

ECM-XF

EtherCAT主站晶片 使用手冊

訊成科技股份有限公司

2024/05/22

Ver.049

簡介	5
介面	5
SPI模式	6
封包格式	8
標頭(Header)	9
命令碼列表	13
ECM_CMD_INFO_UPDATE_OP	17
ECM_CMD_ECAT_INIT_OP	18
ECM_CMD_ECAT_RECONFIG_OP	19
ECM_CMD_ECAT_INIT_DC_OP	19
ECM_CMD_ECAT_PDO_WC_GET	19
ECM_CMD_ECAT_PDO_DATA_FIFO_OP	20
ECM_CMD_ECAT_PDO_DATA_OP	20
ECM_CMD_ECAT_PDO_CONFIG_SET	21
ECM_CMD_ECAT_PDO_CONFIG_REQ	23
ECM_CMD_ECAT_PDO_CONFIG_GET	23
ECM_CMD_ECAT_SDO_REQ	25
ECM_CMD_ECAT_SDO_GET	26
ECM_CMD_ECAT_STATE_SET	27
ECM_CMD_ECAT_STATE_GET	27
ECM_CMD_ECAT_SLV_INFO_GET	28
ECM_CMD_ECAT_SLV_CNT_GET	29
ECM_CMD_FIFO_ENABLE	29
ECM_CMD_FIFO_PACK_SIZE_GET	29
ECM_CMD_SPI_PACK_SIZE_GET	30
ECM_CMD_SPI_RECONFIG_OP	30
ECM_CMD_CRC_ERR_CNT_CLR	31
ECM_CMD_CRC_TYPE_SET	31
ECM_CMD_402_CONFIG_SET	32
ECM_CMD_402_STATE_SET	32
ECM_CMD_402_STATE_GET	33
ECM_CMD_402_CTL_SET	33
ECM_CMD_402_CTL_GET	34
ECM_GPIO_CONFIG_SET(ECM_GpioSetMode)	34
ECM_GPIO_CONFIG_SET(ECM_GpioSetMode)	35
ECM_GPIO_CONFIG_SET(ECM_GpioEnableDebounce)	35
ECM_GPIO_CONFIG_SET(ECM_GpioEnableDebounce)	36
ECM_GPIO_CONFIG_SET(ECM_GpioSetDebounceClock)	36

ECM_GPIO_CONFIG_SET(ECM_GpioIntEnable)	37
ECM_GPIO_CONFIG_SET(ECM_GpioIntClear)	37
ECM_GPIO_CONFIG_SET(ECM_GpioIntClear)	37
ECM_GPIO_FUNC_OP(ECM_GpioSetValue)	38
ECM_GPIO_FUNC_OP(ECM_GpioExtSetValue)	38
ECM_GPIO_FUNC_OP(ECM_GpioGetValue)	39
ECM_GPIO_FUNC_OP(ECM_GpioExtGetValue)	39
ECM_GPIO_FUNC_OP(ECM_GpioGetIntFlag)	40
ECM_GPIO_FUNC_OP(ECM_GpioExtGetIntFlag)	40
ECM_QEI_FUNC_OP(ECM_EncOpen)	41
ECM_QEI_FUNC_OP(ECM_EncStart)	41
ECM_QEI_FUNC_OP(ECM_EncStop)	41
ECM_QEI_FUNC_OP(ECM_EncGetCount)	41
ECM_DAC_FUNC_OP(ECM_DacOpen)	42
ECM_DAC_FUNC_OP(ECM_DacClose)	42
ECM_DAC_FUNC_OP(ECM_DacSetDelayTime)	43
ECM_DAC_FUNC_OP(ECM_DacSetData)	43
ECM_DAC_FUNC_OP(ECM_DacStartConv)	43
ECM_ADC_FUNC_OP(ECM_AdcOpen)	44
ECM_ADC_FUNC_OP(ECM_AdcClose)	44
ECM_ADC_FUNC_OP(ECM_AdcConfigSampleModule)	44
ECM_ADC_FUNC_OP(ECM_AdcGetDataValidFlag)	45
ECM_ADC_FUNC_OP(ECM_AdcStartConv)	45
ECM_ADC_FUNC_OP(ECM_AdcGetConvData)	46
ECM_EEPROM_REQ	46
ECM_EEPROM_GET	47
ECM_CMD_ECAT_STATE_CHECK	48
ECM_CMD_ECAT_DCSYNC	48
ECM_CMD_FIFO_CLR_OP	49
ECM_CMD_FIFO_SET_TX_CNT	49
ECM_CMD_FIFO_GET_TX_CNT	49
ECM_CMD_FIFO_SET_RX_CNT	50
ECM_CMD_FIFO_GET_RX_CNT	50
ECM_CMD_FW_VERSION_GET	50
ECM_CMD_ECAT_STATE_UPDATE	51
ECM_CMD_ECAT_INT_SET_ENABLE	51
ECM_CMD_ECAT_INT_GET_ENABLE	52
ECM_CMD_FIFO_INIT	54
ECM_CMD_ECAT_SDO_ABORTCODE_GET	54
ECM_CMD_ECAT_CONFIG_SM	56

ECM_CMD_ECAT_CONFIG_FMMU	56
ECM_CMD_ECAT_CONFIG_MAP	57
ECM_CMD_402_GET_STATUSWORD	57
ECM_CMD_ASYNC_CMD_RESET	58
ECM_CMD_MDIO	58
ECM_CMD_DIRECT_ASSIGN	59
ECM_CMD_COE_EMERGENCY	60
ECM_CMD_WARM_RESET	60
ECM_CMD_ECAT_WKC_CONTI_ERR_CNT_GET	61
ECM_CMD_ECAT_WKC_CONTI_ERR_MAX_GET	61
ECM_CMD_ECAT_WKC_CONTI_ERR_MAX_SET	62
ECM_CMD_RAW_ECAT_FUNC_REQ	62
ECM_CMD_RAW_ECAT_FUNC_GET	63
ECM_CMD_FOE_FUNC_REQ	64
ECM_CMD_FOE_FUNC_GET	65
ECM_CMD_ENABLE_LRW	65
ECM_CMD_RECV_MAILBOX	66
ECM_CMD_MBX_STAT	67
ECM-XF 命令及回應收送流程說明	68
ECM-XF 初始化EtherCAT網路及從站配置說明	69
GPIO	71
ECM-XF中斷	71
附錄 CiA 402 狀態機 CiA 402 State Machine	72
更新紀錄	77

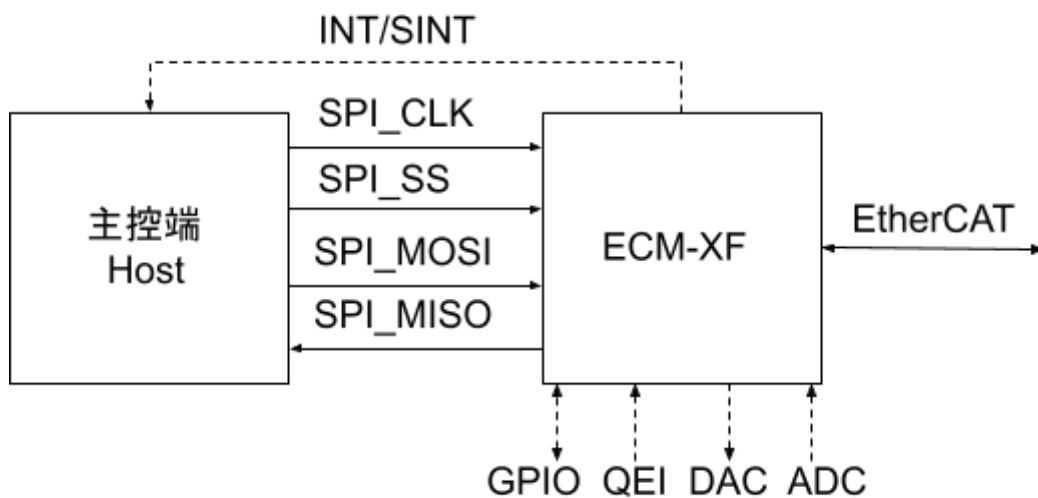
簡介

ECM-XF 為一高性價比EtherCAT主站晶片，透過SPI介面與主控端連接，協助使用者實現EtherCAT通訊，達到最小DC週期時間125us及最多連接128從站的能力，具備多種周邊IO、中斷及應用功能，適合PLC控制器、機器人控制器及各式自動化控制器使用。

ECM-XF可透過USB進行韌體升級，若要使用USB控制EtherCAT從站裝置，則需改用ECM-XFU晶片。

支援DC功能	Yes
最大從站數	128
最小週期時間	125us
工業乙太網	EtherCAT
郵箱通訊(mailbox protocol)	CoE / FoE
周邊IO	GPIO / QEI / ADC / DAC
應用功能	CiA402狀態機控制 FIFO緩衝

介面



圖一、ECM-XF介面連接示意圖

* PHY Address需設置為1

SPI模式

名稱	腳位	意義	說明
SPI_CLK	Pin38	頻率訊號	由 SPI Master 產生並控制, 依控制EtherCAT通訊週期有最低頻率要求
SPI_MOSI	Pin40	主出從入	SPI Master資料輸出, SPI Slave資料接收
SPI_MISO	Pin39	主入從出	SPI Master資料接收, SPI Slave資料輸出
SPI_SS	Pin37	晶片致能	選擇信號, 由Master控制, Slave只有在SPI_SS 信號為低電位時, 才會對 Master 的操作指令有反應

雙工模式	全雙工(full duplex)		
傳輸率	96Mbps(max.)		
時序模式	SPI_CLK閒置低電位, 上升緣鎖存資料, 下降緣發送資料 (Mode 0)		
	POL=0	PHA=0	
	POL=0	TXNEG=1	RXNEG=0

SPI 模式傳輸說明

ECM的SPI為Slave模式, 空閒時為低電位, 在下降緣發送, 並於上升緣接收, 高位資料先傳 (MSB), 請參考下圖說明。

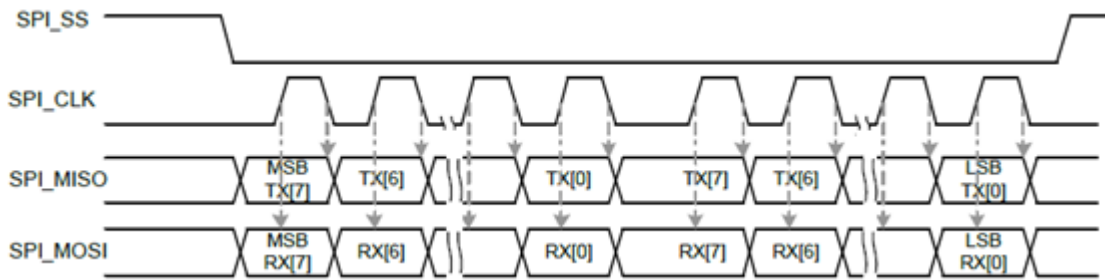
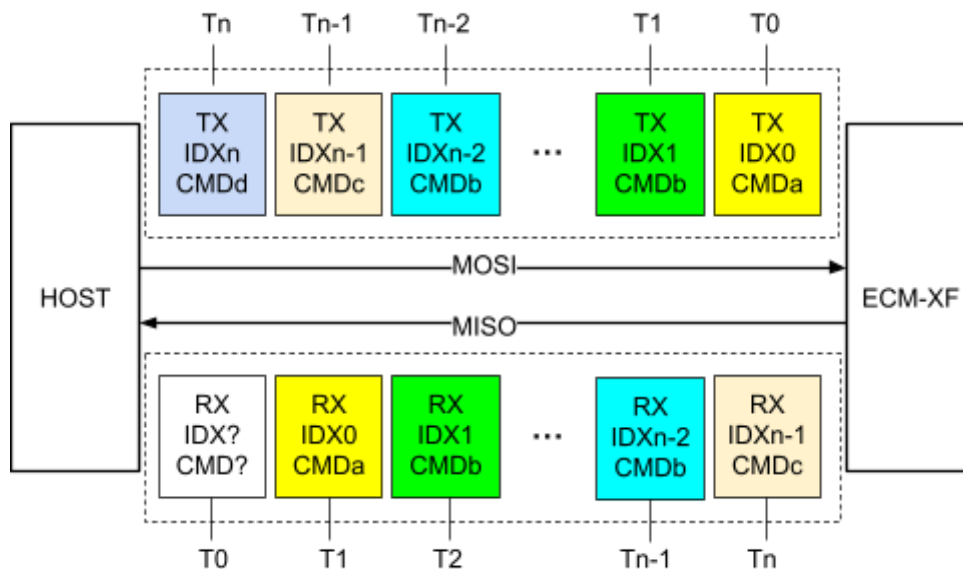


圖 SPI時序圖

上位控制器端的SPI為Master模式，需產生CLK並提供給SPI Slave，並在上升緣發送，並於下降緣接收。SPI傳輸以Byte為單位，每次將由低地址開始傳，依序傳至最高位址，換言之SPI傳輸將從Byte0開始，再依序傳Byte1、Byte2...直至最後一個Byte為止。而SPI傳輸單一Byte時，採取MSB模式，亦即高位先傳輸。

SPI 命令傳送與回應接收說明



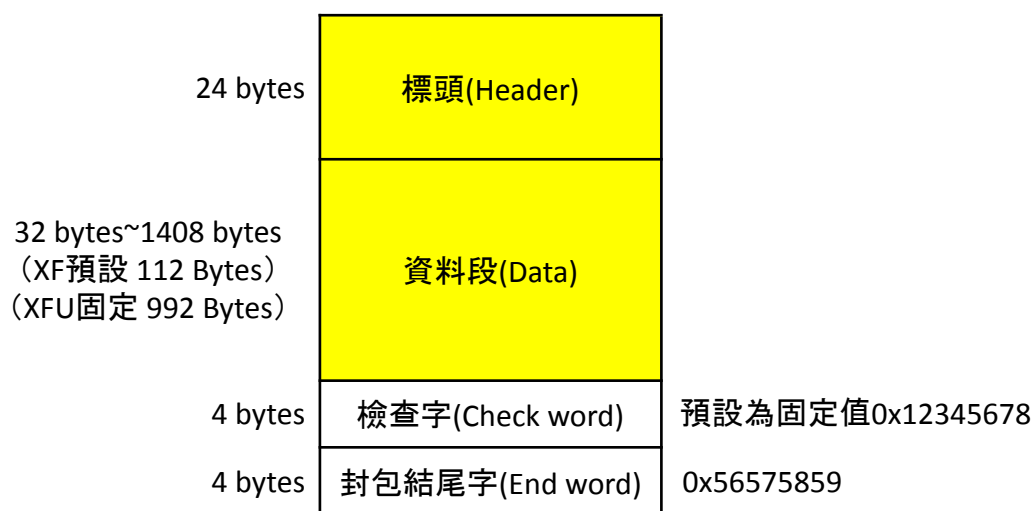
圖二，SPI全雙工資料交換封包排序示意圖

注意

1. 主控端透過SPI將命令封包傳送完成後，會接收到上一次命令的回應封包
2. ECM-XF啟動後，第一個回傳給主控端的回應封包，因無上一次命令故不包含命令回應。
3. 若連續命令封包的索引號及命令碼一致，除了第一筆命令ECM-XF會視為有效命令外，其餘命令皆視為ECM_CMD_INFO_UPDATE_OP命令。

