



# MAX1110 Evaluation System/ Evaluation Kit

## General Description

The MAX1110 evaluation system (EV system) is a complete, 8-channel data-acquisition system consisting of a MAX1110 evaluation kit (EV kit) and a Maxim 3V 68L11D microcontroller ( $\mu$ C) module. It is designed to evaluate the MAX1110, a low-power, +3V, 8-channel, 8-bit analog-to-digital converter (ADC) with an internal 2.048V reference that directly interfaces to 3V microprocessors. Windows 3.1®/Windows 95® software provides a handy user interface to exercise the MAX1110's features.

The EV system is intended for comprehensive evaluation of the MAX1110 using a personal computer. Order the stand-alone EV kit if the 68L11D  $\mu$ C module has already been purchased with a previous Maxim EV system, or for custom use in other  $\mu$ C-based systems.

The MAX1110 EV kit and EV system can also be used to evaluate the MAX1112, which operates on +5V and has a 4.096V internal reference. Simply order a free sample of the MAX1112CPP along with the MAX1110 EV kit.

## MAX1110 EV System Component List

| PART             | QTY | DESCRIPTION            |
|------------------|-----|------------------------|
| MAX1110EVKIT-DIP | 1   | MAX1110 evaluation kit |
| 68L11DMODULE     | 1   | 68L11D $\mu$ C module  |

## MAX1110 EV Kit Component List

| DESIGNATION   | QTY | DESCRIPTION                             |
|---------------|-----|-----------------------------------------|
| C1–C8         | 8   | 0.01 $\mu$ F ceramic capacitors         |
| C9, C11       | 2   | 4.7 $\mu$ F, 10V tantalum capacitors    |
| C10           | 1   | 0.1 $\mu$ F ceramic capacitor           |
| J1            | 1   | 2x20 right-angle socket                 |
| J2            | 1   | 8-pin header                            |
| JU1, JU2, JU3 | 3   | 2-pin jumper blocks                     |
| JU4           | 1   | 3-pin jumper block                      |
| JU5           | 0   | Open                                    |
| R1–R8         | 8   | 10k $\Omega$ , 5% resistors             |
| R9            | 1   | 10 $\Omega$ , 5% resistor               |
| U1            | 1   | Maxim MAX1110CPP                        |
|               | 1   | 20-pin socket                           |
| None          | 1   | PC board                                |
| None          | 1   | Software disk, "MAX1110 EVALUATION KIT" |

Windows is a registered trademark of Microsoft Corp.

## Features

- ♦ Proven PC Board Layout
- ♦ Complete Evaluation System Samples to 5ksps
- ♦ Convenient Test Points Provided on Board
- ♦ Data-Logging Software
- ♦ Fully Assembled and Tested

## Ordering Information

| PART             | TEMP. RANGE  | BOARD TYPE    |
|------------------|--------------|---------------|
| MAX1110EVKIT-DIP | 0°C to +70°C | Surface Mount |
| MAX1110EVL11-DIP | 0°C to +70°C | Surface Mount |

## MAX1110 Stand-Alone EV Kit

The MAX1110 EV kit provides a proven PC board layout to facilitate evaluation of the MAX1110. It must be interfaced to appropriate timing signals for proper operation. Refer to the MAX1110 data sheet for timing requirements. Apply your supply voltage between the VDD pad and the GND pad. Refer to Table 1 for jumper configurations.

## MAX1110 EV System

The MAX1110 EV system operates from a user-supplied 9V to 15V DC power supply. The Maxim 68L11D 3V  $\mu$ C board uses a MAX667 linear regulator to generate the 3V logic supply.

### Quick Start

- 1) Install the MAX1110 EV kit software on your computer by running the INSTALL.EXE program on the floppy disk. The program files are copied and icons are created for them in the Windows 3.1 Program Manager (or the Windows 95 Start Menu).
- 2) Check the jumper settings on the MAX1110 EV kit board. Refer to Table 1, *Jumper Functions*, and Table 2, *Default Jumper Settings*.
- 3) Carefully connect the boards by aligning the 40-pin header of the MAX1110 EV kit with the 40-pin connector of the  $\mu$ C module. Gently press them together. The two boards should be flush against one another.

