

# VATek MDK 說明文件

1. 數位電視概述 .....	3
2. Transport Stream(TS)組成 .....	3
3. VATEK A Series 晶片介紹 .....	4
3.1 A Series 系統架構 .....	4
I2C 協定 .....	6
USB 協定 .....	8
TS PROCESSOR 單元 .....	8
PSI CREATOR 單元 .....	8
MUXER 單元 .....	9
MODULATOR 單元 .....	9
RF MIXER 單元 .....	9
開機流程 .....	9
3.2 系統功能 .....	10
系統功能操作 .....	10
4. MDK .....	13
4.1 前言 .....	13
4.2 MDK 軟體架構 .....	13
4.3 A 系列 MDK 說明 .....	13
4.3.1 A 系列軟體系統架構 .....	13
4.3.2 編譯定義 .....	16
4.3.3 API 使用範例 .....	17
電視廣播轉換初始化 .....	17
鎖定傳輸串流訊號(Tuner 與 DEMOD) .....	18
廣播制式轉換或更換頻率 .....	19
過濾電視廣播節目 .....	20
停止廣播轉換 .....	21
4.3.4 MDK Porting 功能 .....	22
4.3.5 建置 MDK 環境 .....	23
4.3.6 MDK sample code 流程 .....	24
4.4 API 功能說明 .....	25
4.4.1 系統通用 API .....	25
4.4.2 A 系列晶片 API(Transform) .....	25
4.4.3 外部裝置相關 API .....	27
RF .....	27

DEMOD.....	27
TUNER.....	28

# 1. 數位電視概述

數位電視是將原本為類比訊號的影像、聲音等資訊轉換成數位訊號，也就是只有 0 和 1 所組成的二進位形式訊號，並進行壓縮後輸出在電視上，相較傳統類比電視可以播放更大量的節目資訊。

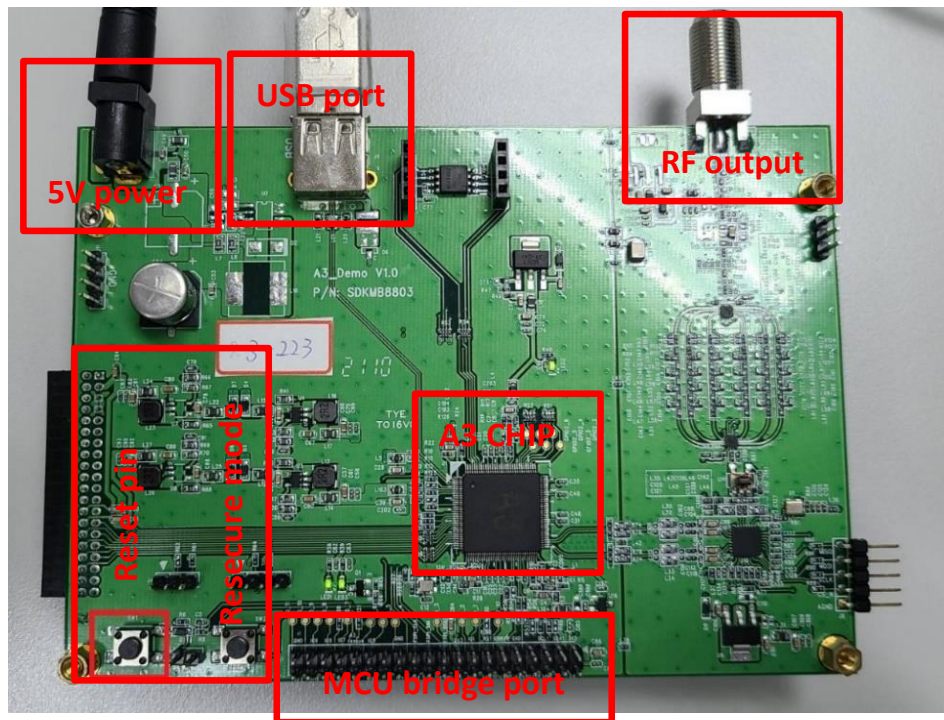
## 2. Transport Stream(TS)組成

TS 為數位電視播放的來源格式，完整的 TS 須包含下列三個元素：

- PES：封裝影音訊號及其他如 CC 字幕等訊號的封包
- PCR：後端輸出至電視時 DEMOD 需要參考的時間，主要決定封包播放順序，其好壞會影響播放品質
- PSI：於各國電視播放時需遵從的規範，不同國家具備不同 PSI TABLE，如美國為 PSIP、日本為 ARIB...等

### 3. VATEK A Series 晶片介紹

#### 3.1 A Series 系統架構



A 系列晶片為數位電視調變晶片，可藉由外部硬體介面輸入 TS 資料流，將接收到的訊號進行再製，提供加入 PSI table 及 PCR 校正的功能，並依照需求轉換成不同調變後輸出 TS 訊號進行廣播，系統架構主要分為三個部分。

系統 IO：

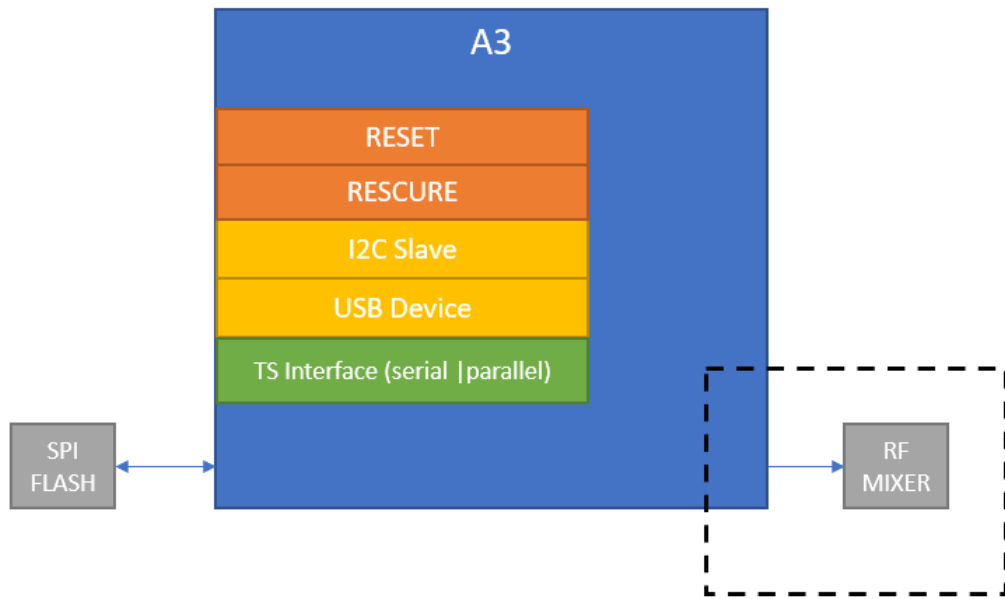
- RESET 訊號 - 系統 RESET 控制訊號，可由外部進行硬體復位 ( 參閱開機流程 )。
- RESCUE 訊號 - 救援功能，當韌體損壞時進入救援模式訊號 ( 參閱開機流程 )。

