

LPO HENRI BRISSON

# BT : Bus SPI

---

Bus de Terrain

**M. BURNET**

**18/09/2017**

<b>I. PRESENTATION DU BUS I<sup>2</sup>C.....</b>	<b>3</b>
I. 1. ORIGINE .....	3
I. 2. LA PHILOSOPHIE DU BUS I <sup>2</sup> C .....	3
I. 3. LES CHAMPS D'APPLICATION DU BUS I <sup>2</sup> C .....	3
<b>II. PROPRIETES PHYSIQUES DU BUS I<sup>2</sup>C.....</b>	<b>4</b>
II. 1. LE SUPPORT PHYSIQUE.....	4
II. 2. SYNOPTIQUE D'UNE LIAISON SPI MAITRE-ESCLAVE .....	5
II. 3. FORMAT DE TRANSFERT DE DONNEES .....	5
II. 4. SYNOPTIQUE D'UNE LIAISON SPI MAITRE-MULTI-ESCLAVES .....	7
II. 5. BUS SPI ET NIVEAUX ELECTRIQUES.....	8
<b>III. LES DIFFERENTS AUTRES NOMS UTILISES AVEC UN BUS SPI.....</b>	<b>8</b>
III. 1. SCK.....	8
III. 2. SDI, DI, SI, SDO, DO, SO.....	8
III. 3. NCS, CS ,NSS, STE .....	8
III. 4. QSPI QUEUED SERIAL PERIPHERAL INTERFACE .....	8
III. 5. AVANTAGES ET INCONVENIENTS DU BUS SPI .....	8
<b>IV. CONSTITUTION D'UN SYSTEME MOTOROLA.....</b>	<b>9</b>
IV. 1. LE DIAGRAMME BLOC SPI.....	9
IV. 2. LE REGISTRE SPSR : SERIAL PERIPHERAL STATUS REGISTER.....	10
IV. 3. LE REGISTRE SPDR : SERIAL PERIPHERAL DATA REGISTER .....	10
IV. 4. LE SYSTEME DE DETECTION D'ERREUR SPI.....	10
<b>V. CONSTITUTION POUR UN SYSTEME A MICROCONTROLEUR PIC (MICROCHIP) .....</b>	<b>10</b>
V. 1. LE MSSP CONTROL REGISTER 1 (SSPCON1).....	10
V. 2. LE MSSP STATUS REGISTER (SSPSTAT) .....	11
V. 3. LE SERIAL RECEIVE/TRANSMIT BUFFER REGISTER SSPBUF .....	12
V. 4. LE MSSP SHIFT REGISTER.....	12
<b>VI. COMPLEMENT 'SD CARD'.....</b>	<b>12</b>

## I. Présentation du bus I<sup>2</sup>C

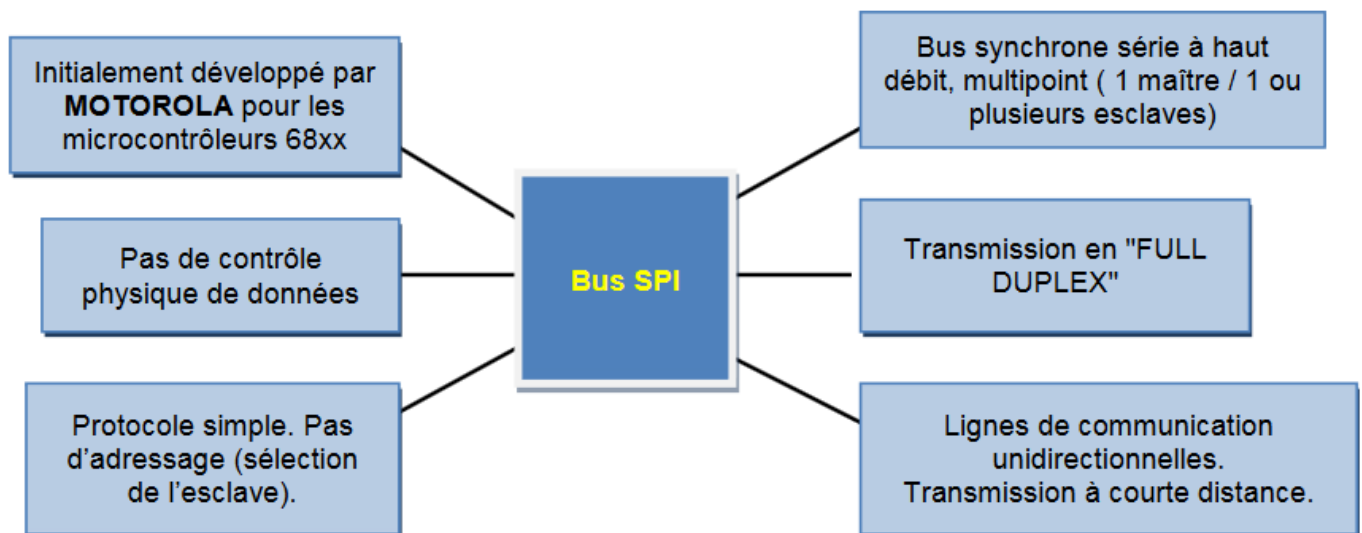
### I. 1. Origine

La sophistication et la complexité des appareils grand public ne permet plus l'échange entre composants électroniques par liaisons parallèles, ces liaisons devenaient de plus en plus nombreuses et encombrantes. Dans ces systèmes à base de microcontrôleurs (téléphones mobiles...), les échanges de données entre les composants se font maintenant avec des bus série.

Dans le but de minimiser ces liaisons, et par conséquent d'en augmenter la fiabilité, a été créé le bus série SPI (Synchronous Peripheral Interface), initialement développé par Motorola.

