



Le bus I2C : synthèse et spécifications

1. Introduction

Une grande variété de protocole de communication entre périphériques.

Transmission série des données sur 2 fils.

Parmi les différents protocoles disponibles : le bus I2C.

- Origine de la spécification du bus I2C : Philipps.
- Objectif : disposer d'un moyen simple pour faire communiquer des périphériques divers (le plus souvent sous la forme d'un circuit intégré (IC), en utilisant un minimum de broches.
- Un ensemble de spécifications matérielles et logicielles simples, pour construire un bus universel, en assurant la compatibilité avec des périphériques de fabricants d'horizon.
- Pas de connecteur spécifique : le plus souvent, le bus est constitué de 2 pistes sur un circuit imprimé.
- Constitue désormais un standard reconnu par tous les fabricants majeurs de circuits intégrés.
- I2C protocol : Inter IC protocol.

Etat des spécifications :

- 1992 : V1.0
- 1998 : V2.0;
- 2001 : V2.1;
- Document de référence : « *The I2C bus specification, version 2.1*, janvier 2000 (number 9398-393-40011) », Phillips.

Domaines d'application :

Récepteurs infrarouges (télécommande)

Capteurs de température

Circuits audio (égaliseur, contrôle de volume, etc.)

Drivers d'affichage LCD ou à LEDS

Chargeurs de batterie

PLL pour tuner HF,...

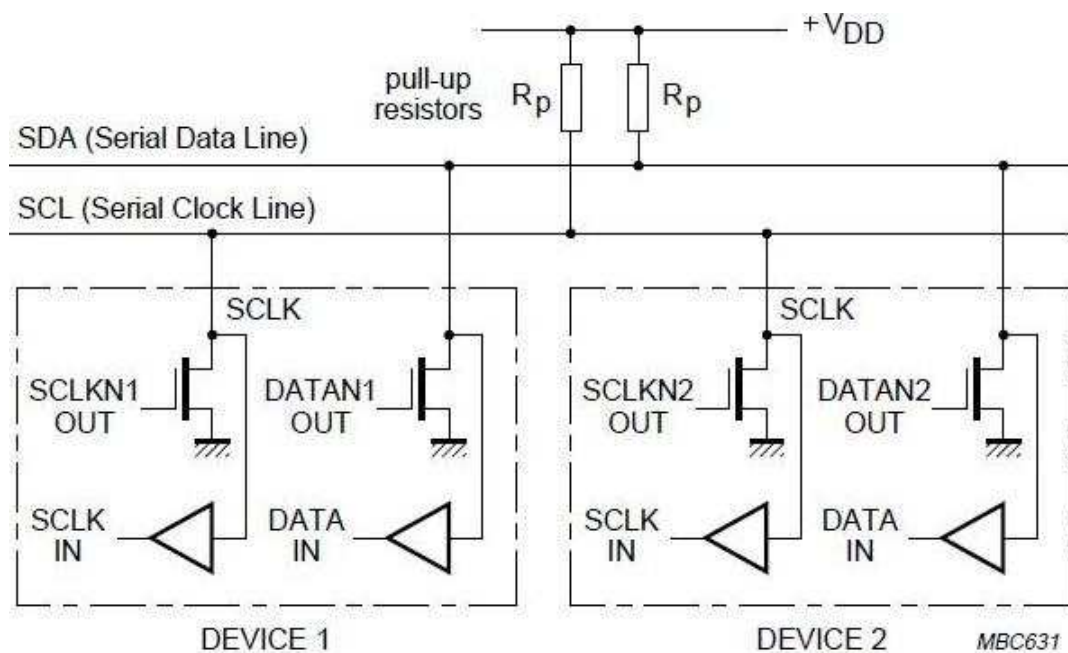
2. Concepts de base

- Utiliser un bus 2 fils pour échanger des données entre des périphériques connectés sur le bus.
 - o Une ligne de transmission des données : SDA (Serial DATA line)
 - o Une ligne de synchronisation des échanges : SCL (Serial CLOCK)
- Chaque périphérique possède une adresse unique
- Bus de type Maître – Esclave
- Protocole série, bi directionnel, 8 bits de données
 - o Débit :

Débit (kbits / s)	Mode
100	Standard
400	Fast mode
3400	High speed mode

- Longueur maximum du bus : de l'ordre du mètre.

