

CI2C

Généré par Doxygen 1.8.8

Vendredi 27 Mars 2015 11 :27 :33



# Table des matières

<b>1</b>	<b>Classe CI2C</b>	<b>1</b>
<b>2</b>	<b>CI2C</b>	<b>3</b>
<b>3</b>	<b>Index hiérarchique</b>	<b>5</b>
3.1	Hiérarchie des classes . . . . .	5
<b>4</b>	<b>Index des classes</b>	<b>7</b>
4.1	Liste des classes . . . . .	7
<b>5</b>	<b>Index des fichiers</b>	<b>9</b>
5.1	Liste des fichiers . . . . .	9
<b>6</b>	<b>Documentation des classes</b>	<b>11</b>
6.1	Référence de la classe CI2C . . . . .	11
6.1.1	Description détaillée . . . . .	12
6.1.2	Documentation des constructeurs et destructeur . . . . .	12
6.1.2.1	CI2C . . . . .	12
6.1.2.2	~CI2C . . . . .	12
6.1.3	Documentation des fonctions membres . . . . .	12
6.1.3.1	isSetInterface . . . . .	12
6.1.3.2	readByte . . . . .	12
6.1.3.3	readWord . . . . .	13
6.1.3.4	setSlaveAddr . . . . .	13
6.1.3.5	setupInterface . . . . .	13
6.1.3.6	writeByte . . . . .	14
6.1.3.7	writeWord . . . . .	14
6.1.4	Documentation des données membres . . . . .	14
6.1.4.1	dev . . . . .	14
6.1.4.2	isSetDev . . . . .	14
6.1.4.3	isSetSlave . . . . .	14
6.2	Référence de la classe CI2C : :ErreurDevNotDefine . . . . .	15
6.2.1	Description détaillée . . . . .	15

6.2.2	Documentation des fonctions membres	15
6.2.2.1	what	15
6.3	Référence de la classe CI2C : :ErreurOpenDev	16
6.3.1	Description détaillée	16
6.3.2	Documentation des constructeurs et destructeur	17
6.3.2.1	ErreurOpenDev	17
6.3.3	Documentation des fonctions membres	17
6.3.3.1	what	17
6.3.4	Documentation des données membres	17
6.3.4.1	m_devName	17
6.4	Référence de la classe CI2C : :ErreurRead	17
6.4.1	Description détaillée	18
6.4.2	Documentation des constructeurs et destructeur	18
6.4.2.1	ErreurRead	18
6.4.3	Documentation des fonctions membres	18
6.4.3.1	what	18
6.4.4	Documentation des données membres	19
6.4.4.1	m_reg	19
6.5	Référence de la classe CI2C : :ErreurSetAddrSlave	19
6.5.1	Description détaillée	20
6.5.2	Documentation des constructeurs et destructeur	20
6.5.2.1	ErreurSetAddrSlave	20
6.5.3	Documentation des fonctions membres	20
6.5.3.1	what	20
6.5.4	Documentation des données membres	20
6.5.4.1	m_addr	20
6.6	Référence de la classe CI2C : :ErreurSlaveNotDefine	20
6.6.1	Description détaillée	21
6.6.2	Documentation des fonctions membres	21
6.6.2.1	what	21
6.7	Référence de la classe CI2C : :ErreurWrite	21
6.7.1	Description détaillée	22
6.7.2	Documentation des constructeurs et destructeur	22
6.7.2.1	ErreurWrite	22
6.7.3	Documentation des fonctions membres	22
6.7.3.1	what	22
6.7.4	Documentation des données membres	23
6.7.4.1	m_data	23
6.7.4.2	m_reg	23

