mbed-lib

1.0

Generated by Doxygen 1.7.5.1

Mon Nov 28 2011 12:54:03

# **Contents**

1	Mod	ule Inde	ex								1
	1.1	Module	es			 	 	 			1
2	File	Index									3
	2.1	File Lis	st			 	 	 			3
3	Mod	ule Doc	umentatio	n							5
	3.1	CAN b	audrates			 	 	 			5
		3.1.1	Detailed I	Description		 	 	 			5
	3.2	MBED	_CAN0 but	fers		 	 	 			6
		3.2.1	Detailed I	Description		 	 	 			6
	3.3	MBED	_CAN1 but	fers		 	 	 			7
		3.3.1	Detailed I	Description		 	 	 			7
	3.4	Initaliza	ation flags			 	 	 			8
		3.4.1	Detailed I	Description		 	 	 			8
		3.4.2	Define Do	ocumentation	n	 	 	 			8
			3.4.2.1	INIT_CANO	)	 	 	 			8
			3.4.2.2	INIT_CAN1		 	 	 			8
			3.4.2.3	INIT_GPI .		 	 	 			8
			3.4.2.4	INIT_GPO		 	 	 			8
			3.4.2.5	INIT_LED		 	 	 			9
			3.4.2.6	INIT_MUSE	3	 	 	 			9
			3.4.2.7	INIT_SERI	ALO	 	 	 			9
			3.4.2.8	INIT_SERI	AL1	 	 	 			9
			3.4.2.9	INIT_SERI	AL2	 	 	 			9
			2 / 2 10	INIT TIME	RΛ						۵

ii CONTENTS

		3.4.2.11	INIT_TIMER1 9
		3.4.2.12	INIT_TIMER2 9
		3.4.2.13	INIT_TIMER3 9
3.5	Status	flags	
	3.5.1	Detailed	Description
	3.5.2	Define De	ocumentation
		3.5.2.1	MBED_CAN0_INIT
		3.5.2.2	MBED_CAN1_INIT
		3.5.2.3	MBED_GPI_INIT
		3.5.2.4	MBED_GPO_INIT
		3.5.2.5	MBED_LED_INIT
		3.5.2.6	MBED_MUSB_INIT
		3.5.2.7	MBED_SERIALO_INIT
		3.5.2.8	MBED_SERIAL1_INIT
		3.5.2.9	MBED_SERIAL2_INIT
		3.5.2.10	MBED_SYS_INIT
		3.5.2.11	MBED_TIMER0_INIT
		3.5.2.12	MBED_TIMER1_INIT
		3.5.2.13	MBED_TIMER2_INIT
		3.5.2.14	MBED_TIMER3_INIT
3.6	Genera	al purpose	I/O pins
	3.6.1	Detailed	Description
	3.6.2	Define De	ocumentation
		3.6.2.1	MBED_GPIO10
		3.6.2.2	MBED_GPIO11
		3.6.2.3	MBED_GPIO12
		3.6.2.4	MBED_GPIO13 13
		3.6.2.5	MBED_GPIO14
		3.6.2.6	MBED_GPIO15
		3.6.2.7	MBED_GPIO16
		3.6.2.8	MBED_GPIO17
		3.6.2.9	MBED_GPIO18
		3.6.2.10	MBED_GPIO19
		3.6.2.11	MBED_GPIO20

