



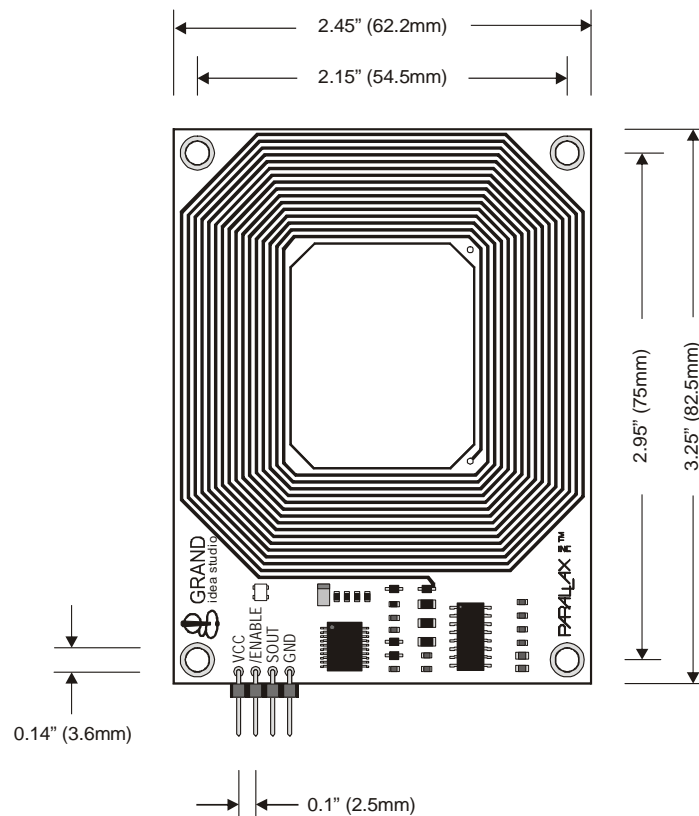
599 Menlo Drive, Suite 100
Rocklin, California 95765, USA
Office: (916) 624-8333
Fax: (916) 624-8003

General: info@parallax.com
Technical: support@parallax.com
Web Site: www.parallax.com
Educational: www.stampsinclass.com

RFID Reader Module (#28140)

RFID 54 mm x 85 mm Rectangle Tag (#28141)

RFID 50 mm Round Tag (#28142)



Introduction

Designed in cooperation with Grand Idea Studio (<http://www.grandideastudio.com/>), the Parallax Radio Frequency Identification (RFID) Reader Module is the first low-cost solution to read passive RFID transponder tags up to 1 3/4" - 3" inches away depending on the tag (see list below). The RFID Reader Module can be used in a wide variety of hobbyist and commercial applications, including access control, automatic identification, robotics navigation, inventory tracking, payment systems, and car immobilization.

- Fully-integrated, low-cost method of reading passive RFID transponder tags
- 1-wire, 2400 baud Serial TTL interface to PC, BASIC Stamp® and other processors
- Requires single +5VDC supply
- Bi-color LED for visual indication of activity
- 0.100" pin spacing for easy prototyping and integration

