Travail Pratique Gestion de Train

1

Generated by Doxygen 1.7.6.1

Thu Jun 19 2014 20:01:51

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

BusCan.c
: BusCan configuration initialization, write ans read busCan . . . . . **??**

BusCan.h
: Content constants and header of functions used on BusCan.h . . . **??**

ControlTrain.c
: BusCan configuration . . . . . **??**

ControlTrain.h
: Content constants and header of functions used on BusCan.h . . . **??**

main.c
: Receive all datas from UART (XBee device) and send them . . . . . **??**

SendUARTFormat.c
: Contain functions to send frames on the UART with a defined . . . . **??**

SendUARTFormat.h
: Contain function to send frames on the UART with a defined . . . . **??**

uart.c
: Contain all functions to initialize, write and read on UART 0 and 3 . . **??**

uart.h
: Contain all functions to initialize, write and read on UART 0 and 3 . . **??**

Chapter 3

Data Structure Documentation

3.1 str_bus Struct Reference

`#include <BusCan.h>`

Data Fields

- uint32_t id_Bus
- uint8_t dlc
- uint8_t data [8]

3.1.1 Field Documentation

3.1.1.1 uint8_t data[8]

3.1.1.2 uint8_t dlc

3.1.1.3 uint32_t id_Bus

The documentation for this struct was generated from the following file:

- BusCan.h

Chapter 4

File Documentation

4.1 BusCan.c File Reference

BusCan configuration initialization, write ans read busCan.

`#include "BusCan.h"` Include dependency graph for BusCan.c:

Functions

- void Init_BusCan ()
- void Write_BusCan (str_bus *s)
- void Read_BusCan (str_bus *l)

4.1.1 Detailed Description

BusCan configuration initialization, write ans read busCan.

Author

Marques Rafael, Berger Antoine et David Da Silva

Version

1.0

Date

1 April 2014

4.1.2 Function Documentation

4.1.2.1 void Init_BusCan ( )

Initialization BusCan

4.1.2.2 void Read_BusCan (str_bus * /)

Read BusCan

Parameters

str_bus	contain the structure with the datas from CAN bus

4.1.2.3 void Write_BusCan (str_bus * *s*)

Write BusCan

Parameters

str_bus	
uint32_t	id_Bus uint8_t dlc uint8_t data[8]

4.2 BusCan.h File Reference

Content constants and header of functions used on BusCan.h.

`#include "LPC17xx.h"` Include dependency graph for BusCan.h: This graph shows which files directly or indirectly include this file:

Data Structures

- struct str_bus

Functions

- void Init_BusCan ()
- void Write_BusCan ()
- void Read_BusCan ()

4.2.1 Detailed Description

Content constants and header of functions used on BusCan.h.

Author

Marques Rafael, Berger Antoine et Da Silva David

Version

1.0

Date

1 avril 2014

4.2.2 Function Documentation

4.2.2.1 void Init_BusCan ( )

Initialization BusCan

4.2.2.2 void Read_BusCan ( )

4.2.2.3 void Write_BusCan ( )

4.3 ControlTrain.c File Reference

BusCan configuration.

`#include "ControlTrain.h"` Include dependency graph for ControlTrain.c:

Functions

- void StopGoTrain (str_bus *str, int statu)
- void ChangeDirection (str_bus *str, int NoTrain, int direction)
- void ChangeSpeed (str_bus *str, int NoTrain, int speed)
- void TurnLight (str_bus *str, int NoTrain, int statu)

4.3.1 Detailed Description

BusCan configuration.

Author

Marques Rafael, Berger Antoine et Da Silva David

Version

1.0

Date

1 avril 2014

4.3.2 Function Documentation

4.3.2.1 void ChangeDirection (str_bus * *str*, int *NoTrain*, int *direction*)

Switch train direction

Parameters

str_bus	Return the structure configured with the params
NoTrain	Train number
direction	Train direction

4.3.2.2 void ChangeSpeed (str_bus * *str*, int *NoTrain*, int *speed*)

Change speed of train

Parameters

str_bus	Return the structure configured with the params
NoTrain	Train number
speed	Train speed

4.3.2.3 void StopGoTrain (str_bus * *str*, int *statu*)

Stop and Go Power

Parameters

str_bus	Return the structure configured with the params
statu	Stop (0) and go (1)

4.3.2.4 void TurnLight (str_bus * *str*, int *NoTrain*, int *statu*)

Turn on/off lights of train

Parameters

str_bus	Return the structure configured with the params
NoTrain	Train number
statu	Turn the light on (1) or off (0)

4.4 ControlTrain.h File Reference

content constants and header of functions used on BusCan.h

`#include "LPC17xx.h"` `#include "BusCan.h"` Include dependency graph for ControlTrain.h: This graph shows which files directly or indirectly include this file:

