

Le BUS I²C



Créé par PHILIPS (début années 80)

Objectif :

Relier de manière économique des composants numériques (Circuits intégrés) dans un téléviseur, un magnétoscope, PC,

Utilise seulement 2 fils + masse :

SDA = Serial Data

SCL = Serial Clock



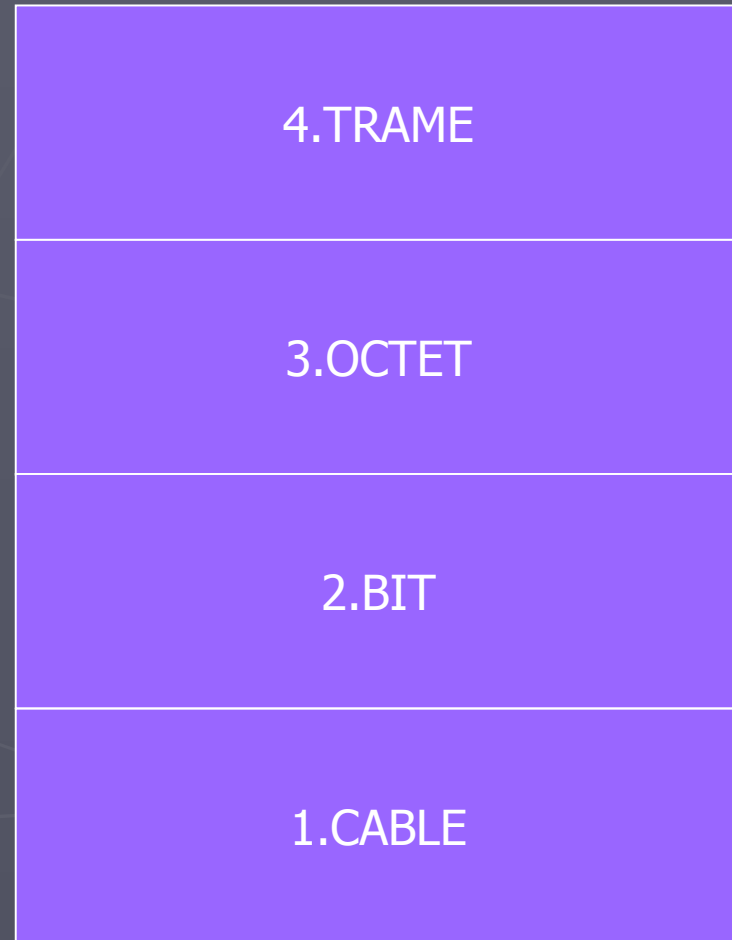
Economique

Analyse du BUS I2C :

