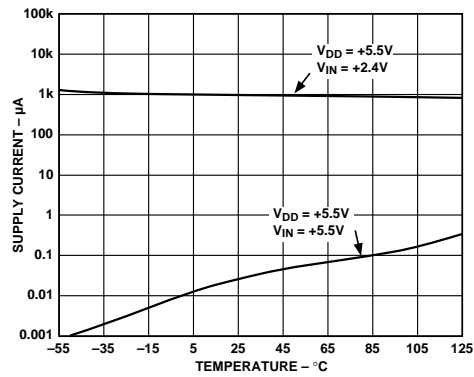


Figure 6. Supply Current vs. Temperature

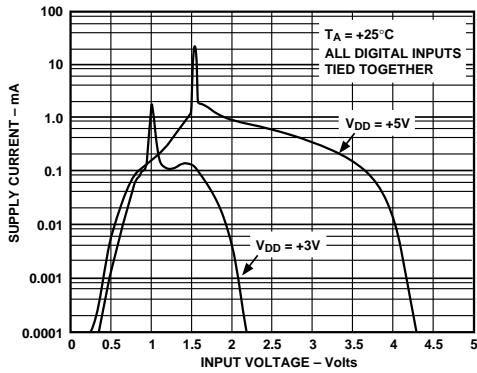


Figure 7. Supply Current vs. Logic Input Voltage

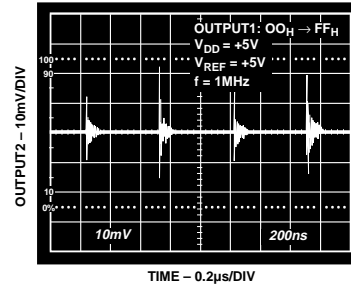


Figure 10. Adjacent Channel Clock Feedthrough

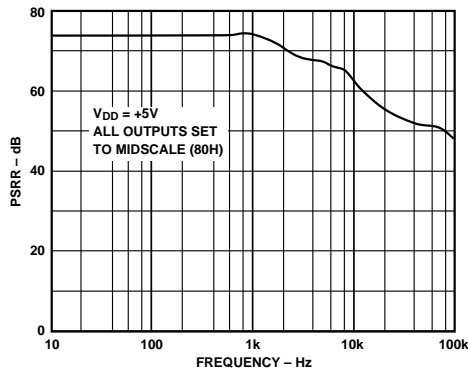


Figure 8. Power Supply Rejection vs. Frequency

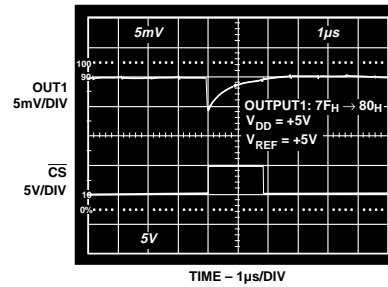


Figure 11. Midscale Transition

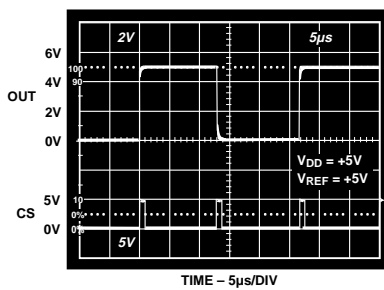


Figure 9. Large-Signal Settling Time

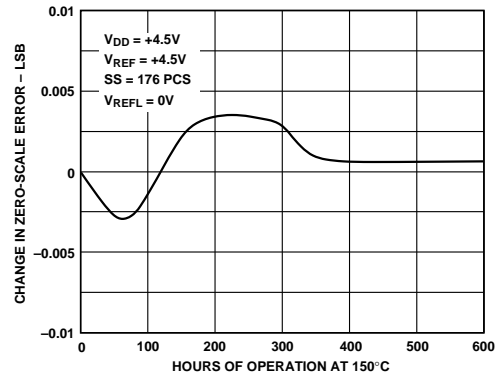


Figure 12. Zero-Scale Error Accelerated by Burn-In

# AD8802/AD8804

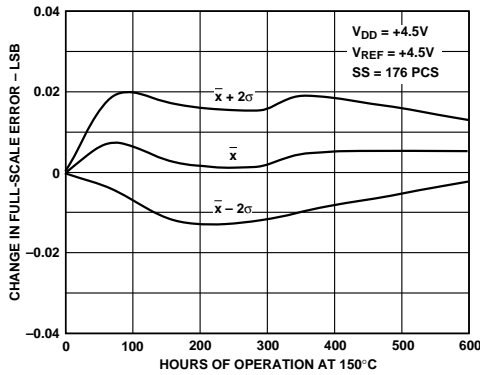


Figure 13. Full-Scale Error Accelerated by Burn-In

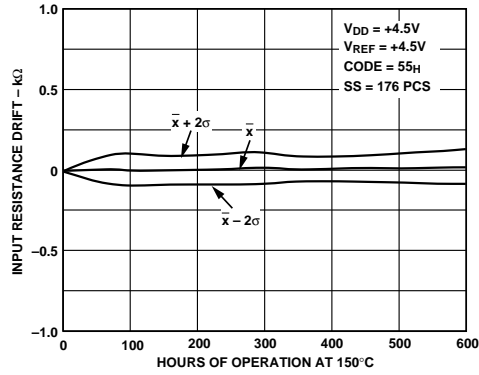


Figure 14. REF Input Resistance Accelerated by Burn-In

## OPERATION

The AD8802/AD8804 provides twelve channels of programmable voltage output adjustment capability. Changing the programmed output voltage of each DAC is accomplished by clocking in a 12-bit serial data word into the SDI (Serial Data Input) pin. The format of this data word is four address bits, MSB first, followed by 8 data bits, MSB first. Table I provides the serial register data word format. The AD8802/AD8804 has the following address assignments for the ADDR decode which determines the location of the DAC register receiving the serial register data in Bits B7 through B0:

$$DAC\# = A3 \times 8 + A2 \times 4 + A1 \times 2 + A0 + 1$$

DAC outputs can be changed one at a time in random sequence. The fast serial-data loading of 33 MHz makes it possible to load all 12 DACs in as little time as 4.6 μs (13 × 12 × 30 ns). The exact timing requirements are shown in Figure 15.