封包格式

1. 封包由四個部分組成 :標頭、資料段、檢查字、封包結尾字
2. 命令標頭包含命令、參數資料及系統相關控制...等
3. 回應標頭包含命令返回值、返回資料、錯誤狀態及系統狀態...等
4. 資料段內容為命令相關資料或PDO資料 使用可依實際需求配置足夠大小
5. 檢查字有兩種格式提供使用者選擇(固定值0x12345678檢查、CRC-32)
6. 封包結尾字為一固定值(0x56575859), 用來識別封包完整性
7. 資料段長度32bytes~1408bytes, 預設為112bytes



圖三 SPI封包格式

標頭(Header)

命令標頭(Command Header) : 主控端傳送給ECM-XF封包的標頭

封包起始字 0xA1A2A3A4				0x00
命令資料3	命令資料2	命令資料1	命令資料0	0x04
索引號	控制字元	保留		0x08
有效資料段長度		命令參數	命令碼	0x0c
中斷清除				0x10
GPIO輸出1	GPIO輸出0	GPIO中斷清除1	GPIO中斷清除0	0x14

圖四 命令標頭

命令碼關聯
 中斷相關
 系統或硬體相關

回應標頭(Response Header) : ECM-XF回傳給主控端封包的標頭

封包起始字 0xA1A2A3A4				0x00
RxPDO待傳送數	TxPDO待接收數	WKC錯誤次數	CRC錯誤次數	0x04
回應索引號	ECM狀態	ECM錯誤狀態	ECAT狀態	0x08
有效資料段長度		返回值	回應命令碼	0x0c
中斷旗幟				0x10
GPIO輸入1	GPIO輸入0	GPIO中斷旗幟1	GPIO中斷旗幟0	0x14

圖五 回應標頭

命令碼關聯
 中斷相關
 系統或硬體相關
 錯誤狀態

標頭欄位說明

表一 命令封包標頭欄位列表

命令標頭			
位置	長度	名稱	說明
0x00	4	封包起始字	固定值 0xA1A2A3A4
0x04	1	命令資料0	傳遞與命令相關參數或資料
0x05	1	命令資料1	傳遞與命令相關參數或資料
0x06	1	命令資料2	傳遞與命令相關參數或資料
0x07	1	命令資料3	傳遞與命令相關參數或資料
0x08	2	保留	bit 15: 1代表標頭檔的GPIO輸出0(0x16)、GPIO輸出1(0x17)有效
0x0A	1	控制字元	控制或清除系統錯誤 bit0: 標頭IO輸出欄位有效 bit1: 自從站更新ECAT狀態 (非即時命令) bit3: 非即時命令錯誤清除 bit4: SPI通訊錯誤清除 bit5: FIFO錯誤清除 bit6: SPI CRC錯誤清除 bit7: 命令錯誤清除
0x0B	1	索引號	主要用途為識別用
0x0C	1	命令碼	請參考命令列表
0x0D	1	命令參數	命令碼相關參數
0x0E	2	有效資料段長度	命令碼相關參數 或 有效資料段長度
0x10	4	中斷清除	Bit 31: EtherCAT資料包中斷清除 Bit 29: 編碼器中斷清除 Bit 0 ~ 28, 30: 保留
0x14	1	GPIO中斷清除0	清除GPIO低位元中斷旗幟
0x15	1	GPIO中斷清除1	清除GPIO高位元中斷旗幟
0x16	1	GPIO輸出0	GPIO低位元輸出值 (GPIO 0~7) (0x08 bit15須為1)
0x17	1	GPIO輸出1	GPIO高位元輸出值 (GPIO 8~11) (0x08 bit15須為1)

- 若要控制GPIO 12 ~15 請使用CMD ID 32 (ECM_GPIO_FUNC_OP)
- 版本0x1E之後可控制bit12~15, 但bit12控制GPIO13, bit13控制GPIO12

表二 回應封包標頭欄位列表

回應標頭			
位置	長度	名稱	說明
0x00	4	封包起始字	固定值 0xA1A2A3A4
0x04	1	CRC錯誤次數	SPI通訊CRC錯誤次數
0x05	1	WKC錯誤次數	PDO通訊working count錯誤次數
0x06	1	TxPdoFIFO數量	FIFO內尚未讀取TxPDO數量
0x07	1	RxPdoFifo數量	FIFO內尚未傳送RxPDO數量
0x08	1	ECAT狀態	0x00 : NONE(未初始化狀態) 0x01 : INIT(初始化狀態) 0x02 : PRE_OP(預操作狀態) 0x03 : BOOT(啟動狀態) 0x04 : SAFE_OP(安全操作狀態) 0x08 : OPERATIONAL(操作狀態) 0x10 : ERROR(錯誤狀態)
0x09	1	錯誤/狀態	bit0: 檢查網口與ECM-XF PIN 52 LINK連接狀態 (連線時值為1或0視硬體電路設計) bit1: 軟體於Ecat_Init時確認PHY連線狀態 (連線時值為1或0視硬體電路設計) bit2: 保留 bit3: 非即時命令返回錯誤 bit4: SPI通訊錯誤 bit5: FIFO錯誤 bit6: SPI CRC錯誤 bit7: 命令錯誤 註: bit 7 為1代表命令錯誤包含, 該命令所帶參數錯誤, 或於非即時命令忙碌狀態時收到新的非即時命令
0x0A	1	ECM狀態	bit1-bit0 : 0 : 固定值檢查 1: 未定義 2: 未定義 3: CRC32檢查 bit2 : DC週期時間穩定旗幟 bit3 : 網口線路初始化完成旗幟

			bit4 : ECAT PDO配置完成旗幟 bit5 : 回應封包為NOP旗幟 bit6 : FIFO啟動旗幟 bit7 : 非即時命令忙碌旗幟
0x0B	1	索引號	回應命令標頭索引號
0x0C	1	命令碼	回應命令標頭命令碼
0x0D	1	命令返回值	回應命令的執行返回值
0x0E	2	資料段長度	資料段內有效資料長度
0x10	4	中斷旗幟	Bit 31: EtherCAT資料包中斷旗幟 Bit 29: 編碼器中斷旗幟 Bit 0 ~ 28, 30: 保留
0x14	1	GPIO中斷旗幟0	GPIO低位元中斷旗幟
0x15	1	GPIO中斷旗幟1	GPIO高位元中斷旗幟
0x16	1	GPIO輸入0	GPIO低位元輸入值
0x17	1	GPIO輸入1	GPIO高位元輸入值

命令碼列表

命令依照回應時間分兩類

1. 即時命令(NON-BLOCKING) - 接受命令後 操作即刻完成並返回
 2. 非即時命令(BLOCKING) - 接受命令後 需等待一定時間才能完成並返回
 3. ECM-XF一次只能執行一個非即時命令 因此當使用非即時命令操作前 需確定ECM狀態非即時命令忙碌旗幟為0 才傳送非即時命令
 4. 非即時命令運作同時 可進行即時命令或其他應用操作
- 請參考圖六 命令及回應收送流程圖

表三 命令碼列表

非即時命令	即時命令
ECM_CMD_ECAT_INIT_OP ECM_CMD_ECAT_RECONFIG_OP ECM_CMD_ECAT_DCSYNC ECM_CMD_ECAT_PDO_CONFIG_SET ECM_CMD_ECAT_STATE_SET	ECM_CMD_INFO_UPDATE_OP ECM_CMD_FW_VERSION_GET ECM_CMD_ECAT_PDO_DATA_FIFO_OP ECM_CMD_ECAT_PDO_DATA_OP ECM_CMD_ECAT_STATE_GET ECM_CMD_ECAT_PDO_WC_GET ECM_CMD_ECAT_SLV_INFO_GET ECM_CMD_ECAT_SLV_CNT_GET ECM_CMD_FIFO_ENABLE ECM_CMD_FIFO_PACK_SIZE_GET ECM_CMD_SPI_PACK_SIZE_GET ECM_CMD_SPI_RECONFIG_OP ECM_CMD_CRC_ERR_CNT_CLR ECM_CMD_CRC_TYPE_SET ECM_CMD_402_CONFIG_SET ECM_CMD_402_STATE_SET ECM_CMD_402_STATE_GET ECM_CMD_402_CTL_SET ECM_CMD_402_CTL_GET
ECM_CMD_ECAT_PDO_CONFIG_REQ ECM_CMD_ECAT_SDO_REQ ECM_EEPROM_REQ	ECM_CMD_ECAT_PDO_CONFIG_GET ECM_CMD_ECAT_SDO_GET ECM_EEPROM_GET

表四 命令說明列表

ECM_CMD_INFO_UPDATE_OP	更新標頭資料
ECM_CMD_ECAT_INIT_OP	初始化EtherCAT網路及從站, 使用EtherCAT功能前必須先初始化EtherCAT網路

ECM_CMD_ECAT_RECONFIG_OP	當重新規劃PDO後, 需使用此命令使ECM-XF重新配置記憶體空間
ECM_CMD_ECAT_PDO_WC_GET	讀取PDO中具有輸入資料從站數量、具有輸出資料從站數量或目前Working counter值
ECM_CMD_ECAT_PDO_DATA_FIFO_OP	對FIFO存取PDO資料
ECM_CMD_ECAT_PDO_DATA_OP	讀取PDO資料
ECM_CMD_ECAT_PDO_CONFIG_SET	配置從站PDO
ECM_CMD_ECAT_PDO_CONFIG_REQ	請求執行PDO配置讀取, 操作完成後使用ECM_CMD_ECAT_PDO_CONFIG_GET取回配置
ECM_CMD_ECAT_PDO_CONFIG_GET	讀取從站PDO配置
ECM_CMD_ECAT_SDO_REQ	請求執行SDO讀寫命令, 寫操作指令資料段為寫入資料, 讀操作完成後透過ECM_CMD_ECAT_SDO_GET指令取回讀取資料
ECM_CMD_ECAT_SDO_GET	讀回ECM_CMD_ECAT_SDO_REQ讀操作結果
ECM_CMD_ECAT_STATE_SET	變更EtherCAT狀態
ECM_CMD_ECAT_STATE_GET	取回EtherCAT狀態
ECM_CMD_ECAT_SLV_INFO_GET	讀取從站資訊
ECM_CMD_ECAT_SLV_CNT_GET	讀取從站數量
ECM_CMD_FIFO_ENABLE	啟動RxPDO FIFO輸出, ECM-XF預設啟動RxPDO FIFO輸出, 使用者可以透過此命令關閉RxPDO FIFO週期輸出, 可用於預儲存輸出。
ECM_CMD_FIFO_PACK_SIZE_GET	讀取PDO資料長度, FIFO內一筆資料為一個PDO
ECM_CMD_SPI_PACK_SIZE_GET	讀取SPI資料長度
ECM_CMD_SPI_RECONFIG_OP	重置SPI資料長度
ECM_CMD_CRC_ERR_CNT_CLR	清除CRC檢查累積錯誤次數
ECM_CMD_CRC_TYPE_SET	設定檢查碼類型
ECM_CMD_402_CONFIG_SET	指定從站控制字及狀態字偏移量

ECM_CMD_402_STATE_SET	切換指定從站402狀態
ECM_CMD_402_STATE_GET	讀取指定從站402狀態
ECM_CMD_402_CTL_SET	ECM-XF內部402狀態機控制位元組設定
ECM_CMD_402_CTL_GET	ECM-XF內部402狀態機控制位元組讀取
ECM_GPIO_CONFIG_SET (ECM_GpioSetMode)	設置GPIO模式
ECM_GPIO_CONFIG_SET (ECM_GpioEnableDebounce)	啟用/停用 GPIO彈跳
ECM_GPIO_CONFIG_SET (ECM_GpioSetDebounceClock)	設置彈跳時脈
ECM_GPIO_CONFIG_SET (ECM_GpioIntEnable)	啟用/停用 GPIO中斷
ECM_GPIO_CONFIG_SET (ECM_GpioIntClear)	清除GPIO中斷旗幟
ECM_GPIO_FUNC_OP (ECM_GpioSetValue)	設置GPIO數值
ECM_GPIO_FUNC_OP (ECM_GpioExtSetValue)	設置GPIO數值 (擴充功能, 第9版以後版本才能使用)
ECM_GPIO_FUNC_OP (ECM_GpioGetValue)	取得GPIO數值
ECM_GPIO_FUNC_OP (ECM_GpioExtGetValue)	取得GPIO數值 (擴充功能, 第9版以後版本才能使用)
ECM_GPIO_FUNC_OP (ECM_GpioGetIntFlag)	取得GPIO中斷旗幟
ECM_GPIO_FUNC_OP (ECM_GpioExtGetIntFlag)	取得GPIO中斷旗幟 (擴充功能, 第9版以後版本才能使用)
ECM_QEI_FUNC_OP (ECM_EncOpen)	開啟並設置編碼器模式
ECM_QEI_FUNC_OP (ECM_EncStart)	開始編碼器計數
ECM_QEI_FUNC_OP (ECM_EncStop)	停止編碼器計數
ECM_QEI_FUNC_OP	取得編碼器計數

