Serial Transmissions

René Beuchat

LAP/EPFL

rene.beuchat@epfl.ch

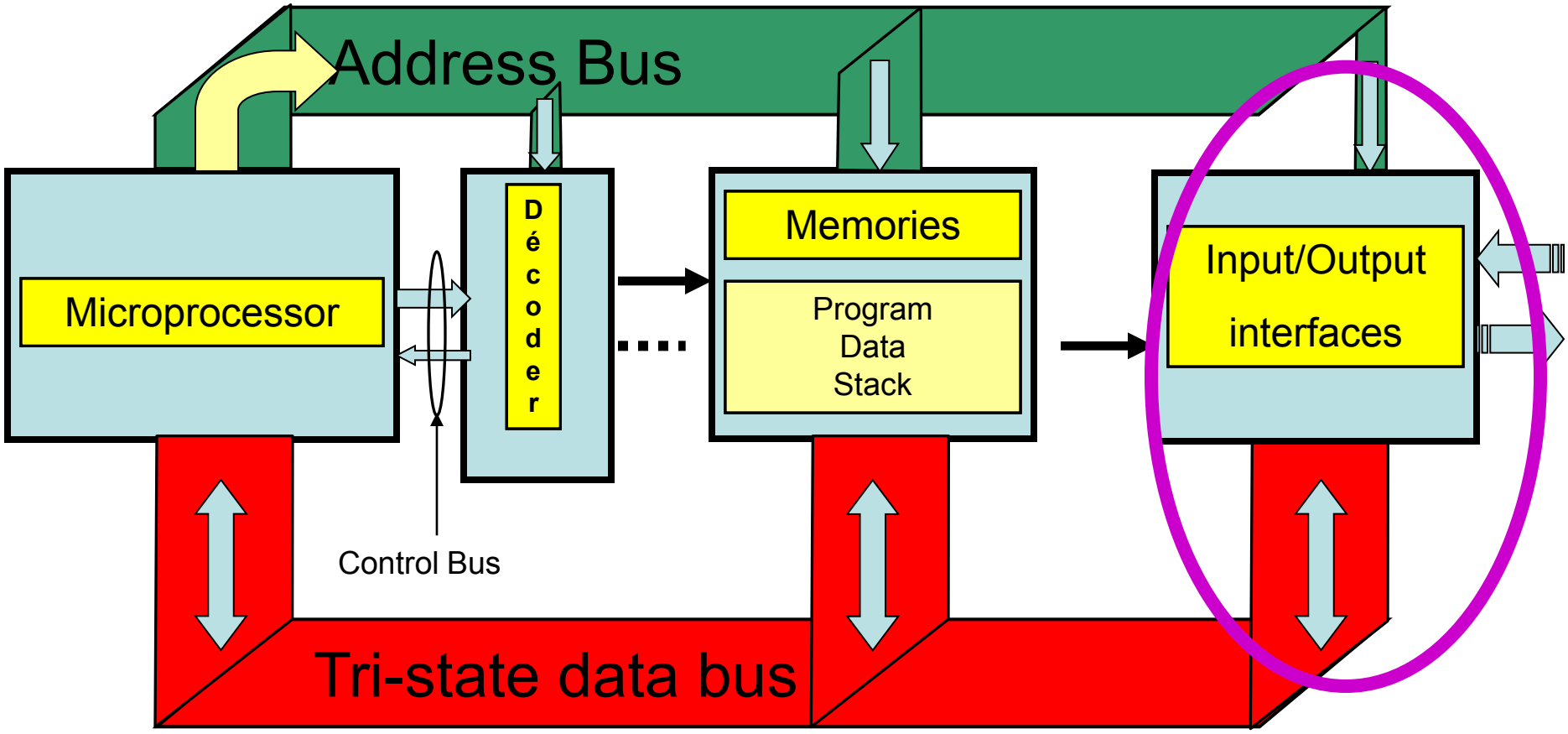
LSN/hepia/HESSO

rene.beuchat@hesge.ch

Outline

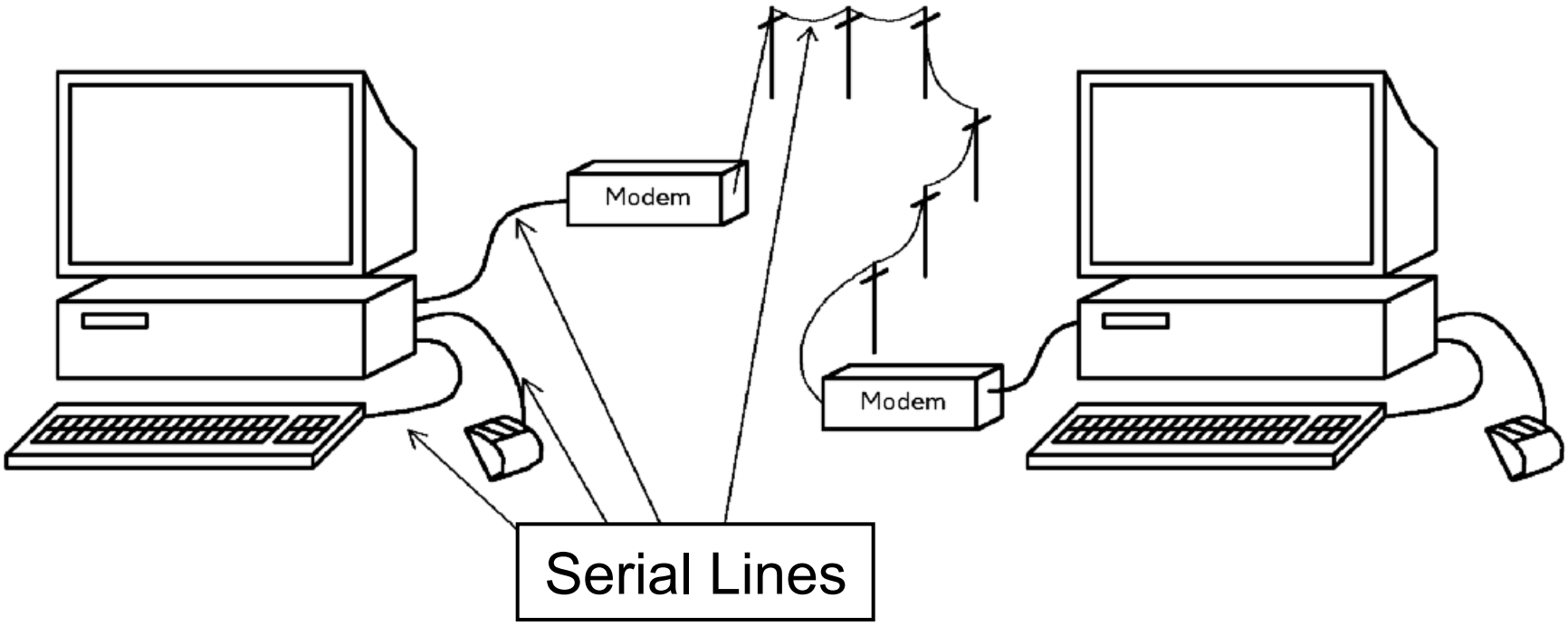
- General view of cable transmission
- Bit coding
- Bit/Byte/Message synchronization
- Examples of transmission systems:
  - Asynchronous RS-232
  - Synchronous packet oriented
- Some Communication protocols:
  - I2c, spi, one-wire
  - …

Computer architecture (general and simplified)



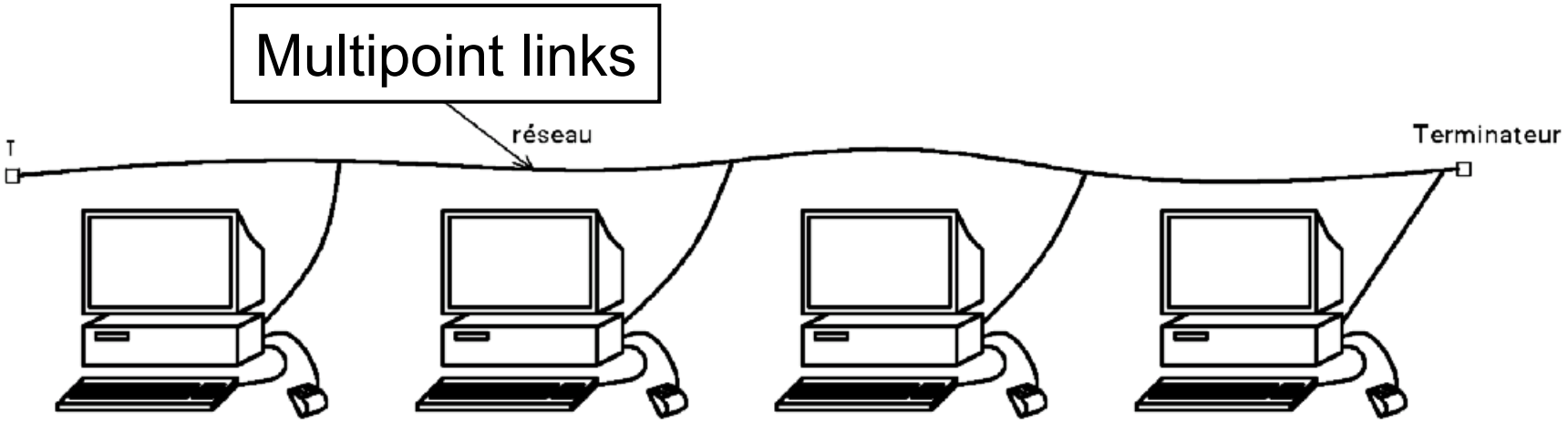
Computer serial communications

- Keyboard, mouse (PS2, USB)
- Modem (RS232, V24)
- Phone lines (analog, digital)



