KEYPRO

專題學生: 00857035戴芷柔
00857047邱莘瑜
指導教授: 嚴茂旭 教授

動機與摘要

由於資訊科技愈發進步，資訊傳送的安全也愈發重要。本專題目標是透過DE0板子對資料進行加解密。我們以Verilog撰寫程式，配合RS-232 通訊裝置從電腦端取得使用者欲加解密之資料，並比對使用者輸入的金鑰與內建金鑰是否相符，若相符則以AES技術進行加解密，資料在FPGA加解密處理後以RS-232 通訊裝置回傳資訊至電腦端。

封包格式

- **RX接收格式**

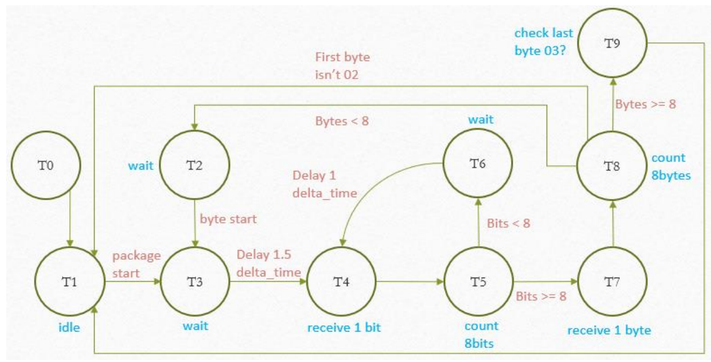
一次接收8組bytes，以下為每組bits的功能
第1組：輸入02以表示開始接收資料。
第2組：1~7個bits控制資料ram位置，第8個bit為1則為寫入，0則為輸出。
第3~6組：需要存放的資料。
第7組：check bit(未使用)。
第8組：輸入03已表示資料接收完畢。

- **TX傳送格式**

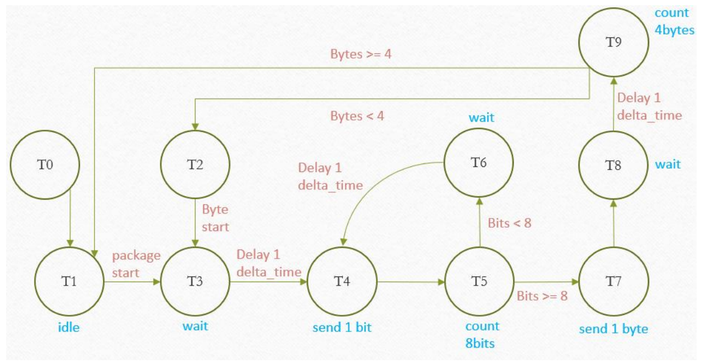
一次傳送4組bytes
上述RX存取的第3~6組bytes(4bytes)會存入32bits的ram，判斷address後將對應ram位置的32bits全部讀出。

資料輸入/輸出

以fsm控制判斷資料輸入/輸出是否符合接收格式開頭02/結尾03，以及實作watchdog判斷是否超時。



- 輸入FSM



- 輸出FSM

Aes實作加密資料規格

加密資料寬度為128，每筆RAM資料為4組bytes，但每組byte的第1個bit不使用，因此每筆RAM的可用資料為28bits，因此每筆AES資料須使用5筆RAM，第5筆僅使用16bits。

e128_in	f355891224466...	f355891224466cd9b34489122224891
ram[109]	11111111	11111111
ram[110]	22222222	22222222
ram[111]	33333333	33333333
ram[112]	44444444	44444444
ram[113]	88776655	88776655