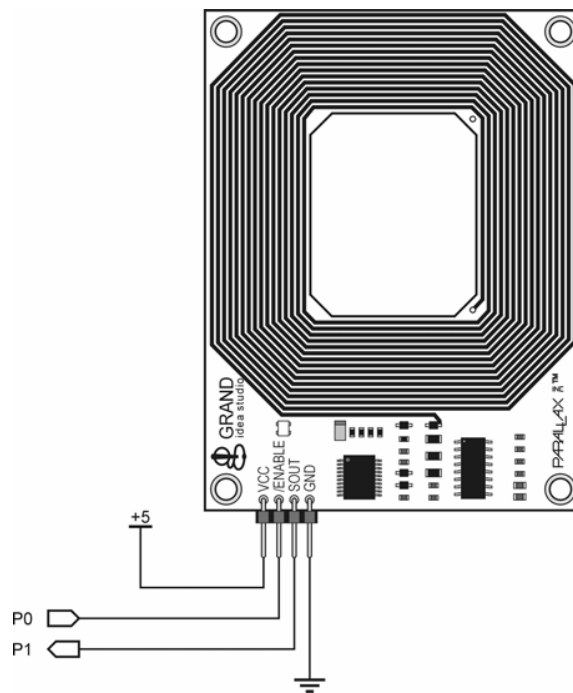
The Parallax RFID Reader Module works exclusively with the EM Microelectronics-Marin SA EM4100-family of passive read-only transponder tags. A variety of different tag types and styles exist with the most popular made available from Parallax. Each transponder tag contains a unique identifier (one of 2^{40} , or 1,099,511,627,776, possible combinations) that is read by the RFID Reader Module and transmitted to the host via a simple serial interface.

Electronic Connections

The Parallax RFID Reader Module can be integrated into any design using only four connections (VCC, /ENABLE, SOUT, GND). Use the following circuit for connecting the Parallax RFID Reader Module to the BASIC Stamp microcontroller:

Pin	Pin Name	Type	Function
1	VCC	P	System power, +5V DC input.
2	/ENABLE	I	Module enable pin. Active LOW digital input. Bring this pin LOW to enable the RFID reader and activate the antenna.
3	SOUT	O	Serial Out. TTL-level interface, 2400bps, 8 data bits, no parity, 1 stop bit.
4	GND	G	System ground. Connect to power supply's ground (GND) terminal.

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



Communication Protocol

Implementation and usage of the RFID Reader Module is straightforward. BASIC Stamp 1, 2, and SX28AC/DP code examples (SX/B) are included at the end of this documentation.

The RFID Reader Module is controlled with a single TTL-level active-low /ENABLE pin. When the /ENABLE pin is pulled LOW, the module will enter its active state and enable the antenna to interrogate for tags. The current consumption of the module will increase dramatically when the module is active.

A visual indication of the state of the RFID Reader Module is given with the on-board LED. 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 and the antenna is transmitting, the LED will be RED.

The face of the RFID tag should be held parallel to the front or back face of the antenna (where the majority of RF energy is focused). If the tag is held sideways (perpendicular to the antenna) you'll either get no reading or a poor reading. 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 confuse the reader. The two 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.

When a valid RFID transponder tag is placed within range of the activated reader, the unique ID will be transmitted as a 12-byte ASCII string via the TTL-level SOUT (Serial Output) pin in the following format:

MSB										LSB	
Start Byte (0x0A)	Unique ID Digit 1	Unique ID Digit 2	Unique ID Digit 3	Unique ID Digit 4	Unique ID Digit 5	Unique ID Digit 6	Unique ID Digit 7	Unique ID Digit 8	Unique ID Digit 9	Unique ID Digit 10	Stop Byte (0x0D)

The start byte and stop byte are used to easily 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 actual tag's unique ID.

All communication is 8 data bits, no parity, 1 stop bit, non-inverted, least significant bit first (8N1). The baud rate is configured for 2400bps, a standard communications speed supported by most any microprocessor or PC, and cannot be changed. The Parallax RFID Reader Module initiates all communication. The Parallax RFID Reader Module can connect directly to any TTL-compatible UART or to an RS232-compatible interface by using an external level shifter.

Absolute Maximum Ratings and Electrical Characteristics

Condition	Value
Operating Temperature	-40°C to +85°C
Storage Temperature	-55°C to +125°C
Supply Voltage (V _{CC})	+4.5V to +5.5V
Ground Voltage (V _{SS})	0V
Voltage on any pin with respect to V _{SS}	-0.3V to +7.0V

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.