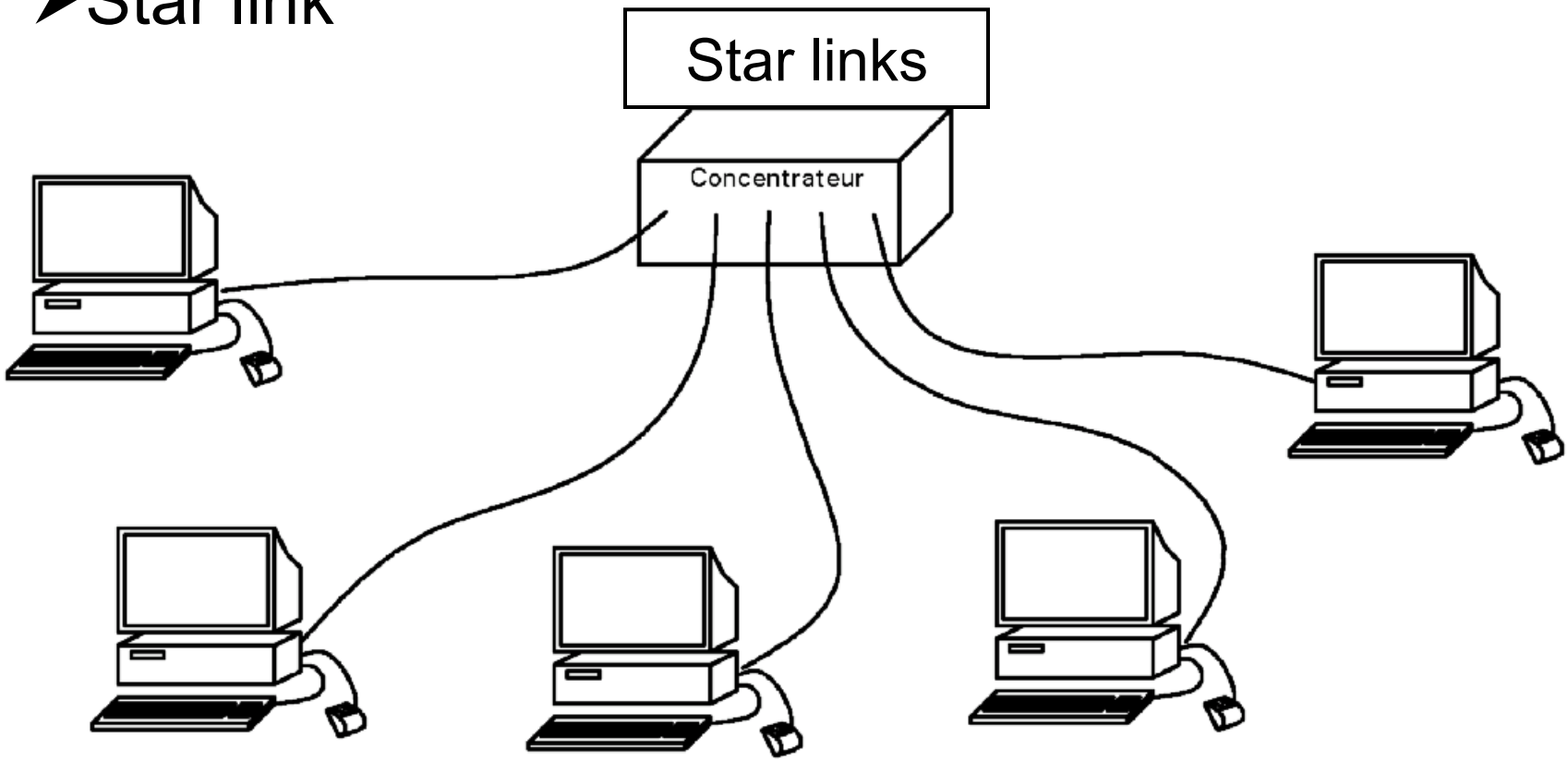
Computer serial communications

- Multipoint links (i.e. Ethernet 10 base2)
- Same cable for multiple devices



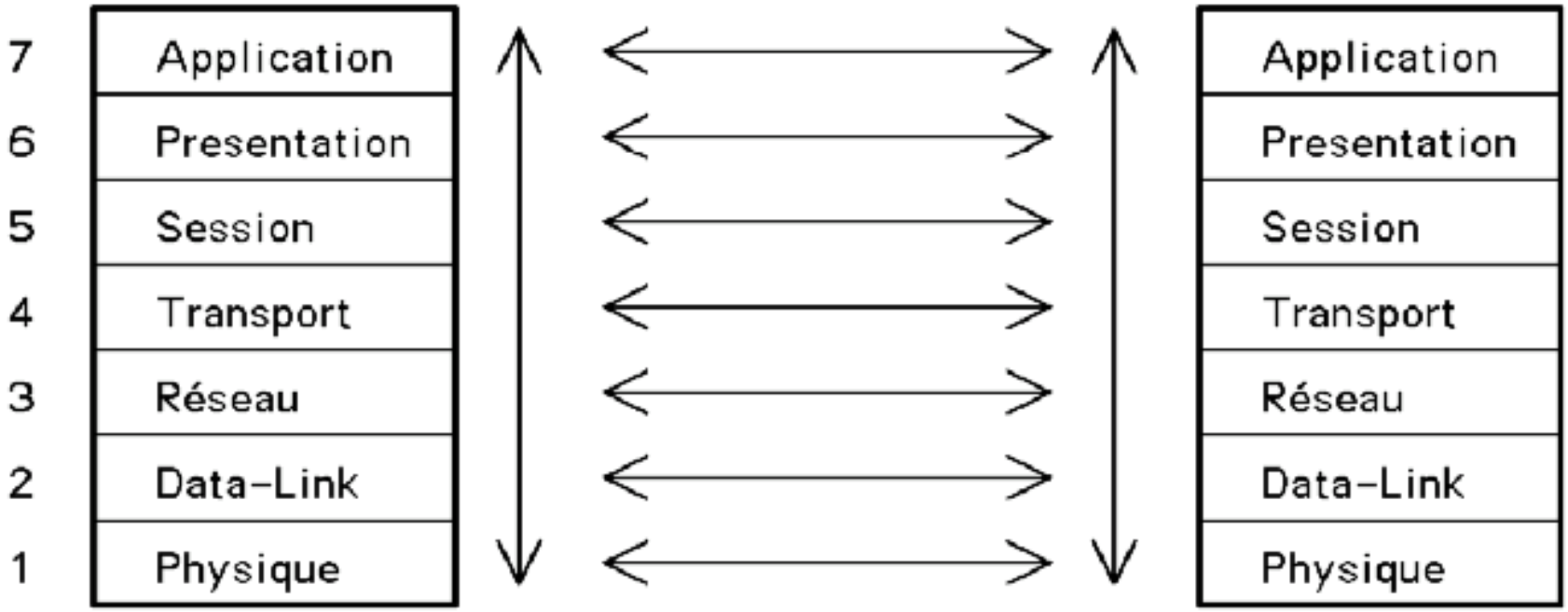
Computer serial communications

- Point to point links (Ethernet 10/100 Twisted pair)
- Star link



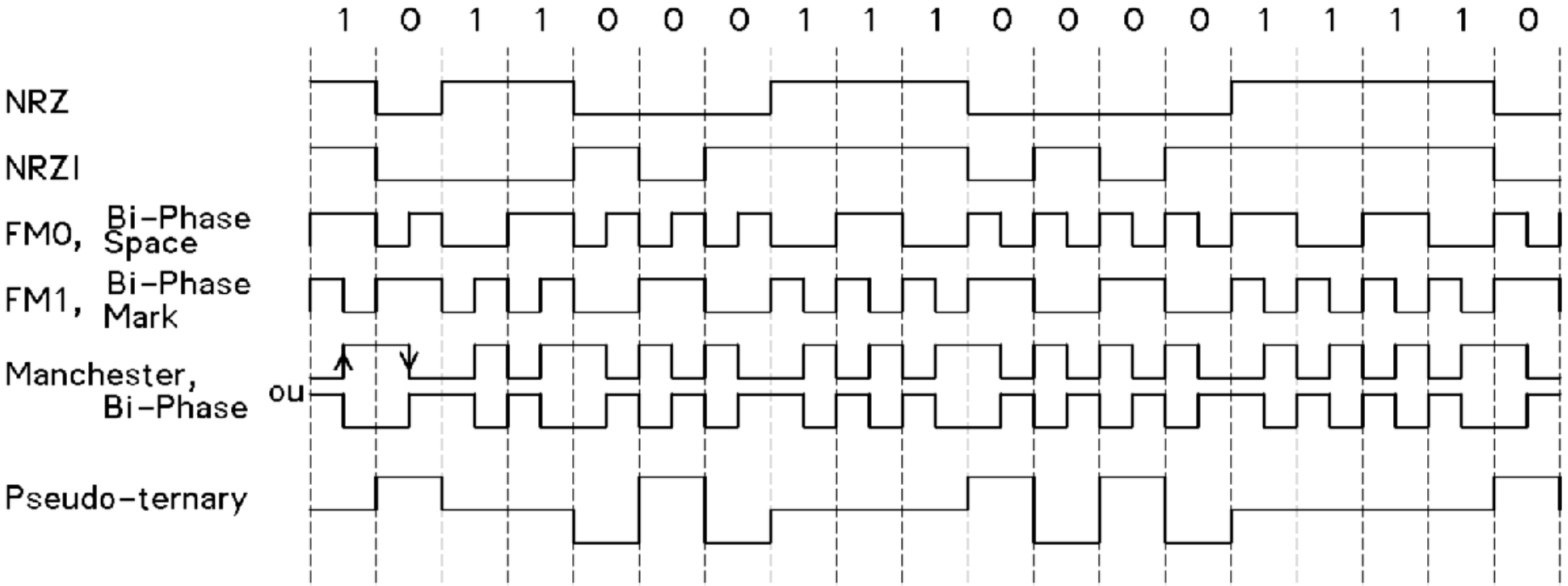
7 levels OSI of ISO

- Open System Interconnection
- International Standard Organization



Bits encoding

Bit coding examples



Bits encoding

Name	'0'	'1'	Synchro
NRZ	Low Voltage/courant	High Voltage/courant	No
NRZI	Transition at begin of bit	NO transition	No
FM0	Transition at center of bit	NO transition At center of bit	Transition between bits
FM1	NO transition At center of bit	Transition at center of bit	Transition between bits
Manchester Bi-phase	0→1 At center	1→0 At center	Transition at center of bits
Pseudo-ternary	Alternatively Pos / Neg level	Gnd	No

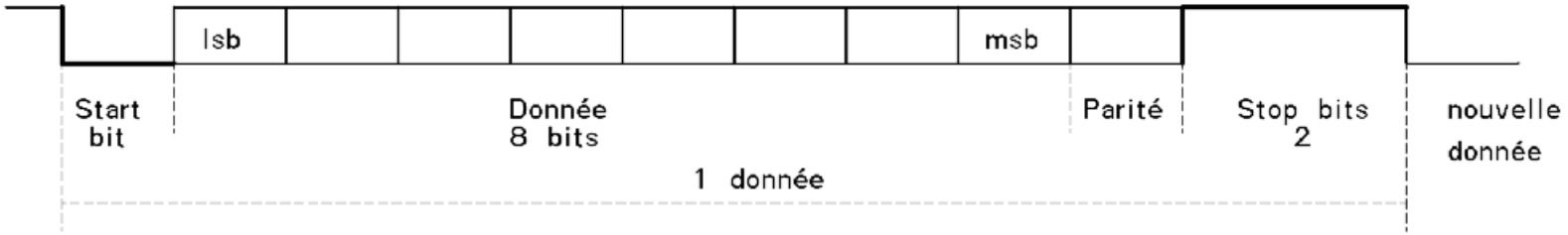
Synchronizations

Problems of synchronizations :

- At the bit level
- At the word level (bits assembly)
- At the message level (word assembly)

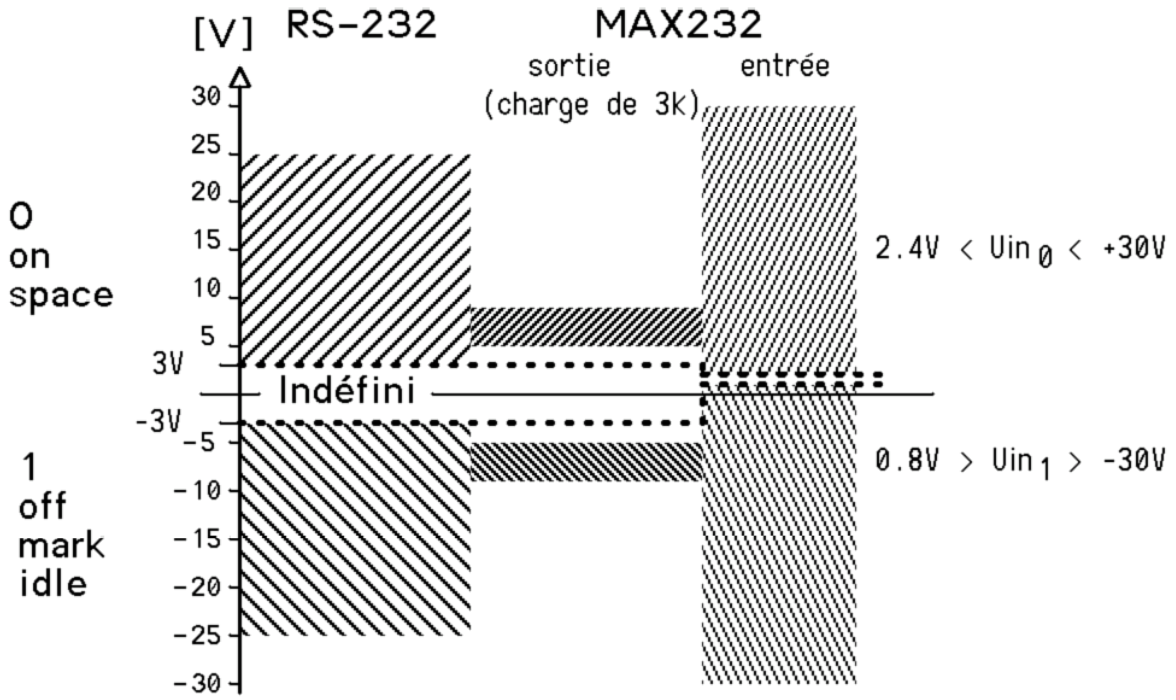
- Transmission speed

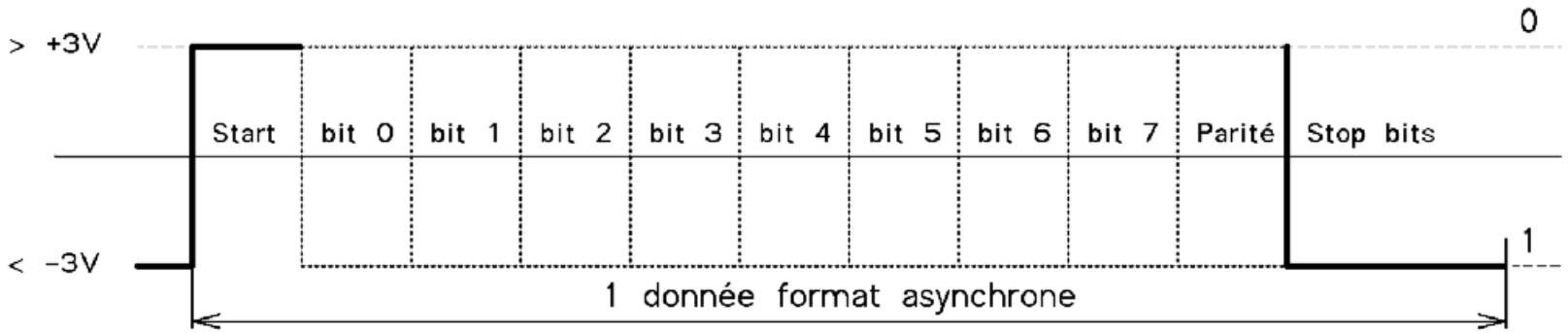
Asynchronous Transmission



- Word Synchronization :
  - start bit
  - data (5..8 bits)
  - parity (optional, even/odd)
  - stop bit (1, 1.5, 2 at minimum)
- Transmission speed by EXTERNAL choice

Asynchronous Transmission, RS-232

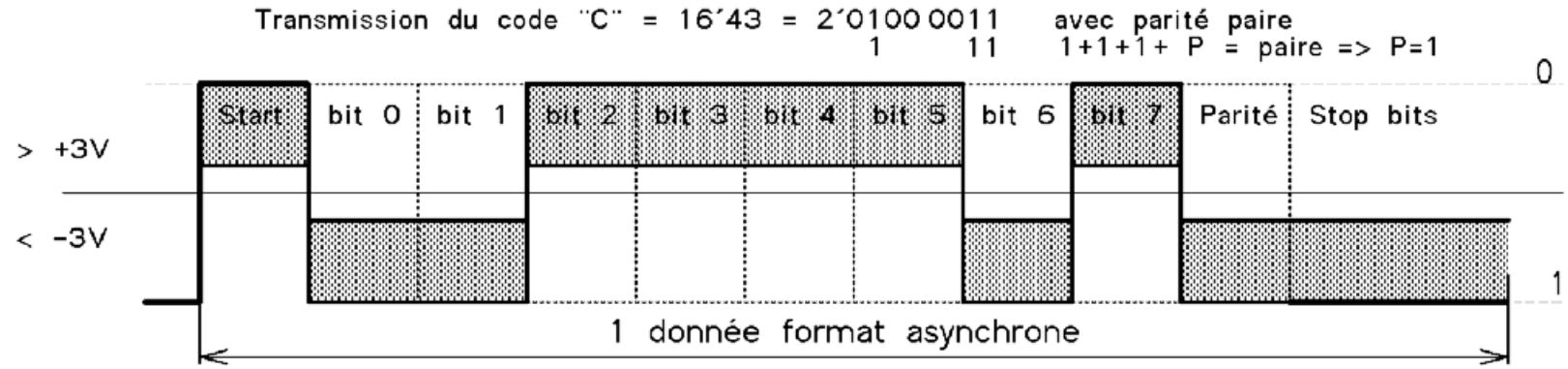




Example, RS-232

Transmission of code "C", even parity :

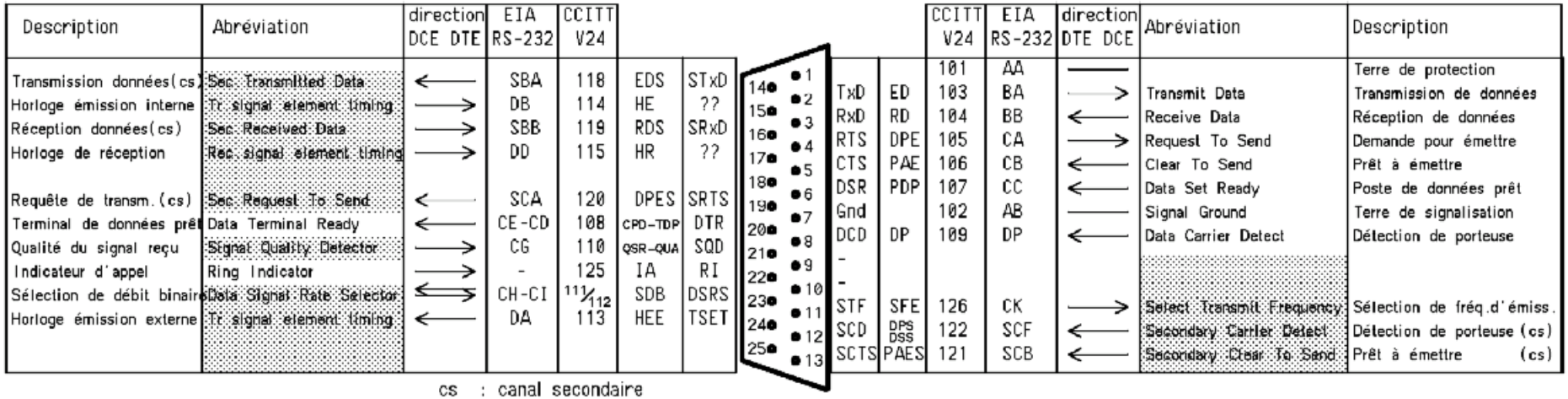
- ASCII C → 0x43 → b' 0100 0011
- LSb (bit 0) first generally)
- '0' → > +3V
- '1' → < -3V



