RFID Read/Write Module, Serial (#28440)

Designed in cooperation with Grand Idea Studio (www.grandideastudio.com), the Parallax Radio Frequency Identification (RFID) Read/Write Module provides a low-cost solution to read and write passive RFID transponder tags up to 3 inches away. The RFID transponder tags provide a unique serial number and can store up to 116 bytes of user data, which can be password protected to allow only authorized access.

The RFID Read/Write Module can be used in a wide variety of hobbyist and commercial applications, including access control, user identification, robotics navigation, inventory tracking, payment systems, car immobilization, and manufacturing automation.

Features

- Low-cost method for reading and writing passive, 125 kHz RFID transponder tags
- Up to 116 bytes of user data storage on a single tag
- Optional security features prevent tag from being read or written without password
- Bi-color LED for visual indication of status

Key Specifications

- Power requirements: +5 VDC; < 10 mA idle, approx. 100-200 mA active
- Communication: asynchronous serial 9600 bps, (8N1); 5 V TTL-level, non-inverted
- Dimensions: 3.25 x 2.45 in (8.26 x 6.2 cm)

RFID Tag Compatibility

The RFID Read/Write Module works exclusively with the EM Microelectronics EM4x50 1kbit R/W transponder tags. A variety of different tag types and styles exist, with the most popular ones made available from Parallax.

Each tag contains a unique, read-only serial number (one of $2^{32}$, or 4,294,967,296 possible combinations) and 116 bytes of user data area stored in a non-volatile EEPROM (Electrically Erasable Read Only Memory). The user data area can be optionally secured with a 32-bit password to allow only authorized read and write operations.

The RFID Read/Write Module provides, with a single command, legacy support for the EM Microelectronics EM4100 read-only tags that are used with Parallax's RFID Card Reader Serial (#28140) and USB (#28340).
Connections

The RFID Read/Write Module easily interfaces to any host microcontroller using only four connections (VCC, SIN, SOUT, GND).

<table>
<thead>
<tr>
<th>Pin</th>
<th>Pin Name</th>
<th>Type</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>VCC</td>
<td>P</td>
<td>System power. +5V DC input.</td>
</tr>
<tr>
<td>2</td>
<td>SIN</td>
<td>I</td>
<td>Serial input from host. TTL-level interface, 9600 bps, 8 data bits, no parity, 1 stop bit.</td>
</tr>
<tr>
<td>3</td>
<td>SOUT</td>
<td>O</td>
<td>Serial output to host. TTL-level interface, 9600 bps, 8 data bits, no parity, 1 stop bit.</td>
</tr>
<tr>
<td>4</td>
<td>GROUND</td>
<td>G</td>
<td>System ground. Connect to power supply's ground (GND) terminal.</td>
</tr>
</tbody>
</table>

Note: Type: I = Input, O = Output, P = Power, G = Ground

Use the following example circuit for connecting the Parallax RFID Read/Write Module:
Usage

There are many ways to use the RFID Read/Write Module and associated tags, but the three most common are as follows:

1) Read Only: Read the tag's unique, non-changing 32-bit serial number

2) Read/Write: Read from and write up to 116 bytes of publicly accessible user data on the tag

3) Read/Write (Protected): Read from and write up to 116 bytes of password-protected user data on the tag. In this mode, you must enter the correct 32-bit password in order for read and write operations to be successful.

The RFID Read/Write Module is controlled by the host via a serial communications interface. The unit waits in an idle state until it receives a valid header and command from the host, at which time the module will perform the command and return a status/error byte indicating a success or failure and command-specific data (if any). The module will then re-enter the idle state and wait for the next valid header and command.

A visual indication of the RFID Read/Write Module's state is given with the on-board LED (Light Emitting Diode). When the module is successfully powered-up and is in an idle state, the LED will be GREEN; when the module is in an active state (for example, searching for a valid tag or performing an operation on the tag), the LED will be RED.

The face of the tag should be held parallel to the front or back face of the antenna (where the majority of RF energy is emitted). If the tag is held sideways (for example, perpendicular to the antenna), you'll either get no reading or a poor reading distance. Only one transponder tag should be held up to the antenna at any time. The use of multiple tags at one time will cause tag collisions and the reader may not detect any of them. The tags available in the Parallax store have a read distance of approximately 3 inches. Actual distance may vary slightly depending on the size of the transponder tag and environmental conditions of the application.