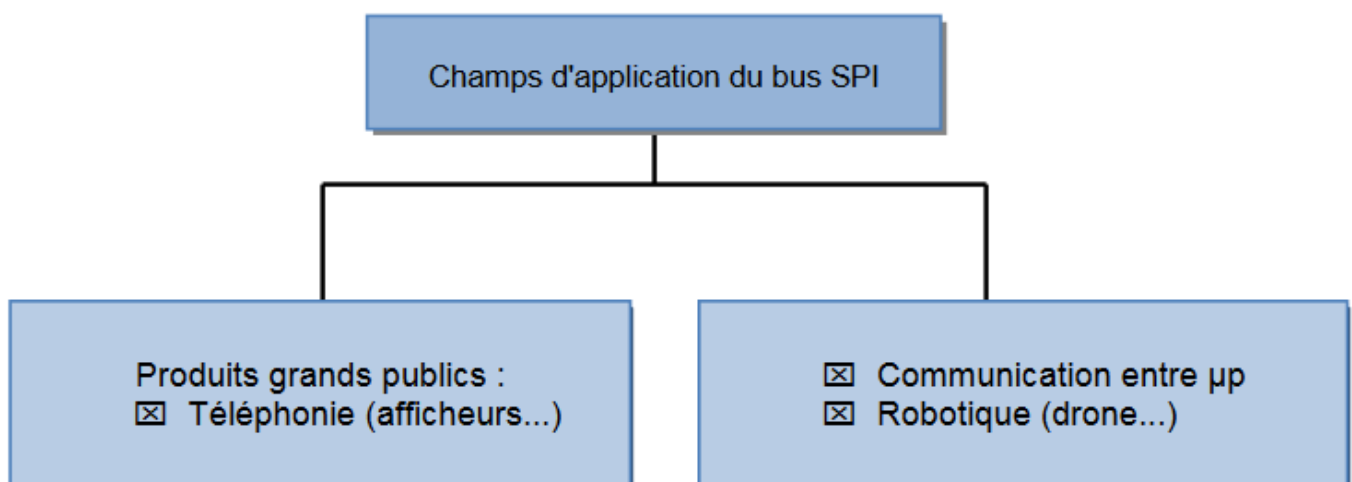
D'autres fabricants (Microchip, Atmel, Texas Instrument...) ont adopté pour ce type de liaison et de nombreux composants sont apparus (mémoires, capteurs, micro contrôleurs ...)

### I. 2. La philosophie du bus I<sup>2</sup>C



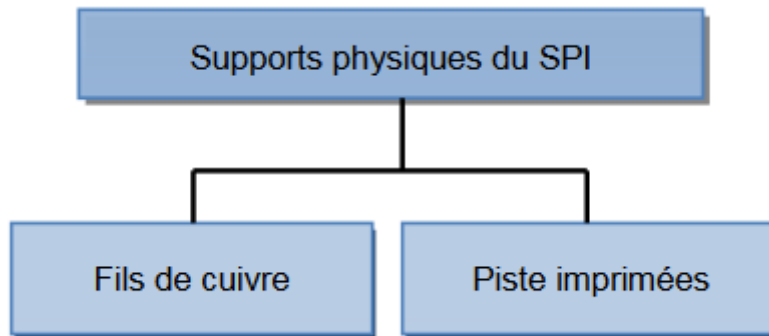
### I. 3. Les champs d'application du bus I<sup>2</sup>C

Il est utilisé pour la communication de données entre périphériques d'un appareil comme par exemple mémoires, les systèmes affichages, les cartes SD, etc.



## II. Propriétés physiques du bus I<sup>2</sup>C.

### II. 1. Le support physique



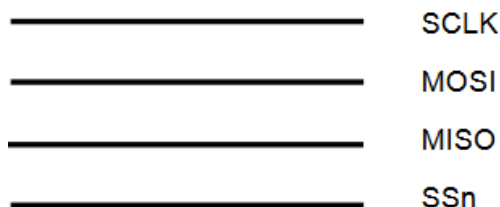
#### a) Les caractéristiques de fonctionnement du bus SPI

Les données échangées sont des octets. La transmission s'effectue sur 2 fils monodirectionnels (nommés MOSI, MISO). Une horloge indépendante fixée par le maître synchronise les échanges (en général sur front).

La fréquence de l'horloge de transmission est comprise entre 1Mhz et 20Mhz (selon les performances des circuits reliés au bus).

Il n'y a pas d'adressage des esclaves (comme sur un bus i2C par exemple). L'esclave devient actif au moyen d'une ligne de sélection de boîtier dédiée (généralement active à l'état bas).

La ligne est constituée de 3 fils auxquels il faut ajouter les fils de sélection d'esclave.



#### **SCLK (serial clock) :**

Horloge du bus (produite par le maître)

