

第四章 影像感應晶片相關硬體單元設計

第一節 初始化影像感應晶片單元

初始化影像感應晶片單元(ISU)任務為初始化影像感應晶片。本研究採用的 CMOS 影像感應晶片，為原相公司 PAS109，圖 4-1 為其硬體方塊圖。由於 PAS109 採用 I^2C (Inter-Integrated Circuit)匯流排作為參數傳遞的介面。因此 ISU 必須內嵌 I^2C 的傳輸協定標準，以便設定參數至 PAS109，所以以下先對 I^2C 傳輸協定進行初步的介紹，而後再對硬體實現架構說明。

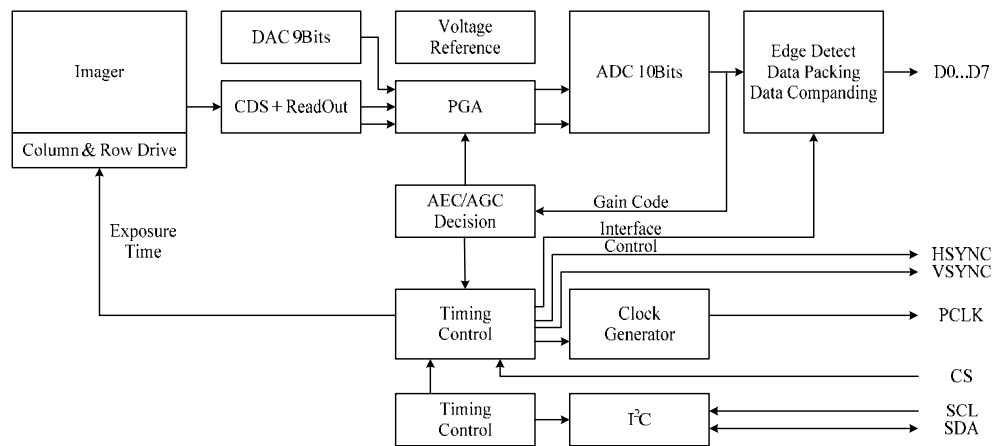


圖 4-1. CMOS 影像感應晶片 PAS109 內部方塊圖

壹、 I^2C 傳輸協定簡介

I^2C 傳輸協定[20][21]最早是由飛利浦於 1980 年所發展，其技術為繼承 C Bus 而來，由於其功能優越，硬體接線容易，因此在國內外已成為工業界的標準，其特色為使用兩序列雙向訊號線 SCL (Serial Clock)與 SDA (Serial Data)，即可進行資料傳送，並且在 I^2C 匯流排可同時掛載多個裝置，如圖 4-2 所示，使用時兩訊號線必須加入提升電阻約 $4.7K\Omega \sim 10K\Omega$ 至電源。

若就其傳送速度，以 1980 年所發表 I²C 標準，傳送速度可達 100 KBits/s，如以 1993 年所改版的 I²C 快速模式傳送可達 400KBits/s。雖其速度雖然無法媲美 USB 或 IEEE1394 等序列介面，但由於其協定較容易被實現，用於小型化、不需高速的參數傳遞，已經相當夠用。目前市面上，可以購買到的許多電子產品都有 I²C 規格的产品，例如：文字型 LCD、類比數位轉換器、電子式可抹寫記憶體...等產品。

