

CI2C

Généré par Doxygen 1.8.8

Mardi 17 Mars 2015 08 :23 :16

Table des matières

1	Classe CI2C	1
2	CI2C	3
3	Index des classes	5
3.1	Liste des classes	5
4	Index des fichiers	7
4.1	Liste des fichiers	7
5	Documentation des classes	9
5.1	Référence de la classe CI2C	9
5.1.1	Description détaillée	9
5.1.2	Documentation des constructeurs et destructeur	9
5.1.2.1	CI2C	9
5.1.2.2	~CI2C	10
5.1.3	Documentation des fonctions membres	10
5.1.3.1	readByte	10
5.1.3.2	readWord	10
5.1.3.3	setSlaveAddr	10
5.1.3.4	setupInterface	10
5.1.3.5	writeByte	10
5.1.3.6	writeWord	11
5.1.4	Documentation des données membres	11
5.1.4.1	dev	11
6	Documentation des fichiers	13
6.1	Référence du fichier /home/jam/Bureau/C++/Classes/CI2C/CI2C.cpp	13
6.2	Référence du fichier /home/jam/Bureau/C++/Classes/CI2C/CI2C.h	13
6.3	Référence du fichier /home/jam/Bureau/C++/Classes/CI2C/README.md	13

Chapitre 1

Classe CI2C

Auteur

Simon Moinet

Date

Mars 2015

Cette classe [CI2C](#) permet de lire/écrire les valeurs de registres des esclaves d'un bus I2C

Chapitre 2

CI2C

Classes [CI2C](#)

Classes de communication bus I2C

Chapitre 3

Index des classes

3.1 Liste des classes

Liste des classes, structures, unions et interfaces avec une brève description :

CI2C	9
------------	---

Chapitre 4

Index des fichiers

4.1 Liste des fichiers

Liste de tous les fichiers avec une brève description :

/home/jam/Bureau/C++/Classes/CI2C/CI2C.cpp	13
/home/jam/Bureau/C++/Classes/CI2C/CI2C.h	13

Chapitre 5

Documentation des classes

5.1 Référence de la classe CI2C

```
#include <CI2C.h>
```

Fonctions membres publiques

- `CI2C ()`
Constructeur de la classe `CI2C`.
- `~CI2C ()`
Destructeur de la classe `CI2C`.
- `bool setupInterface (string dev, __u8 addr_slave)`
Methode `setupInterface`.
- `bool setSlaveAddr (__u8 addr_slave)`
Paramétrage de la nouvelle adresse de l'esclave sur le bus I2C.
- `int readByte (__u8 reg=0)`
Lecture du registre sur un octet. Par default la valeur est fixé à zéro dans le cas d'un esclave sans registre.
- `int readWord (__u8 reg=0)`
Lecture du registre sur deux octets. Par default la valeur est fixé à zéro dans le cas d'un esclave sans registre.
- `bool writeByte (__u8 reg, __u8 data)`
Ecriture dans un registre sur un octet.
- `bool writeWord (__u8 reg, __u16 data)`
Ecriture dans un registre sur deux octets.

Attributs privés

- `int dev`
Contient les informations du bus I2C.

5.1.1 Description détaillée

Cette classe `CI2C` permet de lire/écrire les valeurs de registres des esclaves d'un bus I2C

Définition à la ligne 24 du fichier `CI2C.h`.

5.1.2 Documentation des constructeurs et destructeur

5.1.2.1 `CI2C::CI2C ()`

Constructeur de la classe `CI2C`.

Ce constructeur initialise les attributs de la classe.

5.1.2.2 `CI2C::~~CI2C ()`

Destructeur de la classe `CI2C`.

5.1.3 Documentation des fonctions membres

5.1.3.1 `int CI2C::readByte (__u8 reg = 0)`

Lecture du registre sur un octet. Par default la valeur est fixé à zéro dans le cas d'un esclave sans registre.

Paramètres

<code>__u8</code>	<code>reg</code> : le registre que l'on veut lire
-------------------	---

Renvoie

Un int contenant l'octet du registre lu

5.1.3.2 `int CI2C::readWord (__u8 reg = 0)`

Lecture du registre sur deux octets. Par default la valeur est ficé à zéro dans le cas d'un esclave sans registre.

Paramètres

<code>__u8</code>	<code>reg</code> : le registre que l'on veut lire
-------------------	---

Renvoie

Un int contenant les deux octets du registre lu

5.1.3.3 `bool CI2C::setSlaveAddr (__u8 addr_slave)`

Paramétrage de la nouvelle adresse de l'esclave sur le bus I2C.

Paramètres

<code>__u8</code>	<code>addr_slave</code> : la nouvelle adresse de l'esclave
-------------------	--

Renvoie

true si l'initialisation à reussi false sinon

5.1.3.4 `bool CI2C::setupInterface (string dev, __u8 addr_slave)`

Methode setupInterface.

Initialisation de l'interface I2C et de son esclave

Paramètres

<code>in</code>	<code>string</code>	<code>dev</code> : le nom de l'interface I2C
<code>in</code>	<code>__u8</code>	<code>addr_slave</code> : l'adresse de l'esclave sur le bus I2C

Renvoie

true si l'initialisation à réussi false sinon

5.1.3.5 `bool CI2C::writeByte (__u8 reg, __u8 data)`

Ecriture dans un registre sur un octet.

Paramètres

<code>__u8</code>	reg : le registre où on veut écrire
<code>__u8</code>	data : l'octet que l'on veut écrire dans le registre

Renvoie

true si l'écriture a réussi, false sinon

5.1.3.6 bool Cl2C::writeWord (__u8 reg, __u16 data)

Ecriture dans un registre sur deux octets.

Paramètres

<code>__u8</code>	reg : le registre où on veut écrire
<code>__u16</code>	data : les deux octets que l'on veut écrire dans le registre

Renvoie

true si l'écriture a réussi, false sinon

5.1.4 Documentation des données membres

5.1.4.1 int Cl2C::dev [private]

Contient les informations du bus I2C.

Définition à la ligne 27 du fichier Cl2C.h.

La documentation de cette classe a été générée à partir du fichier suivant :

— /home/jam/Bureau/C++/Classes/Cl2C/[Cl2C.h](#)

Chapitre 6

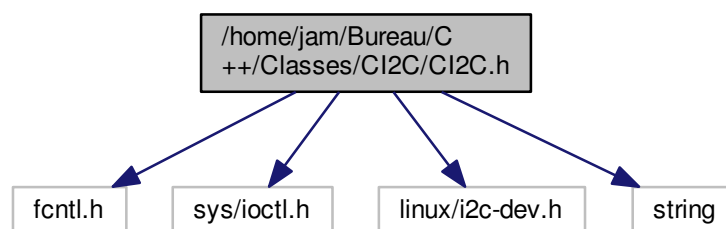
Documentation des fichiers

6.1 Référence du fichier /home/jam/Bureau/C++/Classes/CI2C/CI2C.cpp

6.2 Référence du fichier /home/jam/Bureau/C++/Classes/CI2C/CI2C.h

```
#include <fcntl.h>
#include <sys/ioctl.h>
#include <linux/i2c-dev.h>
#include <string>
```

Graphe des dépendances par inclusion de CI2C.h :



Classes

— class [CI2C](#)

6.3 Référence du fichier /home/jam/Bureau/C++/Classes/CI2C/README.md