



EPCIO Series

驅動函式庫

範例手冊

版本：V.6.00

日期：2020.06

<http://www.epcio.com.tw>



目 錄

1. 驅動函式庫範例簡介	3
1.1 DDA 模組流程圖	4
1.2 ENC 模組流程圖	5
1.3 DDA & ENC 模組流程圖	6
1.4 DAC 模組流程圖	7
1.5 ADC 模組流程圖	8
1.6 PCL 模組流程圖	9
2. DDA Cycle 中斷產生脈波輸出功能	10
3. DDA 脈波輸出控制及使用 Cycle 中斷功能	11
4. 讀取 DDA FIFO 旗標狀態	12
5. DDA 脈波輸出控制及使用 FIFO 中斷功能	13
6. DDA 脈波輸出計錄器控制	14
7. 在輸出脈波過程時進行緊急停止控制	15
8. DDA 脈波輸出控制	16
9. DDA FIFO 內命令筆數讀取控制	17
10. DDA FIFO 中斷功能	18
11. 開啟編碼器 INDEX (Z Phase)觸發中斷功能	19
12. 開啟編碼器比較值中斷功能	20
13. 開啟編碼器輸入功能	21
14. 開啟編碼器的門鎖(Latch)功能	22
15. 開啟編碼器的比較器觸發 DAC 類比電壓輸出	23
16. Timer 計時器中斷功能	24
17. ADC 比較器中斷功能控制	25
18. ADC 電壓輸入(連續雙極性模式)讀取控制	26
19. ADC 電壓輸入與讀取工作狀態	27
20. 開啟 ADC 標籤 Channel 中斷功能	28
21. ADC 輸入電壓比較器觸發 DAC 類比電壓輸出	29
22. ADC 電壓輸入(單次雙極性模式)讀取控制	30
23. DAC 類比電壓輸出	31
24. LIO 近端輸入點觸發 DAC 類比電壓輸出	32
25. RIO 遠端輸入點讀取及中斷功能	33



26. RIO 遠端輸出入點的讀取與寫出控制	34
27. RIO 傳輸錯誤中斷控制	35
28. RIO 傳輸狀態讀取控制	36
29. RIO 遠端輸入點觸發 DAC 類比電壓輸出	37
30. PCL 硬體位置閉迴路控制功能	38
31. 硬體位置閉迴路 PCL 控制及使用 DDA Cycle 中斷功能	39



