Security Communications Center

Preliminary Design Report
Team 3

Date: 09.12.2011

Authors:

Oğuzhan Avcı (HW Engineer)

Abdullah Başar Akbay (HW Engineer)

Kemal Cağrı Bardakçı (SW Engineer)

İsmail Akpolat (SW Engineer)

Version 1

Table of Contents

1. Product Overview

Secure Messaging and Location Tracking project can be called as a messenger program being run on smart phones (Android operating systems) and having the feature of enabling users to see the positions of each other on a map. All the information shared within the group is encrypted.

In this project, a portable cryptographic device, which encodes the data sent from the mobile device, will be designed. Public key encryption with RSA Algorithm will be used. The connection between the cryptographic device and mobile phone is done through Bluetooth technology.

A user interface, which displays the locations of the other users on a map, is designed. The location information is read from the integrated GPS chip inside the smart phone. Obtained coordinates are interpreted and users are demonstrated on the map with symbols. This user interface also enables the user to send and receive encrypted messages via this application. Together with the location information, the plaintext is sent to the cryptographic device via Bluetooth. Encrypted information is transmitted to the other users via a HTTP Server application.

The received and transmitted messages are stored during the session. Last few locations of the users are also stored internally in the program in case of a GPS or Internet connection loss. In such unexpected emergency cases, other users who still have the connection can reach the last location of the user who has lost his connection.

2. Preliminary Design

2.1 Block Diagram

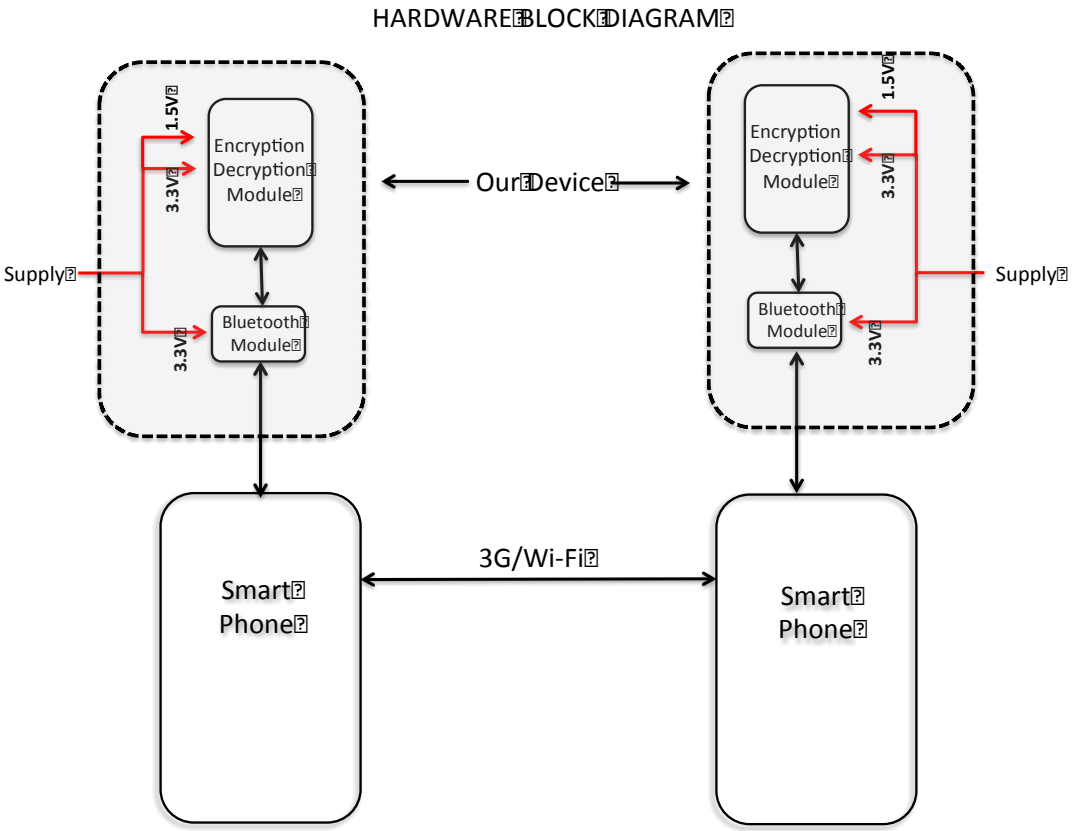


Figure 1: Block Diagram

2.2 Components

2.2.1 Bluetooth Module

We will use Texas Instruments' CC2540 Bluetooth chip, which is shown in figure 2. This module's task is to receive the location data from the smart phone via Bluetooth, and transmit the encrypted data to smart phone. The specifications of the CC2540 are as follows: [1]