Table I. Serial-Data Word Format

ADDR				DATA							
B11	B10	B9	B8	B7	B6	B5	B4	B3	B2	B1	B0
A3	A2	A1	A0	D7	D6	D5	D4	D3	D2	D1	D0
MSB			LSB	MSB				LSB			
2 <sup>11</sup>	2 <sup>10</sup>	2 <sup>9</sup>	2 <sup>8</sup>	2 <sup>7</sup>	2 <sup>6</sup>	2 <sup>5</sup>	2 <sup>4</sup>	2 <sup>3</sup>	2 <sup>2</sup>	2 <sup>1</sup>	2 <sup>0</sup>

The AD8802 offers a midscale preset activated by the  $\overline{RS}$  pin simplifying initial setting conditions at first power-up. The AD8804 has both a  $V_{REFH}$  and a  $V_{REFL}$  pin to establish independent positive full-scale and zero-scale settings to optimize resolution. Both parts offer a power shutdown SHDN which places the DAC structure in a zero power consumption state resulting in only leakage currents being consumed from the power supply and  $V_{REF}$  inputs. In shutdown mode the DACX register settings are maintained. When returning to operational mode from power shutdown the DAC outputs return to their previous voltage settings.

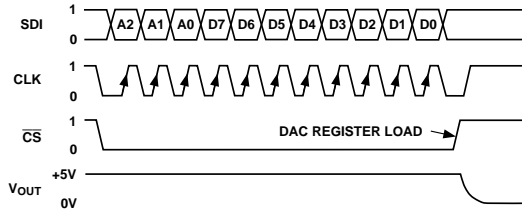


Figure 15a. Timing Diagram

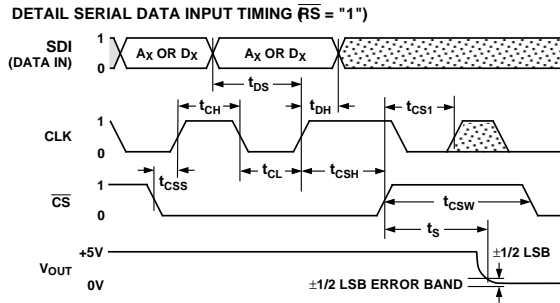


Figure 15b. Detail Timing Diagram

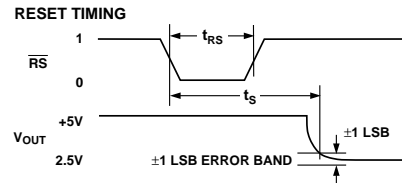


Figure 15c. Reset Timing Diagram

## PROGRAMMING THE OUTPUT VOLTAGE

The output voltage range is determined by the external reference connected to  $V_{REFH}$  and  $V_{REFL}$  pins. See Figure 16 for a simplified diagram of the equivalent DAC circuit. In the case of the AD8802 its  $V_{REFL}$  is internally connected to GND and therefore cannot be offset.  $V_{REFH}$  can be tied to  $V_{DD}$  and  $V_{REFL}$  can be tied to GND establishing a basic rail-to-rail voltage output programming range. Other output ranges are established by the use of different external voltage references. The general transfer equation which determines the programmed output voltage is:

$$VO(Dx) = (Dx)/256 \times (V_{REFH} - V_{REFL}) + V_{REFL} \quad \text{Eq. 1}$$

where  $Dx$  is the data contained in the 8-bit DACx register.

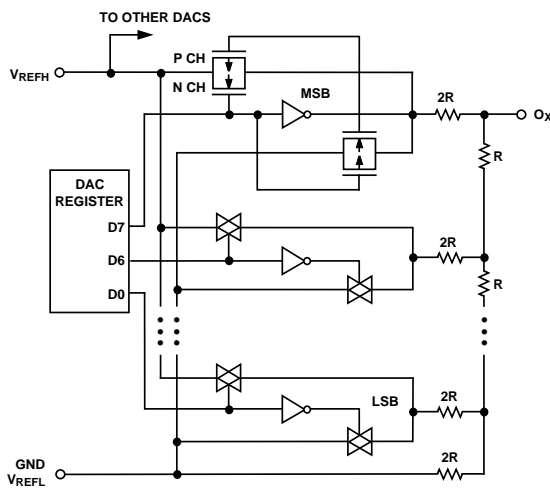


Figure 16. AD8802/AD8804 Equivalent TrimDAC Circuit

For example, when  $V_{REFH} = +5\text{ V}$  and  $V_{REFL} = 0\text{ V}$ , the following output voltages will be generated for the following codes:

D	VO <sub>x</sub>	Output State ( $V_{REFH} = +5\text{ V}$ , $V_{REFL} = 0\text{ V}$ )
255	4.98 V	Full Scale
128	2.50 V	Half Scale (Midscale Reset Value)
1	0.02 V	1 LSB
0	0.00 V	Zero Scale

## REFERENCE INPUTS ( $V_{REFH}$ , $V_{REFL}$ )

The reference input pins set the output voltage range of all twelve DACs. In the case of the AD8802 only the  $V_{REFH}$  pin is available to establish a user designed full-scale output voltage. The external reference voltage can be any value between 0 and  $V_{DD}$  but must not exceed the  $V_{DD}$  supply voltage. The AD8804 has access to the  $V_{REFL}$  which establishes the zero-scale output voltage, any voltage can be applied between 0 V and  $V_{DD}$ .  $V_{REFL}$  can be smaller or larger in voltage than  $V_{REFH}$  since the DAC design uses fully bidirectional switches as shown in Figure 16. The input resistance to the DAC has a code dependent variation which has a nominal worst case measured at  $55H$ , which is approximately 1.2 k $\Omega$ . When  $V_{REFH}$  is greater than  $V_{REFL}$ , the REFL reference must be able to sink current out of the DAC

ladder, while the REFH reference is sourcing current into the DAC ladder. The DAC design minimizes reference glitch current maintaining minimum interference between DAC channels during code changes.