#### **MOSI (Master Out Slave In) :**

Données du maître vers l'esclave actif

#### **MISO (Master In Slave Out) :**

Données de l'esclave actif vers le maître

#### **SSn (Slave Select n) :**

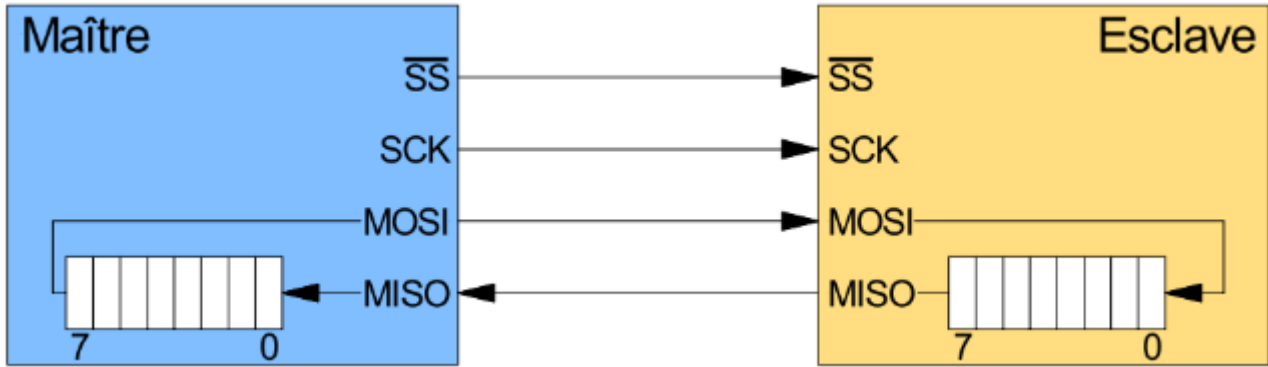
Sélection de l'esclave n à destination de la transmission

#### b) La fréquence d'horloge de transmission (maître) est paramétrable

Par exemple pour un AT80C5112 en mode maître, les 3 bits du registre SPCON (Serial Peripheral CONTROL register) SPR2, SPR1 et SPR0 permettent de choisir parmi 7 fréquences, obtenues par division de la fréquence de l'horloge du microcontrôleur.

SPR2	SPR1	SPR0	Fréquence de l'horloge SPI
0	0	0	$F_{\mu c}/2$
0	0	1	$F_{\mu c}/4$
0	1	0	$F_{\mu c}/8$
0	1	1	$F_{\mu c}/16$
1	0	0	$F_{\mu c}/32$
1	0	1	$F_{\mu c}/64$
1	1	0	$F_{\mu c}/128$

## II. 2. Synoptique d'une liaison SPI Maître-Esclave



On utilise le principe du registre à décalage. Dans le cas ci-dessus, en 8 périodes d'horloge, l'octet passe du registre du maître à celui de l'esclave et réciproquement le contenu du registre d'esclave est passé dans celui du maître (full-duplex: simultanéité des transferts). Puisqu'il ne peut pas y avoir de collisions lors du transfert, il n'y a pas besoin d'arbitrage.

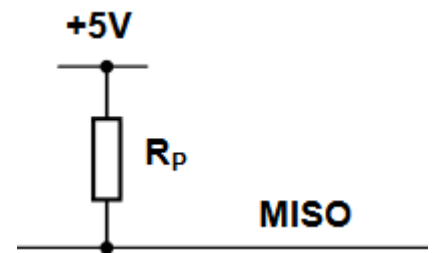
### a) Lien avec le modèle OSI

La couche 2 est réalisée par le contrôleur SPI intégré dans un microcontrôleur.

### b) Polarisation de la ligne MISO

Lorsque le bus est inutilisé, ce qui revient à dire qu'aucun esclave n'est sélectionné, la ligne MISO est à l'état haute impédance, ce qui ne permet pas d'en définir l'état logique.

On évite cela par l'utilisation d'une résistance de polarisation, de 5 à 50 kOhms, qui n'a aucune influence sur la vitesse de transmission (contrairement à ce qui se passe pour un bus i2C).



### c) Rôle des bits CPOL et CPHA (pour un AT80C5112) dans le mode de service du bus SPI

Dans les caractéristiques du bus SPI du microcontrôleur AT80C5112, on peut déterminer, grâce à deux paramètres, les fronts où les données sont transmises (acquisitions des valeurs) et les moments où elles peuvent être modifiées.

Ces deux paramètres sont les bits CPOL (ClockPolarity) et CPHA (ClockPhase).

Il existe donc 4 modes de transmission différents (voir tableau ci-dessous). Pour une transmission correcte il faut que ces paramètres soient réglés de la même manière pour tous les composants reliés au bus.

Mode SPI	CPOL	CPHA
0	0	0
1	0	1
2	1	0
3	1	1

Le **CPOL** détermine si au repos l'horloge est au niveau BAS (CPOL=0) ou HAUT (CPOL=1).  
Le **CPHA** détermine à quel front de l'horloge les données sont transmises. CPHA=0 les données sont valides au premier front d'horloge, CPHA=1 elles sont valides au deuxième front.