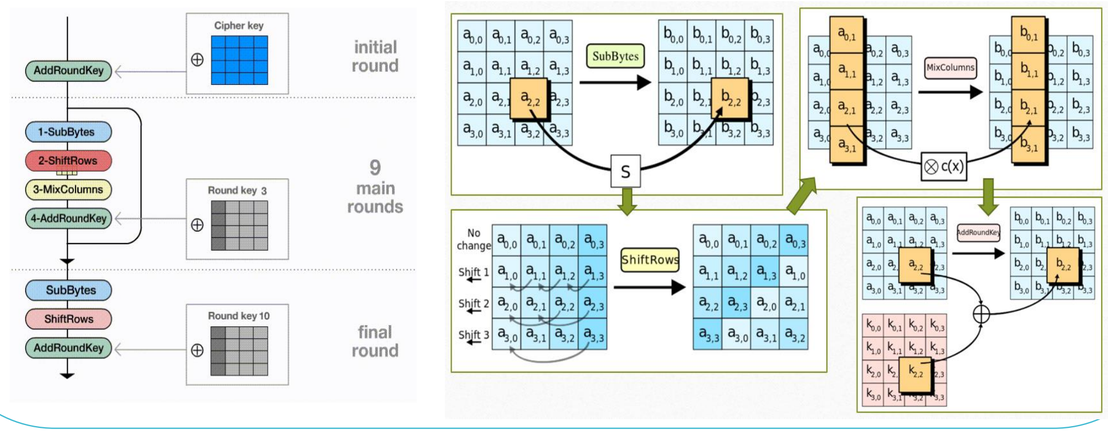
RAM存放格式

使用128個address紀錄:
[4]-[8] : 使用者輸入資料(5筆RAM)
[9]-[10] : 執行的加解密種類
[13]-[16] : 使用者輸入資料(4筆RAM)
[24]-[33] : 使用者輸入金鑰
[60] : AES解密後資料種類
[64]-[68] : AES解密後資料(5筆RAM)
[73]-[76] : AES解密後資料(4筆RAM)

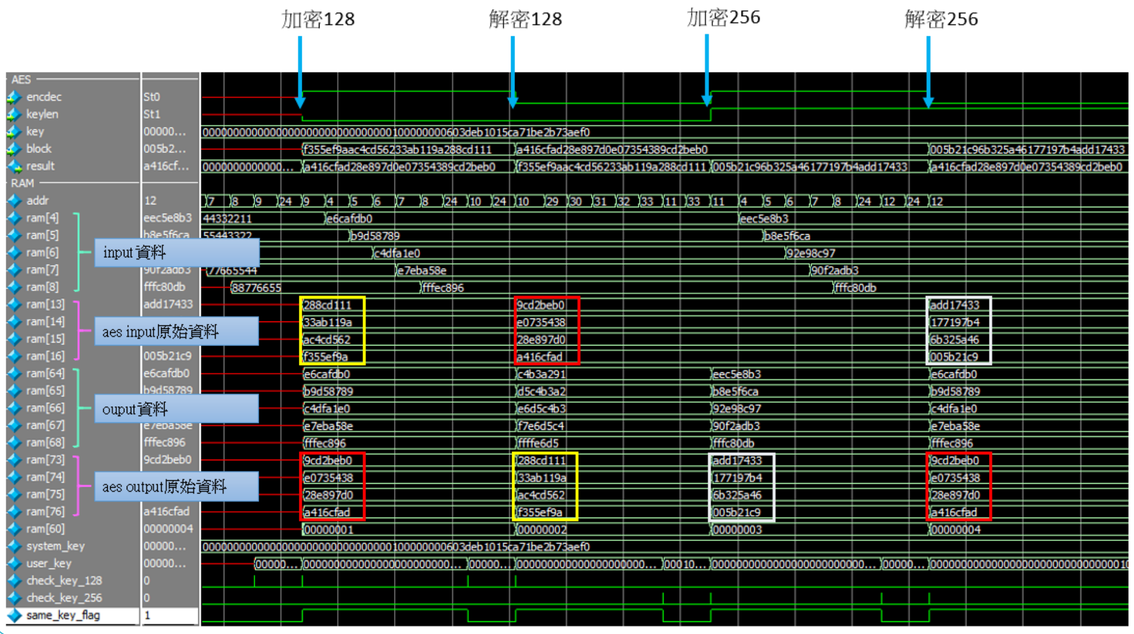
```
//////WRITE AND READ//////////          //////////////READ ONLY////////////
//*[4]-[8]    INPUT DATA(5)      *//    //*[60]       AES OUTPUT TYPE    *//
//*[9]        DO 128 ENCRYPT     *//    //*[64]-[68]  AES OUTPUT(5)      *//
//*[10]       DO 128 DECRYPT     *//    //*[73]-[76]  AES OUTPUT(4)      *//
//*[11]       DO 256 ENCRYPT     *//    ////////////////////////////////////
//*[12]       DO 256 DECRYPT     *//    ////////AES TYPE//////
//*[13]-[16]  INPUT DATA(4)      *//    //* 1:128 ENCRYPT    *//
//*[24-33]    KEY                *//    //* 2:128 DECRYPT    *//
////////////////////////////////////    //* 3:256 ENCRYPT    *//
                                        //* 4:256 DECRYPT    *//
                                        ////////////////////////
```