- Dimensions: 6-mm × 6-mm Package
- Operating voltage: 2V-3.6V
- Current Consumptions:

  Active Mode RX Down to 19.6 mA

  Active Mode TX (–6 dBm): 24 mA

  Power Mode 1 (3-µs Wake-Up): 235 µA

  Power Mode 2 (Sleep Timer On): 0.9 µA

  Power Mode 3 (External Interrupts): 0.4 µA
- Operating Temperature: -40C – 85C
- Operating Frequency: 4MHz (Master & Slave RX and TX)

Figure 2: CC2540 Texas Ins. [1]

2.2.2 FPGA Chip

Overview

FPGA Chip is used to implement Decryption/Encryption Block (D/E Block). In Product Specification Document, under Functional and Performance Specifications heading it has been expressed that the encryption method must be Public Key Cryptography. Furthermore, it has been added that it should use a key with sufficiently higher number of bits. The minimum length of the key is determined as 128 bits. ALTERA EP1C3T100C8N, which has 2910 logic elements, will be used to implement the encryption/decryption algorithm. The FPGA chip has the following specifications: [2]

- Supply voltage (V_{CCINT}): -0.5V-2.4V
- DC Input Voltage: -0.5V-4.6V
- Output supply voltage: 3.0V-3.6V
- High-level input voltage: 1.7V-4.1V
- Low-level input voltage: -0.5V-0.7V
- Supply voltage for internal logic and input buffers: 1.425V-1.575V
- Supply voltage for output buffers, 3.3-V operation: 3.0V-3.6V
- Storage temperature: -65C-150C
- Ambient temperature: -65C-135C
- Junction temperature: 135C(max)

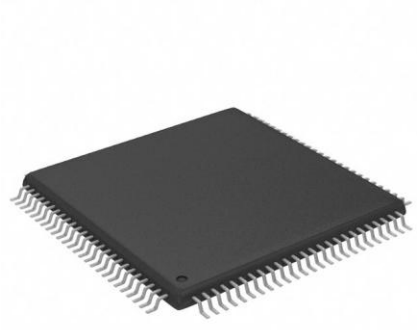


Figure 3: ALTERA EP1C3T100C8N [2]

Objectives of the D/E Block

Decryption/Encryption Block (D/E Block) is responsible for the decryption and encryption of the data. In Product Specification Document, under Functional and Performance Specifications heading it has been expressed that the encryption method must be Public Key Cryptography. Furthermore, it has been added that it should use a key with sufficiently higher number of bits. The minimum length of the key is determined as 128 bits. RSA public key encryption algorithm is chosen since it is the most widely-used public key algorithm [3] and extensive research which has been made on RSA Algorithms and their hardware implementations can be reached.

It has been decided that the hardware implementation of the Public Key algorithm is done on FPGA Chips. It has been pointed out in Block Diagram report that FPGAs offer two important features: flexibility and integration. With the same circuitry and connections, the implementation of different encryption algorithms can be realized. It is surely depended upon the numbers of pins; however, it can be said that FPGA Chips have relatively small sizes and weights which does not violate the size and weight conditions expressed in the Product Specification Report.

The National Institute of Standards and Technology (NIST) is a measurement standards laboratory which is appurtenant to the United States Department of Commerce [4]. It is not a regulatory agency authority which does not possess official control on the products; however, it is a prestigious constitution and determines the minimum specifications in any field technology [5]. In its one of very recent publication on Computer Security, it is clearly expressed that using n bits of security strength (RSA Key Lengths) such that $1024 \leq |n| \leq 2048$ is acceptable for 2010-2013 in digital signal generation and verification; however, after 2013 only security systems which can provide n bits of security strength

larger than 2048 will be allowed [6, pg:5]. Same requirements are also valid for RSA-Keys which are used in the decryption/encryption of the other keys for transportation [6, pg: 6].

It is very coherent that our specification of 128 bits of key length does not reflect the current widely accepted standards in cryptography. This is a misunderstanding caused by the key length strength difference between symmetric and asymmetric key algorithms. Another publication of NIST [7, pg:63], comparable strengths of symmetric key algorithms and RSA is clearly expressed that 80 bits of security used in symmetric key algorithms (2TDEA) corresponds to 1024 bits of security in RSA algorithm. Similarly, 112 and 128 bits of security in symmetric key schemes correspond to 2048 and 3072 length of key sizes in RSA Algorithm.

Generating 2048 bits of RSA Key Lengths is a challenging task in terms of both hardware constraints and algorithm difficulty. A recent study [8] shows that a 1024-Bit RSA Encryption Algorithm can be implemented on XC3S400 FPGA using 50MHz clock system. Same study shows that “For a practical application the circuits can encrypt/decrypt a sequence of 1024 with an exponent on 1024 bits in 212.99ms” [8, pg: 4]. Comparing to these results with the cost, time and security conditions expressed in both Product Requirements and Product Specifications Documents, it can be concluded that 1024 bit RSA Encryption can be implemented on mid-cost (costs of XC3S400 FPGAs are around 25 Dollars [9]).