<b>7</b>	<b>Documentation des fichiers</b>	<b>25</b>
7.1	Référence du fichier /home/jam/Bureau/C++/Classes/CI2C/CI2C.cpp . . . . .	25
7.1.1	Description détaillée . . . . .	25
7.2	Référence du fichier /home/jam/Bureau/C++/Classes/CI2C/CI2C.h . . . . .	25
7.2.1	Description détaillée . . . . .	26
7.2.2	Documentation des macros . . . . .	26
7.2.2.1	I2C_SLAVE . . . . .	27
7.3	Référence du fichier /home/jam/Bureau/C++/Classes/CI2C/README.md . . . . .	27



# 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 hiérarchique

### 3.1 Hiérarchie des classes

Cette liste d'héritage est classée approximativement par ordre alphabétique :

CI2C . . . . .	11
exception	
CI2C : :ErreurDevNotDefine . . . . .	15
CI2C : :ErreurOpenDev . . . . .	16
CI2C : :ErreurRead . . . . .	17
CI2C : :ErreurSetAddrSlave . . . . .	19
CI2C : :ErreurSlaveNotDefine . . . . .	20
CI2C : :ErreurWrite . . . . .	21



## Chapitre 4

# Index des classes

### 4.1 Liste des classes

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

<a href="#">CI2C</a>	Cette classe <a href="#">CI2C</a> permet de lire/écrire les valeurs de registres des esclaves d'un bus I2C . . .	11
<a href="#">CI2C : :ErreurDevNotDefine</a>	. . . . .	15
<a href="#">CI2C : :ErreurOpenDev</a>	. . . . .	16
<a href="#">CI2C : :ErreurRead</a>	Cette classe est une exception qui est levé lorsqu'il y a un erreur de lecture . . . . .	17
<a href="#">CI2C : :ErreurSetAddrSlave</a>	. . . . .	19
<a href="#">CI2C : :ErreurSlaveNotDefine</a>	. . . . .	20
<a href="#">CI2C : :ErreurWrite</a>	Cette classe est une esception qui est levé lorsqu'il y a une erreur d'écriture . . . . .	21



## Chapitre 5

# Index des fichiers

### 5.1 Liste des fichiers

Liste de tous les fichiers avec une brève description :

<a href="#">/home/jam/Bureau/C++/Classes/CI2C/CI2C.cpp</a>	
Contient la définition de la classe <a href="#">CI2C</a>	25
<a href="#">/home/jam/Bureau/C++/Classes/CI2C/CI2C.h</a>	
Contient la declaration de la <a href="#">CI2C</a>	25





## Chapitre 6

# Documentation des classes

### 6.1 Référence de la classe CI2C

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

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

#### Classes

- class [ErreurDevNotDefine](#)
- class [ErreurOpenDev](#)
- class [ErreurRead](#)  
*Cette classe est une exception qui est levé lorsqu'il y a un erreur de lecture.*
- class [ErreurSetAddrSlave](#)
- class [ErreurSlaveNotDefine](#)
- class [ErreurWrite](#)  
*Cette classe est une esception qui est levé lorsqu'il y a une erreur d'écriture.*

#### Fonctions membres publiques

- [CI2C](#) ()  
*Constructeur de la classe [CI2C](#).*
- [~CI2C](#) ()  
*Destructeur de la classe [CI2C](#).*
- void [setupInterface](#) (string [dev](#), \_\_u8 addr\_slave)  
*Methode setupInterface.*
- void [setSlaveAddr](#) (\_\_u8 addr\_slave)  
*Methode setSlaveAddr.*
- int [readByte](#) (\_\_u8 reg=0)  
*Methode readByte.*
- int [readWord](#) (\_\_u8 reg=0)  
*Methode readWord.*
- void [writeByte](#) (\_\_u8 reg, \_\_u8 data)  
*Methode writeByte.*
- void [writeWord](#) (\_\_u8 reg, \_\_u16 data)  
*Methode writeWord.*

#### Fonctions membres privées

- void [isSetInterface](#) ()  
*Methode isSetInterface.*

#### Attributs privés

- int [dev](#)

- *Contient les informations du bus I2C.*  
bool [isSetDev](#)  
*Bool qui permet de savoir si le dev est défini.*
- bool [isSetSlave](#)  
*Bool qui permet de savoir si l'esclave est défini.*

### 6.1.1 Description détaillée

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

Définition à la ligne 30 du fichier CI2C.h.