CONTENTS iii

	3.6.2.12	MBED_GPIO21
	3.6.2.13	MBED_GPIO22
	3.6.2.14	MBED_GPIO23
	3.6.2.15	MBED_GPIO24
	3.6.2.16	MBED_GPIO25
	3.6.2.17	MBED_GPIO26
	3.6.2.18	MBED_GPIO27
	3.6.2.19	MBED_GPIO28
	3.6.2.20	MBED_GPIO29
	3.6.2.21	MBED_GPIO30
	3.6.2.22	MBED_GPIO5
	3.6.2.23	MBED_GPI06
	3.6.2.24	MBED_GPI07
	3.6.2.25	MBED_GPI08
	3.6.2.26	MBED_GPIO9
	3.6.2.27	MBED_GPIOX
Port co	onfiguration	1
3.7.1	Detailed	Description
3.7.2	Define Do	ocumentation
	0704	
	3.7.2.1	MBED_GPIO_P0
	3.7.2.1	MBED_GPIO_P0         10           MBED_GPIO_P1         10
Device	3.7.2.2 3.7.2.3	MBED_GPIO_P1
Device	3.7.2.2 3.7.2.3 IDs	MBED_GPIO_P1
	3.7.2.2 3.7.2.3 IDs Detailed	MBED_GPIO_P1
3.8.1	3.7.2.2 3.7.2.3 IDs Detailed	MBED_GPIO_P1       10         MBED_GPIO_P2       11
3.8.1	3.7.2.2 3.7.2.3 IDs Detailed Define Do	MBED_GPIO_P1       10         MBED_GPIO_P2       10
3.8.1	3.7.2.2 3.7.2.3 IDs Detailed Define Do 3.8.2.1	MBED_GPIO_P1       10         MBED_GPIO_P2       10
3.8.1	3.7.2.2 3.7.2.3 IDs Detailed Define Do 3.8.2.1 3.8.2.2	MBED_GPIO_P1       10         MBED_GPIO_P2       10
3.8.1	3.7.2.2 3.7.2.3 PIDS Detailed Define Do 3.8.2.1 3.8.2.2 3.8.2.3	MBED_GPIO_P1       10         MBED_GPIO_P2       10
3.8.1	3.7.2.2 3.7.2.3 IDs Detailed Define Do 3.8.2.1 3.8.2.2 3.8.2.3 3.8.2.4	MBED_GPIO_P1       10         MBED_GPIO_P2       10
3.8.1	3.7.2.2 3.7.2.3 IDs Detailed Define Do 3.8.2.1 3.8.2.2 3.8.2.3 3.8.2.4 3.8.2.5	MBED_GPIO_P1       10         MBED_GPIO_P2       10
3.8.1	3.7.2.2 3.7.2.3 PlDs Detailed Define Do 3.8.2.1 3.8.2.2 3.8.2.3 3.8.2.4 3.8.2.5 3.8.2.6	MBED_GPIO_P1       10         MBED_GPIO_P2       10
	3.7.1	3.6.2.15 3.6.2.16 3.6.2.17 3.6.2.18 3.6.2.19 3.6.2.20 3.6.2.21 3.6.2.22 3.6.2.23 3.6.2.24 3.6.2.25 3.6.2.25 3.6.2.26 3.6.2.27  Port configuration 3.7.1 Detailed 3.7.2 Define Def

iv CONTENTS

	3.9	CANop	en objects		9
		3.9.1	Detailed I	Description	9
		3.9.2	Define Do	ocumentation	9
			3.9.2.1	CAN_COB_EMCY	9
			3.9.2.2	CAN_COB_ERROR_CONTROL	9
			3.9.2.3	CAN_COB_NMT	9
			3.9.2.4	CAN_COB_RPDO1	9
			3.9.2.5	CAN_COB_RPDO2	0
			3.9.2.6	CAN_COB_RPDO3	0
			3.9.2.7	CAN_COB_RPDO4	0
			3.9.2.8	CAN_COB_RSDO	0
			3.9.2.9	CAN_COB_SYNC	0
			3.9.2.10	CAN_COB_TIME	0
			3.9.2.11	CAN_COB_TPDO1	0
			3.9.2.12	CAN_COB_TPDO2	0
			3.9.2.13	CAN_COB_TPDO3	0
			3.9.2.14	CAN_COB_TPDO4	0
			3.9.2.15	CAN_COB_TSDO	1
	3.10	Networ	k managei	ment commands	2
		3.10.1	Detailed I	Description	2
		3.10.2	Define Do	ocumentation	2
			3.10.2.1	CAN_NMT_PREOP	2
			3.10.2.2	CAN_NMT_RESET 22	2
			3.10.2.3	CAN_NMT_RSCOMM	2
			3.10.2.4	CAN_NMT_START	2
			3.10.2.5	CAN_NMT_STOP	2
					_
4			entation	23	
	4.1			rence	
		4.1.1		Description	
		4.1.2		ocumentation	
			4.1.2.1	MBED_H	
		4.1.3		Documentation	
			4.1.3.1	initSys	5

CONTENTS

	4.1.4	Variable	Documentation
		4.1.4.1	mbedStatus
4.2	mbed_	can.h File	Reference
	4.2.1	Detailed	Description
	4.2.2	Define D	ocumentation
		4.2.2.1	CAN_BUFSIZE
		4.2.2.2	MBED_CAN_H
	4.2.3	Function	Documentation
		4.2.3.1	CAN_IRQHandler
		4.2.3.2	CANopenRecv
		4.2.3.3	CANopenSend
		4.2.3.4	flushCANRxBuffer
		4.2.3.5	flushCANTxBuffer
		4.2.3.6	initCAN
		1237	setCANBaudrate 30

# **Chapter 1**

# **Module Index**

## 1.1 Modules

## Here is a list of all modules:

CAN baudrates
MBED_CAN0 buffers
MBED_CAN1 buffers
Initalization flags
Status flags
General purpose I/O pins
Port configuration
Device IDs
CANopen objects
Network management commands

2 Module Index

# Chapter 2

# File Index

## 2.1 File List

Here is a list of all documented files with brief descriptions:

ed.h	
In this file are all device definitions for the mbed platform	23
ed_can.h	
Various definitions for the CANopen implementation	26
ed_gpio.h	?
ed_led.h	?
ed_musb.h?	?
ed_serial.h	?
ad timer h	2

4 File Index

# **Chapter 3**

# **Module Documentation**

## 3.1 CAN baudrates

## 3.1.1 Detailed Description

Baudrates for the CAN interfaces. These are set via setCANBaudrate().