控制介面：

- 支援 MDK 結合外部 MCU 透過 I2C。
- 使用 SDK 提供的 USB 介面進行晶片功能設定與控制。

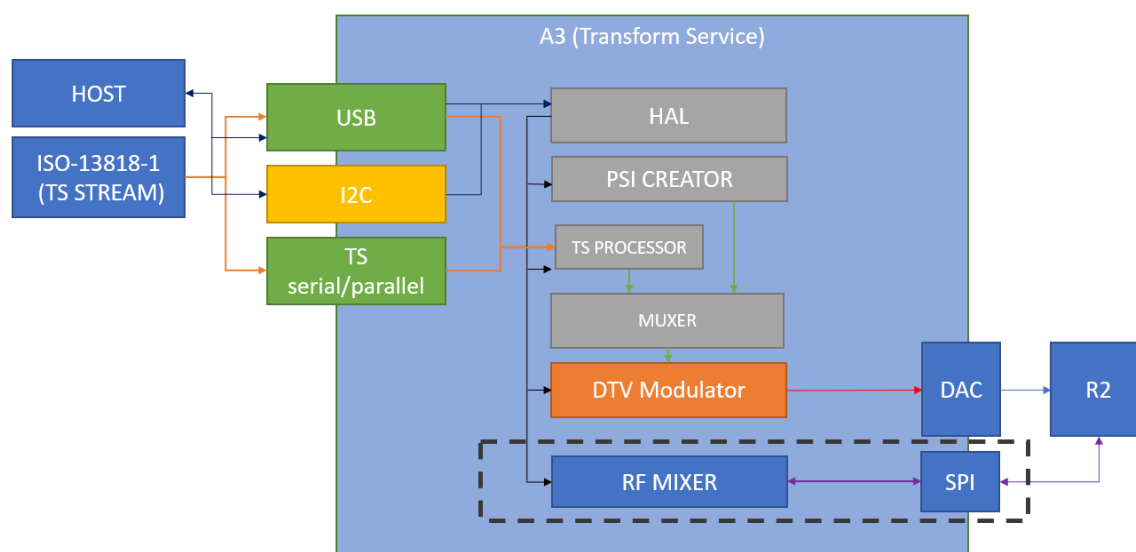
輸入介面：

- TS 介面 - 支援 Serial 與 Parallel 模式
- USB BULK 介面 - USB 除了控制與設定外支援透過 Bulk Endpoint 輸入多媒體流



外部硬體另外包含兩個部分，第一為韌體載入功能，A 系列需依賴韌體(服務)，才可以完整提供相關功能，上電時需由 SPI 介面載入韌體 (服務)，細節可參閱開機流程章節。第二部分為選用功能，如果搭配完整開發方案設計，使用搭配的 RF MIXER 則需透過同一介面連結對應 RF MIXER 以方便韌體 (服務) 提供 RF 相關能力。除外部介面外，晶片內由許多功能單元結合而成，主要可以區分為：

1. HAL：外部控制單元支援 I2C 與 USB 的外部協定實作，用來控制與協調內部功能。
2. PSI CREATOR：數位電視需要之 PSI TABLES 產生單元
3. TS PROCESSOR：具備解析與處理輸入多媒體流功能。
4. MUXER：將 TS PROCESSOR 與 PSI CREATOR 結合 PCR 產生符合調製單元需要的 TS stream。
5. DTV Modulator：數位電視調製器，多格式數位電視調製單元。
6. RF Mixer：控制統一單元，控制預設支援的 RF MIXER (可選)。



## I2C 協定

透過 CHIP 支援的 I2C Slave 介面，支援最高 400 Kb 傳輸速度，可以對 HAL REGISTER 進行寫入與讀取操作。A 系列 CHIP 位址為 7-bits 的 0x10 (可透過 IO 設置為 0x18)。

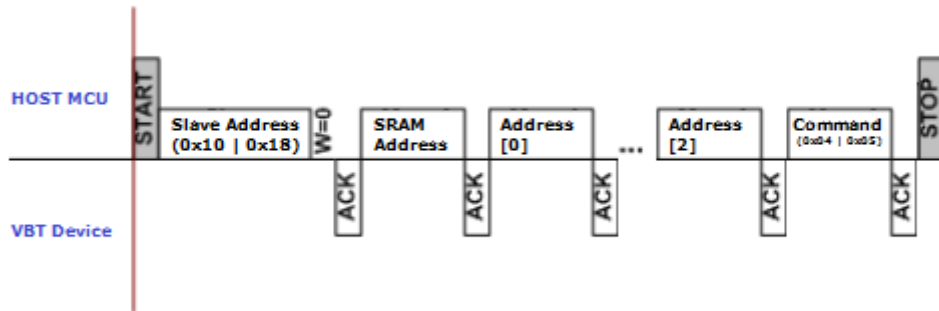
CHIP 溝通格式如下表所示：

SRAM Address	Context
0 : D0	Data [ 7 : 0 ]
1 : D1	Data [ 15 : 8 ]
2 : D2	Data [ 23 : 16 ]
3 : D3	Data [ 31 : 24 ]
4 : A0	ADDRESS[ 7 : 0 ]
5 : A1	ADDRESS[ 15 : 8 ]
6 : A2	ADDRESS[ 23 : 16 ]
7 : Command	VBTI2C_MEM_READ_TAG      0x05
	VBTI2C_MEM_WRITE_TAG      0x07

