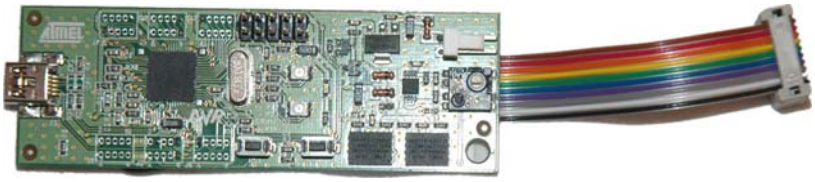
Fast prototyping of a complete RFID System using the Atmel AT90USBKEY and the Melexis EVB90131 RF Interface to Communicate with Atmel RF CryptoMemory® Devices

1. Introduction

- This application note demonstrates the ease of setting up a complete secure RFID system using Atmel's AT90USBKEYdevelopment board, AT88SCxxxxCRF family of cryptographic RFID tags, and Melexis' EVB90131 RF front-end interface. This system prototypes an ISO 14443 type B contactless reader-tag system with USB connection to a PC complete with a graphical user interface application for an interactive user experience. This application note assumes knowledge of AT90USBKEY, AT88SCxxxxCRF, and MLX90131. For additional information, please visit www.atmel.com for ATxxxx product datasheets and www.melexis.com for MLX90131 datasheets.
- The AT90USBKEY is an assembled module built around the Atmel® AT90USB1287 8-bit AVR Microcontroller. It interfaces to a PC through its USB port.

Figure 1. AT90USBKEY Development Board



- Application firmware developed on the PC in downloaded into the flash memory of the AT90USB1287 device using the Atmel “Flip” utility software.
- Alternatively, the AVR® JTAGICE MKII can also serve the purpose of firmware downloads and debug.

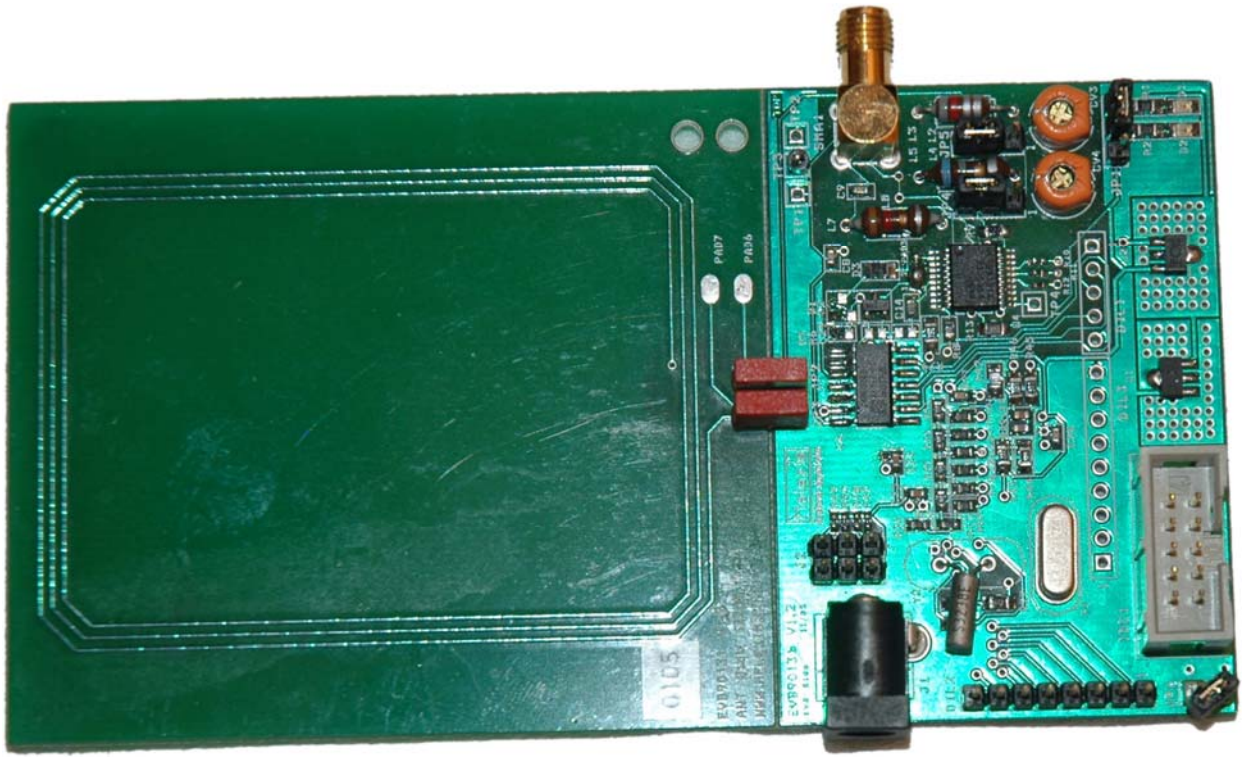
Fast Prototyping of a Contactless Reader for CryptoRF®