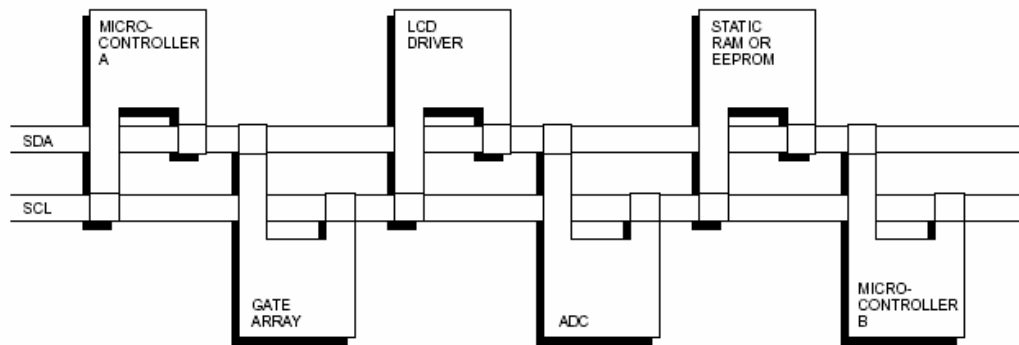


圖 4-2. I²C 匯流排掛載多個裝置

在 I²C 匯流排上，每個裝置都有主(Master)、僕(Slaver)之分，I²C 也允許兩個以上 Master 同時對 I²C 匯流排來控制，但其方式在本研究中並不探討，詳細內容可參考文獻飛利浦公司技術文件[20][21]，而本硬體所採用的 PAS109 屬於 Slaver 的角色，因此 ISU 的角色訂定位為 I²C Master 控制器。

貳、I²C 傳輸協定規則

由於 I²C 的傳輸規則相當複雜，本研究僅針對 I²C 欄位意義、寫入封包模式說明。

一、I²C 欄位意義

(一) 起始欄位(*Start*)

起始欄位將會加在I²C封包的最前端，用以表示一次新的I²C傳輸即將開始。

(二) 結束欄位(*Stop*)

結束欄位將會加在I²C封包的最後端，用以表示一次I²C傳輸正式結束。

(三) 回應欄位(*Acknowledge, Ack*)

每次傳送資料完畢，資料接收端為告知資料傳送端，接收到資料，便會由資料接收端發出回應欄位，以確認資料傳送完成。

(四) 裝置位址(*Device Address, DevAddr*)

如在圖4-2所談，I²C匯流排可以掛載一個以上的裝置，因此每個裝置皆共享SDA訊號線，為在傳送時能夠分辨封包要送到哪一個裝置，必須給予每個裝置一個位址，此位址稱為Device Address，簡稱DevAddr，DevAddr資料寬度為7位元，可定址128個位址。

(五) 子位址(*Sub Address, SubAddr*)

為使每個裝置能更彈性的運作有限的空間，因此每個裝置再加掛一子位址，此位址稱為Sub Address，簡稱SubAddr，SubAddr寬度有8位元，可定址256個位址。如以簡單一點的方式舉例，DevAddr就像是一棟公寓的門牌號碼，而公寓裡每戶人家的編號就是SubAddr，這樣的作法可讓有限的空間容納更多變數。

(六) 傳輸資料(*Data*)

I²C所要傳輸資料。

(七) 回應(*Ack*)

每次傳送完DevAddr、SubAddr或Data，必須要接收一個位元的資料，此單位元資料稱為Ack，可使資料傳送端確認傳送的DevAddr、

SubAddr或Data，正確被接收。若Ack為低準位表示正確接收，反之高準位為接收失敗。

二、I²C 寫入封包模式

(一) Byte Write模式

Byte Write模式是I²C最簡單的寫入模式，其封包主要包含DevAddr、SubAddr與要寫入的資料，此資料將會填入DevAddr裝置內部的SubAddr空間，此空間以Data(SubAddr)表示。封包詳細格式如圖4-3所示。

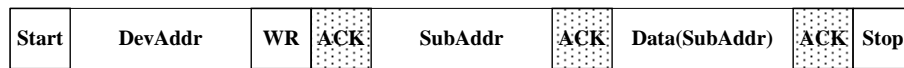


圖 4-3. Byte Write 封包

(二) Page Write模式

Byte Write為寫入單筆資料所用，若要寫入連續資料，則需用Page Write模式。Start開始後，Master便一直送資料至Slaver，此時Slaver除接收資料外，也會判斷Stop是否出現，若Stop出現Slaver便不再被寫入，傳輸亦宣告終止，傳輸情形如圖4-4所示。

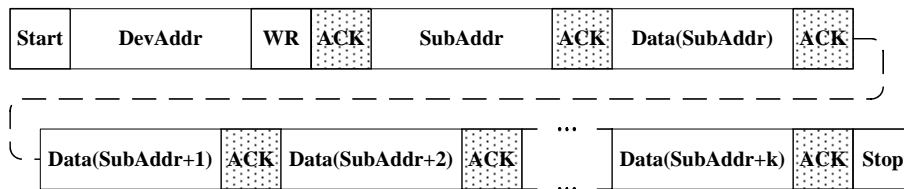


圖 4-4. Page Write 封包

參、ISU 硬體架構設計

ISU 硬體架構可分為控制單元(Control Unit)與資料路徑(Data Path)兩部分，由於 ISU 必須掌管 I²C 的複雜動作，為有系統地管理 ISU，控制單元將使用微指令設計方式，ISU 所使用到的微指令，將會存在參數管理單元(MAU)的 ISU_ARG 記憶體供 ISU 的 Control Unit 提取，微指令解碼完畢再送至 Data Path，將資料傳送至 I²C 匯流排，為詳細說明將先對微指令的格式進行說明。