## DAC OUTPUTS (O1–O12)

The twelve DAC outputs present a constant output resistance of approximately 5 k $\Omega$  independent of code setting. The distribution of  $R_{OUT}$  from DAC-to-DAC typically matches within  $\pm 1\%$ . However device-to-device matching is process lot dependent having a  $\pm 20\%$  variation. The change in  $R_{OUT}$  with temperature has a 500 ppm/ $^{\circ}\text{C}$  temperature coefficient. During power shutdown all twelve outputs are open-circuited.

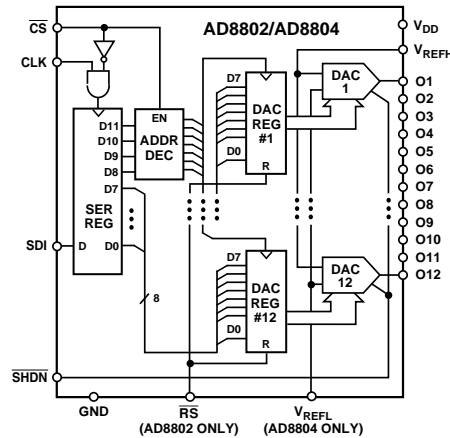


Figure 17. Block Diagram

## DIGITAL INTERFACING

The AD8802/AD8804 contains a standard three-wire serial input control interface. The three inputs are clock (CLK),  $\overline{\text{CS}}$  and serial data input (SDI). The positive-edge sensitive CLK input requires clean transitions to avoid clocking incorrect data into the serial input register. Standard logic families work well. If mechanical switches are used for product evaluation, they should be debounced by a flip-flop or other suitable means. Figure 17 block diagram shows more detail of the internal digital circuitry. When  $\overline{\text{CS}}$  is taken active low, the clock can load data into the serial register on each positive clock edge, see Table II.

Table II. Input Logic Control Truth Table

$\overline{\text{CS}}$	CLK	Register Activity
1	X	No effect.
0	P	Shifts Serial Register One bit loading the next bit in from the SDI pin.
P	1	Clock should be high when the $\overline{\text{CS}}$ returns to the inactive state.

P = Positive Edge, X = Don't Care.

The data setup and data hold times in the specification table determine the data valid time requirements. The last 12 bits of the data word entered into the serial register are held when  $\overline{\text{CS}}$  returns high. At the same time  $\overline{\text{CS}}$  goes high it gates the address decoder which enables one of the twelve positive-edge triggered DAC registers, see Figure 18 detail.

# AD8802/AD8804

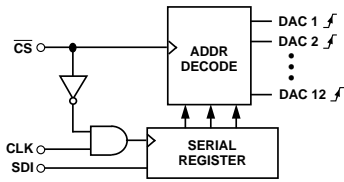


Figure 18. Equivalent Control Logic

The target DAC register is loaded with the last eight bits of the serial data-word completing one DAC update. Twelve separate 12-bit data words must be clocked in to change all twelve output settings.

All digital inputs are protected with a series input resistor and parallel Zener ESD structure shown in Figure 19. Applies to digital input pins CS, SDI, RS, SHDN, CLK

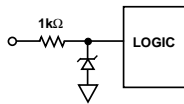


Figure 19. Equivalent ESD Protection Circuit

Digital inputs can be driven by voltages exceeding the AD8802/AD8804  $V_{DD}$  supply value. This allows 5 V logic to interface directly to the part when it is operated at 3 V.

## APPLICATIONS

### Supply Bypassing

Precision analog products, such as the AD8802/AD8804, require a well filtered power source. Since the AD8802/AD8804 operate from a single +3 V to +5 V supply, it seems convenient to simply tap into the digital logic power supply. Unfortunately, the logic supply is often a switch-mode design, which generates noise in the 20 kHz to 1 MHz range. In addition, fast logic gates can generate glitches hundreds of millivolts in amplitude due to wiring resistances and inductances.

If possible, the AD8802/AD8804 should be powered directly from the system power supply. This arrangement, shown in Figure 20, will isolate the analog section from the logic switching transients. Even if a separate power supply trace is not available, however, generous supply bypassing will reduce supply-line induced errors. Local supply bypassing consisting of a 10  $\mu$ F tantalum electrolytic in parallel with a 0.1  $\mu$ F ceramic capacitor is recommended (Figure 21).

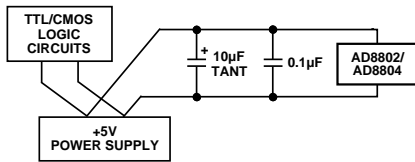


Figure 20. Use Separate Traces to Reduce Power Supply Noise

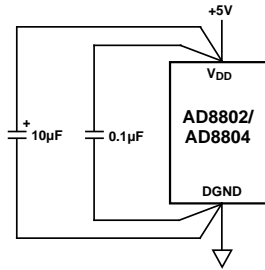


Figure 21. Recommended Supply Bypassing for the AD8802/AD8804

### Buffering the AD8802/AD8804 Output

In many cases, the nominal 5 k $\Omega$  output impedance of the AD8802/AD8804 is sufficient to drive succeeding circuitry. If a lower output impedance is required, an external amplifier can be added. Several examples are shown in Figure 22. One amplifier of an OP291 is used as a simple buffer to reduce the output resistance of DAC A. The OP291 was chosen primarily for its rail-to-rail input and output operation, but it also offers operation to less than 3 V, low offset voltage, and low supply current.

The next two DACs, B and C, are configured in a summing arrangement where DAC C provides the coarse output voltage setting and DAC B can be used for fine adjustment. The insertion of R1 in series with DAC B attenuates its contribution to the voltage sum node at the DAC C output.