- Terminologie
 - Transmetteur : périphérique qui envoie des données sur le bus ;
 - Récepteur : périphérique qui reçoit des données depuis le bus ;
 - Maître :
 - Périphérique à l'initiative du transfert de données ;
 - Périphérique qui génère le signal d'horloge ;
 - Périphérique qui met un terme à l'échange des données
 - Esclave : périphérique adressé par le Maître ;
 - Multi maître : possibilité d'avoir plusieurs périphérique maîtres su le bus ;
 - Arbitrage : procédure de prise de contrôle du bus entre plusieurs Maîtres ;
 - Synchronisation : synchronisation des horloges entre deux périphériques.
- Connexion d'un périphérique I2C sur le bus



R_p : résistance de tirage, de l'ordre quelques $k\Omega$.

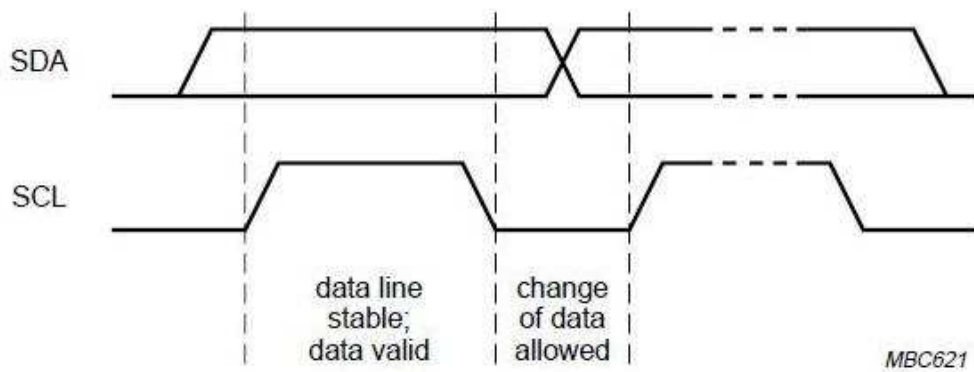
3. Transfert d'un bit

3.1. Caractéristique générales

- SDA et SCL : lignes bi directionnelles ;
- Etat du bus : considéré libre si SDA et SCL sont à l'état haut (« 1 », HIGH).

3.2. Validité des données transmises

La donnée (le bit transmis) sur la ligne SDA doit être stable lorsque SCL est à l'état haut.
Les transitions de niveau sur la ligne SDA ne peuvent se produire que lorsque SCL est à l'état bas (« 0 », LOW).



3.3. Conditions de départ et d'arrêt (START and STOP conditions)

START (S) et STOP (P) conditions sont toujours générées à l'initiative d'un périphérique Maître ;

Ligne SCL : toujours à l'état haut ;

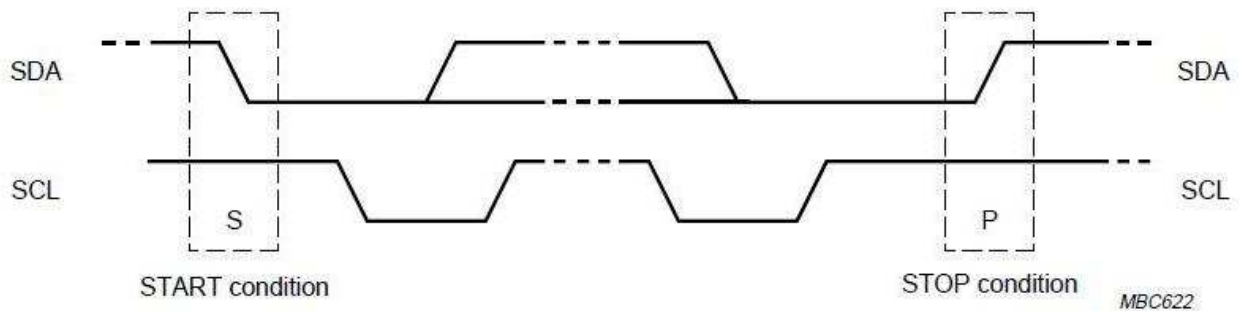
Ligne SDA :