Evaluates: MAX1110/MAX1112

# MAX1110 Evaluation System/ Evaluation Kit

- 4) Connect a 9V to 15V DC power source to the  $\mu$ C module at the terminal block located next to the on/off switch, in the upper-right corner of the  $\mu$ C module. Observe the polarity marked on the board.
- 5) Connect a cable from the computer's serial port to the  $\mu$ C module. If using a 9-pin serial port, connect 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 MAX1110 program by opening its icon in the Program Manager (or Start Menu).
- 7) The program prompts you to connect the  $\mu$ C module and turn its power on. Slide SW1 to the ON position. Select the correct serial port, and click OK. The program automatically downloads KIT1110.L11 to the module.
- 8) Apply input signals to the CH0–CH7 inputs at the right edge of the MAX1110 EV kit board. Observe the readout on the screen.

## *Evaluating the MAX1112*

To evaluate the MAX1112, turn off power to the kit, replace U1 with a MAX1112CPP, and set  $V_{DD}$  to 5V by adjusting trim pot R2 on the 68L11D module. No other hardware changes are necessary. Start the MAX1110 software, and change the assumed reference voltage from 2.048V to 4.096V. Refer to the section *Changing the Reference Voltage*.

## Detailed Description of Software

### *Shutdown Power Cycling*

To reduce average supply-current demand, the MAX1110 can be shut down between conversions. From the Power menu, select Full Power-Down mode. The amount of power savings depends primarily on how long the part is off between conversions. The accuracy of the conversions depends on the power-up delay, the reference capacitor, and the time in power-down. Adjust the off-time with the Delay Between Samples command. Adjust the 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 the power-up delay. Increase the delay until no further change in accuracy is observed. The

power-up delay requirement depends on the value of the reference capacitor (C9) and the off-time (delay between samples).

The MAX1110 EV kit software performs power-up by starting a conversion and discarding the reading. When the power-up delay is complete, the reference voltage is correct, and an accurate measurement can be performed.

### *Measuring Supply Current*

On the EV kit board, the MAX1110 draws all its power through jumper JU5, which is wired closed when the board is shipped from the factory. To measure the MAX1110's supply current, modify the board (with the power off) by cutting jumper JU5 and connecting a current meter across JU5.

### *Low-Speed Data Logging*

The RS-232 serial link limits the data-logging sample rate to no more than 10 samples per second. The Log menu can be used to write data to a user-specified file in comma-spaced-value text format. From the Log menu, select Select Channels, and select the channels you want to log. Next, pick 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*

For sampling rates up to 5ksps, the high-speed sampling commands can be used. 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 by inserting a delay between samples, or choosing one of the preset sample rates. To begin collecting data, use the Collect command. After the samples have been collected, the data is automatically uploaded to the host and is graphed. The data can optionally be saved to a file.

### *Changing the Reference Voltage*

The MAX1110 EV kit software assumes a 2.048V reference voltage, unless otherwise specified. To change the reference voltage assumption, use Set Reference Voltage under the Device menu.

# MAX1110 Evaluation System/ Evaluation Kit

## Detailed Description of Hardware

U1, the MAX1110, is an 8-channel, 8-bit, low-power analog-to-digital converter (ADC) with serial interface and a shutdown mode. R1–R8 and C1–C8 act as anti-aliasing input filters. V<sub>DD</sub> and GND are the analog supply rails. Refer to the MAX1110 data sheet for additional information.

C9 and C11 are 4.7μF bypass capacitors for the reference and the power supply. For reduced board size

and/or lower cost, C9 and C11 may be reduced to 1μF. R9 and C10 filter the digital noise out of the analog power supply.

### Input Filtering

The EV kit has an RC filter on each input with a time constant of approximately 100μs (R = 10kΩ, C = 0.01μF). Acquisition time with a 500kHz clock is 4μs. The RC filter's settling time may increase the acquisition time required for full accuracy when switching input channels.