Communication Protocol

The RFID Read/Write Module is controlled by the host via a standard, asynchronous serial communications interface configured at 9600 bps, 8 data bits, no parity, 1 stop bit, and least significant bit first (8N1). The serial data expected and transmitted by the RFID Read/Write Module is 5V TTL-level, non-inverted.

To communicate with the RFID Read/Write Module, the user must first send a three-byte header string of !RW (in ASCII), followed by the desired single-byte command (in hexadecimal). Certain commands require additional data to be sent along with the command. A status/error byte and a varying number of data bytes will be returned depending on the command called.
Command Set

0x01: RFID_Read
Read data from a specified address
Input: Address location (1 byte), valid locations are 1-33 decimal
Output: Status byte (1 byte) + data (4 bytes)
Example:

SEROUT RFID_TX, Baud, ["!RW", RFID_Read, ADDR_Serial]  ' Read tag's serial number
SERIN RFID_RX, Baud, [err, STR data\4]  ' Get status byte & data bytes

0x02: RFID_Write
Write data to a specified address
Input: Address location (1 byte), valid locations are 3-31 decimal + data (4 bytes)
Output: Status byte (1 byte)
Example:

' Write $FEEDBEEF to address 3 (User Data Area)
SEROUT RFID_TX, Baud, ["!RW", RFID_Write, 3, $FE, $ED, $BE, $EF]
SERIN RFID_RX, Baud, [err]  ' Get status byte

Note: After writing the specified data to the tag, the RFID Read/Write Module automatically performs a read function to verify that the data has properly been written.

0x03: RFID_Login
Login to the tag (required to use password protection features)
Input: Password (4 bytes), default = 0x00000000
Output: Status byte (1 byte)
Example:

' Login to tag with password
SEROUT RFID_TX, Baud, ["!RW", RFID_Login, $00, $00, $00, $00]
SERIN RFID_RX, Baud, [err]  ' Get status byte

0x04: RFID_SetPass
Change the tag’s password
Input: Current password (4 bytes) + new password (4 bytes)
Output: Status byte (1 byte)
Example:

' Change password from current ($00000000) to new ($FEEDBEEF)
SEROUT RFID_TX, Baud, ["!RW", RFID_SetPass, REP $00\4, $FE, $ED, $BE, $EF]
SERIN RFID_RX, Baud, [err]  ' Get status byte
**0x05: RFID_Protect**

Enable (lock) or disable (unlock) the tag's password protection.

Input: Mode (1 byte), 0x00 to unlock, 0x01 to lock

Output: Status byte (1 byte)

Example:
```
SEROUT RFID_TX, Baud, ["!RW", RFID_Protect, 1] ' Lock the tag
SERIN RFID_RX, Baud, [err]        ' Get status byte
```

Note: When the tag is locked, a successful login using the RFID_Login command is required before you can enable/disable password protection, change the tag's password, or read/write to the tag. If the tag is locked and you are not successfully logged in, the RFID_Read command will return four bytes of 0x00 in place of the actual user data and the RFID_Write command will return an error.

**0x06: RFID_Reset**

Log out and reset the tag.

Input: None

Output: Status byte (1 byte)

Example:
```
SEROUT RFID_TX, Baud, ["!RW", RFID_Reset] ' Reset the tag
SERIN RFID_RX, Baud, [err]        ' Get status byte
```

Note: All user data will remain stored on the tag. You will need to re-login using RFID_Login if you want to access a locked tag after calling this command. You can also reset the tag by removing it from the proximity of the RFID Read/Write Module.

**0x0F: RFID_ReadLegacy**

Read the 40-bit unique serial number from an EM Microelectronics EM4100 read-only tag (used with Parallax's RFID Card Reader Serial, #28140, and USB, #28340).

Input: None

Output: Header (1 byte) + data (10 bytes) + footer (1 byte)

Example:
```
SEROUT RFID_TX, Baud, ["!RW", RFID_ReadLegacy] ' Read EM4100 unique serial #
SERIN RFID_RX, Baud, [header, STR data\10, footer] ' Get header, data, and footer
```

Note: The header and footer are used to identify that a correct string has been received from the reader (they correspond to a line feed and carriage return characters, respectively). The middle ten bytes are the tag's unique ID sent in a printable ASCII string. For example, for a tag with a valid ID of 0F018F07A, the following bytes would be sent: 0x0A, 0x30, 0x46, 0x30, 0x31, 0x38, 0x34, 0x46, 0x30, 0x37, 0x41, 0x0D.
Error Checking

The RFID Read/Write Module responds to each command with a single-byte status/error code in hexadecimal format followed by any returned data, if applicable. In most standard implementations, the user will repeatedly call the desired function until the ERR_OK status byte is returned, indicating that the function successfully completed with no errors.

