



LE  
BUS I<sup>2</sup>C<sup>®</sup>



# Sommaire

I. Présentation.....	3
I.I. Caractéristiques : .....	3
I.II. Principe : .....	4
I.III. D'autres bus bifilaires :.....	4
II. Le protocole I <sup>2</sup> C :.....	5
II.I. La prise de contrôle du bus :.....	5
II.II. La transmission d'un octet :.....	5
II.III. La transmission d'une adresse :.....	6
II.IV. Écriture d'une donnée :.....	7
II.V. Lecture d'une donnée :.....	7
III. La gestion des conflits :.....	8
III.I. Principe :.....	8
III.II. Exemple :.....	9
V. Les adresses réservées :.....	11
VI. Terminologie du bus I <sup>2</sup> C.....	12
VII. Terminologie d'un transfert sur le bus I <sup>2</sup> C.....	12
VIII. Bibliographie et liens.....	12

# I. Présentation

**L**e bus I<sup>2</sup>C ( Inter Integrated Circuit ) a été développé au début des années 80 par **Philips semiconductors** pour permettre de relier facilement à un microprocesseur les différents circuits d'un téléviseur moderne.



## *I.1. Caractéristiques :*

Le bus I<sup>2</sup>C permet de faire communiquer entre eux des composants électroniques très divers grâce à seulement trois fils : Un signal de donnée (SDA), un signal d'horloge (SCL), et un signal de référence électrique ( Masse ). Ceci permet de réaliser des équipements ayant des fonctionnalités très puissantes ( en apportant toute la puissance des systèmes microprogrammés ) et conservant un circuit imprimé très simple, par rapport à un schéma classique ( 8bits de données, 16 bits d'adresse + les bits de contrôle ).

Les données sont transmises en série à 100Kbits/s en mode standard et jusqu'à 400Kbits/s en mode rapide. Ce qui ouvre la porte de cette technologie à toutes les applications où la vitesse n'est pas primordiale.<sup>1</sup>

De nombreux fabricants ayant adopté le système, la variété des circuits disponibles disposant d'un port I<sup>2</sup>C est énorme : Ports d'E/S bidirectionnels, Convertisseurs A/N et N/A, Mémoires (RAM,

---

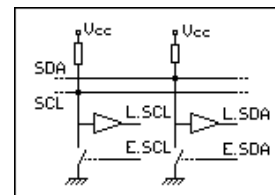
<sup>1</sup> Il existe maintenant un mode « haute vitesse » appelé High-Speed; la vitesse atteint 3,4 Mbits/s. Le mode multivitesse est possible.

EPROM, EEPROM, etc...), Circuits Audio ( Egaliseur, Contrôle de volume, ... ) et autre drivers ( LED , LCD , ... )

Le nombre de composants qu'il est ainsi possible de relier est essentiellement limité par la charge capacitive des lignes SDA et SCL :  $400\text{ pF}$  .

### ***1.II. Principe :***

Afin d'éviter les conflits électriques les Entrées/Sorties SDA et SCL sont de type « Collecteur Ouvert » ou « drain ouvert» et permet de réaliser la fonction ET câblé pour les broches de même nom. Le circuit peut ainsi s' auto surveiller et vérifier qu'il a bien sur sa sortie le niveau qu'il voulait y voir afficher.



*Illustration 1 Structure d'E/S d'un module I<sup>2</sup>C.*

### ***1.III. D'autres bus bifilaires :***

- **Le CBus de Philips (ancêtre du bus I<sup>2</sup>C.)**
- **Le bus SPI de Motorola.**
- **Le  $\mu$ Wire de National SemiConductor.**
- **Le 1-Wire de Maxim-Dallas.**

Plusieurs circuits pouvant être branchés en même temps sur le même bus, il a été nécessaire d'instaurer un protocole entre eux, afin d'éviter les problèmes dus à une « prise de parole » simultanée de différents modules. C'est le *protocole I<sup>2</sup>C*.

## II. Le protocole I<sup>2</sup>C :

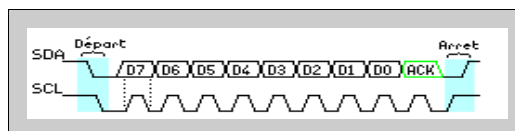
**L**e protocole I<sup>2</sup>C définit la succession des états logiques possibles sur SDA et SCL, et la façon dont doivent réagir les circuits en cas de conflits.



