CANopen FD

Starter kit: Hardware and software

Emotas' CANopen FD starter kit provides a CAN FD micro-controller board, an extension board with CAN FD transceiver, and a CAN FD USB interface to start with CANopen FD immediately.

ANopen FD as specified in the CAN in Automation (CiA) specification 1301 uses the new features of CAN FD such as a higher data bit-rate and longer frames up to 64 bytes. Most principles of classic CANopen are reused, but some are extended or modified. The most notable improvement is the new USDO service that provides arbitrary access to CANopen FD objects. Compared to the SDO service of classic CANopen it is not only faster but also provides a broad cast mechanism. The other major improvement is the extended PDO length supporting 64 bytes instead of only 8 bytes in one

the extended PDO length supporting *Figure 1: CANopen* 64 bytes instead of only 8 bytes in one PDO. Last but not least the Emergency messages have been extended as well to provide more detailed information about errors detected by the device. Unfortunately, the number of CANopen FD devices available on the market is currently limited. This is also a problem for developers who would like to evaluate the new features of the improved protocol. In order to provide a cost-effective solution to get started with CANopen FD, Emotas embedded communication, a German company well-known for its CAN and CANopen FD expertise, offers a CANopen FD starter kit.

The starter kit unboxed

The CANopen FD starter kit is based on an STM32 Nucleo-64 board with a powerful STM32G4 microcontroller that internally uses an CAN FD controller that supports both Classical CAN and CAN FD. To connect the CAN controller to a CAN FD network a CAN FD transceiver is required. Emotas has developed a specific expansion board with a CAN FD transceiver and DSUB-9 connector to connect the CAN FD network conveniently. The CAN FD transceiver TJA1051 supports up 5 Mbit/s in data phase. In addition to CAN Y-cables to connect also additional devices and two termination resistors the starter kit comes with an Ixxat USB-to-CAN FD interface that supports these bit-rates as well. In addition to the hardware components, the CANopen FD starter kit also includes software: A CANopen FD slave stack to run on the STM32G4 and a CANopen FD tool with CANopen FD master and CAN FD analyzer capabilities.

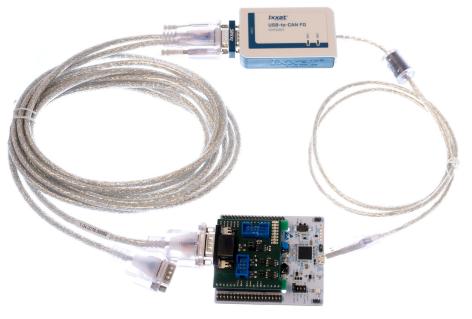


Figure 1: CANopen FD starter kit hardware (Source: Emotas)

CANopen FD slave stack

<u>As reported</u> on the CAN Newsletter Online, the Emotas CANopen FD stack was already published in 2017. Based on already three years of CANopen FD experience and recognizing the demand for smaller micro-controllers,

Edit Sou	irce Refactor Navigate Search Project Run Window Help				
- 80	🐘 🛞 + 🐔 + 🚋 i 🔌 i ຝ i 📸 + 😂 + 🗗 + 🎯 + i 🎋 + 🔘 + 🗛 + 🎒 🛷 + 🇾 🐼 🗐 🖷				
- 🐻 -			T 2	6	x 7
0					
🚺 maii	ne 🛛				
498				^	
	static void simuApp(void* pData)				
500	1 UNSIGNED8 1;				
	INTEGER16 valAI; /* analog input objects value */				
503	UNSIGNED8 highestIndexAI; /* analog input objects count */				
504					
505	/* not used */				
506 507	(void)pData;				
5080	/* simulate an application by increasing digital input and analog input values				
509	* which are sent out via PDO */				
510	<pre>if (coOdGetObj_u8(I_READ_ANALOGUE_INPUT_16_BIT, 0, &highestIndexAI) == RET_OK) {</pre>				
511	<pre>for (i = lu; i <= highestIndexAI; i++) {</pre>				
512 513	<pre>if (coOdGetObj_i16(I_READ_ANALOGUE_INPUT_16_BIT, i, &valAI) == RET_OK) { valAI += i;</pre>				
514	if (coOdPutObj i16(I READ ANALOGUE INPUT 16 BIT, i, valAI) != RET OK)	{			
515	<pre>printf("error increasing value!\n");</pre>	•			
516	}				
517	}				
518 519	}				
520	1				
521	/* read GPIO inputs and save state into the object dictionary*/				
522	<pre>if (HAL_GPIO_ReadPin(DIGI_IN_1_PORT, DIGI_IN_1_PIN) == GPIO_PIN_RESET) {</pre>				
523 524	(void)coOdPutObj_u8(0x6000, 1u, 1u);				
524	<pre>} else { (void)coOdPutObj u8(0x6000, 1u, 0u);</pre>				
526	}				
527	if (HAL_GPIO_ReadPin(DIGI_IN_2_PORT, DIGI_IN_2_PIN) == GPIO_PIN_RESET) {			- 64	
528	<pre>(void)coOdPutObj_u8(0x6000, 2u, 1u);</pre>				
529 530	} else {				
530	(void)coOdPutObj_u8(0x6000, 2u, 0u);				
532	<pre>if (HAL_GPIO_ReadPin(DIGI_IN_3_PORT, DIGI_IN_3_PIN) == GPIO_PIN_RESET) {</pre>				
533	(void)coOdPutObj_u8(0x6000, 3u, 1u);				
534	} else {				
535 536	(void)coOdPutObj_u8(0x6000, 3u, 0u);				
537	<pre>if (HAL_GPIO_ReadPin(DIGI_IN_4_PORT, DIGI_IN_4_PIN) == GPIO_PIN_RESET) {</pre>				
538	(void)coOdPutObj_u8(0x6000, 4u, 1u);				
539	} else {				
540 541	(void)coOdPutObj_u8(0x6000, 4u, 0u);				
541	}				
543	,				
544				~	
	¢			>	