A 系列的 I2C 協定與標準 I2C 協定相同，支援 200 KHz 的讀寫操作，下面將詳細說明雙邊溝通的讀/取操作。

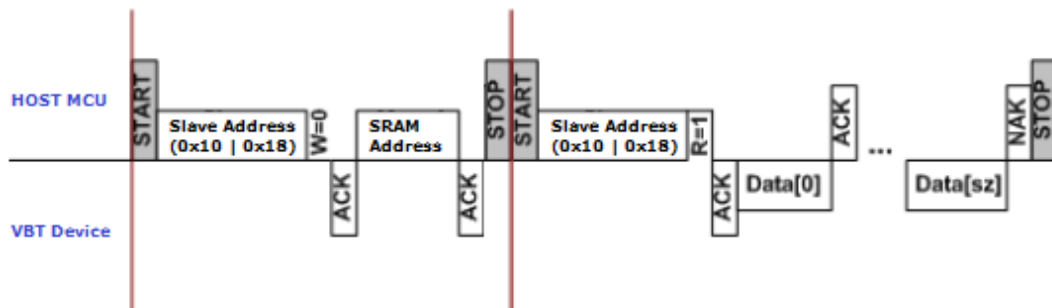
### ● 讀取操作

## 1. 變更 SRAM 的 Address 與 Command



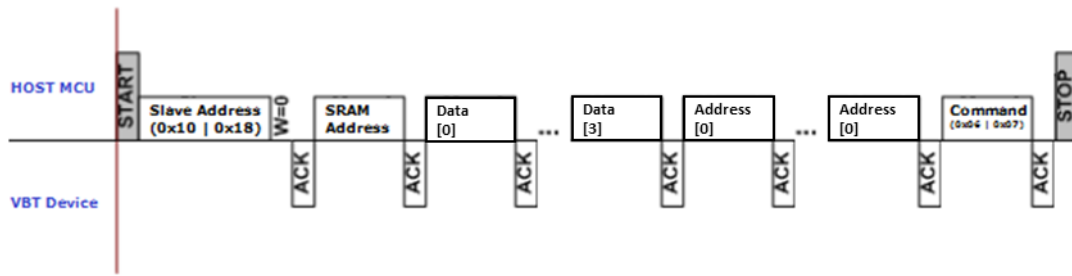
1. HOST MCU Send Start Signal.
2. HOST MCU Send Slave Write Address (0x10 | 0x18) and SRAM Address (0x04).
3. HOST MCU Send Current Access Address. (3 Bytes Length)
4. HOST MCU Send Register or Memory Read Command. (0x05)
5. HOST MCU Send Stop Signal.

## 2. 將資料從 SRAM 中取回，詳細操作如下：



1. HOST MCU Send Start Signal.
2. HOST MCU Send Slave Write Address (0x10 | 0x18) and SRAM Address (0x00).
3. HOST MCU Send Stop Signal. (Switch SRAM Address To 0x00)
4. HOST MCU Send Start Signal.
5. HOST MCU Send Slave Read Address (0x10 | 0x18).
6. HOST MCU Read Data. (4 Bytes).
7. HOST MCU Send Stop Signal.

### ● 寫入操作



1. HOST MCU Send Start Signal.
2. HOST MCU Send Slave Write Address (0x10 | 0x18) and SRAM Address (0x00).
3. HOST MCU Write Data. (4 Bytes)
4. HOST MCU Write Current Access Address. (3 Bytes)
5. HOST MCU Send Write Command. (0x07)
6. HOST MCU Send Stop Signal.

## USB 協定

透過 CHIP 支援的 USB Device 介面，透過標準的 EP0(end point) Setup Packet 對 HAL REGISTER 進行寫入與讀取操作。

## TS PROCESSOR 單元

由於不同數位電視調變規格與參數的設置會需要不同 **bitrate** 與需求的 TS stream，此單元主要用來將輸入與 PSI CREATOR 結合，針對輸出需求重新產生新的 TS stream，核心具備下列功能。

1. **TS FILTER**：過濾功能可以用來重新擷取 TS stream 的特別多媒體內容並輸出，最高可過濾 16 組 PID。
2. **TS DEMUX**：解析功能，可以解析來源 TS stream 的內容架構與簡易訊息，便於產品設置時設置過濾或開發 PSI TABLE 使用。
3. **REMUX**：複用功能，依據後端調製需求 ( Bitrate ) 將有效輸入結合 PSI TABLE 重新複用為實際輸出的 TS stream。
4. **TS CAPTURE**：擷取功能，可用來擷取小部分的流片段，已取得產生 PSI TABLE 所需要的內容詳細資訊。

## PSI CREATOR 單元

數位電視含聲音影像及其他資料流，依據不同國家與標準，需加入 PSI



TABLE，用來識別與定義頻道與多媒體內容，此單元提供兩種不同方式協助開發者完成所需要的 PSI TABLE。

1. **PURE TABLE**：由開發者依據應用情境參考所需規格書，自行加入自定義的 PSI TABLE。
2. **DEFAULT**：針對不同國家與規格所定義的基礎 PSI TABLE，透過參數的設置即可使用。

## MUXER 單元

基於不同調變模式與來源狀況，如 Bitrate、多媒體流時間...需經過 MUXER 加以重新排序，以符合數位電視廣播相關規範，MUXER 單元即是依據輸出需求與輸入條件進行 TS stream 的重新編排。

1. **PADDING 功能**：於資料無法滿足時會使用 NULL PACKET 進行填充，MUXER 提供自定義與標準功能。
2. **PCR INSERT**：PCR 的插入功能可加入獨立 PID 的 PCR，並且可控制 Interval 以符合應用需求。
3. **PCR REPLACE**：提供 PCR 覆寫功能，於不同輸入源 Bitrate 轉換時可能使原始 PCR 精度變更，透過覆寫功能可不同程度的修正此問題。

## MODULATOR 單元

廣泛支援全球數位電視調變規格，包括 DVB-T、DVB-C (J83a)、ATSC、j83b、DTMB、ISDB-T、J83c、DVB-T2。

## RF MIXER 單元

RF 控制輸出

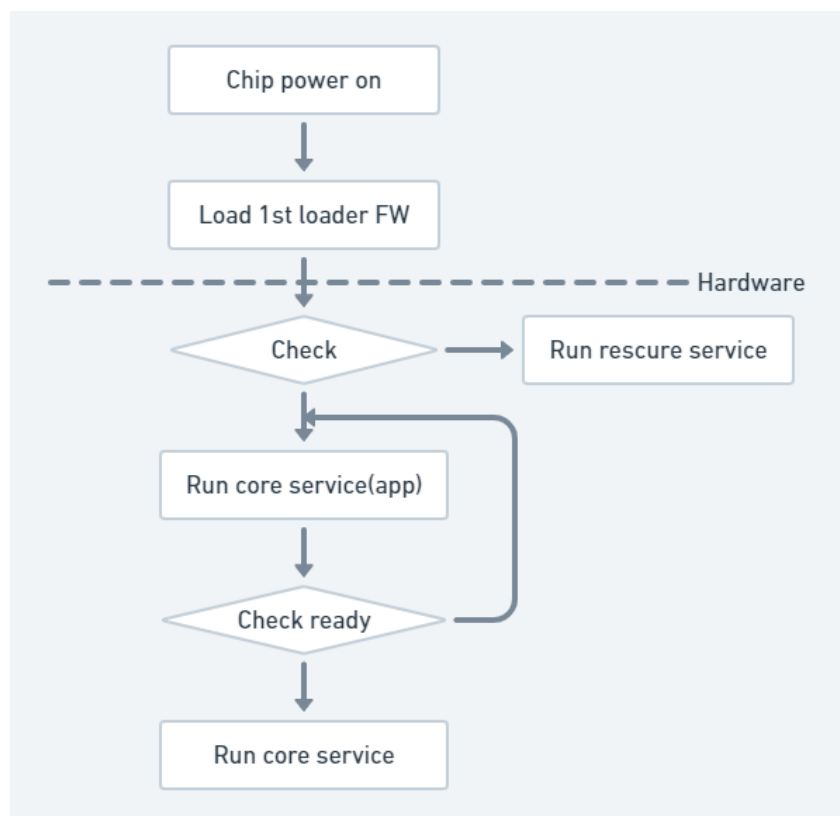
## 開機流程

系統執行主要由 Loader (開機程式) 與 Service (服務) 組成，當晶片完成上電程序，晶片會自動透過 SPI 介面載入並運行，運行後會檢測 RESCURE 訊號狀態，決定是否強制執行 RESCURE 服務，RESCURE 主要提供更新韌體功能。如果不須進入 RESCURE 服務則會檢查韌體，如果為有效韌體則載入並運行 Service (服務)。