## 3.2 MBED\_CAN0 buffers

## 3.2.1 Detailed Description

Receive and transmit buffers for the MBED\_CAN0 interface.

## 3.3 MBED\_CAN1 buffers

## 3.3.1 Detailed Description

Receive and transmit buffers for the MBED\_CAN0 interface.

## 3.4 Initalization flags

#### **Defines**

#define INIT\_LED (1 << 1)</li>
#define INIT\_MUSB (1 << 2)</li>
#define INIT\_SERIAL0 (1 << 3)</li>
#define INIT\_SERIAL1 (1 << 4)</li>
#define INIT\_SERIAL2 (1 << 5)</li>
#define INIT\_CAN0 (1 << 6)</li>
#define INIT\_CAN1 (1 << 7)</li>
#define INIT\_TIMER0 (1 << 8)</li>
#define INIT\_TIMER1 (1 << 9)</li>
#define INIT\_TIMER2 (1 << 10)</li>
#define INIT\_TIMER3 (1 << 11)</li>

#define INIT\_GPI (1 << 12)</li>#define INIT\_GPO (1 << 13)</li>

## 3.4.1 Detailed Description

These flags indicate which functionality should be initialized.

## See also

initSys()

## 3.4.2 Define Documentation

```
3.4.2.1 #define INIT_CAN0 (1 << 6)
```

CAN on DIP 9/10 (CAN1).

3.4.2.2 #define INIT\_CAN1 (1 << 7)

CAN on DIP 29/30 (CAN2).

3.4.2.3  $\,$  #define INIT\_GPI (1 << 12)

Initialize general purpose input ports.

3.4.2.4 #define INIT\_GPO (1 << 13)

Initialize general purpose output ports.

3.4.2.5 #define INIT\_LED (1 << 1)

Initialize the four blue LEDs.

3.4.2.6 #define INIT\_MUSB (1 << 2)

Serial on mini USB-B port (UART0).

3.4.2.7 #define INIT\_SERIAL0 (1 << 3)

Serial on DIP 13/14 (UART1).

3.4.2.8 #define INIT\_SERIAL1 (1 << 4)

Serial on DIP 27/28 (UART2).

3.4.2.9 #define INIT\_SERIAL2 (1 << 5)

Serial on DIP 9/10 (UART3).

3.4.2.10 #define INIT\_TIMER0 (1 << 8)

Initialize TIMER0.

3.4.2.11 #define INIT\_TIMER1 (1 << 9)

Initialize TIMER1.

3.4.2.12 #define INIT\_TIMER2 (1 << 10)

Initialize TIMER2.

3.4.2.13 #define INIT\_TIMER3 (1 << 11)

Initialize TIMER3.

## 3.5 Status flags

#### **Defines**

- #define MBED SYS INIT (1 << 0)</li>
- #define MBED\_LED\_INIT (1 << 1)</li>
- #define MBED\_MUSB\_INIT (1 << 2)
- #define MBED\_SERIAL0\_INIT (1 << 3)</li>
- #define MBED\_SERIAL1\_INIT (1 << 4)
- #define MBED SERIAL2 INIT (1 << 5)
- #define MBED\_CAN0\_INIT (1 << 6)
- #define MBED CAN1 INIT (1 << 7)</li>
- #define MBED\_TIMER0\_INIT (1 << 8)
- #define MBED\_TIMER1\_INIT (1 << 9)
- #define MBED\_TIMER2\_INIT (1 << 10)
- #define MBED\_TIMER3\_INIT (1 << 11)
- #define MBED\_GPI\_INIT (1 << 12)</li>
- #define MBED\_GPO\_INIT (1 << 13)</li>

## 3.5.1 Detailed Description

These flags indicate which functionality has been successfully initialized.

## 3.5.2 Define Documentation

3.5.2.1 #define MBED\_CAN0\_INIT (1 << 6)

CAN1 initialized.

3.5.2.2 #define MBED\_CAN1\_INIT (1 << 7)

CAN2 initialized.

3.5.2.3 #define MBED\_GPI\_INIT (1 << 12)

General purpose inputs initialized.

3.5.2.4 #define MBED\_GPO\_INIT (1 << 13)

General purpose outputs initialized.

3.5.2.5 #define MBED\_LED\_INIT (1 << 1)

The four LEDs are initialized.

3.5 Status flags

3.5.2.6 #define MBED\_MUSB\_INIT (1 << 2)

Serial communication functionality on the mini USB-B port initialized.