Application Note

5256A• CRYPT• 06/08

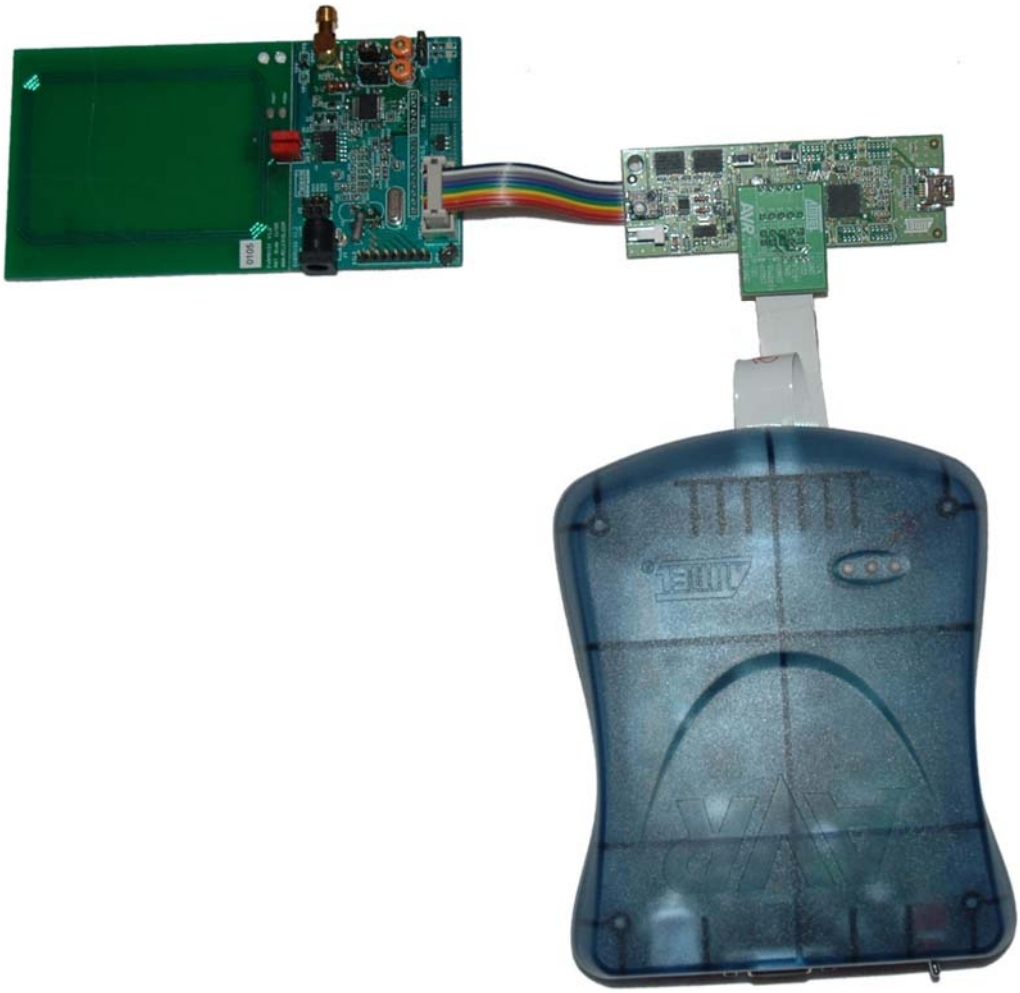
- The Melexis EVB90131 is an assembled module which allows evaluation of the performance of the Melexis MLX90131 13.56 MHz transceiver IC in order to facilitate development of RFID applications.