透過外部 I2C 與 USB 透過 HAL 控制單元控制晶片功能，兩個路徑可控制階段將不同，由於 USB 介面由服務提供底層 USB 服務，故進入 RESCURE

與 Service ( 服務 ) 前無法與晶片溝通，而 I2C 則是在系統上電後即可進行讀取狀態與控制操作。

- 開機流程圖



## 3.2 系統功能

### 系統功能操作

在 HALREG\_TRANSFORM\_CNTL 中而外提供兩個系統功能分別為 TR\_REBOOT 與 TR\_REBOOT\_RESCURE，操作系統功能只能在系統狀態閒置時操作。兩個功能皆對系統進行軟體重置，TR\_REBOOT 為一般重置，系統重置後會重新載入 TRANSFORM 服務，而 TR\_REBOOT\_RESCURE 則會強制進入 RESCURE 服務。重置命令執行完成後檢查 HALREG\_BCINFO\_STATUS 確認執行結果，於 200 ms 後實際進行系統重置，開發者需重新運行檢查狀態流程。

A 系列晶片進行 TS 介面播放前須先確保晶片流程完整啟動，才能開始對晶片輸入 TS 訊號，否則可能會造成晶片無法取得 TS 封包產生錯誤。

- 晶片依據前端輸入介面分為 USB 及 TS 介面來接收 TS 訊號

輸入介面	描述
<b>USB interface</b>	USB 介面可控制 TS 輸入速度，USB2.0 最高可達 480Mbps。
<b>TS interface</b>	須為連續資料，若中途 TS 中斷將造成晶片發生錯誤。

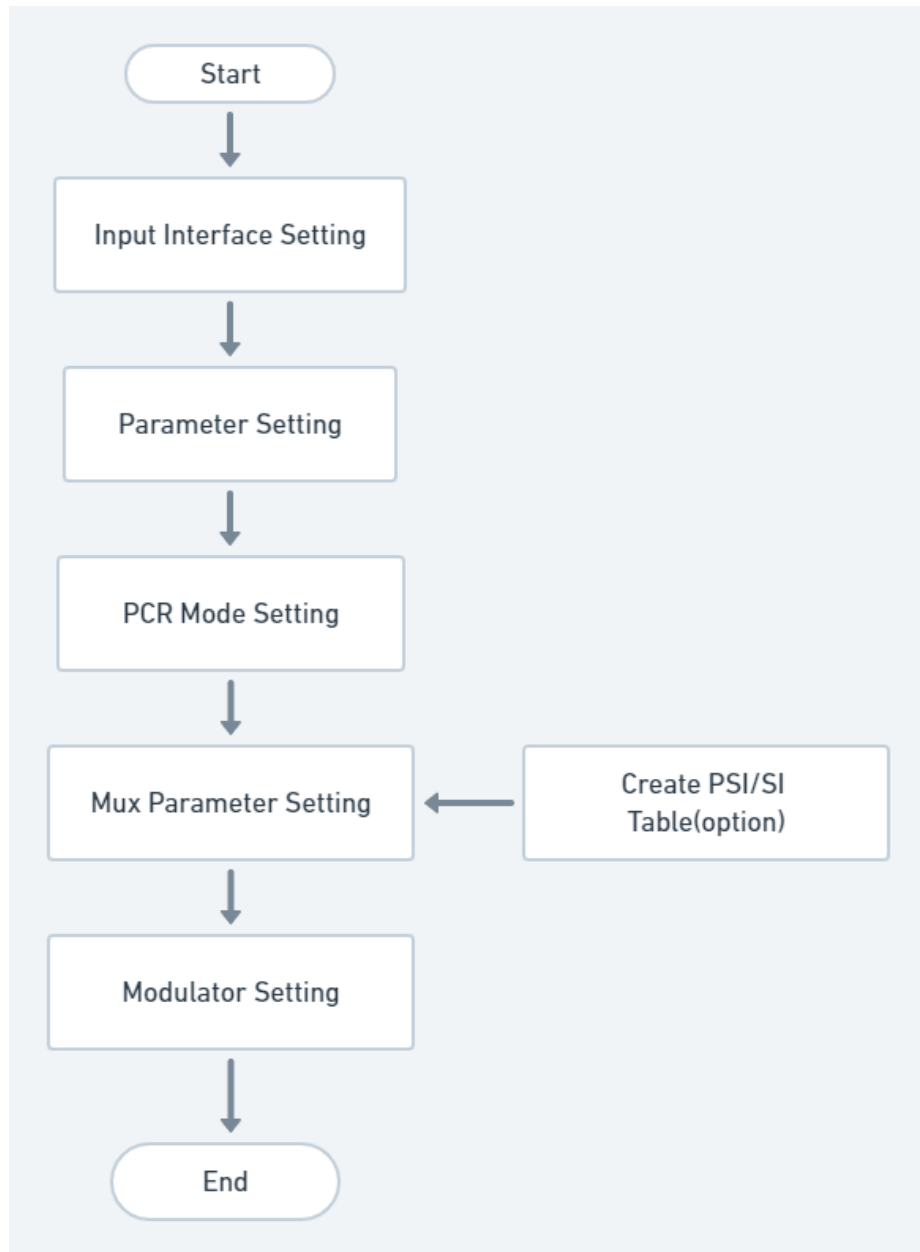
- 晶片依據前端輸入的 TS 完整度不同可以使用 PASSTHROUGH 及 REMUX 模式。

模式	適用情境
<b>PASSTHROUGH</b>	前端輸入 TS 已具備完整的 PES、PCR 及 PSI 內容，不希望更動到原本的 TS，僅需將訊號轉為數位電視訊號廣播。
<b>REMUX</b>	<ol style="list-style-type: none"> <li>1. 前端輸入 TS 不具備 PSI TABLE，需協助加入 PSI TABLE</li> <li>2. 前端輸入 TS 的 PCR 需進行校正。</li> <li>3. 加入 null packet 補足輸出端設定的 bitrate</li> </ol>

- A3 進行 REMUX 模式提供三種功能處理，可以針對 PCR 進行校正功能，如下表所敘

REMUX 功能	適用情境
<b>RETAG</b>	欲提高輸出的 PCR 精度，可使用該模式，其原理為依據前端 TS 輸入的 PCR 產生精度更高的 PCR 提供後端 TS 輸出的 PCR 參考
<b>ADJUST</b>	校正 DAC 的 PCR
<b>DISABLE</b>	不對 PCR 進行校正

## ● A 系列晶片系統流程圖



## 4. MDK

### 4.1 前言

VATEK MDK 提供開發者更容易且快速的使用 VATEK 數位廣播系統晶片，執行各種操作像是即時影音編碼、數位電視調變、數位電視串流複用、PSI 更新等各種數位廣播功能。