### 6.1.2 Documentation des constructeurs et destructeur

#### 6.1.2.1 CI2C::CI2C ( )

Constructeur de la classe [CI2C](#).

Ce constructeur initialise les attributs de la classe.

Définition à la ligne 9 du fichier CI2C.cpp.

#### 6.1.2.2 CI2C::~~CI2C ( )

Destructeur de la classe [CI2C](#).

Définition à la ligne 17 du fichier CI2C.cpp.

### 6.1.3 Documentation des fonctions membres

#### 6.1.3.1 void CI2C::isSetInterface ( ) [private]

Methode isSetInterface.

Cette méthode permet de vérifier si le dev et l'esclave sont bien défini

##### Exceptions

<i>StringIndexOutOfRangeException</i> <i>Exception</i>	if index is not between 0 and length() - 1.
<i>Erreur(TYPE_ERREUR</i>	erreur, string const& phrase) Si le dev n'est pas défini, leve une exception de type Erreur : :TYPE_ERREUR : :DEV_DEF
<i>Erreur(TYPE_ERREUR</i>	erreur, string const& phrase) Si l'esclave n'est pas défini, leve une exception de type Erreur : :TYPE_ERREUR : :SLAVE_DEF

Définition à la ligne 48 du fichier CI2C.cpp.

#### 6.1.3.2 int CI2C::readByte ( \_\_u8 reg = 0 )

Methode readByte.

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

##### Paramètres

<i>in</i>	<i>reg</i>	: le registre que l'on veut lire
-----------	------------	----------------------------------

**Renvoie**

Un int contenant l'octet du registre lu

Définition à la ligne 58 du fichier CI2C.cpp.

**6.1.3.3 int CI2C::readWord ( \_\_u8 reg = 0 )**

Methode readWord.

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

**Paramètres**

<i>in</i>	<i>reg</i>	: le registre que l'on veut lire
-----------	------------	----------------------------------

**Renvoie**

Un int contenant les deux octets du registre lu

Définition à la ligne 72 du fichier CI2C.cpp.

**6.1.3.4 void CI2C::setSlaveAddr ( \_\_u8 addr\_slave )**

Methode setSlaveAddr.

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

**Paramètres**

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

**Exceptions**

<i>Erreur(TYPE_ERREUR</i>	erreur, string const& phrase) Si le dev n'est pas défini, leve une exception de type Erreur : :TYPE_ERREUR : :DEV_DEF
<i>Erreur(TYPE_ERREUR</i>	erreur, string const& phrase) Si la spécification de l'adresse de l'esclave avec lequel on veut communiquer à échoué. Leve une exception de type Erreur : :TYPE_ERREUR : :SLAVE_DEF

Définition à la ligne 34 du fichier CI2C.cpp.

**6.1.3.5 void CI2C::setupInterface ( string dev, \_\_u8 addr\_slave )**

Methode setupInterface.

Initialisation de l'interface I2C et de son esclave.

**Paramètres**

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

**Exceptions**

<i>Erreur(TYPE_ERREUR</i>	erreur, string const& phrase) Si il est impossible de définir le device du bus I2C. Leve une exception de type Erreur : :TYPE_ERREUR : :DEV_DEF
---------------------------	---

<i>Erreur</i> ( <i>TYPE_ERREUR</i>	erreur, string const& phrase) Si la spécification de l'adresse de l'esclave avec lequel on veut communiquer à échoué. Leve une exception de type Erreur : :TYPE_ERREUR : :SLAVE_DEF
------------------------------------	---

Définition à la ligne 22 du fichier Cl2C.cpp.

#### 6.1.3.6 void Cl2C : :writeByte ( \_\_u8 reg, \_\_u8 data )

Methode writeByte.