Figure 2. Melexis MLX90131RF Front-End Interface Board



- The downloaded software manages the ISO 14443-B protocol between the Melexis EVB90131 front end RF interface and the AT88SC6416CRF or any other member of the family of CryptoMemory® contactless devices.

Figure 3. RF Reader System Comprising AT90USBKEY connected to MLX90131 and complete with an AVR JTAGICE MKII Programmer and Debugger



- Firmware running on the AT90USBKEY interfaces to a PC via a Windows® application running on the PC. This application, developed by Atmel is Atmel public software named “Quickterm”. It’s functionality is very similar to that of the well known Windows ”Hyperterminal” application.
- RF CryptoMemories, also known as CryptoRF®, follow the ISO Standard ISO 14443 type B protocol up to layer 3. There is no need to embed CryptoRF commands at level 4 of the protocol. The command byte itself comprises 2 nibbles, the MSB nibble being the CID value and the LSB nibble being the command value itself.

2. Melexis EVB90131 –AT90USBKEY Hardware Interface

- The AT90USBKEY is linked to the Melexis EVB90131 board through a flat cable of 10 connectors

Table 1. Pin Descriptions for Hardware Interface

Atmel AT90USBKEY		Melexis EVB90131 IDC1 connector	
Pin	**Function**	**Pin**	**Function**
J7 10	PA0	1 Dout	3 state data output
J5 10	PC0	2 A	Address
J2 10	PE0	3 WR	Write
J2 09	PE1	4 RD	Read
J5 09	PC1	5 CS	Chip Select
J7 10	PA0	6 DIN	Data In
J2 04	PE6	7 IRQ	Interrupt Request
J7 01	Vcc	8 Raw Pwr	Power supply
J4 02	Gnd	9 Gnd	
J6 02	Gnd	10 Gnd	
Note: External memory interface see AT90USB1287 data sheet page 30			

- RD signal is PE1
- WR signal is PE0

- A8 signal is PC0
- A9 signal is PC1

- Signal PC1 (A9) is used to select register in Melexis chip : PC1 value of 0 implies data register while PC1 value of 1 implies config register.

3. AT90USBKEY USB Firmware

- The firmware in AT90USBKEY includes a Windows Operating system USB driver.
- The USB driver enables the AT90USBKEY to emulate a serial port device (COM X, where X is an available serial communication port number). This Quickterm application running on the PC then interfaces to the AT90USBKEY through this port.
- When loaded the USB driver can be detected using Hardware>Device Manager> Ports as AT90USBxxx CDC USB to UART MGM (COM X).
- The AT90USBKEY can be put back in bootloader mode (to be able to use “Flip” and download code) any time by simultaneously pressing on the Hardware Byte and Reset buttons and then releasing the Reset button. As earlier mentioned, the AVR JTAGICE MKII tool can also be used to interface the AT90USBKEY and update code.

4. QUICKTERM

- Quickterm is an Atmel application software developed on Windows. It comes with an ISO 14443 oriented graphic user interface customizable to create new commands. We have used this feature to create CryptoRF specific commands.
- The newly introduced commands must of course be implemented as specific functions in the code running on the AT90USBKEY driver. The main files for this are RFshell.c and RFshell.h.