Example, RS-232

- 3 signaux sont indispensables pour la communication:
  - **Gnd**: Terre de transmission (pin 7)
  - **TxD**: transmission de données
  - **RxD**: réception de données
- Viennent ensuite 2 signaux de gestion du contrôle de flux de transmision entre l´émetteur et le récepteur:
  - **RTS**: *Requets To Send*, l´émetteur désire émettre
  - **CTS**: *Clear To Send*, le récepteur autorise l´émetteur à émettre
- Un signal est utilisé pour indiquer que la communication est établie:
  - **DCD**: *Data Carrier Detect*, la porteuse est valide
- 2 signaux indiquent que les équipements sont prêts pour communiquer:
  - **DTR**: *Data Terminal Ready*, le terminal est prêt
  - **DSR**: *Data Set Ready*, le modem est prêt
- Un dernier signal utilisé avec certains modems:
  - **RI**: *Ring Indicator*, sonnerie

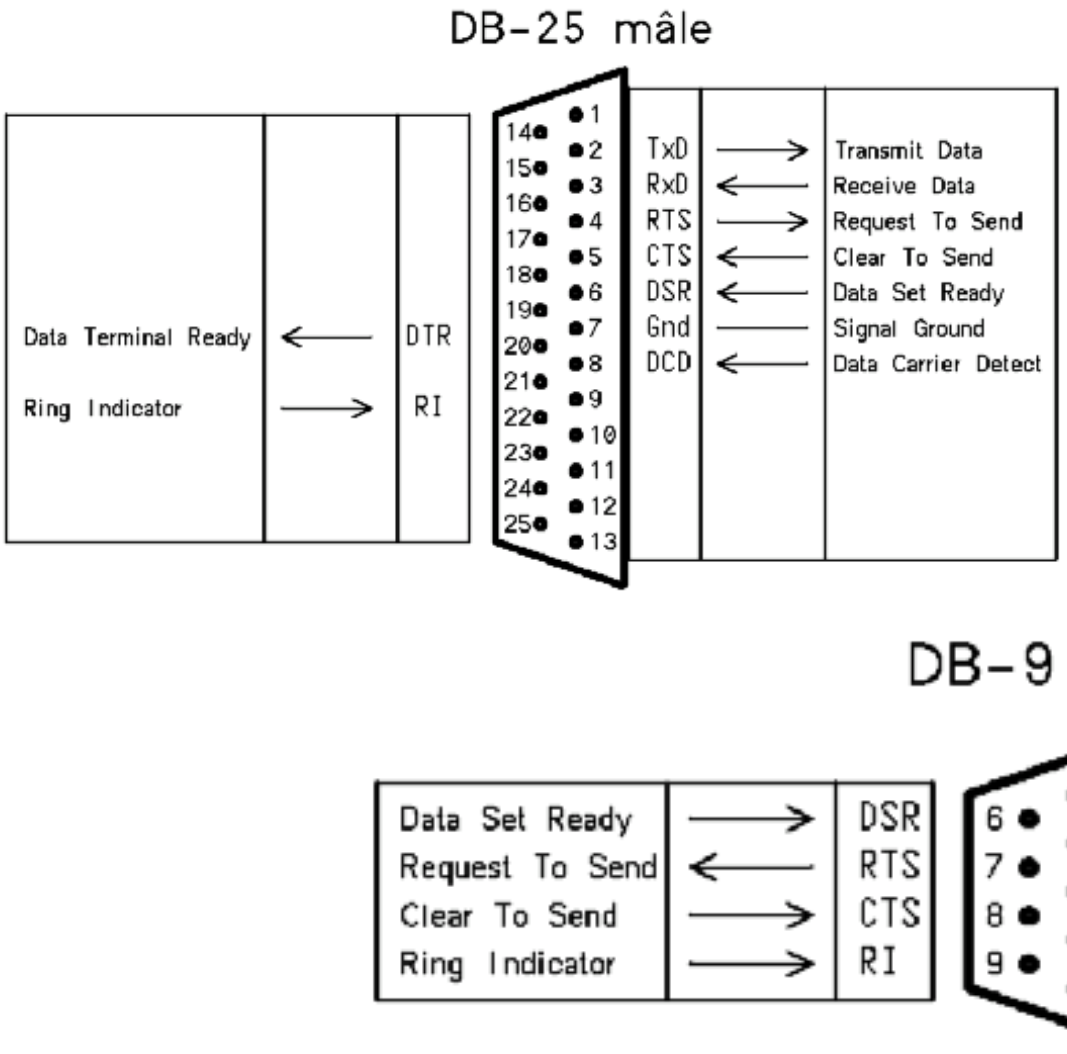
Connector RS-232, 25 pins



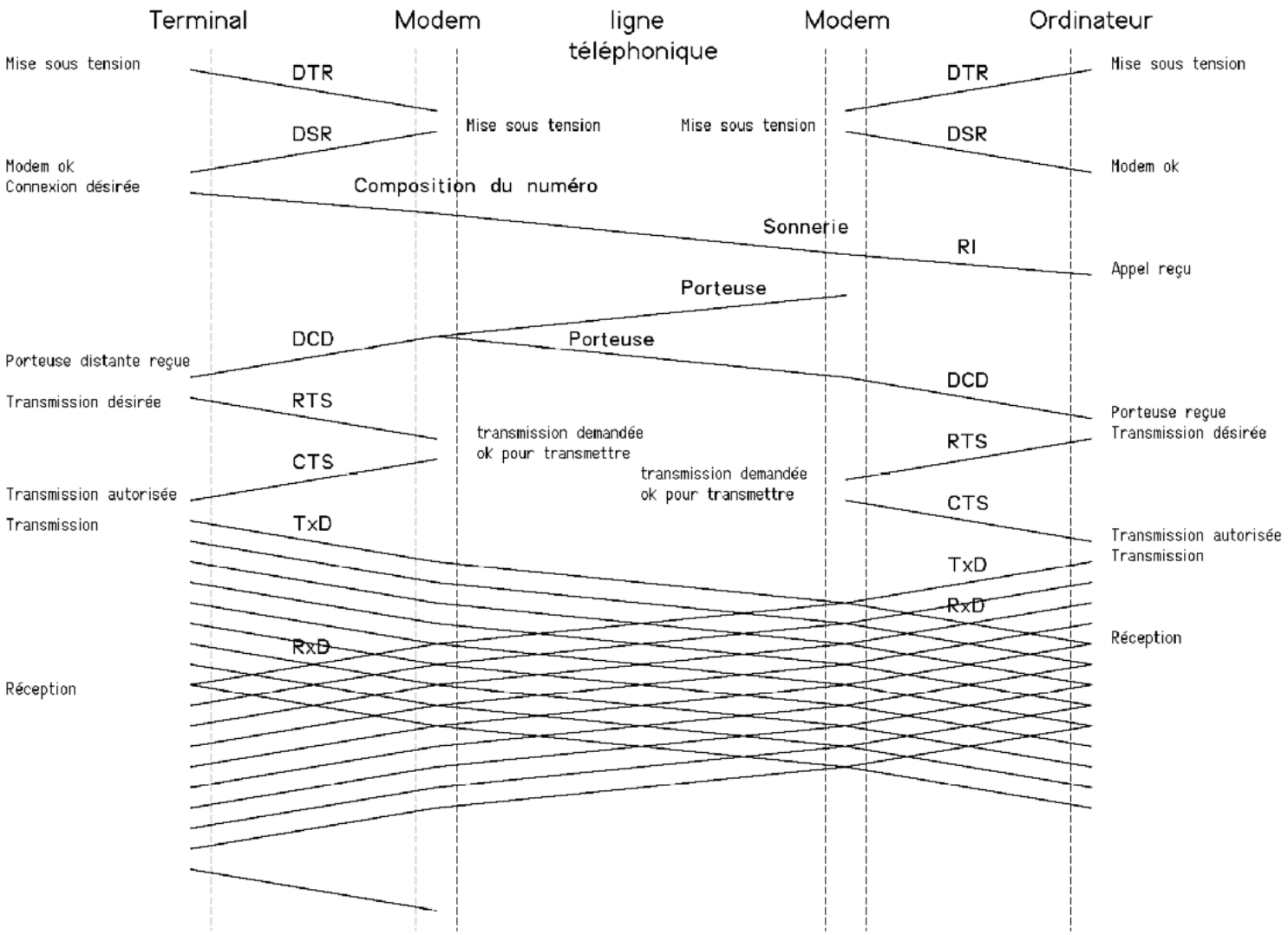
Description	Abréviation	direction DCE DTE	EIA RS-232	CCITT V24				CCITT V24	EIA RS-232	direction DTE DCE	Abréviation	Description
Transmission données(cs)	Sec. Transmitted Data	←	SBA	118	EDS	STxD	1	101	AA	—		Terre de protection
Horloge émission interne	Tr signal element timing	→	DB	114	HE	??	2	103	BA	→	Transmit Data	Transmission de données
Réception données(cs)	Sec. Received Data	→	SBB	119	RDS	SRxD	3	104	BB	←	Receive Data	Réception de données
Horloge de réception	Rec. signal element timing	→	DD	115	HR	??	4	105	CA	→	Request To Send	Demande pour émettre
Requête de transm.(cs)	Sec. Request To Send	←	SCA	120	DPES	SRTS	5	106	CB	←	Clear To Send	Prêt à émettre
Terminal de données prêt	Data Terminal Ready	←	CE-CD	108	CPD-TDP	DTR	6	107	CC	←	Data Set Ready	Poste de données prêt
Qualité du signal reçu	Signal Quality Detector	→	CG	110	QSR-QUA	SQD	7	102	AB	—	Signal Ground	Terre de signalisation
Indicateur d'appel	Ring Indicator	→	-	125	IA	RI	8	109	DP	←	Data Carrier Detect	Détection de porteuse
Sélection de débit binaire	Data Signal Rate Selector	⇄	CH-CI	111/112	SDB	DSRS	9					
Horloge émission externe	Tr signal element timing	←	DA	113	HEE	TSET	10					
							11					
							12					
							13					
							14	TxD	ED	103	BA	→	Transmit Data	Transmission de données
							15	RxD	RD	104	BB	←	Receive Data	Réception de données
							16	RTS	DPE	105	CA	→	Request To Send	Demande pour émettre
							17	CTS	PAE	106	CB	←	Clear To Send	Prêt à émettre
							18	DSR	PDP	107	CC	←	Data Set Ready	Poste de données prêt
							19	Gnd		102	AB	—	Signal Ground	Terre de signalisation
							20	DCD	DP	109	DP	←	Data Carrier Detect	Détection de porteuse
							21	-						
							22	-						
							23	STF	SFE	126	CK	→	Select Transmit Frequency	Sélection de fréq.d'émiss.
							24	SCD	DPS DSS	122	SCF	←	Secondary Carrier Detect	Détection de porteuse (cs)
							25	SCTS	PAES	121	SCB	←	Secondary Clear To Send	Prêt à émettre (cs)

cs : canal secondaire

Connector RS-232, 9 pins

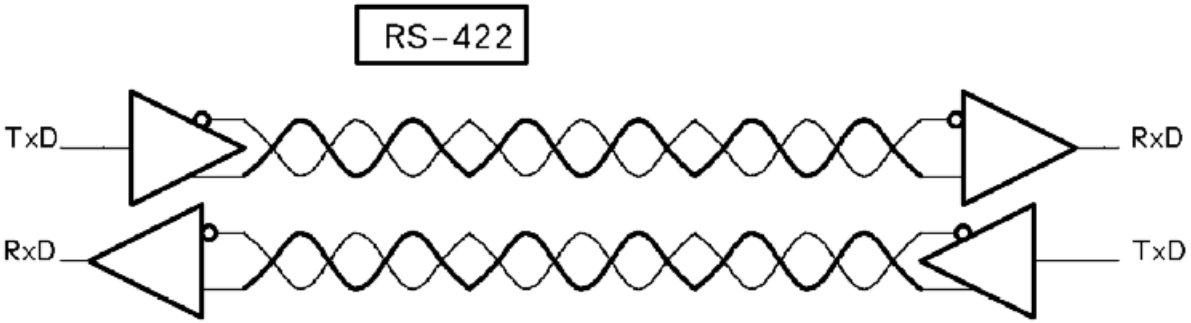


RS-232 data & control flow

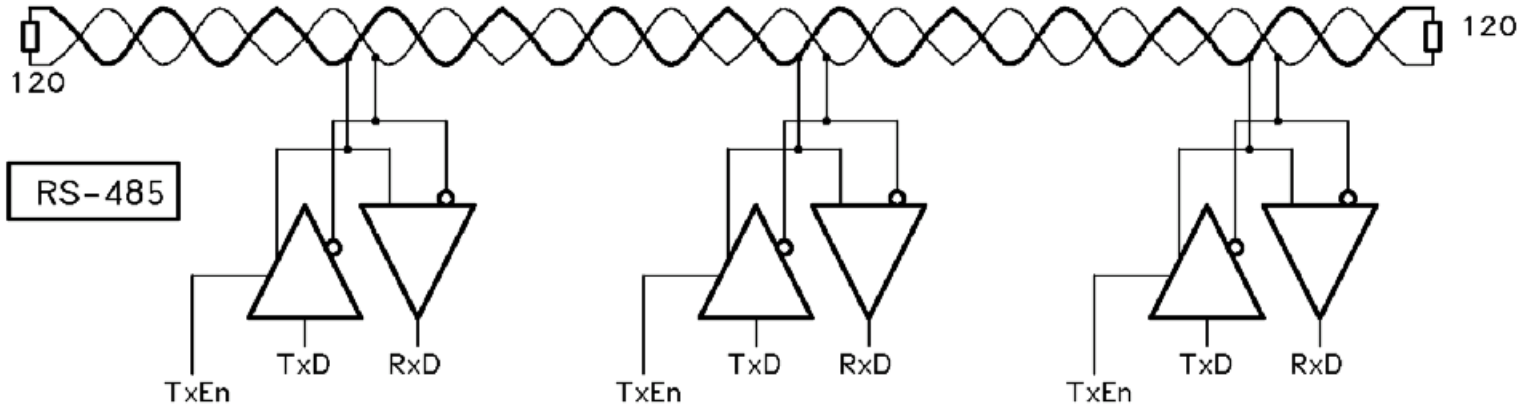


RS422- RS485, physical connection

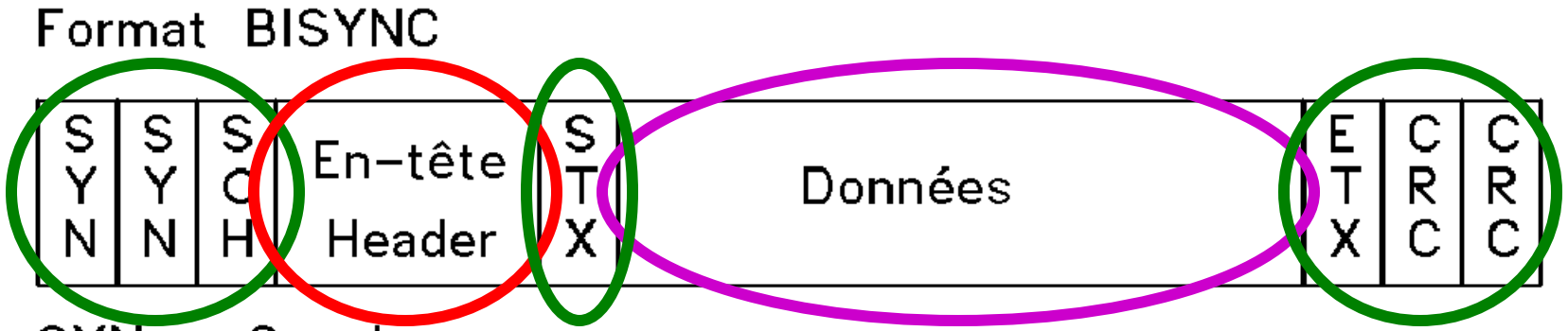
Differential Transmission, point to point