Although decryption and encryption of data can be done less than a second, as it has been put as a specification in Product Specification Report, using RSA algorithm with 1024 bits of keys, it is actually a slower way of cryptography comparing to symmetric

algorithms [10]. This is especially valid if the data which is desired to be transmitted is large. Therefore, asymmetric key algorithms such as RSA algorithm is used when a session key generated for a symmetric-key encryption algorithm is desired to be transported [10]. In practice, symmetric and asymmetric key algorithms are not regarded as the alternatives of each other. Symmetric key algorithms are used to encrypt and decrypt data and its keys are transmitted using public key encryption methodology.

In this preliminary design, the discussion made above has been discarded. The discussion on the revises of Product Specification Document and Block Diagram & Flowchart Report is postponed to the results of the implementation of the 128-Bit RSA D/E Block. This may be regarded as a cautious approach; however, it is necessary since the hardware and software engineers who participate in the implementation of this product ("Cher Ami") are inexperienced. The papers referenced above have been written by the experts of this field. Therefore, at this stage of the product design; main purpose is the successful implementation of an D/E Block which uses RSA Public Key Encryption algorithm with 128-bit length keys in order to decrypt and encrypt data coming from smart phone of the user. Consistent with this decision, FPGA choice is kept as ALTERA EP1C3T100C8N which has 2910 logic elements as it has been indicated in Block Diagram & Flowchart Report.

A similar discussion can also be added for the implementation of symmetric key methodology. Increasing the number of security bits from 128 to 1024 will cause a significant increase key generation, encryption and decryption periods. As it has been explained under Explanation of Internal Working of D/E Block subheading below, using Public Key Algorithm to encrypt message and location data is already an exhausting procedure since it requires that each cipher text must be created using the recipient's

public key and unique to the recipient. After completion of 1024 bit public key encryption algorithm is developed, a symmetric key scheme for the share of the data will be developed. However, symmetric-key encryption is not a challenging algorithm comparing with the RSA Algorithm and can be implemented following the completion of 1024 Bit RSA Algorithm implementation on FPGA. It should also be noted that proper implementation of symmetric-key algorithms require usage of public-Key algorithms; therefore, it is natural to regard implementation of symmetric-key algorithm on FPGA as a further step which follows implementation of RSA algorithm.

In the next stages of the product development, it is targeted to develop the D/E Block to a level which can satisfy the specifications of NIST. These developments can be done by improving the D/E Block algorithm and altering the FPGA Chip in the future. By the end of December, it is planned to implement 128-bit RSA Encryption algorithm on Basys™2 Spartan-3E FPGA Boards. In January, D/E Block algorithm will be developed to satisfy the requirements of NIST. According to these experiences and simulation results, final FPGA Chip will be decided. This choice may cause to change the power regulator circuitry afterwards. However, power regulator circuitry designs are taken from the producers of the FPGA and power regulator chips. Therefore, variation of the power regulator circuitry design does not require an extra effort on the HW design group except a time delay. Considering that the main challenging part of this product design is algorithm development, these minor changes can be accepted.

Relationship between Smart Phone and D/E Block

As it has been explained under the product description heading, all data shared within the network is encrypted. Consider a network consisting four different users (User A, B1, B2, B3). Notice that Figure1 is only a simplified version of the real network diagram. It only shows the data transfer diagram from the perspective of User A and explain the function of D/E Block in the example of User A.

During the implementation of RSA algorithm, two different keys are generated: public and private. Data is encrypted using the public key; however, it can only be decrypted using the private key. Considering the user network given above, User A generates his public/private keys at the beginning of the session. He declares his public key afterwards and calls for the declaration of the other users' public keys. He stores his own private key and other users' private keys. This process can be called as Key Exchange and it has been shown on Figure1. Details of this procedure will be determined by the software group.

In key exchange process, D/E Block communicates with the User A smart phone via Bluetooth connection using a predetermined protocol. When user logs into the network, smart phone sends a signal to D/E Block to start key generation procedure. When keys are generated, D/E Block sends a signal to the smart phone and indicates that keys are ready. Another protocol for the transfer of the public keys coming from different users is also defined. Smart phone sends the public keys of the users. It has been indicated in the Product Specification Report (page: 8) that “A group consists of 10 person maximum”. This indicates that, 9 different user identities will be defined beforehand.