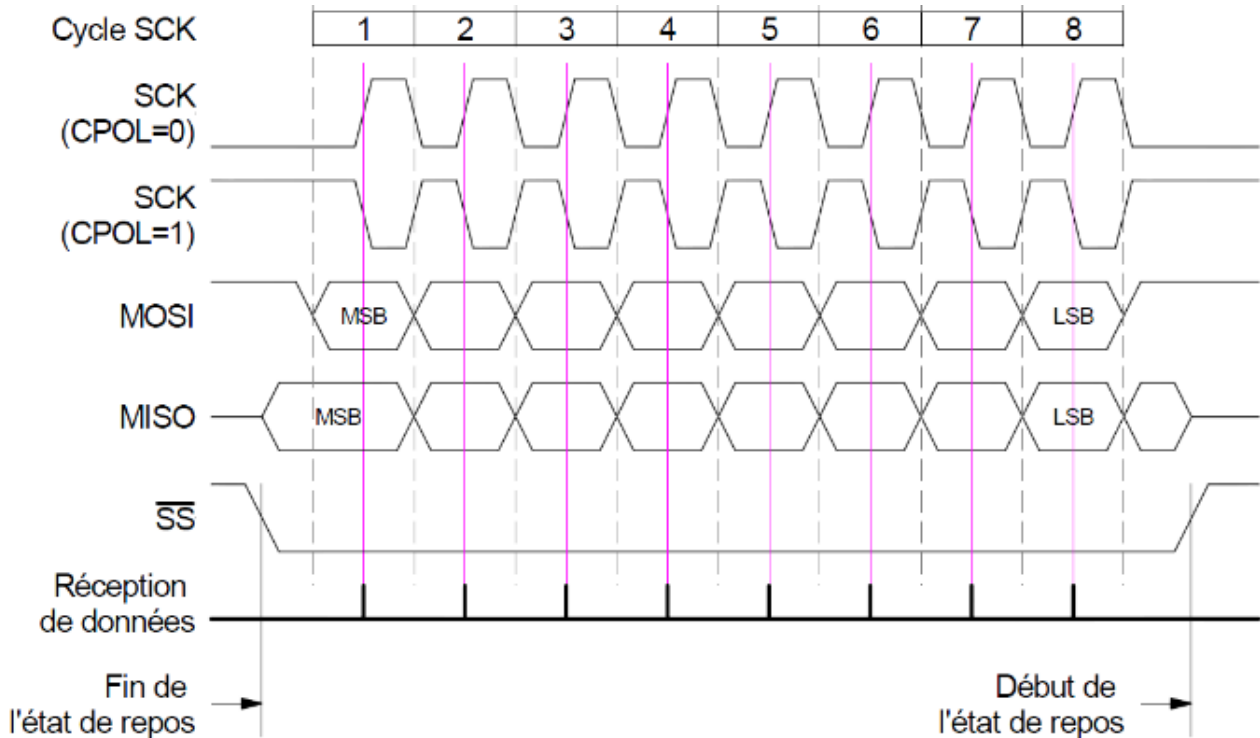
## II. 3. Format de transfert de données

Lorsque CPHA = 0, les données sont valides au premier front du signal d'horloge. La polarité CPOL détermine s'il s'agit d'un front montant ou descendant.

## BT : Bus SPI

En effet, pour CPOL=0, au repos, l'horloge est au niveau BAS; le premier front est donc un front montant.

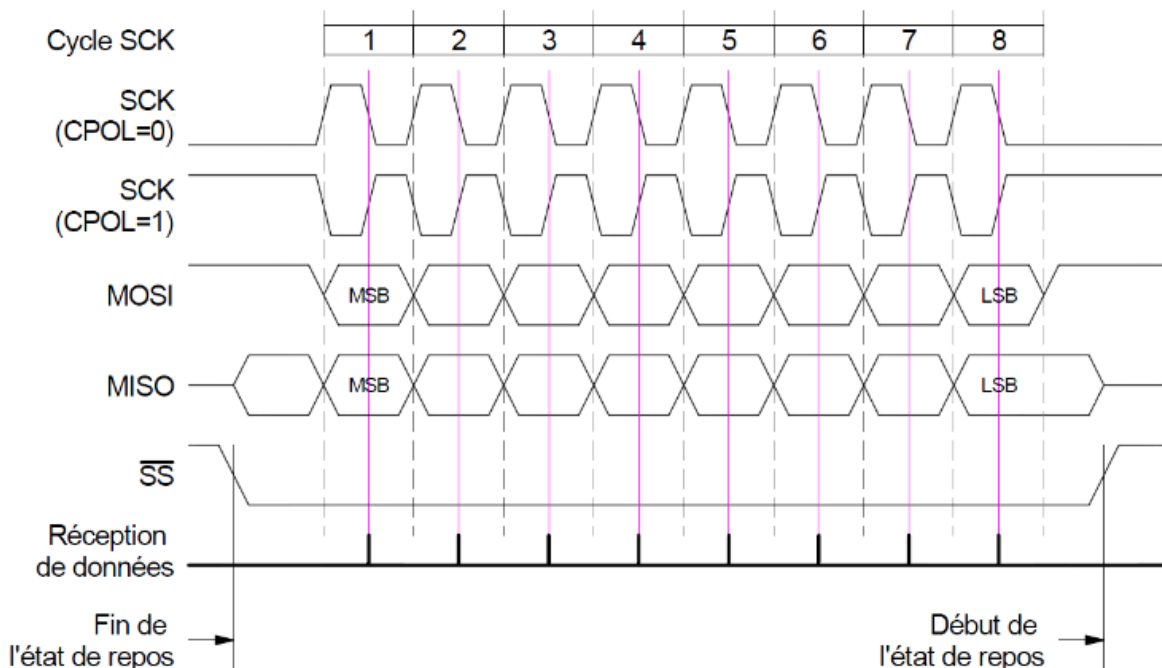
Pour CPOL=1, au repos, l'horloge est sur le niveau HAUT; le premier front est donc un front descendant. La polarité de l'horloge n'ayant pas d'influence sur le moment où le premier bit de données est valide elle n'a pas d'effet sur le format du transfert de données (voir figure ci-dessous).



Lorsque CPHA =1, les données sont réceptionnées avec le deuxième front du signal d'horloge.

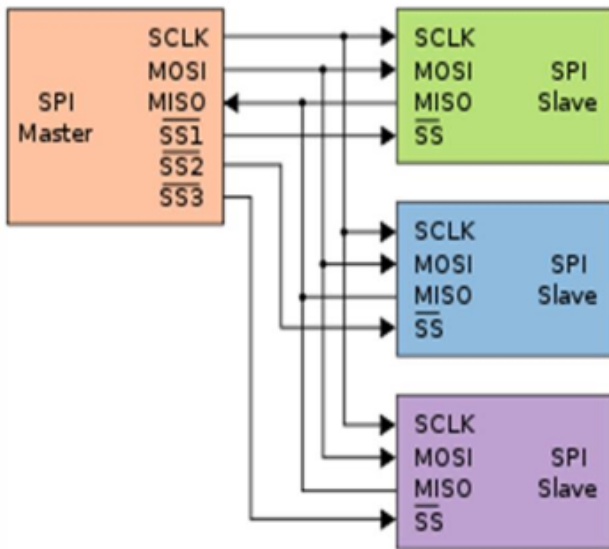
Pour CPOL=0, au repos, l'horloge est au niveau BAS et monte au niveau HAUT après le premier front, le deuxième front est donc un front descendant.

