

Module: Informatique Industrielle 2

Carole Lavault

(All médical)

(carole.lavault@aii-biomedical.com)



Module: Informatique Industrielle 2

Objectifs:

*Connaître les principales caractéristiques des bus utilisés dans l'industrie et **Maîtriser les techniques d'interfaçage** (logiciel et matériel), pour:*

- **Savoir mettre en œuvre les cartes du commerce,**
- **Concevoir des cartes spécifiques dédiées aux bus courants,**
- **Interfacer des circuits périphériques aux microcontrôleurs,**
- **Sélectionner l'architecture adaptée à une application.**

Module: Informatique Industrielle 2

Pré requis

- Langages: algorithmique, *assembleur* & C.
- Éléments de base en électronique numérique et analogique,
- Notions élémentaires sur les Processeurs et les Microcontrôleurs.
- Programmation du périphérique GPIO du STM32
- Utilisation d'un oscilloscope

Module: Informatique Industrielle 2

Plan du cours (cours suivi de TPs)

1. Introduction, principaux types de bus, caractéristiques.
2. Les bus Inter-composants: le SPI. (3 séances 4h)
3. *Le bus PC104, techniques d'interfaçage, prototypage, exemples de cartes industrielles.*
4. Liaison série asynchrone: RS232 (1 séance 4h)
5. Introduction à Modbus (1 séance 4h)
6. Introduction aux Bus série USB. (2 séances 4h)

Module: Informatique Industrielle 2

Bibliographie (Livres, ouvrages & références)

- Architecture de l'ordinateur, 3^{ième} édition (*A. Tanenbaum*, InterEdition),
- Informatique Industrielle (*P. Dumas*, Dunod),
- Universal Bus System Architecture (*D. Anderson*, Mindshare - anglais),
- Le bus USB (*X. Fenard*, Dunod).
- Organisme de normalisation bus PCI: www.pcisig.com
- Spécifications bus USB: www.usb.org/developers/docs
- MODBUS Application Protocol Specification V1.1b3 (www.modbus.org/docs)

Module: Informatique Industrielle 2

1. Introduction, principaux types de bus, caractéristiques

Vocabulaire

- **Un bus informatique** permet de partager des données entre plusieurs composants d'un système numérique.
- **Transaction:** Echange entre 1 initiateur et 1 cible (ou destinataire)
- **Bande passante:**
 - Quantité d'information transmise dans un intervalle de temps.
 - Unité normative pour un bus série: bit/s.
 - " " " " " parallèle: byte/s.
- Bus **full duplex**: *transaction maitre à esclave & esclave à maitre simultanée.*
- Bus **bidirectionnel**: échange de données dans les 2 sens sur 1 fil.

Module: Informatique Industrielle 2

1. Introduction, principaux types de bus, caractéristiques

Principaux types de bus

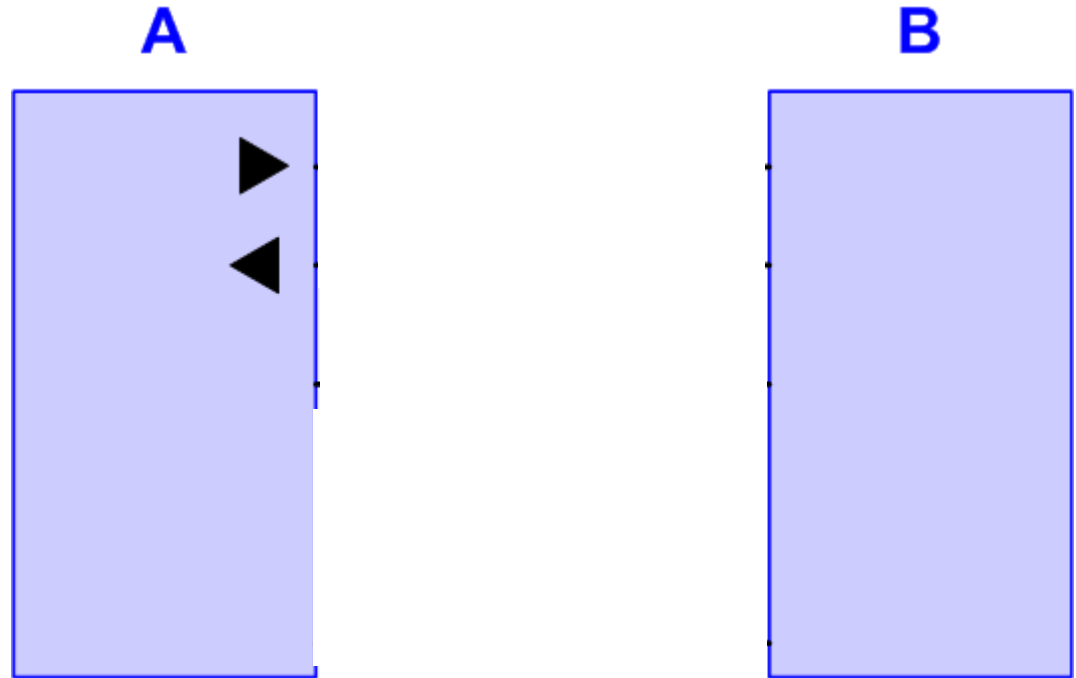
- Série synchrone, ex: SPI, I2C
- Série asynchrone, ex: RS232, RS485, CAN, LIN
- Parallèle synchrone: ex: PC104, PCI, DDRx-SDRAM...
- Parallèle asynchrone, ex: VME, interfaces mémoire (SRAM, Norflash) ...
- Liaison série avec clock recovery: PCIe, Ethernet, USB, ...