3.5.2.7 #define MBED\_SERIAL0\_INIT (1 << 3)

UART1 initialized.

3.5.2.8 #define MBED\_SERIAL1\_INIT (1 << 4)

UART2 initialized.

3.5.2.9 #define MBED\_SERIAL2\_INIT (1 << 5)

UART3 initialized.

3.5.2.10 #define MBED\_SYS\_INIT (1 << 0)

System initialization complete.

3.5.2.11 #define MBED\_TIMER0\_INIT (1 << 8)

TIMER0 initialized.

3.5.2.12 #define MBED\_TIMER1\_INIT (1 << 9)

TIMER1 initialized.

3.5.2.13 #define MBED\_TIMER2\_INIT (1 << 10)

TIMER2 initialized.

3.5.2.14 #define MBED\_TIMER3\_INIT (1 << 11)

TIMER3 initialized.

## 3.6 General purpose I/O pins

## **Defines**

 #define MBED GPIOX (1 << 0)</li> #define MBED\_GPIO5 (1 << 1)</li> #define MBED\_GPIO6 (1 << 2)</li> • #define MBED\_GPIO7 (1 << 3) • #define MBED GPIO8 (1 << 4) • #define MBED\_GPIO9 (1 << 5) • #define MBED\_GPIO10 (1 << 6) #define MBED\_GPIO11 (1 << 7)</li> #define MBED\_GPIO12 (1 << 8)</li> • #define MBED\_GPIO13 (1 << 9) • #define MBED\_GPIO14 (1 << 10) #define MBED GPIO15 (1 << 11)</li> #define MBED\_GPIO16 (1 << 12)</li> #define MBED GPIO17 (1 << 13)</li> • #define MBED\_GPIO18 (1 << 14) • #define MBED\_GPIO19 (1 << 15) • #define MBED GPIO20 (1 << 16) #define MBED\_GPIO21 (1 << 17)</li> #define MBED GPIO22 (1 << 18)</li> • #define MBED\_GPIO23 (1 << 19) #define MBED GPIO24 (1 << 20)</li> #define MBED\_GPIO25 (1 << 21)</li> #define MBED GPIO26 (1 << 22)</li> • #define MBED GPIO27 (1 << 23) #define MBED GPIO28 (1 << 24)</li> #define MBED\_GPIO29 (1 << 25)</li>

## 3.6.1 Detailed Description

Enumeration of general purpose I/O pins.

• #define MBED\_GPIO30 (1 << 26)

#### 3.6.2 Define Documentation

```
3.6.2.1 \, #define MBED_GPIO10 (1 << 6)
```

DIP 10.

3.6.2.2 #define MBED\_GPIO11 (1 << 7)

DIP 11.

```
3.6.2.3 #define MBED_GPIO12 (1 << 8)
DIP 12.
3.6.2.4 #define MBED_GPIO13 (1 << 9)
DIP 13.
3.6.2.5 #define MBED_GPIO14 (1 << 10)
DIP 14.
3.6.2.6 #define MBED_GPIO15 (1 << 11)
DIP 15.
3.6.2.7 #define MBED_GPIO16 (1 << 12)
DIP 16.
3.6.2.8 #define MBED_GPIO17 (1 << 13)
DIP 17.
3.6.2.9 #define MBED_GPIO18 (1 << 14)
DIP 18.
3.6.2.10 #define MBED_GPIO19 (1 << 15)
DIP 19.
3.6.2.11 #define MBED_GPIO20 (1 << 16)
DIP 20.
3.6.2.12 #define MBED_GPIO21 (1 << 17)
DIP 21.
```

```
3.6.2.13 #define MBED_GPIO22 (1 << 18)
DIP 22.
3.6.2.14 #define MBED_GPIO23 (1 << 19)
DIP 23.
3.6.2.15 #define MBED_GPIO24 (1 << 20)
DIP 24.
3.6.2.16 #define MBED_GPIO25 (1 << 21)
DIP 25.
3.6.2.17 #define MBED_GPIO26 (1 << 22)
DIP 26.
3.6.2.18 #define MBED_GPIO27 (1 << 23)
DIP 27.
3.6.2.19 #define MBED_GPIO28 (1 << 24)
DIP 28.
3.6.2.20 #define MBED_GPIO29 (1 << 25)
DIP 29.
3.6.2.21 #define MBED_GPIO30 (1 << 26)
DIP 30.
3.6.2.22 #define MBED_GPIO5 (1 << 1)
```

DIP 5.

3.6.2.23 #define MBED\_GPIO6 (1 << 2)

DIP 6.

3.6.2.24  $\,$  #define MBED\_GPIO7 (1 << 3)

DIP 7.

3.6.2.25 #define MBED\_GPIO8 (1 << 4)

DIP 8.

