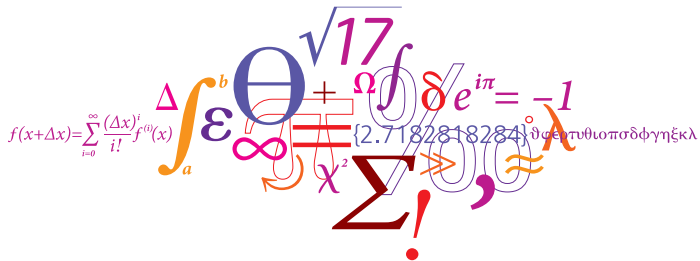
Project - Smart Lock System

Embedded C/C++ Smart Applications - Course 62547

Jonas Stenholt Melchior Jensen - s205129

22/05 - 2023



DTU Diplom
Center for Diplomingeniøruddannelse

Outline

- What is a smart lock ?
- Design and implementation in this project
- Key Functionality on PIC24FJ128GA202

A rough sketch I

Figure: Smart Lock System

RF comm. could be many things

- NFC

- BLE

- WiFi

System architecture I

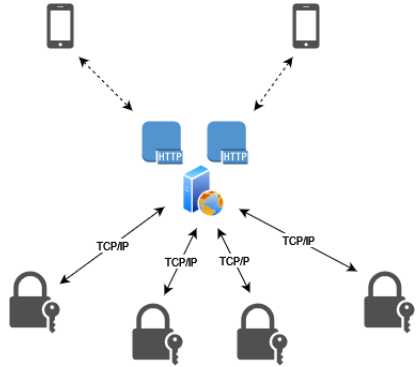


Figure: System sketch

Lock activation flow I

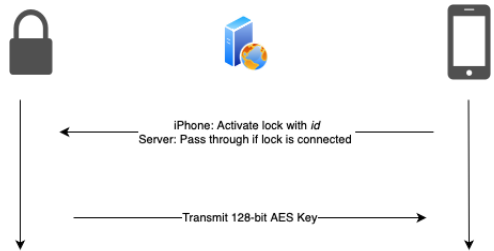


Figure: Activation flow

Unlocking flow I

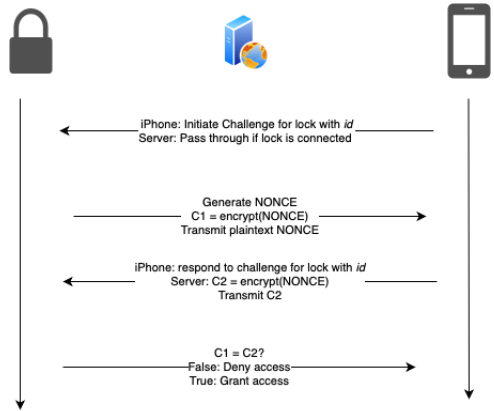


Figure: Unlocking flow

Hardware cryptographic engine I

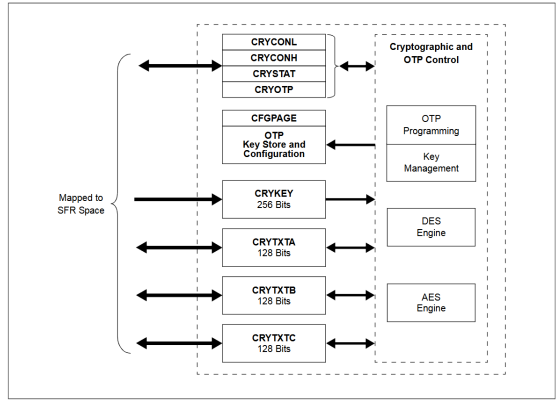


Figure: Crypto engine registers PIC24 family

Initialization I

```
1  void initCryptoEngine_AES_ECB() {
2      CRYCONLbits.CRYON = 1; // Enable crypto engine
3      CRYCONHbits.KEYSRC = 0b0000; //Select the key source (CRYKEY)
4      CRYCONLbits.CPHRSEL = 0b1; // AES Engine
5      CRYCONLbits.CPHRMOD = 0b000; // ECB Mode
6      CRYCONHbits.KEYMOD = 0b00; //Set the key strength (128-bit key)
7  }
```

Encrypting I

```
1  void encryptECB(unsigned char plaintext[16]) {
2      memcpy((void*)&CRYKEY0, ECB_key, 16); //Load the key into CRYKEY
3      CRYCONLbits.OPMOD = 0b0000; // Encryption Mode
4
5      if (CRYSTATbits.KEYFAIL == 1) {
6          printf("\r\nWrong encryption config..\r\n");
7      }
8      else {
9          printf("\r\nEncrytpion mode configured\r\n");
10     }
11
12     //Load the plaintext block into CRYTXTA (16 bytes for AES)
13     memcpy( (void*)&CRYTXTA0, plaintext, 16);
14
```

Encrypting II

```
15      CRYCONLbits.CRYGO = 1;
16
17      // Wait for encryption to be finished
18      while(CRYCONLbits.CRYGO == 1){}
19
20      printf("Cipher Text:\r\n");
21      printCRYTXTB();
22  }
```

Decrypting I

```
1  void decryptECB(unsigned char ciphertext[16]) {
2      memcpy((void*)&CRYKEY0, ECB_key, 16); //Load the key into CRYKEY
3
4      CRYCONLbits.OPMOD = 0b0010; // Key Expansion Mode
5      CRYCONHbits.KEYSRC = 0b0000; //Select the key source (CRYKEY)
6      CRYCONLbits.CRYGO = 1;
7
8      // Wait for encryption to be finished
9      while(CRYCONLbits.CRYGO != 0);
10
11     //unsigned char ciphertext[16];
12
13     CRYCONLbits.OPMOD = 0b0001; // Decryption Mode
14     // 0010 = key expansion
```

Decrypting II

```
15
16      if (CRYSTATbits.KEYFAIL == 1) {
17          printf("\r\nWrong decryption config..\r\n");
18      }
19      else {
20          printf("\r\Decryption mode configured\r\n");
21      }
22
23      memcpy((void*)&CRYTXTA0, ciphertext, 16);
24
25      CRYCONLbits.CRYGO = 1;
26
27      // Wait for encryption to be finished
28      while(CRYCONLbits.CRYGO == 1);
29
```

Decrypting III

```
30      printf("Plain Text:\r\n");
31      printCRYTXTB();
32  }
```