Implementing Tiny Encryption Algorithm (TEA)

On SAM4S Xplained Pro

Technical Document

Table of Contents

Definitions, Acronyms and Abbreviations

TEA	Tiny Encryption algorithm
ASF	Atmel software Framework

1. Purpose of document

This document is generic technical document of Crypto firmware developed for SAM4S Xplained Pro. This explains the implementation block, code flow and operational procedures.

2. Overview of implementation

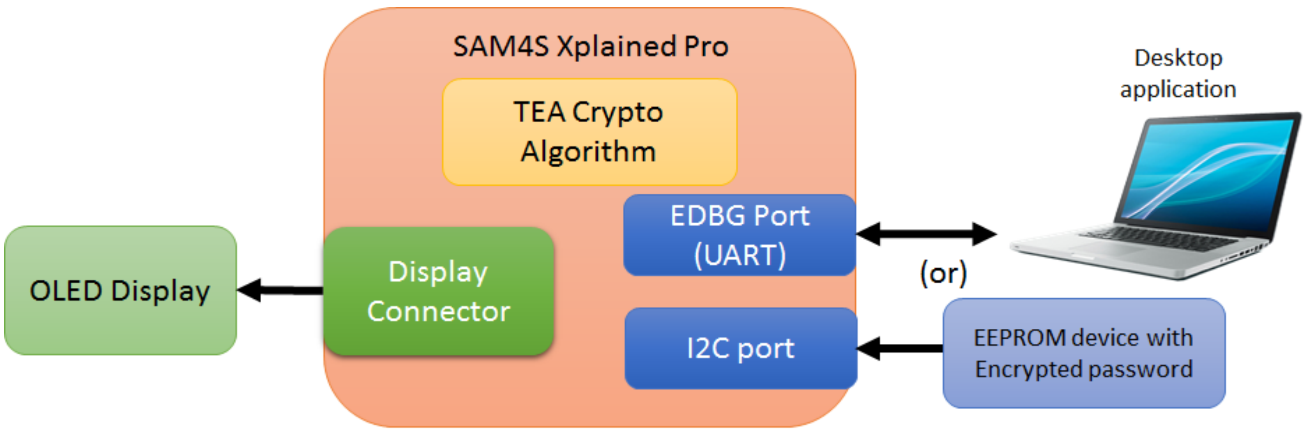
In this project, I'm using TEA encryption crypto algorithm to achieve "Hardware Based Authentication". Also, developed one desktop application which can get 'password' and 'Encryption Key' as input from the user and do encryption.

This desktop application can also send the encrypted data via serial port.

Firmware running on SAM4S Xplained Pro will receive the data sent from desktop application. Then decrypt the data with stored 'Encryption Key' and validate the decrypted password with the 'password' stored in firmware.

This project uses password as "satheesh" and Encryption key as "microchip".

3. Implementation Block Diagram



The above diagram shows the modules involved in project implementation. And the detailed block level description given below.

3.1 TEA Crypto Module

This module takes care of doing Encryption and Decryption of data based on the "Encryption key ".

This module used in both applications,

1. In firmware implemented to decrypt the password data received from desktop application.
2. In desktop application implemented using "c#.net" to get password and Encryption key as input then Encrypt the password.

3.2 EDBG Port(UART)

This serial port module will receive encrypted data from desktop application and use the same for authentication using TEA Decryption module.

Also, sends back the status of validation over UART to Desktop application.

3.3 Display

This module show the authentication status. Which displays the validation information along with the Decrypted password displayed on screen.

3.4 I2C module

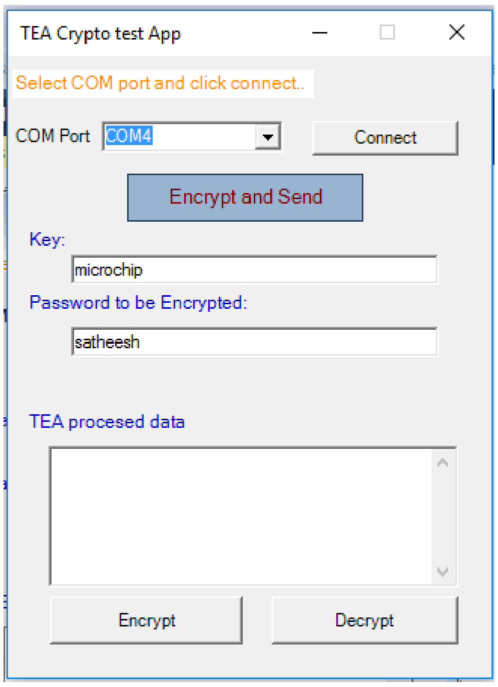
This is an ***alternate approach***, in which we can store the encrypted password inside I2C based EEPROM. This will work as hardware based encryption device.