MCU 使用 I2C 介面對 VATEk 晶片及其周邊裝置進行控制，因此若要對晶片讀寫暫存器須先建立 I2C 的溝通介面。

MDK 以 STM 平台介面開發使用，針對 VATEk 晶片及周邊裝置如 PHY 及 R2 進行控制。若使用者 MCU 平台不同，MDK 提供移植功能，可以將 GPIO、I2C、UART 等介面移植至其他平台使用。

### 4.2 MDK 軟體架構

以下說明 MDK 架構，以幫助使用者更加了解 MDK 及 VATEk 晶片提供的功能，以利加速後續開發及使用。

#### ● MDK 資料夾架構

	資料夾名稱	描述
	Inc	提供的 API、晶片的設定參數、外部設備的設定參數...
	main	晶片相關的功能實作
	peripheral	外部設備的功能實作
	project	VATEk 以 Keil MDK 提供的範例專案
	sample	VATEk 提供的範例程式
	system	系統相關的功能實作，如傳輸介面、平台移植...等

### 4.3 A 系列 MDK 說明

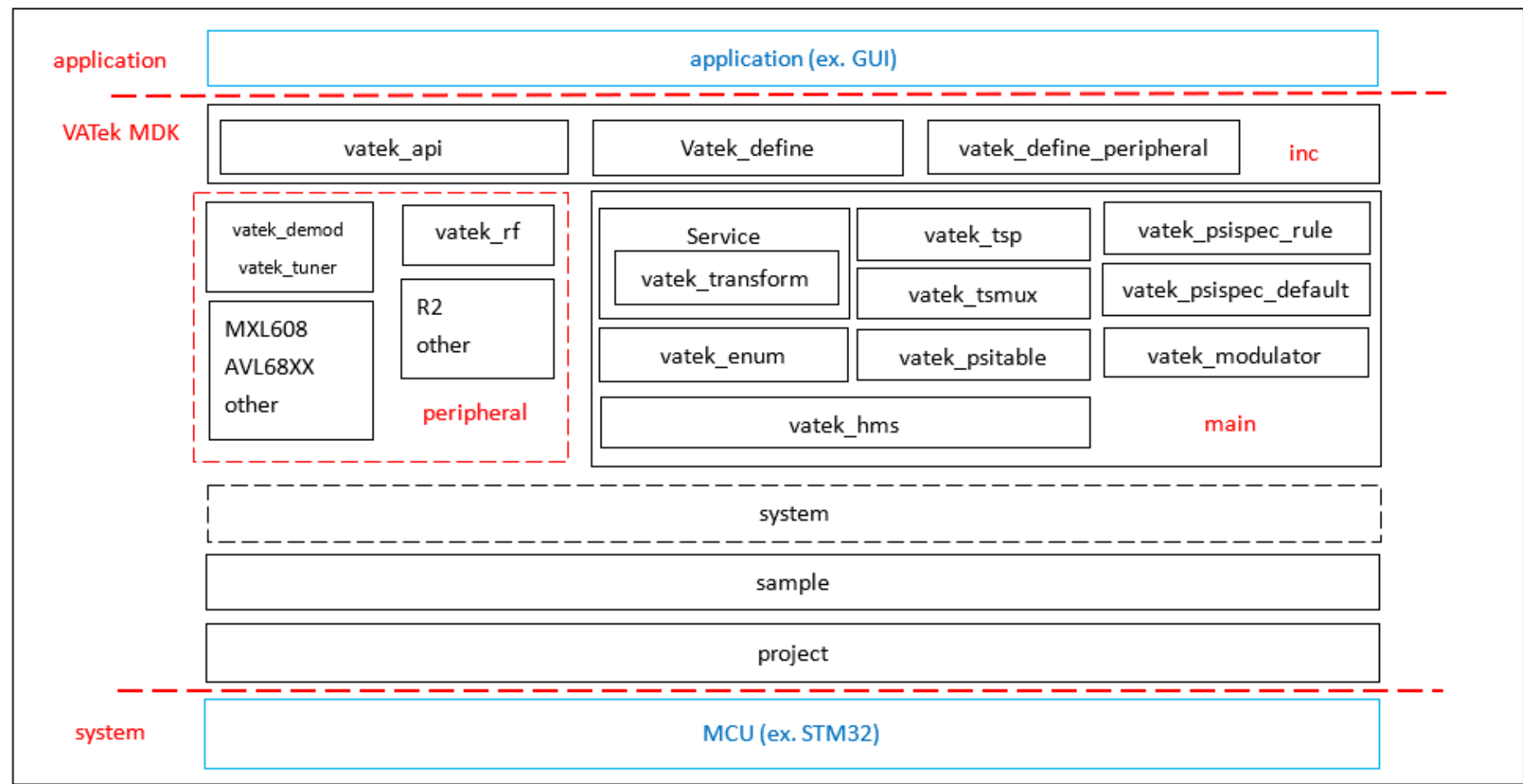
#### 4.3.1 A 系列軟體系統架構

下圖為 A 系列軟體架構圖，中間部分為 MDK 的軟體架構，以下介紹廣播時主要會使用到的 MDK 內容：

- **vatek\_api**：提供使用者直接呼叫使用的 API
- **vatek\_transform**：A 系列晶片所提供的功能實作，如設定輸入介面、處理方式及加入 PSI table 方法等流程

- **vatek\_demod, vatek\_tuner** : 為 VATEk 提供的特定前端 DEMOD 及 Tuner 驅動，例如 MXL608, AVL68XX...等
- **vatek\_rf** : VATEk 提供晶片輸出端的 RF mixer 驅動
- **vatek\_hms** : 暫存器讀寫功能的實作
- **system** : MCU 平台的介面設計，使用者可以藉由 **system** 資料夾內的 porting 功能替換成其他非 STM 平台進行使用

A 系列軟體系統架構圖



### 4.3.2 編譯定義

因為 SDK 支援數個周邊驅動，為了 SDK 的一致性與節省記憶體空間，而使用了不同的編譯標誌去區別不同的驅動程式。使用者可依據實際環境去選擇不同的驅動程式進行編譯，以下為 SDK 內各驅動程式的定義：

類別	定義
VATEK 晶片	
A3	VATEK_A3
Demodulator	
AVL68xx	DEMOD_AVL68XX
LGDT3305	DEMOD_LGDT3305
Tuner	
MXL608	TUNER_MXL608
MXL603	TUNER_MXL603
RF	
R2	RF_R2
R2 (control by VATEK)	RF_R2_VIA_VATEK
MCU	
STM32F407xx	STM32F407xx
STM32F401xC	STM32F401xC