Module: Informatique Industrielle 2

1. Introduction, principaux types de bus, caractéristiques

Architecture minimum

➤ Série synchrone

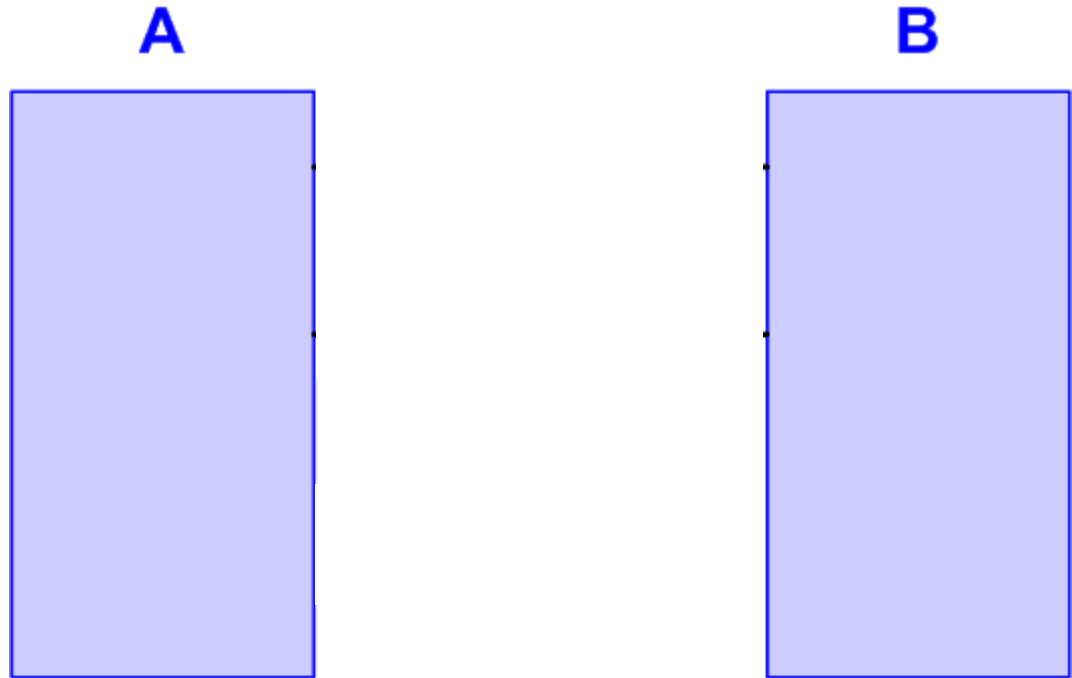


Module: Informatique Industrielle 2

1. Introduction, principaux types de bus, caractéristiques

Architecture minimum

➤ Série asynchrone



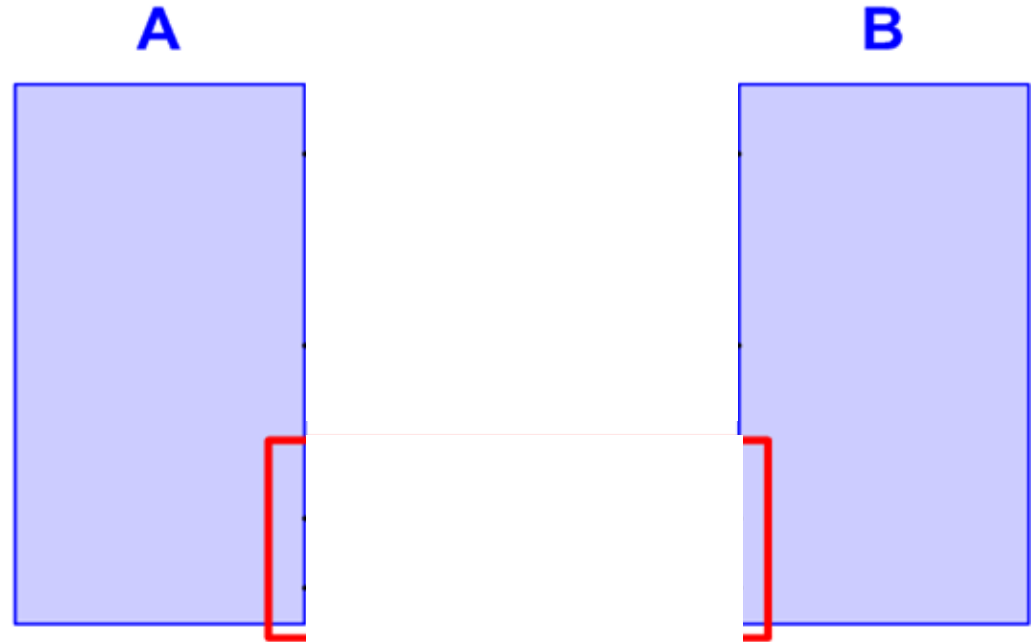
A & B => même horloge

Module: Informatique Industrielle 2

1. Introduction, principaux types de bus, caractéristiques

Architecture minimum

- Parallèle asynchrone

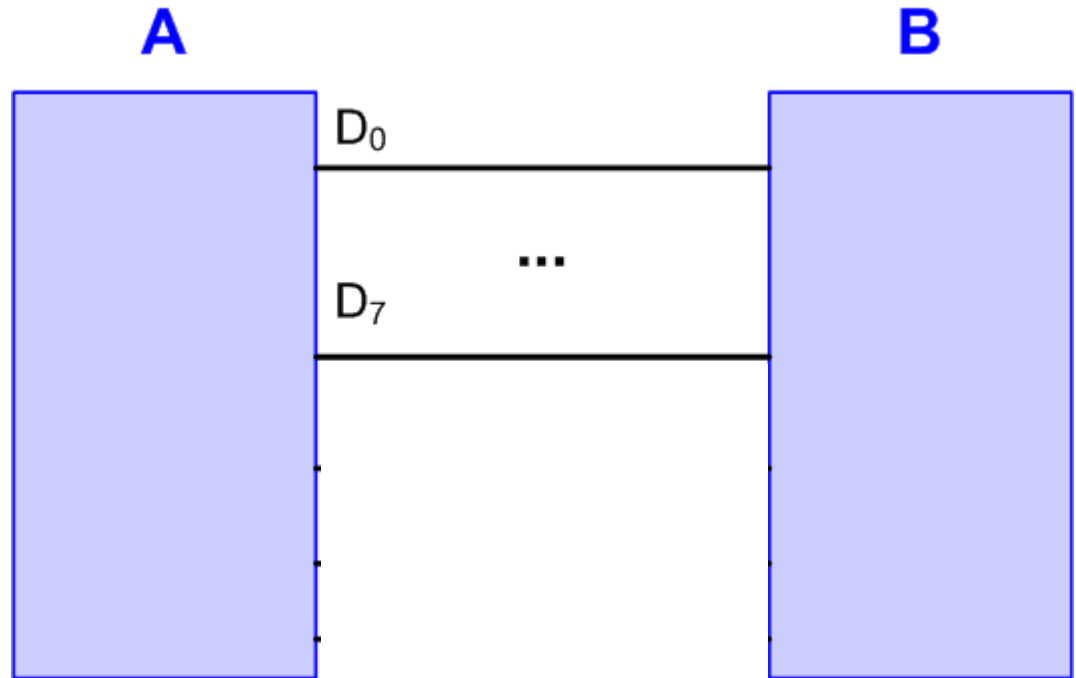


Module: Informatique Industrielle 2

1. Introduction, principaux types de bus, caractéristiques

Architecture minimum

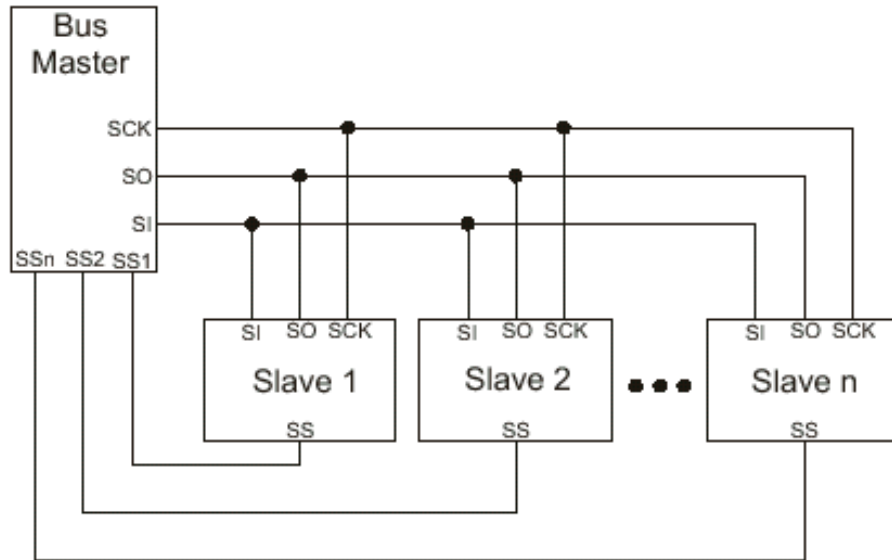
- Parallèle synchrone



Module: Informatique Industrielle 2

2. Les bus inter-composants, le SPI

- Bus série synchrone développé par Motorola Freescale et destiné à faire communiquer :
 - un processeur (en général un microcontrôleur)
 - des circuits périphériques (EEPROM, CAN, CNA, afficheurs LCD etc....)
- Interface SPI répandue sur de nombreux microcontrôleurs (HCS12, Freescale, PIC, Intel, ST, Microchip ...)



Partage d'un bus commun pour l'horloge, MISO et MOSI entre les périphériques



life.augmented

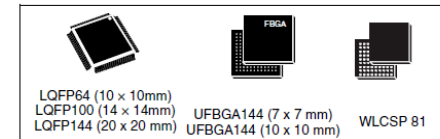
STM32F446xC/E

ARM® Cortex®-M4 32b MCU+FPU, 225DMIPS, up to 512kB Flash/128+4KB RAM,
USB OTG HS/FS, 17 TIMs, 3 ADCs, 20 comm. interfaces

Datasheet - production data

Features

- Core: ARM® 32-bit Cortex®-M4 CPU with FPU, Adaptive real-time accelerator (ART Accelerator™) allowing 0-wait state execution from Flash memory, frequency up to 180 MHz, MPU, 225 DMIPS/1.25 DMIPS/MHz (Dhrystone 2.1), and DSP instructions
- Memories
 - 512 kB of Flash memory
 - 128 KB of SRAM
 - Flexible external memory controller with up to 16-bit data bus: SRAM, PSRAM, SDRAM/LPDDR SDRAM, Flash NOR/NAND memories
 - Dual mode Quad SPI interface