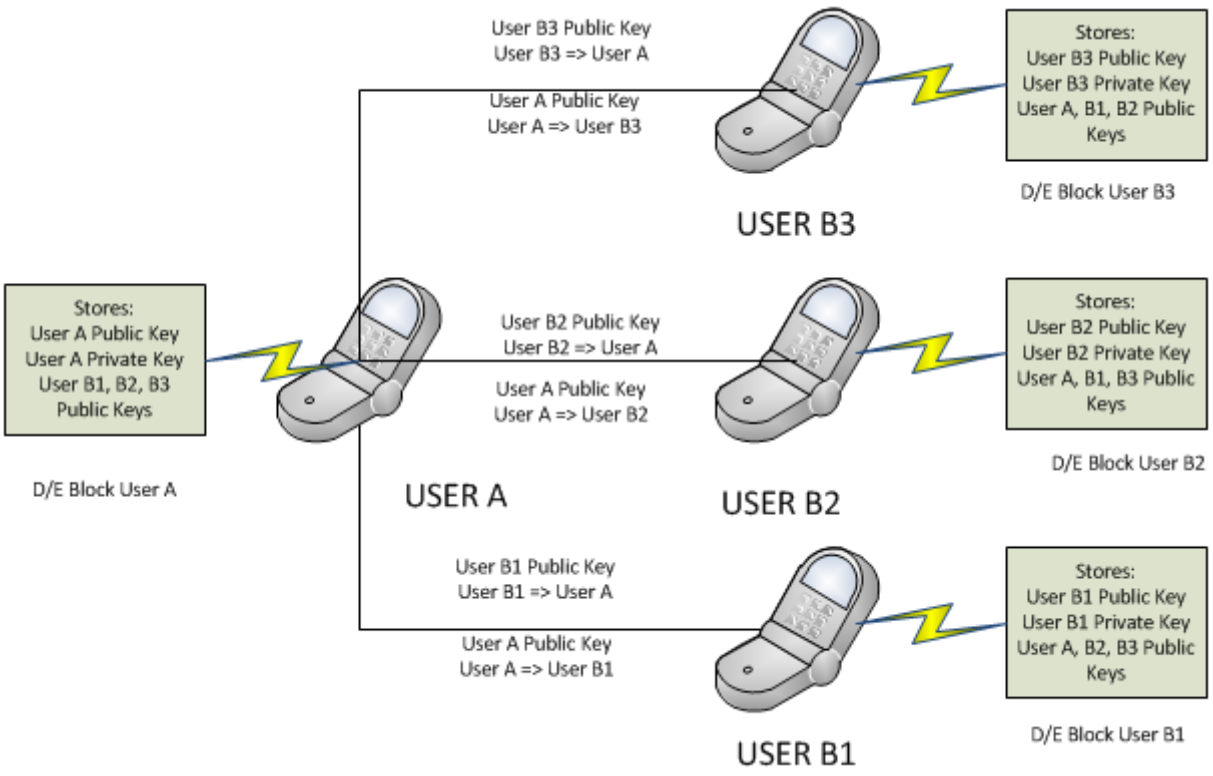


Figure 4 – User Network and Key Exchange Procedure

Similar to the key exchange procedure, other protocols for the transmission of the encrypted and decrypted data between smart phone - D/E block and between smart phones are required to be defined. In this part of the Preliminary Design Report, details of the data transmission protocol within the network are not given. It is a duty of SW group.

In RSA Algorithm, a plaintext is represented as an integer within {0, 1, ... ,n-1} where n is the key size [8, pg: 286]. For 128-bit RSA encryption scheme, n = 128 which indicates that plaintext must be represented as an integer within {0, 1, ... ,n-1}. An ASCII character is represented by an integer between 0 and 127. Smart phone sends the message and location data represented by ASCII characters and D/E Block encrypts each character using the same key and independently.

When a data from User A will be sent to other users, smart phone first send the plaintext to D/E Block. D/E Block receives the incoming signal from the Bluetooth module. Since Bluetooth is a serial communication channel; however, Bluetooth module is capable of sending data as a 8 bit vectors. Smart phone sends the data as 10 Byte packages. To exemplify, if the message has 300 characters (which is determined as the maximum length for a single message in Product Specification Document), smart phone sends this message in 30 successive packages. For k^{th} term, phone sends the k^{th} package, waits for k^{th} encrypted package and sends the $(k+1)^{th}$ package after the reception of the k^{th} encrypted package. Notice that this procedure has to be repeated for the number of users since each user has a different key pair and therefore must receive a different ciphertext. In addition, phone indicates the user identity at the beginning of the procedure.

When a data is received by User A from other users, smart phone first send the ciphertext to D/E Block. D/E Block receives the incoming signal from the Bluetooth module. Again this transmission process is done by separating the ciphertext into 10 byte packages. In this time, D/E Block decrypts the message using its own private key and transmits to the phone.