Pour CPOL=1, au repos, l'horloge est au niveau HAUT et descend au niveau BAS après le premier front; le deuxième front est donc un front montant.



## II. 4. Synoptique d'une liaison SPI Maître-Multi-Esclaves

### Plusieurs esclaves



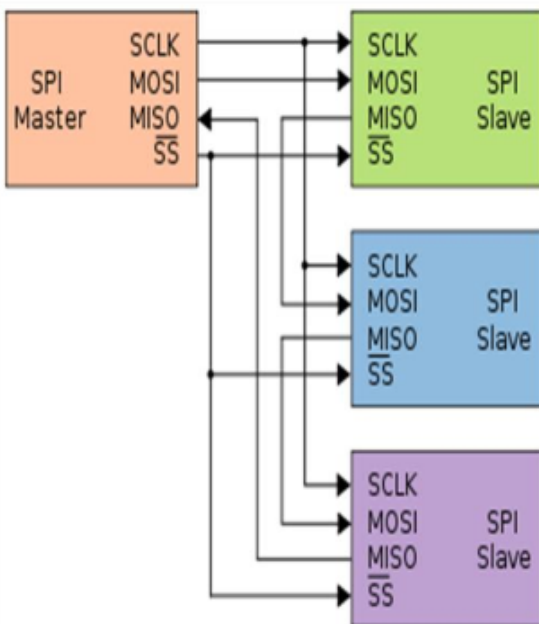
Le maître sélectionne un seul et unique esclave avec lequel il veut rentrer en communication par la mise à niveau logique zéro de /SS 1 2 3, puis, après 8 fronts d'horloge, l'octet de donnée est transféré.

La patte MISO de l'esclave non sélectionné est à l'état haute impédance.

La seule limite au nombre d'esclaves est en fait la possibilité de broches SS du maître.

#### a) Cas de la daisy chain (esclave en guirlande, en cascade...)

### Plusieurs esclaves en *daisy chain*



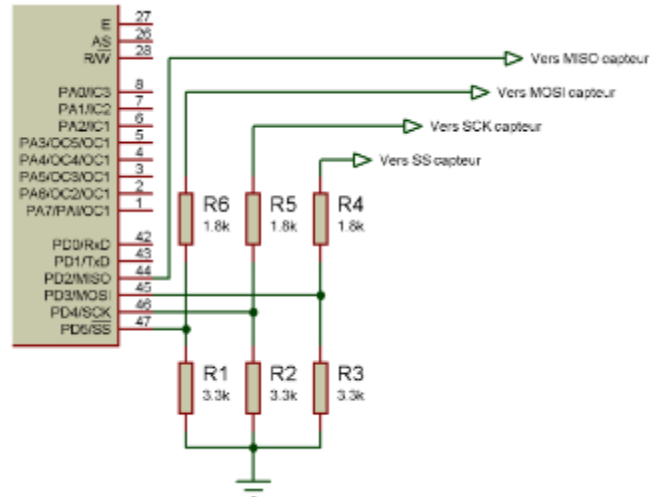
Le maître sélectionne tous les esclaves par la mise à niveau logique zéro de /SS, puis après 3\*8 fronts d'horloge, les 3 octets de données sont transférés (dans le cas d'un octet par esclave).

Cette disposition permet de réduire le nombre de lignes /SS, mais en contre partie il faudra un "buffer" plus grand dans le maître (ou une gestion du soft plus élaborée).

## II. 5. Bus SPI et niveaux électriques

Les signaux échangés sont de types TTL ou CMOS. Il pourra-t-être envisagé dans certains cas de placer des résistances de Pull-up de 47kΩ

Si on utilise des composants de tension d'alimentation différente par exemple un µp en 5v et un capteur en 3.3v, il convient de procéder à une adaptation du niveau de tension des broches MOSI, SS, SCLK (la MISO n'est pas affectée car compatible avec le µp) par diviseur de tension à base de résistance 1/4 de watt.



## III. Les différents autres noms utilisés avec un bus SPI

### III. 1. SCK

Il s'agit de l'horloge de synchronisation des échanges, elle est générée par le maître et est commune à tous les éléments du bus. Cela a pour avantage d'éviter que chaque composant possède son propre quartz avec les problèmes de précision que cela impose.

### III. 2. SDI, DI, SI, SDO, DO, SO

Il s'agit de la patte de Serial Data In; dans ce cas de convention d'écriture il convient de relier la SDI du maître à la SDO du ou des esclaves.

Il s'agit de la patte de Serial Data Out; dans ce cas de convention d'écriture il convient de relier la SDO du maître à la SDI du ou des esclaves.

### III. 3. nCS, CS ,nSS, STE

Il s'agit de la patte /SS Slave Select générée par le maître.

### III. 4. QSPI Queued Serial Peripheral Interface

- C'est un type de contrôleur dédié et spécifique qui permet certains transferts de données sans l'utilisation du processeur, mais en faisant appel à des pointeurs programmables, pointant une file d'attente de données.

