# ECM-XF/ECM-XFU範例程式使用導引

訊成科技股份有限公司

V.06 2022.12.25

ECM-XF範例列表	3
ECM-XFU-SK / ECM-XF-PCIe範例列表	3
STM32範例環境建立導引(NUCLEO-F401RE)	4
STM32 F401RE 腳位建立導引	20
STM32 H7 腳位建立導引	20
新唐範例環境建立導引	22
修改範例程式為實際應用導引	25
常見錯誤排除	29
維護紀錄	30

# 範例程式說明

ECM-XF提供針對不同應用的多種範例,使用者可依實際需要選擇一個類似的範例參考,並修改成適合的應用程式。

若不是使用STM32或是新唐的控制晶片,則僅需替換SPI控制層韌體(與實際硬體相關),及控制晶片實際SPI連接腳位之設定,其餘部分仍可參考範例程式。XF與XFU除了介面不同,預設的SPI Data Size 不同外,其餘控制流程皆相同,因此各平台間的範例可相互參考使用。

	與上位的介面	預設SPI Data Size	SPI Data Size 範圍
ECM-XF IC	SPI	112 Bytes	32~1408 Bytes
ECM-XF-PCle	PCle	112 Bytes	32~1408 Bytes
ECM-XFU-SK	USB 或 SPI (擇一)	996 Bytes	使用SPI介面32~1408Bytes 使用USB介面固定992Bytes

### ECM-XF範例列表

提供STM32 F401與新唐M487的範例, STM32或新唐透過SPI介面與EXM-XF溝通, STM32的範例路徑為Core\ECM\example, 新唐範例路徑為User\example

- ex\_CSP 驅動器馬達配置及CSP (Cyclic Sync Position mode)模式範例
- ex\_PP 驅動器馬達配置及PP (Profile Position mode)模式範例
- ex\_PV 驅動器馬達配置及PV (Profile Velocity mode)模式範例
- ex\_PT 驅動器馬達配置及PT (Profile Torque mode)模式範例
- ex\_home 驅動器馬達配置及Homing (Homing mode)模式範例
- ex\_rta 驅動器馬達Real Time Application範例

### ECM-XFU-SK / ECM-XF-PCIe範例列表

提供Windows電腦透過USB介面與ECM-XFU-SK溝通的範例, 並有C語言與C#與語言 兩種, 可透過微軟Visual Stdio邊程軟體進行編譯。

- DA\_AD\_QEI\_LIO 晶片上DAC、ADC、QEI及GPIO的功能範例
- Drive\_CSP 驅動器馬達配置及CSP (Cyclic Sync Position mode)模式範例
- Drive\_CSP\_500us 驅動器馬達配置及CSP (Cyclic Sync Position mode)模式, 週期時間500us範例 此範例當中每次傳送2筆(TEST\_PDO\_TO\_FIFO\_ONCE = 2)週期命令資料
- Drive\_CSV 驅動器馬達配置及CSP (Cyclic Sync Velocity mode)模式範例
- Drive\_Homing\_PP 驅動器馬達配置及Homing (Homing mode)模式、PP (Profile Position mode)模 式範例
- Drive\_IO 驅動器馬達配置及IO類型從站混合範例
- EEPROM 主站透過SII(Slave Information Interface)讀取從站EEPROM資料範例
- IO
   IO類型從站操作範例
- SimpleTest / Sample
   不配置從站(使用預設PDO)進入OP後直接交換資料,若FIFO有累積將顯示狀態

請注意, XF\_SetDevType的參數將決定USB介面或是PCIe介面, 請務必確認此參數正確。

### STM32範例環境建立導引(NUCLEO-F401RE)

- 1. 開啟STM32Cube IDE.
- 2. 在左側視窗點擊滑鼠右鍵並選擇"STM32 Project" 或選擇 "Create a New STM32 project"



3. 選擇所使用的板子。這邊例子以STM32 NUCLEO-F401RE為例



4. 選擇完成後點選下一步

The STM - STM 32CubelDE Ne Edit Source Refector Navigate Search P + G2 (0) N + % + (0) 1   1   1   1   1   1   1   1   1   1	STM32 Project					- of X
Project Dolover II P 9, 7 0	elect STM32 torget				IUL	C Se Outine IX (R Build Taro, C D
> 115 > 116 > 116 > 117 > 117 118 \ 118 \ 119	MCUMPU Selector Board Selector Course Selector	Features	Large Picture Docs & Ress	unces 🖨 Datasbeet	T Buy	There is no active editor that provides an outline.
5 11 120 5 11 121 5 11 122 5 11 121 5 11 1	Part Number Search	NUCLEO-F401RE		ŭ		
> C STM125PLECM07_GPID	Q NUCLEO-F401R V		STMicroelectronics NUCLEO-F40	IRE Board Support and Examples		
	Vendor >	STM22 F4	ACTIVE ACTIVE	Unit Price (US\$): \$3.0		
	Type >		Product is in mass production	Mounted device: STHE32F401RETs		
	MCUMPU Series		The STM32 Nucleo-64 bo new concepts and build pr	ards provide an affordable and flexible way for u tototypes by choosing from the various combina	isers to try out tions of	
	Other V		For the compatible boards	onsumption features, provided by the STM32 mi s, the external SMPS significantly reduces power	crocontroller.	
	Price = 13.0		The Arduino** Uno V3 co	nectivity support and the ST morpho headers a	dow the easy	
	Oscillator Freq. = 0 (MHz)		wide choice of specialized	shields.		
		Boards List: 1 item			🔥 Export	
	Peripheral ~	1 Overview	Part No 🗢 Type	Manadog Status Unit Price (233)	Mounted Device	
	Accelersmeter     0     0     Analog I/O     0     Adulas Form Factor     0     0	*	EO-F401RE Nucleo64	Active 13.0	879-329-4410875	
	Addio Line In 0 0 Addio Line Out 0 0 Battery					aly. A Static Stack. # Debug (S) = D
	OButton     O     O     O					OT WITE
	Camera Company					
	Oustom Form Factor     0     0     Digital I/O     0     108					
	Merana -					
						至[股定]以啟用 Windows。
< Diterrs selected	2			< Back Next >	Finish Cencel	1

5. 輸入專案名稱與專案位置

DE STM32 Pro	ject	_		×
🔥 Project loca	ation already exists		I	DE
Project				
Project Nam	e: 123			
🗌 Use defau	It location			
Location:	E:\FireFox Download\123		Brows	se
C     C     Targeted     Execut	C++ Binary Type able O Static Library Resist Tune			
<ul> <li>STM32</li> </ul>	Cube C Empty			
?	< Back Next > Finish		Cance	•