In the future, RAM Components driven by a microprocessor to be used as buffers can be considered in order to increase the data transfer rate. The clock frequency of the D/E Block has not been determined yet.

Examples of the Protocol Signals:

Necessity of the protocol signals have been defined above. Under this subheading, three of these protocols have been defined. Determination of the rest of the protocols including reset and initialization procedures have not been discussed in this report. They are

postponed to the later stages of the product design. These examples are given to illustrate the communication protocols which are mentioned many times in the explanation of the relationship between D/E Block and Phone.

It has been indicated above that characters will be transmitted as ASCII Codes. In ASCII coding, some codes are separated as control codes. In key exchange and data transmission protocols, these control codes have been used [11].

1.At the beginning of any procedure, smart phone checks the connection between the D/E block and itself. This procedure is defined as the following:

Phone sends BELL (BEL) control character to D/E Block as a request for a response from the D/E block. BEL Control character has decimal and hexadecimal value of 7. If there does not exist any connection problem, D/E block responses affirmatively by sending ACKNOWLEDGE (ACK) control character. ACK control character has decimal and hexadecimal value of 6.

2. Any process in the D/E Block can be stopped by phone. Phone sends CANCEL (CAN) control character to D/E Block in order to terminate the current procedure between phone and D/E block. After the reception of this signal, D/E cancels its process and responses to the phone by sending ACK signal to indicate cancellation command has been implemented. CAN control character has decimal value of 24 and hexadecimal value of 18.

3. When a user is log on, session keys are generated as explained above. Phone starts key generation procedure by sending DEVICE CONTROL ONE (DC1) signal. DC1 control character has decimal value of 17 and hexadecimal value of 11. After the reception of

DC1, D/E Block calls for an approval of this action by sending a DC1 signal to phone. Phone responses this by sending another DEVICE CONTROL TWO (DC2) signal. D/E Block starts generations of new keys and when it has concluded it sends DC2 signal to indicate key generation process has concluded. Phone responses this DC2 signal with ACK signal and waits for the transmission of the public key. In the rest of the procedure, data transmission protocol from D/E Block to phone is applied.

A key generation process is started after the approval since key generation is a delicate and important procedure. If an error occurs and D/E Block is invoked for key generation, it must call for an approval. If phone receives an accidental DC1signal, it must respond with a CAN signal to cancel the process in the D/E Block.

2.2.3 Power Supply

For power supply, we are going to use SANYO Li-ion Prismatic (UF103450P) 3.7V 2000mAh with Fuse battery, which will supply 3.3V and 5V to the Bluetooth and FPGA chips. The battery has the following specifications: [12]

- Nominal Voltage: 3.7V
- Nominal Capacity: Min. 1880mAh
- Charging Voltage: 4.2V
- Charging Time: 2.5hrs
- Weight: 38.5g
- Dimensions: 10.60mm x 33.90mm x 48.80mm (TxWxH)

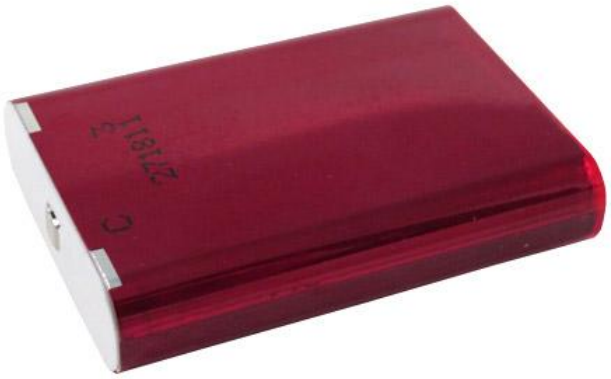


Figure 5: SANYO Li-ion Prismatic (UF103450P) Battery [12]

We are also going to use voltage regulators to ensure 1.5V and 3.3V for FPGA as it is very sensitive to the voltage and current variations. For 1.5V, LF15AB, which has the following specifications, will be used: [13]

- Output voltage: 1.485V-1.515V
- Operating input voltage: -2.5V-16V
- Output noise voltage: 50uV
- Control input logic low: 0.8V
- Control input logic high: 2V
- Temperature range: -40^{o}C – 125^{o}C

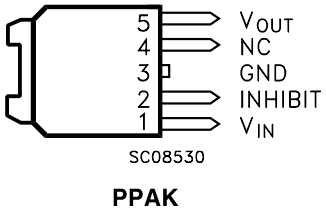


Figure 6: Pin configurations of LF15AB [13]

For, 3.3V, LF33CDT-TRY type regulator, which has the following specifications will be used: [13]