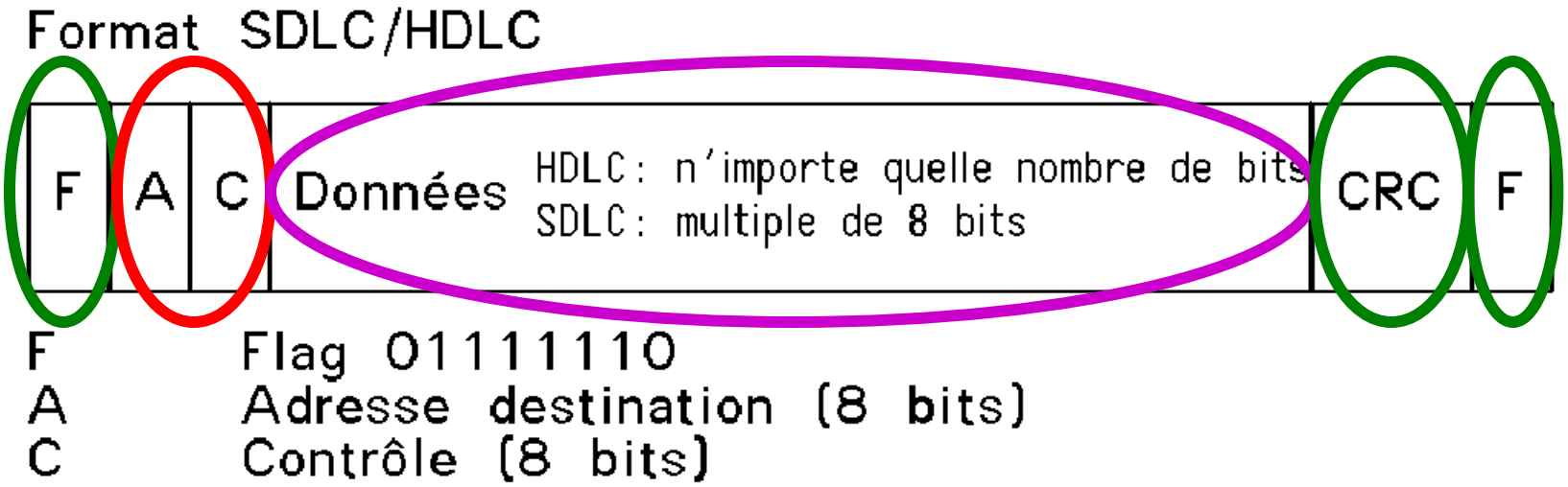
Differential Transmission, multipoints



Packet Transmission (1), BISYNC



Packet Transmission (2), HDLC

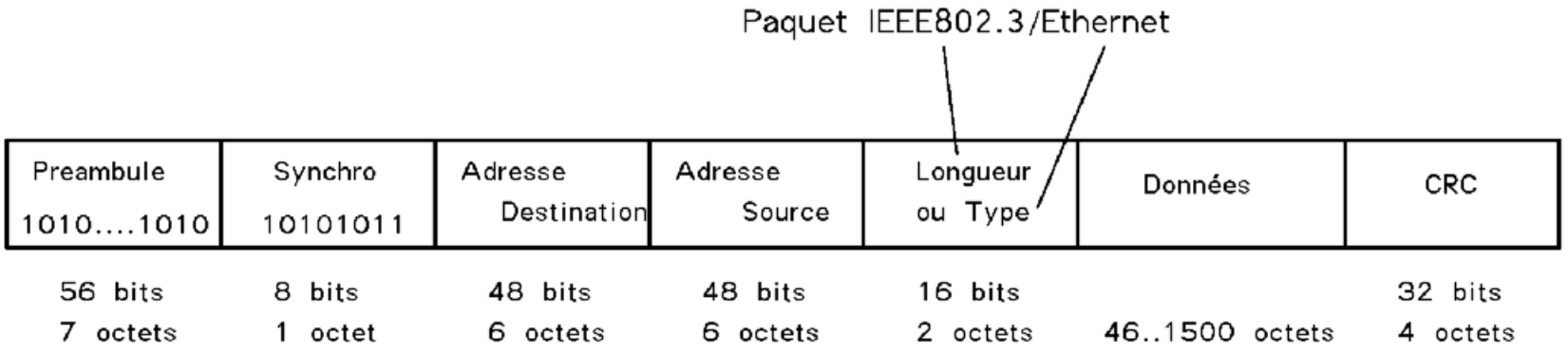


Data payload:

- HDLC : bit granularity
- SDLC : byte granularity (multiple of 8 bit)

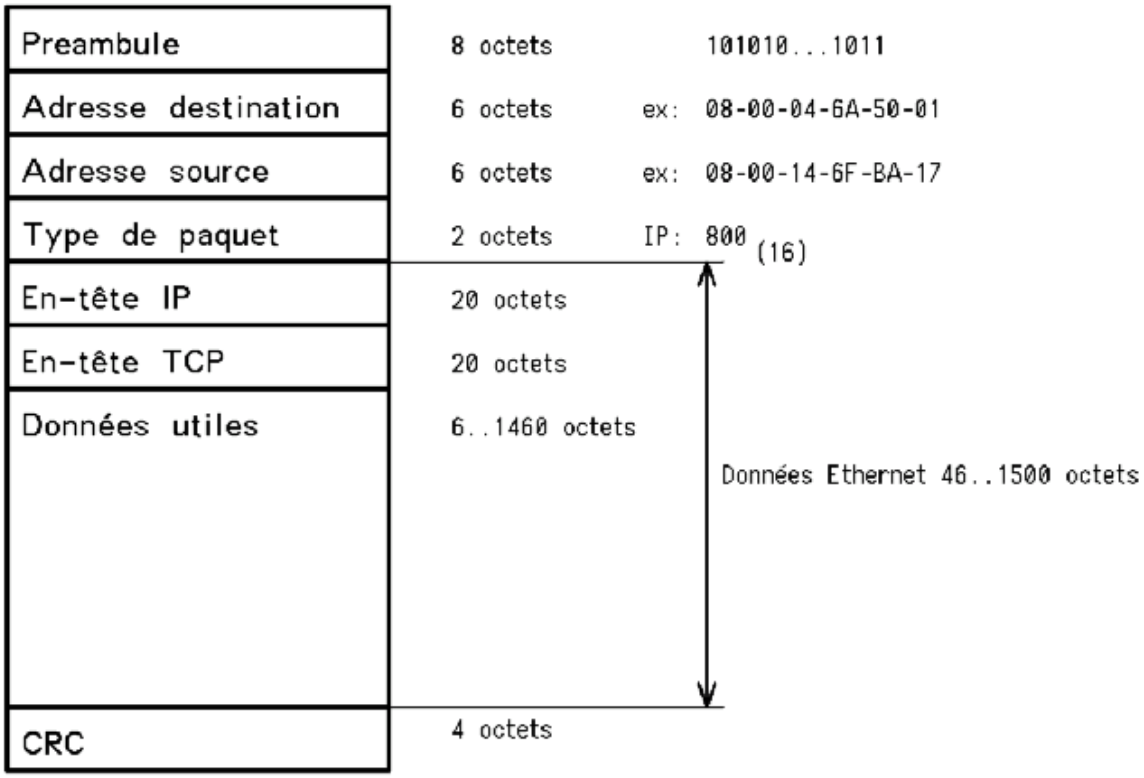
Ethernet/IEEE 802.3

- Packet Transmission
- Clock include/coded with the data →
  - Manchester Coding



Ethernet/IEEE 802.3

- TCP/IP



Paquet avec TCP/IP

Serial Interfaces

Bus I^2C

Bus SPI

Bus 1-Wire

Synchronous Transmission

Clock and data transmission, ex :

- I^2C, packet transmission, start/stop bits
- SPI, word transmission (8 bits, QSPI 16 bits),
  - Sel signal for synchronization

Serial Interfaces

- Components examples :
  - Asynchronous Interface
    - UART/SCI Interface of microcontrollers
  - Circuit i-Wire, uLAN Memories Dallas/maxim uLAN
    - Serial Memories DS2224
  - Circuit I^2C I2C, example PCF8574 port //
    - Parallel Port extension PCF8574 (old components)

Serial Interfaces

Bus I^2C

SMBus

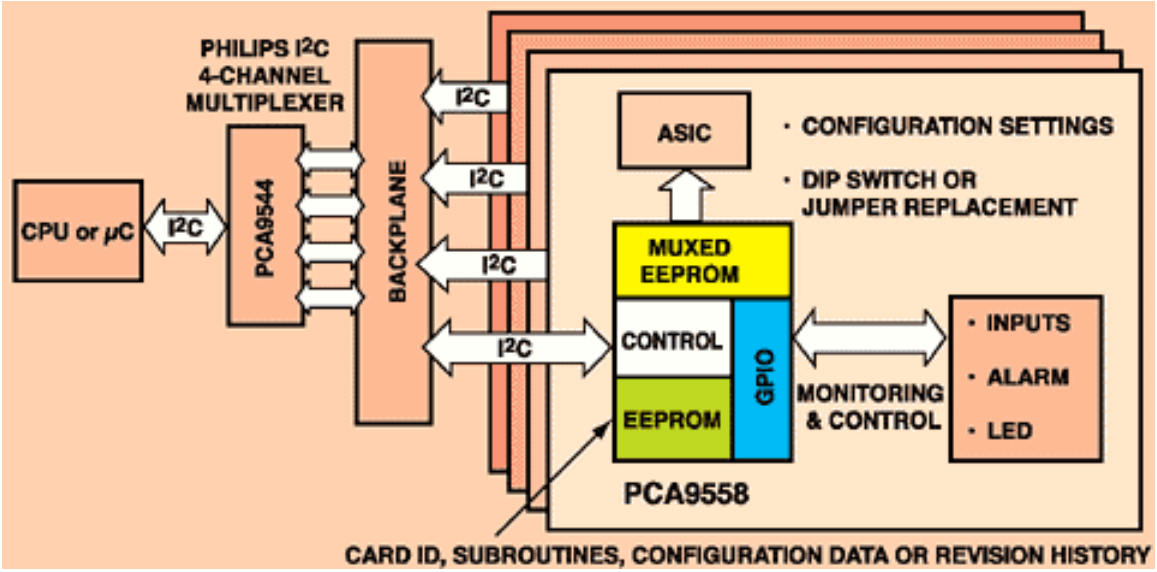
(System Management Bus)

Philips semiconductor

I²C

- Inter devices Bus
- Short distance (~1 m)
- Multi-masters
- Multi-slaves
- Clock provided by the Master
- Open Collector/drain Bus
- Transmission at
  - ➢100 kbits/s (normal), 400 kbits/s (fast)

I²C, example in a system



- Port I/O extension by simple serial bus : port //, converter A/D, D/A, serial memories, tuning, etc…
- Ex. of utilization : hi-fi, TV, etc…

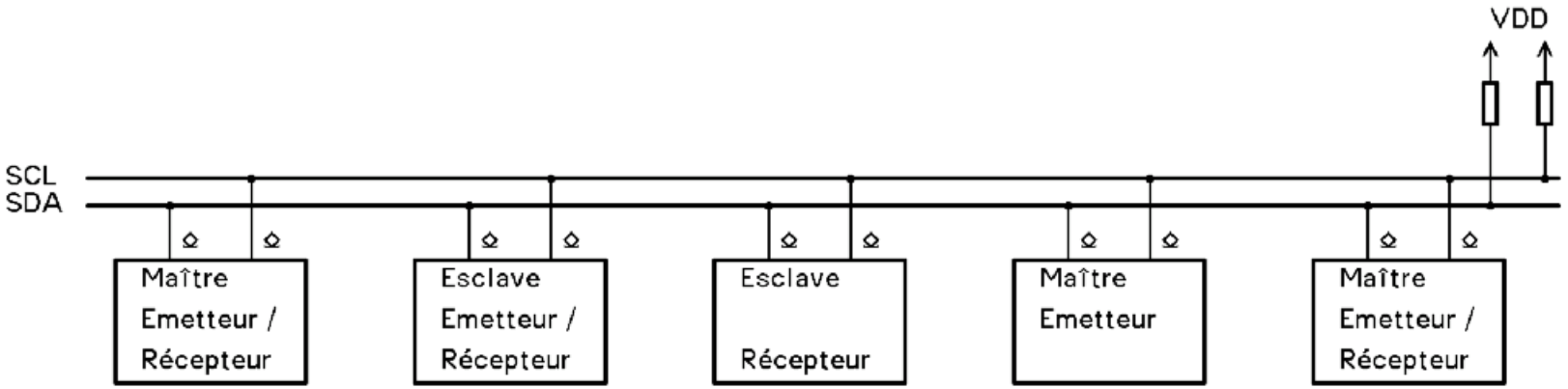
I²C, definitions

- *Master* :
  - Unit which send Clock, start and stop the transfer
  - Provide the slave address and direction of cycle (read/write)
- *Slave* :
  - Selected unit by a master
  - Receive the Address and direction of transfer
  - Acknowledge the address if it's selected
- *Multimaster system* :
  - System where multi master can start a transfer. One master is able to finish
- *Arbitration :*
  - Process to take the bus if multiple masters want to do at the same time