(ECM_EncGetCount)	
ECM_DAC_FUNC_OP (ECM_DacOpen)	開啟DAC功能
ECM_DAC_FUNC_OP (ECM_DacClose)	關閉DAC功能
ECM_DAC_FUNC_OP (ECM_DacSetDelayTime)	設置DAC延遲時間
ECM_DAC_FUNC_OP (ECM_DacSetData)	設置DAC資料
ECM_DAC_FUNC_OP (ECM_DacStartConv)	開始DAC轉換
ECM_ADC_FUNC_OP (ECM_AdcOpen)	開啟ADC功能
ECM_ADC_FUNC_OP (ECM_AdcClose)	關閉ADC功能
ECM_ADC_FUNC_OP (ECM_AdcConfigSampleModule)	設置ADC觸發來源
ECM_ADC_FUNC_OP (ECM_AdcGetDataValidFlag)	取得ADC有效資料旗幟
ECM_ADC_FUNC_OP (ECM_AdcStartConv)	開始ADC轉換
ECM_ADC_FUNC_OP (ECM_AdcGetConvData)	取得ADC轉換資料
ECM_EEPROM_REQ	請求執行EEPROM讀寫命令, 寫操作指令資料段為寫入資料, 讀操作完成後透 ECM_EEPROM_GET指令取回讀取資料
ECM_EEPROM_GET	讀回ECM_EEPROM_REQ讀操作結果
ECM_CMD ECAT STATE CHECK	更新並確認從站狀態
ECM_CMD ECAT DCSYNC	設置DC sync信號源、週期時間與偏移時間
ECM_CMD FIFO CLR OP	清除FIFO
ECM_CMD FIFO SET TX CNT	設置Tx FIFO數量
ECM_CMD FIFO GET TX CNT	取得Tx FIFO數量
ECM_CMD FIFO SET RX CNT	設置Rx FIFO數量

ECM_CMD_FIFO_GET_RX_CNT	取得Rx FIFO數量
ECM_CMD_FW_VERSION_GET	讀取韌體版本號
ECM_CMD_ECAT_STATE_UPDATE	更新ECAT狀態
ECM_CMD_ECAT_INT_SET_ENABLE	設置中斷啟用遮罩
ECM_CMD_ECAT_INT_GET_ENABLE	取得中斷啟用遮罩
ECM_CMD_FIFO_INIT	初始化FIFO
ECM_CMD_ECAT_SDO_ABORTCODE_GET	取得SDO Abort Code (SDO錯誤代碼)
ECM_CMD_ECAT_CONFIG_SM	設定Sync Manager
ECM_CMD_ECAT_CONFIG_FMMU	設定FMMU
ECM_CMD_ECAT_CONFIG_MAP	設定PDO映射
ECM_CMD_ASYNC_CMD_RESET	重設非週期命令
ECM_CMD_WARM_RESET	重設ECM-XF
ECM_CMD_ECAT_WKC_CONTI_ERR_MAX_GET	讀取最大工作錯誤數值
ECM_CMD_ECAT_WKC_CONTI_ERR_MAX_SET	設定最大工作錯誤數值

命令說明

本章節主要介紹命令碼相關標頭欄位及資料段內容 請參考圖三 圖四 圖五

1. 黃色為命令碼相關欄位 根據命令碼給定相關參數及資料
2. 索引號用來識別回應封包
3. 連續命令封包的索引號及命令碼一致, 除了第一筆命令被視為有效命令外, 其餘命令皆視為ECM_CMD_INFO_UPDATE_OP命令

ECM_CMD_INFO_UPDATE_OP 更新標頭資料			
命令封包			
標頭	1	命令碼	0
	2	索引號	自定義
	3	資料長度	0

ECM_CMD_ECAT_INIT_OP

初始化EtherCAT網路及從站, 使用EtherCAT功能前必須先初始化EtherCAT網路

命令封包

標頭	1	命令碼	1
	2	索引號	自定義
	3	資料長度	16
	4	命令參數	48:特殊從站 其他:一般從站
	5	保留	bit 12: 1代表master shift mode, 0代表bus shift mode(版本0x22後) bit 11: 1代表disable LRW command(版本0x22後)

資料段	1	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;">3</td> <td style="text-align: center;">2</td> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td></td> </tr> <tr> <td style="text-align: center;">DC Activate</td> <td style="text-align: center;">保留</td> <td style="text-align: center;">保留</td> <td style="text-align: center;">0</td> <td></td> </tr> <tr> <td colspan="3" style="text-align: center;">SYNC0 Cycle Time (ns)</td> <td style="text-align: center;">4</td> <td></td> </tr> <tr> <td colspan="3" style="text-align: center;">SYNC1 Cycle Time (ns)</td> <td style="text-align: center;">8</td> <td></td> </tr> <tr> <td colspan="3" style="text-align: center;">Shift Time (ns)</td> <td style="text-align: center;">12</td> <td></td> </tr> </table>			3	2	1	0		DC Activate	保留	保留	0		SYNC0 Cycle Time (ns)			4		SYNC1 Cycle Time (ns)			8		Shift Time (ns)			12	
	3	2	1	0																									
	DC Activate	保留	保留	0																									
	SYNC0 Cycle Time (ns)			4																									
	SYNC1 Cycle Time (ns)			8																									
	Shift Time (ns)			12																									
		位置	長度	名稱	說明																								
		0x00	1	保留	固定值0xFF																								
		0x01	1	保留	固定值0x81 預設關閉FIFO 如需使用FIFO功能,進入OP後再行啟用FIFO功能																								
		0x02	2	DC Activate	請參考ESI中的AssignActivate欄位																								
	0x04	4	SYNC0 Cycle Time	SYNC0週期時間(單位ns)																									
	0x08	4	SYNC1 Cycle Time	SYNC1週期時間(單位ns)																									
	0x12	4	Shift Time	週期時間偏移量(單位ns)																									

ECM_CMD_ECAT_RECONFIG_OP 當重新規劃PDO後, 需使用此命令使ECM-XF重新配置記憶體空間			
命令封包			
標頭	1	命令碼	2
	2	索引號	自定義
	3	資料長度	0

* 若從站使用ECM_CMD_ECAT_RECONFIG_OP後Slave的PDO配置恢復預設, 則請改用ECM_CMD_ECAT_CONFIG_MAP (命令碼73)代替

ECM_CMD_ECAT_INIT_DC_OP 初始化分散時鐘			
命令封包			
標頭	1	命令碼	3
	2	索引號	自定義
	3	資料長度	0

*不同從站間若要指定不同的DC Activate, 可於ECM_CMD_ECAT_INIT_OP後透過ECM_CMD_ECAT_DCSYNC變更單一從站的DC Active Code, 變更後須使用ECM_CMD_ECAT_INIT_DC_OP初始化分散時鐘

ECM_CMD_ECAT_PDO_WC_GET 讀取PDO中具有輸入資料從站數量、具有輸出資料從站數量或目前Working counter值			
命令封包			
標頭	1	命令碼	4
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	1: 具有輸入資料從站數量 2: 具有輸出資料從站數量 3: Working counter值
回應封包			
標頭	1	命令碼	4
	2	索引號	同命令封包索引號
	3	資料長度	4

資料段	1	讀取數量返回值
-----	---	---------

Working counter理論值 = 具有輸入資料從站數量 *1 + 具有輸出資料從站數量*2

ECM_CMD_ECAT_PDO_DATA_FIFO_OP 對FIFO存取PDO資料			
命令封包			
標頭	1	命令碼	5
	2	索引號	自定義
	3	資料長度	RxPDO長度
	4	命令參數	資料段內PDO數量
	5	命令資料0	讀寫操作碼 bit0 : 寫操作 bit1 : 讀操作
資料段	1	PDO資料	RxPDO資料
回應封包			
標頭	1	命令碼	5
	2	索引號	同命令封包索引號
	3	資料長度	TxPDO長度
	4	返回值	讀寫操作碼 bit0 : 寫操作 bit1 : 讀操作
資料段	1		TxPDO資料

ECM_CMD_ECAT_PDO_DATA_OP 讀取PDO資料		
命令封包		

標頭	1	命令碼	6
	2	索引號	自定義
	3	資料長度	RxPDO長度
	4	命令資料0	操作碼 bit1 : 讀操作 bit2 : 1 讀寫操作碼有效, 若此bit為0, 則一律進行讀及寫操作
資料段	1	RxPDO資料	
回應封包			
標頭	1	命令碼	6
	2	索引號	同命令封包索引號
	3	資料長度	TxPDO長度
資料段	1	TxPDO資料	

* 若欲寫入PDO, 請改以ECM_CMD_ECAT_PDO_DATA_FIFO_OP

ECM_CMD_ECAT_PDO_CONFIG_SET 配置從站PDO 註: 本API設計只能容納3個PDO各8個Objects, 如果超過請使用SDO直接進行設置			
命令封包			
標頭	1	命令碼	7
	2	索引號	自定義
	3	資料長度	112

資料段

1

3	2	1	0	
PDO指定索引號		PDO數量	從站站號	0x00
2nd PDO映射		1st PDO映射		0x04
1st PDO物件數量		3rd PDO映射		0x08
3rd PDO物件數量		2nd PDO物件數量		0x0C
1st PDO物件0				0x10
.....				
1st PDO物件7				
2nd PDO物件0				0x30
.....				
2nd PDO物件7				
3rd PDO物件0				0x50
.....				
3rd PDO物件7				

位置	長度	名稱	說明
0x00	1	從站站號	0~127
0x01	1	PDO數量	0~127
0x02	2	PDO指定索引號	TxPDO或RxPDO指定索引號
0x04	2	1st PDO映射	第一個PDO映射索引號
0x06	2	2nd PDO映射	第二個PDO映射索引號
0x08	2	3rd PDO映射	第三個PDO映射索引號
0x0A	2	1st PDO物件數量	第一個PDO物件數量
0x0C	2	2nd PDO物件數量	第二個PDO物件數量
0x0E	2	3rd PDO物件數量	第三個PDO物件數量
0x10	4	第一個PDO物件0	

			3	2	1	0
			索引號		副索引號	長度(bits)
	...		第一個PDO物件N(N=1~7)			
	0x30	4	第二個PDO物件0			
	...		第二個PDO物件N(N=1~7)			
	0x50	4	第三個PDO物件0			
	...		第三個PDO物件N(N=1~7)			

* 如使用ECM_CMD_ECAT_PDO_CONFIG_SET無法完成PDO配置或PDO物件數量不符需求，可改用SDO對從站相關物件進行讀寫。

* 部分從站不支援主站配置PDO，請參閱從站說明。

ECM_CMD_ECAT_PDO_CONFIG_REQ												
請求執行PDO配置讀取，操作完成後使用ECM_CMD_ECAT_PDO_CONFIG_GET取回配置												
命令封包												
標頭	1	命令碼	8									
	2	索引號	自定義									
	3	資料長度	4									
資料段	1	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;">3</td> <td style="text-align: center;">2</td> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> </tr> <tr> <td colspan="2" style="text-align: center;">PDO指定索引號</td> <td style="text-align: center;">保留</td> <td style="text-align: center;">從站站號</td> </tr> </table>		3	2	1	0	PDO指定索引號		保留	從站站號	0x00
	3	2	1	0								
	PDO指定索引號		保留	從站站號								
		位置	長度	名稱	說明							
		0x00	1	從站站號	0~127							
	0x01	1	保留									
	0x02	2	PDO指定索引號	TxPDO或RxPDO指定索引號								

ECM_CMD_ECAT_PDO_CONFIG_GET	
讀取從站PDO配置	
命令封包	

標頭	1	命令碼	9																																																										
	2	索引號	自定義																																																										
	3	資料長度	0																																																										
回應封包																																																													
標頭	1	命令碼	9																																																										
	2	索引號	同命令封包索引號																																																										
	3	資料長度	112																																																										
資料段	1	<table border="1" style="width:100%; text-align:center;"> <tr> <td style="width:25%;">3</td> <td style="width:25%;">2</td> <td style="width:25%;">1</td> <td style="width:25%;">0</td> </tr> <tr> <td>PDO指定索引號</td> <td>PDO數量</td> <td>從站站號</td> <td>0x00</td> </tr> <tr> <td colspan="2">2nd PDO映射</td> <td>1st PDO映射</td> <td>0x04</td> </tr> <tr> <td colspan="2">1st PDO物件數量</td> <td>3rd PDO映射</td> <td>0x08</td> </tr> <tr> <td colspan="2">3rd PDO物件數量</td> <td>2nd PDO物件數量</td> <td>0x0C</td> </tr> <tr> <td colspan="3">1st PDO物件0</td> <td>0x10</td> </tr> <tr> <td colspan="3">.....</td> <td></td> </tr> <tr> <td colspan="3">1st PDO物件7</td> <td></td> </tr> <tr> <td colspan="3">2nd PDO物件0</td> <td>0x30</td> </tr> <tr> <td colspan="3">.....</td> <td></td> </tr> <tr> <td colspan="3">2nd PDO物件7</td> <td></td> </tr> <tr> <td colspan="3">3rd PDO物件0</td> <td>0x50</td> </tr> <tr> <td colspan="3">.....</td> <td></td> </tr> <tr> <td colspan="3">3rd PDO物件7</td> <td></td> </tr> </table>				3	2	1	0	PDO指定索引號	PDO數量	從站站號	0x00	2nd PDO映射		1st PDO映射	0x04	1st PDO物件數量		3rd PDO映射	0x08	3rd PDO物件數量		2nd PDO物件數量	0x0C	1st PDO物件0			0x10				1st PDO物件7				2nd PDO物件0			0x30				2nd PDO物件7				3rd PDO物件0			0x50				3rd PDO物件7			
		3	2	1	0																																																								
		PDO指定索引號	PDO數量	從站站號	0x00																																																								
		2nd PDO映射		1st PDO映射	0x04																																																								
		1st PDO物件數量		3rd PDO映射	0x08																																																								
		3rd PDO物件數量		2nd PDO物件數量	0x0C																																																								
		1st PDO物件0			0x10																																																								
																																																												
		1st PDO物件7																																																											
		2nd PDO物件0			0x30																																																								
																																																												
		2nd PDO物件7																																																											
		3rd PDO物件0			0x50																																																								
																																																												
		3rd PDO物件7																																																											
		位置	長度	名稱	說明																																																								
		0x00	1	從站站號	0~127																																																								
		0x01	1	PDO數量	0~127																																																								
		0x02	2	PDO指定索引號	TxPDO或PxPDO指定索引號																																																								

	0x04	2	1st PDO映射	第一個PDO映射索引號				
	0x06	2	2nd PDO映射	第二個PDO映射索引號				
	0x08	2	3rd PDO映射	第三個PDO映射索引號				
	0x0A	2	1st PDO物件數量	第一個PDO物件數量				
	0x0C	2	2nd PDO物件數量	第二個PDO物件數量				
	0x0E	2	3rd PDO物件數量	第三個PDO物件數量				
	0x10	4	第一個PDO物件0 3 2 1 0 <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 33%;">索引號</td> <td style="width: 33%;">副索引號</td> <td style="width: 33%;">長度(bits)</td> </tr> </table>			索引號	副索引號	長度(bits)
	索引號	副索引號	長度(bits)					
	...		第一個PDO物件N(N=1~7)					
	0x30	4	第二個PDO物件0					
	...		第二個PDO物件N(N=1~7)					
	0x50	4	第三個PDO物件0					
...		第三個PDO物件N(N=1~7)						