**Table 1. Jumper Functions**

| JUMPER | STATE  | FUNCTION                                                                     |
|--------|--------|------------------------------------------------------------------------------|
| JU1    | open   | Use an external user-supplied reference.                                     |
|        | closed | Use the internal reference (REFIN is connected to REFOUT).                   |
| JU2    | open   | Analog common voltage must be set by user.                                   |
|        | closed | Analog common (COM) = GND.                                                   |
| JU3    | open   | SHDN is allowed to float or be driven by JU4.                                |
|        | closed | SHDN is driven by the μC module.                                             |
| JU4    | open   | MAX1110 is active with internal reference enabled.                           |
|        | 1–2    | SHDN = V <sub>DD</sub> ; MAX1110 is active with internal reference disabled. |
|        | 2–3    | SHDN = GND; MAX1110 is inactive.                                             |
| JU5    | closed | Current-sense jumper. The MAX1110 draws its power through this trace.        |
|        | open   | Do not operate kit with JU5 open.                                            |

**Table 2. Default Jumper Settings**

| JUMPER | STATE  | FUNCTION                                                   |
|--------|--------|------------------------------------------------------------|
| JU1    | closed | Use the internal reference (REFIN is connected to REFOUT). |
| JU2    | closed | COM = GND.                                                 |
| JU3    | open   | SHDN is allowed to float or be driven by JU4.              |
| JU4    | open   | MAX1110 is active with internal reference enabled.         |
| JU5    | closed | The MAX1110 draws its power through this trace.            |

Evaluates: MAX1110/MAX1112

# MAX1110 Evaluation System/ Evaluation Kit

*This pseudo-code subroutine reads a value from the MAX1110 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 ( $-\frac{1}{2}V_{REF}$  ...  $+\frac{1}{2}V_{REF}$ ).*

*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** is TRUE, then the MAX1110 is powered down after the conversion completes.*

*The **cs** pin is the chip select input to the MAX1110.*

```
int read_max1110 (int channel, int unipolar, int differential, int power_down)
{
    control_byte = 0x83;                                /* 0x83 = decimal 131 */
    if (power_down) then control_byte = 0x80;            /* 0x80 = decimal 128 */
    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 MAX1110 to drive the SSTRB pin high */

    Issue two pulses on the SCK line /* (Motorola SPI must be disabled during these pulses) */

    write 0x00 through SPI port; store result            /* get the conversion result */
    drive cs pin high
    return result;
}
```

Listing 1. MAX1110 Pseudo-Code Example

# MAX1110 Evaluation System/ Evaluation Kit

Evaluates: MAX1110/MAX1112

*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 substitute 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 MAX1110*