#### 6. 選擇韌體並點選完成。如果沒有目前所選擇的版本,系統會自動下載所選擇的版本

a simisz Project					×
irmware Library	Package Setup				IDE
Setup STM32 targ	jet's firmware				IVE
Target and Firmv	vare Package				
Target Reference	e.	NUCLEO-F	401RE		
Firmware Packag	e Name and Versio	on: STM32Cub	e FW_F4 V1.25.	1 ~	
Firmware packag	e Repository				
Location:					
C:\Users\dttb1\S	TM32Cube\Repos	itory			
C:\Users\dttb1\S See <u>'Firmware U</u>	TM32Cube\Repos <u>pdater'</u> for setting:	itory s related to firm	ware package in	stallation	
C:\Users\dttb1\S See <u>'Firmware U</u>	TM32Cube\Repos	itory s related to firm	ware package in	stallation	
C:\Users\dttb1\S See <u>'Firmware U</u> Code Generator	TM32Cube\Repos <u>pdater'</u> for settings Options	itory s related to firm	ware package in	stallation	- Cl-
C:\Users\dttb1\S See <u>'Firmware U</u> Code Generator O Add necessary	TM32Cube\Repos <u>pdater'</u> for setting: Options / library files as refe	itory s related to firm rrence in the to	ware package in olchain project c	onfiguratic	on file
C:\Users\dttb1\S See <u>'Firmware U</u> Code Generator O Add necessary Copy all used	TM32Cube\Repos <u>pdater'</u> for setting: Options / library files as refe libraries into the p	itory s related to firm erence in the to roject folder	ware package ir olchain project c	istallation onfiguratic	on file
C:\Users\dttb1\S See <u>'Firmware U</u> Code Generator O Add necessary Copy all used @ Copy only the	TM32Cube\Repos pdater' for setting: Options I library files as refe libraries into the p necessary library fi	itory s related to firm erence in the to roject folder les	ware package ir olchain project c	istallation onfiguratic	n file
C:\Users\dttb1\S See <u>'Firmware U</u> Code Generator O Add necessary Copy all used © Copy only the	TM32Cube\Repos pdater' for setting: Options / library files as refe libraries into the pr necessary library fi	itory s related to firm rence in the to roject folder les	ware package ir olchain project c	nstallation onfiguratic	on file
C:\Users\dttb1\S See <u>'Firmware U</u> Code Generator O Add necessary Copy all used @ Copy only the	TM32Cube\Repos <u>pdater</u> for setting: Options / library files as refe libraries into the pr necessary library fi	itory s related to firm erence in the to roject folder les	ware package ir olchain project c	onfiguratic	on file
C:\Users\dttb1\S See <u>'Firmware U</u> Code Generator O Add necessary Copy all used @ Copy only the	TM32Cube\Repos <u>pdater</u> for setting: Options / library files as refe libraries into the pr necessary library fi	itory s related to firm rence in the to roject folder les	ware package in olchain project c	istallation onfiguratic	on file
C:\Users\dttb1\S See ' <u>Firmware U</u> Code Generator O Add necessary Copy all used @ Copy only the	TM32Cube\Repos pdater' for setting: Options I library files as refe libraries into the pr necessary library fi	itory s related to firm rence in the to roject folder les	ware package in	onfiguratic	n file

7. 視窗詢問初始化所有外部裝置至預設模式時選擇"是"

DE STM32 Project			×
Firmware Library Package Setup			IDE
Setup STM32 target's firmware			IVE
Target and Firmware Package			
Target Reference: NUCLEO-F401RE			
Firmware Package Name and Version: STM32Cube FW_F4	1.25.1	~	
Board Project Options:			×
Initialize all peripherals with their default Mode ?		N	lo
Code Generator Options Add necessary library files as reference in the toolchain proj Copy all used libraries into the project folder © Copy only the necessary library files	ect cor	figuratio	on file
Perform Project Creation. Please Wait For Completion			

#### 8. 詢問開啟STM32CubeMx時選擇"是"並開始定義腳位

	hunny Dackage Set				
	Drary Package Sett	, p			D
Setup STM:	32 target's firmware				
- Target and	Firmware Package				
Target Ref	erence:	NUCLEO-F	401RE		
Eirmware P	ackage Name and V	ersion: STM32Cub	e FW F4 V1.25.1	$\sim$	
Open Asso	ciated Perspective?				
This k want	ind of project is asso to open this perspec	ciated with the STI tive now?	VI32CubeMx pers	pective. D	o you
	1				
Remembe	r my decision				
Remembe	er my decision		Yes	N	0
Remembe	r my decision erator Options		Yes	N	0
Remembe Code Gene	er my decision erator Options :essary library files as	reference in the to	Yes olchain project co	N onfiguratio	<b>o</b> n file
Remember	er my decision erator Options cessary library files as l used libraries into th	reference in the to	Yes olchain project co	No nfiguratio	o n file
Code Gene Add nec Copy al Copy or	er my decision erator Options :essary library files as I used libraries into th nly the necessary libra	reference in the to ne project folder ary files	<b>Yes</b> olchain project co	Nonfiguratio	o n file
Code Gene Add nec Copy al Copy or	r my decision erator Options essary library files as I used libraries into th nly the necessary libra	reference in the to ne project folder ary files	Yes olchain project co	<b>N</b>	o n file
Remembe Code Gene Add nec Copy al Copy or Perform Proje	rr my decision erator Options essary library files as I used libraries into th hly the necessary libra ect Creation. Please V	reference in the to ne project folder any files Nait For Completio	Yes olchain project cc n	N-	o n file
Remember Code Gene Add nec Copy al Copy or Perform Proje	rr my decision erator Options essary library files as l used libraries into th nly the necessary libra ect Creation. Please V	reference in the to ne project folder ary files Wait For Completio	Yes olchain project cc n	N	o n file
Remember Code Gene Add nec Copy all Copy or Perform Proje	rr my decision erator Options essary library files as used libraries into th nly the necessary libra ect Creation. Please \	reference in the to re project folder ary files Wait For Completio	Ves olchain project cc n	Nonfiguratio	o n file