3.6.2.26 #define MBED\_GPIO9 (1 << 5)

DIP 9.

3.6.2.27 #define MBED\_GPIOX (1 << 0)

Not configured.

## 3.7 Port configuration

#### **Defines**

- #define MBED\_GPIO\_P0 (MBED\_GPIO5 | MBED\_GPIO6 | MBED\_GPIO7 | MBED\_GPIO8 | MBED\_GPIO9 | MBED\_GPIO10 | MBED\_GPIO11 | MBED\_GPIO12 | MBED\_GPIO13 | MBED\_GPIO14 | MBED\_GPIO15 | MBED\_GPIO16 | MBED\_GPIO17 | MBED\_GPIO18 | MBED\_GPIO27 | MBED\_GPIO28 | MBED\_GPIO29 | MBED\_GPIO30)
- #define MBED\_GPIO\_P1 (MBED\_GPIO19 | MBED\_GPIO20)
- #define MBED\_GPIO\_P2 (MBED\_GPIO21 | MBED\_GPIO22 | MBED\_GPIO23 | MBED\_GPIO24 | MBED\_GPIO25 | MBED\_GPIO26)

#### 3.7.1 Detailed Description

Configuration of GPIO pins on the different ports.

#### 3.7.2 Define Documentation

3.7.2.1 #define MBED\_GPIO\_P0 (MBED\_GPIO5 | MBED\_GPIO6 | MBED\_GPIO7 | MBED\_GPIO8 | MBED\_GPIO9 | MBED\_GPIO10 | MBED\_GPIO11 | MBED\_GPIO12 | MBED\_GPIO13 | MBED\_GPIO14 | MBED\_GPIO15 | MBED\_GPIO16 | MBED\_GPIO17 | MBED\_GPIO18 | MBED\_GPIO27 | MBED\_GPIO28 | MBED\_GPIO29 | MBED\_GPIO30)

GPIO on port 0.

3.7.2.2 #define MBED\_GPIO\_P1 (MBED\_GPIO19 | MBED\_GPIO20)

GPIO on port 1.

3.7.2.3 #define MBED\_GPIO\_P2 (MBED\_GPIO21 | MBED\_GPIO22 | MBED\_GPIO23 | MBED\_GPIO24 | MBED\_GPIO25 | MBED\_GPIO26)

GPIO on port 2.

3.8 Device IDs 17

## 3.8 Device IDs

## **Defines**

- #define MBED SERIAL0 0
- #define MBED\_SERIAL1 1
- #define MBED\_SERIAL2 2
- #define MBED\_CAN0 0
- #define MBED CAN1 1
- #define MBED\_TIMER0 0
- #define MBED\_TIMER1 1
- #define MBED\_TIMER2 2
- #define MBED\_TIMER3 3

## 3.8.1 Detailed Description

ID enumeration of various devices.

## 3.8.2 Define Documentation

3.8.2.1 #define MBED\_CAN0 0

CAN1 on DIP 9/10.

3.8.2.2 #define MBED\_CAN1 1

CAN2 on DIP 29/30.

3.8.2.3 #define MBED\_SERIAL0 0

UART1 on DIP 13/14.

3.8.2.4 #define MBED\_SERIAL1 1

UART2 on DIP 27/28.

3.8.2.5 #define MBED\_SERIAL2 2

UART3 on DIP 9/10.

3.8.2.6 #define MBED\_TIMER0 0

TIMER0.

3.8.2.7 #define MBED\_TIMER1 1

TIMER1.

3.8.2.8 #define MBED\_TIMER2 2

TIMER2.

3.8.2.9 #define MBED\_TIMER3 3

TIMER3.

## 3.9 CANopen objects

#### **Defines**

```
    #define CAN_COB_NMT (0 << 7)</li>
```

- #define CAN\_COB\_SYNC (1 << 7)</li>
- #define CAN COB TIME (2 << 7)
- #define CAN COB EMCY (1 << 7)</li>
- #define CAN\_COB\_TPDO1 (3 << 7)
- #define CAN\_COB\_RPDO1 (4 << 7)
- #define CAN COB TPDO2 (5 << 7)
- #define CAN COB RPDO2 (6 << 7)
- #define CAN\_COB\_TPDO3 (7 << 7)
- #define CAN\_COB\_RPDO3 (8 << 7)
- #define CAN\_COB\_TPDO3 (8 << 7)</li>
   #define CAN\_COB\_TPDO4 (9 << 7)</li>
- #define CAN\_COB\_RPDO4 (10 << 7)
- #define CAN COB TSDO (11 << 7)</li>
- #define CAN COB RSDO (12 << 7)
- #define CAN\_COB\_ERROR\_CONTROL (14 << 7)</li>

## 3.9.1 Detailed Description

These define numeric identifiers for a set of CANopen object types.