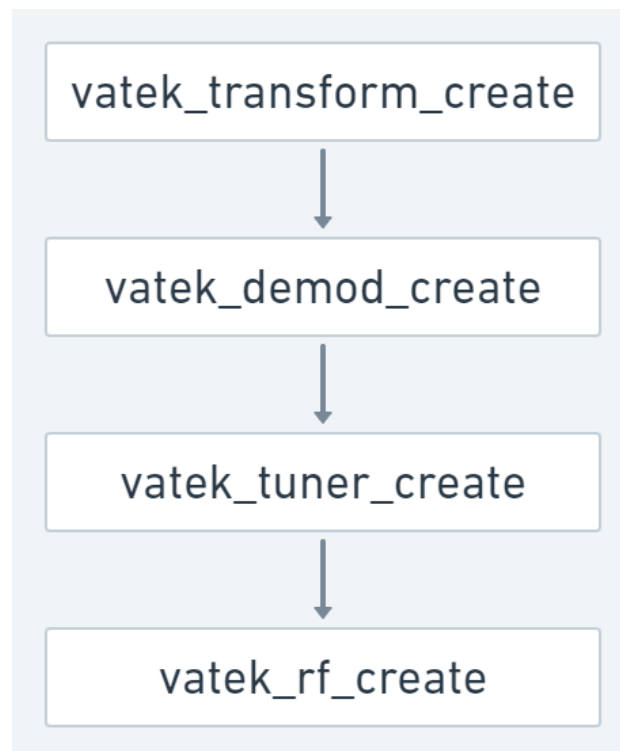


### 4.3.3 API 使用範例

以下提供使用者執行 A 系列晶片轉換及廣播功能所使用到的 API 流程說明

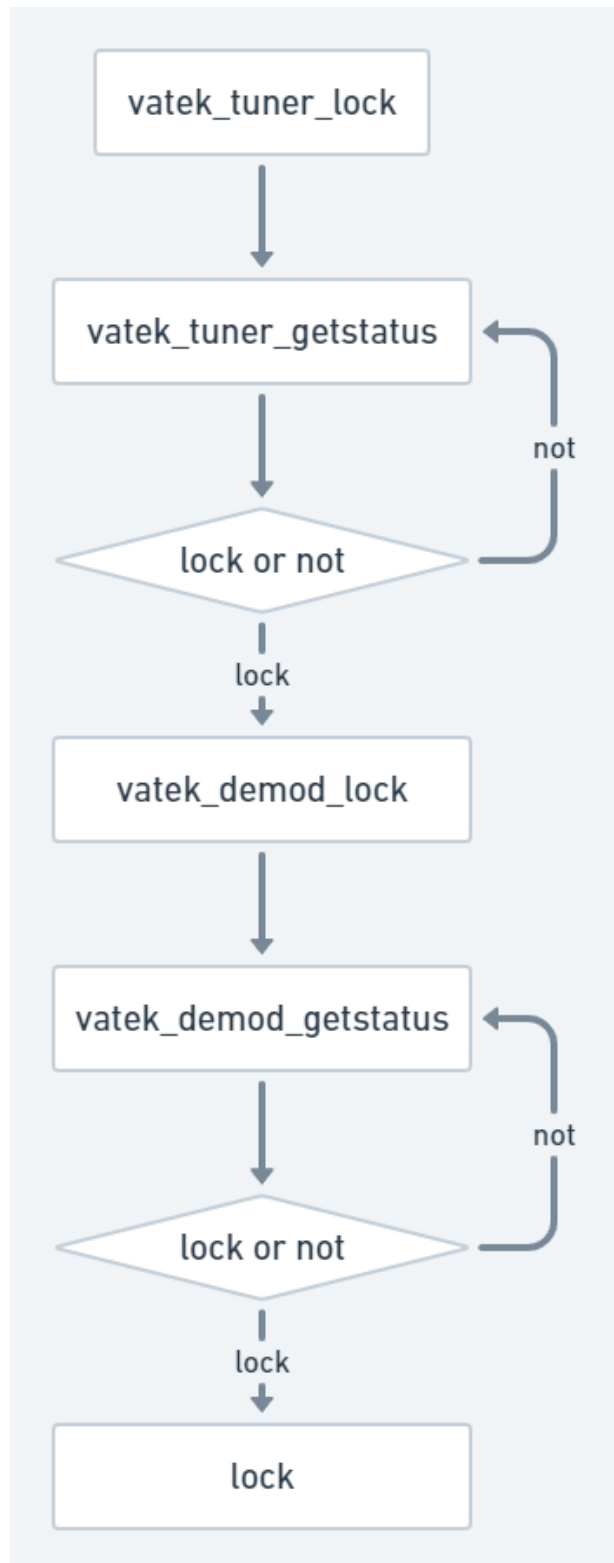
#### 電視廣播轉換初始化

電視廣播轉換系統是由 VATEK 晶片與周邊(Tuner、DEMOD、RF)組成。使用前需對這些裝置作初始設定，流程如下：



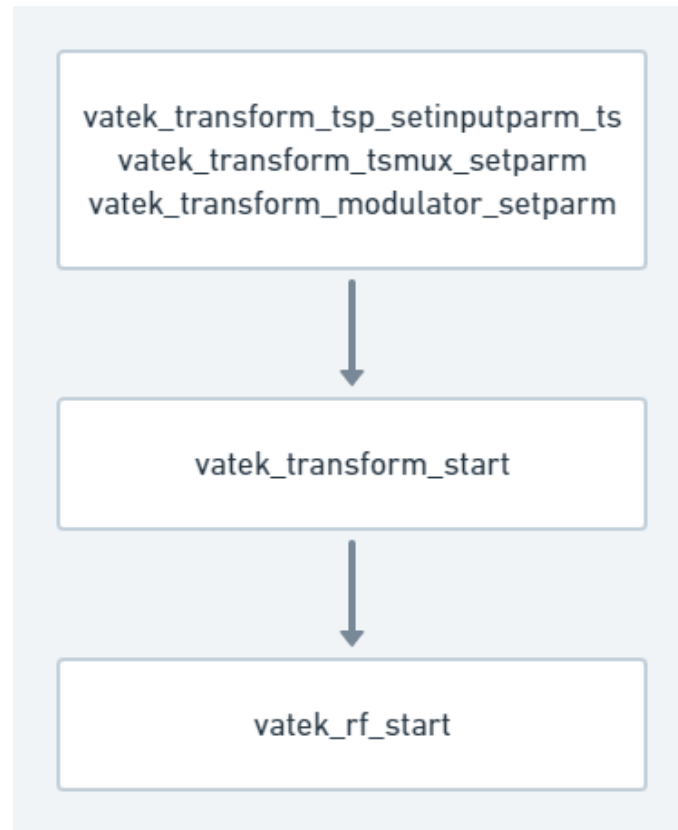
## 鎖定傳輸串流訊號(Tuner 與 DEMOD)