DC Characteristics

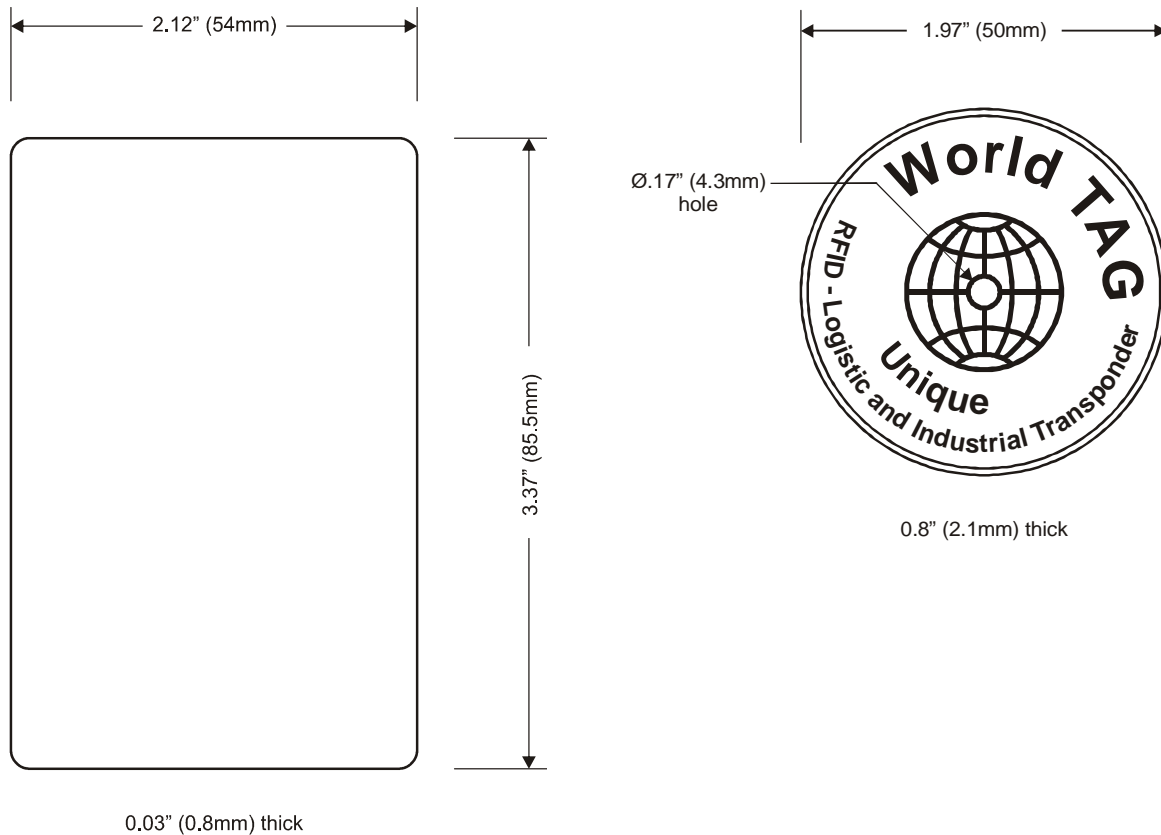
At $V_{CC} = +5.0V$ and $T_A = 25^\circ C$ unless otherwise noted

Parameter	Symbol	Test Conditions	Specification			Unit
			Min.	Typ.	Max.	
Supply Voltage	V_{CC}	---	4.5	5.0	5.5	V
Supply Current, Idle	I_{IDLE}	---	---	10	---	mA
Supply Current, Active	I_{CC}	---	---	90	---	mA
Input LOW voltage	V_{IL}	$+4.5V \leq V_{CC} \leq +5.5V$	---	---	0.8	V
Input HIGH voltage	V_{IH}	$+4.5V \leq V_{CC} \leq +5.5V$	2.0	---	---	V
Output LOW voltage	V_{OL}	$V_{CC} = +4.5V$	---	---	0.6	V
Output HIGH voltage	V_{OH}	$V_{CC} = +4.5V$	$V_{CC} - 0.7$	---	---	V

RFID Tags Available From Parallax

Parallax provides two passive RFID tags from our on-line store. We're stocking the tags because many suppliers have high minimums, yet many of our customers may only want a few tags for their basic experimentation.