***din** is the serial data input to the MAX1110*

***dout** is the serial data output from the MAX1110*

```
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 */

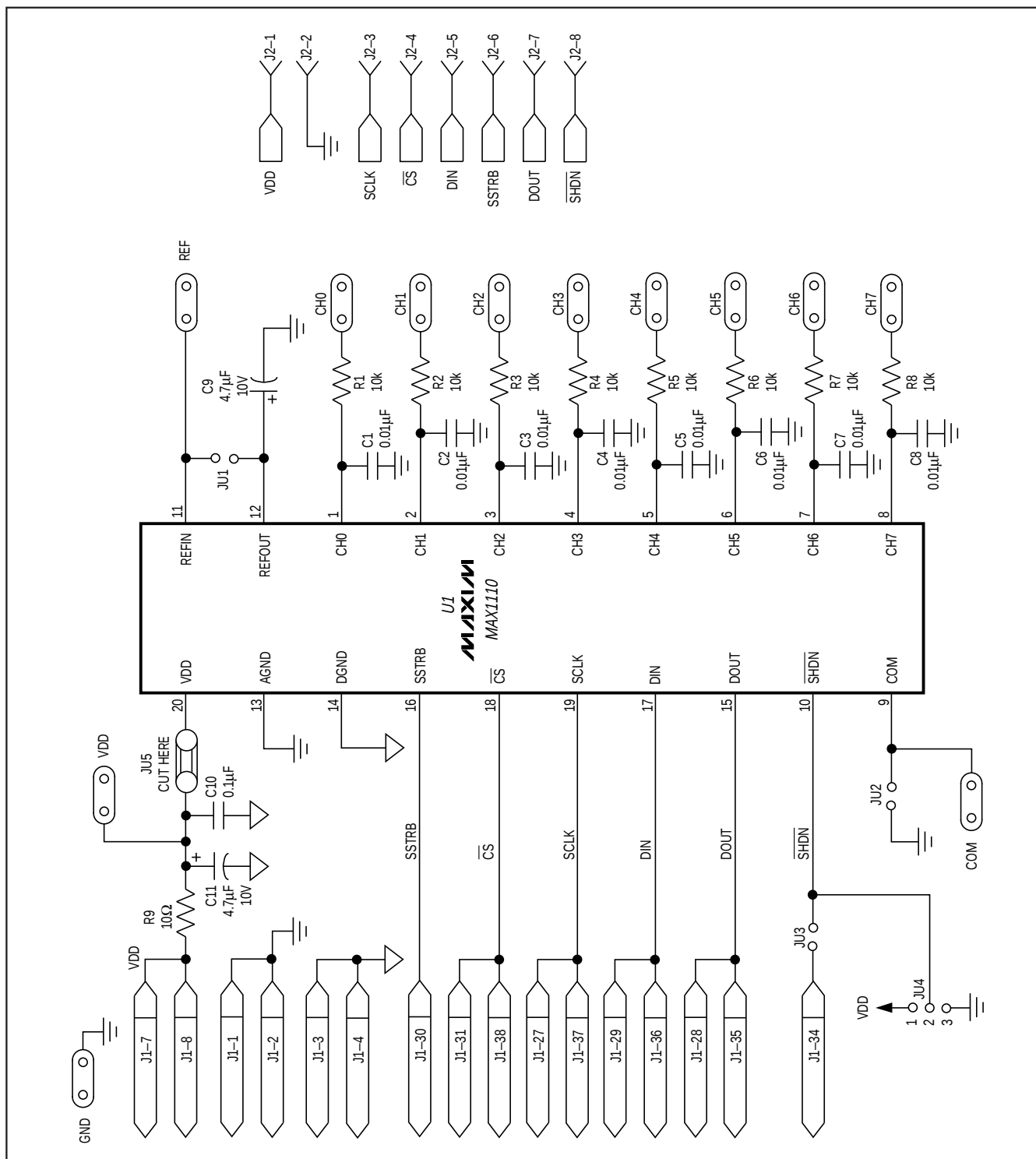
        drive sck pin high            /* clock rising edge happens here */
        if (dout pin is high)         /* sample the received data, LSB last */
            then read_value = read_value + 1

        /* 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