9. 在腳位定義圖中,選擇PA5為SPI1\_SCK、PA6為SPI1\_MISO、PA7為SPI1\_MOSI與 PB6為GPIO\_Output

### 10. 點選左側"System Core"對GPIO進行進階設定,將"GPIO output level"調整至"High"

Q ~ Ø	GPIO Mode and Configuration			
Categories A->Z	Configuration			
System Core 🗸 🗸	Group By Peripherals	$\sim$		
¢ DMA	♀ SYS     ♀ USART     ♀ NVIC       ♀ GPIO     ♀ Single Mapped Signals     ♀ RCC			
GPIO IWDG NVIC VRCC SYS WWDG	Search Signals         Search (CrtI+F)         Pin Signal o GPIO o GPIO m GPIO P Maximu User La Modified P         PB6       n/a       High       Output No pull Low       ✓         PC13-A n/a       n/a       Externa No pull n/a       B1 [Blu ✓	'ins ied		
Analog >				
Timers >				
Connectivity >	PB6 Configuration :	_		
Multimedia >	GPIO output level High	2		
Computing >	GPIO mode Output Push Pull	~		
Middleware >	GPIO Pull-up/Pull-down No pull-up and no pull-down	~		
	Maximum output speed			
	User Label			

# 11. 點選"Timers"中的TIM2並將"Clock Source"改為"Internal Clock"後, 將下方"Counter Period"數值改為"0xFFFFFFF"

Q	~ 🔅	TIM2 Mode and Configuration	
Categories A->Z		Mode	
System Core	>	Slave Mode Disable 🗸 🗸	
		Trigger Source Disable 🗸 🗸	
Analog		Clock Source Internal Clock ~	
Timers	~	Channel1 Disable ~	
\$		Channel2 Disable ~	
RTC TIM1		Channel3 Disable ~	
🔥 TIM2		Channel4 Disable ~	
TIM3		Combined Channels Disable	
▲ TIM5		Use ETR as Clearing Source	
▲ TIM9		C XOR activation	_
TIM10		Configuration	
		Reset Configuration	
Connectivity	>	📀 User Constants 🛛 📀 NVIC Settings 🛛 📀 DMA Settings	
		Parameter Settings	
Multimedia	>	Configure the below parameters :	_
Computing	>	Q Search (CrtI+F)	
NAT JUL		✓ Counter Settings	
widdleware		Counter Mode Up	
		Counter Period (AutoReload 0xfffffff	
		Internal Clock Division (CKD) No Division	
		auto-reioau preioau Disable	

# 12. 接下來跳至"Connectivity"設定"SPI1", 模式設定"Full-Duplex Master"並將鮑率設置在 2~8之間

Q ~	٢	SPI1 Mode and Configuration
Categories A->Z		Mode
System Core	>	Mode Full-Duplex Master ~
Analog	>	Hardware NSS Signal Disable ~
Timers	>	
Connectivity	~	
<ul> <li>€</li> <li>12C1</li> <li>⊘ 12C2</li> <li>12C3</li> <li>SDIO</li> <li>✓ SPI1</li> </ul>		
SPI2		Configuration
SPI3 USART1		Reset Configuration
VUSARI2		📀 DMA Settings 🛛 📀 GPIO Settings
USB OTG FS		Parameter Settings Over Constants Over NVIC Settings
		Frame Format Motorola Data Size 8 Bits
Multimedia	>	First Bit MSB First
Computing	>	Prescaler (for Baud Rate) 8
Middleware	>	Clock Polarity (CPOL) Low Clock Phase (CPHA) 1 Edge V Advanced Parameters
		CRC Calculation Disabled

### 13. 接下來跳到USART2設定

Pino	ut & Configuration	Clock Configurati
		✓ Software Pack
Q ~	USART2 Mod	le and Configuration
Categories A->Z		Mode
NVIC RCC SYS WWDG	Mode Asynchronous Hardware Flow Control (RS232) Disab	le v
Analog >		
Timers 🗸		
<ul> <li>RTC</li> <li>TIM1</li> <li>TIM2</li> <li>TIM3</li> <li>TIM4</li> <li>TIM5</li> <li>TIM9</li> <li>TIM10</li> <li>TIM11</li> </ul>	Cor Reset Configuration	figuration
Connectivity ~	<ul> <li>NVIC Settings</li> <li>⊘ DN</li> <li>⊘ Parameter Settings</li> </ul>	IA Settings
I2C1 I2C2 I2C3 SDIO ✓ SPI1 SPI2 SPI3 USART1 ✓ USART2 USART6 USB_OTG_FS	Configure the below parameters : Q Search (Ctrl+F) ③ ③ > Basic Parameters Baud Rate Word Length Parity Stop Bits > Advanced Parameters Data Direction Over Sampling	115200 Bits/s 8 Bits (including Parity) None 1 Receive and Transmit 16 Samples

14. 完成上面步驟後, 點擊儲存標示儲存並自動產生程式

DE Question			×
Do you want generate Code	e?		
Remember my decision			
		Yes	No

📑 🗕 🖪 🕞 📘

15. 如果在生成的main.c "Private variables"中看到剛設定的SPI、TIM2表示有之前的設定 有完成, 若未看到相關資訊, 可先返回

```
/* Private variables -----*/
SPI_HandleTypeDef hspi1;
```

TIM\_HandleTypeDef htim2;

UART\_HandleTypeDef huart2;

16. 點開"main.h"並依下圖加入下方程式碼

```
/* USER CODE BEGIN Includes */
#include "platform.h"
#include "EcmUsrDriver.h"
/* USER CODE END Includes */
/* Exported macro -----*/
/* USER CODE BEGIN EM */
#ifndef PRINTF
#define PRINTF( str, ...) \
   do{ \
       int n; \
       n = sprintf( printbuf, (str), ## VA ARGS ); \
       HAL_UART_Transmit( &huart2, (uint8_t *)printbuf, n, 0xffffffff); \
   }while(0)
#endif
#ifndef GETCHAR
#define GETCHAR userGetchar
#endif
/* USER CODE END EM */
/* USER CODE BEGIN EFP */
extern UART HandleTypeDef huart2;
extern char printbuf[];
/* USER CODE END EFP */
```

17. 開啟"main.c"並加入下方程式碼

```
/* USER CODE BEGIN PFP */
char printbuf[128];
int main_ini(void);
/* USER CODE END PFP */
/* USER CODE BEGIN 2 */
main_ini();
/* USER CODE END 2 */
/* USER CODE BEGIN TIM2_Init 2 */
HAL TIM Base Start(&htim2);
```

```
/* USER CODE END TIM2 Init 2 */
```