- Output voltage: 3.234V-3.336V
- Operating input voltage: 16V(max)
- Output noise voltage: 50uV
- Control input logic low: 0.8V
- Control input logic high: 2V
- Temperature range: -40^{o}C – 125^{o}C

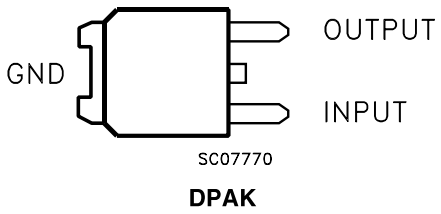


Figure 7: Pin configurations of LFCDT-TRY [13]

2.3 Product Tree

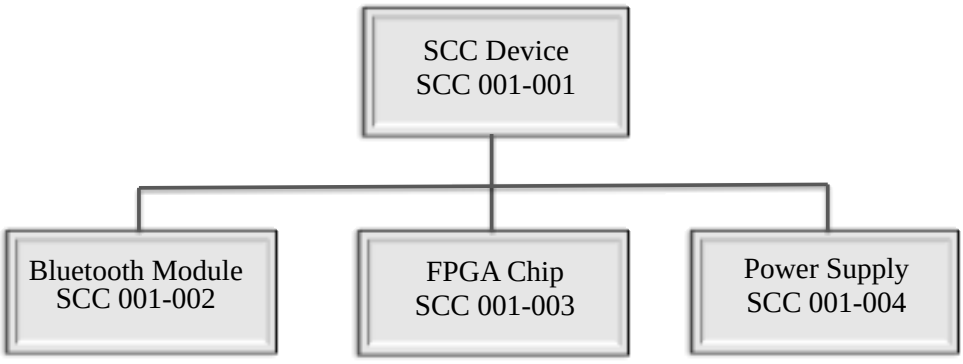


Figure 8: Product Tree

Assembly Stock Number	**Assembly Part Description**	
001001	SCC Device	
Stock Number	**Part Description**	**Number Used**
001002	Bluetooth Module	1
001003	FPGA Chip	1
001004	Power Supply	1
Drawing No	**Prepared By**	**Checked By**
001001-BOM	Oguzhan Avci	A. Basar Akbay
Assembly Part Description	**Date**	
SCC Device	09/12/2011	