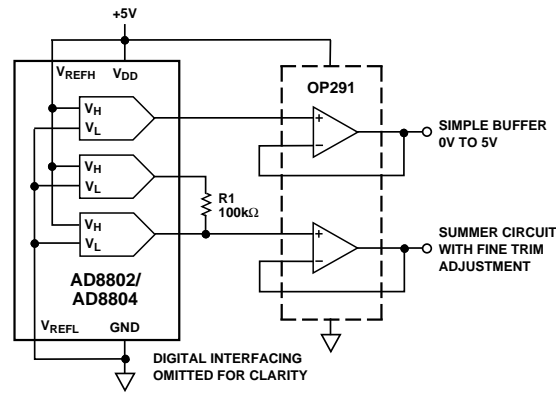


Figure 22. Buffering the AD8802/AD8804 Output

### Increasing Output Voltage Swing

An external amplifier can also be used to extend the output voltage swing beyond the power supply rails of the AD8802/AD8804. This technique permits an easy digital interface for the DAC, while expanding the output swing to take advantage of higher voltage external power supplies. For example, DAC A of Figure 23 is configured to swing from -5 V to +5 V. The actual output voltage is given by:

$$V_{OUT} = \left(1 + \frac{R_F}{R_S}\right) \times \left(\frac{D}{256} \times 5V\right) - 5V$$

where D is the DAC input value (i.e., 0 to 255). This circuit can be combined with the “fine/coarse” circuit of Figure 22 if, for example, a very accurate adjustment around 0 V is desired.



## AD8802/AD8804

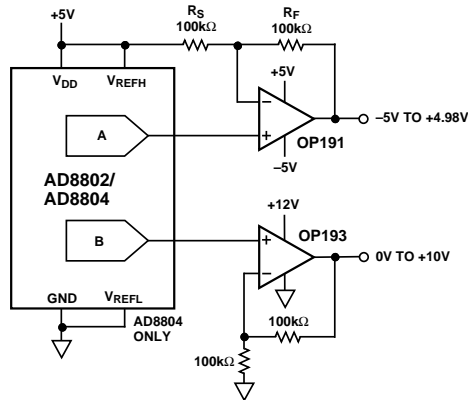


Figure 23. Increasing Output Voltage Swing

DAC B of Figure 24 is in a noninverting gain of two configurations, which increases the available output swing to +10 V. The feedback resistors can be adjusted to provide any scaling of the output voltage, within the limits of the external op amp power supplies.

### Microcomputer Interfaces

The AD8802/AD8804 serial data input provides an easy interface to a variety of single-chip microcomputers ( $\mu$ Cs). Many  $\mu$ Cs have a built-in serial data capability that can be used for communicating with the DAC. In cases where no serial port is provided, or it is being used for some other purpose (such as an RS-232 communications interface), the AD8802/AD8804 can easily be addressed in software.

Twelve data bits are required to load a value into the AD8802/AD8804 (4 bits for the DAC address and 8 bits for the DAC value). If more than 12 bits are transmitted before the Chip Select input goes high, the extra (i.e., the most-significant) bits are ignored. This feature is valuable because most  $\mu$ Cs only transmit data in 8-bit increments. Thus, the  $\mu$ C will send 16 bits to the DAC instead of 12 bits. The AD8802/AD8804 will only respond to the last 12 bits clocked into the SDI port, however, so the serial data interface is not affected.

### An 8051 $\mu$ C Interface

A typical interface between the AD8802/AD8804 and an 8051  $\mu$ C is shown in Figure 24. This interface uses the 8051's internal serial port. The serial port is programmed for Mode 0 operation, which functions as a simple 8-bit shift register. The 8051's Port 3.0 pin functions as the serial data output, while Port 3.1 serves as the serial clock.

When data is written to the Serial Buffer Register (SBUF, at Special Function Register location 99H), the data is automatically converted to serial format and clocked out via Port 3.0 and Port 3.1. After 8 bits have been transmitted, the Transmit Interrupt flag (SCON.1) is set and the next 8 bits can be transmitted.

The AD8802 and AD8804 require the Chip Select to go low at the beginning of the serial data transfer. In addition, the SCLK input must be high when the Chip Select input goes high at the end of the transfer. The 8051's serial clock meets this requirement, since Port 3.1 both begins and ends the serial data in the high state.

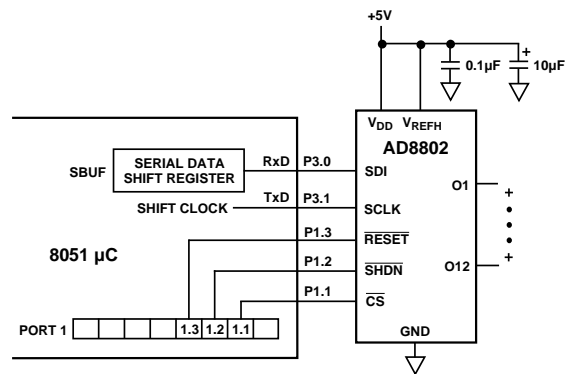


Figure 24. Interfacing the 8051  $\mu$ C to an AD8802/AD8804, Using the Serial Port

### Software for the 8051 Interface

A software for the AD8802/AD8804 to 8051 interface is shown in Listing 1. The routine transfers the 8-bit data stored at data memory location DAC\_VALUE to the AD8802/AD8804 DAC addressed by the contents of location DAC\_ADDR.

The subroutine begins by setting appropriate bits in the Serial Control register to configure the serial port for Mode 0 operation. Next the DAC's Chip Select input is set low to enable the AD8802/AD8804. The DAC address is obtained from memory location DAC\_ADDR, adjusted to compensate for the 8051's serial data format, and moved to the serial buffer register. At this point, serial data transmission begins automatically. When all 8 bits have been sent, the Transmit Interrupt bit is set, and the subroutine then proceeds to send the DAC value stored at location DAC\_VALUE. Finally the Chip Select input is returned high, causing the appropriate AD8802/AD8804 output voltage to change, and the subroutine ends.