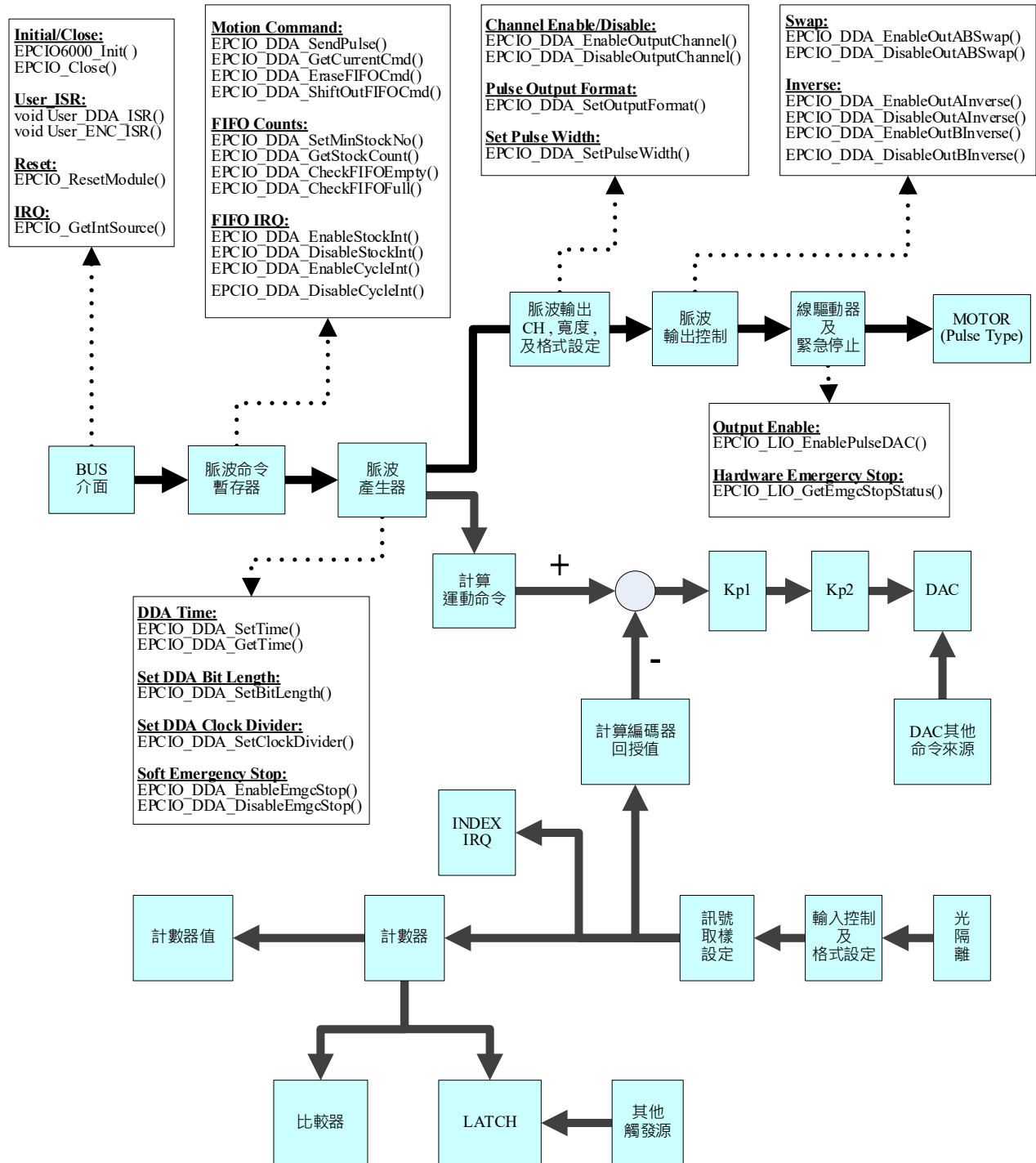
1. 驅動函式庫範例簡介

安裝光碟中所提供的驅動函式庫(EDDL)範例程式，VC++為 Windows Console 的應用程式範例程式，主要針對各功能模組作一使用說明，包含 DDA 模組脈波輸出、ENC 模組編碼器輸入、計時器程式規劃、ADC 類比電壓輸入、DAC 類比電壓輸出、LIO 近端輸出入點控制、RIO 遠端輸出入點控制、PCL 硬體閉迴路控制等；有關 EDDL 各函式詳細說明請參考”*EPCIO Series 驅動函式庫參考手冊*”。

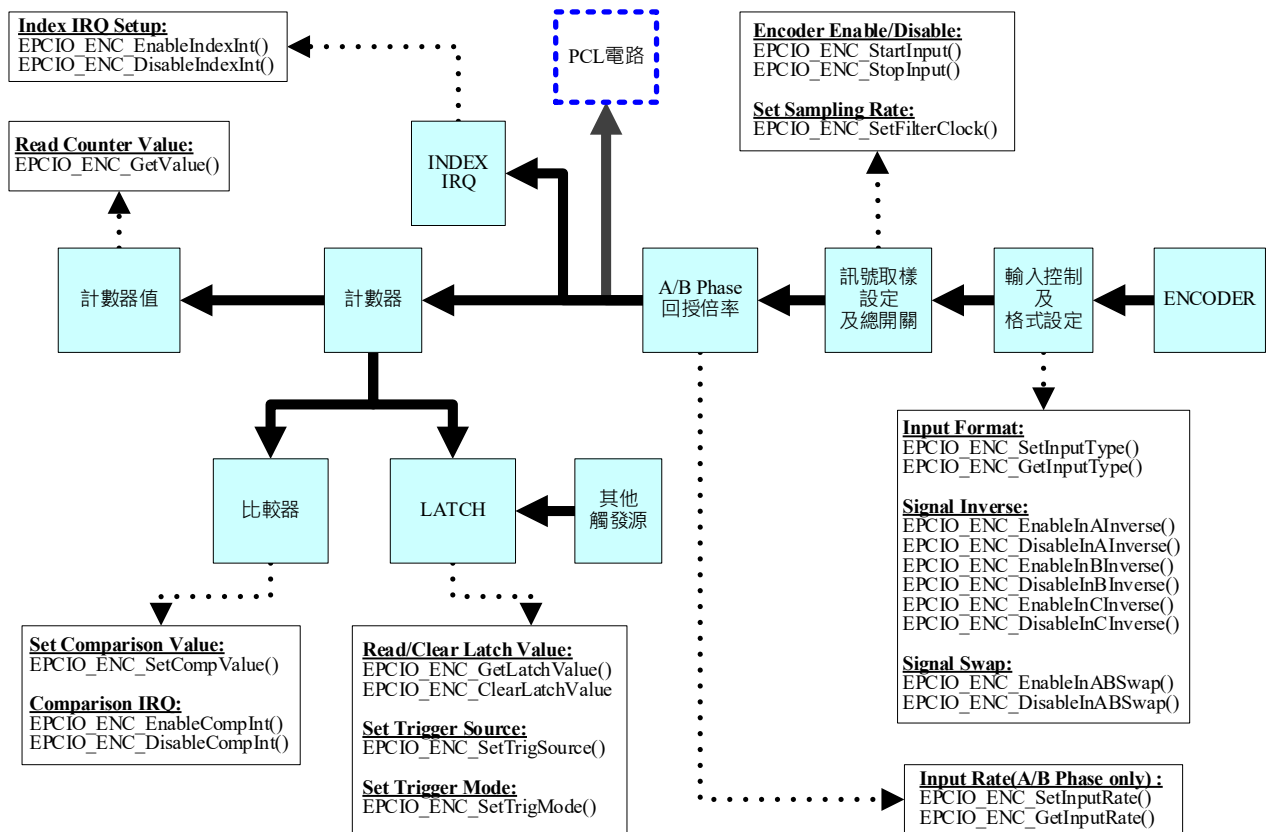
EPCIO Series 安裝光碟的程式會協助使用者把相關的檔案內容放至指定目錄中，使用者只須執行安裝步驟即可；待安裝完成後，可利用 EDDL 範例程式的原始檔與參考後續章節的功能模組流程圖與範例說明，瞭解 EDDL 的使用流程。