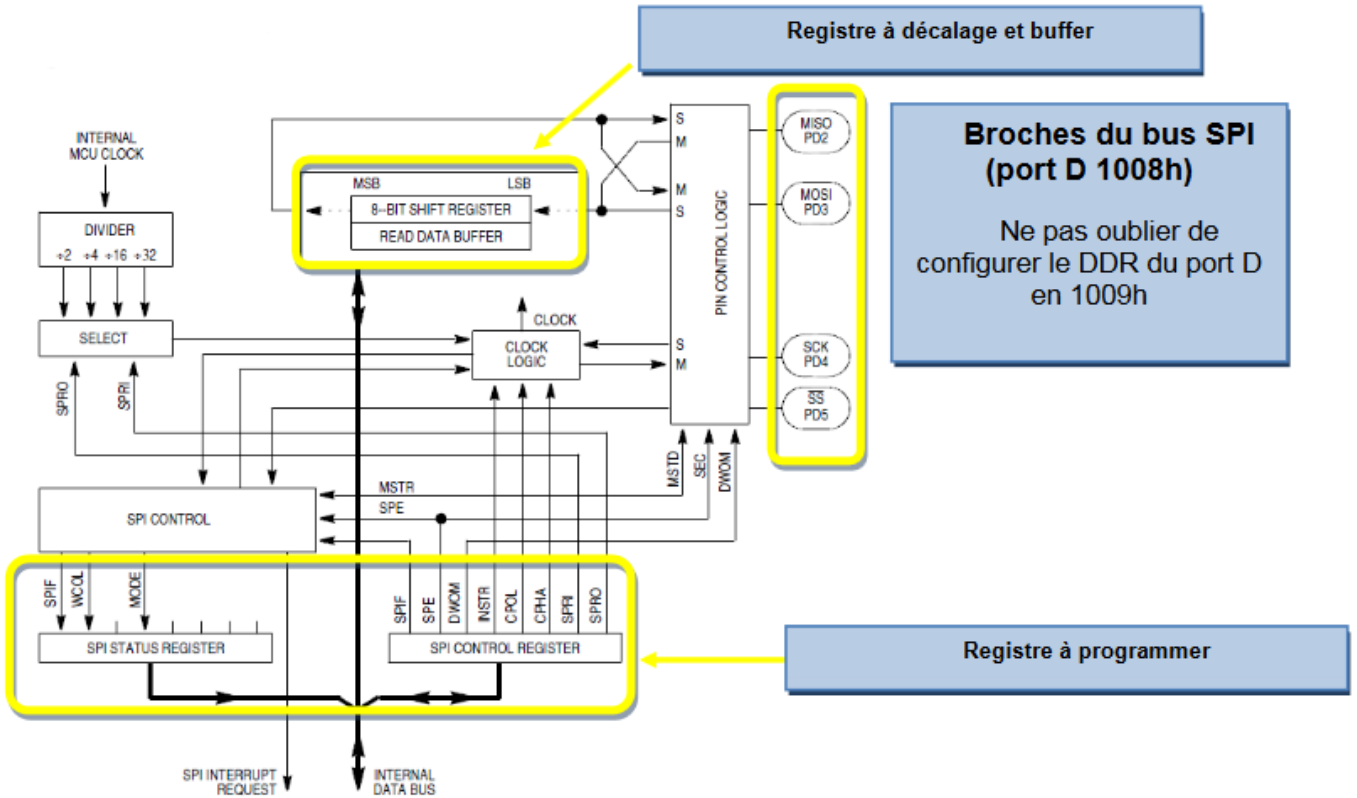
Ce n'est pas un autre type de bus série.

### III. 5. Avantages et Inconvénients du bus SPI

Avantages	Inconvénients
Communication en Full Duplex	Pas d'adressage possible
"Indépendant" du nombre de bits à transmettre	Utilisation sur très courte distance (même carte)
Pas de collision possible	Nécessite plus de fils que I <sup>2</sup> C
Les esclaves utilisent l'horloge du maître pas de problème de précision de quartz	Pas d'acquiescement (le maître ne sait pas s'il est écouté)
Beaucoup plus rapide que I <sup>2</sup> C en mode standard	
Possibilité de configuration à plusieurs maîtres	

## IV. Constitution d'un système Motorola

### IV. 1. Le diagramme bloc SPI



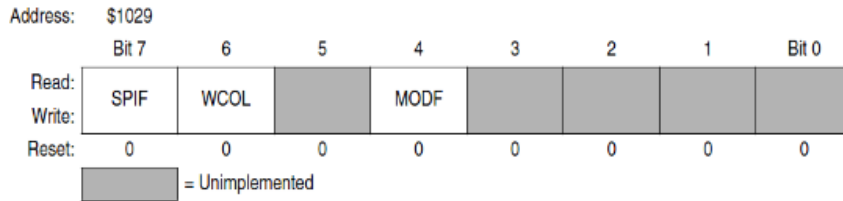
Le mode SPI peut-être configuré pour fonctionner soit en Maître soit en Esclave, la vitesse de transmission varie de 1,5MHz à 3MHz. On remarque la bidirectionnalité du registre à décalage.