ECM_CMD_ECAT_SDO_REQ			
請求執行SDO讀寫命令, 寫操作指令資料段為寫入資料, 讀操作完成後透過ECM_CMD_ECAT_SDO_GET指令取回讀取資料			
命令封包			
標頭	1	命令碼	10
	2	索引號	自定義
	3	資料長度	讀操作 : 12 寫操作 : 12+寫入物件值長度

	3	資料長度	資料段有效長度
資料段	1	前次ECM_CMD_ECAT_SDO_REQ讀操作讀回資料	

ECM_CMD_ECAT_STATE_SET 變更EtherCAT狀態			
命令封包			
標頭	1	命令碼	12
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	0~127:從站站號 0xFF:所有從站
	5	命令資料0	EtherCAT狀態 0x01: EC_STATE_INIT 0x02: EC_STATE_PRE_OP 0x03: EC_STATE_BOOT 0x04: EC_STATE_SAFE_OP 0x08: EC_STATE_OPERATIONAL

ECM_CMD_ECAT_STATE_GET 自XF中取回EtherCAT狀態			
命令封包			
標頭	1	命令碼	13
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	0~127:從站站號 0xFF:所有從站
回應封包			
標頭	1	命令碼	13
	2	索引號	同命令封包索引號
	3	資料長度	0

	4	返回值	<p>EtherCAT狀態</p> <p>0x01: EC_STATE_INIT</p> <p>0x02: EC_STATE_PRE_OP</p> <p>0x03: EC_STATE_BOOT</p> <p>0x04: EC_STATE_SAFE_OP</p> <p>0x08: EC_STATE_OPERATIONAL</p>
--	---	-----	----------------------------------------------------------------------------------------------------------------------------------------------------------------------

* 若需更新從站的EtherCAT狀態, 先請使用ECM_CMD_ECAT_STATE_UPDATE更新後再使用本命令

ECM_CMD_ECAT_SLV_INFO_GET 讀取從站資訊			
命令封包			
標頭	1	命令碼	15
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	0~127: 從站站號
	5	命令資料0	<p>從站資訊碼</p> <p>0: 廠商碼</p> <p>1: 廠商產品碼</p> <p>2: 產品版本號</p> <p>3: 產品名稱</p> <p>4: 配置位置</p> <p>5: 位置別名</p> <p>6: 狀態 (1: Init / 2:PreOP / 4:SafeOP / 8:OP)</p> <p>7: AL狀態 (0:沒有錯誤/ 其他: 請參閱slave AL status code說明)</p> <p>8: 輸出資料長度 (RxPDO Size)</p> <p>9: 輸入資料長度 (TxPDO Size)</p>
回應封包			
標頭	1	命令碼	15
	2	索引號	同命令封包索引號
	3	資料長度	從站資訊資料長度
資料段	1	從站資訊	

* 更多從站資訊請參考EcmDriver.h中ECM_SLV_INFO的定義, 讀回資料為從站上之資料, 主站無法判斷正確性。

ECM_CMD_ECAT_SLV_CNT_GET 讀取從站數量			
命令封包			
標頭	1	命令碼	16
	2	索引號	自定義
	3	資料長度	0
回應封包			
標頭	1	命令碼	16
	2	索引號	同命令封包索引號
	3	資料長度	0
	4	返回值	從站數量

ECM_CMD_FIFO_ENABLE 啟動RxPDO FIFO輸出，ECM-XF預設啟動RxPDO FIFO輸出，使用者可以透過此命令關閉RxPDO FIFO週期輸出，可用於預儲存輸出。			
命令封包			
標頭	1	命令碼	18
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	0：關閉 1：啟動

ECM_CMD_FIFO_PACK_SIZE_GET 讀取PDO資料長度，FIFO內一筆資料為一個PDO			
命令封包			
標頭	1	命令碼	19
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	0：讀取RxPDO資料長度 1：讀取TxPDO資料長度

回應封包			
標頭	1	命令碼	19
	2	索引號	同命令封包索引號
	3	資料長度	2
資料段	1	PDO資料長度	

ECM_CMD_SPI_PACK_SIZE_GET 讀取SPI資料封包長度			
命令封包			
標頭	1	命令碼	20
	2	索引號	自定義
	3	資料長度	0
回應封包			
標頭	1	命令碼	20
	2	索引號	同命令封包索引號
	3	資料長度	2
資料段	1	SPI資料長度	

ECM_CMD_SPI_RECONFIG_OP 重設SPI資料封包長度			
命令封包			
標頭	1	命令碼	23
	2	索引號	自定義
	3	資料長度	SPI封包內的資料欄位長度(32~1408)
回應封包			
標頭	1	命令碼	23

	2	索引號	同命令封包索引號
	3	資料長度	0

注意:使用ECM_CMD_SPI_RECONFIG_OP命令時, CRC欄位一律給固定值(0x12345678), 因SPI Data Size變化, 將可能導致CRC欄位變動, 致使回應標頭中CRC錯誤次數增加, 此為正常現象。

注意:ECM-XFU SPI Data Size固定為992Bytes, 無法變更

ECM_CMD_CRC_ERR_CNT_CLR 清除CRC檢查累積錯誤次數			
命令封包			
標頭	1	命令碼	24
	2	索引號	自定義
	3	資料長度	0
回應封包			
標頭	1	命令碼	24
	2	索引號	同命令封包索引號
	3	資料長度	0
	4	返回值	清除CRC檢查累積錯誤次數

ECM_CMD_CRC_TYPE_SET 設定檢查碼類型			
命令封包			
標頭	1	命令碼	25
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	<p>0: 固定值檢查: 0x12345678</p> <p>1: 未定義</p> <p>2: 未定義</p> <p>3: CRC-32: (從頭碼的第一個位元計算到檢查碼欄位之前的位元, 尾碼不計算)</p> $X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1 \text{ (Poly = 0x04C11DB7)}$

			Init = 0xFFFFFFFF, RefIn = True, RefOut = True, XorOut = 0xFFFFFFFF
--	--	--	---------------------------------------------------------------------

ECM_CMD_402_CONFIG_SET 指定從站控制字及狀態字偏移量			
命令封包			
標頭	1	命令碼	26
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	馬達號數(第一軸為編號0)
	5	命令資料0	控制字偏移量(單位Byte)
	6	命令資料1	狀態字偏移量(單位Byte)
	7	命令資料2	內部402狀態機控制位元組設定 bit3 : 從站狀態機錯誤清除 bit4 : 狀態機控制啟動 bit7 : 從站狀態機錯誤自動清除啟動 啟用此功能會自動控制該軸的 control word, 以達到切換從站402狀態, fault reset等功能。 bit3為1表單次錯誤清除(Fault Reset), bit4為1表啟用(Enable), bit7為1表自動連續錯誤清除(Fault auto reset)
	8	命令資料3	從站站號(第一站為編號0)

*狀態機控制啟動後, 將由XF覆寫從站控制字之bit0~3, 使用者對從站控制字之bit0~3指定無效。

*若開啟錯誤清除機制, 將由XF覆寫從站控制字之bit7, 使用者對從站控制字之bit7指定無效。

ECM_CMD_402_STATE_SET 切換指定從站402狀態			
命令封包			
標頭	1	命令碼	27
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	馬達號數(第一軸為編號0)
	5	命令資料0	402狀態

			NOT READY TO SWITCH ON	0x00
			SWITCHED ON DISABLED	0x40
			READY TO SWITCH ON	0x21
			SWITCHED ON	0x23
			OPERATION ENABLED	0x27
			QUICK STOP ACTIVE	0x07
			FAULT REACTION ACTIVE	0x0F
			FAULT	0x08

ECM_CMD_402_STATE_GET 讀取指定從站402狀態				
命令封包				
標頭	1	命令碼	28	
	2	索引號	自定義	
	3	資料長度	0	
	4	命令參數	馬達號數(第一軸為編號0)	
回應封包				
標頭	1	命令碼	28	
	2	索引號	同命令封包索引號	
	3	資料長度	0	
	4	返回值	目前402狀態	

ECM_CMD_402_CTL_SET ECM-XF內部402狀態機控制位元組設定					
命令封包					
標頭	1	命令碼	29		
	2	索引號	自定義		
	3	資料長度	0		
	4	命令參數	馬達號數(第一軸為編號0)		
	5	命令資料0	bit3 : 從站狀態機錯誤清除 bit4 : 狀態機控制啟動 bit7 : 從站狀態機錯誤自動清除啟動		
	6	命令資料1	固定值 0x5A		

	7	命令資料2	馬達總數
	8	命令資料3	從站編號

*狀態機控制啟動後，將由XF覆寫從站控制字之bit0~3，使用者對從站控制字之bit0~3指定無效。

*若開啟錯誤清除機制，將由XF覆寫從站控制字之bit7，使用者對從站控制字之bit7指定無效。

ECM_CMD_402_CTL_GET ECM-XF內部402狀態機控制位元組讀取			
命令封包			
標頭	1	命令碼	30
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	馬達號數(第一軸為編號0)
回應封包			
標頭	1	命令碼	30
	2	索引號	同命令封包索引號
	3	資料長度	0
	4	返回值	402狀態機控制設定值

ECM_GPIO_CONFIG_SET(ECM_GpioSetMode) 設置GPIO模式			
命令封包			
標頭	1	命令碼	31
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	0
	5	命令資料0	Bit n: 1為設定第n個GPIO模式(n = 0 to 7)
	6	命令資料1	Bit n: 1為設定第n+8個GPIO模式(n = 0 to 7)
	7	命令資料2	GPIO模式定義

		0x0: 輸入模式 0x1: 輸出模式 0x2: 漏極開路(Open-Drain)模式 0x3: 近似雙向(Quasi-bidirectional)模式
8	命令資料3	GPIO上拉設定定義 0x0: 關閉上拉設定 0x1: 上拉模式 0x2: 下拉模式

ECM_GPIO_CONFIG_SET(ECM_GpioSetMode) 設置GPIO模式 (GPIO擴充功能, 僅特定版本適用)			
命令封包			
標頭	1	命令碼	31
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	8
	5	命令資料0	Bit n: 1為設定第n+16個GPIO模式(n = 0 to 3)
	6	命令資料1	保留, 為0
	7	命令資料2	GPIO模式定義 0x0: 輸入模式 0x1: 輸出模式 0x2: 漏極開路(Open-Drain)模式 0x3: 近似雙向(Quasi-bidirectional)模式
	8	命令資料3	GPIO上拉設定定義 0x0: 關閉上拉設定 0x1: 上拉模式 0x2: 下拉模式

ECM_GPIO_CONFIG_SET(ECM_GpioEnableDebounce) 啟用/禁用 GPIO彈跳消除 (針對Input才需設定)			
命令封包			
標頭	1	命令碼	31
	2	索引號	自定義
	3	資料長度	0

	4	命令參數	2
	5	命令資料0	Bit n: 1為設定第n個GPIO彈跳功能(n = 0 to 7)
	6	命令資料1	Bit n: 1為設定第n+8個GPIO彈跳功能(n = 0 to 7)
	7	命令資料2	彈跳消除 0: 禁用 1: 啟用
	8	命令資料3	保留

ECM_GPIO_CONFIG_SET(ECM_GpioEnableDebounce)
 啟用/禁用 GPIO彈跳消除 (針對Input才需設定)
 (GPIO擴充功能, 僅特定版本適用)

命令封包

標頭	1	命令碼	31
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	9
	5	命令資料0	Bit n: 1為設定第n+16個GPIO彈跳功能 (n = 0 to 3)
	6	命令資料1	保留, 為0
	7	命令資料2	彈跳消除 0: 禁用 1: 啟用
	8	命令資料3	保留

ECM_GPIO_CONFIG_SET(ECM_GpioSetDebounceClock)
 設定彈跳時脈

命令封包

標頭	1	命令碼	31
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	3
	5	命令資料0	Bit 0 ~ 3: 彈跳時脈週期, 有效值為0至15

			Bit 4: 時脈來源, 0為192M、1為10K 採樣週期為2的(彈跳周期時脈)*(時脈時間)次方
--	--	--	-------------------------------------------------------

ECM_GPIO_CONFIG_SET(ECM_GpioIntEnable) 啟用/禁用 GPIO中斷			
命令封包			
標頭	1	命令碼	31
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	1
	5	命令資料0	通道(0~15) 特定版本(0~19)
	6	命令資料1	中斷類型 0x0: 禁用中斷 0x1: 上升電壓或下降電壓啟用中斷 0x2: 下降電壓啟用中斷 0x3: 上升電壓啟用中斷

ECM_GPIO_CONFIG_SET(ECM_GpioIntClear) 清除GPIO中斷旗幟			
命令封包			
標頭	1	命令碼	31
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	4
	5	命令資料0	Bit n: 清除第n個GPIO中斷旗幟(n = 0 ~ 7)
	6	命令資料1	Bit n: 清除第n+8個GPIO中斷旗幟(n = 0 ~ 7)

ECM_GPIO_CONFIG_SET(ECM_GpioIntClear) 清除GPIO中斷旗幟 (GPIO擴充功能, 只能在版本9之後使用)			
命令封包			
標	1	命令碼	31

頭	2	索引號	自定義
	3	資料長度	0
	4	命令參數	6
	5	命令資料0	Bit n: 清除第n個GPIO中斷旗幟(n = 0 ~ 7)
	6	命令資料1	Bit n: 清除第n+8個GPIO中斷旗幟(n = 0 ~ 7)
	7	命令資料2	Bit n: 清除第n+16個GPIO中斷旗幟(n = 0 ~ 3) 僅特定版本適用

ECM_GPIO_FUNC_OP(ECM_GpioSetValue)
 設置GPIO數值 (僅為Output mode)

命令封包

標頭	1	命令碼	32
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	0
	5	命令資料0	Bit n: 設置1為第n個GPIO高位, 0為低位(n = 0 ~ 7)
	6	命令資料1	Bit n: 設置1為第n+8個GPIO高位, 0為低位(n = 0 ~ 7)

ECM_GPIO_FUNC_OP(ECM_GpioExtSetValue)
 設置GPIO數值
 (GPIO擴充功能, 只能在版本9之後使用)

命令封包

標頭	1	命令碼	32
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	2
	5	命令資料0	Bit n: 設置1為第n個GPIO高位, 0為低位(n = 0 ~ 7)
	6	命令資料1	Bit n: 設置1為第n+8個GPIO高位, 0為低位(n = 0 ~ 7)
	7	命令資料2	Bit n: 設置1為第n+16個GPIO高位, 0為低位(n = 0 ~ 3) 僅特定版本適用