Table 1: Bill of Materials

2.4 Software Flowchart and User Interface

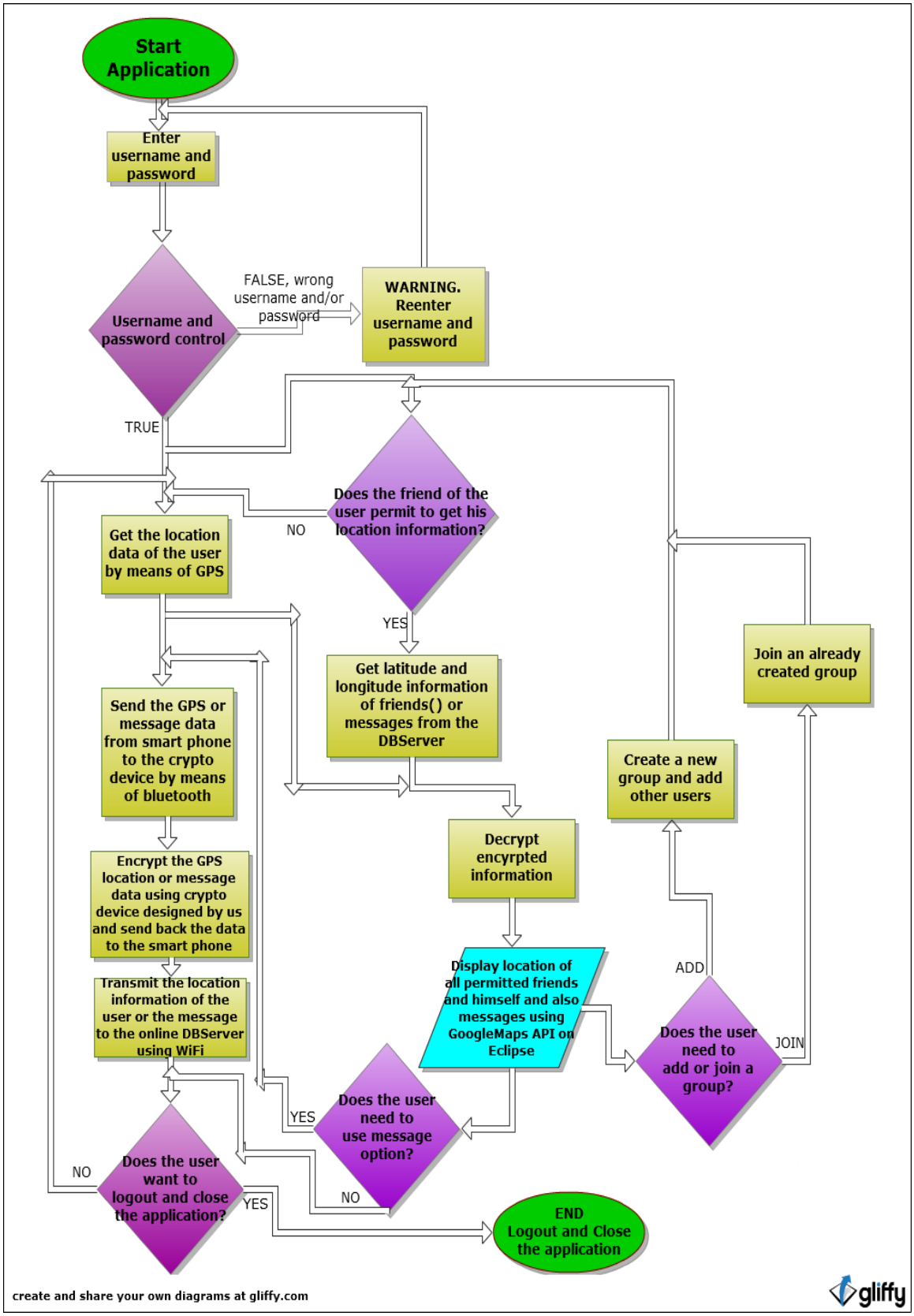


Figure 9: Flowchart of Software

At the beginning of the application, user will be asked for his username and password. If authentication fails user will be asked to reenter his information through warning message, otherwise he will be directed into main page of the application.

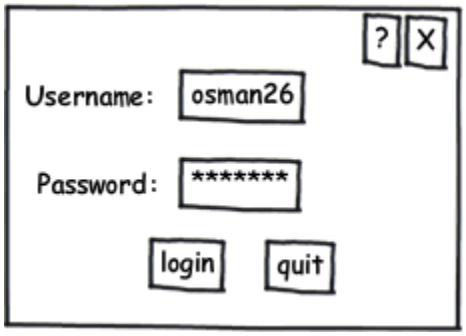


Figure 10: Login Screen

Later on user's location information is taken from gps device and is sent to crypto device for encryption. After the encryption which is stated above in detail as a part of hardware explanation is completed, encrypted data is sent back to the smart phone. Incoming data is transmitted into database server via internet connection. Last 10 locations of each user are hold in database to enable users to guess where their offline friends might be. After gps locations are taken from users, they are distributed to friends of those users to be shown on their screen.

Figure 11: A View from Application

Before showing transmitted data on maps of users, these data are sent to the encryption device for decryption. Online users are represented with green color and offline users are represented with gray color. Users are able to see photos of other users by touching symbols of them. A user is able to see distance between own position and other users' position in terms of longitude and latitude. In addition, last 10 locations of offline users are shown below distance information. In the case of message encryption, texts are sent through the same path as location information sent. However, encrypted texts are sent directly to desired user/s instead of sent into database for recording. Message decryption phase is same with location decryption phase.

Figure 12: Message Dialog Screen

Users are able to create new groups and join groups which are created before. Groups consist of up to 10 users.

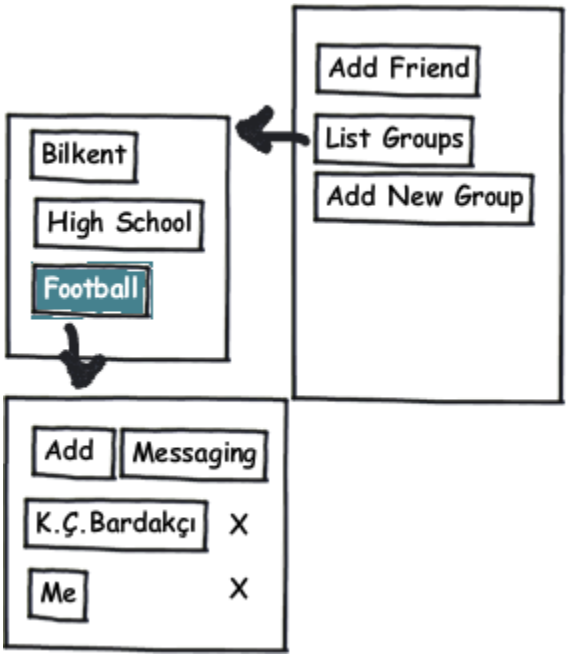
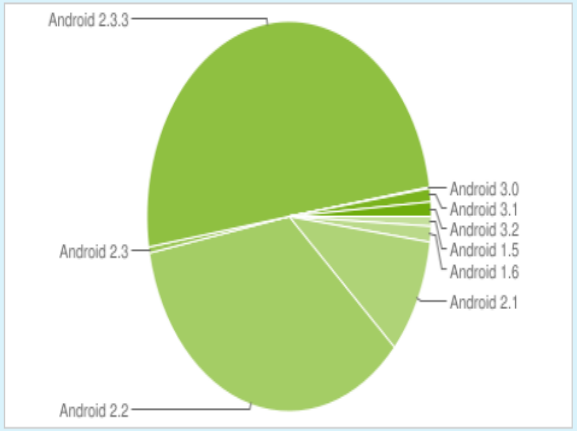