For scenarios where more error checking is desired, the RFID Read/Write Module provides detailed error code values:

<table>
<thead>
<tr>
<th>Value</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x01</td>
<td>ERR_OK</td>
<td>No Errors</td>
</tr>
<tr>
<td>0x02</td>
<td>ERR_LIW</td>
<td>Could not find a Listen Window (LIW) from the tag</td>
</tr>
<tr>
<td>0x03</td>
<td>ERR_NAK</td>
<td>Received a No Acknowledge (NAK), possible communication error or invalid command/data</td>
</tr>
<tr>
<td>0x04</td>
<td>ERR_NAK_OLPW</td>
<td>Received a No Acknowledge (NAK) sending the current password during the RFID SetPass command, possible incorrect password</td>
</tr>
<tr>
<td>0x05</td>
<td>ERR_NAK_NEWPW</td>
<td>Received a No Acknowledge (NAK) sending the new password during the RFID SetPass command</td>
</tr>
<tr>
<td>0x06</td>
<td>ERR_LIW_NEWPW</td>
<td>Could not find a Listen Window (LIW) from the tag after setting the new password during the RFID SetPass command</td>
</tr>
<tr>
<td>0x07</td>
<td>ERR_PARITY</td>
<td>Parity error when reading data from the tag</td>
</tr>
</tbody>
</table>

Tag Details

The RFID Read/Write Module works exclusively with the EM Microelectronics-Marin SA EM4x50 1kbit R/W transponder tags at 125kHz. A variety of different tag types and styles exist with the most popular ones made available from Parallax.

The memory map and corresponding address locations for the EM4x50 tag are shown below. Each address holds a 32-bit (4 byte) value.

<table>
<thead>
<tr>
<th>Address (decimal)</th>
<th>Register</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>ADDR_Password</td>
<td>Password (not readable)</td>
</tr>
<tr>
<td>1</td>
<td>ADDR_Protect</td>
<td>Protection word</td>
</tr>
<tr>
<td>2</td>
<td>ADDR_Control</td>
<td>Control word</td>
</tr>
<tr>
<td>3-31</td>
<td>User Data Area</td>
<td>Non-volatile EEPROM storage area for user data (116 bytes)</td>
</tr>
<tr>
<td>32</td>
<td>ADDR_Serial</td>
<td>Device serial number (unique, read-only)</td>
</tr>
<tr>
<td>33</td>
<td>ADDR_DeviceID</td>
<td>Device identification (used to identify tag family, read-only)</td>
</tr>
</tbody>
</table>

The User Data Area can be optionally secured using a 32-bit password to allow only authorized read and write operations. This password is stored in ADDR_Password and can be changed by the user via the RFID_SetPass command with knowledge of the current password. For security reasons, it cannot be read. As such, if your tag is password-protected and you forget the tag’s password, you will be unable to retrieve the contents of the User Data Area.

ADDR_Protect and ADDR_Control are modified directly by the RFID Read/Write Module and, in normal scenarios, should not be changed by the user.
ADDR_Serial is a unique, read-only serial number with one of $2^{32}$, or 4,294,967,296, possible values. It cannot be changed by the user.

ADDR_DeviceID is a read-only value used to identify the specific tag family (for example, EM4150). The 32-bit value contains a family code, version code, reserved/unused bits, and a checksum value. It cannot be changed by the user.

Further details of the EM4x50 tag can be found at: www.emmicroelectronic.com/Products.asp?idProduct=159