4 niveaux

Le BUS I2C fonctionne
comme un petit réseau
local

Performant

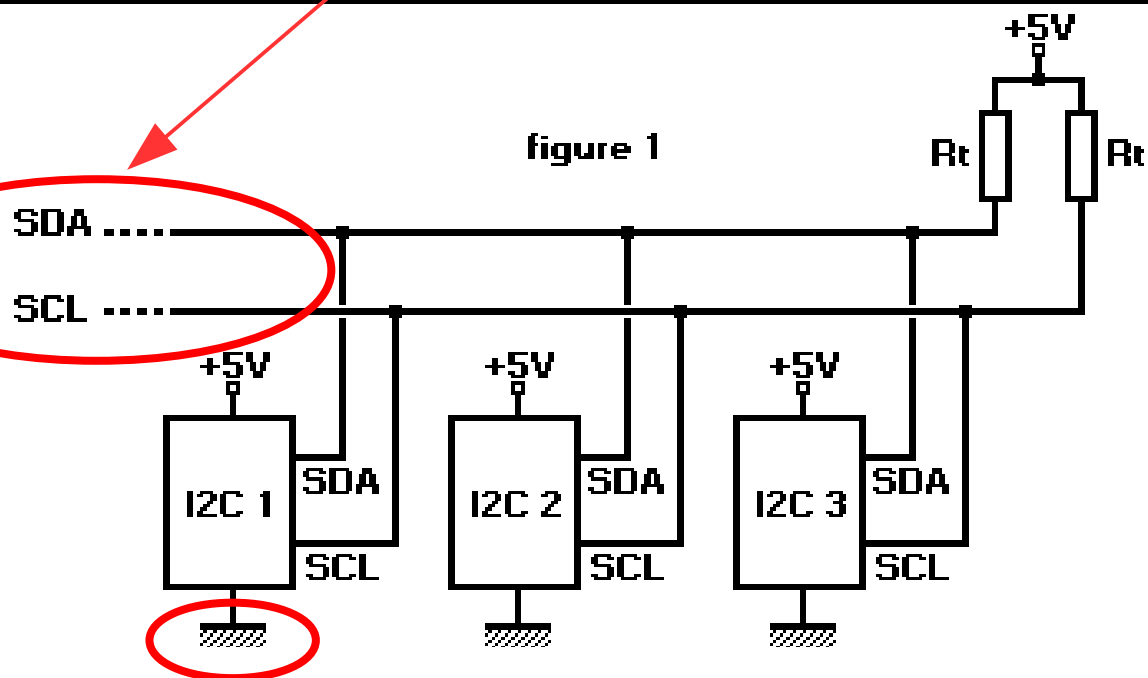


Niveau 1 : CABLE



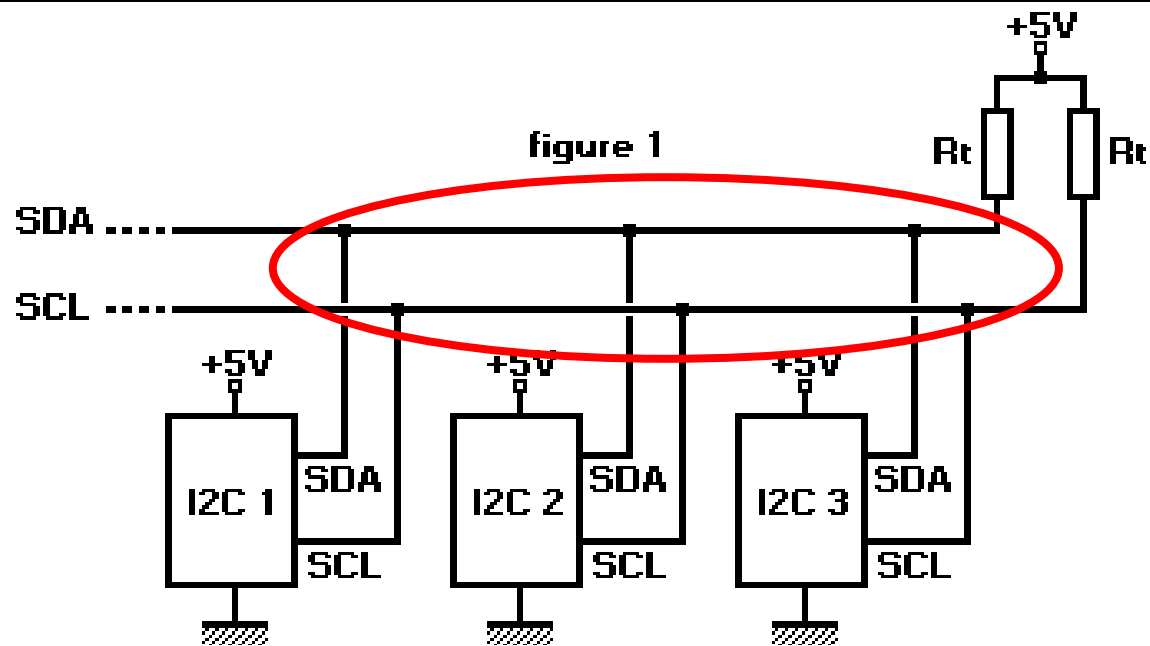
CABLE

figure 1



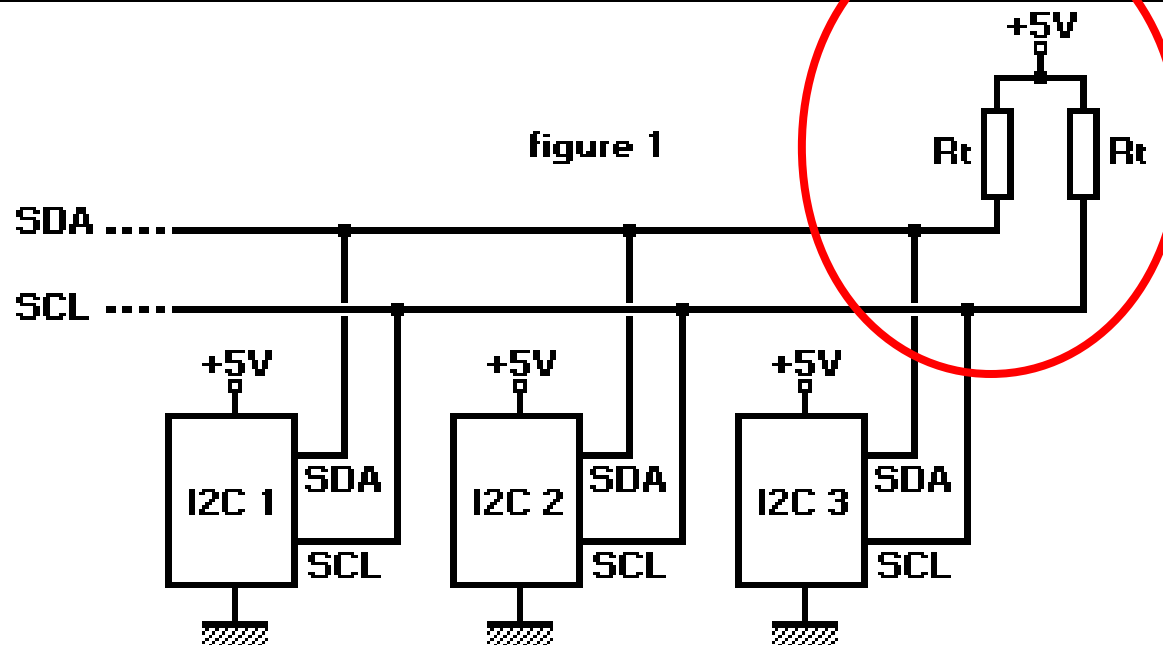
2 fils + masse

CABLE



Tous les composants sont câblés en
parallèle

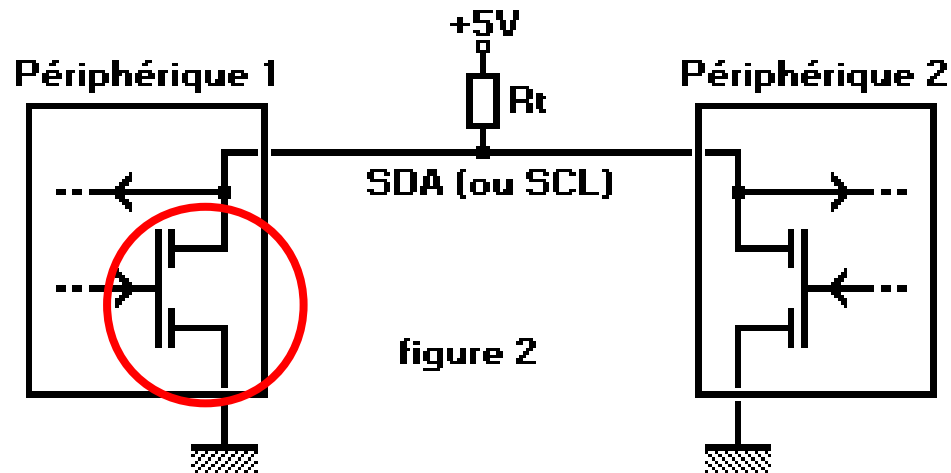
CABLE



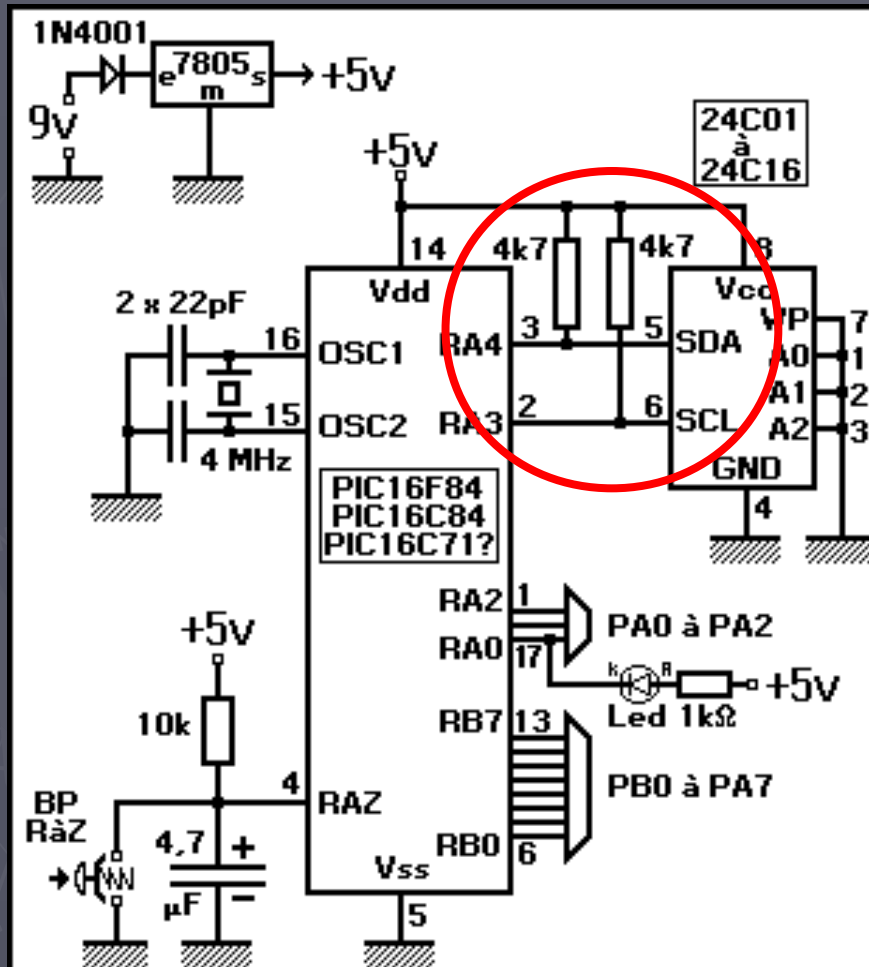
Les niveaux logiques **hauts** sont réalisés par des résistances de pull-up (tirage vers VDD)