The 8051 sends data out of its shift register LSB first, while the AD8802/AD8804 require data MSB first. The subroutine therefore includes a BYTESWAP subroutine to reformat the data. This routine transfers the MSB-first byte at location SHIFT1 to an LSB-first byte at location SHIFT2. The routine rotates the MSB of the first byte into the carry with a Rotate Left Carry instruction, then rotates the carry into the MSB of the second byte with a Rotate Right Carry instruction. After 8 loops, SHIFT2 contains the data in the proper format.

The BYTESWAP routine in Listing 1 is convenient because the DAC data can be calculated in normal LSB form. For example, producing a ramp voltage on a DAC is simply a matter of repeatedly incrementing the DAC\_VALUE location and calling the LD\_8802 subroutine.

If the  $\mu$ C's hardware serial port is being used for other purposes, the AD8802/AD8804 DAC can be loaded by using the parallel port. A typical parallel interface is shown in Figure 25. The serial data is transmitted to the DAC via the 8051's Port 1.6 output, while Port 1.6 acts as the serial clock.

Software for the interface of Figure 25 is contained in Listing 2. The subroutine will send the value stored at location DAC\_VALUE to the AD8802/AD8804 DAC addressed by location DAC\_ADDR. The program begins by setting the AD8802/AD8804's Serial Clock and Chip Select inputs high, then setting Chip Select low

# AD8802/AD8804

```

;
; This subroutine loads an AD8802/AD8804 DAC from an 8051 microcomputer,
; using the 8051's serial port in MODE 0 (Shift Register Mode).
; The DAC value is stored at location DAC_VAL
; The DAC address is stored at location DAC_ADDR
;
; Variable declarations
;
PORT1          DATA          90H          ;SFR register for port 1
DAC_VALUE      DATA          40H          ;DAC Value
DAC_ADDR       DATA          41H          ;DAC Address
SHIFT1         DATA          042H        ;high byte of 16-bit answer
SHIFT2         DATA          043H        ;low byte of answer
SHIFT_COUNT    DATA          44H        ;
;
DO_8802:       ORG             100H        ;arbitrary start
               CLR             SCON.7     ;set serial
               CLR             SCON.6     ;data mode 0
               CLR             SCON.5
               CLR             SCON.1     ;clr transmit flag
               ORL             PORT1.1, #00001110B ;/RS, /SHDN, /CS high
               CLR             PORT1.1   ;set the /CS low
               MOV             SHIFT1, DAC_ADDR ;put DAC value in shift register
               ACALL          BYTESWAP    ;
               MOV             SBUF, SHIFT2 ;send the address byte
ADDR_WAIT:     JNB             SCON.1, ADDR_WAIT ;wait until 8 bits are sent
               CLR             SCON.1     ;clear the serial transmit flag
               MOV             SHIFT1, DAC_VALUE ;send the DAC value
               ACALL          BYTESWAP    ;
               MOV             SBUF, SHIFT2 ;
               JNB             SCON.1, VALU_WAIT ;wait again
VALU_WAIT:     CLR             SCON.1     ;clear serial flag
               SETB            PORT1.1   ;/CS high, latch data
               RET              ; into AD8801
;
BYTESWAP:     MOV             SHIFT_COUNT, #8 ;Shift 8 bits
SWAP_LOOP:    MOV             A, SHIFT1   ;Get source byte
               RLC              A         ;Rotate MSB to carry
               MOV             SHIFT1, A  ;Save new source byte
               MOV             A, SHIFT2  ;Get destination byte
               RRC              A         ;Move carry to MSB
               MOV             SHIFT2, A  ;Save
               DJNZ            SHIFT_COUNT, SWAP_LOOP ;Done?
               RET
               END

```

Listing 1. Software for the 8051 to AD8802/AD8804 Serial Port Interface

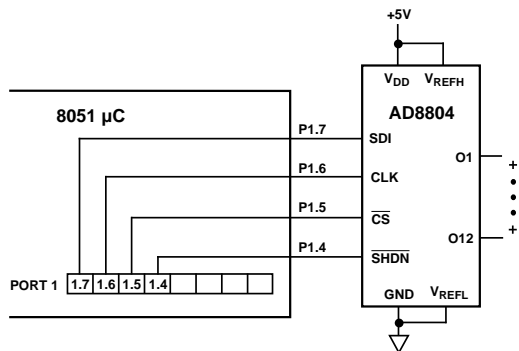


Figure 25. An AD8802/AD8804-8051  $\mu$ C Interface Using Parallel Port 1

to start the serial interface process. The DAC address is loaded into the accumulator and four Rotate Right shifts are performed. This places the DAC address in the 4 MSBs of the accumulator. The address is then sent to the AD8802/AD8804 via the SEND\_SERIAL subroutine. Next, the DAC value is loaded into the accumulator and sent to the AD8802/AD8804. Finally, the Chip Select input is set high to complete the data transfer