5. CryptoRF Specific Commands

- The command set for CryptoRF can be found in the CryptoRF Specification available from Atmel. Information on how to obtain this document is available at your regional Atmel office or from the CryptoMemory product website at http://www.atmel.com/products/securemem.
- Communicating with CryptoRF devices require the use of ISO 14443 type B level 3 atomic commands:
- First, send the protocol REQB command to the CryptoRF tag. The tag will respond with a 4-byte stream corresponding to the PUPI. Using the received PUPI, send an ATTRIB command to set a value for CID. In our demo we set CID to 1.Note that a CID value of 0 is not usable by the CryptoRF device. The tag will respond with an "Answer to ATTRIB" corresponding to, and confirming the CID value. After receiving the CID confirmation, the tag is now ready for tag specific commands as defined by the CryptoRF command set.
- Commands can be classified in 2 categories depending on the required security level.

5.1. Commands Not Using Cryptographic Security Features

- When authentication is not required, the CryptoRF device is protected only by Read and Write passwords for each individual zone. The following commands are available for this level of security:
  - Set User Zone
  - Read User Zone
  - Write User Zone
  - Read System Zone
  - Write System Zone
  - Check Password
- Implementation of these commands do not require specific cryptographic routines .For example, the “Set User Zone” command is coded as:

```
CID (nibble) 0001
PARAM (byte)
CRC1 (byte)
CRC2 (byte)
```

- Where the PARAM byte will be coded as:

Figure 4. PARAM byte code

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
AT	0	0	0	X	X	X	X

AT = Anti-Tearing is enabled if this bit is set (Bit 7 = 1)
xxxx = zone number from 0 to 15

- Notice that the commands do not require ISO 14443-4 protocol level encapsulation.

5.2. Commands Using Cryptographic Security Features

- Since CryptoRF is a highly secure device, some of its commands require background calculations to compute appropriate cryptographic values to enable communication in higher security modes of operation. The requirement for background calculations results from the nature of the cryptographic hardware.
- The hardware cryptographic engine is described in the (confidential) technical specification for CryptoMemory. It is essentially based on a pseudo-random number generator PRNG, with feedback. The same hardware block is used to perform several different functions: authentication, encryption, or encrypted checksum (Message Authentication Code, MAC) computation in order to minimize the silicon area and the cost of the device.
- During high security operation of the CryptoRF device, the reader must mirror cryptographic processes taking place within the device hardware for meaningful communication. The technical specification fully describes these processes in great detail for both host and device, providing pseudo-code to aid implementation.
- To give an example, when working in authentication mode, each command sent by the reader to the device must be mirrored by the processing of an internal state variable in the reader software (gpa_byte). Any loss in synchronization between hardware commands, and software processing will result in unrecoverable cryptographic errors.

6. Demonstration Set up

- This section summarizes all parameters requiring setup for a demonstration.

6.1. Port

- Windows will assign a specific COM port to the USB device. To identify this port simply click on:
  My Computer> Properties>Hardware>Device Manager>Ports and find the port allocated to the CID driver.
- This port must be available in the scrolling window for Port.