- Up to 114 I/O ports with interrupt capability
 - Up to 111 fast I/Os up to 90 MHz
 - Up to 112 5 V-tolerant I/Os
- Up to 20 communication interfaces
 - SPDIF-Rx
 - Up to 4 × I²C interfaces (SMBus/PMBus)
 - Up to 4 USARTs/2 UARTs (11.25 Mbit/s, ISO7816 interface, LIN, IrDA, modem control)
 - Up to 4 SPIs (45 Mbits/s), 3 with muxed I²S

En général, sur les µC plusieurs interfaces SPI ce qui permet des paramétrages différents, différents format des trames, fréquences d'horloge

Module: Informatique Industrielle 2

2. Les bus inter-composants, le SPI

➤ Interface 4 fils :

SCK	Serial Clock	Maître : Sortie Esclave : Entrée	Horloge de synchronisation Les données sont décalées/latchées sur un front (montant, descendant de cette horloge)
MOSI SDO, SO	Master OUT Slave IN	Maître : Sortie Esclave : Entrée	Données transférées en sortie du maître vers l'esclave
MISO SDI, SI	Master IN Slave OUT	Maître : Entrée Esclave : Sortie	Données transférées en sortie de l'esclave vers le maître
/SS	Slave Select	Maître : Sortie Esclave : Entrée	Sélectionne l'esclave avec lequel un transfert va commencer doit rester au niveau bas pendant toute la transaction

Module: Informatique Industrielle 2

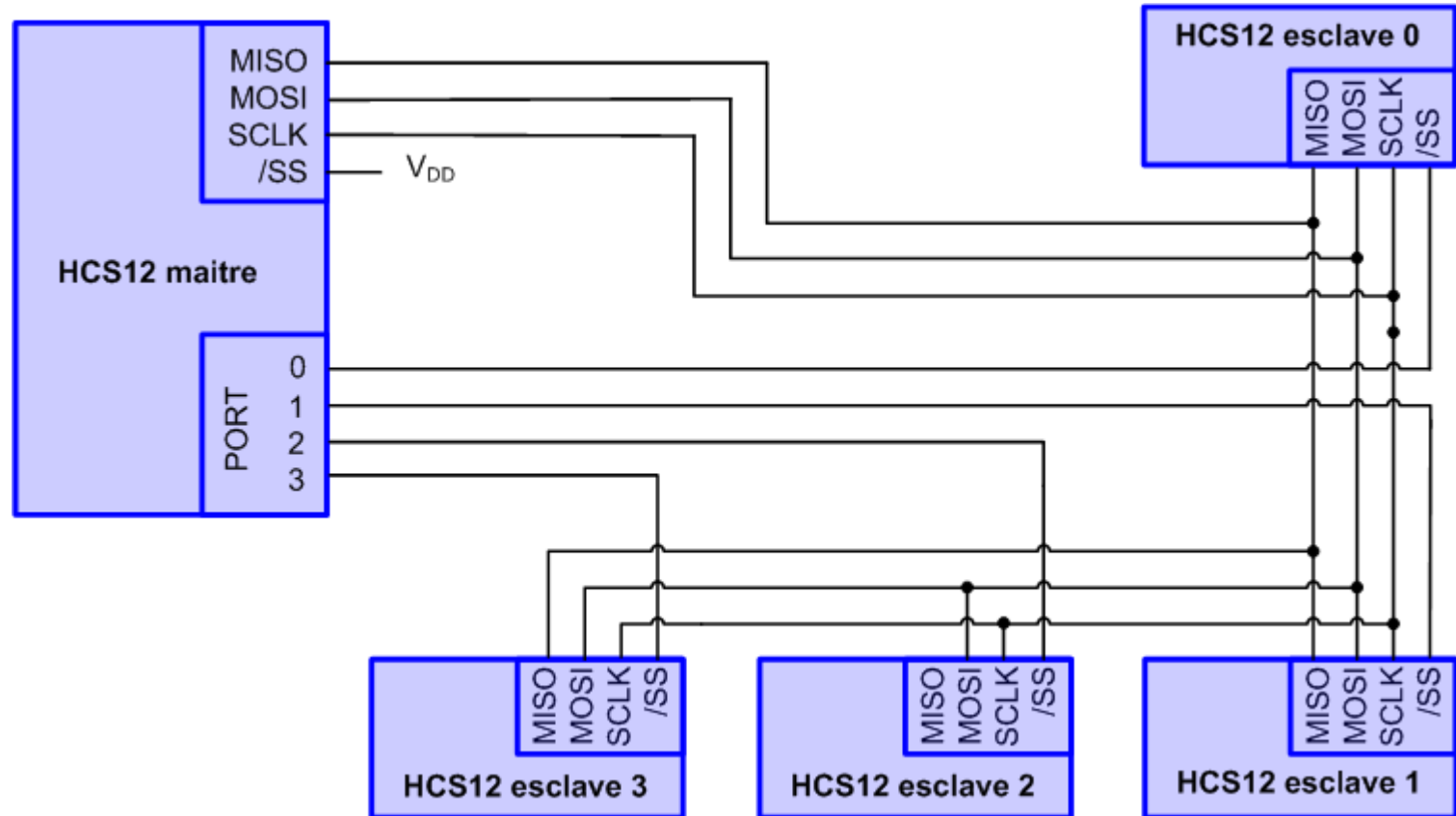
2. Les bus inter-composants, le SPI

- Architecture Maître/Esclave :
 - Le maître est responsable des activités de transfert sur le bus,
 - L'esclave répond aux commandes issues du maître,
 - 1 seul Maître par transaction,
 - Nb d'esclaves limités seulement par le nombre de signaux SS/ disponibles; dans les faits, quelques-uns.
- Horloge en général 1 à plusieurs dizaines MHz => Taux de transfert : 1-5 Mbit/s
- Vitesses supérieures possibles mais en général limitées par les esclaves
- Communication Full-duplex

Module: Informatique Industrielle 2

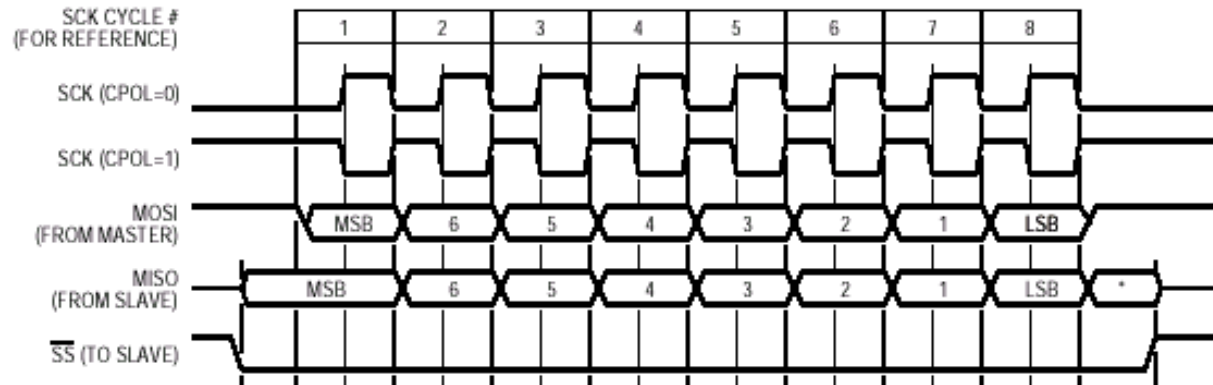
2. Les bus inter-composants le SPI

- Principe de connexion de divers circuits via l'interface SPI



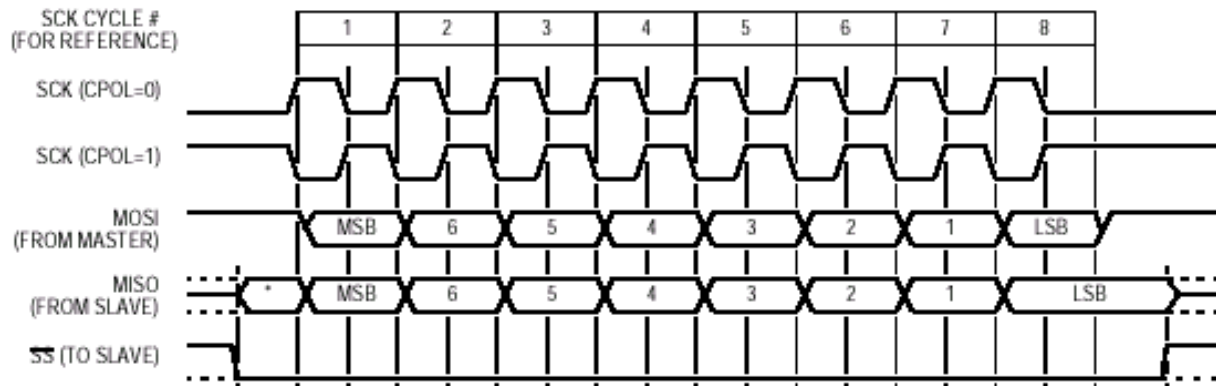
Module: Informatique Industrielle 2

2. Les bus inter-composants, le SPI



*Not defined but normally MSB of character just received.