User has to plug in this I2C module to SAM4S Xplained pro hardware. The firmware will read the encrypted password stored in the EEPROM memory and do authentication.

3.5 Desktop application

This desktop application which can get ‘password’ and ‘Encryption Key’ as input from the user and do encryption. This application can also send the encrypted data via serial port.

User can give any password and any key to do encryption and decryption testing. The application window shown below.


TEA Crypto test App

Select COM port and click connect..

COM Port: COM4

Key: microchip

Password to be Encrypted: satheesh

TEA procesed data

4. Firmware

This firmware developed using Atmel studio 7 and ASF. This has the customized TEA Crypto algorithm.

This takes care of receiving serial encrypted data from desktop application and performs authentication by decrypting the serial data.

Also displays the decrypted password and authentication status

Can find the Entire project workspace used to develop this firmware @ https://github.com/satheelpt/Hands-On-Assignment.git

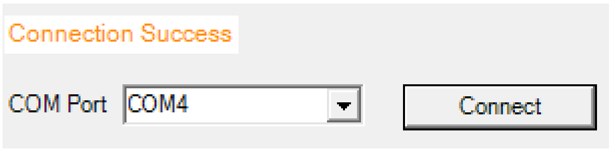
5. Operational procedure

5.1 Using Desktop Application

This application uses serial port with below mentioned configuration.

Baudrate	→115200
Databit	→8
Parity	→None
Stopbit	→1

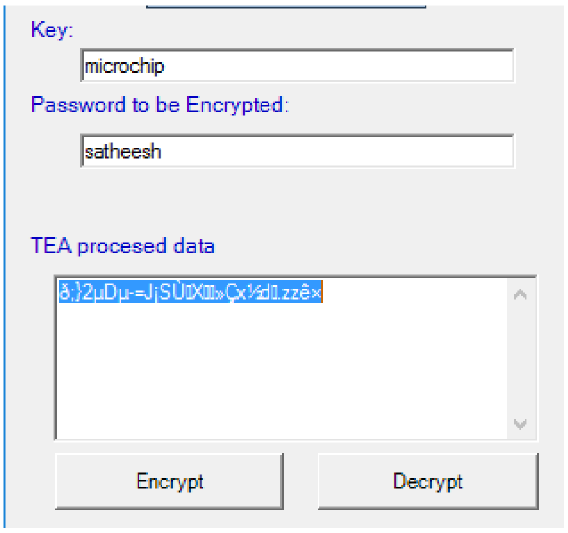
1. User can select COM port from drop down list and click connect button to make connection with SAM4S. connection status will be displayed on top of application window as shown below.



Connection Success

COM Port: COM4

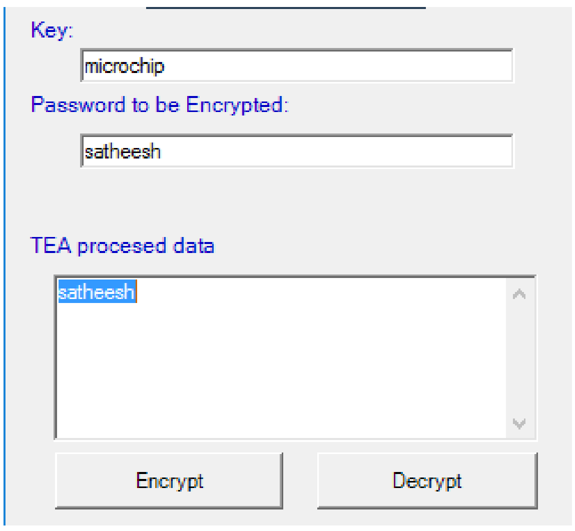
2. Now user can change the 'Key' & 'Password' from the text boxes. Then click 'Encrypt button'. Now the encrypted value will be displayed under 'TEA processed text' field. And click 'Decrypt button' to perform decryption of processed data.



Key: microchip

Password to be Encrypted: satheesh

TEA procesed data: ð;}2µDµ-=J¡SÙXɷ»Çx½d.zzêx

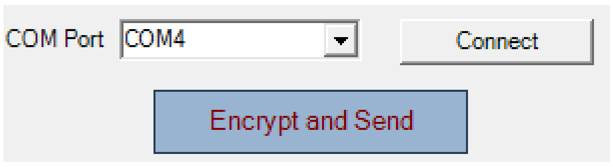


Key: microchip

Password to be Encrypted: satheesh

TEA procesed data: satheesh

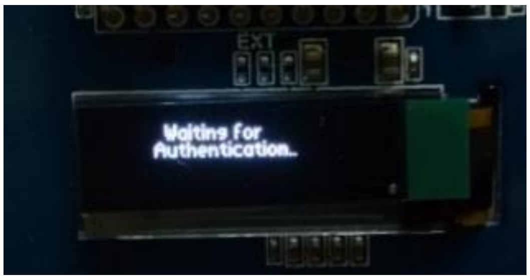
3. Click ‘Encrypt and Send’ button to send the encrypted data to the associated COM port connected.