Unlike the serial port interface of Figure 24, the parallel port interface only transmits 12 bits to the AD8802/AD8804. Also, the BYTESWAP subroutine is not required for the parallel interface, because data can be shifted out MSB first. However, the results of the two interface methods are exactly identical. In most cases, the decision on which method to use will be determined by whether or not the serial data port is available for communication with the AD8802/AD8804.

```

; This 8051 µC subroutine loads an AD8802 or AD8804 DAC with an 8-bit value,
; using the 8051's parallel port #1.
; The DAC value is stored at location DAC_VALUE
; The DAC address is stored at location DAC_ADDR
;
; Variable declarations
PORT1          DATA          90H          ;SFR register for port 1
DAC_VALUE      DATA          40H          ;DAC Value
DAC_ADDR       DATA          41H          ;DAC Address (0 through 7)
LOOPCOUNT    DATA          43H          ;COUNT LOOPS
;
LD_8804:      ORG              100H        ;arbitrary start
              ORL              PORT1,#11110000B ;set CLK, /CS and /SHDN high
              CLR              PORT1.5     ;Set Chip Select low
              MOV              LOOPCOUNT,#4 ;Address is 4 bits
              MOV              A,DAC_ADDR  ;Get DAC address
              RR               A           ;Rotate the DAC
              RR               A           ;address to the Most
              RR               A           ;Significant Bits (MSBs)
              RR               A           ;
              ACALL            SEND_SERIAL ;Send the address
              MOV              LOOPCOUNT,#8 ;Do 8 bits of data
              MOV              A,DAC_VALUE
              ACALL            SEND_SERIAL ;Send the data
              SETB             PORT1.5     ;Set /CS high
              RET              ;DONE

SEND_SERIAL:  RLC               A           ;Move next bit to carry
              MOV              PORT1.7,C   ;Move data to SDI
              CLR              PORT1.6     ;Pulse the
              SETB             PORT1.6     ;CLK input
              DJNZ             LOOPCOUNT,SEND_SERIAL ;Loop if not done
              RET
END

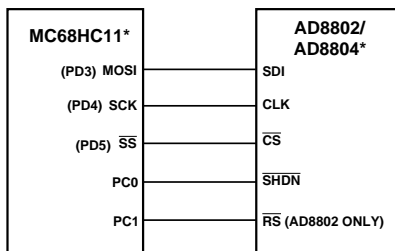
```

*Listing 2. Software for the 8051 to AD8802/AD8804 Parallel Port Interface*

### An MC68HC11-to-AD8802/AD8804 Interface

Like the 8051 µC, the MC68HC11 includes a dedicated serial data port (labeled SPI). The SPI port provides an easy interface to the AD8802/AD8804 (Figure 27). The interface uses three lines of Port D for the serial data, and one or two lines from Port C to control the  $\overline{\text{SHDN}}$  and  $\overline{\text{RS}}$  (AD8802 only) inputs.

A software routine for loading the AD8802/AD8804 from a 68HC11 evaluation board is shown in Listing 3. First, the MC68HC11 is configured for SPI operation. Bits CPHA and CPOL define the SPI mode wherein the serial clock (SCK) is high at the beginning and end of transmission, and data is valid on the rising edge of SCK. This mode matches the requirements of the AD8802/AD8804. After the registers are saved on the stack, the DAC value and address are transferred to RAM and the AD8802/AD8804's  $\overline{\text{CS}}$  is driven low. Next, the DAC's address byte is transferred to the SPDR register, which automatically initiates the SPI data transfer. The program tests the SPIF bit and loops until the data transfer is complete. Then the DAC value is sent to the SPI. When transmission of the second byte is complete,  $\overline{\text{CS}}$  is driven high to load the new data and address into the AD8802/AD8804.



\*ADDITIONAL PINS OMITTED FOR CLARITY

*Figure 26. An AD8802/AD8804-to-MC68HC11 Interface*

# AD8802/AD8804

```

*
* AD8802/AD8804 to M68HC11 Interface Assembly Program
*
* M68HC11 Register definitions
*
PORTC      EQU          $1003      Port C control register
*                                     "0,0,0,0;0,0,RS/, SHDN/"
DDRC       EQU          $1007      Port C data direction
PORTD      EQU          $1008      Port D data register
*                                     "0,0,/CS,CLK;SDI,0,0,0"
DDRD       EQU          $1009      Port D data direction
SPCR       EQU          $1028      SPI control register
*                                     "SPIE,SPE,DWOM,MSTR;CPOL,CPHA,SPR1,SPR0"
SPSR       EQU          $1029      SPI status register
*                                     "SPIF,WCOL,0,MODF;0,0,0,0"
SPDR       EQU          $102A      SPI data register; Read-Buffer; Write-Shifter
*
* SDI RAM variables:
*                                     SDI1 is encoded from 0H to 7H
*                                     SDI2 is encoded from 00H to FFH
*                                     AD8802/AD8804 requires two 8-bit loads; upper 4 bits
*                                     of SDI1 are ignored. AD8802/AD8804 address bits in last
*                                     four LSBs of SDI1.
SDI1       EQU          $00        SDI packed byte 1 "0,0,0,0;A3,A2,A1,A0"
SDI2       EQU          $01        SDI packed byte 2 "DB7-DB4;DB3-DB0"
*
* Main Program
*
INIT        ORG          $C000      Start of user's RAM in EVB
           LDS          #$CFFF      Top of C page RAM
*
* Initialize Port C Outputs
*
           LDAA         #$03        0,0,0,0;0,0,1,1
*                                     /RS-Hi, /SHDN-Hi
           STAA         PORTC       Initialize Port C Outputs
           LDAA         #$03        0,0,0,0;0,0,1,1
           STAA         DDRC        /RS and /SHDN are now enabled as outputs
*
* Initialize Port D Outputs
*
           LDAA         #$20        0,0,1,0;0,0,0,0
*                                     /CS-Hi,/CLK-Lo,SDI-Lo
           STAA         PORTD       Initialize Port D Outputs
           LDAA         #$38        0,0,1,1;1,0,0,0
           STAA         DDRD        /CS,CLK, and SDI are now enabled as outputs
*
* Initialize SPI Interface
*
           LDAA         #$53
           STAA         SPCR        SPI is Master,CPHA=0,CPOL=0,Clk rate=E/32
*
* Call update subroutine
*
           BSR          UPDATE      Xfer 2 8-bit words to AD8402
           JMP          $E000       Restart BUFFALO
*
* Subroutine UPDATE
*
UPDATE     PSHX          Save registers X, Y, and A
           PSHY
           PSHA
*
* Enter Contents of SDI1 Data Register