Figure 8-1. CPHA Equals Zero SPI Transfer Format



*Not defined but normally LSB of previously transmitted character.

Figure 8-2. CPHA Equals One SPI Transfer Format

Module: Informatique Industrielle 2

2. Les bus inter-composants, le SPI

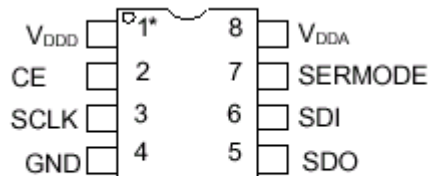


www.dalsemi.com

PRELIMINARY

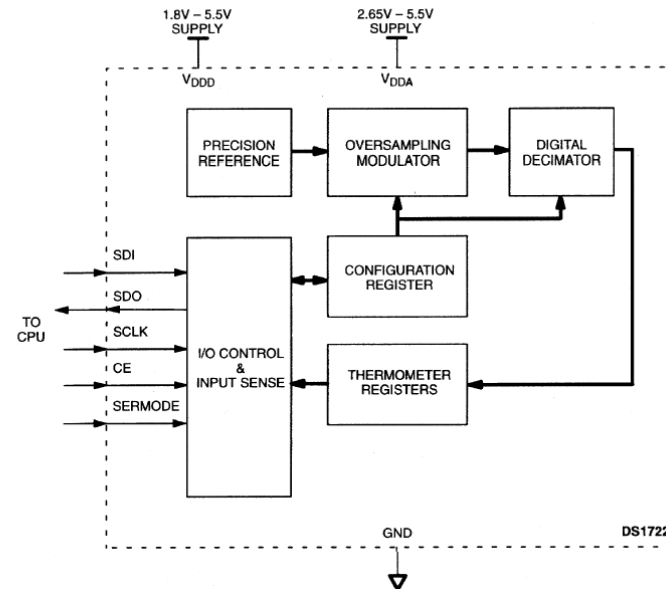
DS1722

Digital Thermometer with
SPI/3-Wire Interface



DS1722U
8-PIN μ -SOP

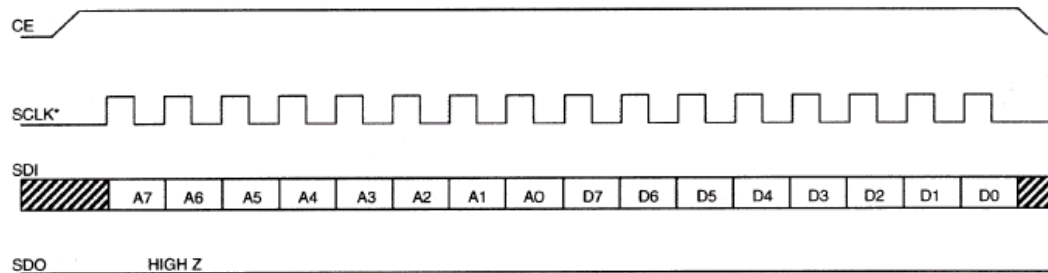
DS1722 FUNCTIONAL BLOCK DIAGRAM Figure 1



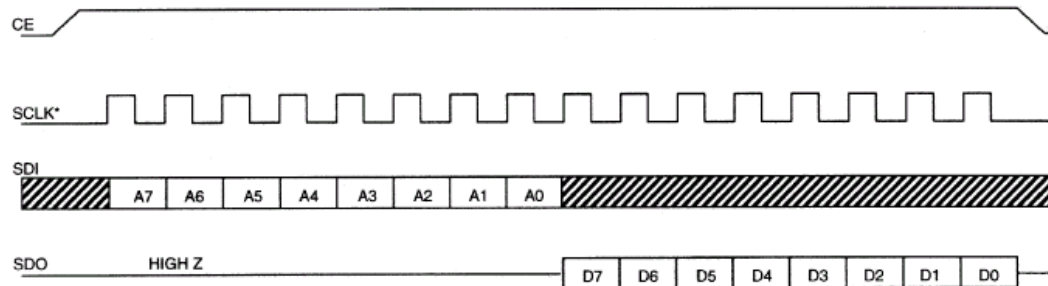
Module: Informatique Industrielle 2

2. Les bus inter-composants, le SPI

SPI SINGLE BYTE WRITE Figure 4



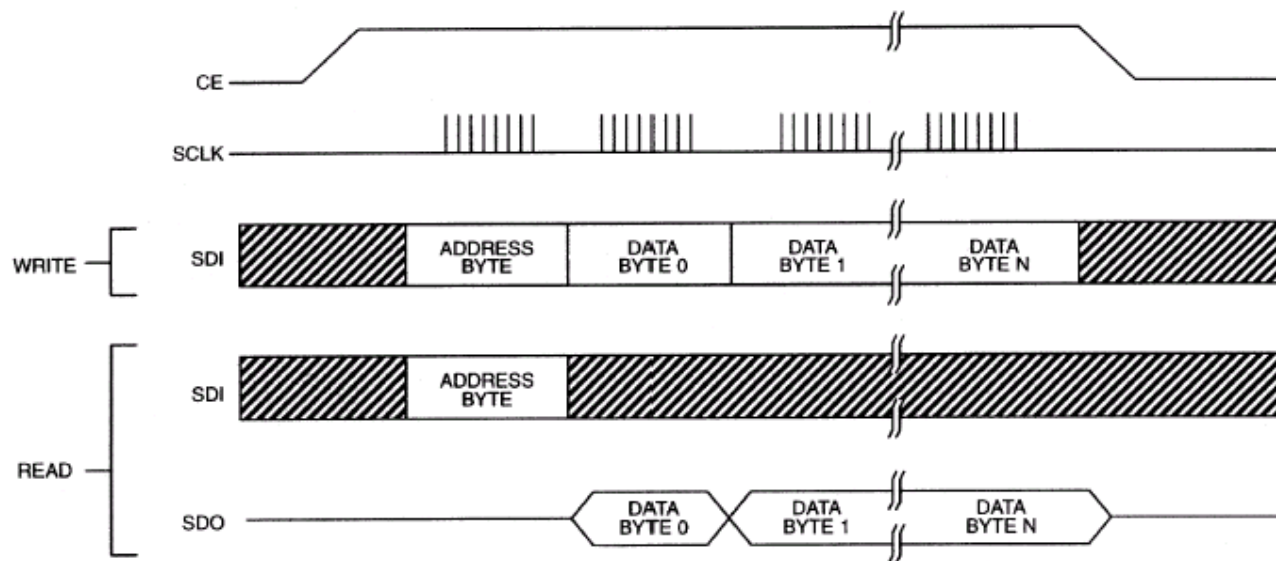
SPI SINGLE-BYTE READ Figure 5



Module: Informatique Industrielle 2

2. Les bus inter-composants, le SPI

SPI MULTIPLE BYTE BURST TRANSFER Figure 6



Lecture ou écriture de plusieurs octets à des adresses consécutives

Module: Informatique Industrielle 2

2. Les bus inter-composants, l'I2C

- Inter-Integrated Circuit bus
- Développé au départ par Philips Semiconductor pour faire communiquer les circuits dans les téléviseurs et les moniteurs et plus généralement,
 - Un processeur (en général un μC)
 - Avec des circuit périphériques (Eeprom, afficheurs, capteurs, ...)
- Dans la majorité des moniteurs modernes, des broches sont réservées pour ce bus sur le connecteur VGA (plug-and-play)
- 2 lignes seulement en collecteur/drain ouvert (+ masse)
 - SDA Serial Data Line bidirectionnelle
 - SCL Serial Clock line bidirectionnelle