Aes原理

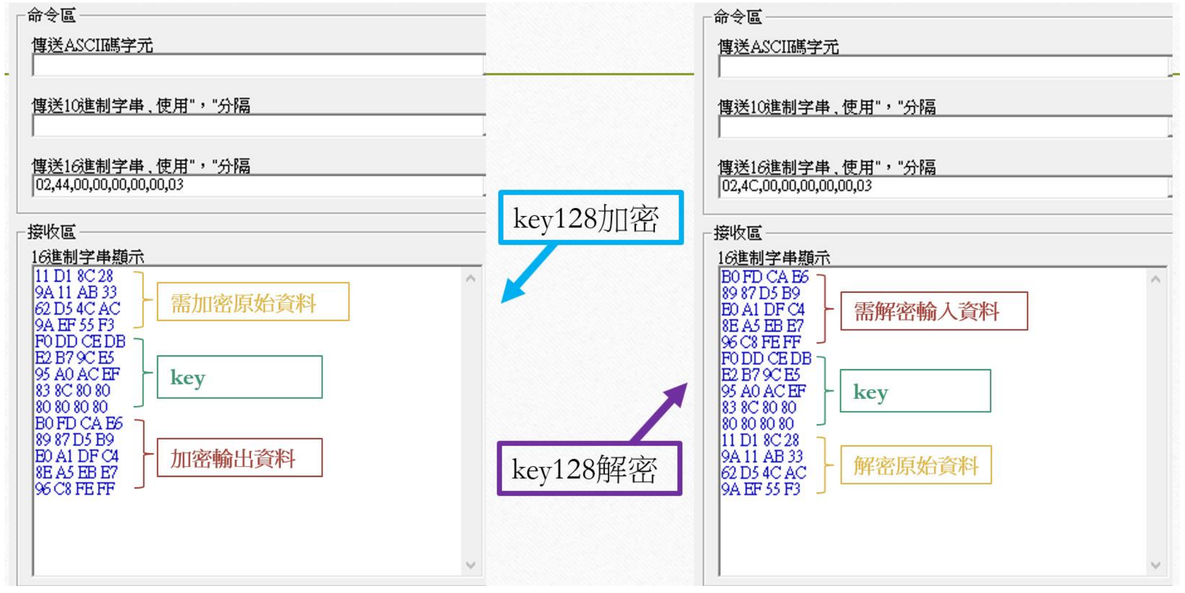
AES 部分主要參照理論以位元組代換(SubByte) 、行移位(ShiftRow) 、列混合(MixColumn) 、輪金鑰加(AddRoundKey) 實現



模擬與實作成果



使用RS232測試軟體實作，令使用者輸入key及data後，若與程式內建key比對符合，可對data進行加解密。



結論

本專題最終可令使用者自行輸入128或256 bits的key並比較是否與內建的key相同，以此達到實作簡易的KEYPRO功能，但還有許多地方值得改進。一方面是預設內建key啟用過一次後如何產生下一組隨機的內建key ，以及更人性化一點的構想 : 如何實作可令使用者於加密時自行設定內建key；另一方面則是本專題如何與生活中的應用結合，這些問題都值得再去探討。