1.1 DDA 模組流程圖

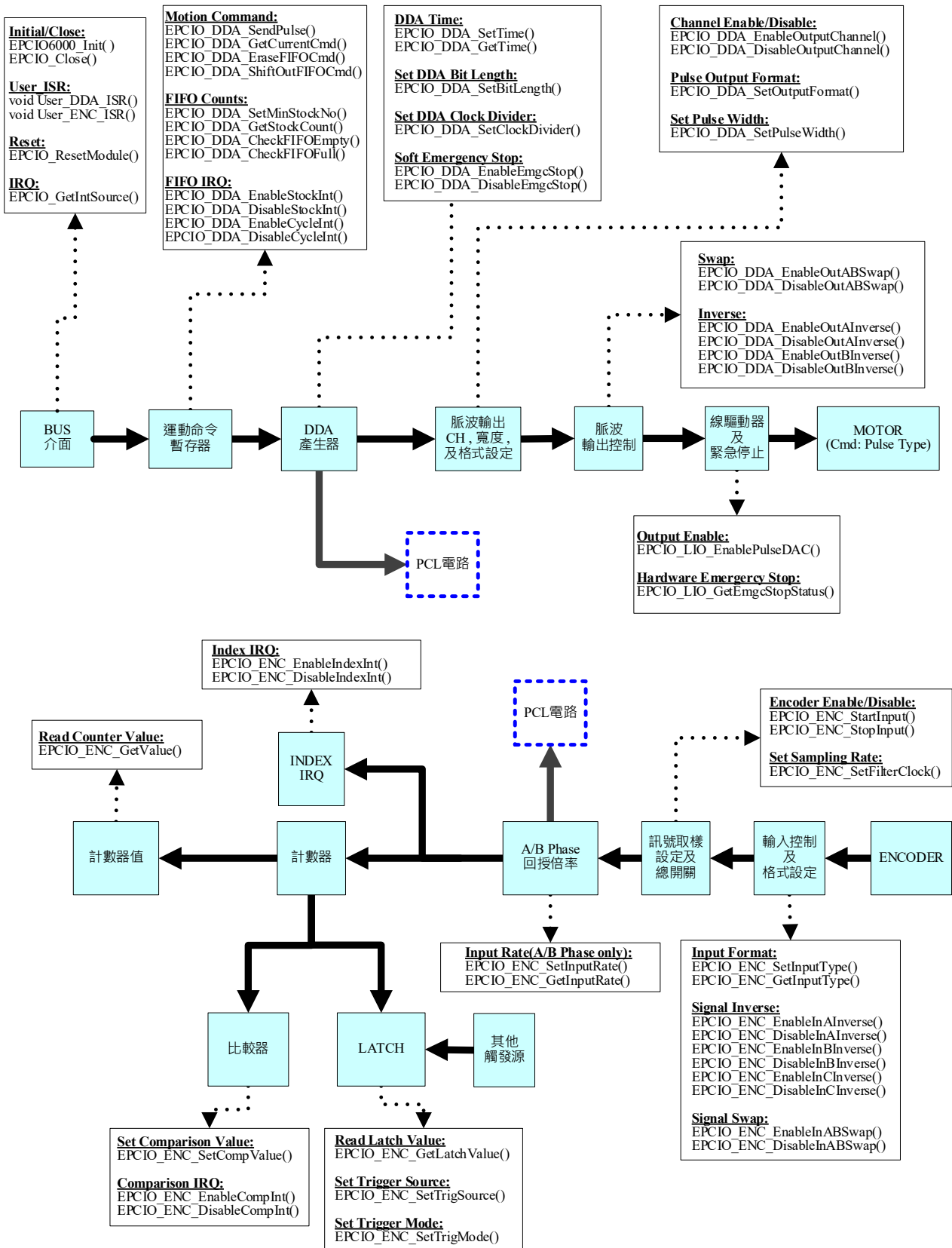


1.2 ENC 模組流程圖

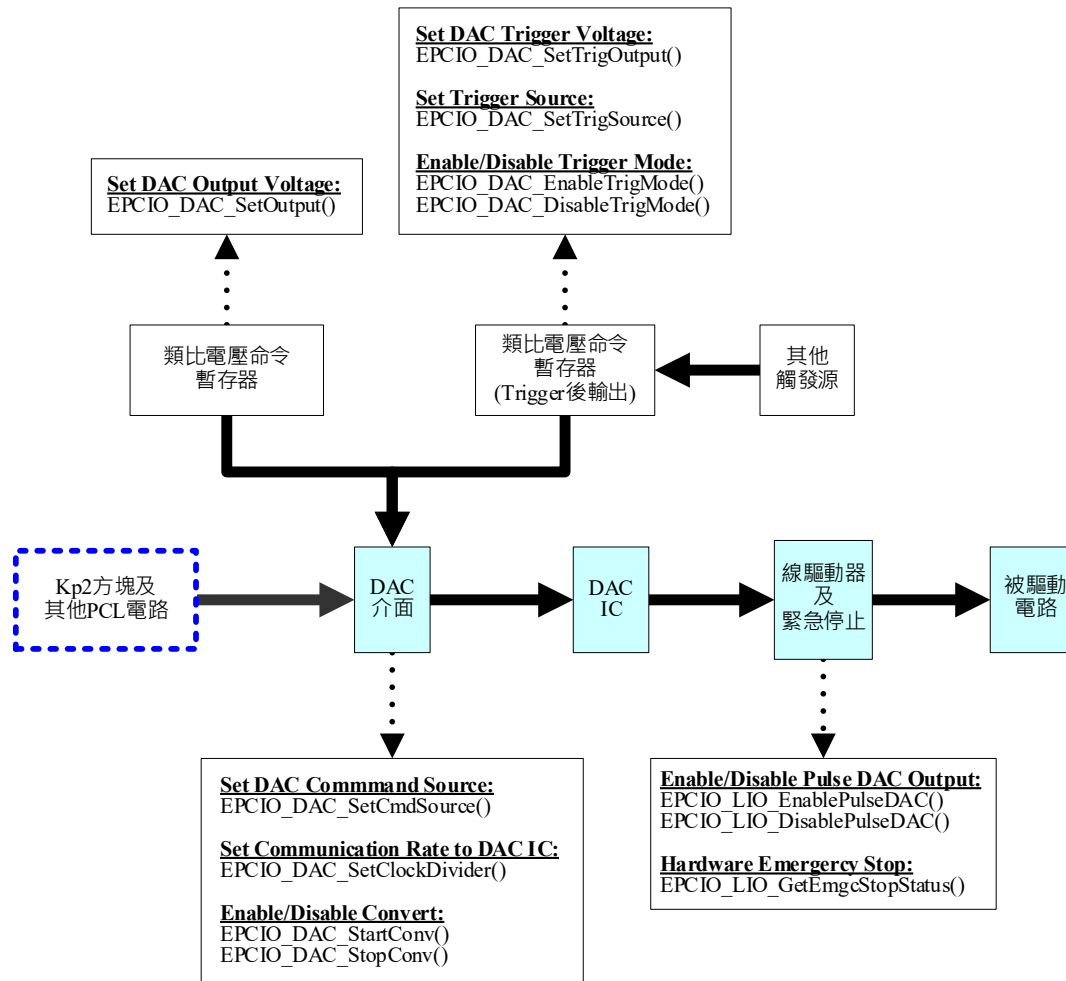




1.3 DDA & ENC 模組流程圖

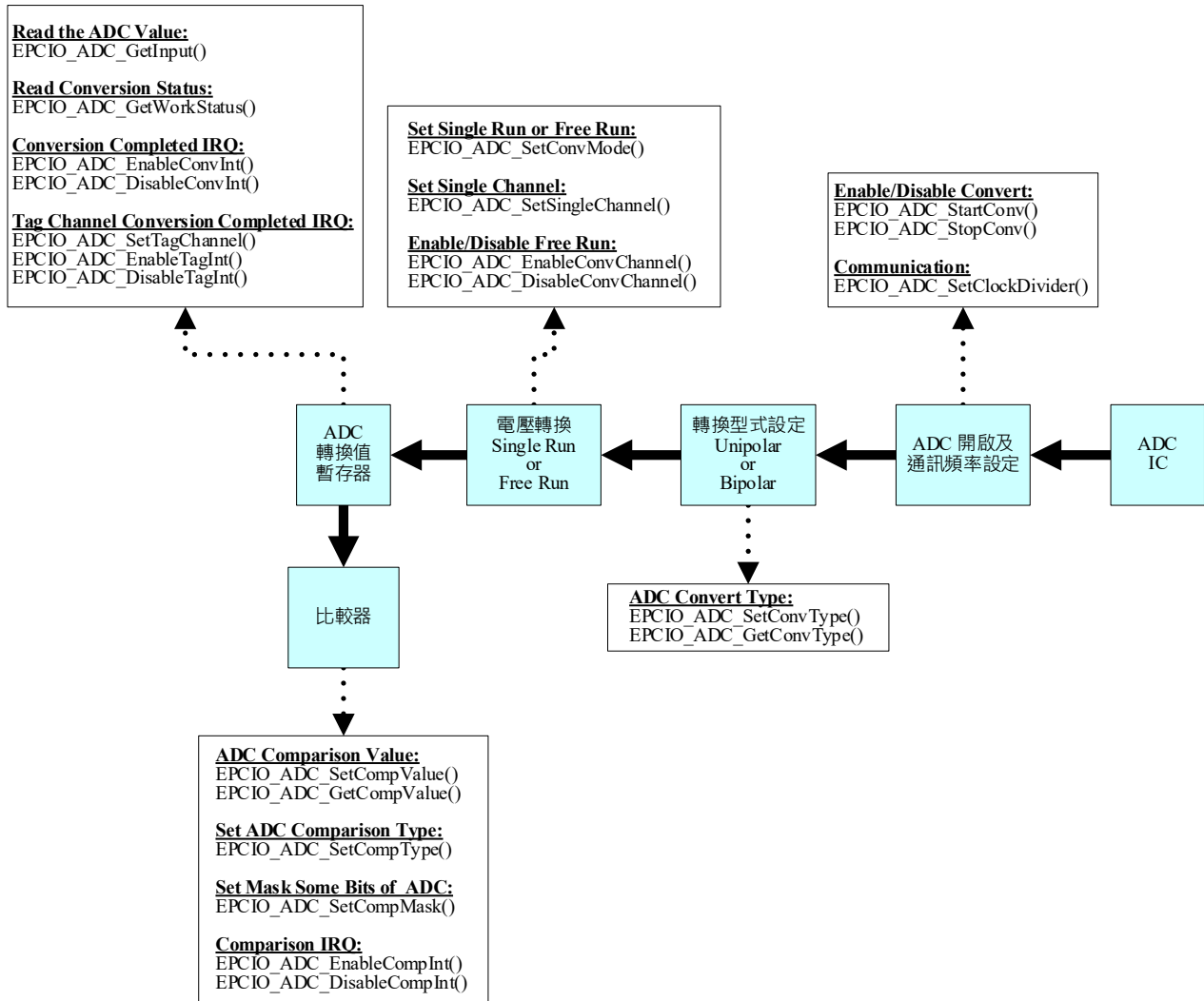


1.4 DAC 模組流程圖

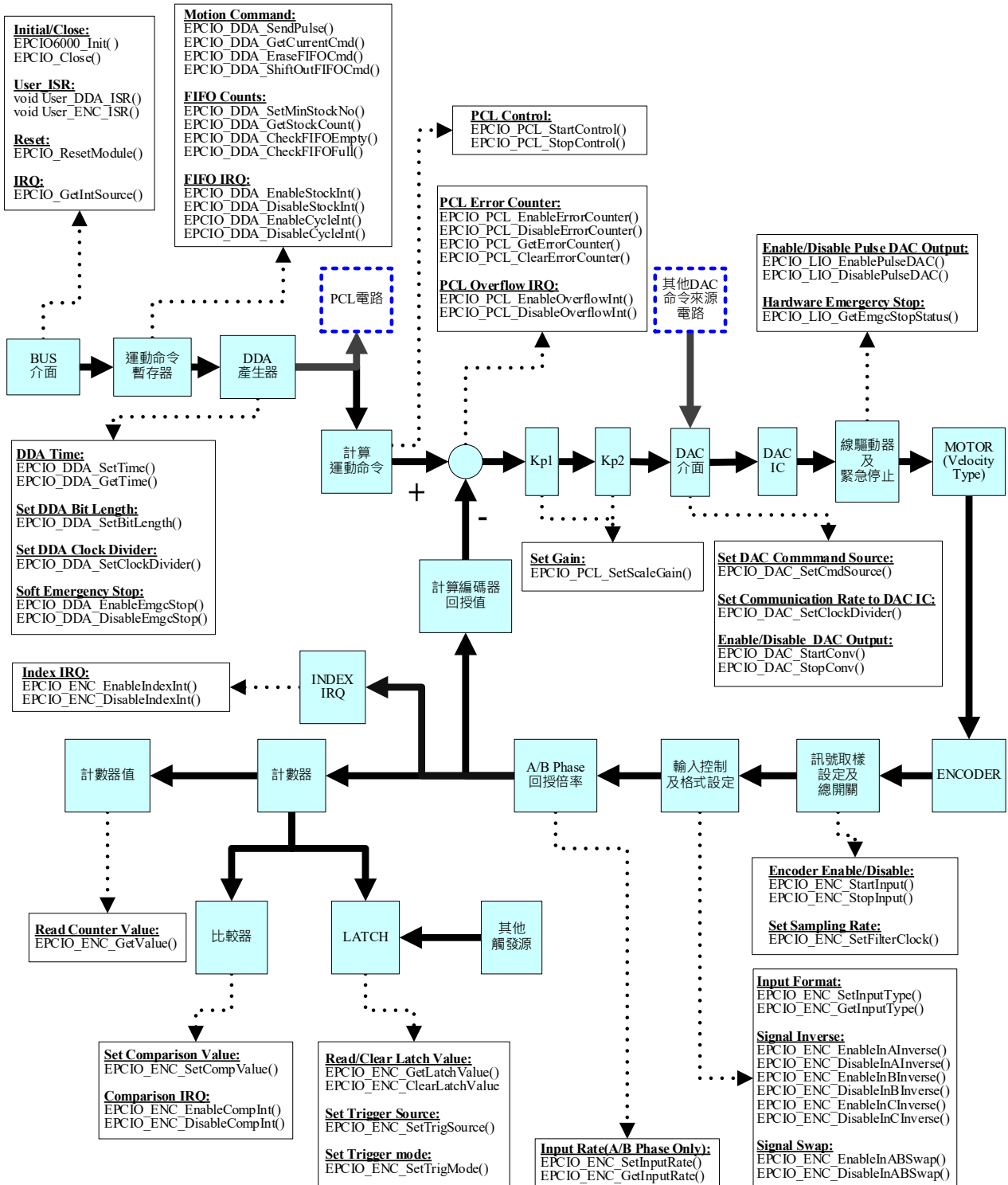




1.5 ADC 模組流程圖



1.6 PCL 模組流程圖





2. DDA Cycle 中斷產生脈波輸出功能

■ 相關函式

EPCIO_LIO_EnablePulseDAC()
EPCIO_DDA_SetTime()
EPCIO_DDA_EnableOutputChannel()
EPCIO_DDA_DisableOutputChannel()
EPCIO_DDA_EnableCycleInt()
EPCIO_DDA_DisableCycleInt()
EPCIO_DDA_StartEngine()
EPCIO_DDA_StopEngine()
EPCIO_DDA_SendPulse()

■ 範例程式

CycleInt.cpp

■ 內容說明

本範例程式將開啟 DDA 的 Cycle 中斷功能，並且在中斷服務函式中，從 Channel 0 送出固定的脈波數。

3. DDA 脈波輸出控制及使用 Cycle 中斷功能

■ 相關函式