CABLE

Les niveaux logiques **bas** sont réalisés par des transistors (ici MOS) en «drain ouvert » ou des transistors bipolaires en « collecteur ouvert »



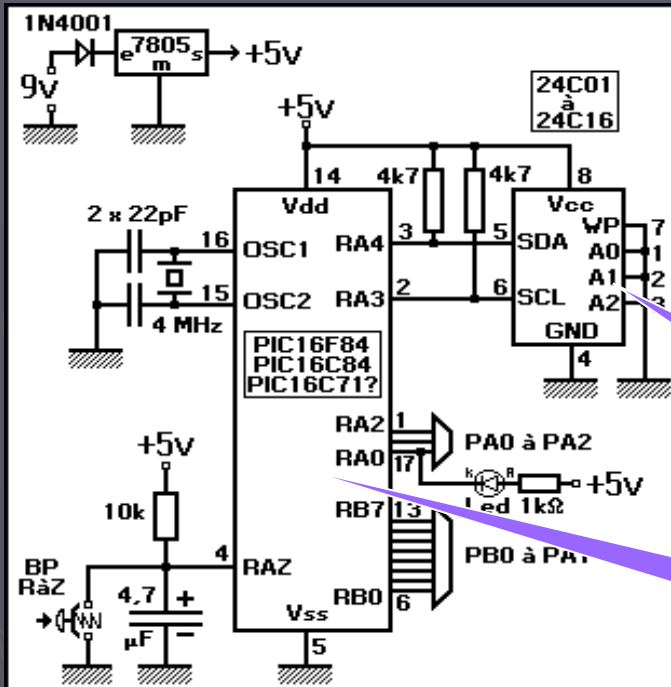
CABLE



Exemple de câblage :
Un micro-contrôleur PIC
associé à une mémoire
EEPROM

Exemple de Fonctionnement

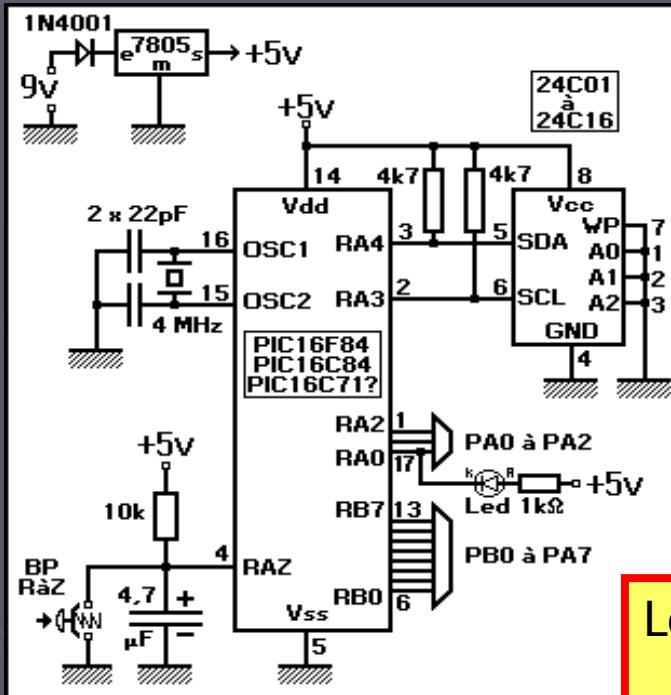




Exemple : Le PIC veut lire le contenu d'une case dans la Mémoire

Mémoire

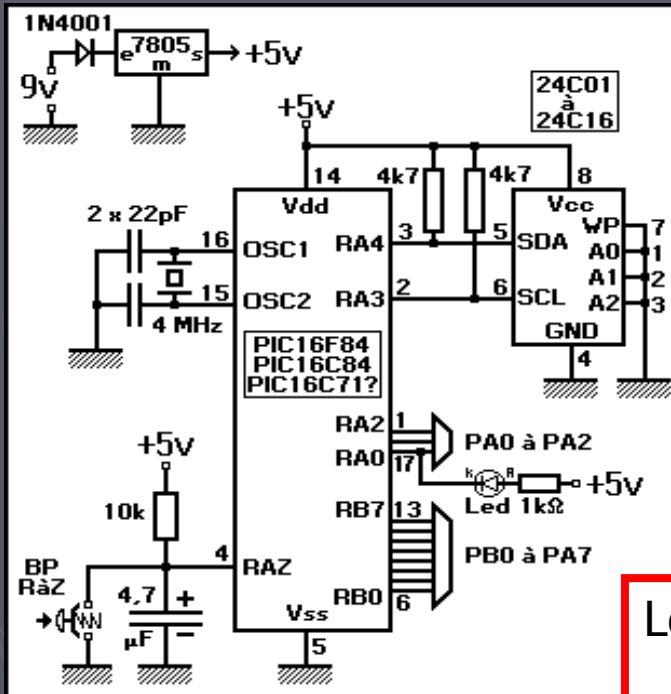
PIC



Exemple : Le PIC veut lire le contenu d'une case dans la Mémoire

Le PIC doit :

- prendre le contrôle du BUS,
- envoyer l'adresse de l'octet qu'il veut lire,
- analyser les réactions de la Mémoire
- récupérer l'octet envoyé par la Mémoire
- libérer le BUS



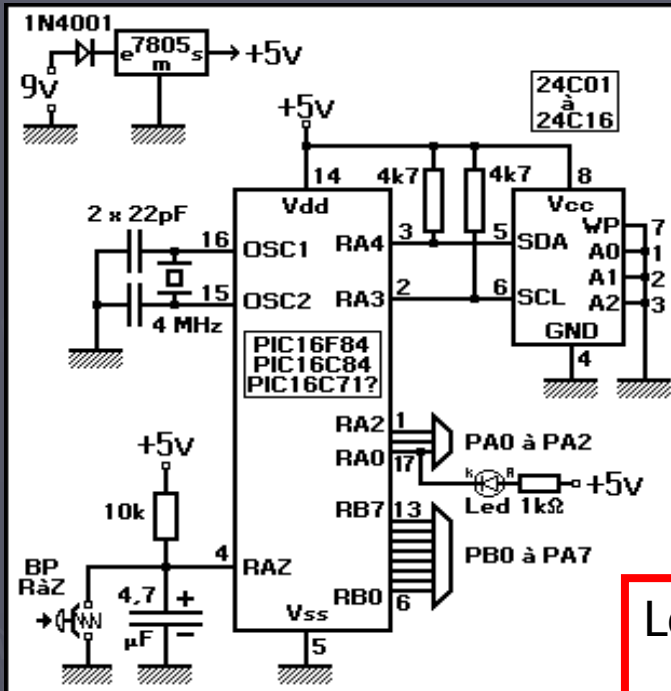
Exemple : Le PIC veut lire le contenu d'une case dans la Mémoire