### ***II.I. La prise de contrôle du bus :***

Pour prendre le contrôle du bus, il faut que celui-ci soit au repos. ( SDA et SCL à '1'). Pour transmettre des données sur le bus, il faut donc surveiller deux conditions particulières :

- x La condition de départ. ( SDA passe à '0' alors que SCL reste à '1' )
- x La condition d'arrêt. ( SDA passe à '1' alors que SCL reste à '1' )



*Illustration 2 : Exemple de condition de départ et d'arrêt .*

Lorsqu'un circuit, après avoir vérifié que le bus est libre, prend le contrôle de celui-ci, il en devient le maître. C'est lui qui génère le signal d'horloge.

### ***II.II. La transmission d'un octet :***

Après avoir imposé la condition de départ, le maître applique sur SDA le bit de poids fort D7. Il valide ensuite la donnée en appliquant pendant un instant un niveau '1' sur la ligne SCL. Lorsque SCL revient à '0', il recommence l'opération jusqu'à

ce que l'octet complet soit transmis. Il envoie alors un bit ACK à '1' tout en scrutant l'état réel de SDA. L'esclave doit alors imposer un niveau '0' pour signaler au maître que la transmission s'est effectuée correctement. Les sorties de chacun étant à collecteurs ouverts, le maître voit le '0' et peut alors passer à la suite.

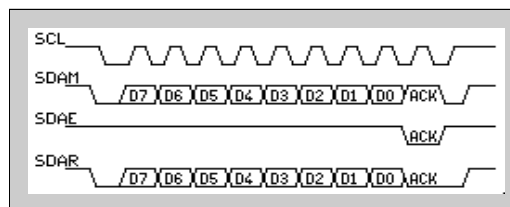


Illustration 3 : Exemple de transmission réussie.

**Dans cet exemple:**

- SCL : Horloge imposée par le maître.
- SDAM : Niveaux de SDA imposés par le maître.
- SDAE : Niveaux de SDA imposés par l'esclave.
- SDAR : Niveaux de SDA réels résultants.

### II.III. La transmission d'une adresse :

Le nombre de composants qu'il est possible de connecter sur un bus I<sup>2</sup>C étant largement supérieur à deux, il est nécessaire de définir pour chacun une adresse unique. L'adresse d'un circuit, codée sur sept bits, est définie d'une part par son type et d'autre part par l'état appliqué à un certain nombre de ses broches . Cette adresse est transmise sous la forme d'un octet au format particulier. On remarque ici que les bits D7 à D1 représentent les adresses A6 à A0, et que le bit D0 est remplacé par le bit de R/W qui

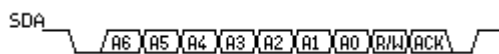


Illustration 4 Exemple d'octet d'adresse.

permet au maître de signaler s'il veut lire ou écrire une donnée.<sup>2 3</sup>

Le bit d'acquiescement ACK fonctionne comme pour une donnée,

2 Cas particulier des mémoires :

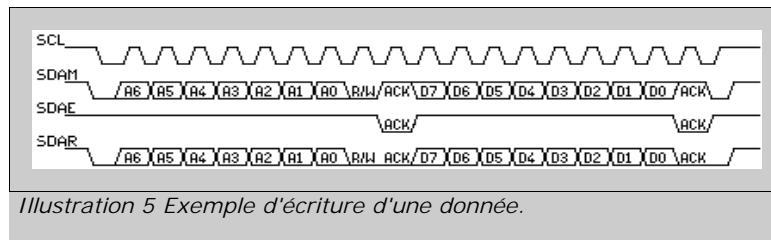
L'espace adressable d'un circuit de mémoire étant sensiblement plus grand que la plupart des autres types de circuits, l'adresse d'une information y est codée sur deux octets ou plus. Le premier représente toujours l'adresse du circuit, et les suivants l'adresse interne de la mémoire.

3 Les adresses réservées.

Les adresses 0000XXX et 111111XX sont réservées à des modes de fonctionnement particuliers.

ceci permet au maître de vérifier si l'esclave est disponible.

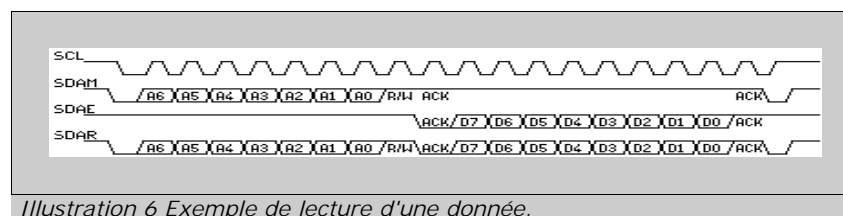
## II.IV. Écriture d'une donnée :



L'écriture d'une donnée par le maître ne pose généralement pas de problème particulier.<sup>4</sup>

## II.V. Lecture d'une donnée :

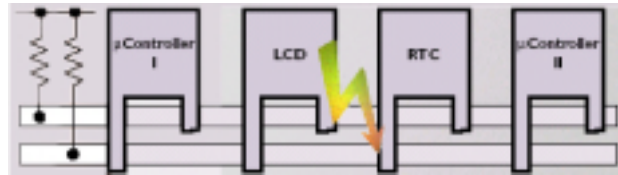
La lecture d'une donnée par le maître se caractérise par l'utilisation spéciale qui est faite du bit ACK. Après la lecture d'un octet, le maître positionne ACK à '0' s'il veut lire la donnée suivante (cas d'une mémoire par exemple) ou à '1' le cas échéant. Il envoie alors la condition d'arrêt.