- 54 mm x 85 mm Rectangle Tag (#28141)
- 50 mm Round Tag (#28142)



Actual tag dimensions may vary. Contact Parallax for specific information.

Optional Tag Information

Even though Parallax only carries a Round Tag and a Rectangle Tag the following values were obtained from different tags available in the market.

ISO Card:	6.3cm (2.5") +/- 10%
World Tag 50mm:	6.8cm (2.7") +/- 10%
World Tag 30mm:	5.3cm (2.1") +/- 10%
Bobsleigh Keyfob:	5.3cm (2.1") +/- 10%
Tear shape:	4.0cm (1.6") +/- 10%
Wristband:	4.0cm (1.6") +/- 10%

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. In the case of the Parallax RFID Reader Module, correctly received digital data is sent serially through the SOUT pin.

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 (known as "parasitic power"). The tag is then able to send back any information stored on the tag by reflecting the electromagnetic waves as described above. "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. The RFID Reader Module is designed specifically for low-frequency (170 kHz) passive tags.

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 170 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). It is generally safe to assume that a higher frequency equates to a faster data transfer rate and longer read ranges, but also more sensitivity to environmental factors such as liquid and metal that can interfere with radio waves.

There really is no such thing as a "typical" RFID tag. 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 v. reading distance v. power v.

manufacturing cost), the Parallax RFID Reader Module's antenna was designed with a specific inductance and "Q" factor for 170kHz RFID operation at a tag read distance of up to 1 3/4" - 3" inches.

