Examples of communication interfaces

Tranditional interfaces

- No intelligence in the interface
- Only physical connection

  This could include changes in voltage levels and transformation from ballanced to unballanced signal
- We define the communication protocol in our application program

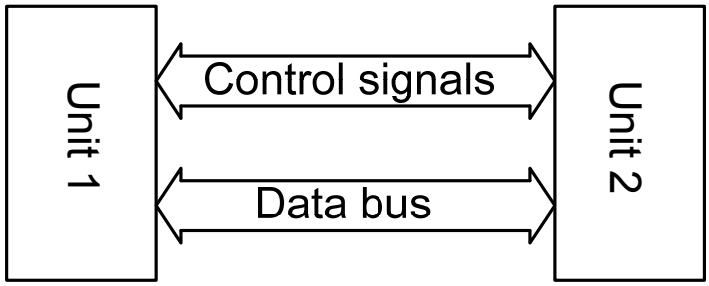
  We might use preprogrammed modules for this
- We have full control of the channel and can use the interface for our own communication protocols if we like

Modern interfaces

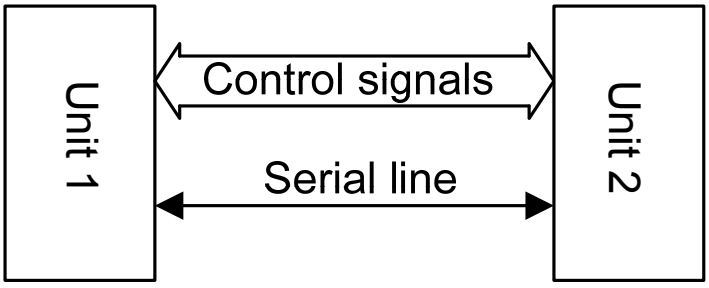
- A high level of intelligence in the interface
- The communication protocol is defined in the interface
- Much of the control of the communication channel is left to the interface and we have to follow the rules of the communication standard to be able to communicate