**Electrical Characteristics**

At $V_{CC} = +5.0V$ and $T_A = 25^\circ C$ unless otherwise noted

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Test Conditions</th>
<th>Specification</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Min.</td>
<td>Typ.</td>
</tr>
<tr>
<td>Supply Voltage</td>
<td>$V_{CC}$</td>
<td>---</td>
<td>4.5</td>
<td>5.0</td>
</tr>
<tr>
<td>Supply Current, Idle</td>
<td>$I_{IDLE}$</td>
<td>---</td>
<td>---</td>
<td>9.4</td>
</tr>
<tr>
<td>Supply Current, Active</td>
<td>$I_{CC}$</td>
<td>---</td>
<td>108</td>
<td>200</td>
</tr>
<tr>
<td>Input LOW voltage</td>
<td>$V_{IL}$</td>
<td>$+4.5V \leq V_{CC} +5.5V$</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>Input HIGH voltage</td>
<td>$V_{IH}$</td>
<td>$+4.5V \leq V_{CC} +5.5V$</td>
<td>2.0</td>
<td>---</td>
</tr>
<tr>
<td>Output LOW voltage</td>
<td>$V_{OL}$</td>
<td>$V_{CC} = +4.5V$</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>Output HIGH voltage</td>
<td>$V_{OH}$</td>
<td>$V_{CC} = +4.5V$</td>
<td>$V_{CC} - 0.7$</td>
<td>---</td>
</tr>
</tbody>
</table>

**Absolute Maximum Ratings**

<table>
<thead>
<tr>
<th>Condition</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating Temperature</td>
<td>-40(^\circ)C to +85(^\circ)C</td>
</tr>
<tr>
<td>Storage Temperature</td>
<td>-55(^\circ)C to +125(^\circ)C</td>
</tr>
<tr>
<td>Supply Voltage ($V_{CC}$)</td>
<td>+4.5V to +5.5V</td>
</tr>
<tr>
<td>Ground Voltage ($V_{SS}$)</td>
<td>0V</td>
</tr>
<tr>
<td>Voltage on any pin with respect to $V_{SS}$</td>
<td>-0.3V to +7.0V</td>
</tr>
</tbody>
</table>

**NOTICE:** Stresses above those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.
RFID Technology Overview

Material in this section is based on information provided by the RFID Journal (www.rfidjournal.com).

Radio Frequency Identification (RFID) is a generic term for non-contacting technologies that use radio waves to automatically identify people or objects. There are several methods of identification, but the most common is to store a unique serial number that identifies a person or object on a microchip that is attached to an antenna. The combined antenna and microchip are called an "RFID transponder" or "RFID tag" and work in combination with an "RFID reader" (sometimes called an "RFID interrogator").

An RFID system consists of a reader and one or more tags. The reader's antenna is used to transmit radio frequency (RF) energy. Depending on the tag type, the energy is "harvested" by the tag's antenna and used to power up the internal circuitry of the tag. The tag will then modulate the electromagnetic waves generated by the reader in order to transmit its data back to the reader. The reader receives the modulated waves and converts them into digital data.

There are two major types of tag technologies. "Passive tags" are tags that do not contain their own power source or transmitter. When radio waves from the reader reach the chip's antenna, the energy is converted by the antenna into electricity that can power up the microchip in the tag (typically via inductive coupling). The tag is then able to send back any information stored on the tag by modulating the reader's electromagnetic waves. "Active tags" have their own power source and transmitter. The power source, usually a battery, is used to run the microchip's circuitry and to broadcast a signal to a reader. Due to the fact that passive tags do not have their own transmitter and must reflect their signal to the reader, the reading distance is much shorter than with active tags. However, active tags are typically larger, more expensive, and require occasional service.

Frequency refers to the size of the radio waves used to communicate between the RFID system components. Just as you tune your radio to different frequencies in order to hear different radio stations, RFID tags and readers have to be tuned to the same frequency in order to communicate effectively. RFID systems typically use one of the following frequency ranges: low frequency (or LF, around 125 and 134.2 kHz), high frequency (or HF, around 13.56 MHz), ultra-high frequency (or UHF, around 868 and 928 MHz), or microwave (around 2.45 and 5.8 GHz).

The read range of a tag ultimately depends on many factors: the frequency of RFID system operation, the power of the reader, and interference from other RF devices. Balancing a number of engineering trade-offs (antenna size vs. reading distance vs. power vs. manufacturing cost), the Parallax RFID Read/Write Module's antenna was designed specifically for use with low-frequency (125 kHz) passive tags.