Ecriture dans un registre sur un octet

##### Paramètres

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

##### Renvoie

true si l'écriture a réussi, false sinon

Définition à la ligne 86 du fichier Cl2C.cpp.

#### 6.1.3.7 void Cl2C : :writeWord ( \_\_u8 reg, \_\_u16 data )

Methode writeWord.

Ecriture dans un registre sur deux octets

##### Paramètres

in	<i>reg</i>	: le registre où on veut écrire
in	<i>data</i>	: les deux octets que l'on veut écrire dans le registre

##### Renvoie

true si l'écriture a réussi, false sinon

Définition à la ligne 98 du fichier Cl2C.cpp.

### 6.1.4 Documentation des données membres

#### 6.1.4.1 int Cl2C : :dev [private]

Contient les informations du bus I2C.

Définition à la ligne 137 du fichier Cl2C.h.

#### 6.1.4.2 bool Cl2C : :isSetDev [private]

Bool qui permet de savoir si le dev est défini.

Définition à la ligne 139 du fichier Cl2C.h.

#### 6.1.4.3 bool Cl2C : :isSetSlave [private]

Bool qui permet de savoir si l'esclave est défini.

Définition à la ligne 141 du fichier Cl2C.h.

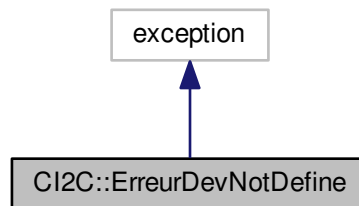
La documentation de cette classe a été générée à partir des fichiers suivants :

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

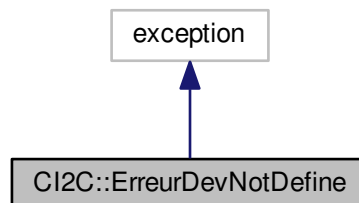
## 6.2 Référence de la classe CI2C : :ErreurDevNotDefine

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

Graphe d'héritage de CI2C : :ErreurDevNotDefine :



Graphe de collaboration de CI2C : :ErreurDevNotDefine :



### Fonctions membres publiques

- virtual const char \* `what` () const noexcept

#### 6.2.1 Description détaillée

Définition à la ligne 121 du fichier CI2C.h.

#### 6.2.2 Documentation des fonctions membres

6.2.2.1 virtual const char\* CI2C : :ErreurDevNotDefine : :`what` ( ) const [inline],[virtual],[noexcept]

Définition à la ligne 123 du fichier CI2C.h.

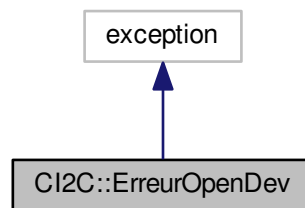
La documentation de cette classe a été générée à partir du fichier suivant :

— /home/jam/Bureau/C++/Classes/CI2C/CI2C.h

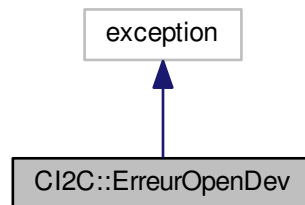
### 6.3 Référence de la classe CI2C : :ErreurOpenDev

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

Graphe d'héritage de CI2C : :ErreurOpenDev :



Graphe de collaboration de CI2C : :ErreurOpenDev :



#### Fonctions membres publiques

- `ErreurOpenDev` (string devName)
- `virtual const char * what () const noexcept`

#### Attributs privés

- string `m_devName`

#### 6.3.1 Description détaillée

Définition à la ligne 96 du fichier CI2C.h.

### 6.3.2 Documentation des constructeurs et destructeur

#### 6.3.2.1 CI2C : :ErreurOpenDev : :ErreurOpenDev ( string devName ) [inline]

Définition à la ligne 101 du fichier CI2C.h.