- 18. 複製EcmDriver.h、EcmUsrDriver.h、PdoDefine.h、platform.h與Utility.h(部分範例)貼入 "Inc"資料夾
- 19. 複製crc32.c(部分範例無此檔案)、EcmUsrDriver.c、main\_ini.c(主程式:部分範例為不同 名)與platform.c、Utility.c(部分範例)貼入Src資料夾
- 20. 已準備好執行範例程式

20. 點擊槌子圖示, 並選取欲執行的程式來進行Build, 或是點擊Project底下的Build all來 建置程式



workspace\_1.9.0 - STM/Core/ECM/example/ex\_csp/PdoDefine.h - STM32CubeIDE

The Fair Source Heinefor Handare Seficit	Tiolect Fan Window Helb	
📑 🕶 🔚 🐚   🥸 🕶 🐔 🕶 🗟 ! 🔌   🕪 💷 🔳	Open Project	▼ 🖸 ▼ 🮯 ▼   券 ▼ 🔘 ▼ 🂁 ▼   🥭 🖋 ▼   ⊿ 💀 🗐 ୩   월 ▼ 🖓 ▼ や 다 ヤ ▼   🛃   🚹
Project Explorer 🗙	Close Project	c) EcmUsrDriver.h € PdoDefine.h ⊨ PdoDefine.h ×
<ul> <li>▼ III STM</li> <li>&gt; ∰ Binaries</li> <li>&gt; ∭ Includes</li> <li>&gt; Go Core</li> <li>&gt; Go ECM</li> <li>&gt; Go ECM</li> </ul>	Build All Ct Build Configurations Build Project Build Working Set Clean	rH+B File : PdoDefine.h Version : 1.52 Date : 2021/11/23 Author : XFORCE
✓ 🦢 example	Build Automatically	Define slave device PDO data structure
> ⊱ ex_home > ⊱ utility ❤ ∅ ex_csp	Build Targets C/C++ Index	<ul> <li>You can configure the PDO structure you wanted, or you can just use the default PDO structure.</li> <li>No matter what, the example program need to know.</li> </ul>
⊠ main.c  ≫ Makefile  ≫ PdoDefine.h	Generate Report Generate Code	@copyright (C) 2020 NEXTW TECHNOLOGY CO., LTD All rights reserved.
> 💋 ex_pp > 💋 ex_pt	Properties	Indef _ECM_USR_POD_DEFINE_H
<ul> <li>jercita</li> &lt;</ul>	(S N	<pre>20 20 21 #define TEST_DRV_CNT 2 22 #define TEST_DC_WT 0 23 #define COUPLER_CNT 0 24 #define TEST_SLV_CNT (TEST_DRV_CNT+COUPLER_CNT+TEST_ID_CNT) 25 26 #define TEST_ST_WORD_OFFSET 0 27 #define TEST_ST_WORD_OFFSET 0 28 #define TEST_MODEOP = 4) 31 #if (TEST_MODEOP = 4) 32 #define TARGET_TYPE intl6_t 33 #define TARGET_TYPE intl6_t 35 #define TARGET_TYPE intl6_t 36 #define TARGET_TYPE intl3_t 37 #endif 38 #define TARGET_TYPE intl3_t 39 #typedef structattribute_((packed)) _axis_rxpdo_st_def_t 40 { 41 uintl6_t ui6CtlWordj // 0x6x40 - Control word 42 TARGET_TYPE Target; // 0x6x74 - Target position 43 // 0x6xFF - Target velocity 44 // 0x6xFF - Target velocity 45 // 0x6xFF - Target tologue  Problems @ Tasks @ Console x @ Properties to consoles to display at this time.</pre>

點擊後會開始建置build

#### 在Console欄中顯示完成訊息 (若有錯誤請檢查錯誤原因並排除,<mark>注意程式路徑不可包含非英文語系文字</mark>)





21.下一步要進行Debug,在上方工具列中點擊蟲子的圖示,並點選Debug Configurations.

在Debug Configurations中點選STM32, 並選擇要Debug的程式再按下Debug就會開始偵錯, 若沒有出現請按上方加的圖示(New launch configurationM, 紅框處)

Debug Configurations			
eate, manage, and run configurations			Ŕ
Image: Status       Image: Status<			
> ter matched 13 of 13 items		Revert	Apply
ð		Debug	Close

程式會下載至MCU中並在Console顯示相關訊息

	🖹 Problems 🧔 Tasks 📮 Console 🗙 🔲 Properties	ж	*	B <sub>k</sub>	N	P	1	- 📬	· -	
	STM ex_home_debug [STM32 Cortex-M C/C++ Application]									
										~
	Verifying									
	Developed constitued accordingly.									
	Download Verified successfully									
										$\checkmark$
	<									>
Î					 					

22. 開啟序列通訊埠程式(如Putty),並在Connection type中勾選Serial

🕵 PuTTY Configuration	? ×	🕵 PuTTY Configuration		? ×
PuTTY Configuration         Category:         Session         Logging         Terminal         Keyboard         Bell         Features         Window         Appearance         Behaviour         Translation	? ×           Basic options for your PuTTY session           Specify the destination you want to connect to           Host Name (or IP address)           Port           [           22           Connection type:           Image: SSH           Serial           Other:           Telnet           Saved Sessions	PuTTY Configuration     Category:     - Session     - Logging     - Terminal     - Keyboard     - Bell     - Features     - Window     - Appearance     - Behaviour     - Translation	Basic options for your PuTTY sessio Specify the destination you want to connect to Serial line Sp COM1 9 Connection type: SSH • Serial Other: Telnet Load, save or delete a stored session Saved Sessions	? × in
Connection     Proxy     SSH     Serial	Saved Sessions  Default Settings COM6 Save Delete	Colours Colours Connection Data Proxy SSH Serial Telest	Saved Sessions           Default Settings           COM6	Load Save Delete
About Help	Close window on exit: Always Never Only on clean exit	About Help	Close window on exit: Always Never  Only on clean Open	n exit Cancel

#### 再在Serial line中輸入對應到的COM port(可以在裝置管理員中的連接埠確認,本例為COM4)



設定好Serial line後, speed輸入 115200, 接著就可以點擊Open

🕵 PuTTY Configuration		? ×
Category: Session Logging Terminal Keyboard Bell Features Window Appearance Behaviour	Basic options for your PuTTY s Specify the destination you want to connection COM4 Connection type: OSSH OSerial Other: Teln Load, save or delete a stored session	ect to Speed 115200
Translation     Translation     Translation     Selection     Colours     Onnection     Data     Proxy     SSH     Serial     Telnet	Saved Sessions Default Settings COM6	Load Save Delete
SUPDUP	Close window on exit: Always Never Only on o	clean exit
About Help	Open	Cancel