ECM_GPIO_FUNC_OP(ECM_GpioGetValue) 取得GPIO數值			
命令封包			
標頭	1	命令碼	32
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	1
回應封包			
標頭	1	命令碼	32
	2	索引號	自定義
	3	資料長度	2
資料段			Bit n:當第n個GPIO為輸入模式時, 輸入為高位時為1 當第n個GPIO為輸入模式時, 輸入為低位時為0 (n = 0 ~ 15)

ECM_GPIO_FUNC_OP(ECM_GpioExtGetValue) 取得GPIO數值 (GPIO擴充功能, 只能在版本9之後使用)			
命令封包			
標頭	1	命令碼	32
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	3
回應封包			
標頭	1	命令碼	32
	2	索引號	自定義
	3	資料長度	4
資料段			Bit n:當第n個GPIO為輸入模式時, 輸入為高位時為1 當第n個GPIO為輸入模式時, 輸入為低位時為0 (n = 0 ~ 15), 僅特定版本(n = 0 ~ 19)

ECM_GPIO_FUNC_OP(ECM_GpioGetIntFlag) 取得GPIO中斷旗幟			
命令封包			
標頭	1	命令碼	32
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	5
回應封包			
標頭	1	命令碼	32
	2	索引號	自定義
	3	資料長度	2
資料段			Bit n:第n個GPIO中斷旗幟 (n = 0 ~ 15)

ECM_GPIO_FUNC_OP(ECM_GpioExtGetIntFlag) 取得GPIO中斷旗幟 (GPIO擴充功能, 僅特定版本適用)			
命令封包			
標頭	1	命令碼	32
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	7
回應封包			
標頭	1	命令碼	32
	2	索引號	自定義
	3	資料長度	4
資料段			Bit n:第n個GPIO中斷旗幟 (n = 0 ~ 19)

ECM_QEI_FUNC_OP(ECM_EncOpen) 開啟並設置編碼器模式			
命令封包			
標頭	1	命令碼	33
	2	索引號	自定義
	3	資料長度	4
	4	命令參數	4
資料段			模式 0x000: X4自由計步模式 0x100: X2自由計步模式 0x200: X4比較計步模式 0x300: X2比較計步模式

ECM_QEI_FUNC_OP(ECM_EncStart) 開始編碼器計步			
命令封包			
標頭	1	命令碼	33
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	5

ECM_QEI_FUNC_OP(ECM_EncStop) 停止編碼器計步			
命令封包			
標頭	1	命令碼	33
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	6

ECM_QEI_FUNC_OP(ECM_EncGetCount) 取得編碼器計步數值			
命令封包			

標頭	1	命令碼	33
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	21
回應封包			
標頭	1	命令碼	33
	2	索引號	自定義
	3	資料長度	4
資料段			編碼器資料

ECM_DAC_FUNC_OP(ECM_DacOpen) 開啟DAC模式			
命令封包			
標頭	1	命令碼	34
	2	索引號	自定義
	3	資料長度	4
	4	命令參數	20
資料段			<p>模式</p> <p>0x0000: 利用ECM_DacSetData()觸發DAC輸出</p> <p>0x0010: 利用ECM_DacStartConv()觸發DAC輸出</p> <p>0x0030: DAC_ST腳位為低位時觸發</p> <p>0x1030: DAC_ST腳位為高位時觸發</p> <p>0x2030: DAC_ST腳位為上升電壓時觸發</p> <p>0x3030: DAC_ST腳位微下降電壓時觸發</p>

ECM_DAC_FUNC_OP(ECM_DacClose) 關閉DAC模式			
命令封包			
標頭	1	命令碼	34
	2	索引號	自定義

	3	資料長度	0
	4	命令參數	21

ECM_DAC_FUNC_OP(ECM_DacSetDelayTime) 設置DAC延遲時間			
命令封包			
標頭	1	命令碼	34
	2	索引號	自定義
	3	資料長度	4
	4	命令參數	22
資料段			可設定的延遲範圍為 0 ~ 1023(單位: 1/96 us) DAC從0x000至0xFFf至少需要8us, 故此值建議應大於768

ECM_DAC_FUNC_OP(ECM_DacSetData) 設置DAC資料			
命令封包			
標頭	1	命令碼	34
	2	索引號	自定義
	3	資料長度	4
	4	命令參數	11
資料段			資料數值 範圍 0 ~ 4095 對應 0 ~ 3.3V

ECM_DAC_FUNC_OP(ECM_DacStartConv) 開始DAC轉換			
命令封包			
標頭	1	命令碼	34
	2	索引號	自定義
	3	資料長度	0

	4	命令參數	0
--	---	------	---

ECM_ADC_FUNC_OP(ECM_AdcOpen) 開啟ADC功能			
命令封包			
標頭	1	命令碼	35
	2	索引號	自定義
	3	資料長度	4
	4	命令參數	40
資料段			0x000: 單端模式 0x100: 差分模式

ECM_ADC_FUNC_OP(ECM_AdcClose) 關閉ADC功能			
命令封包			
標頭	1	命令碼	35
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	41

ECM_ADC_FUNC_OP(ECM_AdcConfigSampleModule) 設定ADC觸發來源											
命令封包											
標頭	1	命令碼	35								
	2	索引號	自定義								
	3	資料長度	12								
	4	命令參數	42								
資料段			<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center; width: 25%;">3</td> <td style="text-align: center; width: 25%;">2</td> <td style="text-align: center; width: 25%;">1</td> <td style="text-align: center; width: 25%;">0</td> </tr> <tr> <td colspan="4" style="border-top: 1px solid black;"></td> </tr> </table>	3	2	1	0				
3	2	1	0								

			保留, 為0	0x0
			0: 軟體觸發	0x4
			保留, 為0	0x8

ECM_ADC_FUNC_OP(ECM_AdcGetDataValidFlag)
取得ADC有效資料旗幟

命令封包

標頭	1	命令碼	35
	2	索引號	自定義
	3	資料長度	4
	4	命令參數	17
資料段			固定值, 為0x1000

回應封包

標頭	1	命令碼	35
	2	索引號	自定義
	3	資料長度	4
資料段			有效旗幟 當有效旗幟非為零時轉換資料為有效

ECM_ADC_FUNC_OP(ECM_AdcStartConv)
開始ADC轉換

命令封包

標頭	1	命令碼	35
	2	索引號	自定義
	3	資料長度	4
	4	命令參數	12

資料段		固定值, 為0x1000
-----	--	--------------

ECM_ADC_FUNC_OP(ECM_AdcGetConvData) 取得ADC轉換資料			
命令封包			
標頭	1	命令碼	35
	2	索引號	自定義
	3	資料長度	4
	4	命令參數	15
資料段		固定值, 為12	
回應封包			
標頭	1	命令碼	35
	2	索引號	自定義
	3	資料長度	4
資料段		轉換資料 0 ~ 4095 對應 0 ~ 3.3V	

ECM_EEPROM_REQ 請求執行EEPROM讀寫命令, 寫操作指令資料段為寫入資料, 讀操作完成後透 ECM_EEPROM_GET指令取回讀取資料			
命令封包			
標頭	1	命令碼	38
	2	索引號	自定義
	3	資料長度	12

資料段	1	<table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 25%;">3</td> <td style="width: 25%;">2</td> <td style="width: 25%;">1</td> <td style="width: 25%;">0</td> </tr> <tr> <td>從站站號</td> <td colspan="2">操作碼</td> <td>0</td> </tr> <tr> <td>寫入資料</td> <td colspan="2">讀寫位置</td> <td>4</td> </tr> <tr> <td colspan="3">操作限時</td> <td>8</td> </tr> </table>				3	2	1	0	從站站號	操作碼		0	寫入資料	讀寫位置		4	操作限時			8
		3	2	1	0																
		從站站號	操作碼		0																
		寫入資料	讀寫位置		4																
	操作限時			8																	
	位置	長度	名稱	說明																	
	0x00	2	操作碼	0: 讀取 1: 寫入																	
	0x02	2	從站站號	0~127																	
0x04	2	讀寫位置	EEPROM位置																		
0x06	2	寫入資料	寫操作有效																		
0x08	4	操作限時	最大等待時間(μs)																		

ECM_EEPROM_GET 讀回ECM_EEPROM_REQ讀操作結果			
命令封包			
標頭	1	命令碼	39
	2	索引號	自定義
	3	資料長度	0
回應封包			
標頭	1	命令碼	39
	2	索引號	同命令封包索引號
	3	資料長度	資料段有效長度
	4	返回值	操作錯誤碼
資料段	1	取回ECM_EEPROM_REQ讀操作讀回資料	

ECM_CMD_ECAT_STATE_CHECK 更新並確認從站狀態			
命令封包			
標頭	1	命令碼	41
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	0~127: 從站站號 0xFF: 所有從站
	5	命令資料0	EtherCAT狀態

* 此命令為ECM_CMD_ECAT_STATE_SET + ECM_CMD_ECAT_STATE_UPDATE + ECM_CMD_ECAT_STATE_GET的複合命令

ECM_CMD_ECAT_DCSYNC 設置DC SYNC信號源、週期時間、偏移時間																							
命令封包																							
標頭	1	命令碼	50																				
	2	索引號	自定義																				
	3	資料長度	16																				
資料段	<table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 25%;">3</td> <td style="width: 25%;">2</td> <td style="width: 25%;">1</td> <td style="width: 25%;">0</td> </tr> <tr> <td>DC Activate</td> <td>保留</td> <td>從站站號</td> <td>0</td> </tr> <tr> <td colspan="3">SYNC0 Cycle Time (ns)</td> <td>4</td> </tr> <tr> <td colspan="3">SYNC1 Cycle Time (ns)</td> <td>8</td> </tr> <tr> <td colspan="3">Shift Time (ns)</td> <td>12</td> </tr> </table>			3	2	1	0	DC Activate	保留	從站站號	0	SYNC0 Cycle Time (ns)			4	SYNC1 Cycle Time (ns)			8	Shift Time (ns)			12
	3	2	1	0																			
	DC Activate	保留	從站站號	0																			
	SYNC0 Cycle Time (ns)			4																			
	SYNC1 Cycle Time (ns)			8																			
Shift Time (ns)			12																				
回應封包																							
標頭	1	命令碼	50																				
	2	索引號	同命令封包索引號																				
	3	資料長度	0																				

*不同從站間若要指定不同的DC Activate, 可於ECM_CMD_ECAT_INIT_OP後透過ECM_CMD_ECAT_DCSYNC變更單一從站的DC Active Code, 變更後須使用

ECM_CMD_ECAT_INIT_DC_OP初始化分散時鐘

ECM_CMD_FIFO_CLR_OP 清除FIFO內容			
命令封包			
標頭	1	命令碼	51
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	0: 清除TxFIFO 及 RxFIFO 1: 僅清除TxFIFO 2: 僅清除RxFIFO

ECM_CMD_FIFO_SET_TX_CNT 設置Tx FIFO數量			
命令封包			
標頭	1	命令碼	52
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	Tx FIFO數量

- * TxFIFO數量預設為64筆, RxFIFO和TxFIFO總空間為0x4000Byte
- * RxFIFO數量 x RxPDOSize + TxFIFO數量 x TxPDOSize <= 0x4000 Byte
- * TxFIFO滿時, 最舊的資料會被丟棄 (也就是永遠保持最新的資料)

ECM_CMD_FIFO_GET_TX_CNT 取得Tx FIFO設置數量			
命令封包			
標頭	1	命令碼	53
	2	索引號	自定義
	3	資料長度	0
回應封包			
標頭	1	命令碼	53
	2	索引號	同命令封包索引號

	3	資料長度	0
	4	返回值	Tx FIFO設置數量

ECM_CMD_FIFO_SET_RX_CNT 設置Rx FIFO數量			
命令封包			
標頭	1	命令碼	54
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	Rx FIFO數量

- * RxFIFO數量預設為64筆, RxFIFO和TxFIFO總空間為0x4000Byte
- * RxFIFO數量 x RxPDOSize + TxFIFO數量 x TxPDOSize <= 0x4000 Byte
- * RxFIFO滿時, 最新的資料會被丟棄(也就是最新的資料無法放入FIFO)

ECM_CMD_FIFO_GET_RX_CNT 取得Rx FIFO設置數量			
命令封包			
標頭	1	命令碼	55
	2	索引號	自定義
	3	資料長度	0
回應封包			
標頭	1	命令碼	55
	2	索引號	同命令封包索引號
	3	資料長度	0
	4	返回值	Rx FIFO設置數量

ECM_CMD_FW_VERSION_GET 讀取韌體版本號			
命令封包			
標頭	1	命令碼	56
	2	索引號	自定義

	3	資料長度	0
回應封包			
標頭	1	命令碼	56
	2	索引號	同命令封包索引號
	3	資料長度	0
	4	返回值	韌體版本號

ECM_CMD_ECAT_STATE_UPDATE 更新從站ECAT狀態至XF			
命令封包			
標頭	1	命令碼	57
	2	索引號	自定義
	3	資料長度	0

ECM_CMD_ECAT_INT_SET_ENABLE 設置中斷啟用遮罩			
命令封包			
標頭	1	命令碼	58
	2	索引號	自定義
	3	資料長度	6
	4	命令參數	u8INTActiveHigh : INT0/INT1中斷訊號極性 BIT0 : INT1訊號極性 (0為低電位啟動, 1為高電位啟動), 預設為0 BIT1 : INT0訊號極性 (0為低電位啟動, 1為高電位啟動), 預設為1
資料段	INT0代表XF是否正處理命令而不接受新的命令		
	INT1的中斷來源		
	3	2	1
			0
	u32CompIntEnable 綜合中斷		0

		u8PeriplntEnable 輸入中斷與QEI 中斷	u8GpioIntEnable GPIO輸入中斷	4
<p>u32CompIntEnable : 綜合中斷</p> <p>BIT31 : EtherCAT封包收取</p> <p>BIT25 : RxFIFO低極限數</p> <p>BIT24 : TxFIFO高極限數</p> <p>BIT23 : CRC錯誤</p> <p>BIT22 : EtherCAT計數器錯誤</p> <p>u8GpioIntEnable : GPIO0~7輸入中斷</p> <p>BIT0 : GPIO00 INT</p> <p>BIT1 : GPIO01 INT</p> <p>BIT2 : GPIO02 INT</p> <p>BIT3 : GPIO03 INT</p> <p>BIT4 : GPIO04 INT</p> <p>BIT5 : GPIO05 INT</p> <p>BIT6 : GPIO06 INT</p> <p>BIT7 : GPIO07 INT</p> <p>u8PeriplntEnable : GPIO8~11 & QEI中斷</p> <p>BIT0 : GPIO08 INT</p> <p>BIT1 : GPIO09 INT</p> <p>BIT2 : GPIO10 INT</p> <p>BIT3 : GPIO11 INT</p> <p>BIT4 : QEI index INT</p> <p>BIT5 : QEI compare INT</p> <p>BIT6 : QEI Over/Under flow INT</p>				