6.2. Baud

- Select 38400

6.3. Parity

- Select none

6.4. Data Bits

- Select 8

6.5. Stop Bits

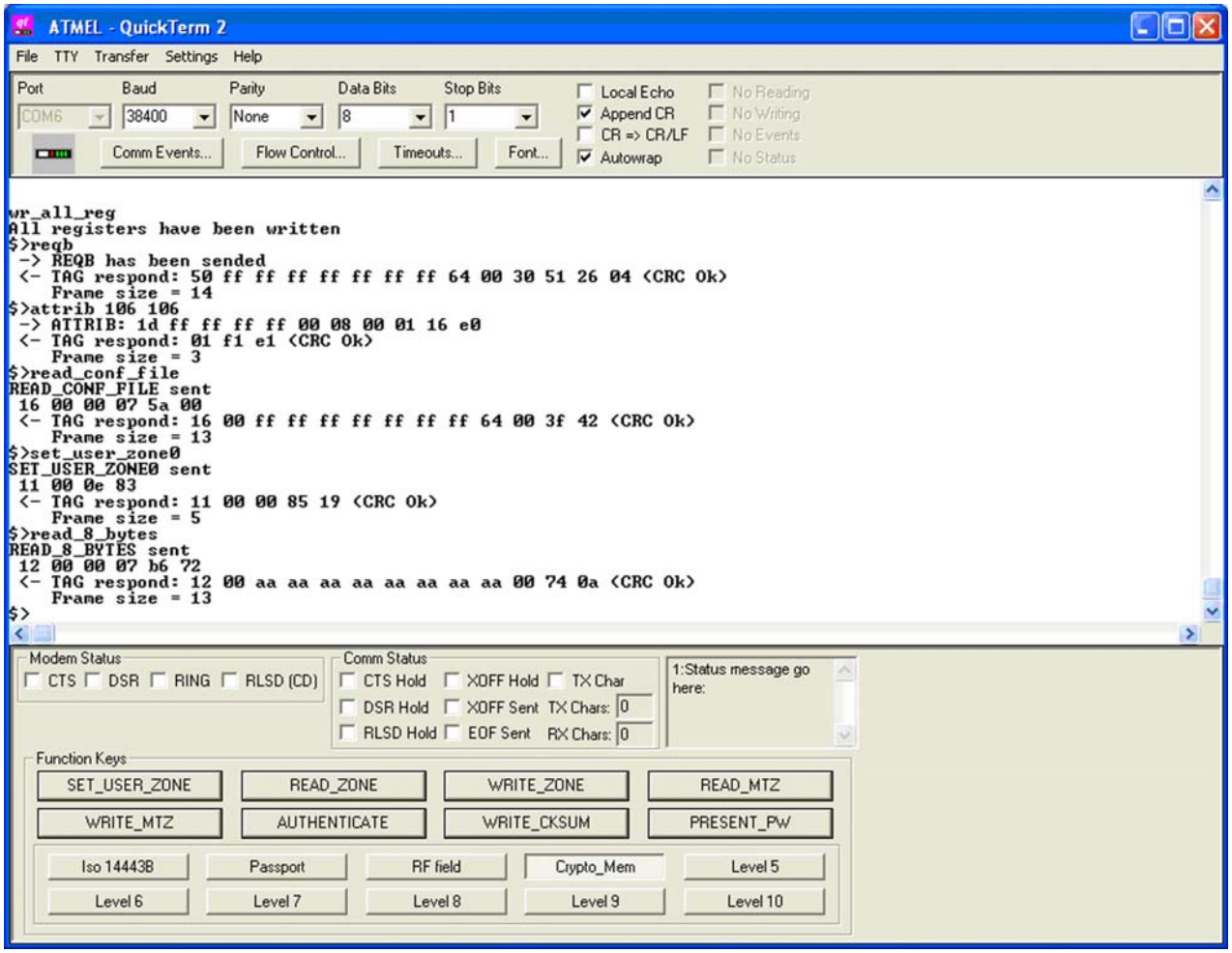
- Select 1
- Then check boxes **Append CR** and **Autowrap**
- The buttons, on the Quickterm interface have been customized to interface with CryptoRF. New buttons have been created on top of the standard Quickterm version:
- The snapshot shown in the next paragraph illustrates the settings of the communication parameters, together with the implementation of non-cryptographic commands.
- The snapshot shows the RF dialog with the contactless device :