<sup>4</sup> Cas particulier d'utilisation d'ACK : L'écriture d'un octet dans certains composants (Mémoires, microcontrôleur, ...) peut prendre un certain temps. Il est donc possible que le maître soit obligé d'attendre l'acquittement ACK avant de passer à la suite.

### III. La gestion des conflits :

La structure même du bus I<sup>2</sup>C a été conçue pour pouvoir accueillir plusieurs



maîtres. Se pose alors le problème commun à tout les réseaux utilisant un canal de communication unique : la prise de parole. En effet, chaque maître pouvant prendre possession du bus dès que celui-ci est libre, il existe la possibilité que deux maîtres prennent la parole en même temps. Si cela ne pose pas de problème sur le plan électrique grâce l'utilisation de collecteurs ouverts, il faut pouvoir détecter cet état de fait pour éviter la corruption des données transmises.

#### III.1. Principe :

Comme nous l'avons vu précédemment, pour prendre le contrôle du bus, un maître potentiel doit d'abord vérifier que celui-ci soit libre, et qu'une condition d'arrêt ai bien été envoyée depuis au moins 4,7µs. Mais il reste la possibilité que plusieurs maîtres prennent le contrôle du bus simultanément. Chaque circuit vérifie en permanence l'état des lignes SDA et SCL, y compris lorsqu'ils sont eux même en train d'envoyer des données. On distingue alors plusieurs cas :

- ① Les différents maîtres envoient les mêmes données au même moment :

*Les données ne sont pas corrompues, la transmission s'effectue normalement, comme si un seul maître avait parlé. Ce cas est rare.*



② Un maître impose un '0' sur le bus :

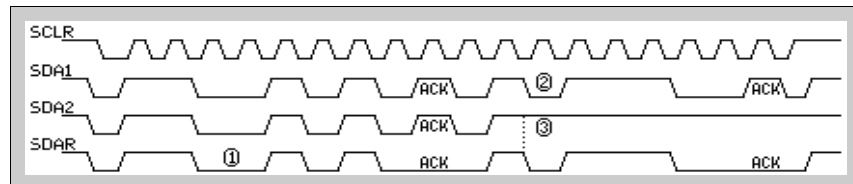
*Il relira forcément '0' et continuera à transmettre. Il ne peut pas alors détecter un éventuel conflit.*

③ Un maître cherche à appliquer un '1' sur le bus :

*Si il ne relit pas un niveau '1', c'est qu'un autre maître a pris la parole en même temps. Le premier perd alors immédiatement le contrôle du bus, pour ne pas perturber la transmission du second. Il continue néanmoins à lire les données au cas celles-ci lui auraient été destinées.*

### III.II. Exemple :

Soit le chronogramme suivant :



Le premier octet est transmis normalement car les deux maîtres imposent les mêmes données. (Cas 1). Le bit ACK est mis à '0' par l'esclave.

#### Dans cet exemple :

SCLR : Horloge résultante.

SDA1 : Niveaux de SDA imposés par le maître n°1.

SDA2 : Niveaux de SDA imposés par le maître n°2.

SDAR : Niveaux de SDA réels résultants lus par les deux maîtres.

Lors du deuxième octet, le maître n°2 cherche à imposer un '1' (SDA2) , mais relit un '0' (SDAR), il perd alors le contrôle du bus et devient esclave (Cas 3) . Il reprendra le contrôle du bus, lorsque celui-ci sera de nouveau libre.

Le maître n°1 ne voit pas le conflit et continue à transmettre

normalement. (Cas 2)

Au total, l'esclave à reçu les données du maître n°1 sans erreurs et le conflit est passé inaperçu.

#### IV. Nouvelles caractéristiques :

Afin de compenser quelques lacunes des premières spécifications du bus I<sup>2</sup>C ( qui datent de 1982 !), quelques nouvelles améliorations ont été apportées à partir de 1993.