* 一般而言發生中斷的順序為[EtherCAT封包收取] -> [TxFIFO高極限數] -> [RxFIFO低極限數]

ECM_CMD_ECAT_INT_GET_ENABLE 取得中斷啟用遮罩			
命令封包			
標頭	1	命令碼	59
	2	索引號	自定義
	3	資料長度	0
回應封包			
標頭	1	命令碼	59

	2	索引號	自定義		
	3	資料長度	6		
	4	返回值	u8INTActiveHigh : INT0/INT1中斷訊號極性 BIT0 : INT1訊號極性 (0為低電位啟動, 1為高電位啟動) BIT1 : INT0訊號極性 (0為低電位啟動, 1為高電位啟動)		
資料段	INT1的中斷來源				
	3		2		1
					0
	u32ComplntEnable 綜合中斷				0
		u8PeripIntEnable 輸入中斷與QEI 中斷		u8GpioIntEnable GPIO輸入中斷	4
<p>u32ComplntEnable : 綜合中斷</p> <p>BIT31 : EtherCAT封包收取</p> <p>BIT25 : RxFIFO低極限</p> <p>BIT24 : TxFIFO高極限</p> <p>BIT23 : CRC錯誤</p> <p>BIT22 : EtherCAT計數器錯誤</p> <p>u8GpioIntEnable : GPIO0~7輸入中斷</p> <p>BIT0 : GPIO00 INT</p> <p>BIT1 : GPIO01 INT</p> <p>BIT2 : GPIO02 INT</p> <p>BIT3 : GPIO03 INT</p> <p>BIT4 : GPIO04 INT</p> <p>BIT5 : GPIO05 INT</p> <p>BIT6 : GPIO06 INT</p> <p>BIT7 : GPIO07 INT</p> <p>u8PeripIntEnable : GPIO8~11 & QEI中斷</p> <p>BIT0 : GPIO08 INT</p> <p>BIT1 : GPIO09 INT</p> <p>BIT2 : GPIO10 INT</p> <p>BIT3 : GPIO11 INT</p> <p>BIT4 : QEI index INT</p> <p>BIT5 : QEI compare INT</p> <p>BIT6 : QEI Over/Under flow INT</p>					

ECM_CMD_FIFO_INIT 初始化FIFO, 根據FIFO數量與PDO大小建置FIFO空間			
命令封包			
標頭	1	命令碼	66
	2	索引號	自定義
	3	資料長度	0
回應封包			
標頭	1	命令碼	66
	2	索引號	自定義
	3	資料長度	0

- 最大空間為0x4000Byte

ECM_CMD_ECAT_SDO_ABORTCODE_GET 取得SDO Abort Code (SDO錯誤代碼)			
命令封包			
標頭	1	命令碼	69
	2	索引號	自定義
	3	資料長度	0
回應封包			
標頭	1	命令碼	69
	2	索引號	自定義
	3	資料長度	4
資料段	1	Byte0~3: SDO Abort Code Abort Code由從站回傳依據ETG定義如下, 僅供參考 0x00000000 No SDO error 0x05030000 Toggle bit not changed 0x05040000 SDO timeout 0x05040001 Command specifier unknown 0x05040005 Out of memory 0x06010000 Unsupported Access 0x06010001 Write only entry 0x06010002 Read only entry	

	<p>0x06010003 Entry can not be written because Subindex0 is not 0</p> <p>0x06010004 The object can not be accessed via complete access</p> <p>0x06020000 Object not existing</p> <p>0x06040041 Object can not be mapped to PDO</p> <p>0x06040042 Mapped Object exceeds PDO</p> <p>0x06040043 Parameter is incompatible</p> <p>0x06040047 Device incompatibility</p> <p>0x06060000 Hardware error</p> <p>0x06070010 Parameter length error</p> <p>0x06070012 Parameter is too long</p> <p>0x06070013 Parameter is too short</p> <p>0x06090011 Subindex (Entry) not exists</p> <p>0x06090030 Value exceeds</p> <p>0x06090031 Value is too great</p> <p>0x06090032 Value is too small</p> <p>0x06090033 Detected Module Ident List (0xF030) and Configured Module Ident list (0xF050) does not match</p> <p>0x06090036 Value is less than minimum value</p> <p>0x08000000 General error</p> <p>0x08000020 Data can not be read or written</p> <p>0x08000021 Data can not be accessed because of local control</p> <p>0x08000022 Data can not be read or written in the current state</p> <p>0x08000023 Object is not in the object dictionary</p>
--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

ECM_CMD_ECAT_SET_FIFO_TH			
設定TxFIFO上限及RxFIFO下限, 當觸及所設極限時, 透過INT1發出中斷			
命令封包			
標頭	1	命令碼	70
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	Disable: 0 Enable: 1 Bit 0 TxFIFO高極限 Bit 1 RxFIFO低極限
	5	命令資料0	TxFIFO的高極限數
	6	命令資料1	RxFIFO的低極限數

ECM_CMD_ECAT_CONFIG_SM 設定Sync Manager (請依從站說明設定)																
命令封包																
標頭	1	命令碼	71													
	2	索引號	自定義													
	3	資料長度	8													
資料段	<table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 25%;">3</td> <td style="width: 25%;">2</td> <td style="width: 25%;">1</td> <td style="width: 25%;">0</td> </tr> <tr> <td colspan="2">Start Address</td> <td>SM編號</td> <td>從站站號</td> </tr> <tr> <td>Activate</td> <td>Control Reg</td> <td colspan="2">SM Length</td> </tr> </table>			3	2	1	0	Start Address		SM編號	從站站號	Activate	Control Reg	SM Length		0
	3	2	1	0												
Start Address		SM編號	從站站號													
Activate	Control Reg	SM Length														
				4												
回應封包																
標頭	1	命令碼	71													
	2	索引號	同命令封包索引號													
	3	資料長度	0													

* ECM-XF會讀取從站記憶體來設置, 若欲設置的參數與從站內記憶體值不同時才需使用此命令手動配置

ECM_CMD_ECAT_CONFIG_FMMU 設定FMMU (FMMU由主站進行配置, 此命令已經失效)																
命令封包																
標頭	1	命令碼	72													
	2	索引號	自定義													
	3	資料長度	5													
資料段	<table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 25%;">3</td> <td style="width: 25%;">2</td> <td style="width: 25%;">1</td> <td style="width: 25%;">0</td> </tr> <tr> <td>FMMU2func</td> <td>FMMU1func</td> <td>FMMU0func</td> <td>從站站號</td> </tr> <tr> <td colspan="3"></td> <td>FMMU3func</td> </tr> </table>			3	2	1	0	FMMU2func	FMMU1func	FMMU0func	從站站號				FMMU3func	0
	3	2	1	0												
FMMU2func	FMMU1func	FMMU0func	從站站號													
			FMMU3func													
				4												

回應封包			
標頭	1	命令碼	72
	2	索引號	同命令封包索引號
	3	資料長度	0

* ECM-XF會配置從站FMMU, 使用者即使使用此命令配置, 依然會被覆蓋, 此命令已失效

ECM_CMD_ECAT_CONFIG_MAP 設定PDO映射			
命令封包			
標頭	1	命令碼	73
	2	索引號	自定義
	3	資料長度	0
回應封包			
標頭	1	命令碼	73
	2	索引號	自定義
	3	資料長度	0

ECM_CMD_402_GET_STATUSWORD 取得CiA402從站的狀態字			
命令封包			
標頭	1	命令碼	74
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	馬達號數(第一軸為編號0)
回應封包			
標頭	1	命令碼	74
	2	索引號	自定義
	3	資料長度	2

資料段	狀態字(status word)的值
-----	--------------------

* 建議直接使用SDO讀取從站的0x6041物件.

ECM_CMD_ASYNC_CMD_RESET 重設非週期命令			
命令封包			
標頭	1	命令碼	75
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	1: 清除AsyncBusy旗標 2: 跳過網口初始化等待 4: 放棄前一筆Async命令

ECM_CMD_MDIO 讀寫MDIO介面										
命令封包										
標頭	1	命令碼	81							
	2	索引號	自定義							
	3	資料長度	0							
	4	命令參數	0: 寫入 1: 讀取							
	5	命令資料	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 25%; text-align: center;">3</td> <td style="width: 25%; text-align: center;">2</td> <td style="width: 25%; text-align: center;">1</td> <td style="width: 25%; text-align: center;">0</td> </tr> <tr> <td colspan="2" style="border: 1px solid black; text-align: center;">寫入資料(寫入時) 保留(讀取時)</td> <td style="border: 1px solid black; text-align: center;">PHY位址</td> <td style="border: 1px solid black; text-align: center;">PHY暫存器</td> </tr> </table>	3	2	1	0	寫入資料(寫入時) 保留(讀取時)		PHY位址
3	2	1	0							
寫入資料(寫入時) 保留(讀取時)		PHY位址	PHY暫存器							
回應封包										
標頭	1	命令碼	81							
	2	索引號	自定義							

	3	資料長度	0(寫入時) 2(讀取時)
資料段		MDIO讀取資料	

ECM_CMD_DIRECT_ASSIGN 手動指定Process data長度																		
命令封包																		
標頭	1	命令碼	82															
	2	索引號	自定義															
	3	資料長度	0															
	4	命令參數	0: 啟用或不啟用功能 1: 指定process data長度															
	5	命令資料	<p>命令參數為0時:</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 25%;">3</td> <td style="width: 25%;">2</td> <td style="width: 25%;">1</td> <td style="width: 25%;">0</td> </tr> <tr> <td>保留</td> <td></td> <td>0為不啟用 1為啟用</td> <td>從站位址</td> </tr> </table> <p>命令參數為1時:</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 25%;">3</td> <td style="width: 25%;">2</td> <td style="width: 25%;">1</td> <td style="width: 25%;">0</td> </tr> <tr> <td>process data長度</td> <td></td> <td>0為指定output 1為指定input</td> <td>從站位址</td> </tr> </table>	3	2	1	0	保留		0為不啟用 1為啟用	從站位址	3	2	1	0	process data長度		0為指定output 1為指定input
3	2	1	0															
保留		0為不啟用 1為啟用	從站位址															
3	2	1	0															
process data長度		0為指定output 1為指定input	從站位址															
回應封包																		
標頭	1	命令碼	82															
	2	索引號	自定義															
	3	資料長度	0															

ECM_CMD_COE_EMERGENCY
列出最近的n筆CoE報警

命令封包

標頭	1	命令碼	83
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	n (0<n≤8)

回應封包

標頭	1	命令碼	83
	2	索引號	自定義
	3	資料長度	12*n

資料段	3 2 1 0				
	報警編號*註2		從站站號	0或0xff*註1	0
	error data	error register	error code		4
	error data				8
	報警編號*註2		從站站號	1或0xff*註1	12
	error data	error register	error code		16
	error data				20
	...				
	報警編號*註2		從站站號	n-1或0xff*註1	12*(n-1)
	error data	error register	error code		12*(n-1)+4
	error data				12*(n-1)+8

*註1:若為0xff時, 表示不存在報警
*註2:此編號表示此報警是ECM開始運作後讀到的第幾筆

ECM_CMD_WARM_RESET
重設ECM-XF

命令封包			
標頭	1	命令碼	127
	2	索引號	自定義
	3	資料長度	0

* 使用ECM_CMD_WARM_RESET後, ECM-XF會重置並初始化, 所有狀態均回到初始預設狀態, 下一筆命令必須等到初始化後才能接受

ECM_CMD_ECAT_WKC_CONTI_ERR_CNT_GET 讀取工作錯誤數值			
命令封包			
標頭	1	命令碼	130
	2	索引號	自定義
	3	資料長度	0
回應封包			
標頭	1	命令碼	130
	2	索引號	自定義
	3	資料長度	0
	4	返回值	現在wkc錯誤次數

ECM_CMD_ECAT_WKC_CONTI_ERR_MAX_GET 讀取最大工作錯誤數值			
命令封包			
標頭	1	命令碼	131
	2	索引號	自定義
	3	資料長度	0
回應封包			
標頭	1	命令碼	131
	2	索引號	自定義

	3	資料長度	0
	4	返回值	0~254 : 設定最大值 255: 設定最大值為無限大

ECM_CMD_ECAT_WKC_CONTI_ERR_MAX_SET

設定最大工作錯誤數值

當working count大於設置最大值時, 會停止PDO交換資料, 同時Head中的Status bit4/bit6為0

命令封包

標頭	1	命令碼	132
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	設置最大值 預設255(255代表設定最大值為無限大), 設定值可為1~254

ECM_CMD_RAW_ECAT_FUNC_REQ

發送任意EtherCAT datagram

命令封包

標頭	1	命令碼	137																								
	2	索引號	自定義																								
	3	資料長度	20+n																								
資料段	<table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 25%;">3</td> <td style="width: 25%;">2</td> <td style="width: 25%;">1</td> <td style="width: 25%;">0</td> </tr> <tr> <td>datagram資料長度</td> <td>保留位</td> <td>OP code</td> <td>0</td> </tr> <tr> <td>ADO</td> <td colspan="2">APD</td> <td>4</td> </tr> <tr> <td colspan="3">邏輯位址(LRD、LWR、LRW指令使用)</td> <td>8</td> </tr> <tr> <td colspan="3">逾時(單位μs)</td> <td>12</td> </tr> <tr> <td colspan="3">保留位</td> <td>16</td> </tr> </table>			3	2	1	0	datagram資料長度	保留位	OP code	0	ADO	APD		4	邏輯位址(LRD、LWR、LRW指令使用)			8	逾時(單位μs)			12	保留位			16
	3	2	1	0																							
	datagram資料長度	保留位	OP code	0																							
	ADO	APD		4																							
	邏輯位址(LRD、LWR、LRW指令使用)			8																							
逾時(單位μs)			12																								
保留位			16																								

		n位元組資料 (n≤256)		20
OP code可為以下數值 1: APRD; 2: APWR; 4: FPRD; 5: FPWR; 7: BRD; 8: BWR; 10: LRD; 11: LWR; 12: LRW; 13: ARMW; 14: FRMW				
回應封包				
標頭	1	命令碼	137	
	2	索引號	自定義	
	3	資料長度	0	