Example Code

The following code examples read tags from a RFID Reader Module and compare the values to known tags (stored in an EEPROM table).

```
' =====
'
' File..... RFID.BS1
' Purpose.... RFID Tag Reader / Simple Security System
' Author..... (c) Parallax, Inc. -- All Rights Reserved
' E-mail..... support@parallax.com
' Started....
' Updated.... 07 FEB 2005
'
'   {$STAMP BS1}
'   {$PBASIC 1.0}
'
' =====

' ----- [ Program Description ]-----
'
' Reads tags from a Parallax RFID reader and compares to known tags (stored
' in EEPROM table). If tag is found, the program will disable a lock.

' ----- [ Revision History ]-----

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

SYMBOL  Enable          = 0           ' low = reader on
SYMBOL  RX              = 1           ' serial from reader
SYMBOL  Spkr            = 2           ' speaker output
SYMBOL  Latch           = 3           ' lock/latch control

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

SYMBOL  LastTag         = 2           ' 3 tags; 0 to 2

' ----- [ Variables ]-----

SYMBOL  tag0            = B0           ' RFID bytes buffer
SYMBOL  tag1            = B1
SYMBOL  tag2            = B2
SYMBOL  tag3            = B3
SYMBOL  tag4            = B4
SYMBOL  tag5            = B5
SYMBOL  tag6            = B6
SYMBOL  tag7            = B7
SYMBOL  tag8            = B8
SYMBOL  tag9            = B9

SYMBOL  tagNum          = B10          ' from EEPROM table
SYMBOL  pntr            = B11          ' pointer to char in table
SYMBOL  char            = B12          ' character from table
```

```
' -----[ EEPROM Data ]-----  
  
Tags:  
  EEPROM ("0F0184F20B")           ' valid tags  
  EEPROM ("0F01D9D263")  
  EEPROM ("04129C1B43")  
  EEPROM ("0000000000")           ' space for other tags  
  EEPROM ("0000000000")  
  
' -----[ Initialization ]-----  
  
Reset:  
  HIGH Enable                       ' turn of RFID reader  
  LOW Latch                          ' lock the door!  
  
' -----[ Program Code ]-----  
  
Main:  
  LOW Enable                         ' activate the reader  
  SERIN RX, T2400, ($0A)             ' wait for header  
  SERIN RX, T2400, tag0, tag1, tag2, tag3, tag4 ' get tag bytes  
  SERIN RX, T2400, tag5, tag6, tag7, tag8, tag9  
  HIGH Enable                         ' deactivate reader  
  
Check_List:  
  FOR tagNum = 0 TO LastTag           ' scan through known tags  
    pntr = tagNum * 10 + 0 : READ pntr, char ' read char from DB  
    IF char <> tag0 THEN Bad_Char         ' compare with tag data  
    pntr = tagNum * 10 + 1 : READ pntr, char  
    IF char <> tag1 THEN Bad_Char  
    pntr = tagNum * 10 + 2 : READ pntr, char  
    IF char <> tag2 THEN Bad_Char  
    pntr = tagNum * 10 + 3 : READ pntr, char  
    IF char <> tag3 THEN Bad_Char  
    pntr = tagNum * 10 + 4 : READ pntr, char  
    IF char <> tag4 THEN Bad_Char  
    pntr = tagNum * 10 + 5 : READ pntr, char  
    IF char <> tag5 THEN Bad_Char  
    pntr = tagNum * 10 + 6 : READ pntr, char  
    IF char <> tag6 THEN Bad_Char  
    pntr = tagNum * 10 + 7 : READ pntr, char  
    IF char <> tag7 THEN Bad_Char  
    pntr = tagNum * 10 + 8 : READ pntr, char  
    IF char <> tag8 THEN Bad_Char  
    pntr = tagNum * 10 + 9 : READ pntr, char  
    IF char <> tag9 THEN Bad_Char  
    GOTO Tag_Found                     ' all match -- good tag  
  
Bad_Char:  
  NEXT  
  
Bad_Tag:  
  SOUND Spkr, (25, 80)                 ' groan  
  PAUSE 1000  
  GOTO Main  
  
Tag_Found:  
  DEBUG #tagNum, CR                     ' for testing  
  HIGH Latch                            ' remove latch  
  SOUND Spkr, (114, 165)                ' beep  
  LOW Latch                             ' restore latch
```

```
GOTO Main

END

'
=====
'
File..... RFID.BS2
Purpose.... RFID Tag Reader / Simple Security System
Author..... (c) Parallax, Inc. -- All Rights Reserved
E-mail..... support@parallax.com
Started....
Updated.... 07 FEB 2005

'
  {$STAMP BS2}
  {$PBASIC 2.5}
'
=====

'
----- [ Program Description ]-----
'
Reads tags from a Parallax RFID reader and compares to known tags (stored
in EEPROM table).  If tag is found, the program will disable a lock.

'
----- [ Revision History ]-----

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

Enable      PIN      0          ' low = reader on
RX           PIN      1          ' serial from reader
Spkr        PIN      2          ' speaker output
Latch       PIN      3          ' lock/latch control

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

#SELECT $STAMP
#CASE BS2, BS2E, BS2PE
  T1200      CON      813
  T2400      CON      396
  T4800      CON      188
  T9600      CON      84
  T19K2      CON      32
  TMidi      CON      12
  T38K4      CON      6
#CASE BS2SX, BS2P
  T1200      CON      2063
  T2400      CON      1021
  T4800      CON      500
  T9600      CON      240
  T19K2      CON      110
  TMidi      CON      60
  T38K4      CON      45
#CASE BS2PX
  T1200      CON      3313
  T2400      CON      1646
  T4800      CON      813
  T9600      CON      396
  T19K2      CON      188
  TMidi      CON      108
  T38K4      CON      84

#ENDSELECT
```



```

SevenBit      CON      $2000
Inverted      CON      $4000
Open          CON      $8000
Baud          CON      T2400

#SELECT $STAMP
#CASE BS2, BS2E
    TmAdj      CON      $100          ' x 1.0 (time adjust)
    FrAdj      CON      $100          ' x 1.0 (freq adjust)
#CASE BS2SX
    TmAdj      CON      $280          ' x 2.5
    FrAdj      CON      $066          ' x 0.4
#CASE BS2P
    TmAdj      CON      $3C5          ' x 3.77
    FrAdj      CON      $044          ' x 0.265
#CASE BS2PE
    TmAdj      CON      $100          ' x 1.0
    FrAdj      CON      $0AA          ' x 0.665
#CASE BS2Px
    TmAdj      CON      $607          ' x 6.03
    FrAdj      CON      $2A           ' x 0.166
#ENDSELECT