BASIC Stamp 2 Program

The example program shown below is a simple system-level test to read the tag ID and write a block of data. This .bs2 program, along with a more comprehensive code example demonstrating the full capabilities of the RFID Read/Write Module, is available for download on the product page; search "28440" at www.parallax.com.
File...... rfid_rw_test.bs2
Purpose... System-level test code for the Parallax RFID Read/Write Module
Author.... Joe Grand, Grand Idea Studio, Inc. [www.grandideastudio.com]
E-mail.... support@parallax.com
Updated... 15 Dec 2009

{$STAMP BS2}
{$PBASIC 2.5}

-----[ Program Description ]--------------------------------------------------------

This program performs a system-level test of the Parallax RFID Read/
Write Module by:

1) Reading tag's unique ID
2) Writing and verifying a block of data to the tag

-----[ I/O Definitions ]------------------------------------------------------------

RFID_TX      PIN   0              ' Connects to RFID R/W Module SIN
RFID_RX      PIN   1              ' Connects to RFID R/W Module SOUT

-----[ Constants ]------------------------------------------------------------------

#SELECT $STAMP
   #CASE BS2, BS2E, BS2PE
      T9600    CON     84
   #CASE BS2SX, BS2P
      T9600    CON     240
   #ENDSELECT
Baud              CON     T9600

RFID R/W Module Commands
' Number of bytes returned in ()
RFID_Read  CON $01   ' Read data from specified address,
   ' valid locations 1 to 33 (5)
RFID_Write CON $02   ' Write data to specified address,
   ' valid locations 3 to 31 (1)
RFID_Login  CON $03  ' Login to tag with password (1)
RFID_SetPass CON $04  ' Change tag's password from old to new (1)
RFID_Protect CON $05  ' Enable/disable password protection (1)
RFID_Reset  CON $06  ' Reset tag (1)
RFID_ReadLegacy CON $0F  ' Read unique ID from EM4102 read-only tag
   ' (for backwards compatibility with Parallax
      ' RFID Card Reader, #28140 and #28340) (12)

Memory map/address locations for EM4x50 tag
' Each address holds/returns a 32-bit (4 byte) value
ADDR_Password  CON 0    ' Password (not readable)
ADDR_Protect   CON 1    ' Protection Word
ADDR_Control   CON 2    ' Control Word
' ADDR 3-31 are User EEPROM area
ADDR_Serial    CON 32   ' Device Serial Number
ADDR_DeviceID  CON 33   ' Device Identification

Copyright © Parallax Inc. RFID Read/Write Module, Serial (#28440) v1.0 6/29/2010 Page 9 of 10
' Status/error return codes
ERR_OK     CON $01 ' No errors
ERR_LIW    CON $02 ' Did not find a listen window
ERR_NAK    CON $03 ' Received a NAK, could be invalid command
ERR_NAK_OLPW CON $04 ' Received a NAK sending old password (RFID_SetPass),
                      ' could be incorrect password
ERR_NAK_NEWPW CON $05 ' Received a NAK sending new password (RFID_SetPass)
ERR_LIW_NEWPW CON $06 ' Did not find a listen window after sending
                      ' old password (RFID_SetPass)
ERR_PARITY CON $07 ' Parity error when reading data

' ------[ Variables ]---------------------------------------------------------------
buf       VAR Byte(12) ' data buffer
idx       VAR Byte    ' index
idy       VAR Byte

' ------[ EEPROM Data ]----------------------------------------------------------------

' ------[ Initialization ]-------------------------------------------------------------

Initialize:
    PAUSE 250 ' let DEBUG open
    DEBUG CLS ' clear the screen
    DEBUG "Parallax RFID Read/Write Module Test Application", CR,
            "------------------------------------------------", CR, CR

' ------[ Program Code ]---------------------------------------------------------------

Main:
    DEBUG "Reading tag's unique serial number..."
Read_Tag:
    SEROUT RFID_TX, Baud, ["!RW", RFID_Read, ADDR_Serial] ' Read tag's serial number
    SERIN  RFID_RX, Baud, [STR buf\5]               ' Get status byte and data bytes
    IF buf(0) <> ERR_OK THEN Read_Tag                ' If we get an error, keep trying
             ' until the read is successful
             ' Print data
    FOR idx = 1 TO 4                                 ' Print data
        DEBUG HEX2 buf(idx)
    NEXT
    DEBUG CR
    DEBUG "Writing and verifying data to tag..."
Write_Tag:
    SEROUT RFID_TX, Baud, ["!RW", RFID_Write, 3, $FE, $ED, $BE, $EF] ' Write $FEEDBEEF
                                                     ' into address 4 (user EEPROM area)
    SERIN  RFID_RX, Baud, [buf(0)]               ' Wait for status byte
    IF buf(0) <> ERR_OK THEN Write_Tag            ' If we get an error, keep trying
             ' until the write is successful
    DEBUG "Success!", CR
    DEBUG "End of test.", CR
END

' ------[ End of File ]---------------------------------------------------------------