Module: Informatique Industrielle 2

2. Les bus inter-composants, l'I2C

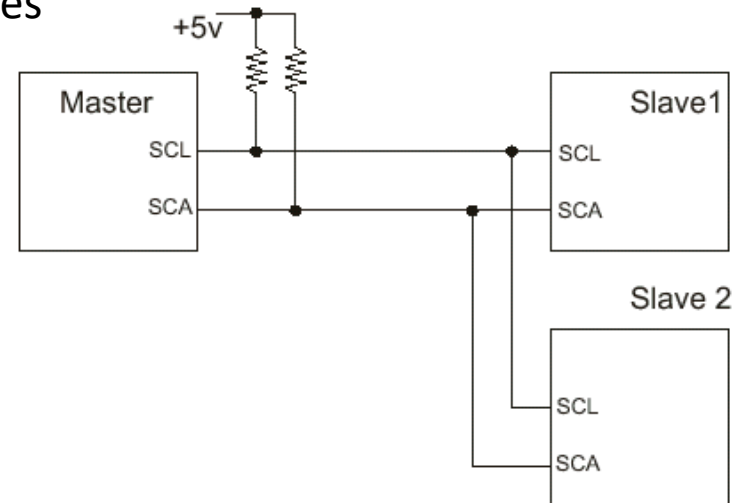
- Communication half-duplex
- Fréquence d'horloge
 - Sm mode 100khz
 - Fm mode 400kHz
 - Ultra-fast mode (UFm) ≤ 5 Mbit/s, unidirectionnel uniquement.
- Limitation nombre de circuits seulement par la capacitance maxi et la limite d'adressage (7 bits d'adressage et un bit R/W (lecture ou écriture), soit 128 périphériques)
- Chaque circuit possède une adresse unique sur le bus, et peut agir comme émetteur ou récepteur, selon sa fonctionnalité

Module: Informatique Industrielle 2

2. Les bus inter-composants, l'I2C

➤ Protocole Maître/Esclave

- Maître : contrôle l'horloge et les transferts de données entre lui et un esclave ou entre 2 esclaves
- Esclave : contrôlé par le maître

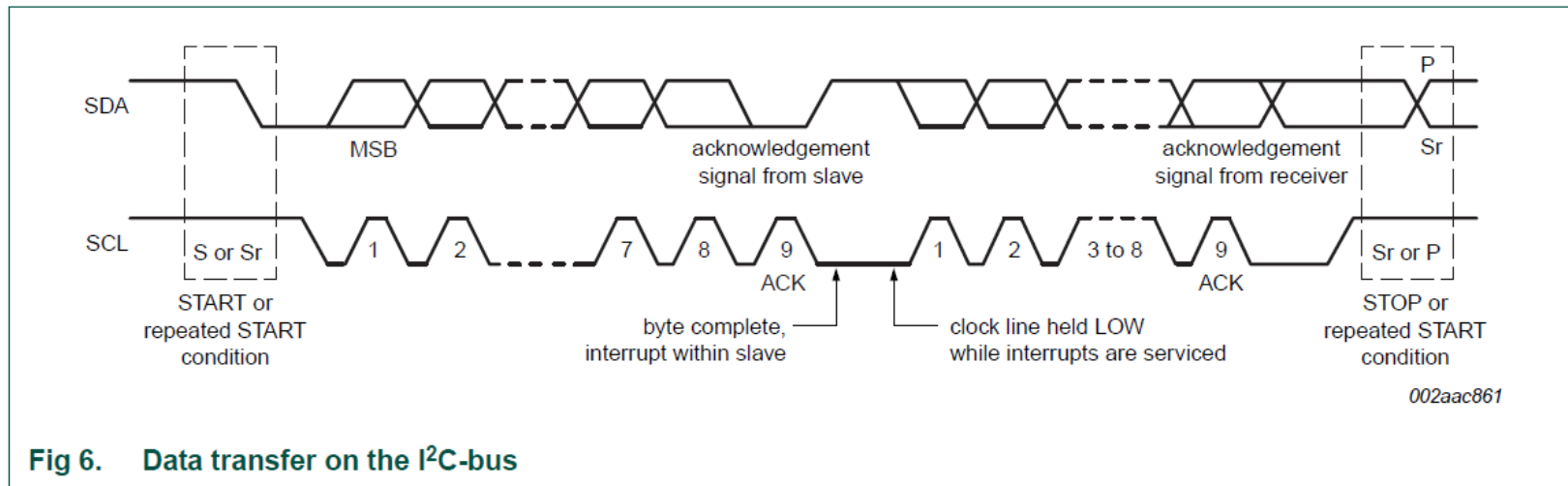


➤ Pour commencer une transaction :

- Le maître émet l'adresse du circuit avec lequel il veut communiquer
- Tous les circuits présents sur le bus testent si l'adresse est la leur
- Seul le circuit concerné répond

Module: Informatique Industrielle 2

2. Les bus inter-composants, l'I2C



Extrait User manuel NXP UM10204

Pas de chip select, remplacé par l'émission d'une adresse, la trame est encadrée par :

La condition de départ : SDA passe à 0, SCL reste à 1

La condition d'arrêt : SDA passe à 1, SCL reste à 1

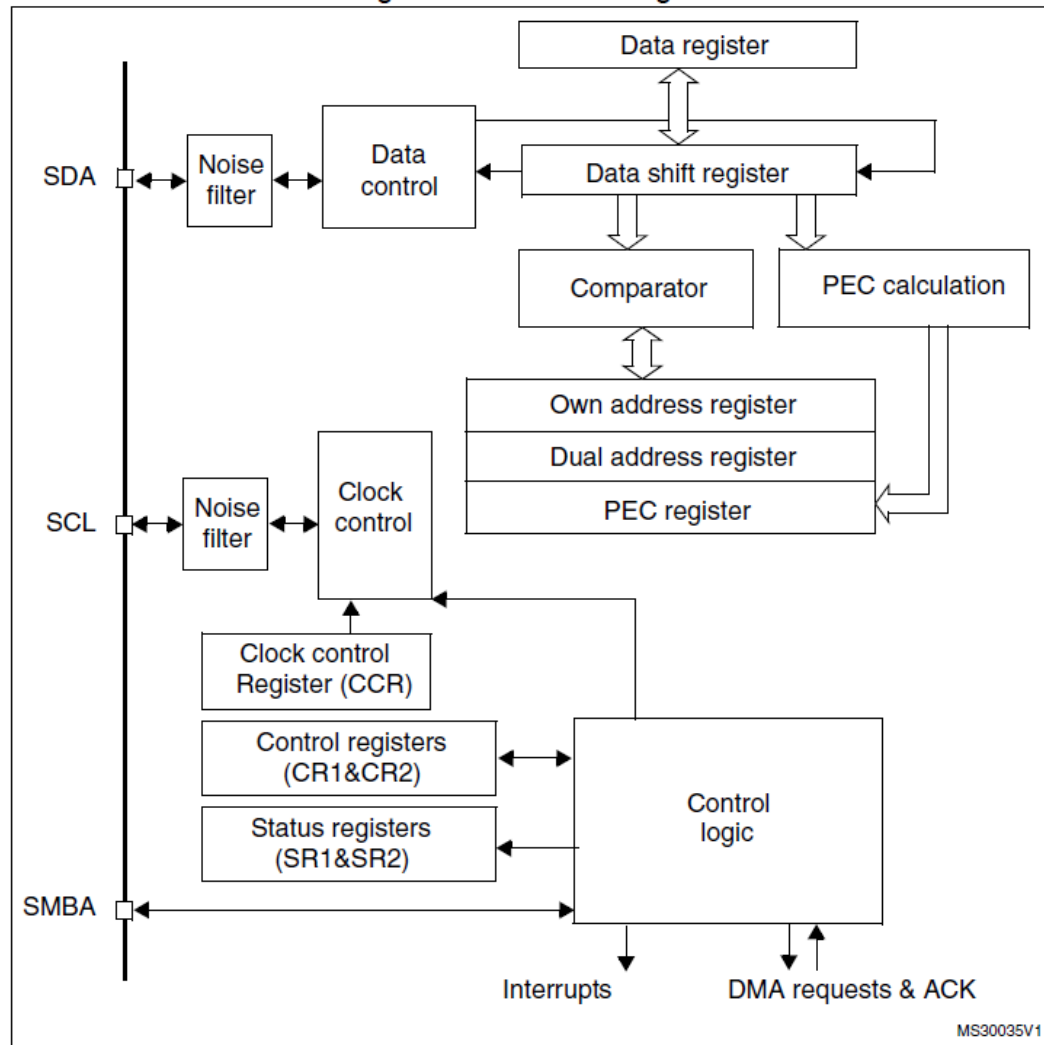
Bit ACK : SDA = 0, NACK SDA = 1 pour indiquer que la donnée a été échantillonnée correctement après un transfert 8bits.