Figure 2: Screenshot application code in STM32CubeIDE (Source: Emotas)

aral Settings			
ce Description			
ct Dictionary			
e here to filter			
	Attributes Description	Meta data	
ect Dictionary • 0x1005 - COB ID SYNC		Value	*
0x1008 - Manufacturer Device Name	Name	Producer Heartbeat Time	
0x1014 - COB ID EMCY	Data Type	UNSIGNED16	
 0x1015 - Inhibit Time Emergency 0x1016 - Consumer Heartbeat Time 			
 0x1016 - Consumer Heartbeat Time 0x1017 - Producer Heartbeat Time 	Access Type	Read Write	· ·
0 - Producer Heartbeat Time	Lower Limit	0	
 0x1018 - Identity Object 	Upper Limit	65535	
0 - Number of entries 1 - Vendor Id	Value	1000	
2 - Product Code	PDO Mapping	Forbidden	Ψ
3 - Revision number	Defaultvalue	yes	~
4 - Serial number	Limit Check	no	*
 0x1030 - Version information 0x1031 - Active error history 	Befuse read on Scan	ne	*
0 - Highest sub-index supported	. Refuse write on Downloa	d no	*
1 - Error history status 2 - Error history command	Defaultvalue in EDS	as defaultvalue	
3 - Error history command 3 - Error history command input		Value	
4 - Error history domain	Implementation Type	Managed Variable	*
5 - Error history processing • 0x1032 - Active error list	Variable Name	Manageu Variable	
 0x1032 - Active error list 0 - Highest sub-index supported 	Size in Bytes		
1 - Error event 1		no	~
2 - Error event 2	Non-volatile storage	no	
0x1400 - Receive PDO Communication Parameter 0x1401 - Receive PDO Communication Parameter	interventatile storage	110	
 0x1401 - Receive PDO Communication Parameter 0x1600 - Receive PDO Mapping Parameter 	Reset Changes		Reset Defaults
 0x1601 - Receive PDO Mapping Parameter 	-		

Figure 3: CANopen Devicedesigner (Source: Emotas)

the CANopen FD slave stack has been ported to the STM32G4 recently. An evaluation version of Emotas' CANopen FD slave stack runs inside the STM32G4. It is a binary library of the companies source code stack and to limit the use case to evaluation purposes, the run-time is limited to one hour after reset.

Nevertheless, it comes with the following CANopen FD features:

- NMT Slave
- USDO server with simultaneous connections (expedited unicast and broadcast, segmented unicast and bulk transfer)
- multiple PDO producers and PDO consumers
- Sync consumer
- Heartbeat producer
- 1 Heartbeat consumer
- Emergency producer

An example application is included as STM32CubeIDE project and it simulates a digital/ analog I/O device with real and simulated values mapped into longer PDO. Several data objects exceeding the length of 54 bytes are included to show

the USDO segmented transfer or USDO bulk transfer as well. By default the bit-rate pair of this example is set to 500 kbit/s nominal bit-rate and 2 Mbit/s data bit-rate. But both the bit-rate pair and the node-ID can be configured from the application. So if one has multiple boards, a network with multiple nodes may be set up. Based on this example, ▷

Exhibition cancelled? We have the solution!