- x Le mode rapide : Le bus a ainsi la capacité de transmettre des données jusqu'à une vitesse de 400 Kbit/s.
- x Des entrées à triggers de Schmitt : le but est de limiter la sensibilité au bruit.
- x La mise en haute impédance d'un circuit non alimenté : Ceci évite de bloquer le bus si un périphérique n'est pas alimenté.
- x Extension à 10 bits de l'adressage des circuits : l'adressage d'un circuits se fait maintenant sur 10 bits répartis dans deux octets d'adresse de la façon suivante:



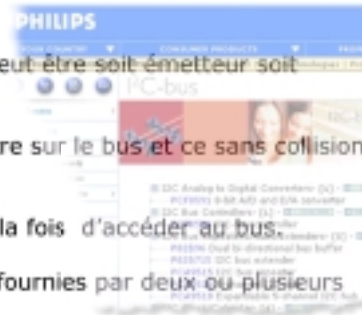
## V. Les adresses réservées :

adresse	fonction	Description
0000 0000	appel général	Après l'émission d'un appel général, les circuits ayant la capacité de traiter ce genre d'appel émettent un acquittement. Le deuxième octet permet définir le contenu de l'appel.
0000 0001	Réservé	Octet de START <sup>5</sup> (utilisé pour synchroniser les périphériques lents avec les périphériques rapides.)
0000 0110	RESET	Remet tout les registres de circuits connectés dans leur état initial ( Mise sous tension ). Les circuits qui le permettent rechargent leur adresse d'esclave.
0000 001x	Début d'adressage protocole CBus	L'émission de cet octet permet de rendre sourd tout les circuits I <sup>2</sup> C présent sur le bus. A partir de ce moment, on peut transmettre ce que l'on désire sur le bus. Le bus repasse en mode normal lors de la réception d'une condition d'arrêt.
0000 010x		Réservé pour d'autres protocoles.
0000 011x		Réservé pour un usage futur.
0000 1xxx		Passage en <b>High-speed mode</b> (Hs-mode) (⇒ 3.4 Mbit/s.)
1111 1xxx		Réservé pour un usage futur.
1111 0xxx		Utilisé en adressage 10bits.

5 Ne pas confondre avec la condition de START !

## VI. Terminologie du bus I<sup>2</sup>C

- **Émetteur** : le composant qui envoie des données sur le bus. Il peut être à l'origine de l'émission (émetteur – maître) ou bien répondre à une demande émanant d'un autre composant (émetteur -esclave).
- **Récepteur** : le composant qui reçoit les données présentes sur le bus.
- **Maître** : le composant qui initialise le transfert, génère le signal d'horloge et termine le transfert. **Un maître peut être soit émetteur soit récepteur.**
- **Esclave** : le composant adressé par le maître. **Un esclave peut être soit émetteur soit récepteur.**
- **Multimaître** : la possibilité de faire coexister plus d'un maître sur le bus et ce sans collision ni perte de données.
- **Arbitrage** : procédure qui ne permet qu'à un seul maître à la fois d'accéder au bus.
- **Synchronisation** : procédure qui synchronise les horloges fournies par deux ou plusieurs maîtres.
- **SDA** : ligne des signaux de données. (Serial DAta)
- **SCL** : ligne des signaux d'horloge. (Serial CLock)



## VII. Terminologie d'un transfert sur le bus I<sup>2</sup>C

- **F (FREE) libre** : le bus est libre ou au repos; la ligne de données SDA et la ligne d'horloge SCL sont à l'état HAUT.
- **S (START) départ** or **SR (Repeated START)** : le transfert de données débute par une condition START. La ligne de données passe à l'état BAS tandis que la ligne d'horloge reste à l'état HAUT: le bus est OCCUPE.
- **C (CHANGE) changement** : pendant que la ligne d'horloge SCL est au niveau BAS, le bit de donnée à transmettre peut être appliqué à la ligne de données SDA par un émetteur tandis que la ligne d'horloge reste à l'état HAUT.
- **D (DATA) donnée** : un bit d'information (HAUT ou BAS sur la ligne de données SDA) est considéré comme valide pendant la durée à l'état HAUT de la ligne d'horloge SCL. L'état présent sur la ligne de donnée doit rester stable tout le temps ou la ligne d'horloge SCL est à l'état HAUT pour ne pas être interprété à tort comme condition de START ou de STOP.
- **P (STOP) arrêt** : un transfert de données se conclut par une condition d'arrêt STOP. Ceci se produit lorsque le niveau sur la ligne de données SDA passe de l'état BAS à l'état HAUT tandis que la ligne d'horloge SCL reste à l'état HAUT. Le bus redevient LIBRE.

## VIII. Bibliographie et liens

<http://www.semiconductors.philips.com>

<http://membres.lycos.fr/webelec/index.html>

<http://f6csx.free.fr>