Evaluates: MAX1110/MAX1112



# MAX1110 Evaluation System/ Evaluation Kit

Evaluates: MAX1110/MAX1112

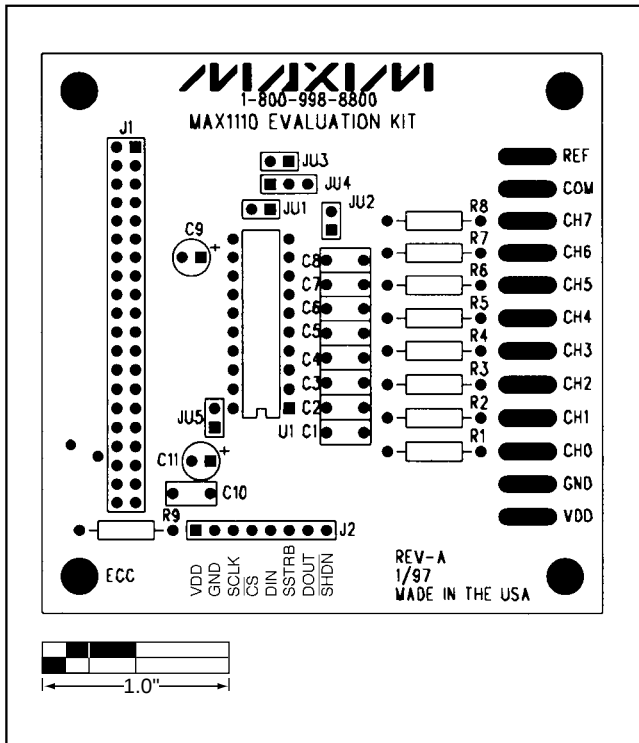


Figure 2. MAX1110 EV Kit Component Placement Guide

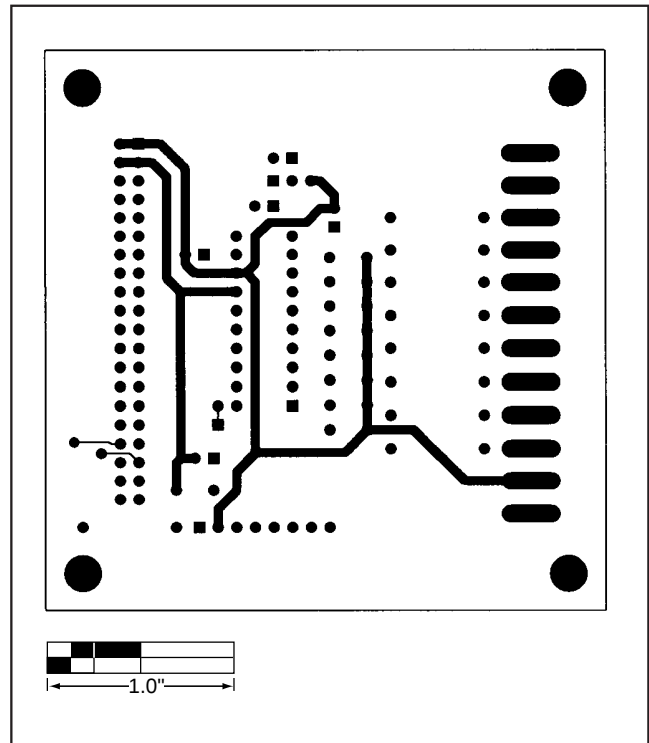


Figure 3. MAX1110 EV Kit PC Board Layout—Component Side

## MAX1110 Evaluation System/ Evaluation Kit

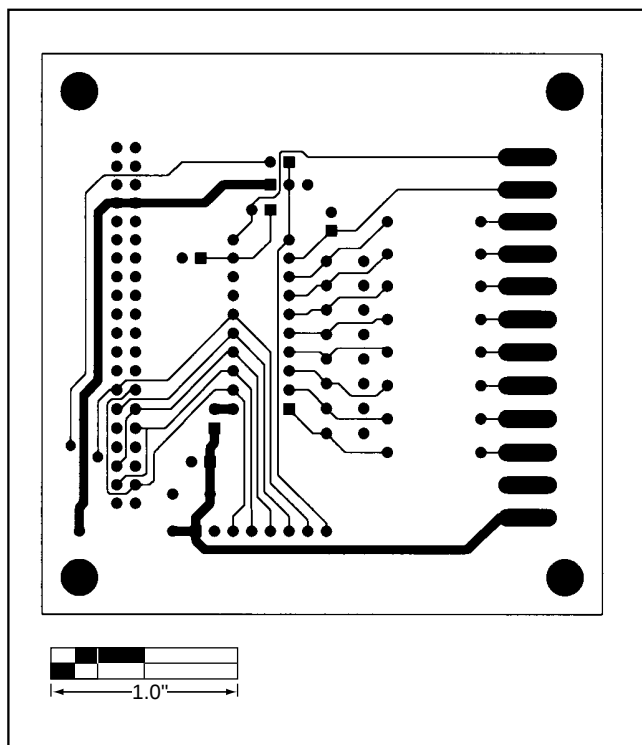


Figure 4. MAX1110 EV Kit PC Board Layout—Solder Side