Join CiA's event stage

advertise your events and tradeshows; <
invite the CAN community worldwide; <
get connected to international CAN <
experts.

For more details please contact CiA office at publications@can-cia.org www.can-cia.org



CAN in Automation

Object Browser	😔 Network 🔍	PDO Configuration	PDO TX & RX			(- L	×
terjown save	*	EDS: C:/Daten/SVN/01	004_trunk/examplefd_sl/	alave 1/canfd_slave	1.eds				1
Objects		*	 Index 0x1008 - Sul 	aindex 0x0			-	-1	1
		*			Attribute				T'
Communication Segment Ox1000 - Device Type Ox1001 - Error Register			Index	0x1008					
			SubIndex	0					
			Name	Manufacturer				1 1	
✓ 0x1008 -	Datatype	VISIBLE_STRING	3			+ +			
0 - Ma	Access	const							
	OB ID EMCY		PDO Mapping Default Value	no					
	 > 0x1015 - Inhibit Time Emergency > 0x1016 - Consumer Heartbeat Time > 0x1017 - Producer Heartbeat Time 			emotas Slave 1					
				emotas Slave 1					
	dentity Object							11	
	mergency Consum	ner	Cyclic Read		Read	Write			
	rror behaviour				Reau	white			-
0111000	ersi 💷 CANopen Device							-	
0 - Hig	cott .	CANopen Interpretation							
1 - Ver	CHAIT ICH	open Interpretation C/	AN Object View CANo	pen Object View	CiA447 Interpretation	31939 Interpretation	ISO-TP Interpretati	ion	
		× 2	HEX ¥				Undet	e Model	Clear view
2 - Ver	O						opuar	e Model	Clear view
2 - Ver * 0x1032 -	Ctivi Autoscrol Circular	Toggle filter Refresh							٩,
2 - Ver * 0x1032 - 0 - Hig	ctivi Autoscroli Circular est : CAN Rx	Toggie filter Refresh					Search		
2 - Ver	ctivi Autosciol Circular est : CAN Rx nde: Time Stamp	CAN-ID Type	Node-Id Data					Interpretation	
2 - Ver * 0x1032 - 0 - Hig 1 - Sub 2 - Sub	Autosciol Circular est: CAN Rx nde: nde: 311.795000	CAN-ID Type 511/0x1ff PDO	Node-Id Data - 04.00	00 00 00 00 02 00 (04 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 00			
2 - Ver	Ctiv Characteria Q Autoscoli Circular est : CAN Rx Inde: 311.755000 312.214000 312.279000	CAN-ID Type 511/0x1ff PDO 1919/0x77f Heartbeat 511/0x1ff PDO	Node-Id Data - 04.00 127 - slave1 05 - 04.00		04 00 00 00 00 00 00 00 00 00		0 00 00 00 00 00 00 00	Operational	
2 - Ver	Ctiv Autocoli Circular est : CAN Rx Time Stamp 311/795000 nde: > 312.214000 312.795000 > 313.276000 313.795000 313.795000	CAN-ID Type 511/0x1ff PDO 1919/0x77f Heartbeat 511/0x1ff PDO 1919/0x77f Heartbeat 511/0x1ff PDO	Node-Id Data - 0400 127 - slave1 05 - 0400 127 - slave1 05 - 0400	00 00 00 00 02 00		00 00 00 00 00 00 00 00 00 00	0 00 00 00 00 00 00 00	Operational Operational	
2 - Ver	Ctiv Actosorial Circlar cest : CAN Rx Can Rx inde: 312.95000 313.2795000 at3.2795000 313.2795000 313.2795000	CAN-ID Type 511/0x1ff PDO 1919/0x77f Heartbeat 511/0x1ff PDO 1919/0x77f Heartbeat 511/0x1ff PDO 1601/0x641 USDO	Node-Id Data - 04.00 127 - slave1 05 - 04.00 127 - slave1 05 - 04.00 65 77 (110)	00 00 00 00 00 02 00 00 00 00 00 02 00 07 00 08 10	04 00 00 00 00 00 00 00 00 00 04 00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 00 00 00 00	0 00 00 00 00 00 00 00	Operational Operational r req 127 65 (0	
2 - Ver * 0x1032 - 0 - Hig 1 - Sub 2 - Sub 3 - Sub	Image O Autocol Circlar est : CAN Rx index 311,795000 at32,795000 313,216000 a13,216000 313,325000 ied cor > 313,232000 > 313,2420000 313,342000	CAN-ID Type 511/0x1ff PDO 1919/0x77f Heartbeat 511/0x1ff PDO 1919/0x77f Heartbeat 1511/0x1ff PDO 1601/0x641 USDO 1535/0x5ff USDO 1919/0x77f Heartbeat	Node-Id Data - 04 00 127 - slave1 05 - 04 00 127 - slave1 05 - 04 00 65 77 111 127 - slave1 05 127 - slave1 05	00 00 00 00 00 02 00 00 00 00 00 02 00 07 00 08 10 07 00 08 10 09 0e	04 00 00 00 00 00 00 00 00 00 04 00 00 00 00 00 00 00 00 65 6d 6f 74 61 73 20 53 6c	00 00 00 00 00 00 00 00 00 00 00 00 00 0	0 00 00 00 00 00 00 00 00 00 00 00 00 0	Operational Operational r req 127 65 (0 r resp 65 127 (Operational	
2 - Ver * 0x1032 - 0 - Hig 1 - Sub 2 - Sub	Image: State	CAN-ID Type 511/0x1H PDO 1919/0x77H Heartbeat 511/0x1H PDO 1919/0x77H Heartbeat 511/0x1H PDO 1601/0x641 USDO 1635/0x5H USDO	Node-Id Data - 04 00 127 - slave1 05 - - 04 00 127 - slave1 05 - - 04 00 65 77 111 127 - slave1 41 31 127 - slave1 05 - - 04 00 61 - 0 - 00 05 - 04 00 127 - slave1 05 - - 04 00	00 00 00 00 02 00 00 00 00 00 02 00 07 00 08 10 07 00 08 10 09 0e 00 00 00 00 02 00	04 00 00 00 00 00 00 00 00 00 04 00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 00 00 00 00 0		Operational Operational r req 127 65 (0 r resp 65 127 (Operational Operational	