I²C, definitions

- *Emitter* :
  - It's the unit which **send** a data on the bus
  - Writing : the emitter is the **master**
  - Reading : the emitter is the **slave**
- *Receptor* :
  - It's the unit which **receive** a data on the bus
  - Writing : the emitter is the **slave**
  - Reading : the emitter is the **master**
- *Synchronization*
  - Method to synchronize the clock between many circuits

I²C, devices interconnection



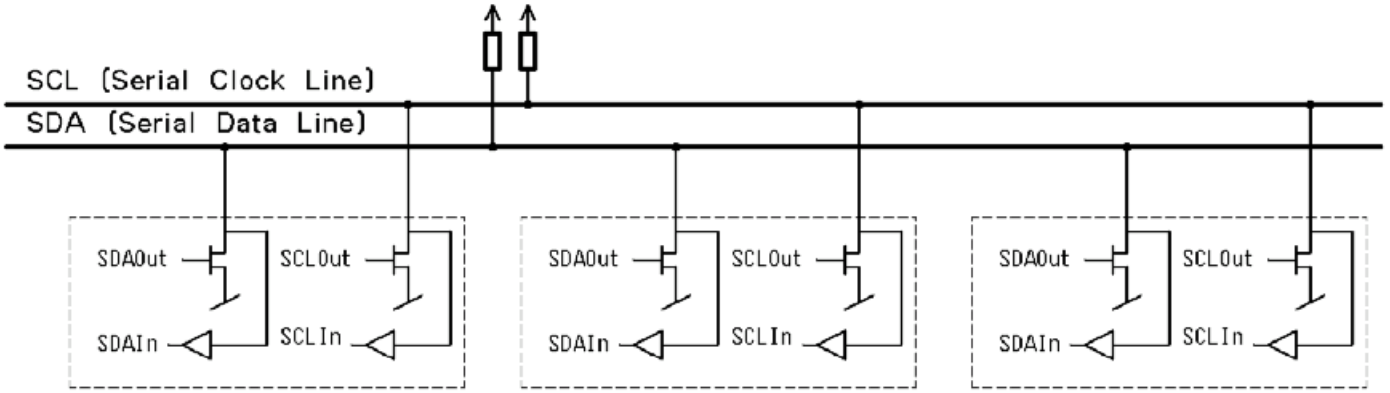
2 lines for everyone :

- **SCL** : Serial CLk
- **SDA** :Serial Data

Transmission :

- multipoint
- synchronous
- by packet

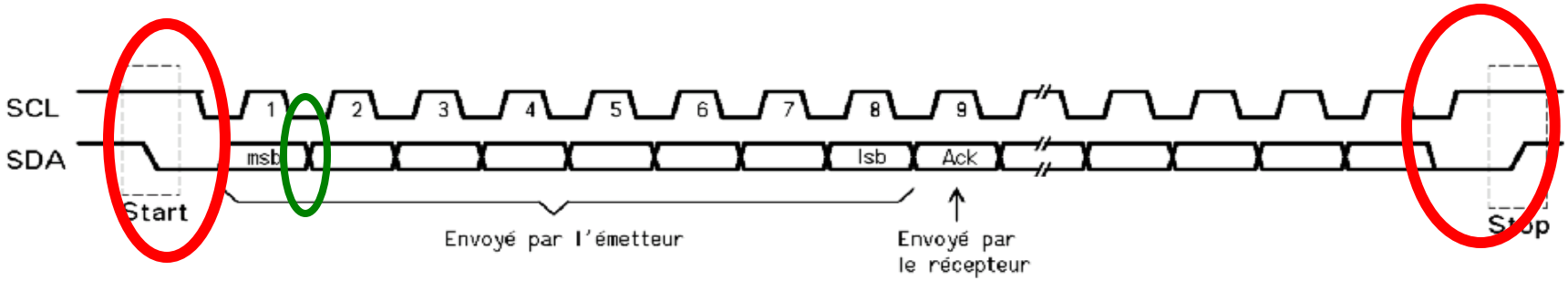
I²C, interface on the bus



- SCL: CLk
  - always send by the master
  - **can be** maintain at '0' by a slave to slow down the transmission speed
- SDA: Data

- bidirectional lines, **open collector**

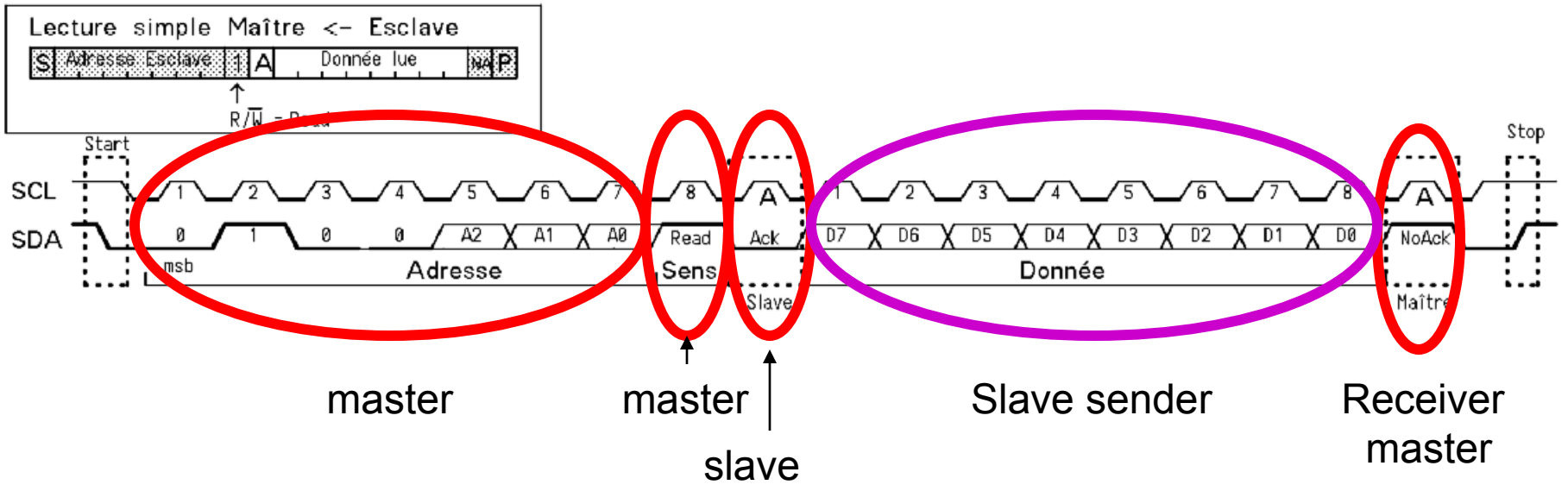
I²C, packet transmission START/STOP



- Idle: lines at 'H'
- Start : SDA 'H' → '0', then SCL 'H' → '0'
- Stop : SCL '0' → 'H', then SDA '0' → 'H'
- Data changes when SCL at '0'

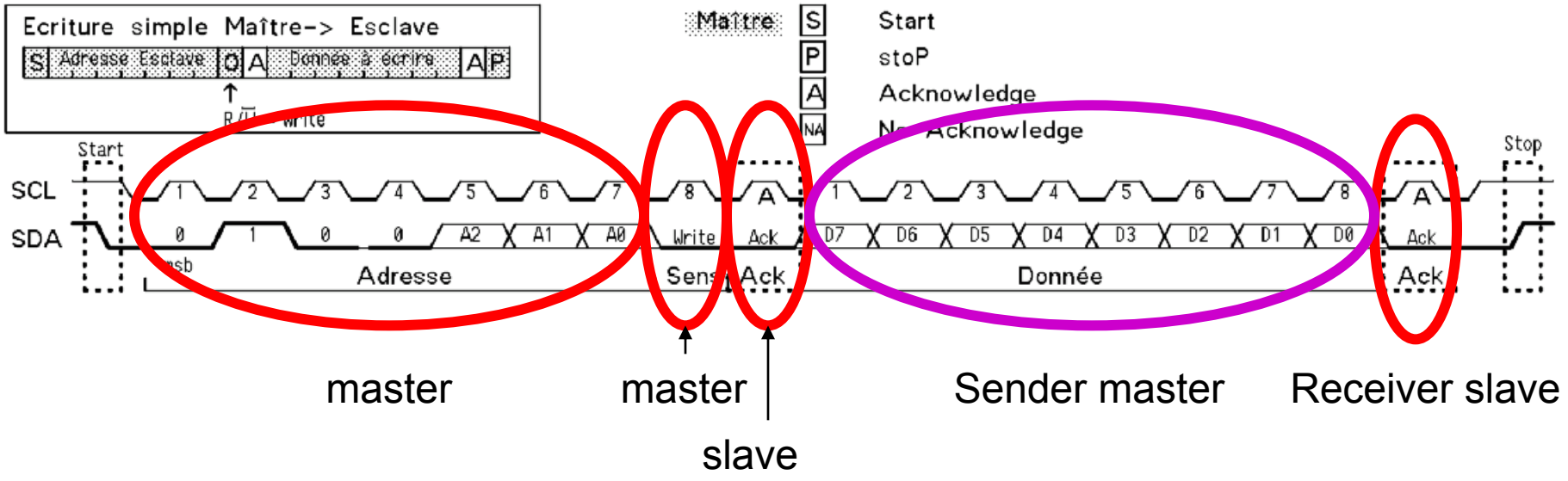
I²C reading

- 1er byte (master):
  - destination address 7 bits
  - direction of transfer 1 bits ('H')
- Acknowledge (slave): '0'

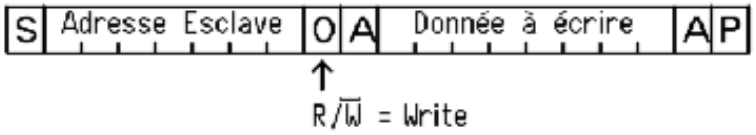


I²C writing

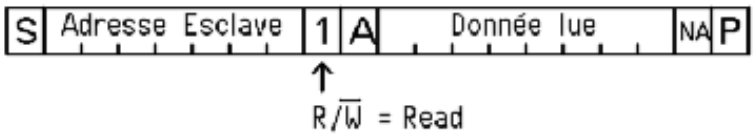
- 1er byte (master):
  - destination address 7 bits
  - direction of transfer 1 bits ('0')
- Acknowledge (slave): '0'



I²C other accesses



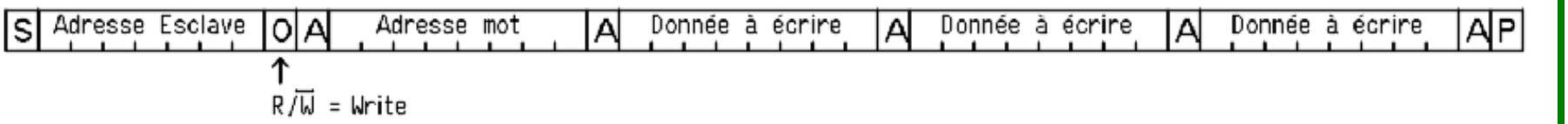
Ecriture simple Maître-> Esclave



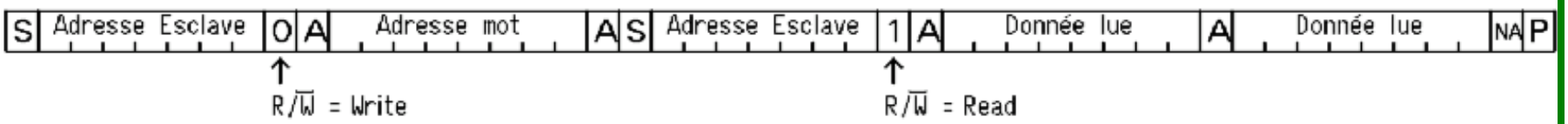
Lecture simple Maître <- Esclave

S Start
P stoP
A Acknowledge
NA No Acknowledge

R/W Read si 1, Write si 0

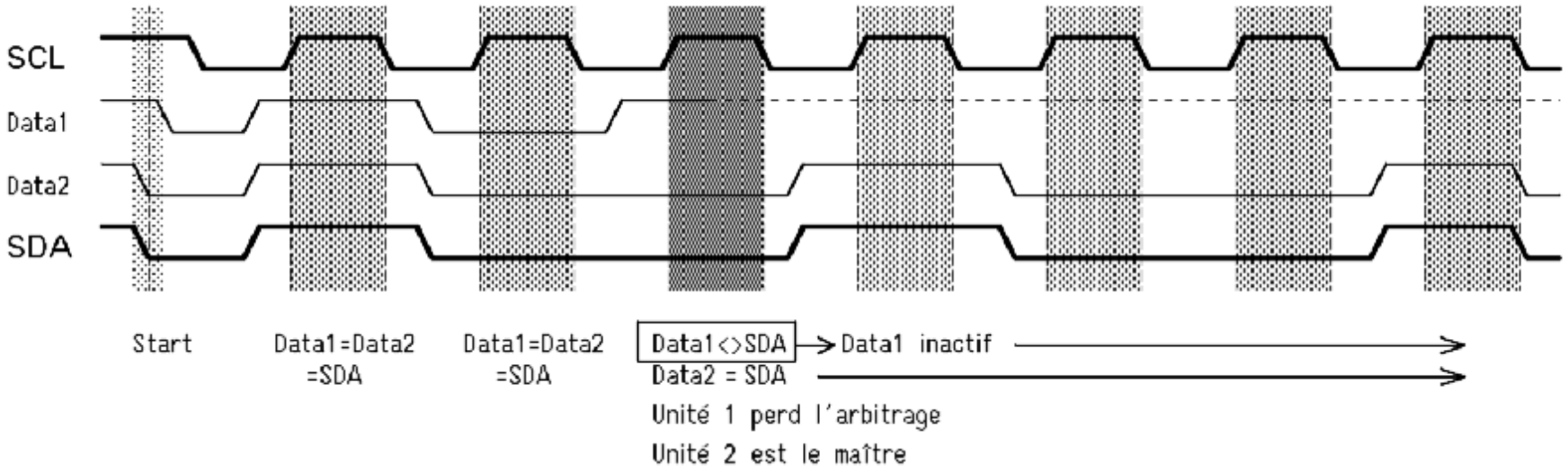


Ecriture multiple avec auto-incrémentation des adresses destination



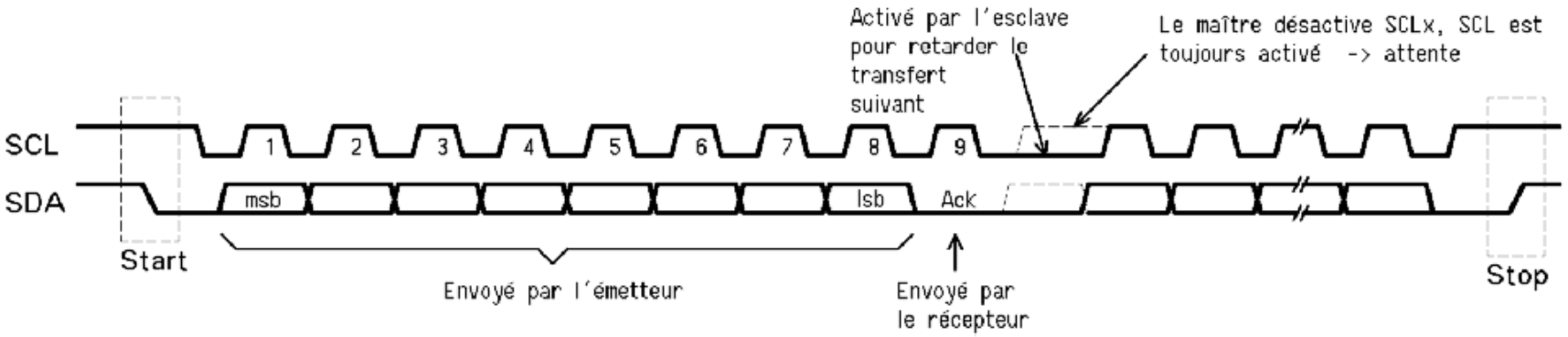
Lecture multiple avec auto-incrémentation des adresses destination