Defines

- #define FORWARD_TRAIN 1
- #define BACK_TRAIN 2

Functions

- void StopGoTrain (str_bus *str, int statu)
- void ChangeSpeed (str_bus *str, int NoTrain, int speed)
- void ChangeDirection (str_bus *str, int NoTrain, int direction)
- void SwitchSelect (uint8_t NoSwitch, uint8_t position)
- void TurnLight (str_bus *str, int NoTrain, int statu)

Variables

- str_bus str

4.4.1 Detailed Description

content constants and header of functions used on BusCan.h

Author

Da Silva Andrade David, Antoine Berger, Dos Santos Rafael

Version

1.0

Date

19 June 2014

4.4.2 Define Documentation

4.4.2.1 #define BACK_TRAIN 2

4.4.2.2 #define FORWARD_TRAIN 1

4.4.3 Function Documentation

4.4.3.1 void ChangeDirection (str_bus * str, int NoTrain, int direction)

Switch train direction

Parameters

str_bus	Return the structure configured with the params
NoTrain	Train number
direction	Train direction

4.4.3.2 void ChangeSpeed (str_bus * str, int NoTrain, int speed)

Change speed of train

Parameters

str_bus	Return the structure configured with the params
NoTrain	Train number
speed	Train speed

4.4.3.3 void StopGoTrain (str_bus * str, int statu)

Stop and Go Power

Parameters

str_bus	Return the structure configured with the params
statu	Stop (0) and go (1)

4.4.3.4 void SwitchSelect (uint8_t NoSwitch, uint8_t position)

4.4.3.5 void TurnLight (str_bus * str, int NoTrain, int statu)

Turn on/off lights of train

Parameters

str_bus	Return the structure configured with the params
NoTrain	Train number
statu	Turn the light on (1) or off (0)

4.4.4 Variable Documentation

4.4.4.1 str_bus str

4.5 main.c File Reference

Receive all datas from UART (XBee device) and send them.

Functions

- void CAN_IRQHandler ()
- int atoi (char *str)
- int main (void)

4.5.1 Detailed Description

Receive all datas from UART (XBee device) and send them.

Author

Da Silva Andrade David, Antoine Berger, Dos Santos Rafael

Version

1.0

Date

19 June 2014 on the CAN bus to the Marklin station to control the trains

4.5.2 Function Documentation

4.5.2.1 int atoi (char * *str*)

String to integer

Parameters

str	String to be converted to a integer

Returns

The value of the string converted

4.5.2.2 void CAN_IRQHandler ( )

String to integer

Parameters

str	String to be converted to a integer

Returns

The value of the string converted

4.5.2.3 int main (void)

Receive all frame from UART (XBee) and process them to send to the Marklin station with the CAN device.

4.6 SendUARTFormat.c File Reference

Contain functions to send frames on the UART with a defined.

`#include "SendUARTFormat.h"` Include dependency graph for SendUARTFormat.c:

Functions

- void send_speed (uint8_t n_train, uint16_t speed_train)
- void send_direction (uint8_t n_train, bool direction)
- void send_lights (uint8_t n_train, bool state)

4.6.1 Detailed Description

Contain functions to send frames on the UART with a defined.

Author

Da Silva Andrade David, Antoine Berger, Dos Santos Rafael

Version

1.0

Date

19 June 2014 syntax to control the train.

4.6.2 Function Documentation

4.6.2.1 void send_direction (uint8_t *n_train*, bool *direction*)

Send the direction on the UART to control the chosen train.

Parameters

n_train	Train number
direction	The direction sent to the train

4.6.2.2 void send_lights (uint8_t *n_train*, bool *state*)

Send the state of the lights on the UART to control the chosen train.

Parameters

n_train	Train number
state	Turn on (true) or turn off (false) on the chosen train.

4.6.2.3 void send_speed (uint8_t *n_train*, uint16_t *speed_train*)

Send the speed on the UART to control the chosen train.

Parameters

n_train	Train number
speed_train	The speed sent to the train

4.7 SendUARTFormat.h File Reference

Contain function to send frames on the UART with a defined.

`#include "uart.h"` `#include "stdbool.h"` Include dependency graph for SendUARTFormat.h: This graph shows which files directly or indirectly include this file:

4.7.1 Detailed Description

Contain function to send frames on the UART with a defined.

Author

Da Silva Andrade David, Antoine Berger, Dos Santos Rafael

Version

1.0

Date

19 June 2014 syntax to control the train.

4.8 uart.c File Reference

Contain all functions to initialize, write and read on UART 0 and 3.

`#include "LPC17xx.h"` `#include "uart.h"` Include dependency graph for uart.c:

Functions

- void uart0_init (uint32_t baudrate)
- void uart3_init (uint32_t baudrate)
- void uart0_send (char ∗data, uint32_t length)
- uint32_t uart0_read (char ∗data, uint32_t length)
- uint32_t uart0_read_one_char (char ∗ch)
- void uart3_send (char ∗data, uint32_t length)
- uint32_t uart3_read (char ∗data, uint32_t length)
- uint32_t uart3_read_one_char (char ∗ch)

4.8.1 Detailed Description

Contain all functions to initialize, write and read on UART 0 and 3.