### 6.3.3 Documentation des fonctions membres

#### 6.3.3.1 virtual const char\* CI2C : :ErreurOpenDev : :what ( ) const [inline],[virtual],[noexcept]

Définition à la ligne 102 du fichier CI2C.h.

### 6.3.4 Documentation des données membres

#### 6.3.4.1 string CI2C : :ErreurOpenDev : :m\_devName [private]

Définition à la ligne 98 du fichier CI2C.h.

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

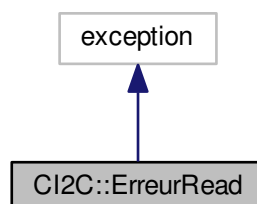
— /home/jam/Bureau/C++/Classes/CI2C/[CI2C.h](#)

## 6.4 Référence de la classe CI2C : :ErreurRead

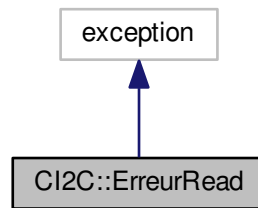
Cette classe est une exception qui est levé lorsqu'il y a un erreur de lecture.

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

Graphe d'héritage de CI2C : :ErreurRead :



Graphe de collaboration de CI2C : :ErreurRead :



### Fonctions membres publiques

- [ErreurRead](#) (int reg)  
*Contient le registre que l'on a essayé de lire.*
- virtual const char \* [what](#) () const noexcept  
*Methode what.*

### Attributs privés

- int [m\\_reg](#)

#### 6.4.1 Description détaillée

Cette classe est une exception qui est levé lorsqu'il y a un erreur de lecture.

Définition à la ligne 70 du fichier CI2C.h.

#### 6.4.2 Documentation des constructeurs et destructeur

##### 6.4.2.1 CI2C : :ErreurRead : :ErreurRead ( int reg ) [inline]

Contient le registre que l'on a essayé de lire.

Constructeur de la classe [ErreurRead](#)

Ce constructeur initialise les attributs de la classe [ErreurWrite](#)

##### Paramètres

in	reg	: le registre que l'on a essayé de lire
----	-----	---

Définition à la ligne 82 du fichier CI2C.h.

#### 6.4.3 Documentation des fonctions membres

##### 6.4.3.1 virtual const char\* CI2C : :ErreurRead : :what ( ) const [inline],[virtual],[noexcept]

Methode what.



#### Renvoie

la description de l'exception avec la valeur du registre que l'on a essayé de lire

Définition à la ligne 89 du fichier CI2C.h.

### 6.4.4 Documentation des données membres

6.4.4.1 `int CI2C : :ErreurRead : :m_reg [private]`

Définition à la ligne 72 du fichier CI2C.h.

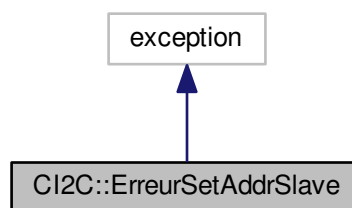
La documentation de cette classe a été générée à partir du fichier suivant :

— /home/jam/Bureau/C++/Classes/CI2C/CI2C.h

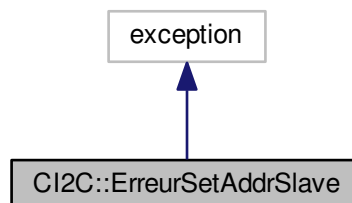
## 6.5 Référence de la classe CI2C : :ErreurSetAddrSlave

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

Graphe d'héritage de CI2C : :ErreurSetAddrSlave :



Graphe de collaboration de CI2C : :ErreurSetAddrSlave :



### Fonctions membres publiques

- `ErreurSetAddrSlave (int addr)`
- `virtual const char * what () const noexcept`

## Attributs privés

— int [m\\_addr](#)

### 6.5.1 Description détaillée

Définition à la ligne 108 du fichier CI2C.h.