一、訂定 ISU 微指令格式

ISU 微指令格式，如下圖所示：

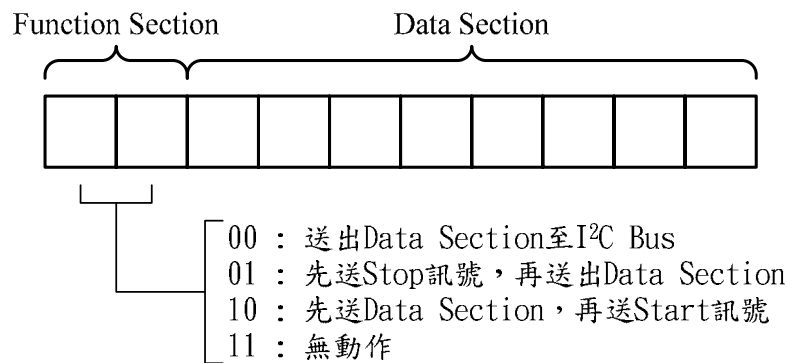


圖 4-5. ISU 微指令格式

每道微指令均含 10Bits，前端的 2Bits 稱為，解釋目前命令將進行的動作，後端的 8Bits 則稱為 Data Section，用來放置將傳送的資料，如 DevAddr、SubAddr 或 Data。由 Byte Write 與 Page Write 波形可知若送出 Data Section，則必須接收 ACK，此規則可由 Function Section 解碼可知，因此 Function Section 不特別設立送出 ACK 的指令。

二、如何以微指令來表示 I²C 的 Byte Write 與 Page Write 兩種模式

(一) Byte Write 之微指令撰寫格式

每道微指令可附帶傳送一筆 DevAddr、SubAddr 或 Data，因此可由 Byte Write 要傳送的資料，將其劃分每個指令所負責的區塊，如下所示。

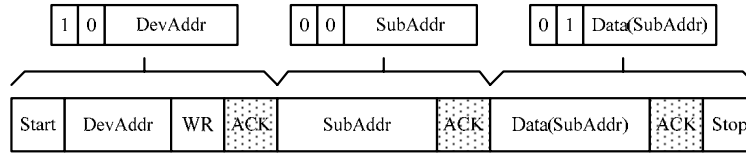


圖 4-6. Byte Write 之微指令與 I²C 輸出對應關係

由上圖可知 Byte Write，必須由三道微指令構成，第一道指令負責送出 Start→DevAddr，第二道送出 SubAddr，第三道送出 Data→Stop。

(二) Page Write 之微指令撰寫格式

觀察 Page Write 與 Byte Write 的傳送波形，可發現差別只在於傳送資料的連續性，所以可猜想得知 Page Write 微指令與 Byte Write 微指令類似，只是 Page Write 必須花更多的微指令來表示，其實際的 Page Write 波形與微指令的對照如下圖所示。