鎖定 TS 訊號來源：



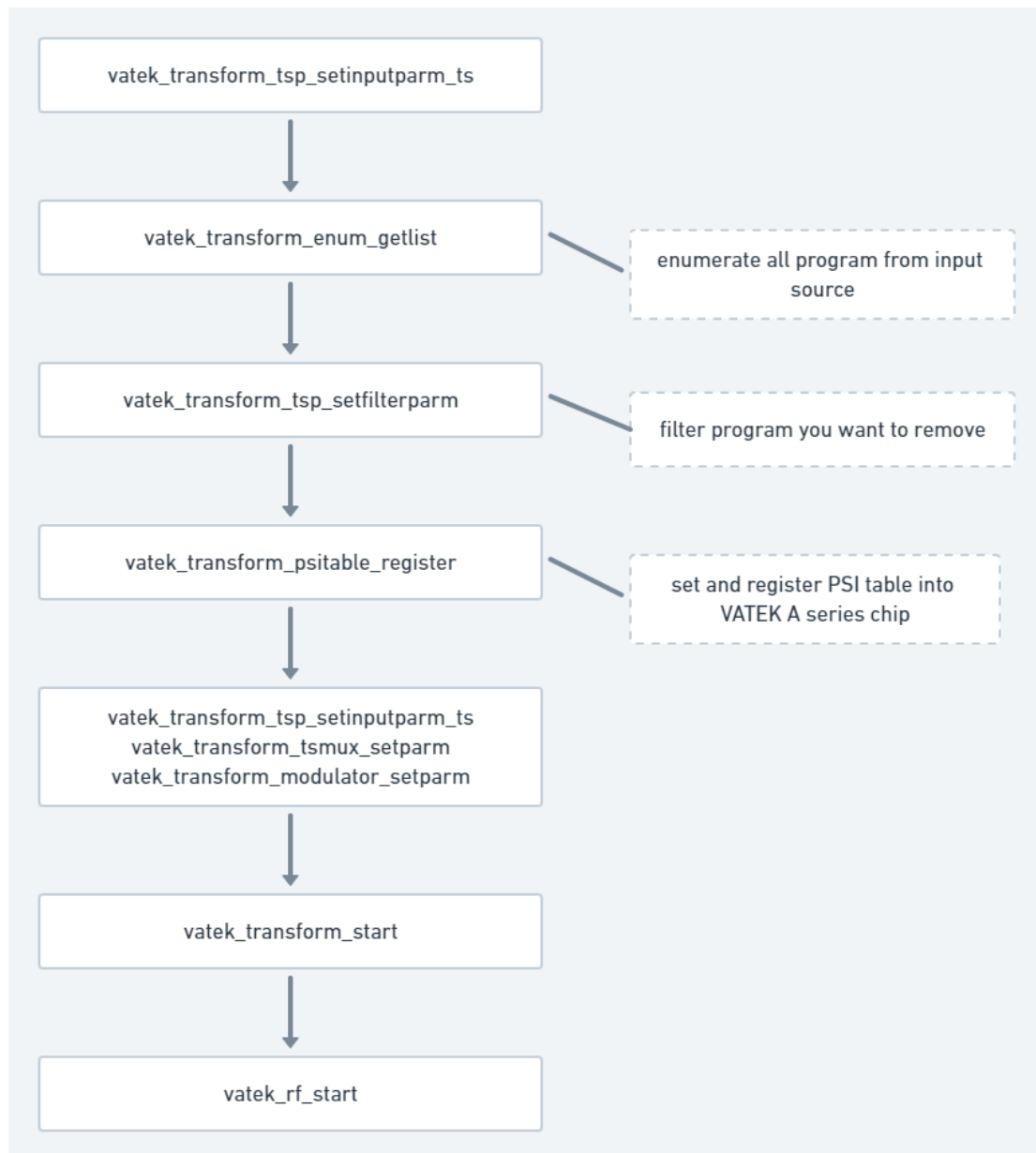
## 廣播制式轉換或更換頻率

使用需求為將傳輸串流更換頻率或轉換廣播制式(DVB-T 轉換成 ATSC) , 操作如下 :



## 過濾電視廣播節目

如使用需求為刪減原傳輸串流內容。操作如下：



## 停止廣播轉換

如需要重新廣播轉換，需要先停止原本的動作。使用如下：



## 4.3.4 MDK Porting 功能

VATEK 提供 I2C, GPIO, UART 介面移植功能，請參考 `system` 資料夾內的 `vatek_porting_xxxx.c`，若使用不同 MCU 僅需將其介面新增到指定的 `porting` 檔案中，就能在不影響整體 `code` 的情況下成功將 MDK 移植到其他 MCU，以 `vatek_porting_gpio.c` 的 `code` 為例，使用者僅需將 MCU 對應的 GPIO 讀寫方法新增到下列功能內即可，MDK 預設使用 STM32F407 及 401

```
vatek_result vatek_porting_gpio_write(gpio_pin pin, uint8_t val)
```

```
{
```

```
    vatek_result result = vatek_result_success;
```

```
#if defined(STM32F407xx) || defined(STM32F401xC)
```

```
    if (val)
```

```
        HAL_GPIO_WritePin((GPIO_TypeDef*)pin.port, pin.index,  
GPIO_PIN_SET);
```

```
    else
```

```
        HAL_GPIO_WritePin((GPIO_TypeDef*)pin.port, pin.index,  
GPIO_PIN_RESET);
```

```
#endif
```

```
    return result;
```

```
}
```

```
vatek_result vatek_porting_gpio_read(gpio_pin pin, uint8_t *val)
```

```
{
```

```
    vatek_result result = vatek_result_success;
```

```
#if defined(STM32F407xx) || defined(STM32F401xC)
```

```
    *val = HAL_GPIO_ReadPin((GPIO_TypeDef*)pin.port, pin.index);
```

```
#endif
```

```
    return result;
```

```
}
```