```

```

*
      LDAA      $0000      Hi-byte data loaded from memory
      STAA      SDI1      SDI1 = data in location 0000H
*
* Enter Contents of SDI2 Data Register
*
      LDAA      $0001      Low-byte data loaded from memory
      STAA      SDI2      SDI2 = Data in location 0001H
*
      LDX       #SDI1      Stack pointer at 1st byte to send via SDI
      LDY       #$1000    Stack pointer at on-chip registers
*
* Reset AD8802 to one-half scale (AD8804 does not have a Reset input)
*
      BCLR      PORTC,Y $02  Assert /RS
      BSET      PORTC,Y $02  De-Assert /RS
*
* Get AD8802/04 ready for data input
*
      BCLR      PORTD,Y $02  Assert /CS
TFRLP  LDAA      0,X        Get a byte to transfer for SPI
      STAA      SPDR        Write SDI data reg to start xfer
*
WAIT    LDAA      SPSR      Loop to wait for SPIF
      BPL       WAIT       SPIF is the MSB of SPSR
*
      INX
      CPX       #SDI2+1    Increment counter to next byte for xfer
      BNE      TFRLP      Are we done yet ?
                          If not, xfer the second byte
*
* Update AD8802 output
*
      BSET      PORTD,Y $20  Latch register & update AD8802
*
      PULA
      PULY
      PULX
      RTS
** Return to Main Program **

```

Listing 3. AD8802/AD8804 to MC68HC11 Interface Program Source Code

**An Intelligent Temperature Control System—Interfacing the 8051  $\mu$ C with the AD8802/AD8804 and TMP14**

Connecting the 80CL51  $\mu$ C, or any modern microcontroller, with the TMP14 and AD8802/AD8804 yields a powerful temperature control tool, as shown in Figure 27. For example, the 80CL51  $\mu$ C controls the TrimDACs allowing the user to automatically set the temperature setpoints voltages of the TMP14 via computer or touch pad, while the TMP14 senses the temperature and outputs four open-collector trip-points. Feeding these trip-point outputs back to the 80CL51  $\mu$ C allow it to sense whether or not a setpoint has been exceeded. Additional 80CL51  $\mu$ C port pins or TMP14 trip-point outputs may then be used to change fan speed (i.e., high, medium, low, off), or increase/decrease the power level to a heater. (Please refer to the TMP14 data sheet for more applications information.)

The  $\overline{\text{CS}}$  (Chip Select) on the AD8802/AD8804 makes applications that call for large temperature sensor arrays possible. In addition, the 12 channels of the AD8802/AD8804 allow independent setpoint control for all four trip-point outputs on up to three TMP14 temperature sensors. For example, assume that the 80CL51  $\mu$ C has eight free port pins available after all user

interface lines, interrupts, and the serial port lines have been assigned. The eight port pins may be used as chip selects, in which case an array of eight AD8802/AD8804s controlling twenty-four TMP14 sensors is possible.

The AD8802/AD8804 and TMP14 are also ideal choices for low power applications. These devices have power shutdown modes and operate on a single 5 Volt supply. When their shutdown modes are activated current consumption is reduced to less than 35  $\mu$ A. However, at high operating frequencies (12 MHz) the 80CL51 consumes far more energy (18 mA typ) than the AD8802/AD8804 and TMP14 combined. Therefore, to achieve a low power design the 80CL51 should operate at its lowest possible frequency or be placed in its power-down mode at the end of each instruction sequence.

To use the power-down mode of the 80CL51  $\mu$ C set PCON.1 as the last instruction executed prior to going into the power-down mode. If INT2 and INT9 are enabled, the 80CL51  $\mu$ C can be awakened from power-down mode with external interrupts. As shown in Figure 28, the TLC555 outputs a pulse every few seconds providing the interrupt to restart the 80CL51  $\mu$ C which then samples the user input pins, the outputs of the

# AD8802/AD8804

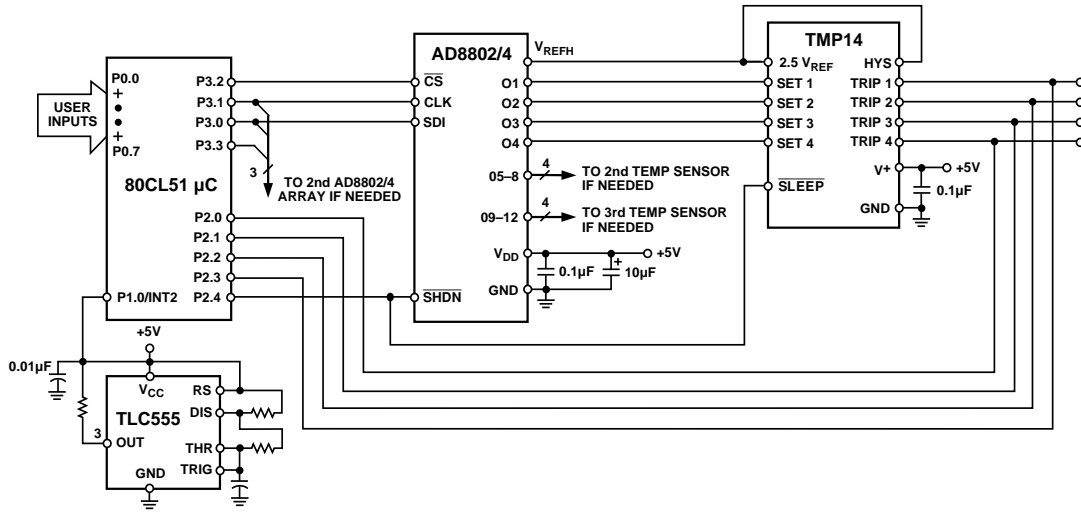


Figure 27. Temperature Sensor Array with Programmable Setpoints

TMP14, and makes the necessary adjustments to the AD8802/AD8804 before shutting down again. The 80CL51 consumes only 50 µA when operating at 32 kHz, in which case there would be no need for the TLC555, which consumes 1 mW typ.

## 12 Channel Programmable Voltage Controlled Amplifier

The SSM2018T is a trimless Voltage Controlled Amplifier (VCA) for volume control in audio systems. The SSM2018T is the first professional quality audio VCA in the marketplace that does not require an external trimming potentiometer to minimize distortion. The TrimDAC shown in Figure 28 is not being used to trim distortion, but rather to control the gain of the amplifier. In this configuration up to twelve SSM2018T can be digitally controlled. (Please refer to the SSM2018T data sheet for more specifications and applications information.)