EPCIO_LIO_EnablePulseDAC()
EPCIO_DDA_SetTime()
EPCIO_DDA_EnableOutputChannel()
EPCIO_DDA_DisableOutputChannel()
EPCIO_DDA_EnableCycleInt()
EPCIO_DDA_DisableCycleInt()
EPCIO_DDA_StartEngine()
EPCIO_DDA_StopEngine()
EPCIO_DDA_SendPulse()

■ 範例程式

CycPulse.cpp

■ 內容說明

本範例程式將對 Channel 0 輸出脈波，並規劃一個發生於固定時脈的 DDA Cycle 中斷，並且在中斷服務函式中輸出下一筆命令。



4. 讀取 DDA FIFO 旗標狀態

■ 相關函式

EPCIO_LIO_EnablePulseDAC()

EPCIO_DDA_CheckFIFOEmpty()

EPCIO_DDA_CheckFIFOFull()

■ 範例程式

FIFOChk.cpp

■ 內容說明

本範例程式將讀取 DDA Channel 0 的 FIFO 狀態。

5. DDA 脈波輸出控制及使用 FIFO 中斷功能

■ 相關函式

EPCIO_LIO_EnablePulseDAC()
EPCIO_DDA_SetTime()
EPCIO_DDA_SetOutputFormat()
EPCIO_DDA_SetMinStockNo()
EPCIO_DDA_SetPulseWidth()