23.進入到Putty或其他RS232序列通訊公用程式,開始執行(按下STM F401RE版上的reset 鍵,或點選STM32CubeIDE中[執行]按鈕),畫面可得程序中printf的文字



確認ECM-XFU-SK已連結至從站,且網路口燈亮起,按下鍵盤上任意鍵就會開始執行程式



有出現版本號表示成功代表SPI連線傳輸正常

若出現failed訊息可能為接觸不良, 調整版子接合處並按下ECM-XF-SK上的reset鍵重新執行

## STM32 F401RE 腳位建立導引



## STM32 H7 腳位建立導引

STM32所生成之main.c與main.h不與F4版本共用,請使用H7範例為範本修改 請使用H7專用platform.h與platform.c

需要F4版本範例可以直接複製main\_ini.c覆蓋至H7 main\_ini.c



### 新唐範例環境建立導引

- 1. 至新唐官網依作業系統下載Nueclipse GCC(開源IDE), 連結為官方下載頁面: <u>M487 Ethernet Series - Nuvoton</u>
- 2. 在同一頁面下載Nu-Link Keil driver
- 3. 同一頁面下載M480\_BSP\_CMSIS
- 4. 安裝IDE與Keil driver
- 5. 下載範例程式並放入workspace位置
- 6. 開啟Eclipse並在Project Explorer的位置滑鼠右鍵選擇"Import"
- 7. 在匯入畫面依以下順序操作:General > existing project into workspace

⊜ Import			×
Select			5
Select an import source:			
<ul> <li>Archive File</li> <li>Existing Projects into Workspace</li> <li>File System</li> <li>Preferences</li> <li>C/C++</li> <li>Git</li> <li>Install</li> <li>Oomph</li> <li>Remote Systems</li> <li>RPM</li> <li>Run/Debug</li> <li>Tasks</li> <li>Tasks</li> <li>Tracing</li> </ul>			<
? < Back Next > Fin	ish	Cance	

8. 選擇預計開啟的根目錄,範例程式會在中間的視窗出現並點選"Finish"

9. 在成功匯入專案後,可以使用左上方工具列的槌子按鈕選擇要進行bulid的專案



10.選擇完畢後,右鍵點選或上方工具列點選建置(build)專案。專案可以允許建置 Debug模式與Release模式,建議使用Debug模式,在此選取Debug Configurations。

☆・	• 🜔 • 🤮 • 🂁 • 🙋 🥔
C	1 M480 ex_csp_debug
C	2 M480 ex_home_debug
C	3 M480 ex_rta_debug
C	4 M480 ex_pv_debug
C	5 M480 ex_pt_debug
C	6 M480 ex_pp_debug
C	7 NuDrive4 Debug
С	8 M480.elf
C	9 NuDrive4 Debug (3)
C	NuDrive4.elf
	Debug As >
	Debug Configurations
	Organize Favorites

11. 當專案建置完成後, 設定Debug配置與Run配置。在GBD Nuvoton Nu-Link Debugging點選左上圖標名稱為"New launch configuration", 來建立新的 configuration接下來應會自動生成剛剛建置完成的專案名稱。

ī.

Debug Configurations				×
Create, manage, and run configure	utions			Ť
<ul> <li>C/C++ Application</li> <li>M480.elf</li> <li>NuDrive_IO Debug</li> <li>NuDrive_IO Debug</li> <li>NuDrive4 Debug</li> <li>NuDrive4.elf</li> <li>C/C++ Attach to Application</li> <li>C/C++ Attach to Application</li> <li>C/C++ Remote Application</li> <li>C/C++ Remote Application</li> <li>GDB Hardware Debugging</li> <li>GDB Nuvoton Nu-Link Debugging</li> <li>GM480 ex_csp_debug</li> <li>M480 ex_pl_debug</li> <li>M480 ex_pl_debug</li> <li>M480 ex_rt_debug</li> <li>M480 ex_rt_debug</li> <li>NuDrive4 Debug (3)</li> <li>Launch Group</li> </ul>	Name: M480 ex_csp_debug Main ★ Debugger → Startup Project: M480 C/C++ Application: ex_csp_debug/M480.elf Build (if required) before launching Build configuration: Select Automatically ○ Enable auto build ● Use workspace settings	Source Common	riables Search Project)	Browse Browse
< > Filter matched 19 of 19 items			Re <u>v</u> ert	Apply
?			Debug	Close

### 12.建立新的New launch configuration後, 點選右方Search Project, 確認選取到相 對應名稱的檔案後點擊OK, 都完成後再點擊Debug, 系統會自動將程式寫入晶 片中

Debug Configurations		Program Selection —	×
Create, manage, and run configurat	ions <u>N</u> ame: M480	Choose a program to run:          Binaries:         M480.elf	
type filter text © C/C++ Application © M480.elf © New_configuration © NuDriveJO Debug © NuDrive4 Debug © NuDrive4.elf © C/C++ Attach to Application © C/C++ Remote Application © GDB Hardware Debugging © GDB Nurdvon Nu-Link Debugging © M480 ex_csp_debug © M480 ex_pt_debug © NuDrive4 Debug (3) ▶ Launch Group	Main * Project: M480 C/C++ Applix ex_csp_debu Build (if req Build config O Enable a © Use work	Qualifier: armle - /M480/ex_csp_debug/M480.elf armle - /M480/ex_pp_debug/M480.elf armle - /M480/ex_pp_debug/M480.elf armle - /M480/ex_pt_debug/M480.elf Cancel	Browse         Variables         Search Project         Browse         variables         variables
< >> Filter matched 19 of 19 items			Re <u>v</u> ert Apply
?			<u>D</u> ebug Close

### 修改範例程式為實際應用導引

正確開啟範例程式後可依以下步驟順序進行改寫 使用Drive的應用以STFDrive或NuDrive為範本進行修改,中間會有其他參考別範例中 的做法

### PdoDefine.h設定

點選並進入PdoDefine.h中後,在#define中輸入各類型從站(如Drive或IO或其他)數量與單站軸數。目前馬達控制器發展已有單控制器可多軸控制,所以若是一個控制器可以控制兩馬達進行運動,這樣的情況就會是一個從站與兩個軸數,下圖TEST\_DRV\_CNT代表DRV類型的從站數量,TEST\_IO\_CNT代表IO類型的從站數量。

```
#define TEST_DRV_CNT 1
#define TEST_IO_CNT 0
```