Author

Da Silva Andrade David, Antoine Berger, Dos Santos Rafael

Version

1.0

Date

19 June 2014

4.8.2 Function Documentation

4.8.2.1 void uart0_init (uint32_t *baudrate*)

Initialize UART0 port, setup pin select, clock, parity, stop bits, FIFO, etc.

Parameters

baudrate	UART0 baudrate [bit/s]

4.8.2.2 uint32_t uart0_read (char * *data,* uint32_t *length*)

Read data from UART0

Parameters

data	Pointer that store the data read from UART 0
length	Number of bytes to read

4.8.2.3 uint32_t uart0_read_one_char (char * *ch*)

Read one byte from UART0

Parameters

ch	Pointer that store the byte read from UART 0

4.8.2.4 void uart0_send (char * *data,* uint32_t *length*)

Send data on UART0

Parameters

data	Pointer on the datas to be sent
baudrate	Number of bytes to send

4.8.2.5 void uart3_init (uint32_t *baudrate*)

Initialize UART3 port, setup pin select, clock, parity, stop bits, FIFO, etc.

Parameters

baudrate	UART3 baudrate [bit/s]

4.8.2.6 uint32_t uart3_read (char * data, uint32_t length)

Read data from UART3

Parameters

data	Pointer that store the data read from UART 3
length	Number of bytes to read

4.8.2.7 uint32_t uart3_read_one_char (char * ch)

Read one byte from UART3

Parameters

ch	Pointer that store the byte read from UART 3

4.8.2.8 void uart3_send (char * data, uint32_t length)

Send data on UART3

Parameters

data	Pointer on the datas to be sent
baudrate	Number of bytes to send

4.9 uart.h File Reference

Contain all functions to initialize, write and read on UART 0 and 3.

`#include <stdint.h>` Include dependency graph for uart.h: This graph shows which files directly or indirectly include this file:

Defines

- #define LSR_RDR 0x01
- #define LSR_OE 0x02
- #define LSR_PE 0x04
- #define LSR_FE 0x08
- #define LSR_BI 0x10
- #define LSR_THRE 0x20
- #define LSR_TEMT 0x40
- #define LSR_RXFE 0x80

Functions

- void uart0_init (uint32_t baudrate)
- void uart0_send (char ∗data, uint32_t length)
- uint32_t uart0_read (char ∗data, uint32_t length)
- uint32_t uart0_read_one_char (char ∗ch)
- void uart3_init (uint32_t baudrate)
- void uart3_send (char ∗data, uint32_t length)
- uint32_t uart3_read (char ∗data, uint32_t length)
- uint32_t uart3_read_one_char (char ∗ch)

4.9.1 Detailed Description

Contain all functions to initialize, write and read on UART 0 and 3.

Author

Da Silva Andrade David, Antoine Berger, Dos Santos Rafael

Version

1.0

Date

19 June 2014

4.9.2 Define Documentation

4.9.2.1 #define LSR_BI 0x10

4.9.2.2 #define LSR_FE 0x08

4.9.2.3 #define LSR_OE 0x02

4.9.2.4 #define LSR_PE 0x04

4.9.2.5 #define LSR_RDR 0x01

4.9.2.6 #define LSR_RXFE 0x80

4.9.2.7 #define LSR_TEMT 0x40

4.9.2.8 #define LSR_THRE 0x20

4.9.3 Function Documentation

4.9.3.1 void uart0_init (uint32_t *baudrate*)

Initialize UART0 port, setup pin select, clock, parity, stop bits, FIFO, etc.

Parameters

baudrate	UART0 baudrate [bit/s]

4.9.3.2 uint32_t uart0_read (char * *data*, uint32_t *length*)

Read data from UART0

Parameters

data	Pointer that store the data read from UART 0
length	Number of bytes to read

4.9.3.3 uint32_t uart0_read_one_char (char * *ch*)

Read one byte from UART0

Parameters

ch	Pointer that store the byte read from UART 0

4.9.3.4 void uart0_send (char * *data*, uint32_t *length*)

Send data on UART0

Parameters

data	Pointer on the datas to be sent
baudrate	Number of bytes to send

4.9.3.5 void uart3_init (uint32_t *baudrate*)

Initialize UART3 port, setup pin select, clock, parity, stop bits, FIFO, etc.

Parameters

baudrate	UART3 baudrate [bit/s]

4.9.3.6 uint32_t uart3_read (char * *data*, uint32_t *length*)

Read data from UART3

Parameters

data	Pointer that store the data read from UART 3
length	Number of bytes to read

4.9.3.7 uint32_t uart3_read_one_char (char * *ch*)

Read one byte from UART3

Parameters

ch	Pointer that store the byte read from UART 3

4.9.3.8 void uart3_send (char * *data*, uint32_t *length*)

Send data on UART3

Parameters

data	Pointer on the datas to be sent
baudrate	Number of bytes to send