Here we will focus on the older type of interfaces

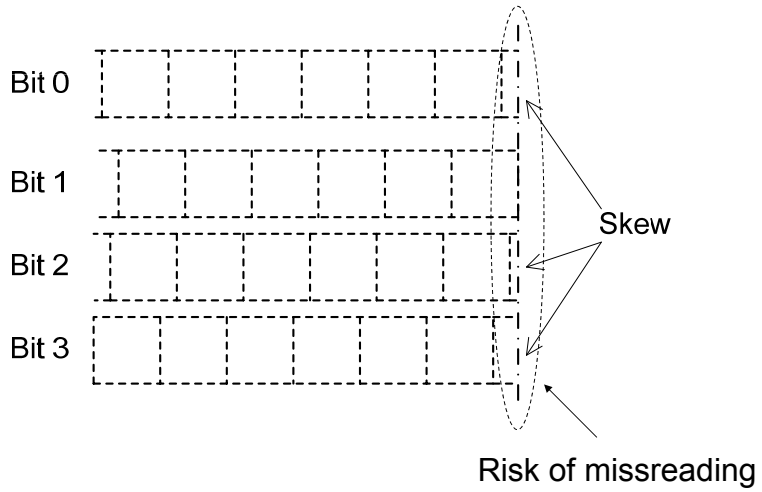
Parallel link



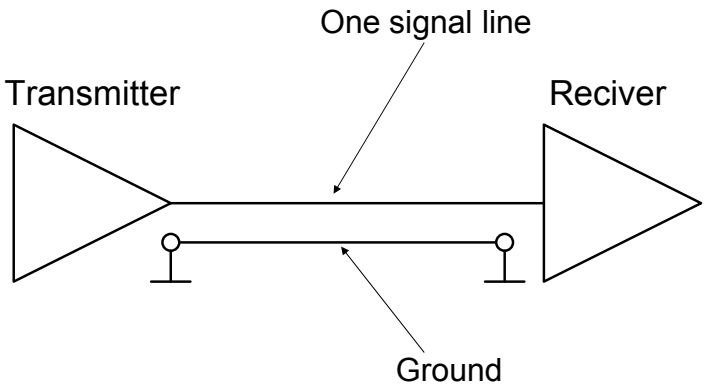
Serial link



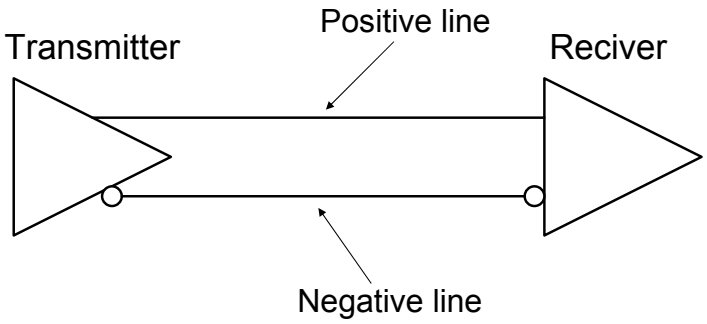
Skew in parallel link



Unbalanced link

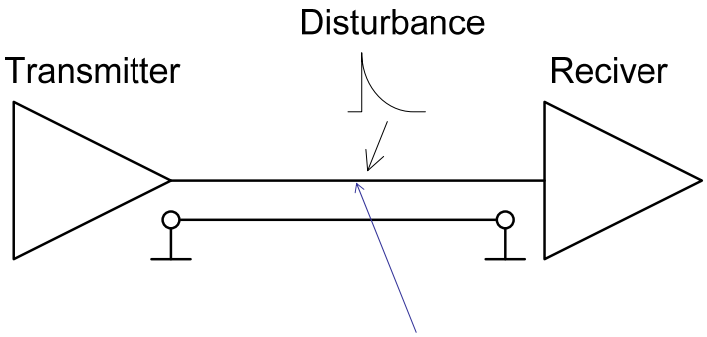


Balanced link



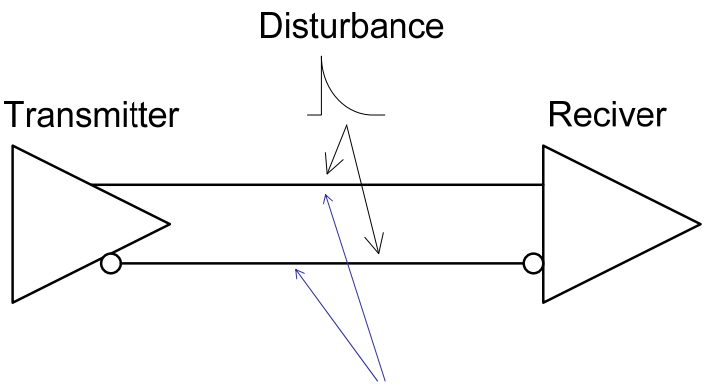
No ground besides maybe shielding

Disturbance on a unbalanced link



The level refered to ground changes
Could give missread signal

Disturbance on a balanced link



Both levels change in the same way so the difference is the same
Lower risk for missread signal

Modem

Modulator/demodulator

Telefon modems

Example V.21

In one direction

'1' – 980 Hz '0' – 1180 Hz

In the other direction

'1' – 1650 Hz '0' – 1850 Hz

Parallel interfaces

Two examples

- Memory interface
- GPIB

Memory interface

Example from HC12

Two emulation modes

- Emulation expanded wide
  - 16 bit address bus — PORTA and PORTB
  - 16 bit data bus — PORTA and PORTB

  Multiplexing
- Emulation expanded narrow
  - 16 bit address bus — PORTA and PORTB
  - 8 bit data bus — PORTA

  Multiplexing

Memory interface cont.

External 8K static RAM, 6264

- 13 Address lines → 8K addresses
- 8 Data lines → byte oriented
- Output enable /OE active low

  Open for reading
- Write enable /WE active low

  Open for writing
- Two chip select signals CS1 and CS2

  Activate chip

Memory interface cont.

Two phases

- Address phase
- Data phase

The memory chip still needs to be addressed during the data phase

Control using external logic

Memory interface cont.

Control signals from the processor

- Read/write R/W

  High for read

  Low for write

- E clock ECLK

  Low during address phase

  High during data phase

Memory interface cont.