 完成設定後拉至下方繼續設定各類型從站的資料結構(PDO structure)。資料結構分為RxPDO(主站傳給從站的object)與TxPDO(從站傳給主站的object)兩種, 不同類型的從站如果資料結構不相同,應該分開設定。

下圖設定了RxPDO共有兩個Object,分別為16bit的Control Word與32bit的 Target。TxPDO也有兩個Object,分別為16bit的Status Word與32bit的Actual。 若為TEST\_HSP\_DEV型式的子站,則RxPDO共有三個Object,分別為16bit的 Control Word、32bit的Target及32bit的Out。TxPDO也有三個Object,分別為 16bit的Status Word、32bit的Actual及32bit的In。

```
typedef struct __attribute_((__packed__)) _axis_rxpdo_st_def_t
   uint16_t
               u16CtlWord;
                              // 0x6x40 - Control word
   int32 t
               n32Target;
                              // 0x6x7A - Target position
#ifdef TEST_HSP_DEV
   int32_t
              n320ut:
                              // Øx6xFE
#endif
AXIS_RXPDO_ST_DEF_T;
typedef struct __attribute __((__packed __)) __axis_txpdo_st_def_t
   uint16 t
              u16StaWord;
                               // 0x6x41 - Status word
                              // 0x6x64 - Actual position
   int32 t
              n32Actual;
#ifdef TEST_HSP_DEV
   int32_t
              n32In;
                               // @x6xED
#endif
}AXIS_TXPDO_ST_DEF_T;
```

#### STM32系列中main\_ini.c或Nuvoton系列中main.c設定

 首先要先設定SPI通訊時間,STM32系列中的SPI通訊時間必須另外安裝 STM32CubeMX來設定,Nuvoton系則以main.c中的#define TEST\_SPI\_FREQ 進行設定或是尋找UserSys\_Init()並填入SPI傳輸值。同時輸入欲測試的週期時 間,SPI相關設定可以參閱上位控制晶片(如MCU)的說明。

#define	TEST_SPI_FREQ	24000000
#define	DC_ACTIVE_CODE	0x300
#define	BASE CYCTIME	1000000

UserSys\_Init(TEST\_SPI\_FREQ);

 範例中關於TEST\_CYCTIME\_NS是定義週期時間,透過MULTI及DIVID的相關 define可以計算最後的週期時間。另外RPM與PPR相關的設定、速度相關設定是 在OP階段馬達運轉的運動參數。

```
* TEST_CYCTIME_DIVID
* 1 : for TEST_CYCTIME_MULTI
 * 2 : 0.5ms
 * 4 : 0.25ms
* 8 : 0.125ms
 * TEST_CYCTIME_MULTI
* 1 : for TEST CYCTIME DIVID
 * 2 : 2ms
 * 4 : 4ms
 * */
#define TEST_CYCTIME_DIVID
                                1
#define TEST_CYCTIME_MULTI
                                 1
#define TEST_CYCTIME_NS
#define ONE_SEC_CYC_CNT
                                  ((BASE_CYCTIME*TEST_CYCTIME_MULTI)/TEST_CYCTIME_DIVID)
                                 (1000000000/TEST CYCTIME NS)
```

3. 設定完成後下方為FIFO設置, FIFO為一暫存空間並依照順序與週期時間將命令 由主站送至從站, 若週期時間較短(如小於1ms)時, 如果每次SPI或USB交換只送 一次命令, 很有可能會造成RxFIFO為空, 當RxFIFO為空時, 會保持原值送出, 在CSP模式下, 很可能造成速度不連續, 導致馬達運轉會有震動的情況, 此時就 應加大每週期時間所傳輸的命令數量, 這邊就會修改

TEST\_PDO\_TO\_FIFO\_ONCE的數值,2為每周期間所傳輸的命令數量為兩筆。 實際上可觀察RxFIFO數量可否累積來決定TEST\_PDO\_TO\_FIFO\_ONCE是否 過小。

```
#define TEST_PDO_TO_FIFO_ONCE 2
#define TEST_RXFIFO_CNT 40
#define TEST_TXFIFO_CNT TEST_PDO_TO_FIFO_ONCE
```

 設定完成後跳至主程式int main\_ini()或int main()中開始進行程式的編寫與說明, 一開始會先將記憶體的位置清空以確保接下來運作正常

```
uint64 t u64Data:
int nStartFIFO = 0;
int i = 0, DrvIdx = 0, AxisIdx = 0, YasIdx = 0, nCnt1Sec=0, nRunTimeCnt=0;
int nVel = 0, nSumVel = 0, n32D0=0x5555;
int nret = 0, nServoState = 1;
int nLogStart = 0;
int n32CurPos[TEST AXIS CNT];
uint16_t u16LastSW[TEST_AXIS_CNT], u16StaWord[TEST_DRV_CNT][N_AXIS_IN_ONE_DRV];
uint32_t u32CycTimeCnt = 0, u32RunTimeCnt = 0, u32LogFifoCnt=0;
uint8 t u8LEDAxis = 0;
uint8 t u8LEDBit = 0;
uint8_t u8Version = 0, u8FifoCnt = 0, u8FifoCntMax = TEST_RXFIF0_CNT;
uint8_t u8State = 0, u8WkcErrCnt = 0, u8CrcErrCnt = 0, u8IsSlvAlive = 0;
uint8 t u8LastState = 0, u8LastWkcErrCnt = 0, u8LastCrcErrCnt = 0;
uint16_t u16RxPDOSize = 0, u16TxPDOSize = 0, u16SpiSize = 0;
int nDriveRxPDOSize = 0, nDriveTxPDOSize = 0;
int8 t SlaveCnt = 0;
RXPDO_ST_DEF_T *pAllDevRx;
TXPDO_ST_DEF_T *pAllDevTx;
AXIS_RXPDO_ST_DEF_T *pRxPDOAxis;
AXIS_TXPDO_ST_DEF_T *pTxPDOAxis;
memset(RxPDOData, 0, sizeof(RxPDOData));
memset(TxPDOData, 0, sizeof(TxPDOData));
memset(nPos, 0, sizeof(nPos));
```

5. EtherCAT主站的第一個命令為ECM\_InitLibrary(&u16SpiSize)。當中會執行SPI 資料大小的設定與確認,同時會確認主站IC的韌體版本等功能。u16SpiSize為欲 設定新SPI資料大小,在C語言中若ECM\_InitLibrary(0)代表不更改SPI Data Size。

u8Version = ECM\_InitLibrary(&u16SpiSize);

