2024 Digital IC Design Homework 5

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NAME | 黃偉峰 | | | |
| Student ID | E34106010 | | | |
| Score = area\*timing (ps) | 24571128000 | | | |
| Cycle time (ns) | 10 ns | | | |
| **Simulation Result** | | | | |
| Functional simulation | | Completed | Gate-level simulation | Completed |
|  | | |  | |
| **Description of your design** | | | | |
| 實做部分較複雜我直接寫在下方 | | | | |

The scoring standard: (The smaller, the better)

*Scoring =*

*Area cost \* Timing cost*

*Area cost =*

*Total logic elements + total memory bits + 9\*embedded multiplier 9-bit elements*

*Timing cost =*

*Simulation time*

**實作介紹在下一頁**

1. AES Inner and Outer Pipeline Implementation

在實現高效的AES加密過程中，此次Project我使用Inner/Outer pipeline作為設計。Inner Pipeline分割了每一輪中的SubByte和其他操作，並在其之間加入Pipeline register，而Outer Pipeline則在各個輪次之間添加Pipeline register，以進一步提高處理效率。Outer Pipeline的部分參考自[1]，並在下面圖片描述了這兩部分的實現方法。

A diagram of a computer flowchart

Description automatically generated

Figure1. Outer Pipeline implementation

A diagram of a flowchart

Description automatically generated

Figure1. Inner Pipeline implementation

在AES當中**SubByte通常具有最長的Delay時間**，因此在每輪的內部設計時，透過將Register加在SubByte之後能夠有效減短Critical Path，並在每輪結束時都加入Register達成Outer Pipeline能夠更進一步加快電路速度並且設計上也更模組化更簡單。

1. AES Inner Module Modify

在AES加密算法中，AddRoundKey、SubByte、ShiftRow、MixColumn和KeyExpansion是其主要的模組。其中，SubByte和KeyExpansion都使用到了S-Box部分。一般實現中，S-Box通常使用Lookup Table的方式，但這在硬體電路中會耗費大量的MUX。因此，在此次專案中，我選擇了基於S-Box底層數學計算的方法來實現。而MixColumn部分涉及GF(28)的乘法，因為乘的皆為固定數字，所以我找出常數矩陣將GF(28)的乘法轉換成單純的XOR邏輯運算。

* + 1. MixColumn

A number of numbers on a white background

Description automatically generated

在MixColumn中我們會在GF(28)下做矩陣乘法，並且可以看到乘的數值固定為1、2、3，故我們只要找出這三個常數所代表的矩陣就可以將乘法變成XOR運算，以下是一個找出常數矩陣3的例子:

A math equations on a white background

Description automatically generated

我們只要將x的各個次方的各項係數表示成如下圖的矩陣即可，並且依此類推可繼續找到1和2的常數矩陣。

A close-up of a number

Description automatically generated

最後將矩陣根據乘上2、3、1、1的順序將常數矩陣加起來即可。

* + 1. S-BOX

在S-Box的實現中，有兩種主要的方法。一種是簡單的方法，即使用MUX構建LUT；另一種則是計算輸入值的multiplicative inverse並進行同構映射isomorphic mapping。在此次Project中，我選擇了後者來實現S-Box。計算乘法反元素是一個相對複雜的操作，為此我參考了[2]和[3]的方法。在此過程中，我對輸入值進行了特定的乘法和指數偏移映射（multiplicative 和 exponential offsets）。這種映射可以降低計算矩陣的Hamming weight。較低的Hamming weight意味著需要更少的XOR操作，從而使得S-Box所需的邏輯閘數量更少，節省了硬體面積。此外，降低了maximum Hamming weight意味著縮短critical path的長度，這使得電路的運算速度更快。總結來說，這種映射技術不僅減少了所需的邏輯閘數量和佔用的面積，還提高了運算速度，使得S-Box在高效能和資源受限的應用場景中表現出色。

* **Linear Mapping:**此改善參考自[2]，在S-BOX當中涵蓋了

Multiplicative inverse、Affine transformation、Isomorphic mapping，而傳統上Affine transformation的矩陣有著較高的Hamming weight，在矩陣中的每個1就代表電路中的一個XOR，故該方法透過multiplicative offsets/exponential offsets將轉換所需要用到的矩陣能有較低的Hamming weight，並且也有較小的maximum hamming weight。A diagram of a mathematical equation

Description automatically generated

根據作者所提供的較優解，我選擇使用Exponential offsets = 1，Multiplicative offsets = 79。

* **Multiplicative inverse**:該算法參考自[3]，當中有清楚寫出流程及相關算式可以直接照著刻電路即可。

A diagram of a stage

Description automatically generated

上圖為GF(28)inversion的結構圖，分為三個主要階段，每個階段的具體實現方法如下：

* + - * 1. Stage 1:計算輸入的16次方和17次方。Normal Basis來簡化這些計算。該階段的輸入為GF(28)中的元素，並將其轉換到GF((24)2)域中進行操作。 圖中H和L分別高位和低位，通過NBtoRRB轉換成Redundantly Represented Basis表示。
        2. Stage2:進行GF(24)的inversion計算。使用Polynomial Ring Representation表示進行GF(24) inversion，利用該表示法可以提高效率。
        3. Stage3:執行GF(24)的乘法操作。 使用RRB進行GF(24)乘法，因為RRB在GF(24)乘法中具有較高的計算效率。

1. Reference

[1]“Pipelined implementation of AES encryption based on FPGA,” IEEE Conference Publication | IEEE Xplore, Dec. 01, 2010. Available: https://ieeexplore.ieee.org/document/5688757/citations#citations

[2]“AES S-Box hardware with efficiency improvement based on linear mapping optimization,” Tohoku University. Available: https://tohoku.elsevierpure.com/en/publications/aes-s-box-hardware-with-efficiency-improvement-based-on-linear-ma/fingerprints/ R. Ueno, N. Homma, Y. Sugawara, Y. Nogami, and T. Aoki,

[3]“Highly efficient GF(28) inversion circuit based on redundant GF arithmetic and its application to AES design,” in Lecture notes in computer science, 2015, pp. 63–80. doi: 10.1007/978-3-662-48324-4\_4. Available: https://www.researchgate.net/publication/285601676\_Highly\_Efficient\_GF28\_Inversion\_Circuit\_Based\_on\_Redundant\_GF\_Arithmetic\_and\_Its\_Application\_to\_AES\_Design