```

```

LastTag      CON      3

#DEFINE __No_SPRAM = ($STAMP < BS2P)          ' does module have SPRAM?

' ----- [ Variables ] -----
#IF __No_SPRAM #THEN
    buf        VAR      Byte(10)          ' RFID bytes buffer
#ELSE
    chkChar    VAR      Byte              ' character to test
#ENDIF

tagNum       VAR      Nib                 ' from EEPROM table
idx          VAR      Byte                ' tag byte index
char         VAR      Byte                ' character from table

' ----- [ EEPROM Data ] -----
Tag1         DATA     "0F0184F20B"        ' valid tags
Tag2         DATA     "0F01D9D263"
Tag3         DATA     "04129C1B43"

Name0        DATA     "Unauthorized", CR, 0
Name1        DATA     "George Johnston", CR, 0
Name2        DATA     "Dick Miller", CR, 0
Name3        DATA     "Mary Evans", CR, 0

' ----- [ Initialization ] -----
Reset:
    HIGH Enable          ' turn of RFID reader
    LOW Latch            ' lock the door!

```

```
' ----- [ Program Code ] -----  
  
Main:  
  LOW Enable                               ' activate the reader  
  #IF __No_SPRAM #THEN  
    SERIN RX, T2400, [WAIT($0A), STR buf\10] ' wait for hdr + ID  
  #ELSE  
    SERIN RX, T2400, [WAIT($0A), SPSTR 10]  
  #ENDIF  
  HIGH Enable                              ' deactivate reader  
  
Check_List:  
  FOR tagNum = 1 TO LastTag                ' scan through known tags  
    FOR idx = 0 TO 9                       ' scan bytes in tag  
      READ (tagNum - 1 * 10 + idx), char   ' get tag data from table  
      #IF __No_SPRAM #THEN  
        IF (char <> buf(idx)) THEN Bad_Char ' compare tag to table  
      #ELSE  
        GET idx, chkChar                   ' read char from SPRAM  
        IF (char <> chkChar) THEN Bad_Char ' compare to table  
      #ENDIF  
    NEXT  
    GOTO Tag_Found                         ' all bytes match!  
  
Bad_Char:                                  ' try next tag  
  NEXT  
  
Bad_Tag:  
  tagNum = 0  
  GOSUB Show_Name                          ' print message  
  FREQOUT Spkr, 1000 */ TmAdj, 115 */ FrAdj ' groan  
  PAUSE 1000  
  GOTO Main  
  
Tag_Found:  
  GOSUB Show_Name                          ' print name  
  HIGH Latch                               ' remove latch  
  FREQOUT Spkr, 2000 */ TmAdj, 880 */ FrAdj ' beep  
  LOW Latch                               ' restore latch  
  GOTO Main  
  
  END  
  
' ----- [ Subroutines ] -----  
  
' Prints name associated with RFID tag  
  
Show_Name:  
  DEBUG DEC tagNum, ": "  
  LOOKUP tagNum,  
    [Name0, Name1, Name2, Name3], idx     ' point to first character  
  DO  
    READ idx, char                        ' read character from name  
    IF (char = 0) THEN EXIT               ' if 0, we're done  
    DEBUG char                            ' otherwise print it  
    idx = idx + 1                         ' point to next character  
  LOOP  
  RETURN
```