Address:	\$1028							
	Bit 7	6	5	4	3	2	1	Bit 0
Read:	SPIE	SPE	DWOM	MSTR	CPOL	CPHA	SPR1	SPR0
Write:								
Reset:	0	0	0	0	0	1	U	U

U = Unaffected

<b>SPIE</b>	Validation de l'interruption en fin de transfert = 1 interruption validée (si le SPIF du SPSR passe à 1 on a une interruption) = 0 interruption inhibée (si le SPIF du SPSR passe à 1 on n'a pas interruption, il faut le scruter)
<b>SPE</b>	Validation SPI = 1 le port D est en SPI = 0 le port D est en I/O normal
<b>DWOM</b>	Port D Wire or Mode Option : = 1 les sorties du port D sont en type collecteur ouvert = 0 elles sont de types CMOS
<b>MSTR</b>	Master Mode Select: = 1 le 68HC est en Master (SS à 1) = 0 le 68HC est en Slave (SS à 0)
<b>CPOL</b>	Clock POLarity = 1 le SCK est à 1 au repos = 0 le SCK est à 0 au repos
<b>CPHA</b>	Clock PHase = 1 le SCK est valide sur front montant = 0 le SCK est valide sur front descendant
<b>SPR1 &amp; 0</b>	= 00 l'horloge interne est divisée par 2 = 01 l'horloge interne est divisée par 4 = 10 l'horloge interne est divisée par 8 = 11 l'horloge interne est divisée par 16 Pas d'effet en mode slave

## IV. 2. Le registre SPSR : Serial Peripheral Status Register



<b>SPIF</b>	Drapeau de fin de transfert SPI / Flag SPI (si SPIE=1) = 1 le transfert est terminé = 0 le transfert n'est pas terminé
<b>WCOL</b>	Write COLLision = 1 collision en écriture détectée = 0 pas de collision
<b>MODF</b>	Mode Fault: drapeau qui indique un défaut de mode = 1 (le 68HC est en master avec un SS=0) = 0

## IV. 3. Le registre SPDR : Serial Peripheral Data Register



C'est le registre où s'effectue le transfert des données.

## IV. 4. Le système de détection d'erreur SPI

Seules 2 types d'erreurs sont détectés:

- Erreur de mode: détection d'un niveau bas sur SS alors que le 68HC est déclaré en Maître, il ne peut y avoir 2 maîtres sur le même bus SPI
- Erreur de collision: détection d'une écriture dans le SPDR alors que le transfert n'est pas terminé.

## V. Constitution pour un système à microcontrôleur PIC (Microchip)

### V. 1. Le MSSP Control Register 1 (SSPCON1)

REGISTER 19-2: SSPCON1: MSSP CONTROL REGISTER 1 (SPI MODE)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
WCOL	SSPOV <sup>(1)</sup>	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0
bit 7							bit 0

<b>Legend:</b>			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

## BT : Bus SPI

bit 7 <b>WCOL</b>	: Write Collision Detect bit (Transmit mode only) 1 = Le registre SSPBUF register est écrit pendant que l'octet prévu est transmis (Doit être effacé par le programme) 0 = Pas de détection
bit 6 <b>SSPOV</b>	: Receive Overflow Indicator bit(1) SPI mode esclave: 1 = un nouvel octet est reçu pendant que le registre SSPBUF est encore en possession de l'ancien. En cas d'"overflow"(débordement) la donnée du SSPSR est perdue. Le débordement n'intervient que dans le mode Esclave. L'utilisateur doit lire le SSPBUF, seulement en cas de transmission de donnée, pour éviter le maintien de l'overflow il doit être effacé dans le soft. 0 = Pas de débordement
bit 5 <b>SSPEN</b>	: Master Synchronous Serial Port Enable bit 1 = validation de la configuration SCK, SDO, SDI and SS comme port série(2) 0 = pas de validation du mode SPI, les broches sont en E/S normales(2)
bit 4 <b>CKP</b>	: Clock Polarity Select bit 1 = horloge au niveau haut logique au repos 0 = horloge au niveau bas logique au repos
bit 3-0 <b>SSPM3:SSPM0</b>	: Master Synchronous Serial Port Mode Select bits 0101 = SPI mode Esclave, clock = SCK, SS non validée, SS peut être utilisée en E/S(3) 0100 = SPI mode Esclave, clock = SCK, SS (select slave) validée(3) 0011 = SPI mode Maître, clock = TMR2 output/2(3) 0010 = SPI mode Maître, clock = FOSC/64(3) 0001 = SPI mode Maître, clock = FOSC/16(3) 0000 = SPI mode Maître, clock = FOSC/4(3)

### V. 2. Le MSSP Status Register (SSPSTAT)

REGISTER 19-1: SSPSTAT: MSSP STATUS REGISTER (SPI MODE)

R/W-0	R/W-0	R-0	R-0	R-0	R-0	R-0	R-0
SMP	CKE <sup>(1)</sup>	D/Ā	P	S	R/W	UA	BF
bit 7							bit 0