## 3.9.2 Define Documentation

3.9.2.1 #define CAN\_COB\_EMCY (1 << 7)

Emergency.

3.9.2.2 #define CAN\_COB\_ERROR\_CONTROL (14 << 7)

Error control.

3.9.2.3 #define CAN\_COB\_NMT (0 << 7)

Network management.

3.9.2.4 #define CAN\_COB\_RPDO1 (4 << 7)

PDO1 receive.

3.9.2.5 #define CAN\_COB\_RPDO2 (6 << 7) PDO2 receive. 3.9.2.6 #define CAN\_COB\_RPDO3 (8 << 7) PDO3 receive. 3.9.2.7 #define CAN\_COB\_RPDO4 (10 << 7) PDO4 receive. 3.9.2.8 #define CAN\_COB\_RSDO (12 << 7) SDO receive. 3.9.2.9 #define CAN\_COB\_SYNC (1 << 7) Synchronization. 3.9.2.10 #define CAN\_COB\_TIME (2 << 7) Timestamp. 3.9.2.11 #define CAN\_COB\_TPDO1 (3 << 7) PDO1 transmit. 3.9.2.12 #define CAN\_COB\_TPDO2 (5 << 7) PDO2 transmit. 3.9.2.13 #define CAN\_COB\_TPDO3 (7 << 7)

3.9.2.14 #define CAN\_COB\_TPDO4 (9 << 7)

PDO4 transmit.

PDO3 transmit.

SDO transmit.

## 3.10 Network management commands

## **Defines**

- #define CAN\_NMT\_START 0x01
- #define CAN\_NMT\_STOP 0x02
- #define CAN\_NMT\_PREOP 0x80
- #define CAN\_NMT\_RESET 0x81
- #define CAN\_NMT\_RSCOMM 0x82

## 3.10.1 Detailed Description

CANOpen network management commands are used for node control.

## 3.10.2 Define Documentation

3.10.2.1 #define CAN\_NMT\_PREOP 0x80

Enter pre-operational state.

3.10.2.2 #define CAN\_NMT\_RESET 0x81

Reset node.

3.10.2.3 #define CAN\_NMT\_RSCOMM 0x82

Reset node communication.

3.10.2.4 #define CAN\_NMT\_START 0x01

Start node.

3.10.2.5 #define CAN\_NMT\_STOP 0x02

Stop node.

## **Chapter 4**

## **File Documentation**

## 4.1 mbed.h File Reference

In this file are all device definitions for the mbed platform.

```
#include "LPC17xx.h" #include "mbed_can.h" #include "mbed-
_gpio.h" #include "mbed_led.h" #include "mbed_musb.h" x
#include "mbed serial.h" #include "mbed timer.h"
```

#### **Defines**

- #define MBED\_H
- #define INIT\_LED (1 << 1)</li>
- #define INIT\_MUSB (1 << 2)
- #define INIT\_SERIAL0 (1 << 3)
- #define INIT\_SERIAL1 (1 << 4)</li>
- #define INIT\_SERIAL2 (1 << 5)</li>
- #define INIT\_CAN0 (1 << 6)
- #define INIT\_CAN1 (1 << 7)</li>
- #define INIT\_TIMER0 (1 << 8)
- #define INIT\_TIMER1 (1 << 9)
- #define INIT\_TIMER2 (1 << 10)
- #define INIT\_TIMER3 (1 << 11)</li>
- #define INIT CDI (1 < < 10)
- #define INIT\_GPI (1 << 12)</li>
- #define INIT\_GPO (1 << 13)</li>
- #define MBED\_SYS\_INIT (1 << 0)</li>
- #define MBED\_LED\_INIT (1 << 1)
- #define MBED\_MUSB\_INIT (1 << 2)</li>
- #define MBED\_SERIAL0\_INIT (1 << 3)</li>#define MBED\_SERIAL1\_INIT (1 << 4)</li>
- #define MBED\_SERIAL2\_INIT (1 << 5)
- #define MBED CANO INIT (1 << 6)