I²C multi-master arbitration



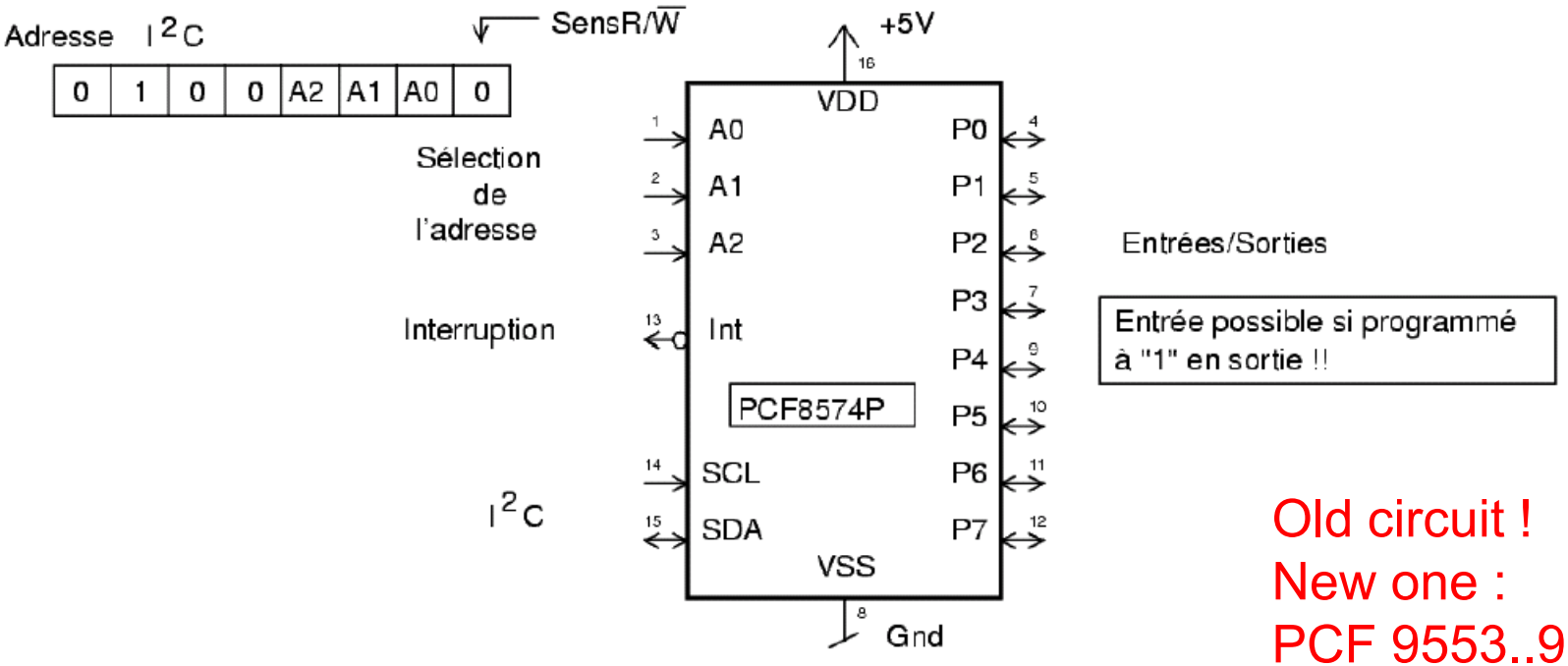
- Open collector and read back of transmitted state:
  - SCL activated by all masters → '0'
  - Destination address on SDA (msb → lsb)
  - When they send à '1', the line ('1'→ 'H') → 'H'
  - While the master see the same state as they send, they continue. '0' wins on 'H'.

I²C Clock speed down



Maintaining the line at '0', a slave can slow down the transmission speed.
Not always supported by all masters !

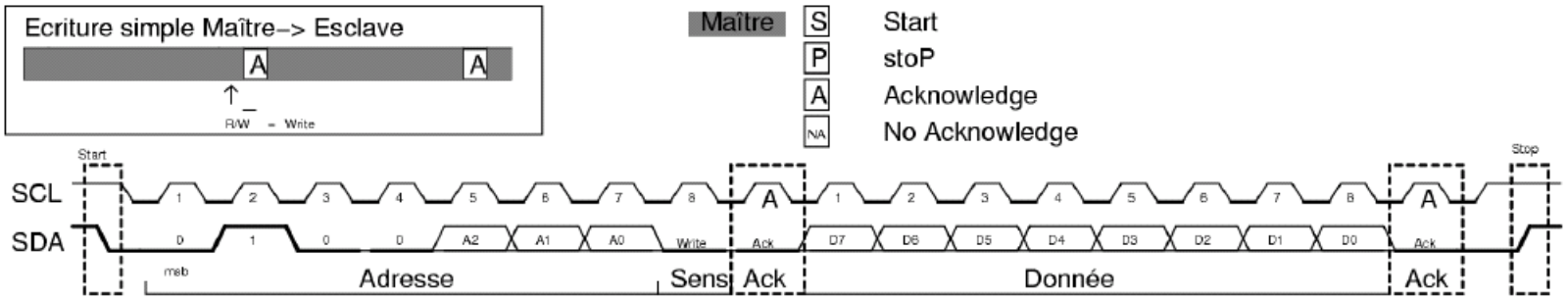
I²C, example PCF8574 port //



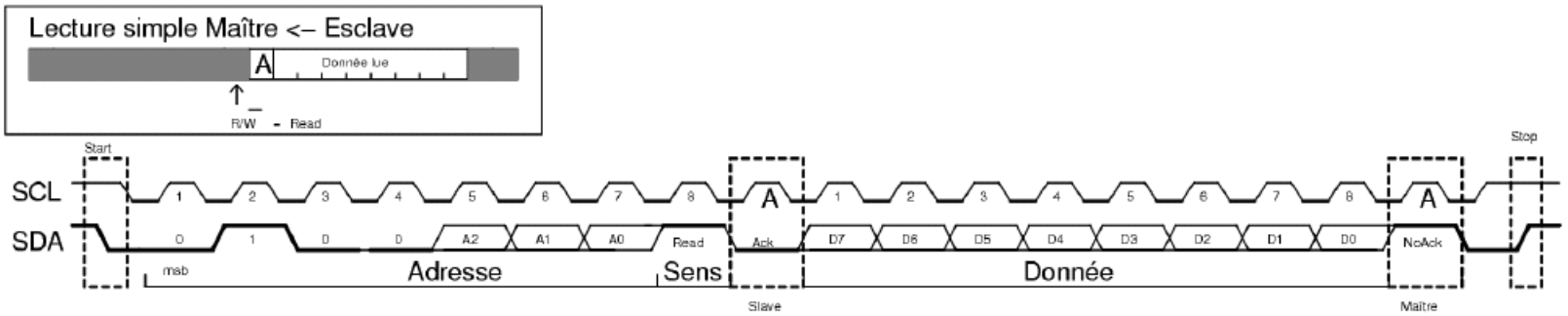
Old circuit !
New one :
PCF 9553..9558

- 8 bits parallel port extension P7..P0
- 8 selectable addresses by 3 input A2..A0
- 4 fixed address bits '0100'

I²C, example PCF8574 port //

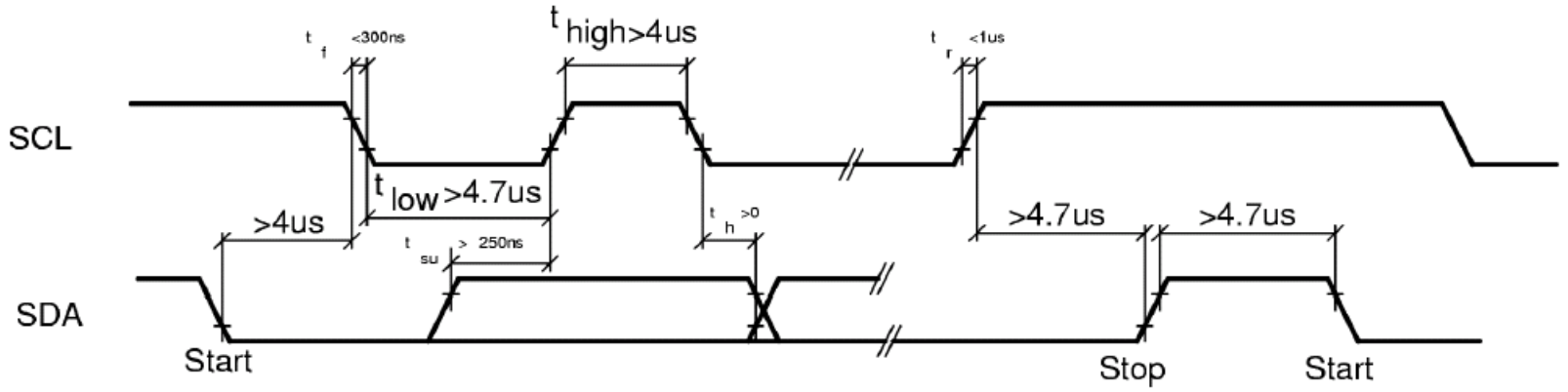


- Writing I²C



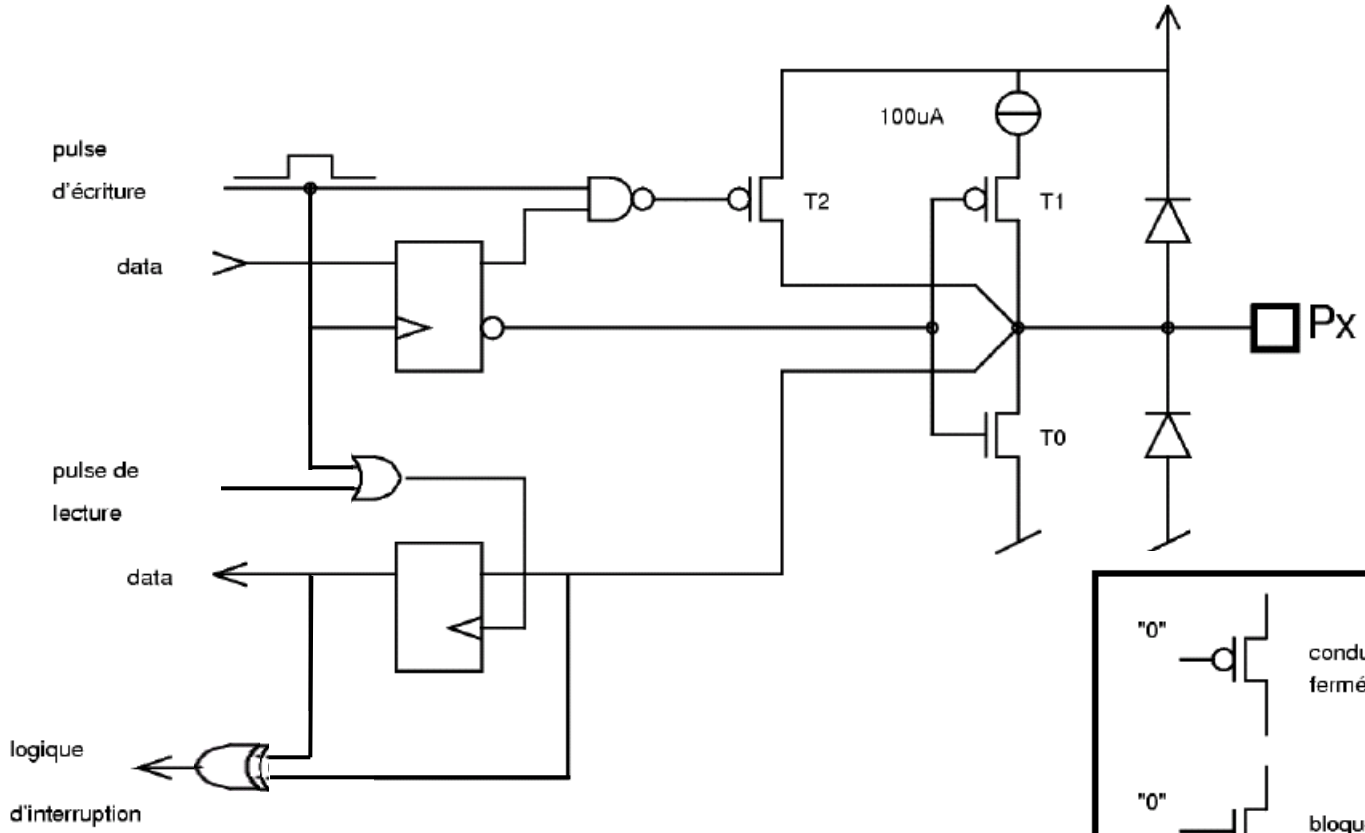
- Reading I²C

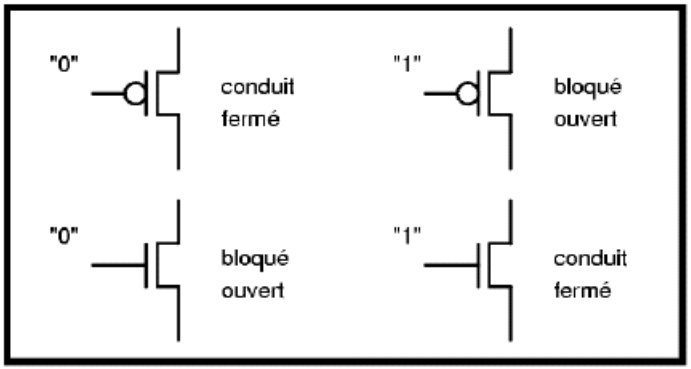
I²C, example PCF8574 port //



- Transmission Timing

I²C, example PCF8574 port //





- One bit of 8574
- NO data direction, pseudo open collector

Serial Interfaces

Bus SPI

Synchronous Peripheral Interface

Motorola

SPI, Synchronous Peripheral Interface (Motorola)