**Legend:**

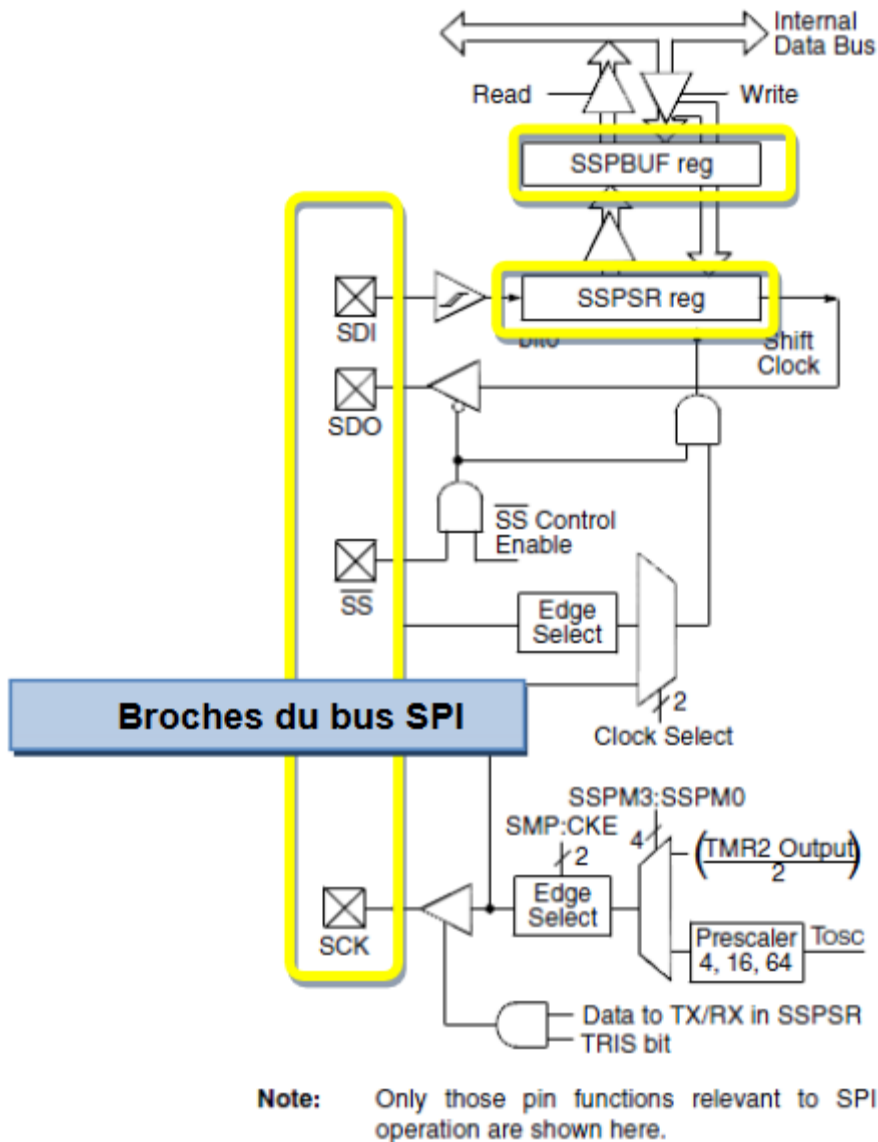
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared      x = Bit is unknown

bit 7 <b>SMP</b>	: Sample bit <b>Dans le mode Maître</b> 1 = Echantillonnage de la donnée entrante à la fin du temps de sortie de la données sortante 0 = Echantillonnage de la donnée entrante au milieu du temps de sortie de la données sortante <b>Dans le mode Esclave mis à 0</b>
bit 6 <b>CKE</b>	: SPI Clock Select bit 1 = Transmission sur front de l'état actif vers état repos de l'horloge 0 = Transmission sur front de l'état repos vers état actif de l'horloge <b>Dans le mode Esclave mis à 0</b>
bit 0 <b>BF</b>	: Buffer Full Status bit (Receive mode only) 1 = Réception complète, SSPBUF plein 0 = Réception incomplète, SSPBUF vide

Les bits 1, 2, 3, 4 et 5 ne sont pas utilisés en mode SPI.

### V. 3. Le Serial Receive/Transmit Buffer Register SSPBUF

Ce registre est utilisé pour lire ou écrire les données de / vers l'extérieur du PIC.



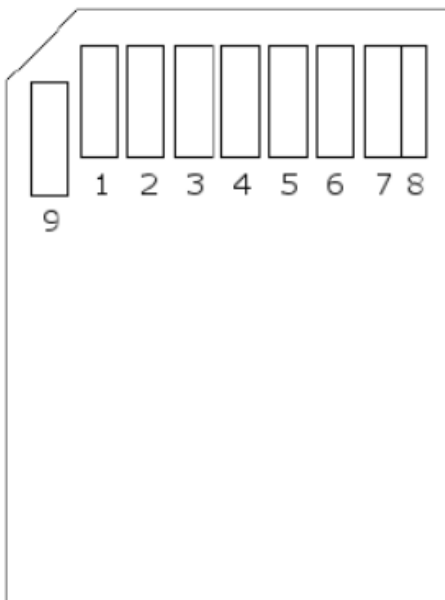
### V. 4. Le MSSP Shift Register

Ce registre est utilisé pour recevoir /envoyer les données de / vers l'extérieur du PIC.

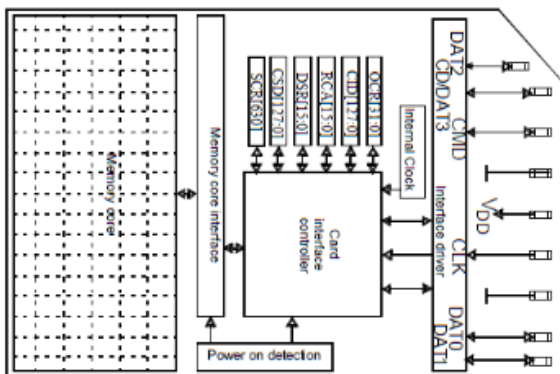
Pendant les opérations de transmission, les 2 registres SSPSR et SSPBUF créent un double tampon receveur, lorsque le SSPSR à reçu un octet, il le transfère au SSPBUF en provoquant la mise à 1 de SSPIF.

## VI. Complément 'SD Card'

Les cartes SD possèdent plusieurs interfaçages possibles dont un bus SPI. Le schéma ci-dessous présente la connectique des cartes SD, on y retrouve les broches spécifiques du bus SPI.



Pin	SPI Function
1	Chip select
2	MOSI
3	GND
4	VCC
5	SCLK
6	GND
7	MISO
8	NC (SD only)
9	NC (SD only)



Avec un microcontrôleur incluant une interface SPI spécifique et une programmation pour les routines supplémentaires d'accès à la mémoire de la carte, on peut développer des applications sur ce type de carte. Cependant, le protocole utilisé est complexe et, si l'on souhaite avoir une structure de fichier utilisable sur un ordinateur, on est confronté au problème de la FAT...