 當確認項目皆正常執行後,開始進行EtherCAT初始化 ECM\_EcatInit(DCActCode, CycleTime),並進入Init state ECM\_EcatInit(DCActCode, CycleTime)中DCActCode值0為關閉DC sync功能、 0x300啟動Sync0、0x700同時開啟Sync0與Sync1,CycleTime為週期時間,單位 為ns。
 ※此函數會針對 "所有" Slave 下達相同的DCActCode,若對於不同Slave要設定 不同的DCActCode,則需使用ECM CMD ECAT DCSYNC(命令代碼50)

ECM\_EcatInit(DC\_ACTIVE\_CODE, (BASE\_CYCTIME\*TEST\_CYCTIME\_MULTI) / TEST\_CYCTIME\_DIVID);

7. 確認主站與從站連接後透過ECM\_StateCheck(Slave, ExpectedState, Timeout) 進入Pro-OP開始設定PDO配置,如果從站已有預設配置則只需要給予相對應的 Map Index即可,如為特殊配置則需配置PDO內容,SetPdoConfig提供最多3組 PDO各8個Object的配置大小,多數範例配置內容在main\_ini.c或main.c中,目前 唯Drive範例在另外在Utility.c中。如果使用ConfigDrive進行配置,內部已有 Reconfig與ShowPDOConfig的動作,所以無需另外加入Reconfig動作,其他範 例則需在完成Configure PDO後加入Reconfigure的動作與ShowPDOConfig。 ECM\_StateCheck(Slave, ExpectedState, Timeout)中Slave如為0xFF則表示針 對所有從站同時命令,ExpectedState則是欲前往之狀態(Pro-OP、Safe-OP與 OP狀態),Timeout為逾時等待時間,如出現無法狀態切換時可能等待時間過短。 ConfigDrive(1, 0, (TEST\_DRV\_CNT - 1), 1, N\_AXIS\_IN\_ONE\_DRV, 0x1602, 0x1A02);

#### 或

```
RxPD0Config[i].SmaIdx = RxPD0_ASSIGN_IDX;
RxPD0Config[i].PD0Cnt = 1;
RxPD0Config[i].MapIdx[0] = RxPD0_MAP_IDX;
RxPD0Config[i].0bjsCnt[0] = 2;
SetPdoConfIb1(&RxPD0Config[i].0,0,0,0x6040,0,16); //control word // 16 bits = 2 bytes for TEST_RXPD0_SIZE
// the 1st parameter is PD0_CONFIG_HEAD
// the 2nd parameter is 0 due to RxPD0Config.PD0Cnt = 1
// the 3nd parameter is 0 due to RxPD0Config.PD0Cnt = 1
// the 3nd parameter is a control word index 0x6040
// the 5th parameter is a control word index 0x6040
// the 6th parameter is a sub-index for 0x6040
// the 6th parameter is the bit size for 4th parameter
SetPdoConfig[i].SmaIdx = TxPD0_ASSIGN_IDX;
TxPD0Config[i].SmaIdx = TxPD0_ASSIGN_IDX;
TxPD0Config[i].MapIdx[0] = TxPD0_MAP_IDX;
TxPD0Config[i].MapIdx[0] = TxPD0_MAP_IDX;
TxPD0Config[i].ObjsCnt[0] = 2;
SetPdoConfTb1(&TxPD0Config[i], 0, 0, 0x6041, 0, 16); //status word // 16 bits = 2 bytes for TEST_TXPD0_SIZE
SetPdoConfTb1(&TxPD0Config[i], 0, 1, 0x6064, 0, 32); //actual position // 32 bits = 4 bytes for TEST_TXPD0_SIZE
```

ECM\_EcatReconfig();

 完成配置後進行ECM\_CheckMEMSpace(TEST\_PDO\_FIFO\_ONCE), 此步驟 為確認各部分相關記憶體的大小, 參數需輸入每次SPI/USB傳輸包含幾筆週期 命令。

```
ECM_CheckMEMSpace(TEST_PD0_T0_FIF0_ONCE);
```

檢查確認沒有問題後利用ECM\_StateCheck(Slave, ExpectedState, Timeout)進入Safe-OP狀態

ECM\_StateCheck(0xFF, EC\_STATE\_SAFE\_OP, 1000);

- 10.可進入Safe-OP狀態後繼續執行ECM\_StateCheck(Slave, ExpectedState, Timeout)進入OP狀態
- 11. 進入OP狀態後執行ECM\_CheckDCStable()確認DC狀態穩定 ECM\_CheckDCStable();
- 12.當確認完成後,以ECM\_Drv402SM\_StateSet(Axes, ServoOn/OffState)或 ECM\_Drv402SM\_Enable(Axes, Slaves)啟用402 state machine至servo on階 段,此階段為自動切換402 state machine的方式,如果從站無法使用以上兩種切 換方式時,可以參考STF4HSP或NuHSP中的手動切換模式進行單步切換至 Servo On狀態

```
ECM_Drv402SM_StateSet((DrvIdx*N_AXIS_IN_ONE_DRV) + AxisIdx, SERVO_ON_STATE);
```

```
PdoExchangeAndGet402State(DrvIdx, AxisIdx, &u8State);
```

或

```
ECM_Drv402SM_Enable(0, 0);
```

13. Servo On完成後若為CSP模式, 必須將編碼器(實際位置)與命令的目標位置對 齊, 可利用AlignmentPosition(), 並使用ECM\_InitFIFO()初始化FIFO同時使用 ECM\_ClearFIFO(Direction)將FIFO內部清空後以ECM\_EnableFIFO(Enable)來 啟用FIFO功能。

ECM\_ClearFIFO(0), 0為Tx與Rx

ECM\_EnableFIFO(1), 1為Enable、0為Disable

```
ECM_InitFIF0();
ECM_ClearFIF0(0); // 0 for TX and RX both
PRINTF("ClearFIF0\r\n");
ECM_EnableFIF0(1); // Enable FIF0
PRINTF("EnableFIF0\r\n");
```

14.完成上方步驟後已完成相關相關設定,接下來的資料交換皆會在while迴圈中不 斷重複運行,以判斷FIFO的數量為基準,如果FIFO已經快要塞滿時,則選擇暫停 塞入命令,此外持續將命令塞入FIFO,速度與位置相關設定在Utility.c中可依需 求進行編寫。