Le PIC doit :

- prendre le contrôle du BUS,
- envoyer l'adresse de l'octet qu'il veut lire,
- analyser les réactions de la Mémoire

L'ensemble constituera 1 TRAME, comportant 4 OCTETS, auxquels s'ajoutent divers signaux et l'ensemble circulant sur les 2 FILS

octet envoyé par la Mémoire
S



Exemple : Le PIC
lire le contenu d'
case dans la Mém

4. TRAME

3. OCTET

2. BIT

1. CABLE

Le PIC doit :

- prendre le contrôle du BUS,
- envoyer l'adresse de l'octet qu'il veut lire,
- analyser les réactions de la Mémoire

L'ensemble constituera 1 TRAME,
comportant 4 OCTETS, auxquels
s'ajoutent divers signaux, l'ensemble
circulant sur les 2 FILS

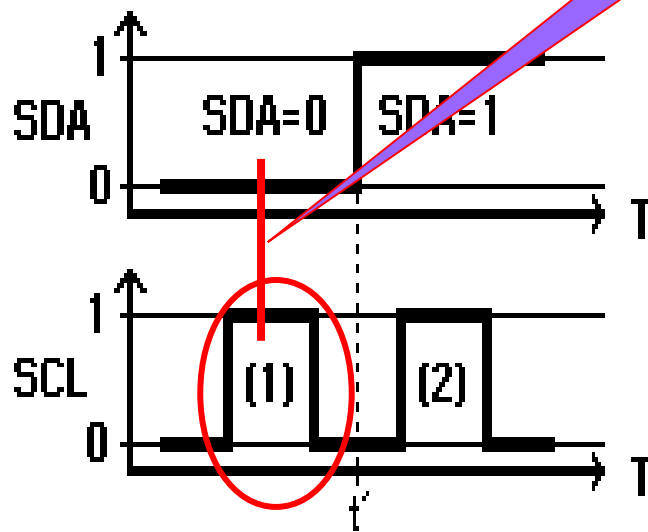
l'octet envoyé par la Mémoire
BUS

Niveau 2 : BITS



BIT

BIT « 0 »



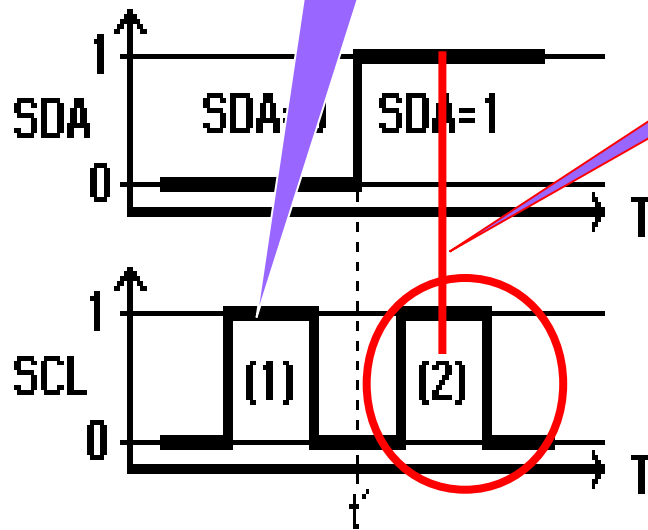
Impulsion sur SCL pendant que
SDA est à l'état **BAS**

figure 3

BIT

BIT « 0 »

BIT « 1 »



Impulsion sur SCL pendant que
SDA est à l'état **HAUT**

figure 3

BIT

BIT « 0 »

BIT « 1 »

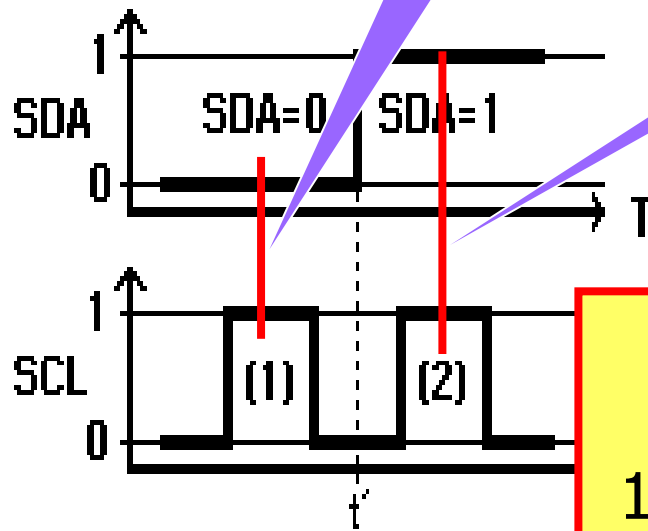


figure 3

1 BIT -->

1 NIVEAU logique de SDA + 1 impulsion de SCL

BIT

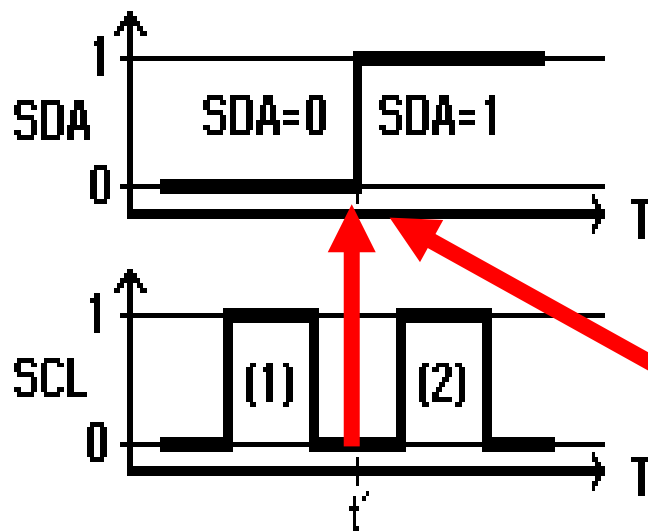


figure 3

REGLE:

SDA ne peut changer de niveau
que lorsque SCL est à l'état **BAS**

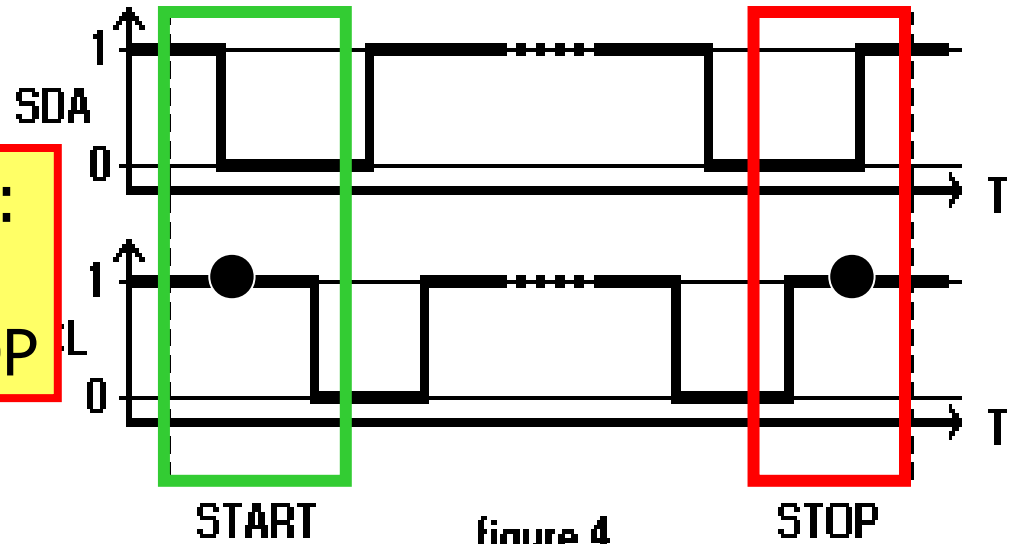
BIT

START

STOP

Deux exceptions à la règle :

Les condition START et STOP

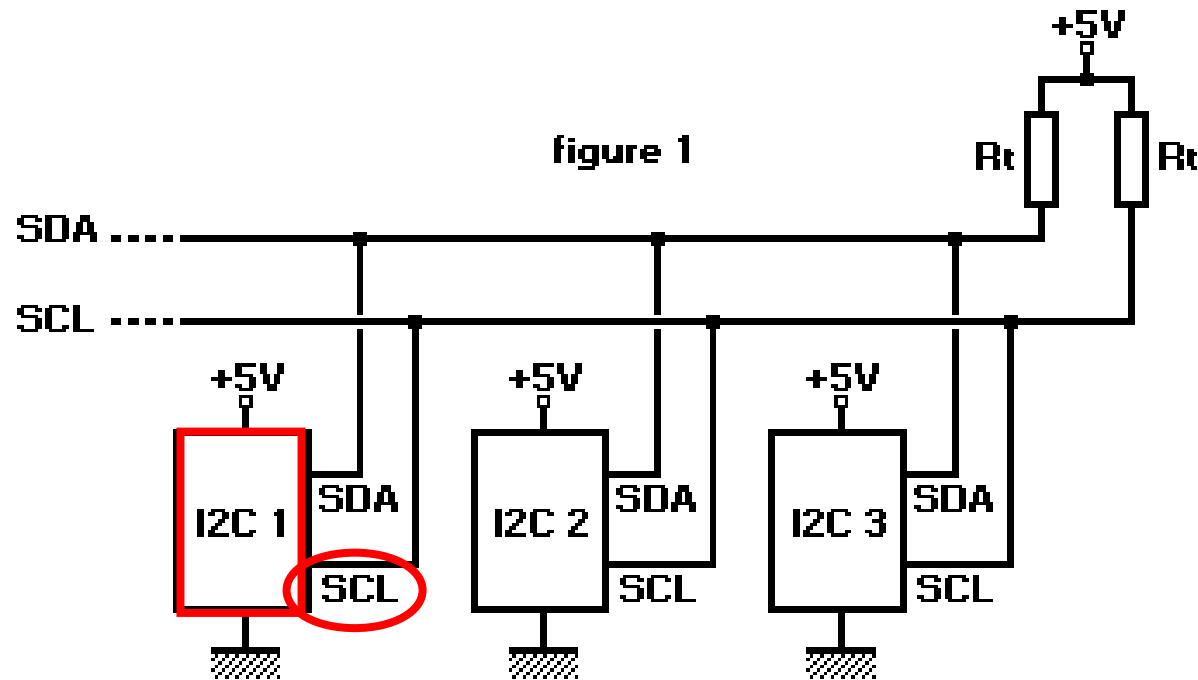


Seuls cas où SDA peut changer alors que SCL est à l'état haut

NIVEAU 4 : La TRAME



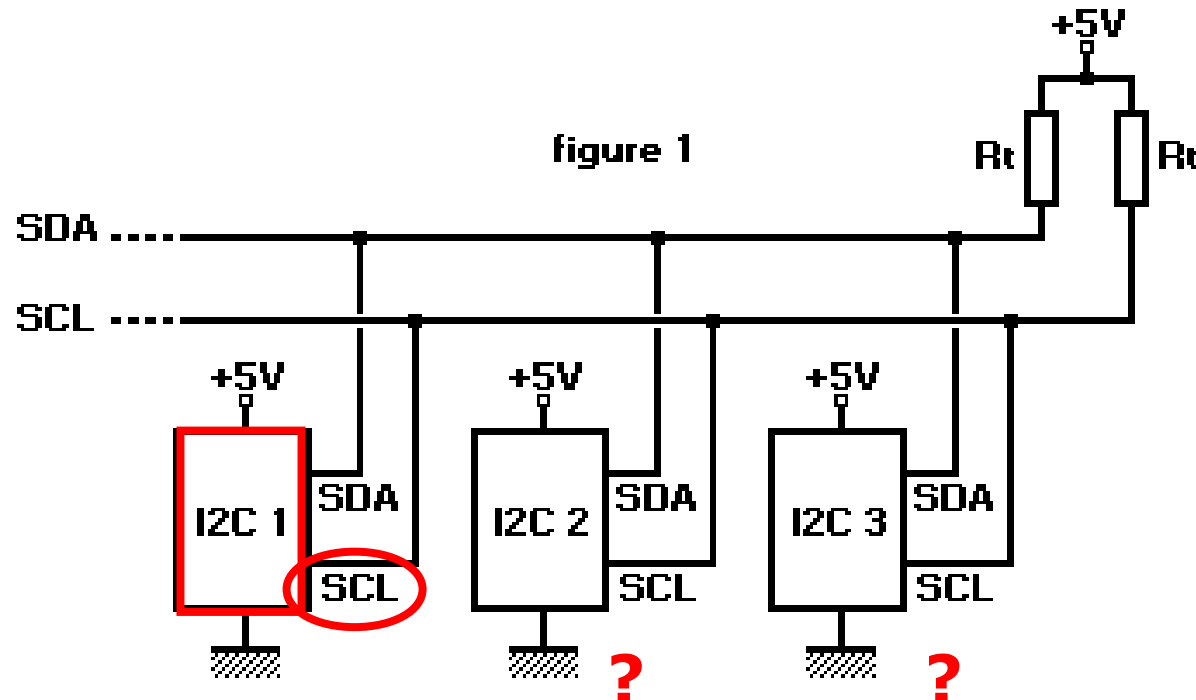
TRAME



Le composant I2C qui a un moment donné prend le contrôle de SCL(par un START) est appelé **MAITRE**

Les autres fonctionnent alors en **ESCLAVES** ...
..... jusqu'au prochain STOP

TRAME



Chaque ESCLAVE est identifié par une ADRESSE

Grâce à l'ADRESSE le MAITRE désigne son interlocuteur.