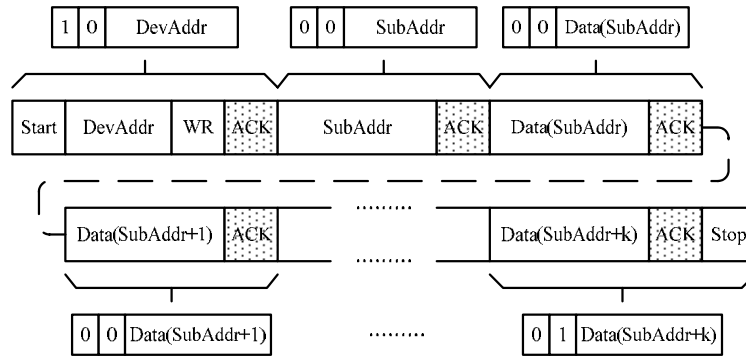


圖 4-7. Page Write 之微指令與 I²C 輸出對應關係

三、硬體內部方塊設計

將上述對 ISU 的功能需求實體化後，提出 ISU 內部的方塊如下：

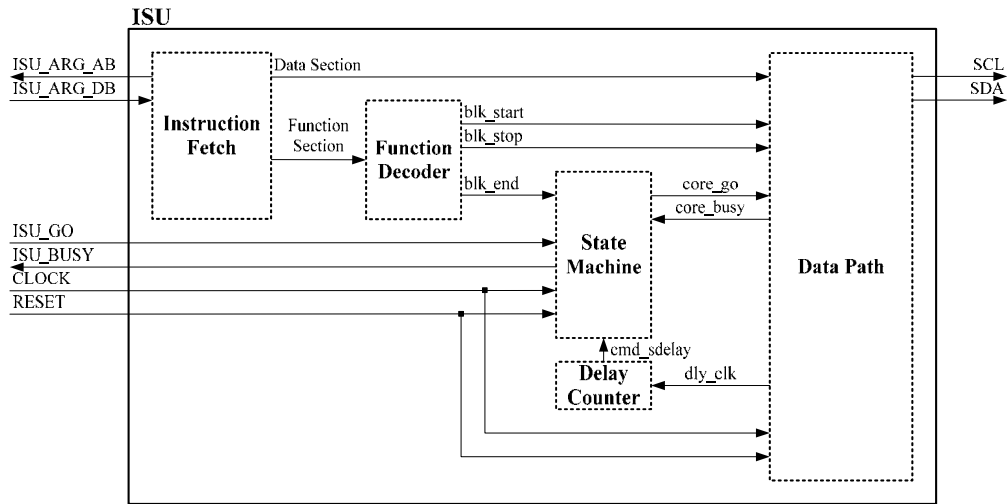


圖 4-8. ISU 硬體內部方塊

ISU 的控制單元分別由 Instruction Fetch、Function Decoder、State Machine 與 Delay Counter 所組成。Instruction Fetch 負責擷取存放於 MAU 的 ISU_ARG 微指令。微指令的 Data Section 直接送至 Data Path，而 Function Section 經解碼產生 blk_start、blk_stop 與 blk_end。blk_start 表示一段微指令的起頭，此時 I²C 應該要產生 Start 訊號，blk_stop 表示一段微指令的結尾，此時 I²C 應該要產生 Stop 訊號。

blk_end 用來表示目前正在處理的微指令，是否為最後一道微指令。在 Page Write 模式中，常需要寫幾筆連續位址的資料，在內部每筆資料的交界時，blk_end 將解碼出低準位，若處於 Page Write 正寫入最後一筆資料，則 blk_end 將解碼出高準位。舉例說明，假設有一 Page Write 需 7 道微指令組成，則寫入第 1 道至第 6 道時，blk_end 皆為低準位，寫第 7 道指令時，blk_end 將轉為高準位。在 Byte Write 模式中，由於固定為三道微指令，因此執行至 Byte Write 第三道微指令時，blk_end 一定解碼出高

準位。

當 blk_end 為高準位時，表示目前正在寫入的模式即將在下一週期結束，根據 I²C 的協定，此時必須延遲一段較長的時間，再接續下一段 Page Write 或 Byte Write 工作。若 blk_end 為低準位，則表示在 Page Write 有連續的指令要執行，此時可不需延遲即可進行寫入工作，根據上述動作需求，將控制單元的 State Machine 設計如下圖所示。

