6811D MODULE AVAILABLE

MAXIM

MAX1202 Evaluation Kit

Evaluates: MAX1202/MAX1203

_____General Description

The MAX1202 evaluation system (EV system) is a complete, 8-channel data-acquisition system consisting of a MAX1202 evaluation kit (EV kit) and a Maxim 3V 68L11D microcontroller (µC) module.

The MAX1202 is a low-power, +5V, 8-channel, 12-bit analog-to-digital converter (ADC) that connects directly to 3V and 5V microprocessors (µPs). Windows 3.1™/ Windows 95™ software provides a handy user interface to exercise the MAX1202's features.

Order the EV system for comprehensive evaluation of the MAX1202 using a personal computer. Order the EV kit if you have already purchased the 68L11D µC module with a previous Maxim EV system, or for custom use in other µC-based systems.

The MAX1202 EV kit and EV system can also be used to evaluate the MAX1203. Simply order a free sample of the MAX1203BCPP along with the MAX1202 EV kit. For 3V-only applications, refer to the MAX147 data sheet.

_____Component List

DESIGNATION	QTY	DESCRIPTION
C1–C8, C13	9	0.01µF ceramic capacitors
C9, C10, C11, C14, C15	5	0.1µF ceramic capacitors
C12	1	4.7µF, 10V tantalum capacitor
J1	1	2 x 20 right-angle socket
JU1, JU2	2	3-pin jumper blocks
JU3	1	2-pin jumper block
JU4	0	Open
R1–R8	8	300Ω, 5% resistors
TP1	1	8-pin header
U1	1	Maxim MAX1202BCPP
	1	20-pin socket
U2	1	Maxim ICL7660CPA
U3	1	78L05 voltage regulator
None	1	PC board
None	1	Software disk, "MAX1202 EVALUATION KIT"

_____Features

♦ **Proven PC Board Layout**

♦ **Complete Evaluation System**

♦ **Convenient On-Board Test Points**

♦ **Data-Logging Software**

♦ **3V/5V Logic Interface**

♦ **Fully Assembled and Tested**

_____Ordering Information

PART	TEMP. RANGE	BOARD TYPE
MAX1202EVKIT-DIP	0°C to +70°C	Through-Hole
MAX1202EVL11-DIP	0°C to +70°C	Through-Hole

MAX1202EVL11
_____System Component List

PART	QTY	DESCRIPTION
MAX1202EVKIT-DIP	1	MAX1202 Evaluation Kit
68L11DMODULE	1	68L11D µC Module

_____MAX1202 Stand-Alone EV Kit

The MAX1202 EV kit provides a proven PC board layout to facilitate evaluation of the MAX1202. It must be interfaced to appropriate timing signals for proper operation. Refer to the MAX1202 data sheet for timing requirements.

Cut JU4 and apply the +5V analog supply between the +5V pad and the pad marked GND. Connect the VLOGIC pad to the microprocessor's power supply. Set JU1 to the 2-3 position (V_{SS} = -5V) to allow input signals between ±4V (Table 1).

Windows is a registered trademark of Microsoft Corporation.

MAX1202 Evaluation Kit

_____MAX1202 EV System

The MAX1202 EV system operates from a user-supplied 9V to 15V DC power supply. The Maxim 68L11D 3V µC board uses a MAX667 linear regulator to generate the 3V logic supply. The MAX1202 board uses a 78L05 linear regulator to generate its own 5V analog supply. No level translators are necessary because the MAX1202 VL pin is connected to the 3V logic supply.

Quick Start

1) Install the MAX1202 EV kit software on your computer by running the *INSTALL.EXE* program on the floppy disk. The Windows 3.1 Program Manager (or the Windows 95 Start Menu) copies the program files and creates icons for them.

2) Check the jumper settings on the EV board. Refer to Tables 1 and 2.

3) Carefully connect the boards by aligning the EV kit's 40-pin header with the µC module's 40-pin connector. Gently press them together. The two boards should be flush against one another.

4) Connect a 9V to 15V DC power source to the µC module at the terminal block located next to the on/off switch, in the upper-right corner of the µC module. Observe the polarity marked on the board.

5) Connect a cable from the computer's serial port to the µC module. If using a 9-pin serial port, use a straight-through, 9-pin female-to-male cable. If the only available serial port uses a 25-pin connector, a standard 25-pin to 9-pin adapter is required. The EV kit software checks the modem status lines (CTS, DSR, DCD) to confirm that the correct port has been selected.

6) Start the MAX1202 program by opening its icon in the Program Manager (or Start Menu).