Bit Restart, Pause...

Module: Informatique Industrielle 2

2. Les bus inter-composants, l'I2C

Figure 272. I²C block diagram



RM0390 Reference manual STM32F446xx

Module: Informatique Industrielle 2

2. Les bus inter-composants, les autres bus

Microwire

- Défini par National Semiconductor
- Equivalent au SPI dans le cas où CPOL = 0 et CPHA = 0

Maxim-3Wire

- Se trouve sur DS1620 et d'autres circuits Maxim
- Les entrées/sorties de données sont multiplexées sur une seule broche (DQ) (au lieu de MISO et MOSI pour le SPI)

Maxim-1Wire

- Permet de connecter (en série, parallèle ou en étoile) des composants avec seulement deux fils (un fil de données et un fil de masse)
- Exemple : DS2431-A1 1kbit Serial EEPROM for Automotive Applications
- Protocole et mise au point compliqués

Module: Informatique Industrielle 2

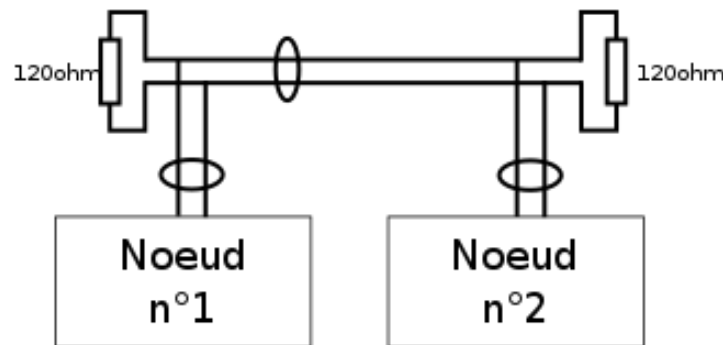
2. Les bus inter-composants, les autres bus

LIN *Local Interconnect Network*

- Le **bus LIN** (Les entrées/sorties de données sont multiplexées sur une seule broche (DQ) (au lieu de MISO et MOSI pour le SPI)
- Économique (entrée de gamme dans l'automobile)

Le bus CAN (*Controller Area Network*)

- Industrie automobile notamment
- ISO 11898
- Bus de terrain
- Créé par BOSCH
- Filaire, fibre, HF



Débit	Longueur
1 Mbit/s	30 m
800 kbit/s	50 m
500 kbit/s	100 m
250 kbit/s	250 m
125 kbit/s	500 m
62,5 kbit/s	1000 m
20 kbit/s	2500 m
10 kbit/s	5000 m

Module: Informatique Industrielle 2

2. Les bus inter-composants, le bloc SPI du STM32

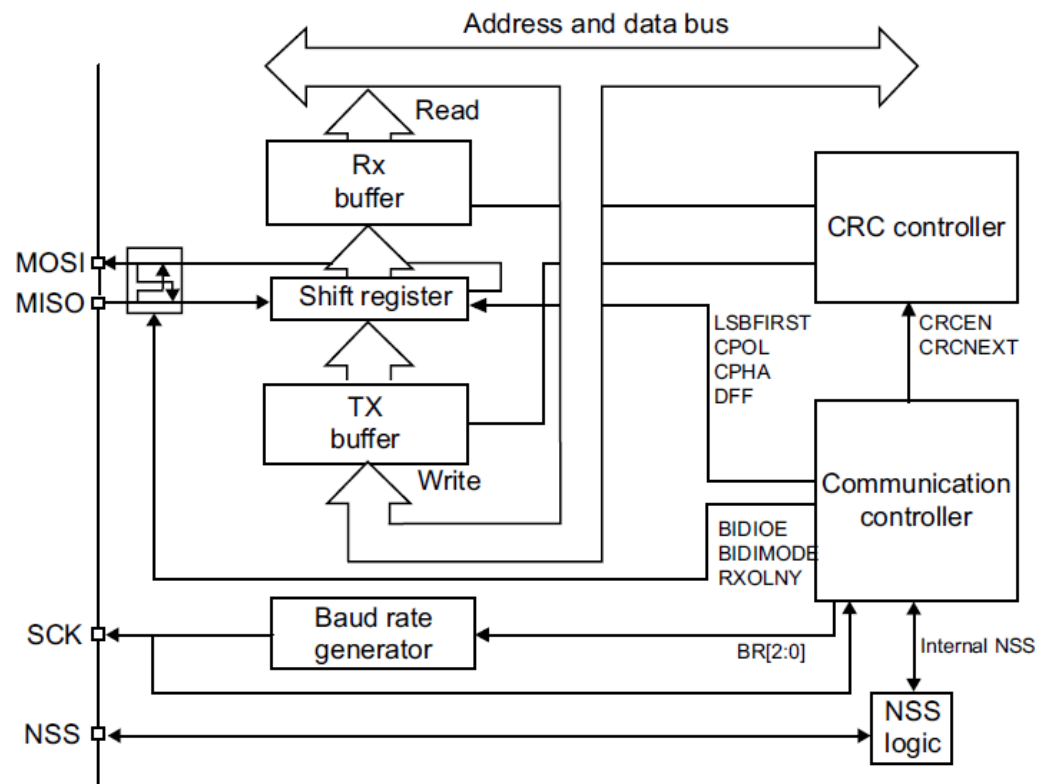
Rôle:

- Communication avec des circuits périphériques
 - ADC, DAC
 - Capteurs
 - Afficheurs LCD, 7 segments LCD
 - Mémoires EEPROM
 - Simple registre à décalage...
- Compatible avec de nombreux périphériques standard
- Bus inter processeurs en mode multi maître

Module: Informatique Industrielle 2

2. Les bus inter-composants, le bloc SPI du STM32

Figure 303. SPI block diagram



MISO : TX in slave mode
RX in master mode

MOSI : TX in master mode
RX in slave mode

SCK : Output pin (SPI masters)
input pin (slaves)

NSS : Slave select pin :
Select an individual slave
Synchronize data frame
Detect a conflict between
multiples masters

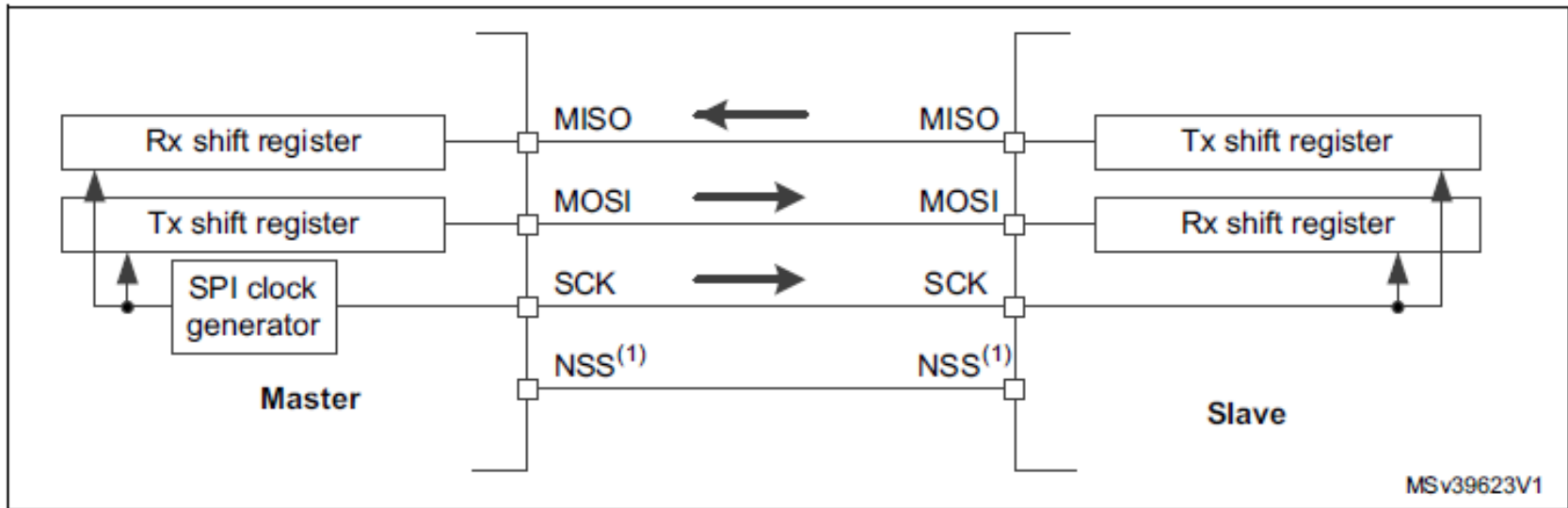
ST, STM32F446xx advanced ARM®-based 32-bit MCUs,
Reference manual, RM0390, July 2017