Figure 4: Screenshot CANopen Deviceexplorer (Source: Emotas)

the user can extend or modify the object dictionary of the CANopen FD device to meet his requirements. This can be realized by Emotas' CANopen Devicedesigner.

Tool based object dictionary design

The CANopen Devicedesigner allows the definition of CANopen FD object dictionaries based on pre-defined databases e.g. for all CiA 1301 objects. Additionally the user may add own manufacturer-specific and device-profile-specific objects and assign them to PDOs and configure other CANopen FD services. Based on these configurations the tool generates .c and .h files that include the object dictionary implementation and configure the stack. Additionally, the tool generates from the same source a documentation of the object dictionary in CSV, TXT, and HTML files and both a classic CANopen EDS file and a new CANopen FD XDD file can be imported into CANopen FD tools or CANopen FD masters so that these tools and masters are aware of the slaves' object dictionaries.

CANopen FD master tool with CAN FD analyzer

In order to control a CANopen FD slave device a CANopen FD master is required. Instead of a master device, a PC-based master tool is used to provide a high flexibility especially for development or evaluation. The CANopen Deviceexplorer is such a CANopen FD master tool and additionally includes a sophisticated CANopen FD analyzer that interprets the raw CAN FD messages according to the CANopen FD specification. By using the tool one can access the device' object dictionary by USDO, configure, send and receive PDO and send NMT commands and monitor the raw CAN FD messages and their CANopen FD interpretation simultaneously.

The delivery includes the version for Windows, but the versions for Linux and macOS are available on request. The provided evaluation version of the CANopen

Deviceexplorer can only be used with a fixed bit-rate pair of 500 kbit/s and 2 Mbit/s. Object dictionary access, PDO configuration, and NMT commands are limited to the node-IDs 1, 2, 32, and 64. In addition to that, all CANopen FD master features of the tool are enabled and the integrated CAN analyzer interprets all CAN FD messages from all nodes.

Conclusion

The CANopen FD starter kit provides all necessary components to start with a CANopen FD slave development or to evaluate the possibilities of the CANopen FD protocol. In order to spread the usage of CANopen FD, Emo-

tas Embedded Communications not just offers the starter kit for sale, but also provides all software components for download for free.

Author

Torsten Gedenk Emotas Embedded Communication ged@emotas.de www.emotas.de





Vision 3

Enhance human machine interaction

TTControl's rugged operator interface family Vision 3 was specially designed to meet the requirements of the harsh and more and more digitized off-highway environment.

Features like four simultaneous video streams, support of hardwareaccelerated 3D animations, user inputs through a multi-touch capable screen and acoustic feedback via an integrated loudspeaker ensure a perfect interaction of the operator with the vehicle for safe and efficient machine operation.

> more information www.ttcontrol.com/vision3 products@ttcontrol.com

<u>"</u>

ontrol