EPCIO_DDA_EnableOutputChannel()
EPCIO_DDA_DisableOutputChannel()
EPCIO_DDA_EnableStockInt()
EPCIO_DDA_DisableStockInt()
EPCIO_DDA_StartEngine()
EPCIO_DDA_StopEngine()
EPCIO_DDA_SendPulse()

■ 範例程式

FIFOCtrl.cpp

■ 內容說明

本範例程式將從 Channel 0 輸出脈波，並規劃 DDA FIFO 中斷，當 FIFO 命令筆數小於所設定之最小命令筆數時將觸發中斷時，將自動呼叫中斷處理函式，而我們將在中斷服務函式中將 FIFO 內之命令增加至最小安全值(筆數)。



6. DDA 脈波輸出計錄器控制

■ 相關函式

EPCIO_LIO_EnablePulseDAC()
EPCIO_DDA_ClearPulseCounter()
EPCIO_DDA_EnablePulseCounter()
EPCIO_DDA_DisablePulseCounter()
EPCIO_DDA_SetOutputFormat()
EPCIO_DDA_EnableOutputChannel()
EPCIO_DDA_DisableOutputChannel()
EPCIO_DDA_StartEngine()
EPCIO_DDA_StopEngine()
EPCIO_DDA_SendPulse()
EPCIO_DDA_GetOutputPulse()