7) The program will prompt you to connect the µC module and turn its power on. Slide SW1 to the on position. Select the correct serial port and click OK. The program automatically downloads *KIT1202.L11* to the module.

8) Apply input signals to the CH0–CH7 inputs at the right edge of the MAX1202 EV board. Observe the readout on the screen.

Table 1. Jumper Functions

JUMPER	STATE	FUNCTION
JU1	1-2	V_{SS} tied to GND
	2-3	V_{SS} tied to -5V
	Open	V_{SS} must be supplied by the user.
JU2	1-2	$\overline{SHDN}$ tied to GND; power-down
	2-3	$\overline{SHDN}$ tied to +5V; internal compensation
	Open	$\overline{SHDN}$ floating; external compensation
JU3	Closed	REFADJ = +5V; VREF must be supplied by the user.
	Open	REFADJ = open; VREF = 4.096V internal reference (MAX1202)
JU4	Closed (default trace)	Current-sense jumper. The MAX1202 draws its +5V analog supply through this trace.
	Open	Do not operate kit with JU4 open.

Table 2. Default Jumper Settings

JUMPER	DEFAULT STATE	FUNCTION
JU1	2-3	V_{SS} tied to -5V
JU2	2-3	$\overline{SHDN}$ tied to +5V
JU3	Open	REFADJ = open; VREF = 4.096V internal reference (MAX1202)
JU4	Closed	The MAX1202 draws its +5V analog supply through this trace.

Evaluating the MAX1203

To evaluate the MAX1203, turn off the power to the EV kit, close JU3, and replace U1 with a MAX1203BCPP. Connect the external voltage reference to the VREF pad. No other hardware changes are necessary. Refer to the section *Changing the Reference Voltage.*

MAXIM

__Detailed Description of Software

Shutdown Power Cycling

The MAX1202 can be shut down between conversions to reduce average supply-current demand. From the "Power" menu, select full power-down (FULLPD) or fast power-down (FASTPD) mode. In fast power-down mode, the bandgap reference remains active. The amount of power saved depends primarily on how long the part is off between conversions. The conversion accuracy depends on the power-up delay, the reference capacitor, and the time in power-down. Adjust off-time with the "Delay Between Samples" command. Adjust on-time with the "Power-Up Delay" command.

Using an adequate power-up delay ensures conversion accuracy during power-cycling modes. The reference must be allowed enough time to stabilize before the measurement is performed. The "Power-Up Delay" command controls power-up delay. Increase the delay until accuracy is constant. The power-up delay requirement depends on the off-time (delay between samples) and the value of the reference capacitor (C12).

The MAX1202 EV kit software performs power-up by starting a conversion in FASTPD mode and discarding the reading. FASTPD mode turns on the reference, but leaves the MAX1202's other circuitry powered down. An accurate reference-voltage measurement can be performed after the power-up delay is complete.

The MAX1203 requires an external reference, so FULLPD mode can always be used, assuming the external reference is always stable when measurements are performed. In this case, set power-up delay to 0.

Measuring Supply Current

On the EV board, the MAX1202 draws all of its +5V analog power through jumper JU4, which is wired closed when the board is shipped from the factory. To measure the MAX1202's supply current, modify the board (with the power off) by cutting jumper JU4 and connecting a current meter across JU4.

Low-Speed Data Logging

The RS-232 serial link limits the data-logging sample rate to no more than 10 samples/sec. The "Log" menu can be used to write data to a user-specified file in comma-spaced-value text format. From the "Log" menu, choose "Select Channels", and select the channels you want to log. Then choose the "New Log File" command from the "Log" menu. Once a log file has been opened, it can be paused or resumed with the "Pause" command. One complete line of data is written after all enabled channels have been sampled. The first line of the log file contains the column headings. Each subsequent line of the log file contains all enabled channels, separated by commas, tabs, or spaces. The program continues to write data to the log file until the "Done" command is selected from the "Log" menu.

High-Speed Data Sampling

The high-speed sampling commands can be used for sampling rates over 10 samples/sec. Data can be collected from any single channel at high speed, using the commands on the "Sample" menu. First select the number of samples. Then set the sampling rate either by inserting a delay between samples, or by choosing one of the preset sample rates. Use the "Collect" command to begin collecting data. After the samples are collected, the data is automatically uploaded to the host and graphed. Additionally, the data can be saved to a file.

Changing the Reference Voltage

The MAX1202 EV kit software assumes a 4.096V reference voltage, unless otherwise specified. The reference-voltage assumption can be changed using "Set Reference Voltage" under the "Device" menu.

If an external reference is used, it must have a temperature coefficient of 20ppm/°C or less to achieve accuracy to within four LSBs over the 0°C to +70°C range. For 12-bit accuracy over this range, the reference must have a temperature coefficient of 4ppm/°C or less.