- **Synchronous Bus**
- **Clock line**
- **Full-duplex Transmission (in, out)**
- **Multi-master possible**
- **Slave selection by separated physical lines**

About 1- 4 Mbits/s

SPI, Synchronous Peripheral Interface (Motorola)

MOSI *Master Out, Slave In,*

MISO *Master In, Slave Out,*

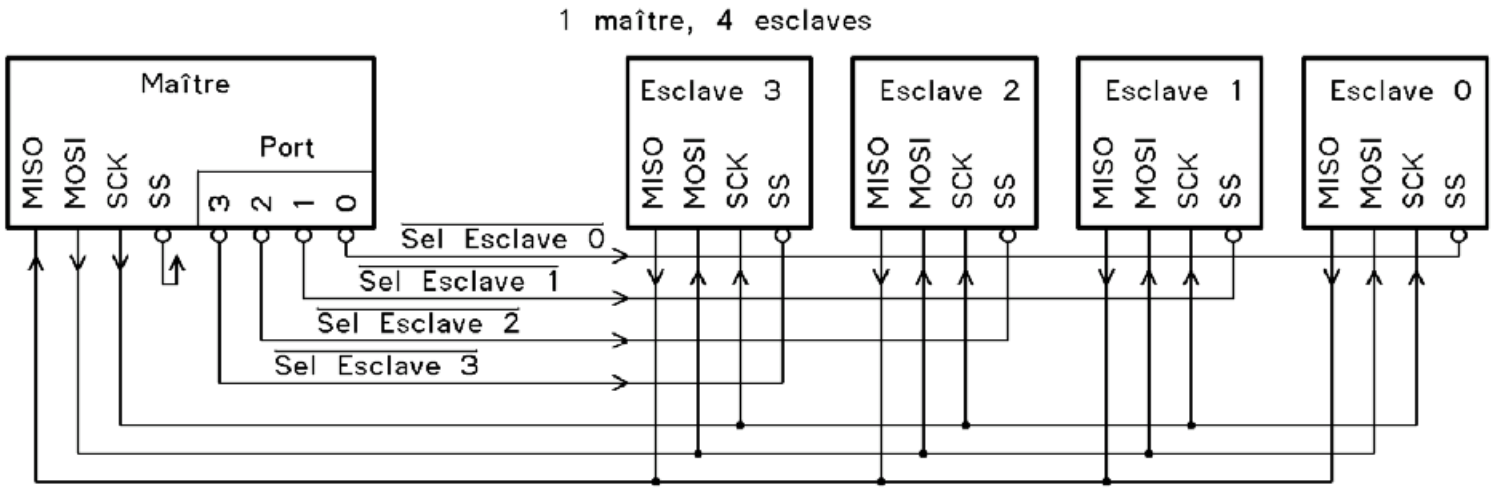
SCK *Serial ClocK*
Provided by the master unit

SS_n *Slave Select,*
1 by slave, generally active low

SPI, Synchronous Peripheral Interface (Motorola)

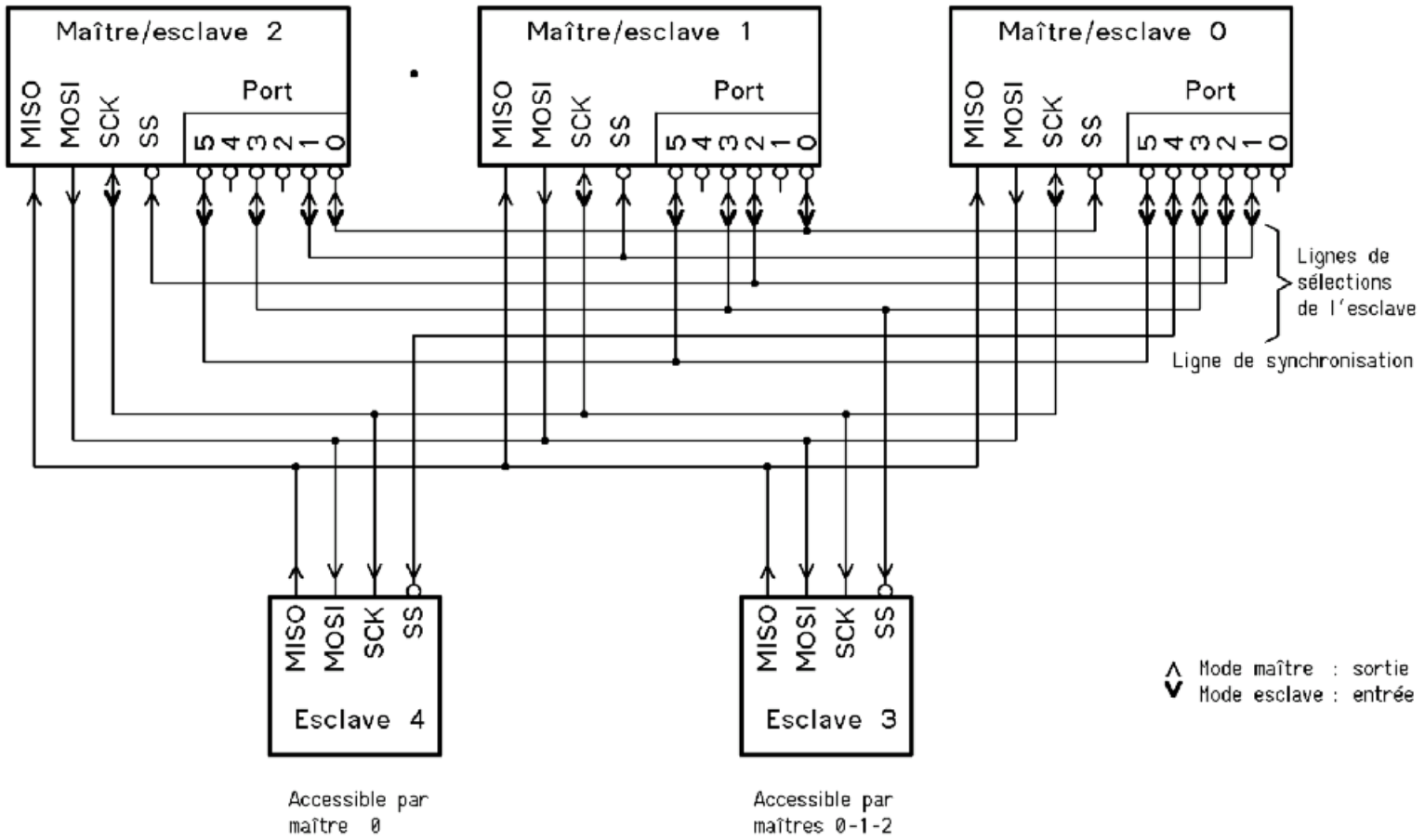
Example :

- 1 master
- 4 slaves



SPI, Synchronous Peripheral Interface (Motorola)

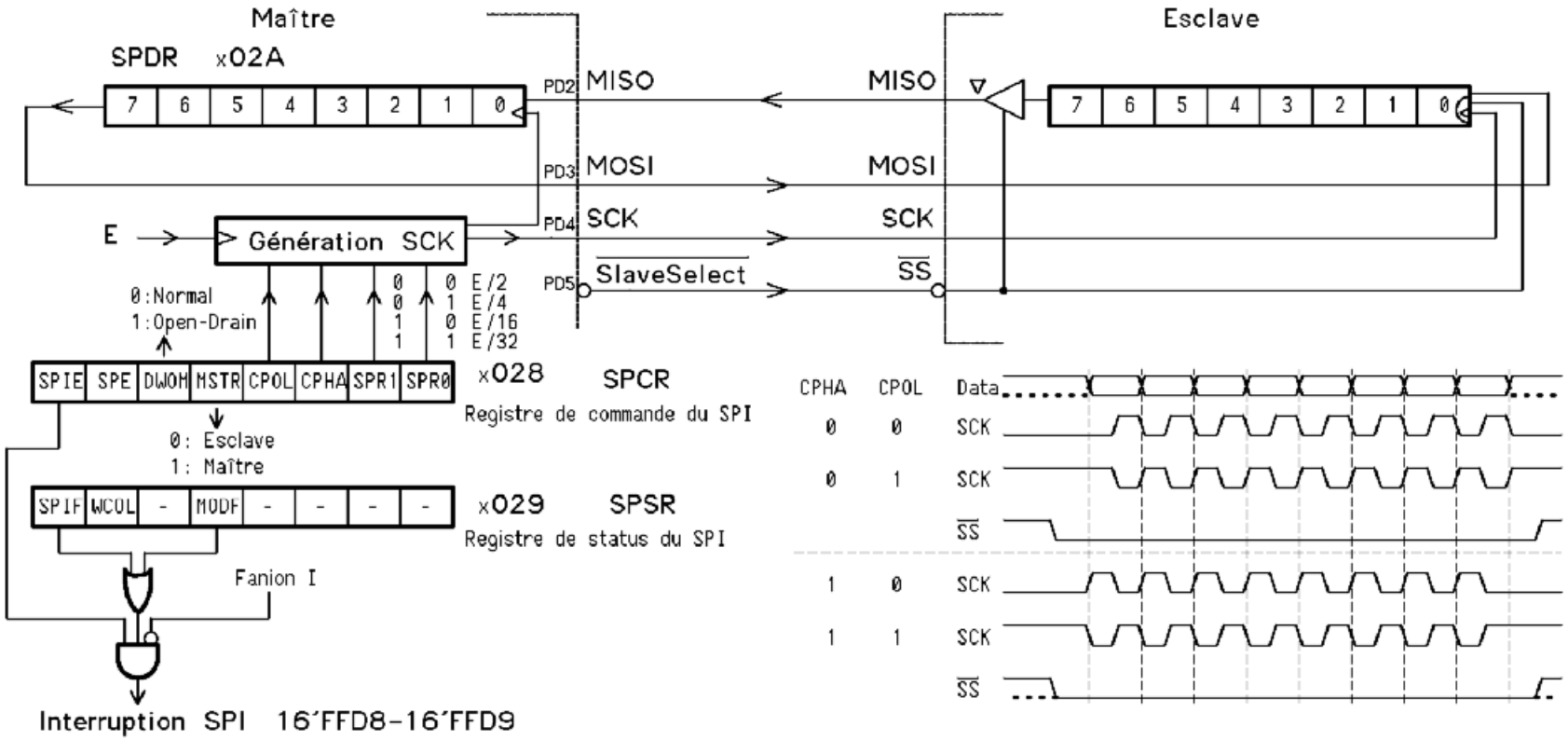
Example : - 3 masters/slaves - 2 slaves



SPI, Synchronous Peripheral Interface (Motorola)

Implementation example on a 68HC11 uC

2 simples registers who exchange their data



Bus 1-Wire, uLAN

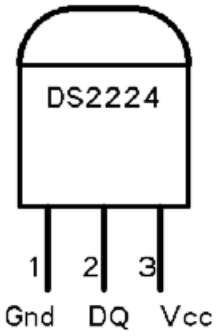
Dallas-Maxim

1-Wire, Dallas/Maxim

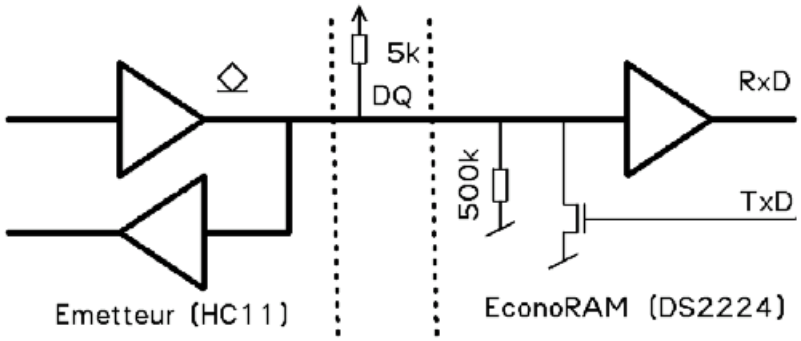
- Serial link with width modulation for reset, '0' and '1' transmission
- Open collector output
- Start bit for each bit activated by the master either for read or write
- ~15 kbit/s
- 50-600m

1-Wire, Dallas/Maxim

- Circuit example, EconoRAM (old circuit)



Master 1-wire device

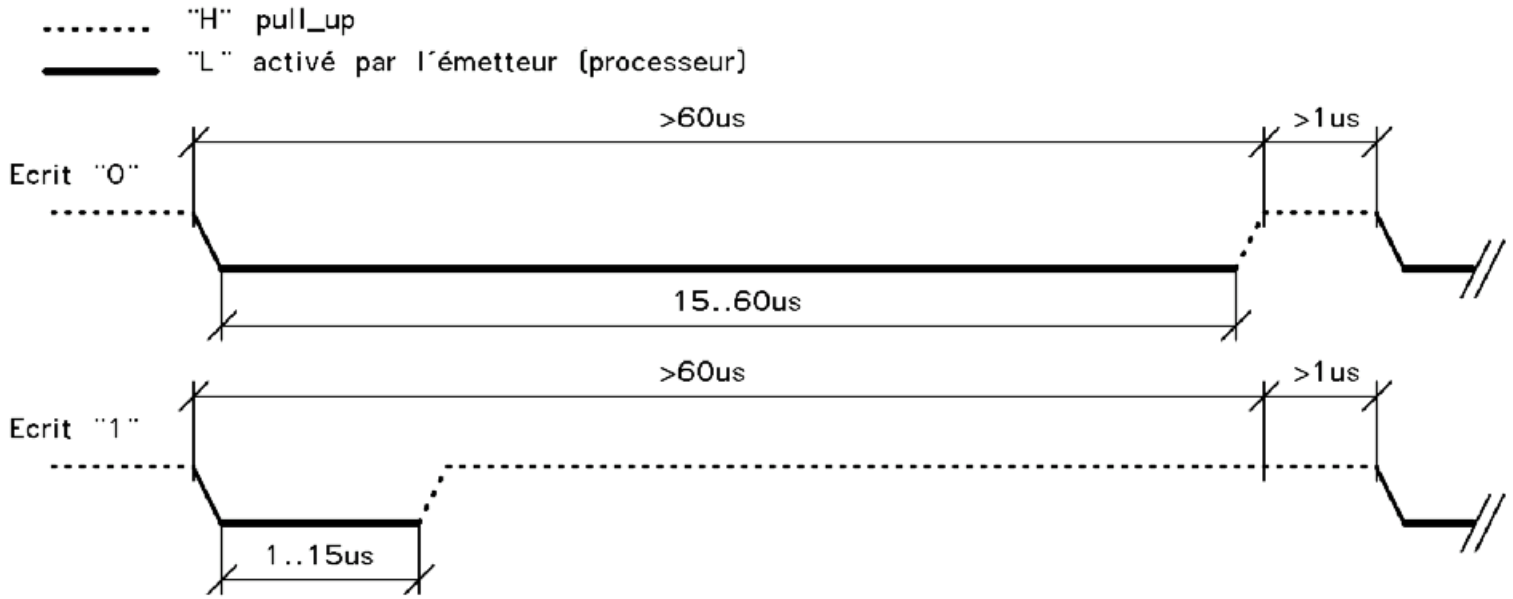


Signal connection

1-Wire, Dallas

- Width modulation
- Open collector
- Start bit for each bit activated by the master either for read or write