Module: Informatique Industrielle 2

2. Les bus inter-composants, le bloc SPI du STM32

Principe de la transmission SPI

Communication Full duplex 1 maître et 1 esclave

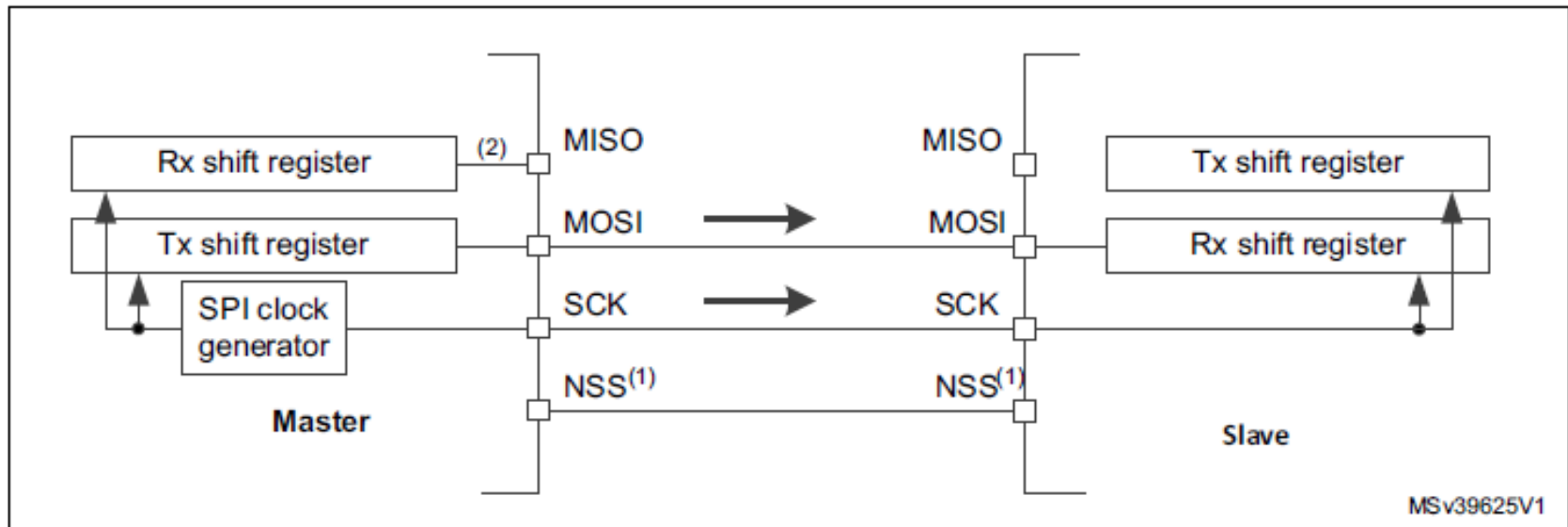


Module: Informatique Industrielle 2

2. Les bus inter-composants, le bloc SPI du STM32

Principe de la transmission SPI

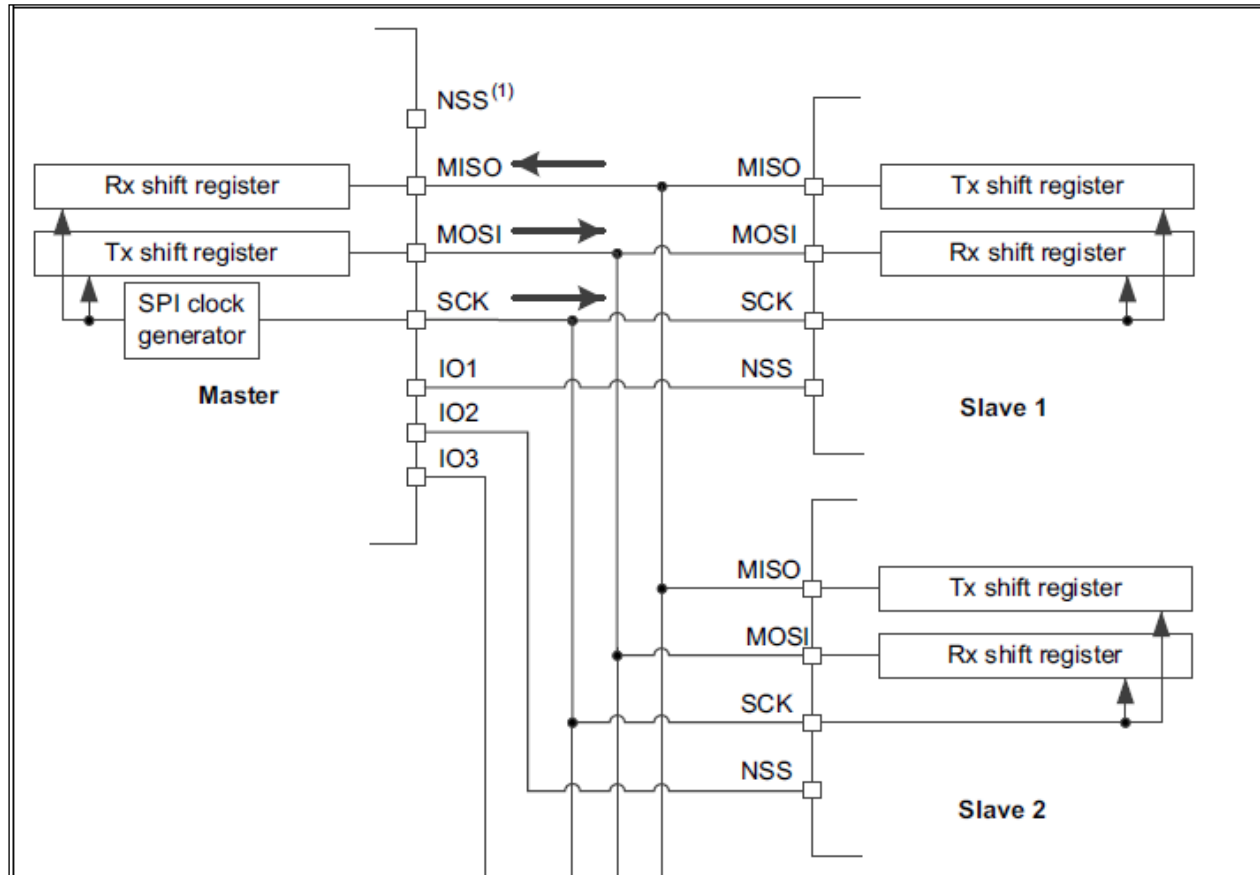
Communication simplex 1 maître et 1 esclave