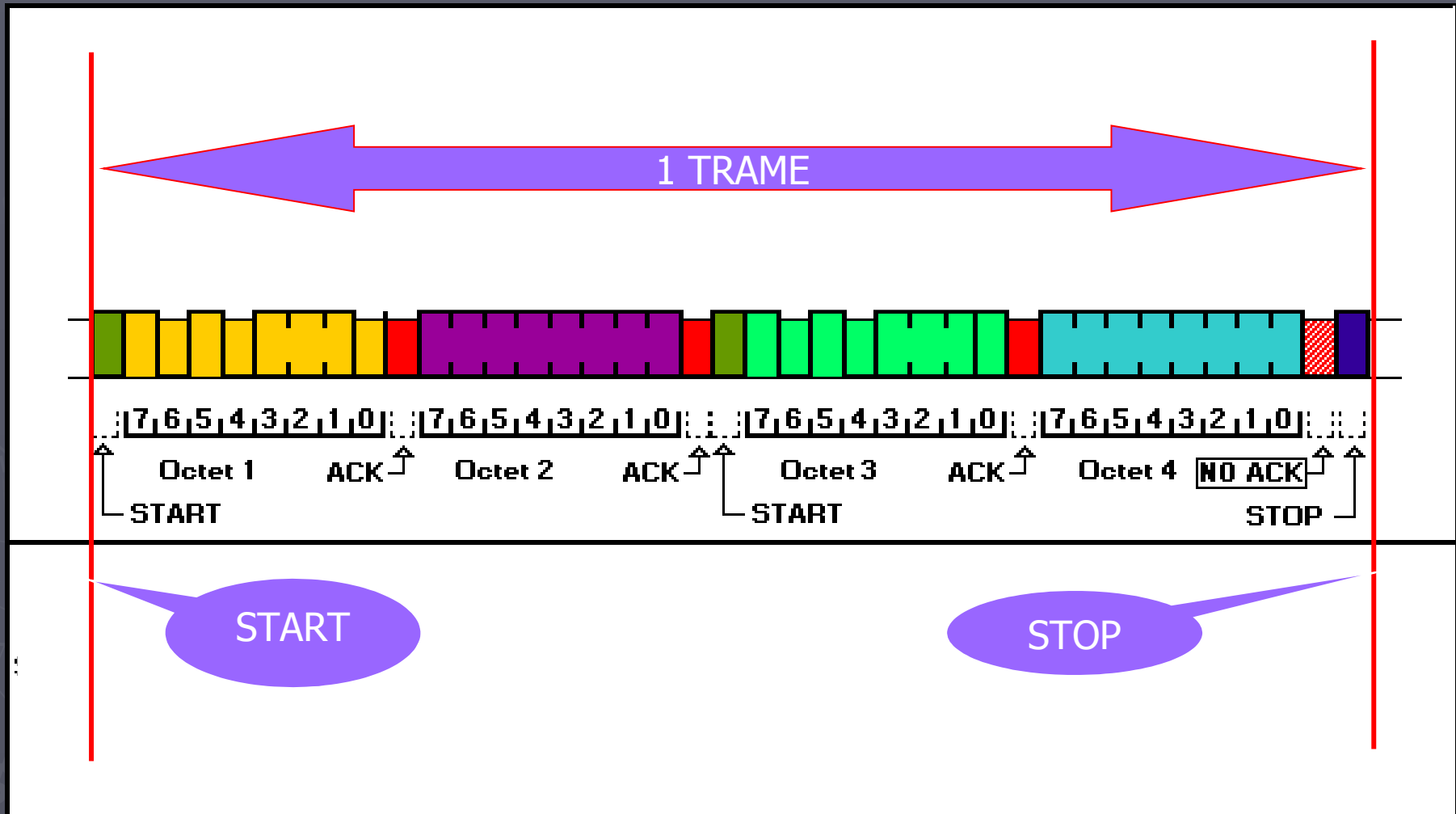
Seul l'ESCLAVE concerné réagit.

TRAME

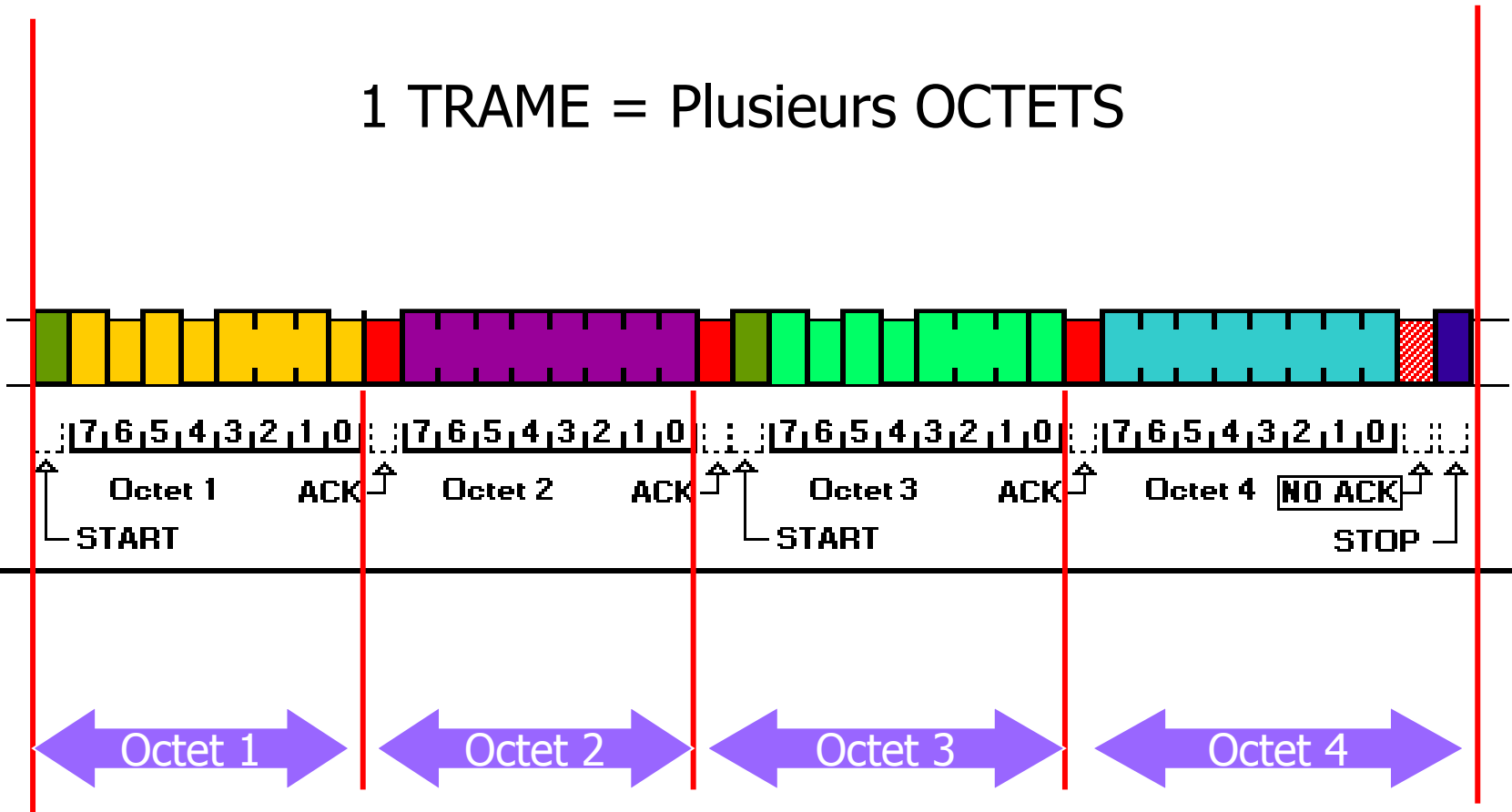
Ne pas confondre MAITRE /ESCLAVE
avec LECTURE / ECRITURE !