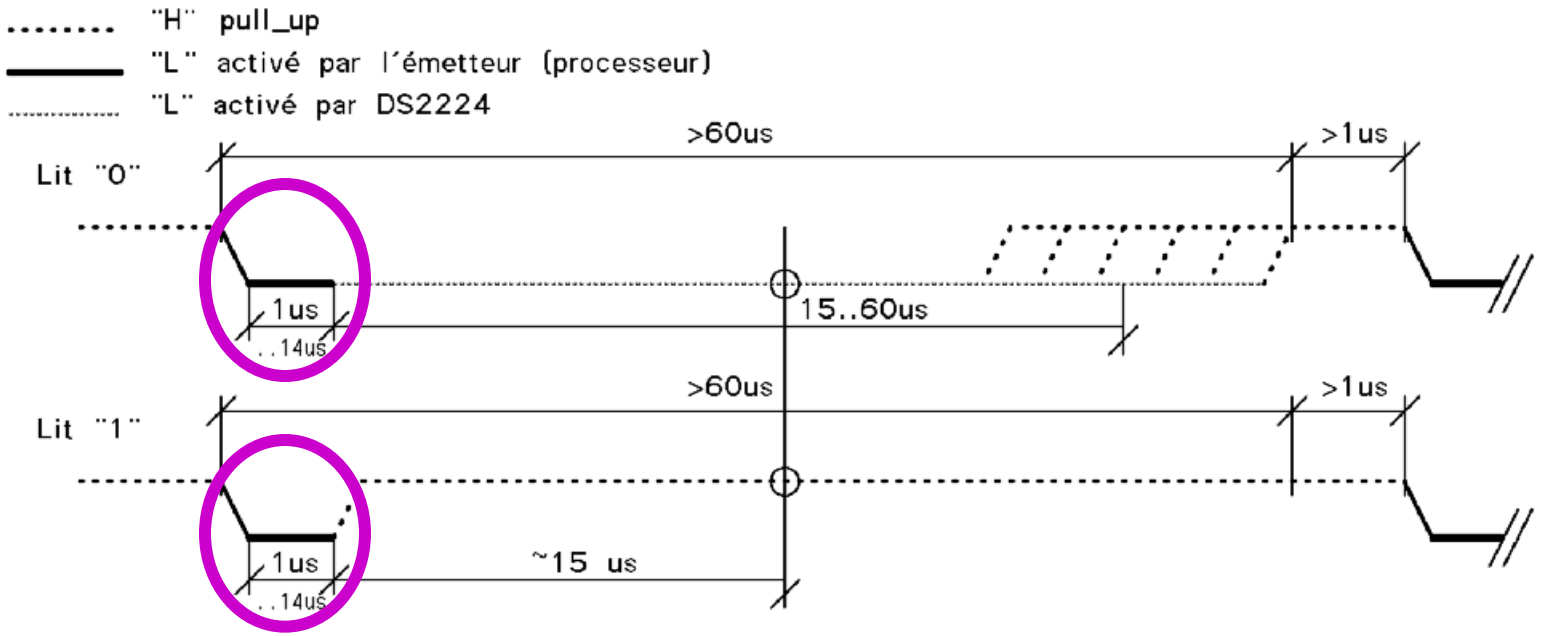
Write access



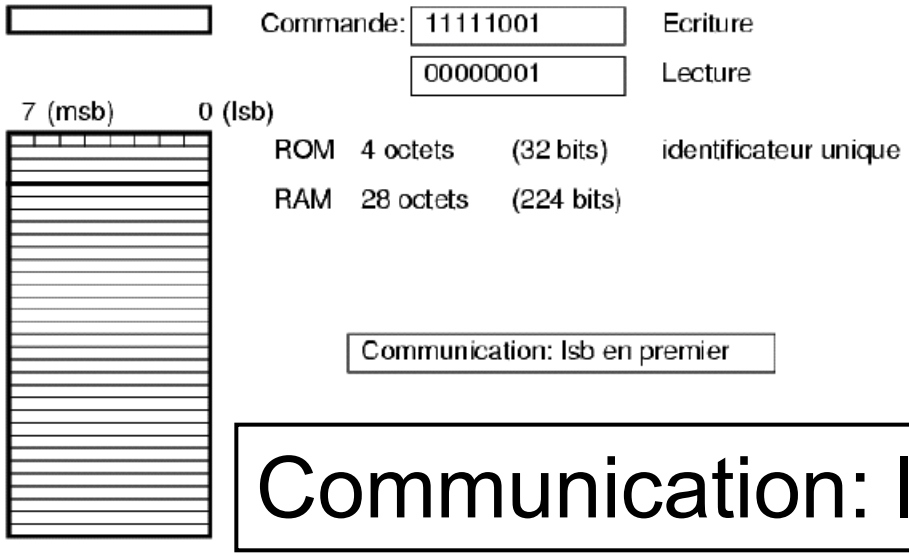
1-Wire, Dallas

- Width modulation
- Open collector
- Start bit for each bit activated by the master either for read or write

Read access

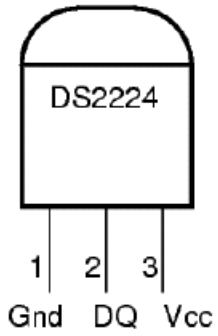


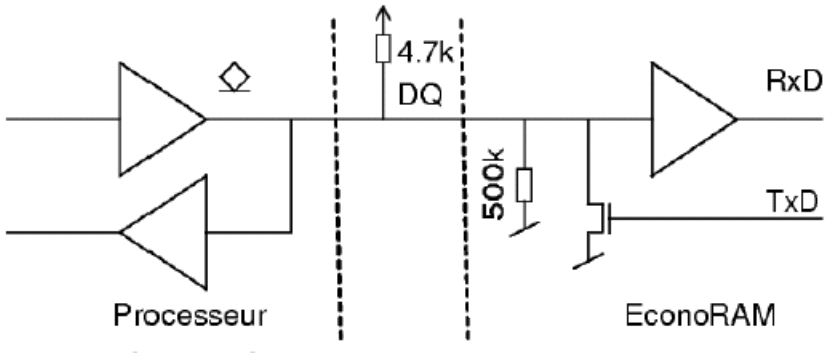
Mémoire Dallas 1-Wire



Communication: lsb first

Mémoire Dallas 1-Wire





Communication :

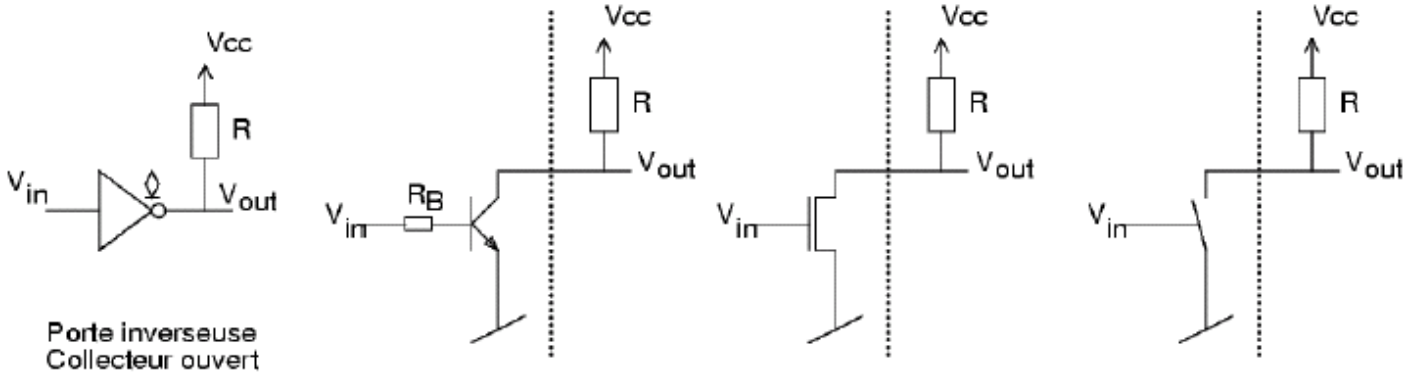
- Pulse width modulation
- Open collector
- Communication start by master

Serial Interfaces

How to do

open collector equivalence with a programmable parallel port ?

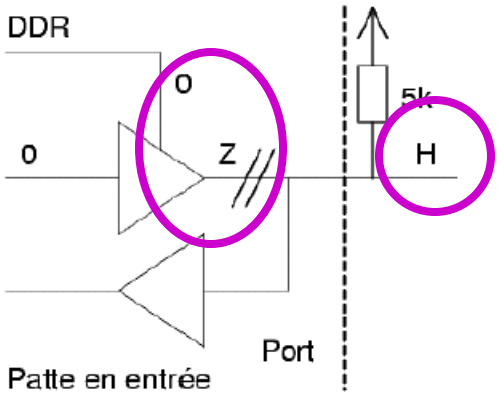
Open collector on // port



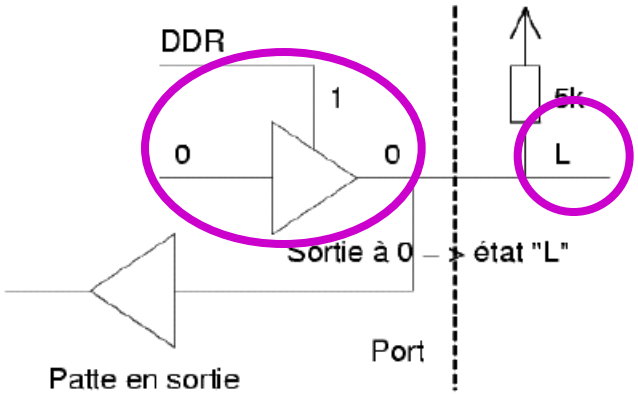
Vcc	alimentation +5V	
V_{in}	2..5V	0..0.8V
V_{out}	0..0.4V	en l'air R –> Vcc

- Open collector model

Open collector on // port



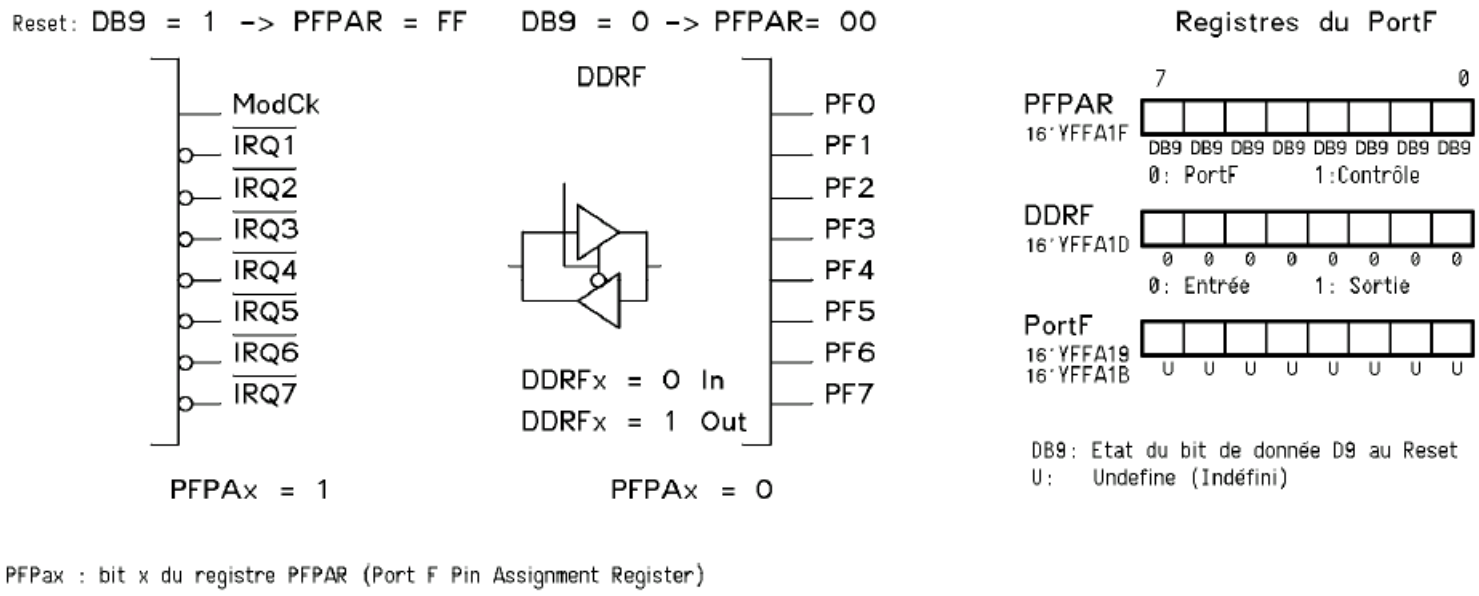
Patte en entrée
Sortie en haute impédance
Etat "H" par la pull–up
Lecture possible de l'état de la ligne



Patte en sortie
Sortie à l'état "0"
Etat "L" par le niveau 0 de la sortie
L'état de la ligne peut être relu sur le port

- Open collector model on a programmable // port
- Programmable Bit direction use :
  - Input → 'Z'
  - Output '0' → '0'

Open collector on // port (ex. 68331)



PFPax : bit x du registre PFPAR (Port F Pin Assignment Register)

DDRx : Data Direction Register

Portx : Port Data

Serial Interfaces

Others serial bus:

USB

GigaEthernet

LVDS serial transmitters

SATA

…