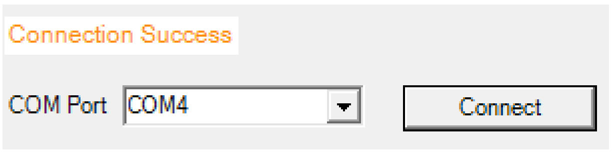
5.2 Project working

1. Connect the EDBG USB port of SAM4S Xplained pro to desktop using Micro USB cable.

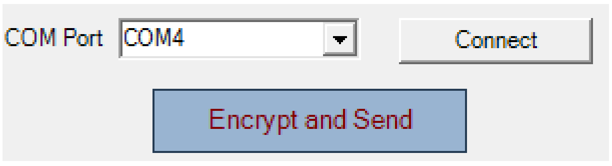
2. Build and run the application on SAM4S Xplained pro using Atmel studio 7. Now OLED will display the text “Waiting for Authentication” as shown below.



3. Now run the ‘TEA crypto test application’ on PC. Select COM port of SAM4S device and click connect. Now it’ll display status as “Connection Success”.



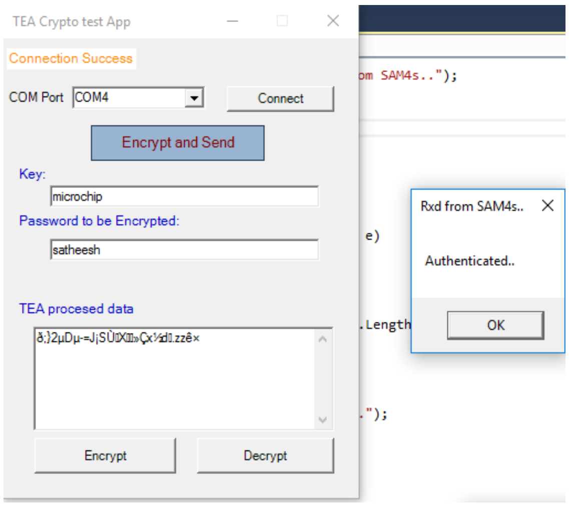
4. Now click ‘Encrypt and Send’ button. This will do encryption and send the encrypted data to SAM4S via serial port.



5. SAM4S uses ‘satheesh’ as password and ‘microchip’ as encryption ley. Any other values apart from this will not authenticate the device. Various combination of inputs and relevant SAM4S outputs are shown below.

A. Password → 'satheesh' and Key → 'microchip'

Desktop application received status as "Authenticated.." from SAM4S. And SAM4S OLED will display "Welcome" along with the Decrypted password "satheesh".


TEA Crypto test App

Connection Success

COM Port: COM4 | Connect

Encrypt and Send

Key: microchip

Password to be Encrypted: satheesh

TEA procesed data: ð;}2µDµ-=J¡SÜÌXII»Çx½đl.zzê×

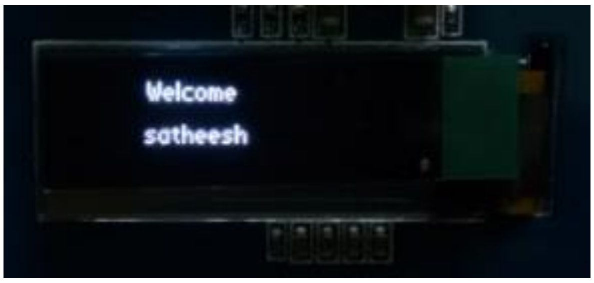
Encrypt | Decrypt

```
om SAM4s..");
e)
.Length
.");
```

Rxd from SAM4s..