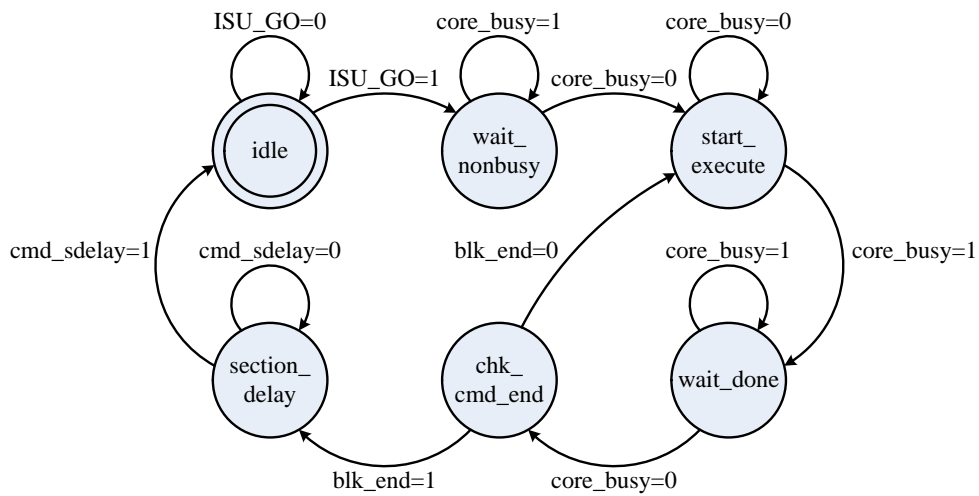


圖 4-9. ISU 控制單元之 State Machine

當系統發出 ISU_GO 時，表示要開始執行初始化影像感應晶片的工作，idle 立即跳至 wait_nonbusy 檢查 Data Path 為非忙碌狀態，僅接著 start_execute 送出 core_go 高準位，通知 Data Path 執行寫入工作，若 core_busy 也轉為高準位，表示 Data Path 接受微指令正在執行，此時進入 wait_done 等待微指令執行完畢，若指令執行完畢，進入 chk_cmd_end 依照 blk_end 訊號，判斷要跳至 start_execute 繼續執行連續的微指令，或跳至 section_delay 進行一段長時間的延遲再回到 idle 狀態。

肆、訂定微指令記憶體內容

接下來根據 PAS109 資料手冊，歸納需要設定的參數，並將之轉換成 Byte Write 或 Page Write 的微指令，並填至 MAU 的 ISU_ARG，ISU_ARG 最多可存放 16 筆微指令。

以本研究為例，必須要設定 PAS109 的 SubAddr 13H~14H,16H 的暫存器內容為 48H,69H,76H，且根據 PAS109 資料手冊 DevAddr 必須訂為 80H，根據這樣的需求，將要設定至記憶體的內容以下表呈現：

表 4-1. MAU 之 ISU_ARG 記憶體內容

Memory Address	Function Section	Data Section	I ² C Mode	Content Description	PAS109 Name
00H	10B	80H	Page Write	Start and DevAddr	
01H	00B	13H	Page Write	SubAddr	
02H	00B	48H	Page Write	Data(13H)	np
03H	01B	69H	Page Write	Data(14H) and Stop	lpf
04H	10B	80H	Byte Write	Start and DevAddr	
05H	00B	16H	Byte Write	SubAddr	
06H	01B	76H	Byte Write	Data(16H) and Stop	pga
Others	11B	FFH	None	None	

第二節 影像擷取單元

影像擷取單元(ICU)負責與前端的 CMOS 影像感應晶片接洽以取得影像。為說明硬體設計方式，以下將先說明 PAS109 如何輸出影像資料，接著針對其輸出圖像的方式，簡述硬體之規劃方塊。

壹、PAS109 影像資料輸出規則

PAS109 傳送影像以像素為單位，自 PAS109 資料匯流排以序列的方式送出。圖 4-10 為 PAS109 將要送出的影像空間，令其寬為 PW，高為 PH。

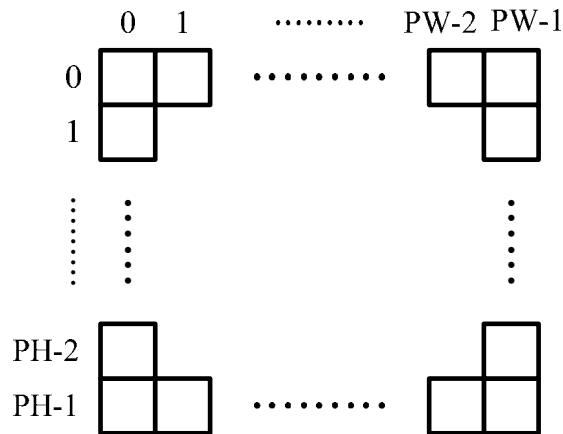


圖 4-10. PAS109 影像感測器影像座標定義

PAS109 將由左而右、上而下依次序輸出像素，如圖 4-11 所示。為在擷取像素時，能區分像素在實際影像的位置，必須配合 *VSYNC*(*Vertical Synchronization*, 垂直同步)、*HSYNC*(*Horizontal Synchronization*, 水平同步) 與 *PCLK*(*Pixel Clock*, 像素時脈)三訊號。