■ 範例程式

PulseCnt.cpp

■ 內容說明

本範例程式將從 Channel 0 輸出脈波，並從 EPCIO 內部的脈波計數器讀回實際輸出的脈波數。



7. 在輸出脈波過程時進行緊急停止控制

■ 相關函式

EPCIO_LIO_EnablePulseDAC()
EPCIO_DDA_SetOutputFormat()
EPCIO_DDA_EnableOutputChannel()
EPCIO_DDA_DisableOutputChannel()
EPCIO_DDA_StartEngine()
EPCIO_DDA_StopEngine()
EPCIO_DDA_SendPulse()
EPCIO_DDA_EnableEmgcStop()

■ 範例程式

EmgcStop.cpp

■ 內容說明

本範例程式將從 Channel 0 輸出脈波，並規劃緊急停止功能。



8. DDA 脈波輸出控制

■ 相關函式

EPCIO_LIO_EnablePulseDAC()
EPCIO_DDA_SetOutputFormat()
EPCIO_DDA_EnableOutputChannel()
EPCIO_DDA_DisableOutputChannel()
EPCIO_DDA_StartEngine()
EPCIO_DDA_StopEngine()
EPCIO_DDA_SendPulse()

■ 範例程式

PulseOut.cpp

■ 內容說明

本範例程式將從 Channel 0 輸出脈波。



9. DDA FIFO 內命令筆數讀取控制

■ 相關函式

EPCIO_LIO_EnablePulseDAC()
EPCIO_DDA_SetOutputFormat()
EPCIO_DDA_EnableOutputChannel()
EPCIO_DDA_DisableOutputChannel()
EPCIO_DDA_StartEngine()
EPCIO_DDA_StopEngine()
EPCIO_DDA_SendPulse()
EPCIO_DDA_GetStockCount()

■ 範例程式

StockChk.cpp

■ 內容說明

本範例程式讀取 DDA Channel 0 的 FIFO 內所儲存，但尚未執行之命令筆數。

10. DDA FIFO 中斷功能

■ 相關函式

EPCIO_LIO_EnablePulseDAC()
EPCIO_DDA_SetOutputFormat()
EPCIO_DDA_EnableOutputChannel()
EPCIO_DDA_DisableOutputChannel()
EPCIO_DDA_SetMinStockNo()
EPCIO_DDA_EnableStockInt()
EPCIO_DDA_DisableStockInt()
EPCIO_DDA_StartEngine()
EPCIO_DDA_StopEngine()
EPCIO_DDA_SendPulse()

■ 範例程式

StockInt.cpp

■ 內容說明

本範例程式開啟 DDA Channel 0 的 FIFO 中斷功能，當 Channel 0 的 FIFO 內之命令筆數等於所設定之最小命令筆數(0 筆)時將觸發中斷，並自動呼叫中斷服務函式。本功能類似 DDA Cycle 中斷，但 DDA Cycle 中斷具有固定的發生頻率，而 DDA FIFO 中斷則需在 FIFO 內之命令筆數符合設定條件時才會產生觸發中斷。

11. 開啟編碼器 INDEX (Z Phase)觸發中斷功能

■ 相關函式

EPCIO_ENC_SetFilterClock()
EPCIO_ENC_SetInputType()
EPCIO_ENC_SetInputRate()
EPCIO_ENC_EnableIndexInt()
EPCIO_ENC_DisableIndexInt()
EPCIO_ENC_ClearCounter()
EPCIO_ENC_StartInput()
EPCIO_ENC_StopInput()
EPCIO_ENC_GetValue()

■ 範例程式

ENCIndex.cpp

■ 內容說明

本範例程式規劃 Channel 0 編碼器輸入，並設定編碼器之 INDEX 中斷功能，當編碼器之 INDEX(Z phase)輸入時，將觸發中斷。

12. 開啟編碼器比較值中斷功能

■ 相關函式

EPCIO_ENC_SetFilterClock()
EPCIO_ENC_SetInputType()
EPCIO_ENC_SetInputRate()
EPCIO_ENC_SetCompValue()
EPCIO_ENC_EnableCompInt()
EPCIO_ENC_DisableCompInt()
EPCIO_ENC_ClearCounter()
EPCIO_ENC_StartInput()
EPCIO_ENC_StopInput()
EPCIO_ENC_GetValue()

■ 範例程式

ENCComp.cpp

■ 內容說明

本範例程式規劃 Channel 0 編碼器輸入功能，並開啟編碼器比較值中斷功能，當編碼計數器的計數值等於比較值時將觸發中斷。



13. 開啟編碼器輸入功能

■ 相關函式

EPCIO_ENC_SetFilterClock()

EPCIO_ENC_SetInputType()

EPCIO_ENC_SetInputRate()

EPCIO_ENC_ClearCounter()

EPCIO_ENC_StartInput()

EPCIO_ENC_StopInput()

EPCIO_ENC_GetValue()

■ 範例程式

ENCCTRL.cpp

■ 內容說明

本範例程式將開啟 Channel 0 的編碼器輸入功能。

14. 開啟編碼器的閃鎖(Latch)功能

■ 相關函式

EPCIO_ENC_SetFilterClock()
EPCIO_ENC_SetInputType()
EPCIO_ENC_SetInputRate()
EPCIO_ENC_SetTrigSource()
EPCIO_ENC_SetTrigMode()
EPCIO_ENC_ClearCounter()
EPCIO_ENC_StartInput()
EPCIO_ENC_StopInput()
EPCIO_ENC_GetLatchValue()

■ 範例程式

ENCLatch.cpp

■ 內容說明

本範例程式規劃 Channel 0 編碼器的輸入設定，並設定 Channel 0 編碼器之 INDEX 為觸發源，當 Channel 0 編碼器之 INDEX(Z phase)輸入時，將觸發編碼器的閃鎖(Latch)功能，此時 Channel 0 編碼器的紀錄值將被記錄在 Channel 0 的閃鎖暫存器內。