Module: Informatique Industrielle 2

2. Les bus inter-composants, le bloc SPI du STM32

Communication maître et 2 .. n esclaves



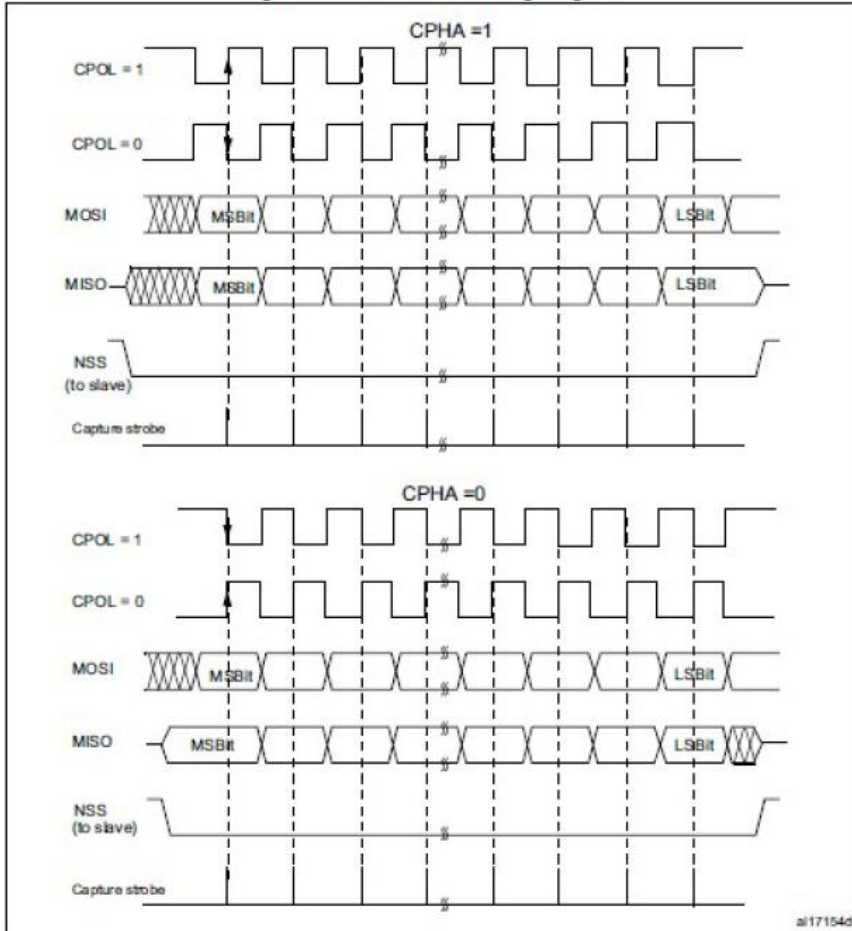
ST, STM32F446xx advanced ARM®-based 32-bit MCUs,
Reference manual, RM0390, July 2017

Module: Informatique Industrielle 2

2. Les bus inter-composants, le bloc SPI du STM32

Data clock timings

Figure 310. Data clock timing diagram



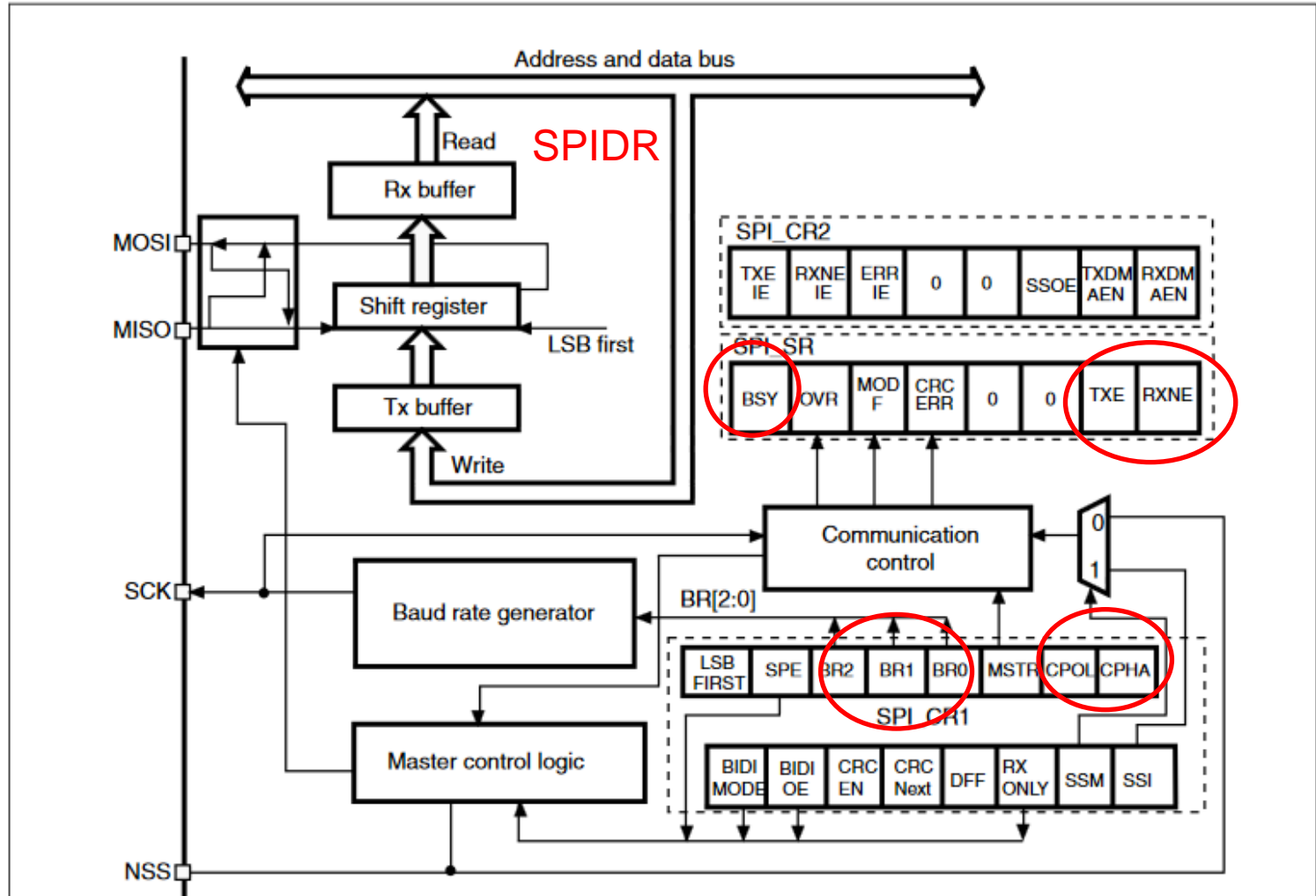
Pour être correctement échantillonnées, les données doivent respecter des timings par rapport au front de l'horloge.

- La donnée doit être présente avant l'arrivée du front d'horloge : on parle de temps de setup (T_{su})
- La donnée doit être maintenue un certain temps après l'arrivée du front d'horloge: on parle de temps de hold (T_{hd})
- Le composant slave impose les contraintes que le composant master devra respecter

→ C'est la raison pour laquelle les contrôleurs de SPI bus sont configurables

2. Les bus inter-composants, le bloc SPI du STM32

Figure 246. SPI block diagram



Module: Informatique Industrielle 2

2. Les bus inter-composants, le bloc SPI du STM32

26.3.12 SPI status flags

Three status flags are provided for the application to completely monitor the state of the SPI bus.

Tx buffer empty flag (TXE)

When it is set, the TXE flag indicates that the Tx buffer is empty and that the next data to be transmitted can be loaded into the buffer. The TXE flag is cleared by writing to the SPI_DR register.

Rx buffer not empty (RXNE)

When set, the RXNE flag indicates that there are valid received data in the Rx buffer. It is cleared by reading from the SPI_DR register.

Busy flag (BSY)

The BSY flag is set and cleared by hardware (writing to this flag has no effect).

When BSY is set, it indicates that a data transfer is in progress on the SPI (the SPI bus is busy). There is one exception in master bidirectional receive mode (MSTR=1 and BDM=1 and BDOE=0) where the BSY flag is kept low during reception.

The BSY flag can be used in certain modes to detect the end of a transfer, thus preventing corruption of the last transfer when the SPI peripheral clock is disabled before entering a low-power mode or an NSS pulse end is handled by software.

The BSY flag is also useful for preventing write collisions in a multimaster system.

The BSY flag is cleared under any one of the following conditions:

- When the SPI is correctly disabled
- When a fault is detected in Master mode (MODF bit set to 1)
- In Master mode, when it finishes a data transmission and no new data is ready to be sent
- In Slave mode, when the BSY flag is set to '0' for at least one SPI clock cycle between each data transfer.

Note: It is recommended to use always the TXE and RXNE flags (instead of the BSY flags) to handle data transmission or reception operations.