Detailed Description _____of Hardware

Components

The MAX1202 (U1) is an 8-channel, 12-bit, low-power analog-to-digital converter (ADC) with serial interface and shutdown. R1–R8 and C1–C8 are anti-aliasing input filters. The analog supply rails are VDD, VSS, and GND. The digital interface is powered by the VL pin. The $\overline{SHDN}$ jumper controls hardware shutdown and selects internal/external-compensation mode. Refer to the MAX1202 data sheet for more information.

The ICL7660 (U2) is a charge pump that converts +5V to -5V for V_{SS}.

The 78L05 (U3) is a +5V linear regulator that provides a clean analog supply for the MAX1202.

Input Filtering

The MAX1202 EV kit has an RC filter on each input with a time constant (τ) of approximately 3µs (R = 300Ω, C = 0.01µF). The MAX1202's acquisition time with a 2MHz clock is 1.5µs. The RC filter's settling time can increase the acquisition time required for full accuracy when switching input channels.

MAX1202 Evaluation Kit

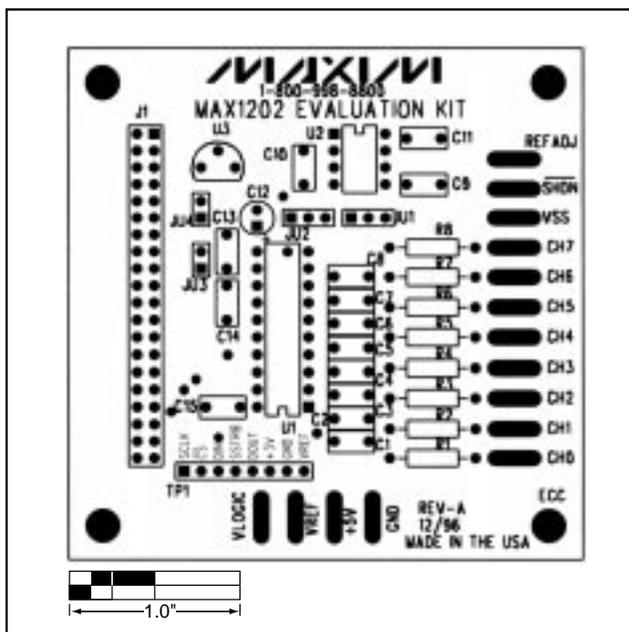Figure 1. MAX1202 EV Kit Schematic

MAXIM

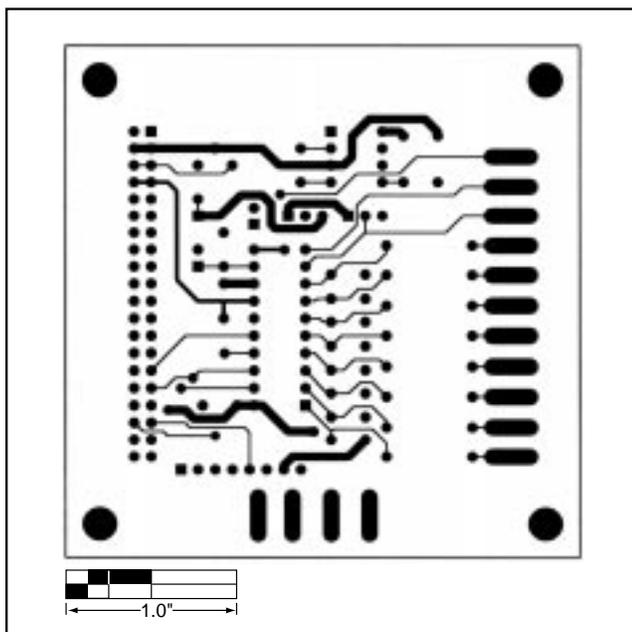Figure 2. MAX1202 EV Kit Component Placement Guide—
Top Silkscreen