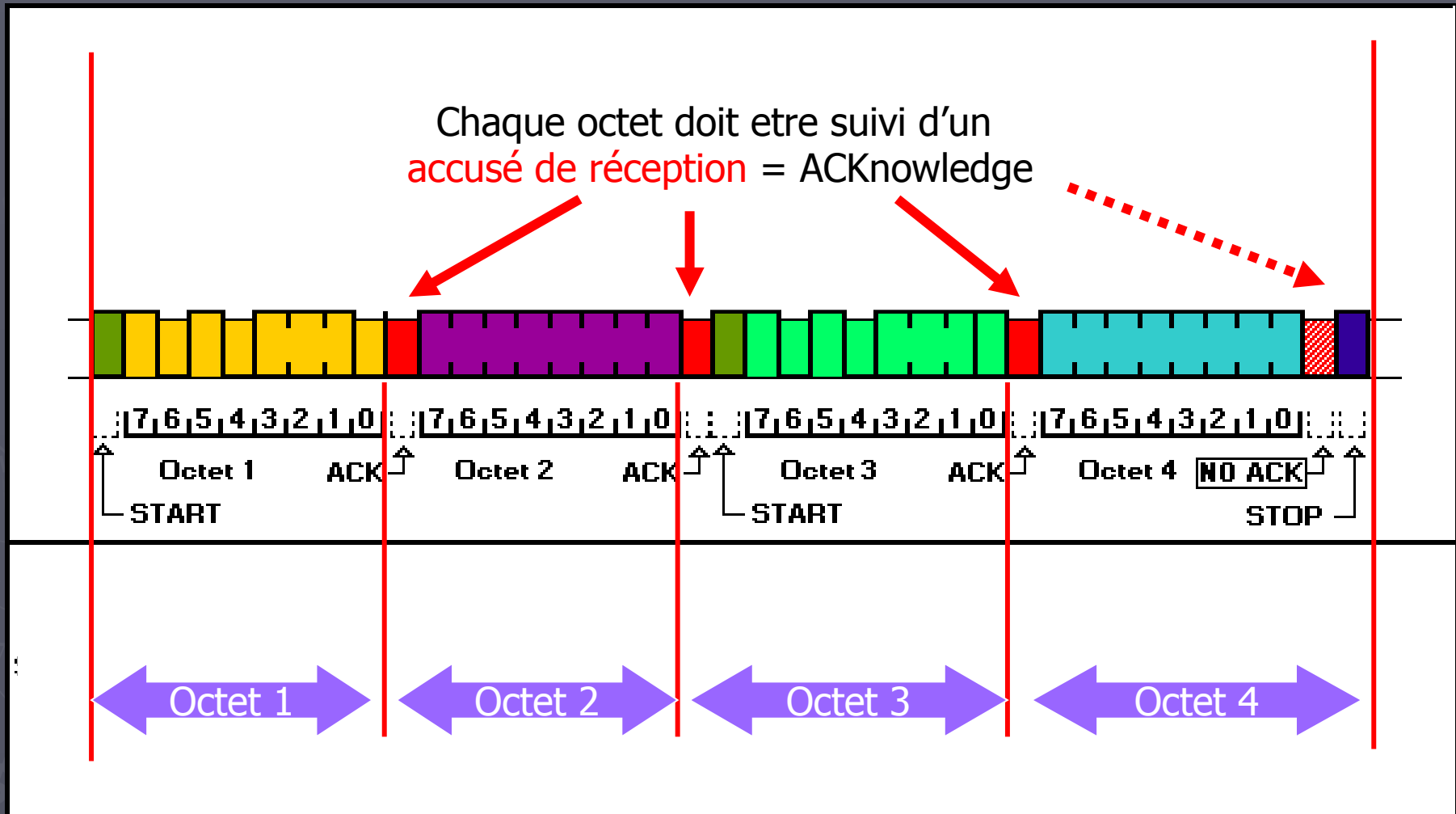
- ▶ Le MAITRE peut LIRE ou ECRIRE
- ▶ L'ESCLAVE peut LIRE ou ECRIRE