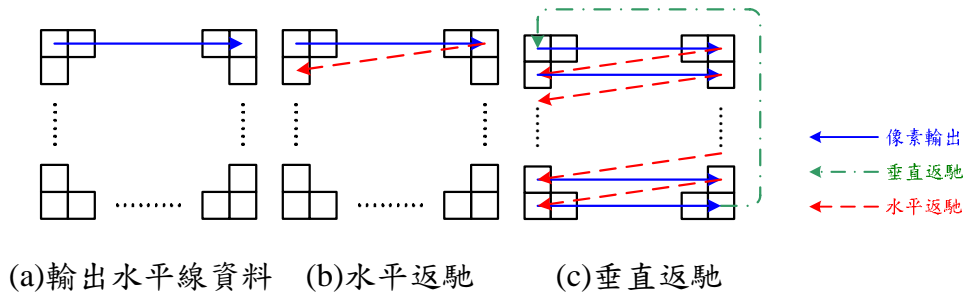


圖 4-11. PAS109 輸出影像順序

圖 4-11(a)送出一列水平影像後，接著進行圖 4-11(b)水平返馳，此時 HSYNC 將被設定為高準位，其他時間皆為低準位。類似地動作，當進行 PH 次水平返馳後，將會進行圖 4-11(c)的垂直返馳，此時 VSYNC 及 HSYNC 皆被設定為高準位。

HSYNC、VSYNC 更直接的物理意義，VSYNC 表示影像的交界，表示上一張影像結束，下一張影像開始傳送，而 HSYNC 則表示一系列的像素傳送完畢。因此計算 HSYNC、VSYNC 的出現情形，即可知正在傳送的像素，在影像的第幾個水平列，也就是正傳送的像素於影像的垂直位址，以下為方便表示垂直位址以 POSY 表示。綜合上述說明，下圖將 HSYNC、VSYNC 的意義更清楚表達出來，在下圖中，每兩條虛線區間，表示一張影像傳送的時間區隔。

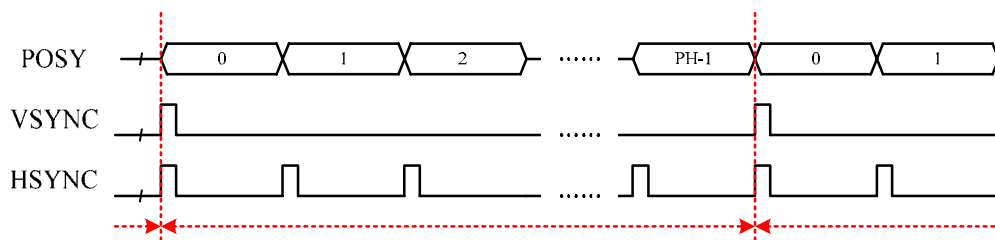


圖 4-12. HSYNC、VSYNC 與像素垂直位址(POSY)的關係

如上說明，由 HSYNC、VSYNC 出現次數，可間接計算出 POSY，若要得知目前送出的像素在第幾垂直列，也就是正傳送的像素於影像的水平位址，以下為方便表示垂直位址以 POSX 表示，可再由計算 PCLK 得出 POSX。當 PCLK 產生一個週期的訊號變化，則表示 PAS109 資料匯流排輸出一筆像素，即 POSX 地增 1。假設由 HSYNC、VSYNC 得知目前位於第 86 水平列，則此時 PCLK 與 POSX 的關係，如下圖所示。在下圖中，每兩條虛線區間，表示一列影像傳送的時間區隔。

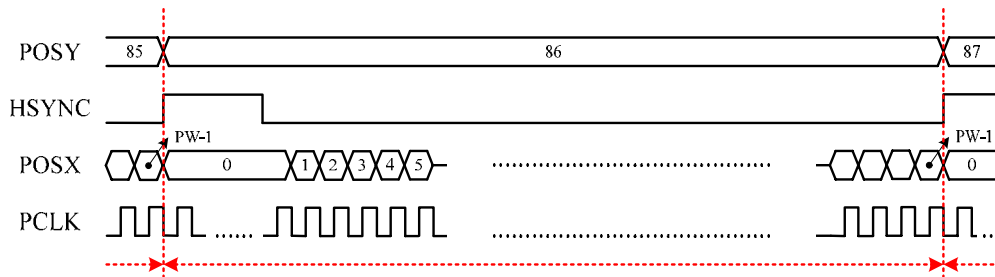


圖 4-13. HSYNC、PCLK 與像素水平位址(POSX)的關係

貳、ICU 硬體規劃

一、位址計算模組(Position Calculate Module, PCM)