ECM_CMD_RAW_ECAT_FUNC_GET 讀取EtherCAT datagram回傳資料 需與ECM_CMD_RAW_ECAT_FUNC_REQ指令搭配					
命令封包					
標頭	1	命令碼	138		
	2	索引號	自定義		
	3	資料長度	0		
回應封包					
標頭	1	命令碼	138		
	2	索引號	自定義		
	3	資料長度	4+n		
資料段	3	2	1	0	
	datagram資料長度n		保留位	OP code	0
	ADO		APD		4
	邏輯位址(LRD、LWR、LRW指令使用)				8
	逾時(單位μs)				12
	Working counter				16
	n位元組資料 (n≤256)				20

--	--	--

ECM_CMD_FOE_FUNC_REQ FoE (File over EtherCAT)命令					
命令封包					
標頭	1	命令碼	139		
	2	索引號	自定義		
	3	資料長度	20+n (n: 檔名長度 或 寫入時的資料長度, 單位byte)		
資料段	FoE上傳/下載命令:				
	3	2	1	0	
	保留位	OP code 1: 上傳 2: 下載	保留位	從站編號	0
	FoE檔案密碼				4
	FoE檔案長度(單位位元組)				8
	逾時(單位μs)				12
	檔案資料在閃存記憶體の起始位置 (0≤起始位置<0x40000)				16
	n位元組のFoE檔案名稱				20
	寫入/讀取閃存記憶體命令:				
	3	2	1	0	
	保留位	OP code 3: 讀取 4: 寫入	保留位	保留位	0
	保留位				4
	寫入/讀取資料長度n(單位位元組, n≤256)				8
	保留位				12

		檔案資料在閃存記憶體的起始位置 (0≤起始位置<0x40000)	16
		n位元組的資料(僅寫入閃存記憶體命令時需要)	20

ECM_CMD_FOE_FUNC_GET 讀取FoE上傳資料 需ECM_CMD_FOE_FUNC_REQ指令搭配			
命令封包			
標頭	1	命令碼	140
	2	索引號	自定義
	3	資料長度	0
回應封包			
標頭	1	命令碼	140
	2	索引號	自定義
	3	資料長度	n
資料段	使用ECM_CMD_FOE_FUNC_REQ命令讀取閃存記憶體完成後, 回傳n位元組的資料		

ECM_CMD_ENABLE_LRW 啟用或禁用LRW指令			
命令封包			
標頭	1	命令碼	142
	2	索引號	自定義
	3	資料長度	0
	4	命令參數	0: 禁用LRW指令(使用LRD/LWR取代LRW指令) 1: 若從站支援LWR指令, 則使用LRW指令(預設)
回應封包			
標	1	命令碼	142

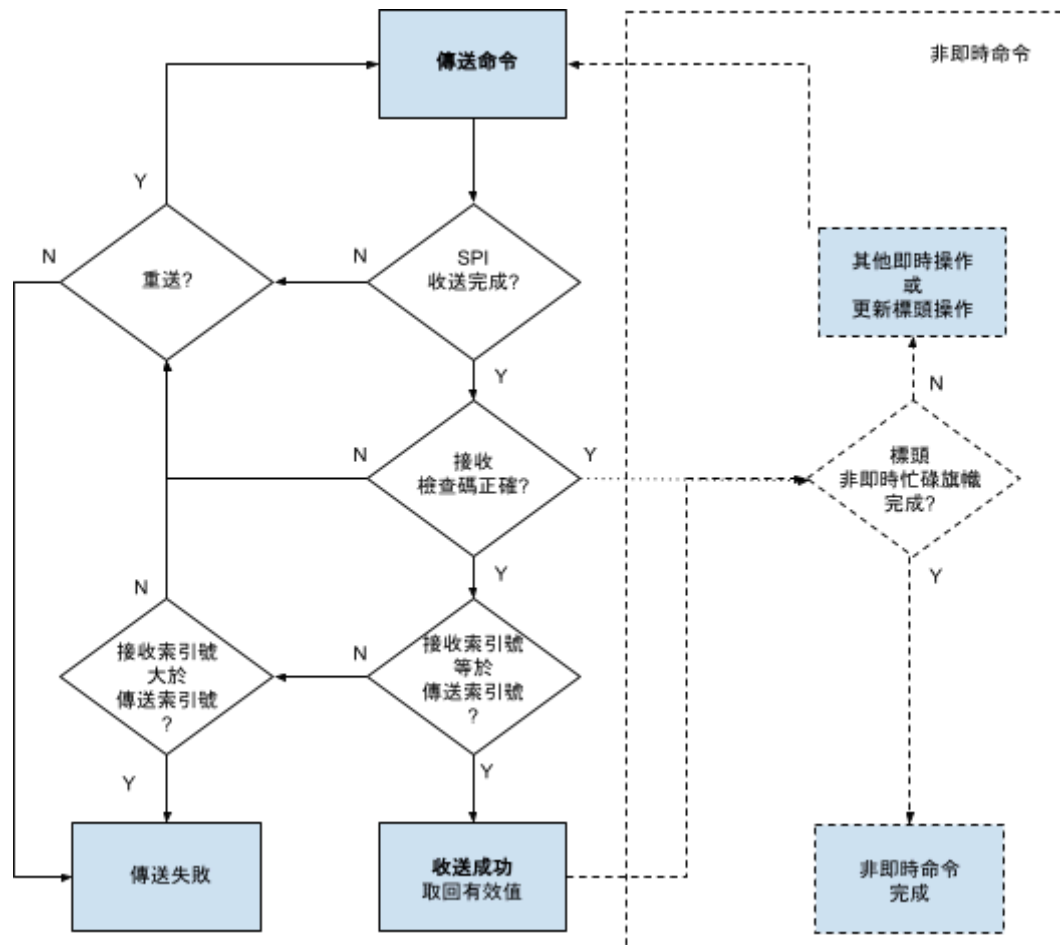
頭	2	索引號	自定義
	3	資料長度	0

ECM_CMD_RECV_MAILBOX 讀取從站接收郵箱資料																		
命令封包																		
標頭	1	命令碼	143															
	2	索引號	自定義															
	3	資料長度	0															
	4	命令參數	0: 發送讀取郵箱的請求 1: 讀取郵箱資料															
	5	命令資料	<p>命令參數為0時:</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td>3</td> <td>2</td> <td>1</td> <td>0</td> </tr> <tr> <td colspan="2">保留</td> <td colspan="2">從站位址</td> </tr> </table> <p>命令參數為1時:</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td>3</td> <td>2</td> <td>1</td> <td>0</td> </tr> <tr> <td colspan="2">讀取資料長度</td> <td colspan="2">讀取資料起始點</td> </tr> </table>	3	2	1	0	保留		從站位址		3	2	1	0	讀取資料長度		讀取資料起始點
3	2	1	0															
保留		從站位址																
3	2	1	0															
讀取資料長度		讀取資料起始點																
回應封包																		
標頭	1	命令碼	143															
	2	索引號	自定義															
	3	資料長度	n(僅當命令參數為1時, n = 命令資料3*64 + 命令資料2) 0(僅當命令參數為0時)															
資料段	n位元組郵箱資料(僅當命令參數為1時) 資料的起始位置是郵箱的第[命令資料1*64 + 命令資料0]個位元組																	

ECM_CMD_MBX_STAT 讀取從站接收郵箱狀態			
命令封包			
標頭	1	命令碼	144
	2	索引號	自定義
	3	資料長度	0 (當命令參數為0或2時) n(當命令參數為1時, $n = \text{ceil}(\text{從站個數}/8)$)
	4	命令參數	0: OP狀態時讀取從站接收郵箱狀態 1: 啟用或不啟用主站輪詢接收郵箱狀態 2: 查看是否啟用主站輪詢接收郵箱狀態
資料段	當命令參數為1時: n位元組是否啟用輪詢接收郵箱 第n位元組的第k個位元表示第 $[8*n+k]$ 個從站 1: 啟用輪詢接收郵箱 0: 不啟用輪詢接收郵箱		
回應封包			
標頭	1	命令碼	144
	2	索引號	自定義
	3	資料長度	n(當命令參數為0或2時, $n = \text{ceil}(\text{從站個數}/8)$)
資料段	當命令參數為0時: n位元組郵箱狀態 第n位元組的第k個位元表示第 $[8*n+k]$ 個從站的狀態 1: 接收郵箱有資料 0: 接收郵箱無資料 當命令參數為2時: n位元組是否啟用輪詢接收郵箱 第n位元組的第k個位元表示第 $[8*n+k]$ 個從站 1: 啟用輪詢接收郵箱 0: 不啟用輪詢接收郵箱		

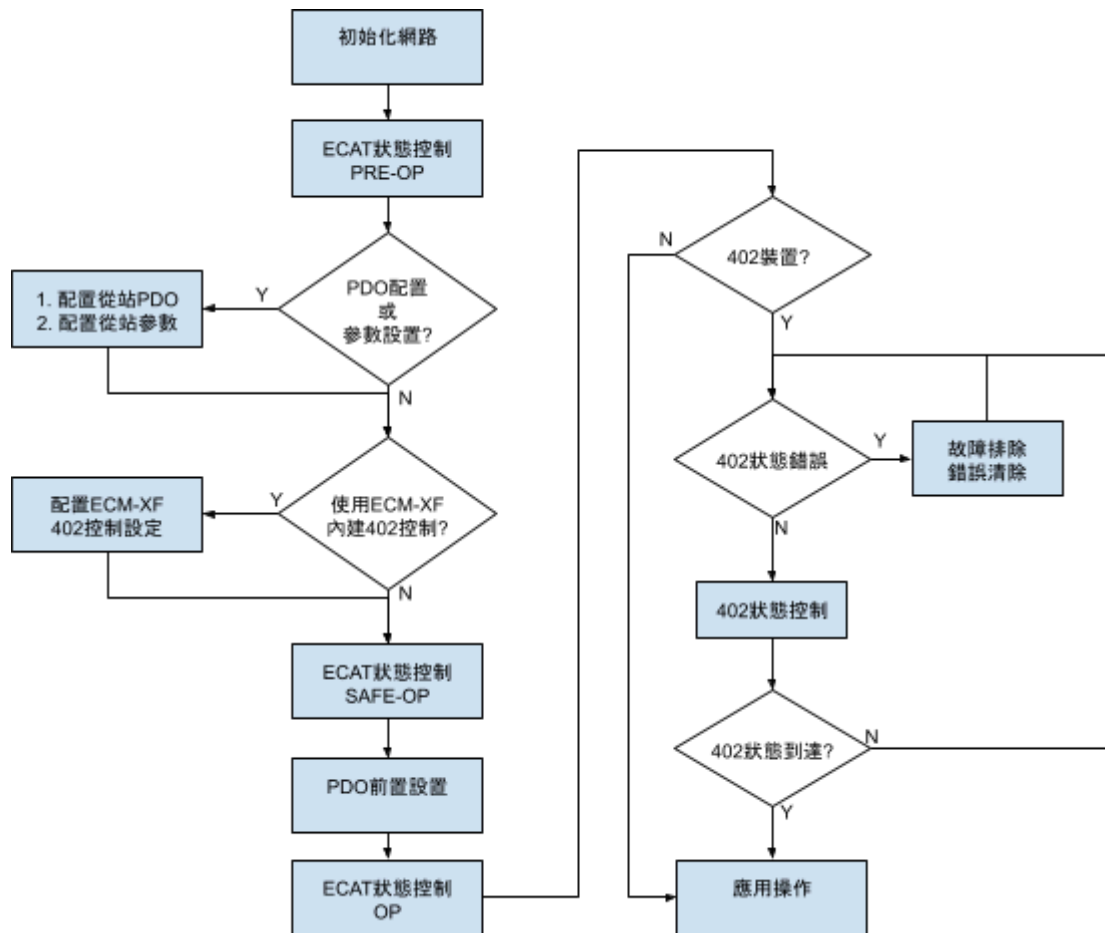
* 使ECM_CMD_MBX_STAT啟用輪詢接收郵箱功能後, 要再發送ECM_CMD_ECAT_RECONFIG_OP命令改變才會生效。

ECM-XF 命令及回應收送流程說明



圖六 命令及回應收送流程

ECM-XF 初始化EtherCAT網路及從站配置說明



圖七 EtherCAT網路初始化及配置流程

流程狀態	相關命令	說明
初始化網路	ECM_CMD_ECAT_INIT_OP	設置DC模式及時間
ECAT狀態控制	ECM_CMD_ECAT_STATE_SET ECM_CMD_ECAT_STATE_GET	切換ECAT狀態
配置從站PDO	ECM_CMD_ECAT_PDO_CONFIG_SET ECM_CMD_ECAT_PDO_CONFIG_REQ ECM_CMD_ECAT_PDO_CONFIG_GET ECM_CMD_ECAT_RECONFIG_OP	指定或配置PDO
配置從站參數	ECM_CMD_ECAT_SDO_REQ ECM_CMD_ECAT_SDO_GET	使用SDO配置參數 ex. 控制模式 扭力限制...
配置402控制	ECM_CMD_402_CONFIG_SET	始能內建402控制 設置錯誤清除
402狀態控制	ECM_CMD_402_STATE_SET ECM_CMD_402_STATE_GET	切換402狀態
PDO前置設置	ECM_CMD_FIFO_PACK_SIZE_GET ECM_CMD_ECAT_PDO_WC_GET ECM_CMD_ECAT_PDO_DATA_OP	獲取PDO資料大小 獲取PDO通訊WKC值 讀取TxPDO輸入 初始化RxPDO輸出
應用操作	ECM_CMD_ECAT_PDO_DATA_FIFO_OP ECM_CMD_ECAT_SDO_REQ ECM_CMD_ECAT_SDO_GET ECM_CMD_INFO_UPDATE_OP	透過FIFO交換PDO資料 使用SDO讀寫資料 狀態更新

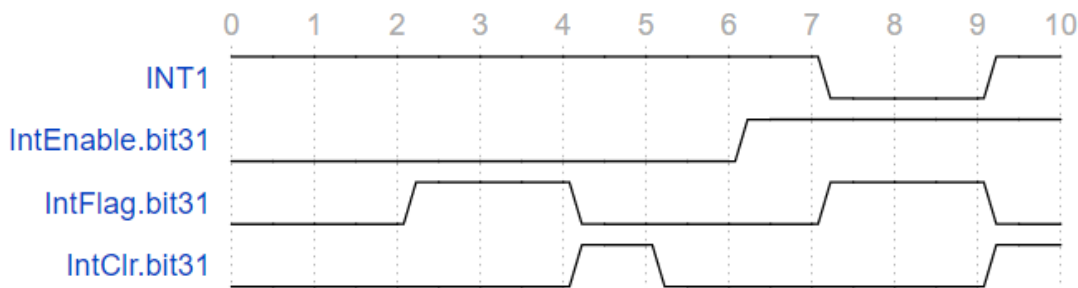
GPIO

ECM-XF的GPIO預設皆為浮動Input。

ECM-XF中斷

ECM-XF有兩個中斷輸出。當SPI就緒時預設INT0腳位為高位，可透過ECM_CMD_ECAT_INT_SET_ENABLE指令變更極性，在一般的情況下，當SPI傳輸完成後，ECM-XF會需要一小段時間處理，建議採用延時150us方式，一般處理時間 $< 150\mu s$ ，若週期時間很短(如 $250\mu s$)，則可參考INT0的訊號可以縮減延時的時間。