TRAME



1 TRAME = Plusieurs OCTETS

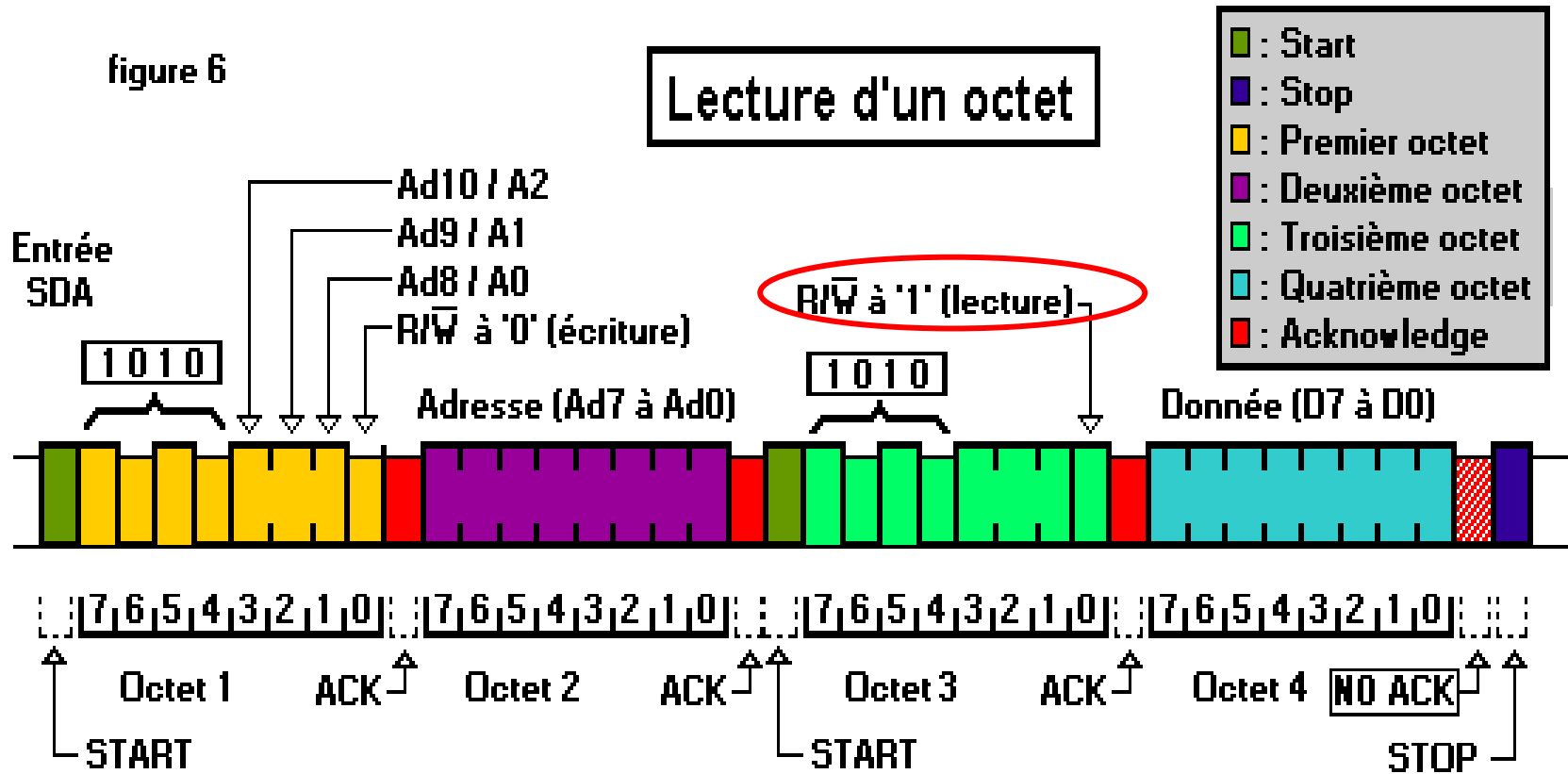




Remarque : Le second START qui précède le 3ème octet est un cas particulier

figure 6

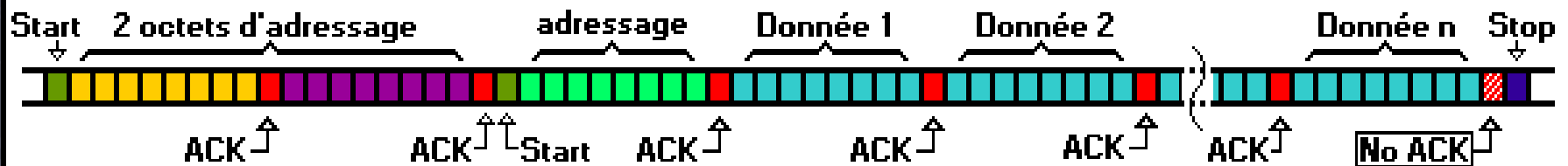
Lecture d'un octet

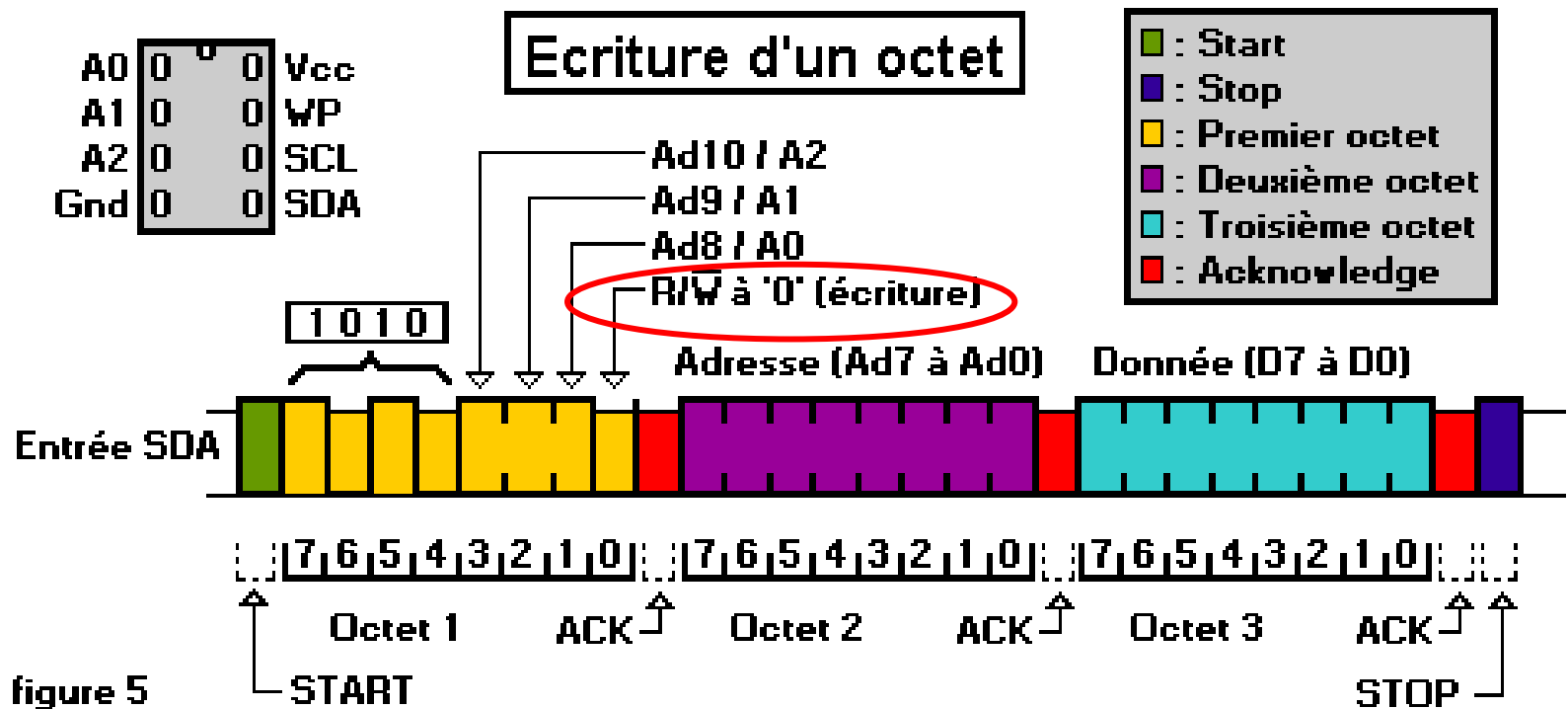


La même TRAME peut permettre la lecture de plusieurs OCTETS
successifs dans la Mémoire

figure 6 bis

Lecture de plusieurs octets





Trame correspondant à une écriture dans la mémoire

Généralisation

- ▶ Nous avons vu ici la gestion d'une EEPROM grâce au bus I2C
- ▶ La simplicité matérielle du bus I2C et ses performances ont donné naissance à un grand nombre de périphériques I2C
- ▶ C'est devenu un standard dans le domaine de la liaison des composants sur une carte imprimée.

FIN

Bon I2C !