>REQB
<ATQB
>ATTRIB
<ANSWER TO ATTRIB
>COMMANDS
<ANSWERS

7. Example of Demo Scenarios

7.1. No Authentication required. Sample AT88SC6416CRF Virgin State

Figure 5. Sample AT88SC6416CRF Virgin State

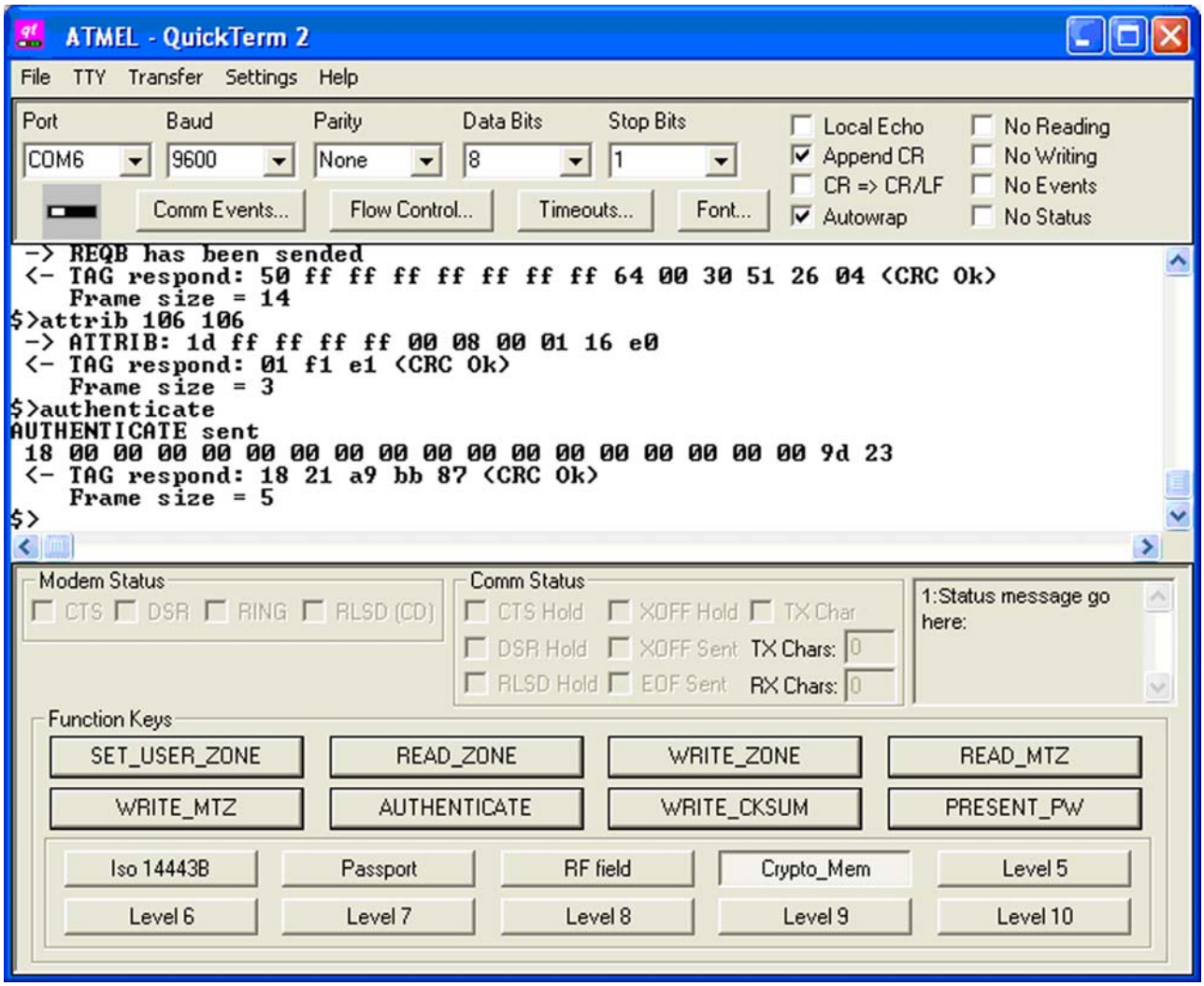


8 Prototyping a Contactless Reader

7.2. Authentication required. AT88SC6416CRF initialized with proper key set

- In this version of the software, commands can only be sent with predefined parameters, therefore it is necessary to assume one specific set of keys in order to play the authenticate command with the correct parameters. (*See software source*). Then we can use the following:

Figure 6. AT88SC6416CRF initialized with proper key set



Appendix A. Source code for the Cryptofunction

- case CMD_AUTHENTICATE:
- In this example we assume a fixed value for the Gi key programmed inside the device; in our case : 0x16 0x6B 0x40 0xB4 0x4A 0xBA 0x4B 0xD6
- We also assume for Q0 the value 0x75 0x23 0x16 0x46 0x74 0x42 0x37 0x12
- Finally we take Ci = 0x00 for bits 0 to 6 and C7 = 0xFF

Cryptofunction Source Code

```
      // compute challenge
Crypto[0] = 0xFF;
Crypto[1] = 0x00;
Crypto[2] = 0x00;
Crypto[3] = 0x00;
Crypto[4] = 0x00;
Crypto[5] = 0x00;
Crypto[6] = 0x00;
Crypto[7] = 0x01;

key[0] = 0x16;
key[1] = 0x6B;
key[2] = 0x40;
key[3] = 0xB4;
key[4] = 0x4A;
key[5] = 0xBA;
key[6] = 0x4B;
key[7] = 0xD6;

Cryptogram[0] = 0x75;
Cryptogram[1] = 0x23;
Cryptogram[2] = 0x16;
Cryptogram[3] = 0x46;
Cryptogram[4] = 0x74;
Cryptogram[5] = 0x42;
Cryptogram[6] = 0x37;
Cryptogram[7] = 0x12;

      SetInit(Crypto,key,Cryptogram);
      Authenticate(q1,q2,q3);

      for (i=0;i<8;i++)
      {authenticate_command[i+2] = Cryptogram[i];
            authenticate_command[i+10] = q1[i];
      }

            crc_14b=
get_crc_14b(authenticate_command,(AUTHENTICATE_LENGTH-2)) ^ 0xFFFF;
      authenticate_command[AUTHENTICATE_LENGTH-2] = crc_14b&0xFF;
      authenticate_command[AUTHENTICATE_LENGTH-1] = crc_14b>>8;
            save_int_conf = disable_int();
```

```
                printf("AUTHENTICATE sent \n\r");
                for (i=0;i<AUTHENTICATE_LENGTH;i++)
                {
                       if (authenticate_command[i] <= 0x0F)
                       printf(" 0%x",authenticate_command[i]);
                else
                       printf(" %x",authenticate_command[i]);
                       }
                       printf("\n\r");
                       sof_14b();
                       for (i=0; i<AUTHENTICATE_LENGTH;i++)
                       {
                              fifo_send_rdy();
                              send_data_14b(authenticate_command[i])
// Send the authenticate command !
                }
                fifo_send_rdy();
                eof_14b();
                fifo_send_empty();
                // rec_data_14b(bFwi, bWtxm);
                //rec_data_14b(0x04,0x01);

                rec_data_14b(0x04,0xFF);

// Receive the answer from the card:
   restore_int(save_int_conf);

   print_report();
                break;
```

8. Revision History

Table 2. Revision History

Doc. Rev.	Date	Comments
5256A	6/2006	Document updated.

Headquarters

Atmel Corporation
2325 Orchard Parkway
San Jose, CA 95131
USA
Tel: 1(408) 441-0311
Fax: 1(408) 487-2600

International

Atmel Asia
Room 1219
Chinachem Golden Plaza
77 Mody Road Tsimshatsui
East Kowloon
Hong Kong
Tel: (852) 2721-9778
Fax: (852) 2722-1369

Atmel Europe
Le Krebs
8, Rue Jean-Pierre Timbaud
BP 309
78054 Saint-Quentin-en-
Yvelines Cedex
France
Tel: (33) 1-30-60-70-00
Fax: (33) 1-30-60-71-11

Atmel Japan
9F, Tonetsu Shinkawa Bldg.
1-24-8 Shinkawa
Chuo-ku, Tokyo 104-0033
Japan
Tel: (81) 3-3523-3551
Fax: (81) 3-3523-7581

Product Contact

Web Site
www.atmel.com

Technical Support
cryptomemory@atmel.com

Sales Contact
www.atmel.com/contacts

Literature Requests
www.atmel.com/literature

5256A–CRYPT–06/08