24 File Documentation

```
    #define MBED CAN1 INIT (1 << 7)</li>

    #define MBED_TIMER0_INIT (1 << 8)</li>

    #define MBED_TIMER1_INIT (1 << 9)</li>

    #define MBED_TIMER2_INIT (1 << 10)</li>

    #define MBED_TIMER3_INIT (1 << 11)</li>

    #define MBED GPI INIT (1 << 12)</li>

• #define MBED_GPO_INIT (1 << 13)

    #define MBED GPIOX (1 << 0)</li>

• #define MBED_GPIO5 (1 << 1)
• #define MBED GPIO6 (1 << 2)

    #define MBED GPIO7 (1 << 3)</li>

    #define MBED GPIO8 (1 << 4)</li>

    #define MBED_GPIO9 (1 << 5)</li>

    #define MBED_GPIO10 (1 << 6)</li>

    #define MBED GPIO11 (1 << 7)</li>

    #define MBED GPIO12 (1 << 8)</li>

    #define MBED GPIO13 (1 << 9)</li>

    #define MBED GPIO14 (1 << 10)</li>

    #define MBED_GPIO15 (1 << 11)</li>

    #define MBED_GPIO16 (1 << 12)</li>

• #define MBED_GPIO17 (1 << 13)

    #define MBED GPIO18 (1 << 14)</li>

    #define MBED GPIO19 (1 << 15)</li>

    #define MBED GPIO20 (1 << 16)</li>

    #define MBED GPIO21 (1 << 17)</li>

    #define MBED_GPIO22 (1 << 18)</li>

    #define MBED_GPIO23 (1 << 19)</li>

    #define MBED GPIO24 (1 << 20)</li>

    #define MBED GPIO25 (1 << 21)</li>

    #define MBED_GPIO26 (1 << 22)</li>

• #define MBED_GPIO27 (1 << 23)

    #define MBED GPIO28 (1 << 24)</li>

    #define MBED GPIO29 (1 << 25)</li>

    #define MBED GPIO30 (1 << 26)</li>

• #define MBED_GPIO_P0 (MBED_GPIO5 | MBED_GPIO6 | MBED_GPIO7 | M-
  BED GPIO8 | MBED GPIO9 | MBED GPIO10 | MBED GPIO11 | MBED GP-
  IO12 | MBED GPIO13 | MBED GPIO14 | MBED GPIO15 | MBED GPIO16 |
  MBED GPIO17 | MBED GPIO18 | MBED GPIO27 | MBED GPIO28 | MBED -
  GPIO29 | MBED_GPIO30)
• #define MBED GPIO P1 (MBED GPIO19 | MBED GPIO20)
• #define MBED_GPIO_P2 (MBED_GPIO21 | MBED_GPIO22 | MBED_GPIO23 |
  MBED GPIO24 | MBED GPIO25 | MBED GPIO26)
• #define MBED SERIAL0 0
```

#define MBED\_SERIAL1 1
#define MBED\_SERIAL2 2
#define MBED\_CAN0 0
#define MBED\_CAN1 1

- #define MBED TIMER0 0
- #define MBED\_TIMER1 1
- #define MBED\_TIMER2 2
- #define MBED TIMER3 3

#### **Functions**

• void initSys (uint32\_t)

System initialization function.

#### **Variables**

• uint32\_t mbedStatus

## 4.1.1 Detailed Description

In this file are all device definitions for the mbed platform. Via these defines, all hardware peripherals are identified and initialized. Initalization flags are used to indicate which peripherals should be initialized upons startup. Status flags set in mbedStatus then indicate which peripherals have been successfully initialized.

A number of device IDs is defined to identify various devices.

All mbed header files are included here for convience, so that including this header file is sufficient.

#### 4.1.2 Define Documentation

4.1.2.1 #define MBED\_H

Header guard.

## 4.1.3 Function Documentation

4.1.3.1 void initSys ( uint32\_t flags )

System initialization function.

This function initializes the LPC1768 clock registers and any other functionality that is required.

## **Parameters**

flags | Functionality to be initialized, indicated by setting corresponding bits.

26 File Documentation

#### See also

Initalization flags

#### 4.1.4 Variable Documentation

## 4.1.4.1 uint32\_t mbedStatus

Global status register.

See also

Status flags

## 4.2 mbed\_can.h File Reference

Various definitions for the CANopen implementation.

```
#include "LPC17xx.h" #include "lpc17xx_can.h" #include
"lpc17xx_pinsel.h"
```

## **Defines**

- #define MBED CAN H
- #define CAN COB NMT (0 << 7)</li>
- #define CAN\_COB\_SYNC (1 << 7)
- #define CAN\_COB\_TIME (2 << 7)
- #define CAN\_COB\_EMCY (1 << 7)
- #define CAN\_COB\_TPDO1 (3 << 7)</li>
- #define CAN\_COB\_RPDO1 (4 << 7)</li>
- #define CAN\_COB\_TPDO2 (5 << 7)</li>
- #define CAN\_COB\_RPDO2 (6 << 7)</li>
- #define CAN\_COB\_TPDO3 (7 << 7)</li>
- #define CAN\_COB\_RPDO3 (8 << 7)</li>
- #define CAN\_COB\_TPDO4 (9 << 7)</li>
- #define CAN\_COB\_RPDO4 (10 << 7)</li>
- #define CAN\_COB\_TSDO (11 << 7)
- #define CAN COB RSDO (12 << 7)</li>
- #define CAN\_COB\_ERROR\_CONTROL (14 << 7)
- #define CAN\_NMT\_START 0x01
- #define CAN\_NMT\_STOP 0x02
- #define CAN\_NMT\_PREOP 0x80
- #define CAN\_NMT\_RESET 0x81
- #define CAN\_NMT\_RSCOMM 0x82
- #define CAN\_BUFSIZE 256

#### **Functions**

```
• void setCANBaudrate (uint8 t, uint32 t)
```

Baudrate configuration function.

void initCAN (uint8\_t)

CAN initialization function.

uint8\_t CANopenRecv (uint8\_t, uint8\_t \*, uint32\_t \*, uint8\_t \*)

Receive a CAN message via CANopen protocol.