START condition : transition de HIGH vers LOW

STOP condition : transition de LOW vers HIGH

Le bus est considéré libre après une STOP condition (avec un délai)

Le bus est considéré comme occupé après une START condition



4. Transfert de données

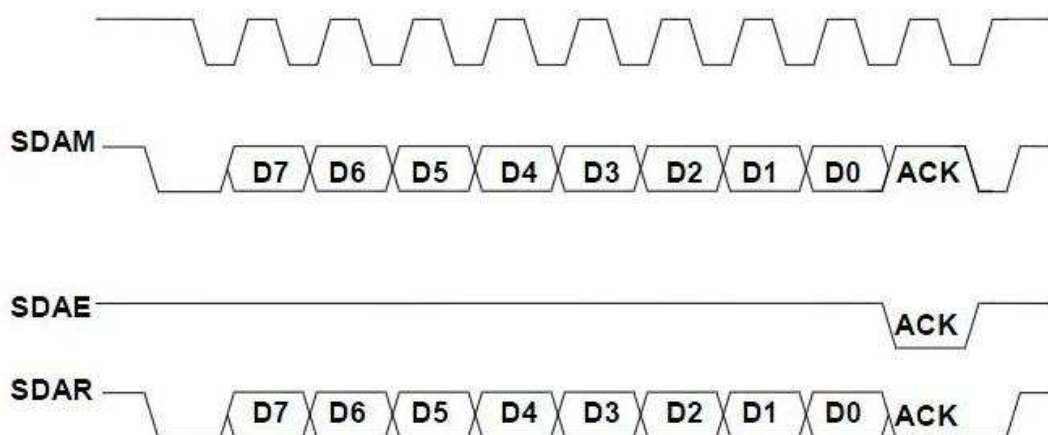
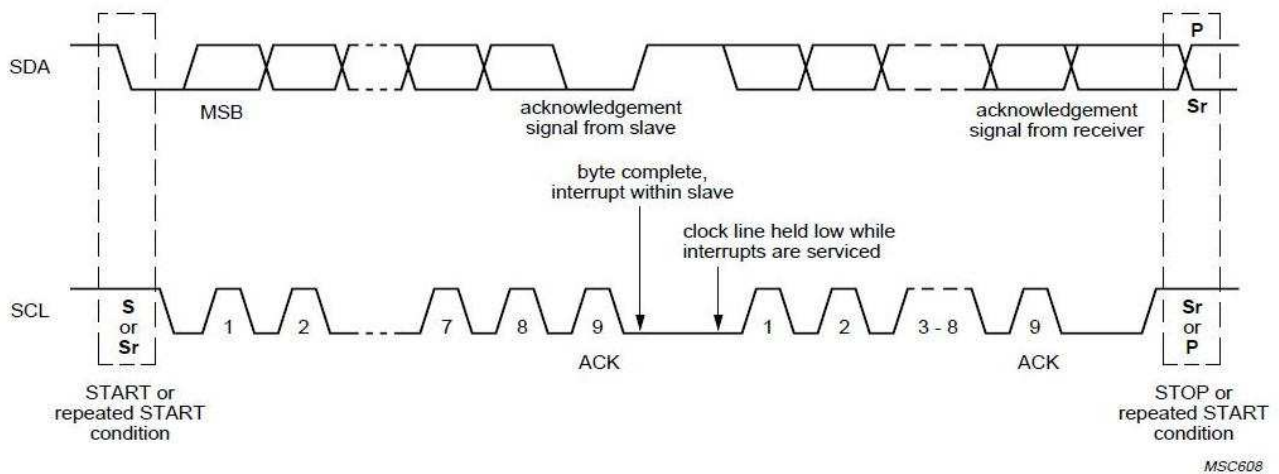
Transmission d'octets uniquement ;

Nombre quelconque d'octets transmis ;

La transmission d'un octet est toujours suivie d'un acquittement (Acknowledgement) ;

Pour chaque octet transmis, le MSB (Most Significant Bit) est transmis en premier, jusqu'au LSB (Last Significant Bit).