INT1為通用中斷功能腳位。當中斷啟用位元(ECM_CMD_ECAT_INT_SET_ENABLE)被觸發時，INT1呈激活(Enable)狀態，下方範例圖八情況為在中斷模式為低位激活(Low active)時，EtherCAT在 $t=2$ 與 $t=7$ 時接受變化



圖八、中斷範例

上圖中INT1為硬體訊號(PIN 7)

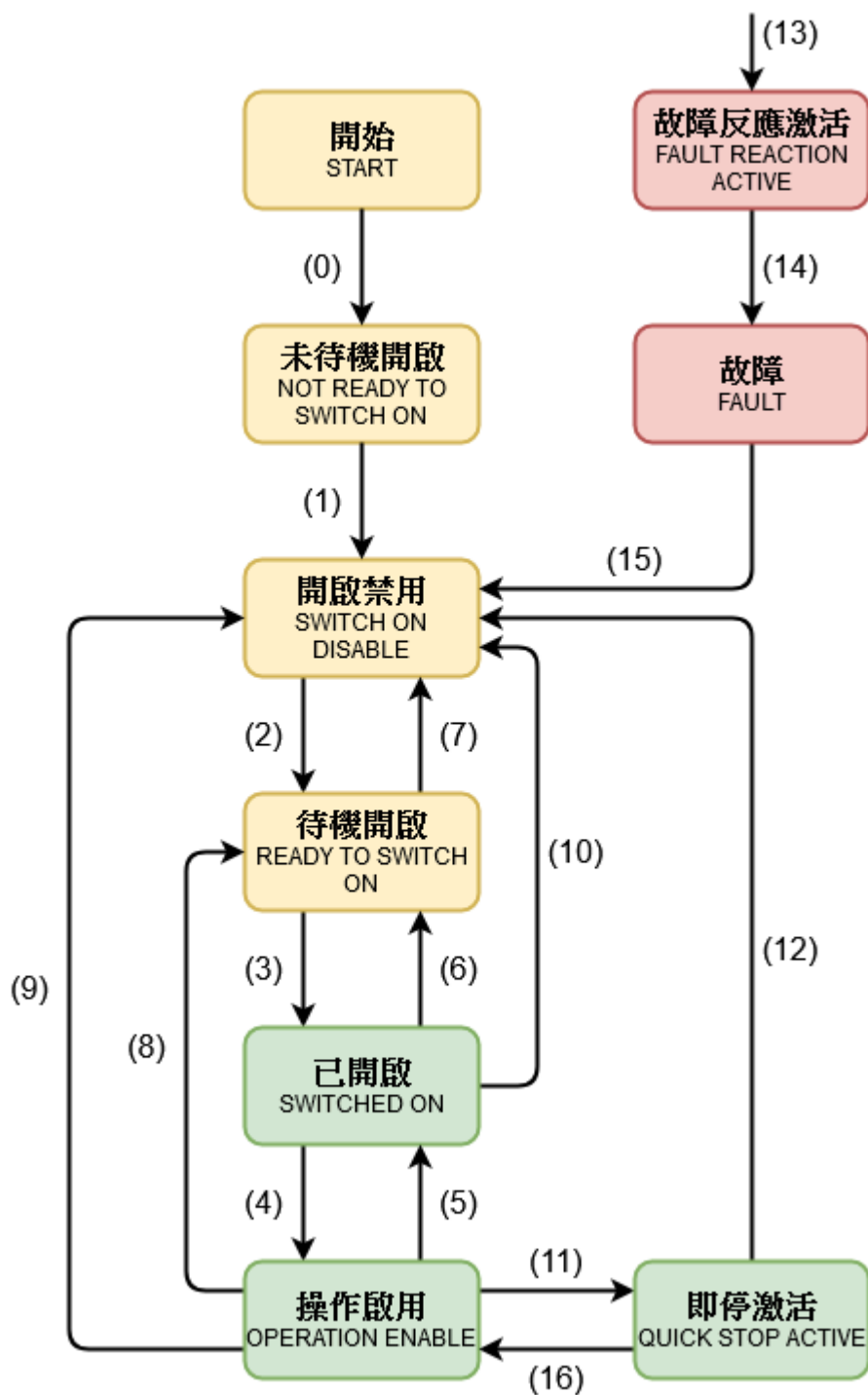
IntEnable為命令ECM_CMD_ECAT_INT_SET_ENABLE

IntFlag 位於回應標頭的0x10至0x13位置

IntClr位於命令標頭的0x10至0x13位置

EtherCAT在 $t=2$ 與 $t=7$ 時觸發。bit 31的IntFlag被設置為 $t=2$ 與 $t=7$ ，同時又在 $t=4$ 時設置IntClr(位於命令標頭0x10~0x15)。INT1只有在IntFlag (位於回應標頭0x10~0x15)與IntEnable(使用ECM_CMD_ECAT_INT_SET_ENABLE設置)同時觸發時才會激活，所以INT1只有在 $t=7$ 時激活。

附錄 CiA 402 狀態機 CiA 402 State Machine



由Status Word(0x0641:0)可得知目前的CiA402狀態

Index	Sub	Name	Data Type	Access	PDO Mapping	Default Value
0x6041	00	Status Word	UINT	RO	Y	*(See below)

Bit 0 to 3 and bit 5 to 6: for the current state of the drive

Command	bit 6	bit 5	bit 3	bit 2	bit 1	bit 0
Not ready to switch on	0	0	0	0	0	0
Switch on disabled	1	0	0	0	0	0
Ready to switch on	0	1	0	0	0	1
Switched on	0	1	0	0	1	1
Operation enabled	0	1	0	1	1	1
Quick stop active	0	0	0	1	1	1
Fault reaction active	0	0	1	1	1	1
Fault	0	0	1	0	0	0

下表為各種功能在不同狀態下可激活的形況

外部剎車只有在存在時才支援, 高位電源只有在控制器可選擇之內嵌接觸器/開關為電源使用時支援

The following table indicates which functionalities can be activated on every state. External brake can only applied if it is present, and high-level power applied is only selectable in controllers with an embedded contactor/switch for the power stage.

功能 Function	未待機開 啟 Not Ready to Switch On	開啟禁用 Switch On Disable	待機開啟 Ready to Switch On	已開啟 Switch On	操作啟用 Operation Enabled	即停激活 Quick Stop Active	故障反應 激活 Fault Reaction Active	故障 Fault
剎車 Brake applied, if present	是 Yes	是 Yes	是 Yes	是 Yes	是/否(*) Yes/No(*)	是/否(*) Yes/No(*)	是/否(*) Yes/No(*)	是 Yes
低位電源應 用 Low-level power applied	是 Yes	是 Yes	是 Yes	是 Yes	是 Yes	是 Yes	是 Yes	是 Yes
高位電源應 用 High-level power applied	是/否(**) Yes/No(**)	是/否(**) Yes/No(**)	是/否(**) Yes/No(**)	是 Yes	是 Yes	是 Yes	是 Yes	是/否 (**) Yes/N o(**)
驅動功能啟 用 Drive function enabled	否 No	否 No	否 No	否 No	是 Yes	是 Yes	是 Yes	否 No
允許配置 Configuration allowed	是 Yes	是 Yes	是 Yes	是 Yes	否 No	否 No	否 No	是 Yes
分流控制已 啟用 Shunt control enabled	否 No	否 No	否 No	否 No	是 Yes	是 Yes	是 Yes	否 No

(*)在某些狀態下，剎車會被手動激活或非激活

In some states, brake could be activated and/or deactivated manually.

(**)在某些控制器上，高位開關並未顯示且可能沒有非激活

In some controllers, high-level switch is not present and therefore high level power could not be deactivated.

流程 Transition	事件 Event	動作 Action
0	上電或重置後自動流程 Automatic transition after power-on or reset application.	驅動裝置自動測試或自動初始化 Drive device self-test and/or self initialization is performed.

1	<p>初始化後自動流程 Automatic transition after initialization.</p>	<p>通訊激活 Communications are activated.</p>
2	<p>從控制裝置或本地訊號接收「斷電命令」 Shutdown command received from control device or local signal.</p>	-
3	<p>從控制裝置、激活啟用訊號或本地訊號接收「開啟命令」 Switch on command received from control device and enable signal is activated (if available) or local signal.</p>	<p>高位電源開啟 The high-level power is switched on.</p>
4	<p>從控制裝置、激活啟用訊號或本地訊號接收「操作啟用」 Enable operation command received from control device and enable signal is activated (if available) or local signal.</p>	<p>啟用驅動功能，執行初始角度確定過程並清除所有內部設定點。 The drive function is enabled, Initial angle determination process is executed and all internal set-points cleared.</p>
5	<p>從控制裝置、非激活啟用訊號或本地訊號接收「操作禁用」 Disable operation command received from control device or enable signal is deactivated (if available) or local signal.</p>	<p>驅動功能禁用 The drive function is disabled.</p>
6	<p>從控制裝置、非激活啟用訊號或本地訊號接收「斷電命令」 Shutdown command received from control device or enable signal is deactivated (if available) or local signal.</p>	<p>高位電源關閉 The high-level power is switched off.</p>
7	<p>從控制裝置或本地訊號接收「即停」或「禁用電壓」 Quick stop or disable voltage command received from control device or local signal.</p>	-

8	<p>從控制裝置或非激活啟用訊號接收「斷電」 Shutdown command received from control device or enable signal is deactivated (if available).</p>	<p>驅動功能禁用、高位電源關閉 The drive function is disabled, and the high-level power is switched off.</p>
9	<p>從控制裝置或本地訊號接收「禁用電壓命令」 Disable voltage command from control device or local signal.</p>	<p>驅動功能禁用、高位電源關閉 The drive function is disabled, and the high-level power is switched off.</p>
10	<p>從控制裝置或本地訊號接收「禁用電壓」或「即停」 Disable voltage or quick stop command received from control device or local signal.</p>	<p>高位電源關閉 The high-level power is switched off.</p>
11	<p>從控制裝置或本地訊號接收「即停」 Quick stop command received from control device or local signal.</p>	<p>即停功能開始 The quick stop function is started.</p>
12	<p>當從控制裝置接收「即停」功能完成或「電壓禁用命令」時自動流程 Automatic transition when the quick stop function is completed or disable voltage command is received from the control device (depending on 0x605A - Quick stop option code).</p>	<p>驅動功能禁用、高位電源關閉 The drive function is disabled, and the high-level power is switched off.</p>
13	<p>「故障」訊號 Fault signal.</p>	<p>配置故障反應執行 The configured fault reaction function is executed.</p>
14	<p>自動流程 Automatic transition.</p>	<p>驅動裝置禁用、高位電源關閉 The drive function is disabled, and the high-level power is switched off.</p>

15	從控制裝置或本地訊號接收「故障重置命令」 Fault reset command received from control device or local signal.	當無故障發生時驅動裝置重置狀態執行；當離開故障狀態後，在控制字元中的故障重置位元會被控制裝置清除 A reset of the fault condition is carried out, if no fault exists currently on the drive device; after leaving the Fault state, the Fault reset bit in the controlword is cleared by the control device.
16	從控制裝置接收「操作命令啟用」 Enable operation command from control device.	驅動功能啟用 The drive function is enabled.

注意 : Caution:

※當驅動模式為禁用時，馬達並不會上電。目標與設定點在此狀態下不會運作

When the drive function is disabled, no energy will be supplied to the motor. Target or set-point(torque, velocity, position) in that situation are not processed.

※在有對應的接觸器或開關的系統中，高位電源才能被關閉

High-level power is switched off only in systems with contractors or switches for this purpose.

※啟用訊號只會被可用的對應註冊影響，詳見啟用/禁用輸入

Enable signal will affect only if it is marked as available in the corresponding register. See Enable/Disable input for further information.

更新紀錄

版本	更新時間	敘述
03	2020.10.15	更新版本對照至英文版Ver.03 (更新中斷、ADC、DAC、GPIO、QEI與FIFO clear功能並重新排序所有敘述順序)
031	2020.10.15	更新 set/get Tx/Rx FIFO count
032	2021.01.18	更新0x09錯誤命令狀態 更新ECM_CMD_ECAT_PDO_CONFIG_SET命令說明
033	2021.01.29	更新 ECM_CMD_402_CONFIG_SET ECM_CMD_402_STATE_SET ECM_CMD_402_STATE_GET ECM_CMD_402_CTL_SET

		ECM_CMD_402_CTL_GET
034	2021.02.18	SPI封包資料長度修正
035	2021.03.11	修正timeout單位(μ s)、ECM_CMD_CRC_TYPE_SET 新增ECM_CMD_ECAT_INIT_DC_OP、 ECM_CMD_ECAT_STATE_UPDATE
036	2021.07.09	新增ECM_CMD_FIFO_INIT 修正ECM_CMD_ECAT_INIT_OP說明 新增DAC delay time說明
037	2021.07.29	新增ECM_CMD_ECAT_INT_SET_ENABLE 與 ECM_CMD_ECAT_INT_GET_ENABLE 說明 修正中斷圖表說明
038	2021.08.04	新增ECM_CMD_ECAT_WKC_CONTI_ERR_MAX_GET 與 ECM_CMD_ECAT_WKC_CONTI_ERR_MAX_SET
038	2021.08.10	新增ECM_CMD_402_CONFIG_SET內部402狀態切換
039	2021.09.22	新增GPIO
040	2021.10.22	更新ECM_CMD_402_CTL_SET
041	2021.12.30	更新回應封包標頭欄位錯誤/狀態說明 更新命令封包標頭欄位控制字元說明 更新ECM_CMD_ECAT_PDO_DATA_OP
042	2022.02.14	更新單位說明
043	2022.05.14	更新ECM_CMD_ASYNC_CMD_RESET說明
044	2022.08.08	修正ECM_CMD_ECAT_WKC_CONTI_ERR_MAX_SET說明 新增標頭檔GPIO欄位說明
047	2023.03.15	更新部分命令說明
048	2024.01.01	新增SDO Abort Code說明 新增ECM_CMD_ECAT_SET_FIFO_TH說明及新命令說明
049	2024.05.22	修正ECM_CMD_ECAT_INT_SET_ENABLE命令說明