綜合上述說明，整理以下規則。由 HSYNC 訊號可得知，目前送完第幾列水平訊號，PCLK 則可得知目前送到第幾行，透過 VSYNC 可得知送完成一張影像。

為方便得知影像資料的對應位置，ICU 內建 PCM 專司於計算像素位址，PCM 將擷取 VSYNC、HSYNC 與 PCLK，最後產出對應之水平座標以 POSX 輸出，垂直座標以 POSY 輸出，PCM 的硬體方塊示意如下。

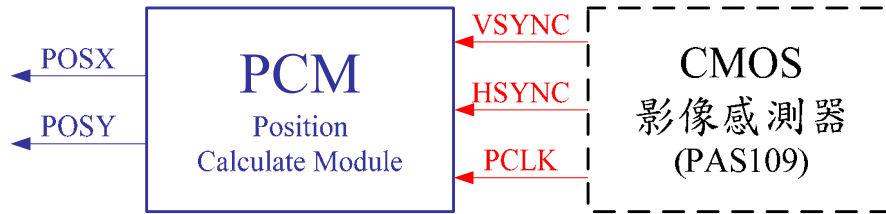


圖 4-14. 位置計算模組(PCM)硬體方塊圖

二、位址比較模組(Compare Position Module, CPM)

許多應用不一定需要完整的影像，也許只需要部份區塊影像，因此 ICU 預留彈性給使用者，ICU 以 CPM 過濾使用者所需要的部分。CPM 包含四個重要的輸入參數，分別為 ICU_BH、ICU_EH、ICU_BV 與 ICU_EV。位置 O(ICU_BH, ICU_BV) 與 O(ICU_EH, ICU_EV) 所囊括的影像區間，即使用者所選擇的影像範圍，如圖 4-15 所示，只有此區域的影像空間才會進行影像濾波等動作。當 CPM 發現該像素落在選取之影像範圍，將送出 Valid 訊號通知外界以接收目前送出來的像素。

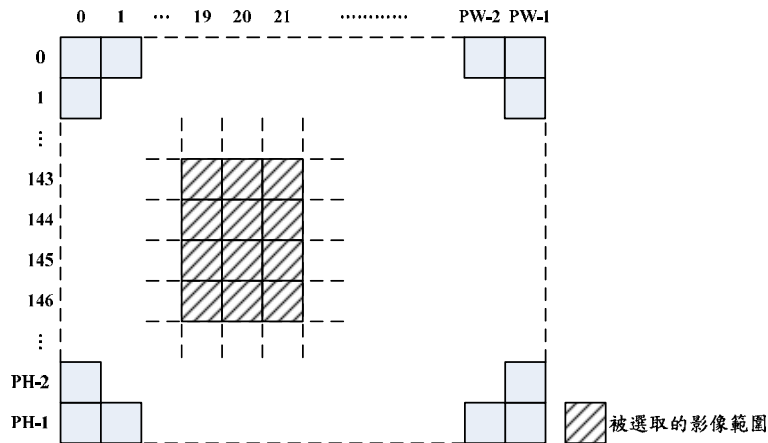


圖 4-15. ICU_BH,ICU_EH,ICU_BV,ICU_EV 為 19,21,143,146 之範圍

三、控制模組(Control Module, CM)

控制模組負責提供 PAS109 的系統時脈(SCLK)與致能線(IS_EN)。SCLK 為利用 FPGA 工作頻率除頻後所提供,IS_EN 低準位將啟動 PAS109 動作,高準位時,則可將 PAS109 重置。

另外控制模組還必須將 CPM 的 Valid 轉換為 PASS 訊號,通知影像處理單元(IPU),把目前 PAS109 的像素取走,像素即 PAS109 送出的 IS_DB 資料匯流排。Valid 轉換至 PASS 的規則如圖 4-16 所示。