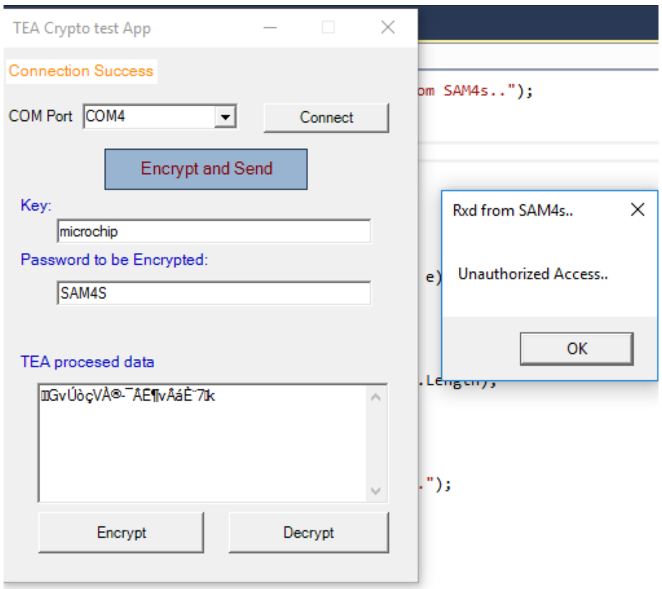
Authenticated..

OK



B. Password → 'SAM4S' and Key → 'microchip'

Desktop application received status as "Unauthorized Access.." from SAM4S. And SAM4S OLED will display "Unauthorized Access.." along with the Decrypted password "SAM4S".


TEA Crypto test App

Connection Success

COM Port: COM4 | Connect

Encrypt and Send

Key: microchip

Password to be Encrypted: SAM4S

TEA procesed data: IIGvÚòçVÀ®¯ÀE¶vÂäÈ7k

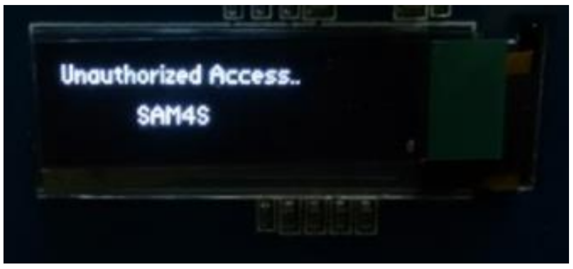
Encrypt | Decrypt

```
om SAM4s..");
e)
.Length);
.");
```

Rxd from SAM4s..

Unauthorized Access..

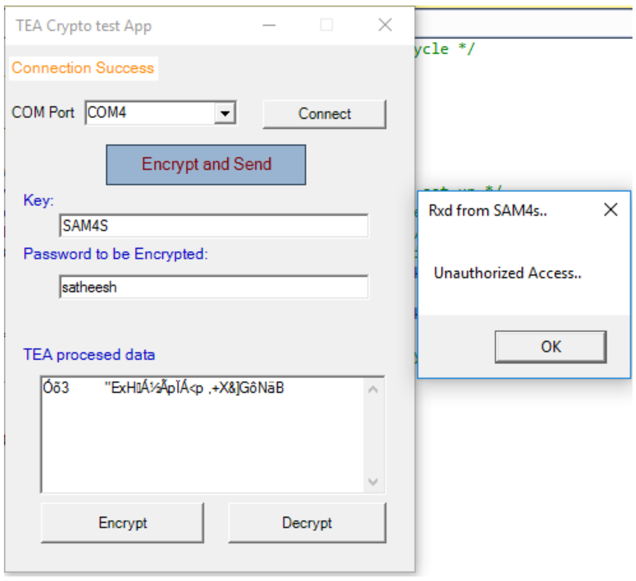
OK



C. Password → ‘satheesh’ and Key → ‘SAM4S’

Desktop application received status as “Unauthorized Access..” from SAM4S. And SAM4S OLED will display “Unauthorized Access..” along with the Decrypted password “”. In this case, decrypted characters may not have printable ASCII characters, so password will not be displayed properly.

If we update the OLED font table for those ASCII values, those characters can also be printed.


TEA Crypto test App

Connection Success

COM Port: COM4

Key: SAM4S

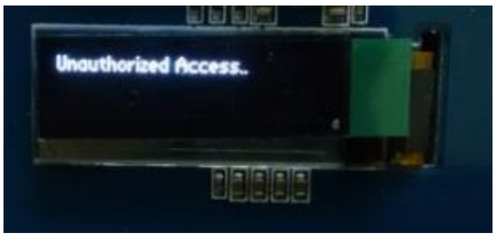
Password to be Encrypted: satheesh

TEA procesed data

```
Óõ3      "ExH▯Á½ĂpÏÁ<p ,+X&]GôNäB
```

Rxd from SAM4s..

Unauthorized Access..



5.3 References

- For TEA implementation used code from http://codereview.stackexchange.com/questions/2050/tiny-encryption-algorithm-tea-for-arbitrary-sized-data and corrected errors ,modified few logics to make it work in ‘c’.
- For Firmware used example applications as reference.
- For Desktop application, keeping the TEA ‘c’ code logic as reference developed GUI in visual studio 2013 using C#.net.