### 常見錯誤排除

在執行相關設定時如果跳出Error相關問題,請依以下進行排除

階段	顯示	解決方法
ECM_Init_Library	wait ASYNC done timeout	請確認主站與從站連接是 否正常, 主站與從站網口 燈號是否亮起
ECM_Init_Library	u8ErrorStatus 0x40	初次設定會跳出的警告, Novoton系列按Enter繼續
通期	wait ASYNC done timeout	等待時間不足或從站錯誤
通期	CRC Error	SPI傳輸訊號錯誤, 請確認 SPI主站與EtherCAT主站

|--|

### Drive 402自動切換與手動切換

Drive 402 指的是驅動器上的電源控制(servo on、激磁或使能),使用者可以透過 Control Word (0x6040)控制,並透過Status Word (0x6041)來確認狀態。另一個方式是 透過ECM\_CMD\_402\_CONFIG\_SET來指定Control Word及Status Word在該子站的位 置的偏移量(offset),隨後再透過ECM\_CMD\_402\_STATE\_SET來設定希望到達的電源 狀態。

 在STF4Drive範例中用ECM\_Drv402SM\_StateSet()與 PdoExchangeAndGet402State()來進行Servo ON

•	其他範例中以ECM_Drv402SM_Enable()的方式進行Servo On			
	ECM Drv402SM Enable中的參數分別為軸編號(Tblldx)與從站編號(Slvldx)			
	ECM Drv402SM Enable(0, 0);			
	ECM Drv402SM Enable(1, 1);			
	如果為一對二控制器且有兩站時命令就會變成			
	ECM Drv402SM Enable(0,0); //第0站第0個軸, 整體的第0軸			
	ECM_Drv402SM_Enable(1, 0); //第0站第1個軸, 整體的第1軸			
	 ECM_Drv402SM_Enable(2, 1);			
	ECM_Drv402SM_Enable(3, 1);			
•	手動切換的方式請參考下面程式碼, 手動切換是透過Control Word來切換			
	<pre>for(DrvIdx=0;DrvIdx<test_slave_cnt;drvidx++){ for(axisidx="0;AxisIdx&lt;N_DRV_IN_ONE_SLV;AxisIdx++){&lt;/pre"></test_slave_cnt;drvidx++){></pre>			
	j = 0; while(1){ 			
	<pre>if(net&gt;0){     ul6logStatus[(DrvIdx*N DRV IN ONE SLV)+AxisIdx] = pTxPD0Data-&gt;DRIVE GROUP 0[DrvIdx].HSP[AxisIdx].ul6StaWord;</pre>			
	<pre>PDOstate[(DrvIdx*N_DRV_IN_ONE_SLV)+AxisIdx] = pTxPDOData-&gt;DRIVE_GROUP_0[DrvIdx].HSP[AxisIdx].u16StaWord &amp; CIA402_SW_STATE_MASK; PRINTF("DrvIdx = %d, AxisIdx = %d, state = 0x%x\r\n", DrvIdx, AxisIdx, PDOstate[(DrvIdx*N_DRV_IN_ONE_SLV)+AxisIdx]);</pre>			
	if(PDOstate[(DrvIdx*N_DRV_IN_ONE_SLV)+AxisIdx]==CIA402_SW_NOTREADYTOSWITCHON){ UserDelay(1000);			
	<pre>} if(PDOstate[(DrvIdx*N_DRV_IN_ONE_SLV)+AxisIdx]==CIA402_SW_SWITCHEDONDISABLED){     pRxPDOData-&gt;DRIVE_GROUP_0[DrvIdx].HSP[AxisIdx].u16CtrlWord = 0x6; //Control word: Shutdown = 0x6</pre>			
	<pre>} if(PDOstate[(DrvIdx*N_DRV_IN_ONE_SLV)+AxisIdx]==CIA402_SW_READYTOSWITCHON){     pRxPDOData-&gt;DRIVE_GROUP_0[DrvIdx].HSP[AxisIdx].u16CtrlWord = 0x7; //Control word: Switch on = 0x7 }</pre>			
	<pre>if(PDOstate[(DrvIdx*N_DRV_IN_ONE_SLV)+AxisIdx]==CIA402_SW_SWITCHEDON){     pRxPDOData-&gt;DRIVE_GROUP_@[DrvIdx].HSP[AxisIdx].u16CtrlWord = 0xF; //Control word: Enable operation = 0xF</pre>			
	<pre>} f(PDOstate[(DrvIdx*N_DRV_IN_ONE_SLV)+AxisIdx]==CIA402_SW_OPERATIONENABLED){     j++;     if(j==3){         break;         break;     } }</pre>			
	} } if(PDOstate[(DrvIdx*N_DRV_IN_ONE_SLV)+AxisIdx]==CIA402_SW_QUICKSTOPACTIVE){ pRxPDOData->DRIVE_GROUP_0[DrvIdx].HSP[AxisIdx].u16CtrlWord = 0x0; //Control word: Disable voltage = 0x0			
	<pre>} if(PDOstate[(DrvIdx*N_DRV_IN_ONE_SLV)+AxisIdx]==CIA402_SW_FAULTREACTIONACTIVE){     UserDelay(1000); }</pre>			
	<pre>if(PDOstate[(DrvIdx*N_DRV_IN_ONE_SLV)+AxisIdx]==CIA402_SW_FAULT){     pRxPDOData-&gt;DRIVE_GROUP_0[DrvIdx].HSP[AxisIdx].ul6CtrlNord = 0x0; //Control word: Fault reset = 0x0-&gt;0x80     nret = ECM_EcatPdoFifoDataExchange(PD0_FIF0_DEFAULT_CNT, RxData, TxData, ul6PDOSize, &amp;u8FifoCnt, &amp;u8WkcErrCnt, &amp;u8CrcErrCnt);     UserDelay(1000);     pRxPDOData-&gt;DRIVE_GROUP_0[DrvIdx].HSP[AxisIdx].ul6CtrlNord = 0x80;     nret = ECM_EcatPdoFifoDataExchange(PD0_FIF0_DEFAULT_CNT, RxData, TxData, ul6PDOSize, &amp;u8FifoCnt, &amp;u8WkcErrCnt, &amp;u8CrcErrCnt);     UserDelay(1000);     UserDelay(1000); </pre>			
	} '			

## 維護紀錄

版本	日期	說明
01	2021.02.09	綜合ECM-XF-MCU User Guide, Sample Code Explanation, and STM32 Setup Instruction(NUCLEO-F401RE)
	2021.02.17	修改新唐環境建立導引說明
02	2021.05.06	更新STF4Drive與NuDrive
03	2021.06.28	更新範例修改指引
04	2021.06.29	修正修改指引細項
05	2021.08.17	新增STM32 H7腳位設定
06	2022.12.25	<b>修正範例</b> 說明