### 6.5.2 Documentation des constructeurs et destructeur

6.5.2.1 `CI2C : :ErreurSetAddrSlave : :ErreurSetAddrSlave ( int addr )` `[inline]`

Définition à la ligne 113 du fichier CI2C.h.

### 6.5.3 Documentation des fonctions membres

6.5.3.1 `virtual const char* CI2C : :ErreurSetAddrSlave : :what ( ) const` `[inline],[virtual],[noexcept]`

Définition à la ligne 114 du fichier CI2C.h.

### 6.5.4 Documentation des données membres

6.5.4.1 `int CI2C : :ErreurSetAddrSlave : :m_addr` `[private]`

Définition à la ligne 110 du fichier CI2C.h.

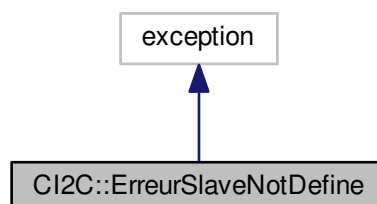
La documentation de cette classe a été générée à partir du fichier suivant :

— [/home/jam/Bureau/C++/Classes/CI2C/CI2C.h](#)

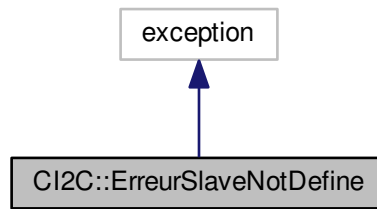
## 6.6 Référence de la classe CI2C : :ErreurSlaveNotDefine

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

Graphe d'héritage de CI2C : :ErreurSlaveNotDefine :



Graphe de collaboration de CI2C : :ErreurSlaveNotDefine :



### Fonctions membres publiques

— virtual const char \* [what](#) () const noexcept

#### 6.6.1 Description détaillée

Définition à la ligne 128 du fichier CI2C.h.

#### 6.6.2 Documentation des fonctions membres

6.6.2.1 virtual const char\* CI2C : :ErreurSlaveNotDefine : :what ( ) const [inline],[virtual],[noexcept]

Définition à la ligne 130 du fichier CI2C.h.

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

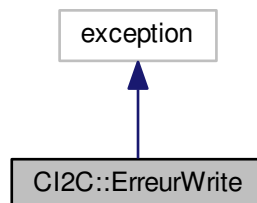
— /home/jam/Bureau/C++/Classes/CI2C/[CI2C.h](#)

## 6.7 Référence de la classe CI2C : :ErreurWrite

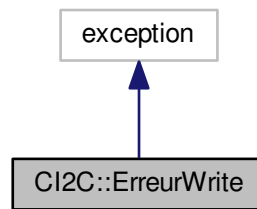
Cette classe est une exception qui est levé lorsqu'il y a une erreur d'écriture.

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

Graphe d'héritage de CI2C : :ErreurWrite :



Graphe de collaboration de Cl2C : :ErreurWrite :



### Fonctions membres publiques

- [ErreurWrite](#) (int reg, int data)  
*Contient la valeur qu'on a essayer d'écrire dans le registre.*
- virtual const char \* [what](#) () const noexcept  
*Methode what.*

### Attributs privés

- int [m\\_reg](#)
- int [m\\_data](#)  
*Contient le registre dans lequel on a essayé d'écrire.*

#### 6.7.1 Description détaillée

Cette classe est une esception qui est levé lorsqu'il y a une erreur d'écriture.

Définition à la ligne 38 du fichier Cl2C.h.

#### 6.7.2 Documentation des constructeurs et destructeur

##### 6.7.2.1 Cl2C : :ErreurWrite : :ErreurWrite ( int reg, int data ) [inline]

Contient la valeur qu'on a essayer d'écrire dans le registre.

Constructeur de la classe [ErreurWrite](#)

Ce constructeur initialise les attributs de la classe [ErreurWrite](#)

##### Paramètres

in	<i>reg</i>	: le registre dans lequel on essaye d'écrire
in	<i>data</i>	: la valeur que l'on essaye d'écrire dans le registre

Définition à la ligne 52 du fichier Cl2C.h.