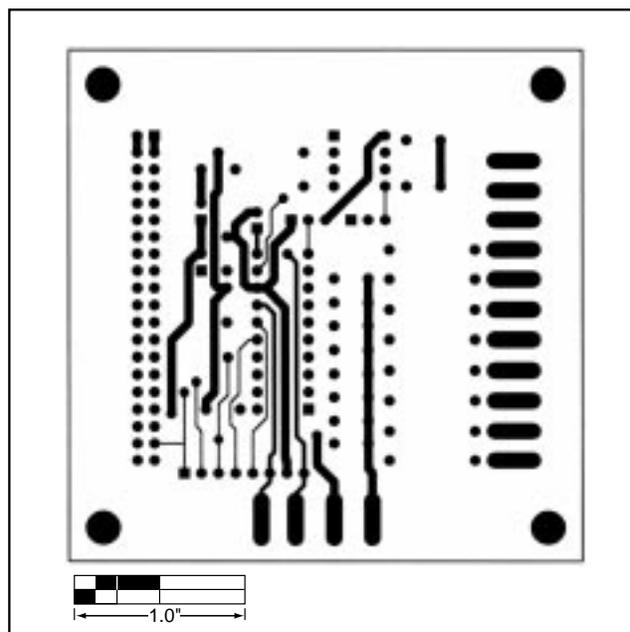Figure 3. MAX1202 EV Kit PC Board Layout—Component Side

Figure 4. MAX1202 EV Kit PC Board Layout—Solder Side

MAX1202 Evaluation Kit

```
This pseudo-code subroutine reads a value from the MAX1202 using external clock mode.
The channel argument selects the input channel (0,1,2,3,4,5,6,7).
If unipolar is TRUE, then the input range is (0...V_REF); otherwise, the input range is (-½V_REF ...+½V_REF.).
If differential is TRUE, then the input is measured with respect to an adjacent input;
        otherwise, the input is measured with respect to analog ground.
The power_down argument may select fullpd (complete power down),
        fastpd (only the bandgap reference is active), or no power down.
The cs pin is the chip select input to the MAX1202.


int read_max1202 (int channel, int unipolar, int differential, int power_down)
{
        control_byte = 0x83;                                        /* 0x83 = decimal 131 */
        if (power_down == fullpd) then control_byte = 0x80;         /* 0x80 = decimal 128 */
        else if (power_down == fastpd) then control_byte = 0x81;    /* 0x81 = decimal 129 */
        if (unipolar) then control_byte = control_byte + 8;
        if (not differential) then control_byte = control_byte + 4;

        if (channel == 1) then control_byte = control_byte + 0x40;          /* 0x40 = decimal 64 */
        else if (channel == 2) then control_byte = control_byte + 0x10;     /* 0x10 = decimal 16 */
        else if (channel == 3) then control_byte = control_byte + 0x50;     /* 0x50 = decimal 80 */
        else if (channel == 4) then control_byte = control_byte + 0x20;     /* 0x20 = decimal 32 */
        else if (channel == 5) then control_byte = control_byte + 0x60;     /* 0x60 = decimal 96 */
        else if (channel == 6) then control_byte = control_byte + 0x30;     /* 0x30 = decimal 48 */
        else if (channel == 7) then control_byte = control_byte + 0x70;     /* 0x70 = decimal 112 */

        drive cs pin low                                           /* start a conversion */
        write control_byte through SPI port; ignore the result
        /* if using internal clock mode, wait for the MAX1202 to drive the SSTRB pin high */
        write 0x00 through SPI port; store result in result_high   /* get the conversion result */
        write 0x00 through SPI port; store result in result_low
        drive cs pin high
        result = (result_high * 256) + result_low;                /* combine high and low bytes */
        result = result * 2;                         /* strip off start bit and left-justify the result */
        return result;
}
```

Listing 1. MAX1202 Pseudo-Code Example

MAXIM

This pseudo-code subroutine simultaneously reads and writes 8 bits through a synchronous serial interface. Some microprocessors offer specialized hardware that performs this job (such as Motorola's 68HC11 SPI and QSPI, National Semiconductor's Microwire, etc.) This outline describes a common substitiute interface, implemented by setting and clearing individual output pins.

Depending on the processor speed, delays may be required to satisfy the interface timing requirements.

Pin definitions:
       sck *is the serial clock input to the MAX1202*
       din *is the serial data input to the MAX1202*
       dout *is the serial data output from the MAX1202*

```
int read_write_synchronous_serial ( int write_value )
{
        read_value = 0                      /* read_value will accumulate the input value */
        drive sck pin low
        for (count = 0 to 7)                /* process each of the eight bits */
        {
                read_value = read_value * 2         /* shift input value left */
                if (write_value bit 7 is 1)         /* write the data, MSB first */
                then drive din pin high
                else drive din pin low

/* delay here to satisfy data setup time and minimum clock low time */

                drive sck pin high                  /* clock rising edge happens here */
                if (dout pin is high)               /* sample the received data, LSB last */
                        then read_value = read_value + 1

/* delay here to satisfy data hold time and minimum clock high time */

                drive sck pin low                   /* clock falling edge happens here */
                write_value = write_value * 2       /* shift output value left */
        }
        return read_value
}
```

Listing 2. Bit-Banging Substitute for SPI Port