The gain of the SSM2018T is controlled by the voltage at Pin 11. For maximum attenuation of -100 dB a control signal of 3.0 V typ is necessary. The control signal has a scale of -30 mV/dB centered around 0 dB gain for 0 V of control voltage, therefore, for a maximum gain of 40 dB a control voltage of -1.2 volts is necessary. Now notice that the normal +5 V to GND voltage range of the AD8802/AD8804 does not cover the 3.0 V to -1.2 V operational gain control range of the SSM2018T. To cover the operating gain range fully and not exceed the maximum specified power supply rating requires the O1 output of AD8802/AD8804 to be level shifted down. In Figure 28, the level shifting is accomplished by a Zener diode and 1/4 of an OP420 quad op amp. For applications that require only

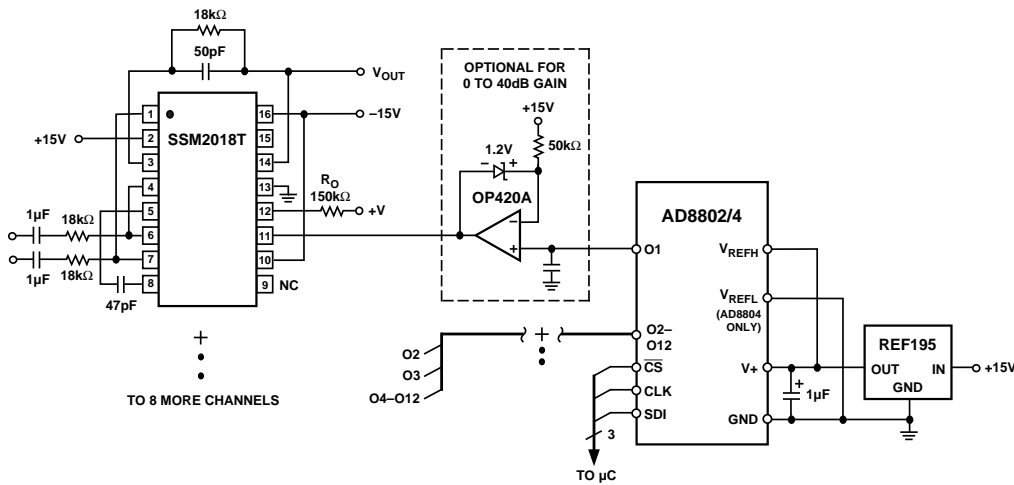


Figure 28. 12-Channel Programmable Voltage Controlled Amplifier



# AD8802/AD8804

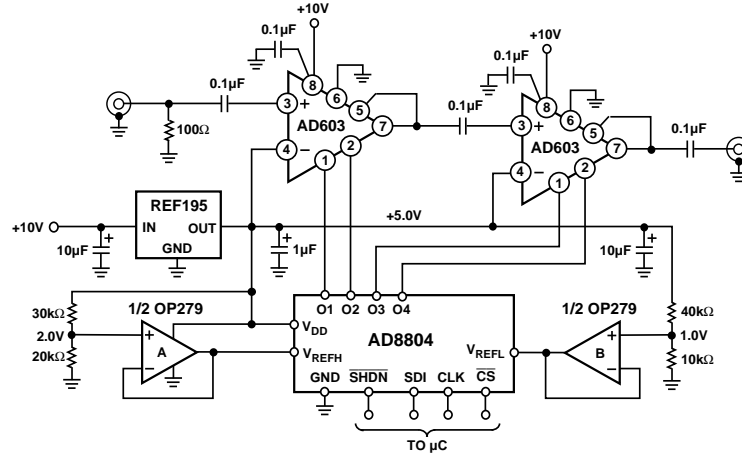
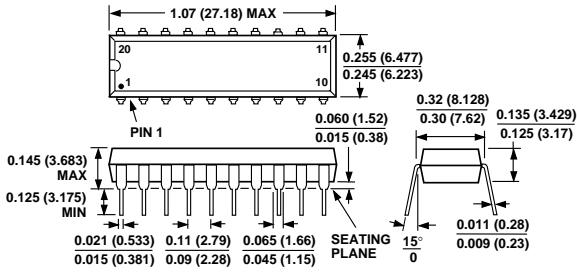


Figure 30. A Low Noise 90 MHz PGA

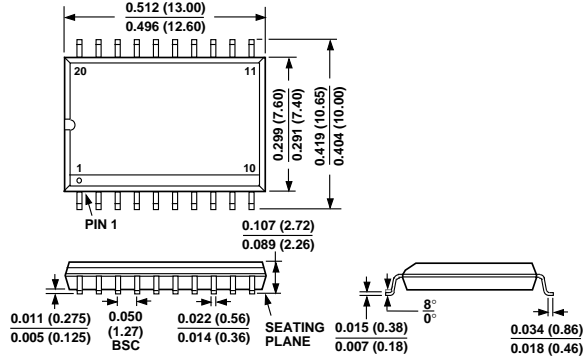
## OUTLINE DIMENSIONS

Dimensions shown in inches and (mm)

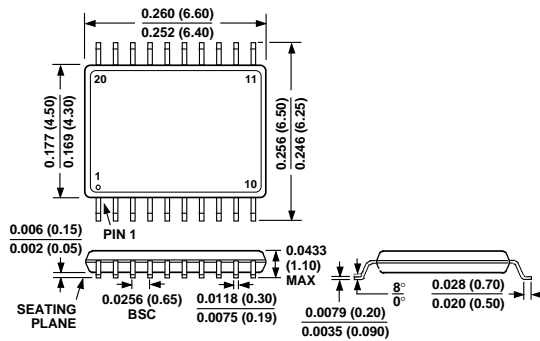
### 20-Pin Plastic DIP Package (N-20)



### 20-Lead SOIC Package (R-20)



### 20-Lead Thin Surface Mount TSSOP Package (RU-20)



C2052-10-7/95

PRINTED IN U.S.A.