#### 6.7.3 Documentation des fonctions membres

##### 6.7.3.1 virtual const char\* Cl2C : :ErreurWrite : :what ( ) const [inline],[virtual],[noexcept]

Methode what.

Renvoie

la description de l'exception avec la valeur et le registre qui ont provoqué l'erreur

Définition à la ligne 59 du fichier Cl2C.h.

## 6.7.4 Documentation des données membres

### 6.7.4.1 int Cl2C : :ErreurWrite : :m\_data [private]

Contient le registre dans lequel on a essayé d'écrire.

Définition à la ligne 41 du fichier Cl2C.h.

### 6.7.4.2 int Cl2C : :ErreurWrite : :m\_reg [private]

Définition à la ligne 40 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 7

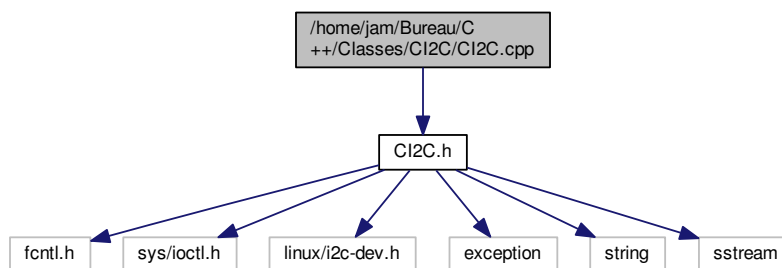
# Documentation des fichiers

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

Contient la définition de la classe [CI2C](#).

```
#include "CI2C.h"
```

Graphe des dépendances par inclusion de CI2C.cpp :



#### 7.1.1 Description détaillée

Contient la définition de la classe [CI2C](#).

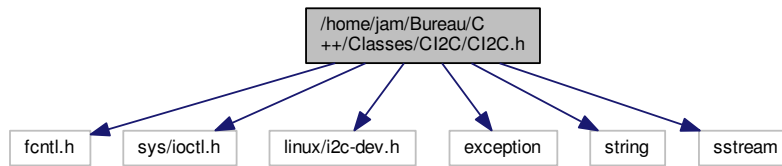
Définition dans le fichier [CI2C.cpp](#).

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

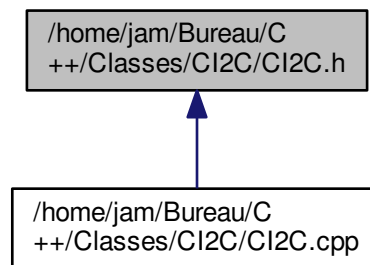
Contient la déclaration de la [CI2C](#).

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

Graphe des dépendances par inclusion de CI2C.h :



Ce graphe montre quels fichiers incluent directement ou indirectement ce fichier :



## Classes

- class `CI2C`  
*Cette classe `CI2C` permet de lire/écrire les valeurs de registres des esclaves d'un bus I2C.*
- class `CI2C : :ErreurWrite`  
*Cette classe est une exception qui est levé lorsqu'il y a une erreur d'écriture.*
- class `CI2C : :ErreurRead`  
*Cette classe est une exception qui est levé lorsqu'il y a un erreur de lecture.*
- class `CI2C : :ErreurOpenDev`
- class `CI2C : :ErreurSetAddrSlave`
- class `CI2C : :ErreurDevNotDefine`
- class `CI2C : :ErreurSlaveNotDefine`

## Macros

- `#define I2C_SLAVE 0x0703`

### 7.2.1 Description détaillée

Contient la declaration de la `CI2C`.

Définition dans le fichier `CI2C.h`.

### 7.2.2 Documentation des macros



**7.2.2.1 #define I2C\_SLAVE 0x0703**

Définition à la ligne 15 du fichier CI2C.h.

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