(一) 時間點 A：

系統送出 ICU_GO,表示需要 ICU 擷取一張影像,但時間點 A 已在 Valid 高準位的一半左右,若此時開始擷取影像,將取得不完整的有效範圍影像,為取得完整的有效影像,必須等待下一次 Valid 再變為高準位才能擷取。

(二) 時間點 B：

VSYNC 為高準位,表示有另一張新的影像要開始傳送,控制單元正式令 ICU_BUSY 為高準位正式啟動擷取機制,系統所送出的 ICU_GO 發現 ICU 開始動作,因此 ICU_GO 轉為低準位。

(三) 時間點 C~D：

時間點 C~D 表示一張完整有效的影像傳送時間範圍,在此時間範圍 PASS 將被設置為高準位,以通知影像處理單元收取影像。

(四) 時間點 E：

此時一張完整的影像已經結束,且 ICU_GO 也沒有再被設置為高準位表示任務完成,因此 ICU_BUSY 被更改為低準位。

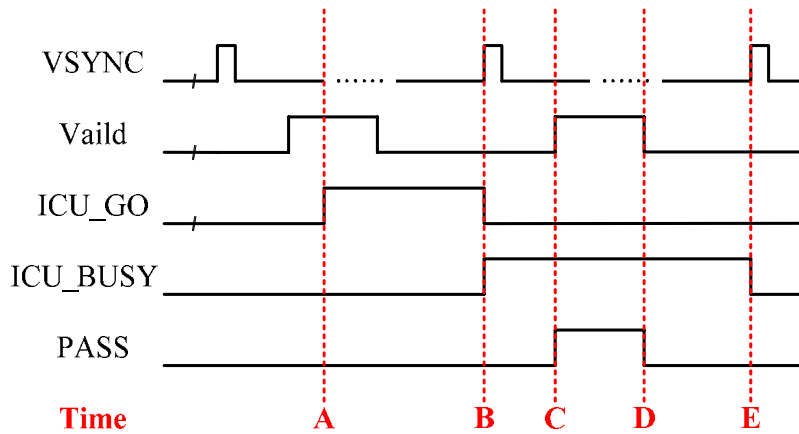


圖 4-16. Valid 轉換至 PASS 之波形圖

上圖轉換的目的，主要為防止擷取到不完整的影像。當 ICU_GO 設為高準位時，此時又逢 Valid 為高準位，則必須要再等待下一次有效的 Valid 來臨才開始擷取影像，較差的情形最多也只需要多等一張影像的傳送時間即可。

將上述硬體方塊實體化，可得細部接線如圖 4-17 所示。

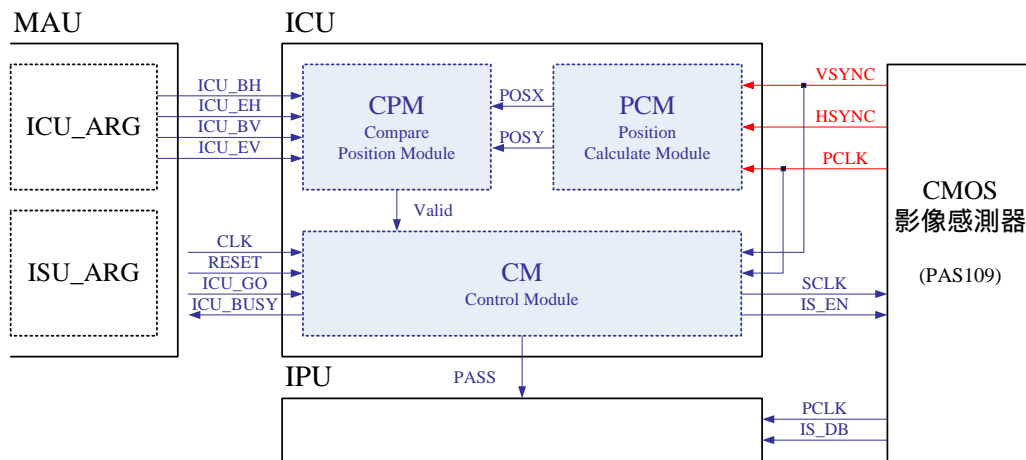


圖 4-17. ICU 硬體方塊圖