15. 開啟編碼器的比較器觸發 DAC 類比電壓輸出

■ 相關函式

EPCIO_DAC_SetClockDivider()
EPCIO_DAC_SetCmdSource()
EPCIO_DAC_EnableTrigMode()
EPCIO_DAC_SetTrigOutput()
EPCIO_DAC_SetTrigSource()
EPCIO_DAC_StartConv()
EPCIO_DAC_StopConv()
EPCIO_ENC_EnableCompInt()
EPCIO_ENC_ClearCounter()
EPCIO_ENC_SetCompValue
EPCIO_ENC_StartInput()
EPCIO_ENC_GetValue()

■ 範例程式

ENCTrig.cpp

■ 內容說明

本範例程式規劃一個編碼計數器 Channel 1 之比較器來觸發 DAC 的 Channel 0，此 DAC 將利用軟體方式輸出預設的電壓 3 V。



16. Timer 計時器中斷功能

■ 相關函式

EPCIO_LIO_SetTimer()

EPCIO_LIO_EnableTimerInt()

EPCIO_LIO_EnableTimer()

EPCIO_LIO_DisableTimerInt()

EPCIO_LIO_DisableTimer()

■ 範例程式

Timer.cpp

■ 內容說明

本範例程式規劃 Timer 計時器功能，Timer 計時器將於一固定時間觸發中斷。

17. ADC 比較器中斷功能控制

■ 相關函式

EPCIO_ADC_SetConvMode()
EPCIO_ADC_SetConvType()
EPCIO_ADC_SetCompValue()
EPCIO_ADC_SetCompType()
EPCIO_ADC_EnableCompInt()
EPCIO_ADC_DisableCompInt()
EPCIO_ADC_EnableConvChannel()
EPCIO_ADC_StartConv()
EPCIO_ADC_StopConv()

■ 範例程式

ADCComp.cpp

■ 內容說明

本範例程式設定 ADC Channel 0 的比較器之比較值，當比較條件成立且電壓由高到低時將觸發中斷。本程式將連續進行 ADC 轉換，也就是當比較條件成立時中斷將被連續觸發。



18. ADC 電壓輸入(連續雙極性模式)讀取控制

■ 範例程式

ADCInput.cpp

■ 內容說明

本範例程式規劃 ADC 的 Channel 0 進行連續的雙極性(Bipolar) 電壓轉換模式，並讀取輸入的電壓值(-5 ~ 5 V)。



19. ADC 電壓輸入與讀取工作狀態

■ 相關函式

EPCIO_ADC_SetConvMode()
EPCIO_ADC_EnableConvChannel()
EPCIO_ADC_DisableConvChannel()
EPCIO_ADC_GetWorkStatus()
EPCIO_ADC_StartConv()
EPCIO_ADC_StopConv()

■ 範例程式

ADCState.cpp

■ 內容說明

本範例程式將規劃 ADC 的 Channel 0 進行連續電壓轉換，並讀取 ADC 的工作狀態。



20. 開啟 ADC 標籤 Channel 中斷功能

■ 相關函式

EPCIO_ADC_SetConvMode()
EPCIO_ADC_SetConvType()
EPCIO_ADC_EnableConvChannel()
EPCIO_ADC_DisableConvChannel()
EPCIO_ADC_SetTagChannel()
EPCIO_ADC_EnableTagInt()
EPCIO_ADC_DisableTagInt()
EPCIO_ADC_StartConv()
EPCIO_ADC_StopConv()

■ 範例程式

ADCTag.cpp

■ 內容說明

本範例程式設定 ADC 的 Channel 0 為標籤 Channel，在標籤 Channel 完成電壓轉換後將觸發中斷。本程式將連續進行 ADC 轉換，因此當完成電壓轉換時中斷將被連續觸發。

21. ADC 輸入電壓比較器觸發 DAC 類比電壓輸出

■ 相關函式

EPCIO_DAC_SetCmdSource()
EPCIO_DAC_SetTrigOutput()
EPCIO_DAC_SetTrigSource()
EPCIO_DAC_EnableTrigMode()
EPCIO_DAC_StartConv()
EPCIO_DAC_SetOutput()
EPCIO_ADC_SetConvMode()
EPCIO_ADC_SetConvType()
EPCIO_ADC_SetCompValue()
EPCIO_ADC_SetCompType()
EPCIO_ADC_EnableCompInt
EPCIO_ADC_EnableConvChannel()
EPCIO_ADC_StartConv()
EPCIO_ADC_GetInput()

■ 範例程式

ADCTrig.cpp

■ 內容說明

本範例程式規劃一個 ADC 比較器來觸發 DAC，此 DAC 將利用軟體方式輸出預設的電壓 5 V。

22. ADC 電壓輸入(單次雙極性模式)讀取控制

■ 相關函式

EPCIO_ADC_SetConvMode()
EPCIO_ADC_SetConvType()
EPCIO_ADC_SetSingleChannel()
EPCIO_ADC_StartConv()
EPCIO_ADC_StopConv()
EPCIO_ADC_GetInput()

■ 範例程式

ADC1Time.cpp

■ 內容說明

本範例程式規劃 ADC 的 Channel 0 進行單次的雙極性(Bipolar) 電壓轉換模式，並讀取輸入的電壓值(-5 ~ 5 V)。



23. DAC 類比電壓輸出

■ 相關函式

EPCIO_DAC_SetClockDivider()

EPCIO_DAC_SetCmdSource()

EPCIO_DAC_StartConv()

EPCIO_DAC_StopConv()

EPCIO_DAC_SetOutput()

■ 範例程式

DACDirWR.cpp

■ 內容說明

本範例程式說明如何規劃 DAC 類比電壓輸出。



24. LIO 近端輸入點觸發 DAC 類比電壓輸出

■ 相關函式

EPCIO_DAC_SetCmdSource()
EPCIO_DAC_EnableTrigMode()
EPCIO_DAC_SetTrigOutput()
EPCIO_DAC_SetTrigSource()
EPCIO_DAC_StartConv()
EPCIO_DAC_SetOutput()
EPCIO_LIO_SetLDIIIntType
EPCIO_LIO_EnableLDIIInt
EPCIO_LIO_DisableLDIIInt
EPCIO_LIO_GetLDIIInput