Figure 13: Add/Join/List Group Screen

Users have the right to prevent "his location to be seen by others".

2.5 Software Structures

The application is developed for smart phones which have android operating system. Java will be used as a programming language and application is developed by using Java Eclipse environment. Google Maps Api for Eclipse IDE will be used to generate maps of the application. Since Google Maps Api with Eclipse IDE is used widely to develop such kind of map applications, finding out solutions to encountered problems will be much easier for us. Android version 2.1 will be used to develop this tracking system because smart phones with a newer Android version will have the chance to use this system. By choosing this version, the application will be supported by almost 98 % of the smart phones with android system as it is seen in figure-13 which shows the version used by the android users who access android market during 14-day period. [14]



Platform	Codename	API Level	Distribution
Android 1.5	Cupcake	3	0.8%
Android 1.6	Donut	4	1.3%
Android 2.1	Eclair	7	9.6%
Android 2.2	Froyo	8	35.3%
Android 2.3 - Android 2.3.2	Gingerbread	9	0.5%
Android 2.3.3 - Android 2.3.7		10	50.1%
Android 3.0	Honeycomb	11	0.1%
Android 3.1		12	1.1%
Android 3.2		13	1.2%

Data collected during a 14-day period ending on December 1, 2011

Figure 14: Android Version Distribution

5. References

[1] 2.4-GHz *Bluetooth*® low energy System-on-Chip Datasheet: http://www.ti.com/lit/ds/swrs084c/swrs084c.pdf
[2] http://datasheet.octopart.com/EP1C3T100C8N-Altera-datasheet-2871.pdf
[3] *"Introduction to Public Key Cryptography"* Last Accessed on 18.11.2011 *http://www.verisign.com.au/repository/tutorial/cryptography/intro1.shtml*
[4] *"National Institute of Standards and Technology"* Last Accessed on 08.12.2011 http://en.wikipedia.org/wiki/National_Institute_of_Standards_and_Technology#Measurements_and_standards
[5] *"NIST General Information"* Last Accessed on 08.12.2011 http://www.nist.gov/public_affairs/general_information.cfm
[6] *"Transitions: Recommendation for Transitioning the Use of Cryptographic Algorithms and Key Lengths"* NIST Special Publication 800-131A written by Elaine Barker and Allen Roginsky from Computer Security Division http://csrc.nist.gov/publications/nistpubs/800-131A/sp800-131A.pdf Last Accessed on 08.12.2011
[7] "*Recommendation for Key Management – Part 1: General (Revised)"* NIST Special Publication 800-57 March, 2007 written by Elaine Barker, William Barker, William Burr, William Polk, and Miles Smid Last Accessed on 08.12. 2011http://csrc.nist.gov/publications/nistpubs/800-57/sp800-57-Part1-revised2_Mar08-2007.pdf
[8] Iana, G.V.; Anghelescu, P.; Serban, G.; , *"RSA encryption algorithm implemented on FPGA,"* Applied Electronics (AE), 2011 International Conference on , vol., no., pp.1-4, 7-8 Sept. 2011 http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6049062&isnumber=6048828
[9] Digi-Key Corporation Website: http://www.digikey.com/ Last Accessed on 08.12.2011
[10] Menezes, A.J.; van Oorschot, P. C.; Vanstone, S. A.; , *"Handbook of Applied Cryptography"* 5th Edition, Chapter 8 - Public-Key Encryption pg: 290. All chapters can be reached via http://www.cacr.math.uwaterloo.ca/hac/. Last Accessed on 08.12.2011
[11] ASCII Table and Description Last Accessed on 09.12.2011 http://www.asciitable.com/
[12] http://www.all-battery.com/SANYOLi-ionPrismatic3.7V2000mAhRechargeableBatterywithFuse-30043.aspx
[13]http://www.st.com/internet/com/TECHNICAL_RESOURCES/TECHNICAL_LITERATURE/DATASHEET/CD00000546.pdf
[14] "Platform Versions." *Android Developers*. Web. 08 Dec. 2011. <http://developer.android.com/resources/dashboard/platform-versions.html>.