• uint8\_t CANopenSend (uint8\_t, uint8\_t, uint32\_t, uint8\_t, uint8\_t \*)

Send a CAN message via CANopen protocol.

void flushCANRxBuffer (uint8\_t)

Flush CAN receive buffers.

void flushCANTxBuffer (uint8\_t)

Flush CAN transmit buffers.

• void CAN\_IRQHandler (void)

CAN IRQ handler.

## 4.2.1 Detailed Description

Various definitions for the CANopen implementation. This file defines a basic number of objects and messages for the CANopen implementation provided by mbed-lib, and it provides function definitions.

## 4.2.2 Define Documentation

4.2.2.1 #define CAN\_BUFSIZE 256

CAN buffer size.

See also

```
MBED_CAN0 buffers MBED_CAN1 buffers
```

4.2.2.2 #define MBED\_CAN\_H

Header guard.

#### 4.2.3 Function Documentation

4.2.3.1 void CAN\_IRQHandler (void)

CAN IRQ handler.

This routine is called when either of the CAN devices raises an interrupt. Do no call this routine yourself.

4.2.3.2 uint8\_t CANopenRecv ( uint8\_t portNo, uint8\_t \* nodelD, uint32\_t \* cobType, uint8\_t \* data )

Receive a CAN message via CANopen protocol.

This routine reads an availabe message from the CAN receive buffer, if available.

#### **Parameters**

portNo	CAN interface to read from. Should be MBED_CAN0 or MBED_CAN1.
nodeID	Address of variable to store the CAN device ID that sent the message.
cobType	Address of variable to store the COB type of the message.
data	Pointer to data storage.

#### Returns

0 if a message was successfully read, or an integer larger than 0 when no message was available.

#### See also

CANopenSend()
CANopen objects

4.2.3.3 uint8\_t CANopenSend ( uint8\_t portNo, uint8\_t nodelD, uint32\_t cobType, uint8\_t len, uint8\_t \* data )

Send a CAN message via CANopen protocol.

This routine sends a message via the CAN interface. If the message could not be sent (because the internal transmit buffers are full), the message is buffered and sent when when of the internal transmit buffers becomes available.

## Parameters

portNo	CAN interface to send on. Should be MBED_CAN0 or MBED_CAN1.
nodeID	ID of the CAN device the message should be sent to.
cobType	COB type of the message.
len	Message length.
data	Data to be sent. The total number of bytes that will be sent depends on
	the provided message length, but will never be more than eight bytes.

#### Returns

 ${\tt 0}$  if a message was successfully sent or buffered, or an integer larger than  ${\tt 0}$  when both sending and buffering failed.

#### See also

CANopenRecv()
CANopen objects

## 4.2.3.4 void flushCANRxBuffer ( uint8\_t portNo )

Flush CAN receive buffers.

This routine flushes the CAN receive buffer by discarding all messages.

#### **Parameters**

portNo	CAN interface of which the buffer should be flushed. Should be MBED-
	_CAN0 or MBED_CAN1.

#### See also

flushCANTxBuffer()

## 4.2.3.5 void flushCANTxBuffer ( uint8\_t portNo )

Flush CAN transmit buffers.

This routine flushes the CAN transmit buffer by discarding all messages.

## Parameters

portNo	CAN interface of which the buffer should be flushed. Should be MBED-	
	_CAN0 or MBED_CAN1.	

## See also

flushCANRxBuffer()

## 4.2.3.6 void initCAN ( uint8\_t portNo )

CAN initialization function.

Initalizes the CAN interface. This function is called from initSys(), only call this function if you know what you are doing.

#### **Parameters**

portNo	CAN interface to be initialized. Should be MBED CANO or MBED CA-
,	N1.

## See also

initSys()

4.2.3.7 void setCANBaudrate ( uint8\_t portNo, uint32\_t baudrate )

Baudrate configuration function.

This function sets the baudrates for the MBED\_CAN0 and MBED\_CAN1 devices. Call this function before calling initSys().

## **Parameters**

ſ	portNo	CAN interface to set the baudrate for. Should be MBED_CAN0 or MB-
		ED_CAN1.
Ī	baudrate	Desired baudrate, in bits per second.