■ 範例程式

LIOTrig.cpp

■ 內容說明

本範例程式規劃 DAC 由軟體方式輸出類比電壓，並利用一個近端輸入點觸發 DAC 輸出類比電壓 6 V。

25. RIO 遠端輸入點讀取及中斷功能

■ 相關函式

EPCIO_RIO_SetIntType()
EPCIO_RIO_EnableInputInt()
EPCIO_RIO_DisableInputInt()
EPCIO_RIO_EnableSetControl()
EPCIO_RIO_DisableSetControl()
EPCIO_RIO_EnableSlaveControl()
EPCIO_RIO_DisableSlaveControl()

■ 範例程式

RIOInput.cpp

■ 內容說明

本範例程式將讀取 Remote I/O 第 0 個 Set 的第 0 個 Slave 的輸入點訊號，並利用輸入點的中斷功能。當第 0 個輸入點的輸入由 LOW 變為 HIGH 時將觸發中斷，並自動呼叫中斷服務函式。



26. RIO 遠端輸出入點的讀取與寫出控制

■ 相關函式

EPCIO_RIO_EnableSetControl()
EPCIO_RIO_DisableSetControl()
EPCIO_RIO_EnableSlaveControl()
EPCIO_RIO_DisableSlaveControl()
EPCIO_RIO_GetInputValue()
EPCIO_RIO_SetOutputValue()

■ 範例程式

RIOCtrl.cpp

■ 內容說明

本範例程式將規劃 Remote I/O 第 0 個 Set 的第 0 個 Slave 的讀寫控制。



27. RIO 傳輸錯誤中斷控制

■ 相關函式

EPCIO_RIO_SetTransError()
EPCIO_RIO_EnableInputInt()
EPCIO_RIO_DisableInputInt()
EPCIO_RIO_EnableSetControl()
EPCIO_RIO_DisableSetControl()
EPCIO_RIO_EnableSlaveControl()
EPCIO_RIO_DisableSlaveControl()

■ 範例程式

RIOError.cpp

■ 內容說明

本範例程式將規劃 Remote I/O 第 0 個 Set 的第 0 個 Slave 的傳輸，並開啟傳輸錯誤中斷功能。當傳輸發生錯誤時將觸發中斷，並自動呼叫中斷服務函式。

28. RIO 傳輸狀態讀取控制

■ 相關函式

EPCIO_RIO_EnableSetControl()
EPCIO_RIO_DisableSetControl()
EPCIO_RIO_EnableSlaveControl()
EPCIO_RIO_DisableSlaveControl()
EPCIO_RIO_GetTransStatus()
EPCIO_RIO_GetMasterStatus()
EPCIO_RIO_GetSlaveStatus()

■ 範例程式

RIOState.cpp

■ 內容說明

本範例程式規劃 Remote I/O 第 0 個 Set 的第 0 個 Slave 的傳輸動作，並讀取目前的傳輸狀態。

29. RIO 遠端輸入點觸發 DAC 類比電壓輸出

■ 相關函式

EPCIO_DAC_SetCmdSource()
EPCIO_DAC_EnableTrigMode()
EPCIO_DAC_SetTrigOutput()
EPCIO_DAC_SetTrigSource()
EPCIO_DAC_StartConv()
EPCIO_LIO_EnablePulseDAC()
EPCIO_RIO_SetIntType()
EPCIO_RIO_EnableInputInt()
EPCIO_RIO_EnableSetControl()
EPCIO_RIO_EnableSlaveControl()
EPCIO_DAC_StartConv()
EPCIO_DAC_SetOutput()

■ 範例程式

RIOTrig.cpp

■ 內容說明

本範例程式規劃 DAC 由軟體方式輸出類比電壓，此輸出可由 RIO 第 0 個 Set 的第 0 個 Slave 的輸入點加以觸發。



30. PCL 硬體位置閉迴路控制功能

■ 相關函式

EPCIO_DAC_SetCmdSource()
EPCIO_DAC_StartConv()
EPCIO_ENC_ClearCounter()
EPCIO_ENC_StartInput()
EPCIO_PCL_ClearErrorCounter()
EPCIO_PCL_SetScaleGain()
EPCIO_PCL_EnableErrorCounter()
EPCIO_PCL_StartControl()
EPCIO_DDA_EnableOutputChannel()
EPCIO_DDA_StartEngine()
EPCIO_LIO_EnablePulseDAC()
EPCIO_DDA_SendPulse()
EPCIO_DDA_GetCurrentCmd()
EPCIO_ENC_GetValue()
EPCIO_PCL_GetErrorCounter()

■ 範例程式

PCLCtrl.cpp

■ 內容說明

本範例程式將規劃 Channel 0 進行硬體位置閉迴路 PCL(Position Cloded-Loop)控制。

31. 硬體位置閉迴路 PCL 控制及使用 DDA Cycle 中斷功能

■ 相關函式

EPCIO_DAC_SetCmdSource()
EPCIO_DAC_StartConv()
EPCIO_ENC_ClearCounter()
EPCIO_ENC_StartInput()
EPCIO_PCL_ClearErrorCounter()
EPCIO_PCL_SetScaleGain()
EPCIO_PCL_EnableErrorCounter()
EPCIO_PCL_StartControl()
EPCIO_DDA_EnableOutputChannel()
EPCIO_DDA_EnableCycleInt()
EPCIO_LIO_EnablePulseDAC()
EPCIO_DDA_SendPulse()
EPCIO_DDA_GetCurrentCmd()
EPCIO_ENC_GetValue()
EPCIO_PCL_GetErrorCounter()

■ 範例程式

PCL_DDA.cpp

■ 內容說明

本範例程式將規劃 Channel 0 進行硬體位置閉迴路 PCL(Position Cloded-Loop)控制，命令輸出方式則利用 DDA Cycle 中斷，並且在中斷服務函式中將命令補至所下達之位置。