## 4.3.5 建置 MDK 環境

以 STM32F401 開發版為例，所需軟硬體如下

### 軟體

MDK：VATek 提供之 MDK 程式碼

IDE：任何可開啟 STM cubeMX 建置環境的編碼工具，例如 Keil C

### 硬體

Demo board：VATek 的 A 系列晶片及 STM32F401 之 PCB 開發版並進行連接

PC：用來開啟 MDK 程式碼及載入韌體檔案

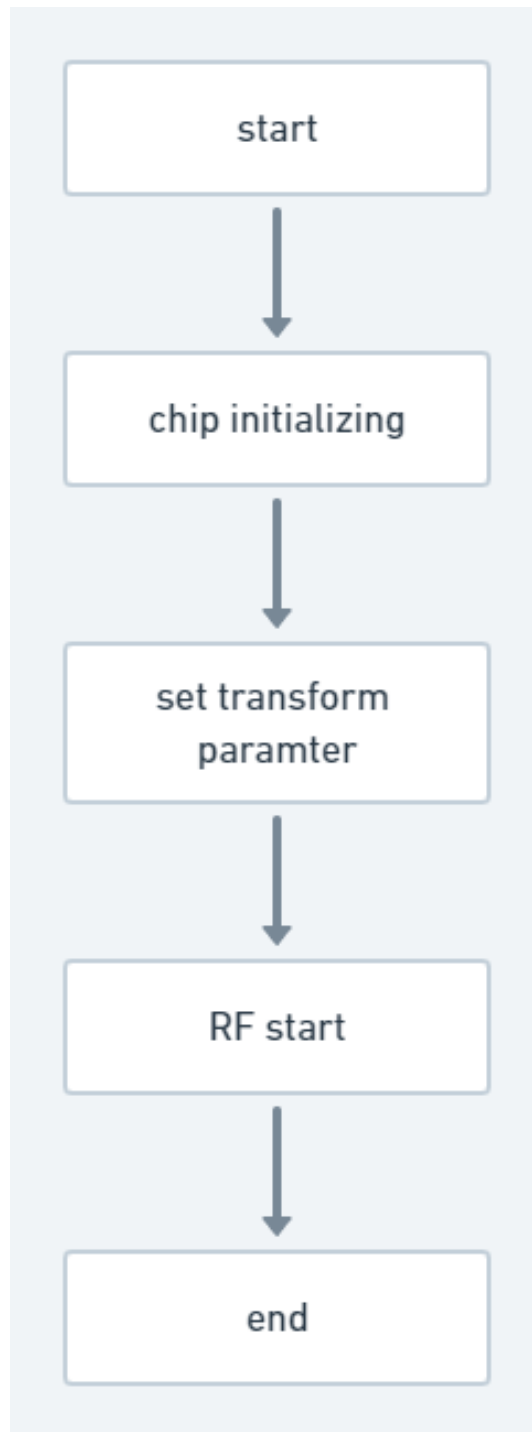
ST-Link：使用電腦對 STM PCB 進行韌體載入時需要的溝通介面

DEMOD or Tuner，用來解析前端 TS 輸入介面的 TS stream

### 建置說明

VATek MDK 提供以 Keil C 環境建置的編碼環境，放置於 project/transform 資料夾內，使用者須以 Keil C 應用程式開啟，專案內已先行編寫好廣播範例的程式碼，使用者僅需將程式碼載入至 MCU，並重置 MCU 及 A 系列 Demo Board，即可開始將 TS 轉換輸出數位電視訊號

#### 4.3.6 MDK sample code 流程





## 4.4 API 功能說明

API 用來對暫存器進行讀寫，可以藉由讀取暫存器得知晶片狀態，以及對暫存器寫入指令來控制晶片。為使開發者能夠對 API 更加認識，以下提供所有 API 索引及簡要說明，若需進一步了解 HALREG 暫存器請參閱 HALREG MAP 說明文件。

### 4.4.1 系統通用 API

API name : vatek\_system\_gettick

描述：取得目前系統經過時間，以毫秒(ms)為單位，主要用來檢查晶片執行功能是否在預期時間內

API name : vatek\_system\_delay

描述：系統延遲功能，以毫秒(ms)為單位，當部分模組功能需要時間準備時可以使用該 API 進行延遲

API name : vatek\_system\_crc32

描述：取得 CRC 檢查參數

### 4.4.2 A 系列晶片 API(Transform)

此節 API 用來控制 A 系列晶片，主要用來寫入及讀取 A 系列晶片的暫存器

API name : vatek\_transform\_create

描述：初始化 A 系列晶片

API name : vatek\_transform\_destroy

描述：終止 A 系列晶片服務

API name : vatek\_transform\_start

描述：開始 A 系列晶片服務

API name : vatek\_transform\_stop

描述：暫停 A 系列晶片服務

API name : vatek\_transform\_reset

描述：重置 A 系列晶片服務

API name：vatek\_transform\_chipstatus

描述：取得 A 系列晶片狀態

API name：vatek\_transform\_bcstatus

描述：取得 A 系列晶片播放狀態

API name：vatek\_transform\_tsp\_setinputparm\_ts

描述：設定 TS 輸入參數

API name：vatek\_transform\_tsp\_setfilterparm

描述：設定 TS 過濾功能

API name：vatek\_transform\_tsmux\_setparm

描述：設定 PSI 產生模式

API name：vatek\_transform\_modulator\_setparm

描述：設定調變參數，依據不同調變需要設定不同參數

API name：vatek\_transform\_psitable\_register

描述：VATek 提供的 PSI PURE 模式中的靜態註冊方式，於晶片 IDLE 狀態時才能設定 PSI TABLE 並進行註冊寫入 PSI Table

API name：vatek\_transform\_psitable\_insert

描述：VATek 提供的 PSI PURE 模式中的動態註冊方式，於晶片廣播狀態時才能設定 PSI TABLE 並寫入 PSI Table

API name：vatek\_transform\_psispec\_default\_init

描述：A 系列晶片在 PSI Default 模式下取得記憶體位置並進行初始化

API name：vatek\_transform\_psispec\_default\_channel\_config

描述：A 系列晶片在 PSI Default 模式下設定頻道的 PSI table 參數

API name：vatek\_transform\_psispec\_default\_program\_add

描述：A 系列晶片在 PSI Default 模式下新增節目資訊及 PCR 相關參數

API name：vatek\_transform\_psispec\_default\_program\_end

描述：A 系列晶片在 PSI Default 模式下完成新增節目資訊後，最後寫入結束的 Tag

API name：vatek\_transform\_psispec\_default\_stream\_add

描述：A 系列晶片在 PSI Default 模式下新增影音及其他資料流的串流

API name：vatek\_transform\_psispec\_default\_start

描述：A 系列晶片在 PSI Default 模式下開始寫入設定完成的 PSI table

### 4.4.3 外部裝置相關 API

VATek MDK 提供 Demo board 使用的周邊裝置的 API，若開發者使用其他的外部裝置則須自行新增相關驅動及 API

#### **RF**

API name：vatek\_rf\_create

描述：初始化 RF 裝置

API name：vatek\_rf\_destroy

描述：終止 RF 裝置

API name：vatek\_rf\_start

描述：開始 RF 輸出

API name：vatek\_rf\_stop

描述：暫停 RF 輸出

API name：vatek\_rf\_getstatus

描述：取得 RF 當前狀態

#### **DEMODO**

在 A 系列晶片使用 TS 輸入介面時，需先解析輸入的 TS，再進行處理

API name：vatek\_demod\_create

描述：初始化 DEMODO 裝置

API name：vatek\_demod\_destroy

描述：終止 DEMODO 裝置

API name : vatek\_demod\_i2cbypass

描述：設定是否藉由 I2C 介面來控制 DEMOD 裝置

API name : vatek\_demod\_lock

描述：鎖定 DEMOD 訊號

API name : vatek\_demod\_getstatus

描述：取得 DEMOD 當前狀態

## **TUNER**

在 A 系列晶片使用 TS 輸入介面時，需先解析輸入的 TS，再進行處理。

API name : vatek\_tuner\_create

描述：初始化 Tuner 裝置

API name : vatek\_tuner\_destroy

描述：終止 Tuner 裝置

API name : vatek\_tuner\_lock

描述：鎖定 Tuner 訊號

API name : vatek\_tuner\_getstatus

描述：取得 Tuner 當前狀態

API name : vatek\_tuner\_gettrfstrength

描述：取得 Tuner 訊號強度