Acquittement : sur le 9^{ième} top horloge. Le **récepteur** maintient la ligne SDA à l'état haut (HIGH).



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

Remarque : cas de l'adressage d'un périphérique (envoi d'une adresse)

- Si **pas** acquittement de l'adresse reçue par un périphérique esclave, celui-ci maintient la ligne SDA à l'état **haut**.

5. Format d'adressage des périphériques I2C

5.1. Adressage sur 7 bits

Chaque périphérique possède une adresse unique définie sur 7 bits.

Le bit de poids faible détermine le type d'opération réalisée (lecture ou écriture)

D7 (MSB)	D6	D5	D4	D3	D2	D1	D0 (LSB)
							R/\bar{W}

$R/\bar{W} = 0$: opération d'écriture (valeur d'adresse paire)

$R/\bar{W} = 1$: opération de lecture (valeur d'adresse impaire)

5.2. Adresses réservées

Les adresses 0000 0xxx et 1111 11xx sont réservées à des modes de fonctionnement particuliers (cf. tableau ci-dessous).

Adresse	R/ \overline{W}	Commentaire
0000 000	0	Adresse générale d'appel
0000 000	1	Octet de « START » ⁽¹⁾
0000 001	X	Adresse CBus
0000 010	X	Réservé (pour d'autres types de bus)
0000 011	X	Réservé pour des usages ultérieurs
0000 1XX	X	HS mode code maître
1111 1XX	X	Réservé pour des usages ultérieurs
1111 0XX	X	Adressage sur 10 bits de périphériques esclave

(1) aucun périphérique n'est autorisé à effectuer un acquittement à la réception de l'octet de START.

Adresse générale d'appel :

- Permet d'effectuer une remise à jour des périphériques esclaves.
- Format :

0	0	0	0	0	0	0	0	A	0	0	0	0	0	1	1	0	A
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Reset et reprogrammation matérielle de l'adresse du périphérique esclave

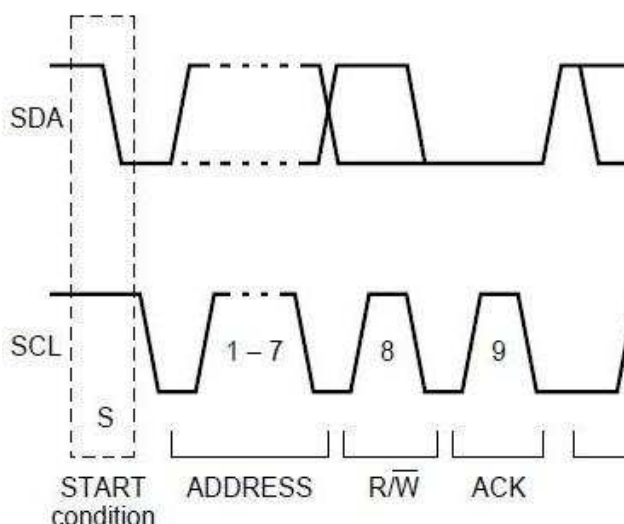
ou

0	0	0	0	0	0	0	0	A	0	0	0	0	0	1	0	0	A
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Reprogrammation matérielle de l'adresse du périphérique esclave

5.3. Transmission d'une adresse

Le diagramme ci-dessous résume l'envoi d'une adresse sur le bus par un périphérique Maître.



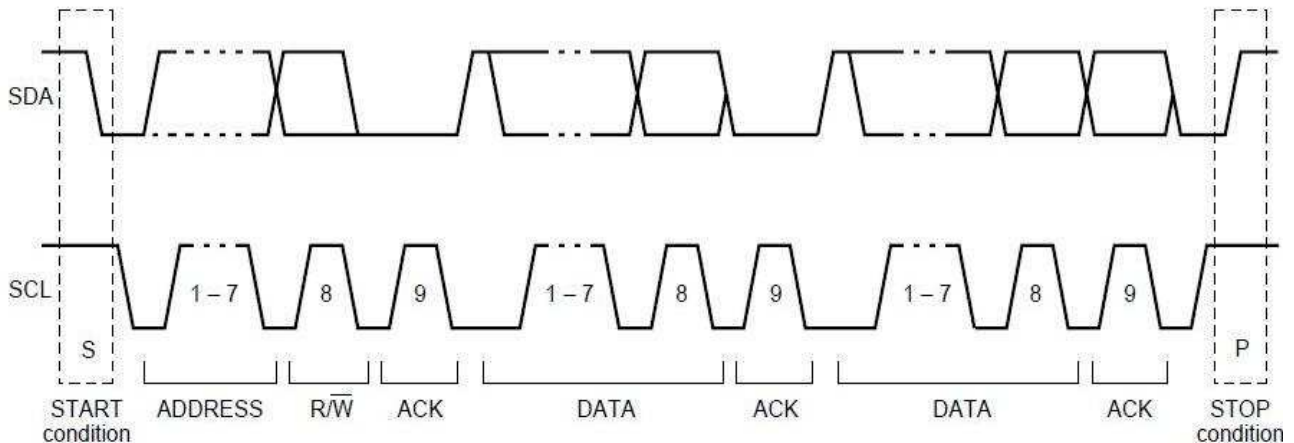
6. Transmission de données

Le schéma général de transmission de données entre des périphériques I2C est le suivant :

1. Envoi d'une condition de départ « START condition »
2. Envoi de l'adresse du périphérique esclave (7 bits) et du bit de Lecture / Ecriture R/\overline{W}
Si $R/\overline{W} = 1 \Rightarrow$ demande de donnée (Lecture)

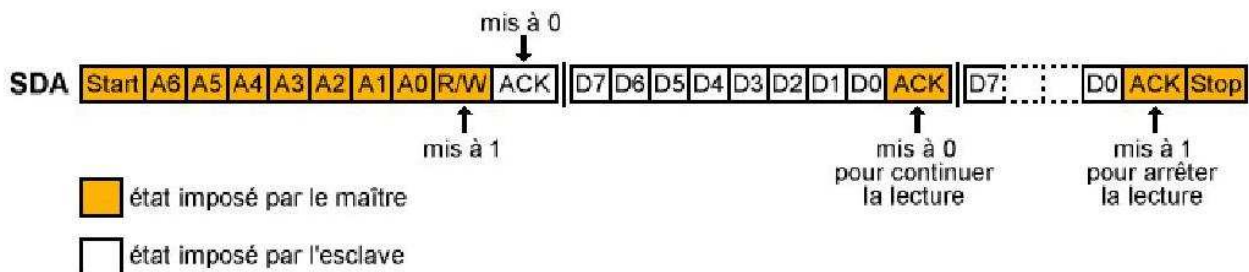
Si $R/\overline{W} = 0 \Rightarrow$ écriture de données (Ecriture)

3. Cycles de lecture ou écriture de données
4. Fin du transfert par une condition d'arrêt (STOP condition) générée par le périphérique Maître.



6.1. Lecture de données depuis un périphérique Maître

- Périphérique Maître envoie adresse
- Attente acquittement adresse depuis le périphérique esclave
- Lecture : envoi données par périphérique esclave puis acquittement des données par périphérique Maître
- Fin du cycle par mise à l'état haut de SDA du Maître



6.2. Ecriture de données

Voir schéma général ci-dessous :