Address decoding

Base address – C000

A15 – A13 = 110

$$CS1 = A_{15} \cdot A_{14} \cdot \overline{A_{13}} = \overline{\overline{A_{15} \cdot A_{14}} \cdot \overline{A_{13}}}$$

Memory interface cont.

Logic for Write Enable /WE

- Active (low) in data phase (ECLK high)

  and when R/W is low

ECLK	R/W	/WE
0	0	1
0	1	1
1	0	0
1	1	1

$$\overline{WE} = \overline{ECLK \cdot \overline{R/W}}$$

Memory interface cont.

Logic for Output Enable /OE

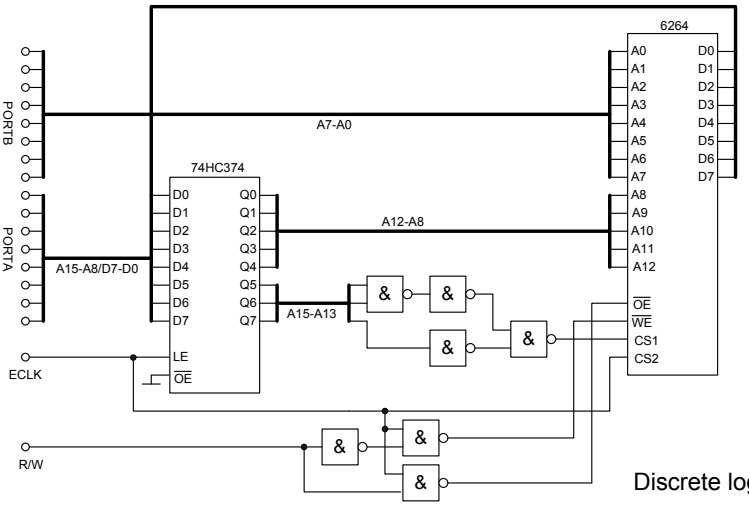
- Active (low) in data phase (ECLK high)

  and when R/W is high

ECLK	R/W	/OE
0	0	1
0	1	1
1	0	1
1	1	0

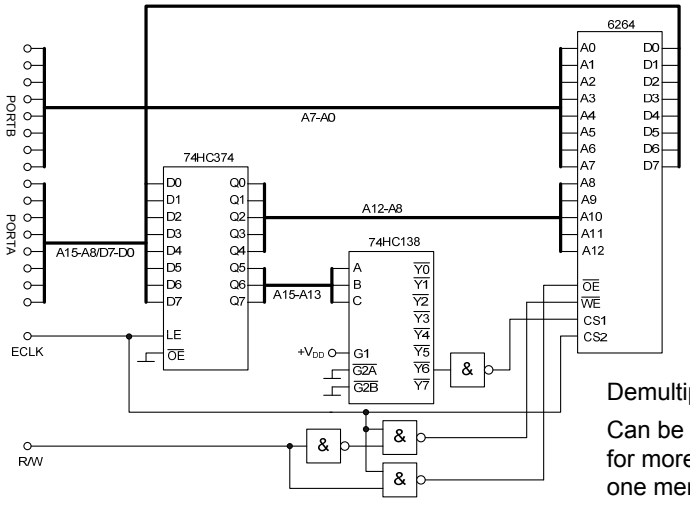
$$\overline{OE} = \overline{ECLK \cdot R/W}$$

Memory interface cont.



Discrete logic

Memory interface cont.



Demultiplexer

Can be used for more than one memory area

GPIB

Parallel bus for measurement instruments

General Purpose Instrument Bus

- IEEE-488

Institute of Electrical and Electronics Engineers

- IEEE-488.1 IEC-625

International Electrotechnical Commision

- IEEE-488.2
- SCPI

Standard Commands for Programmable Instruments

GPIB cont.

Up to 15 instruments

Three types of devices

- Controller
- Talker
- Listener

One instrument can have more than one function

GPIB cont.

Speed

Standard GPIB up to

1.8 Mbyte/second

High speed GPIB HS488 up to

8 Mbyte/second

GPIB cont.

8 data lines

8 ground lines

3 handshake lines

5 interface management lines

GPIB cont.

The handshake lines

NRFD Not Ready for Data

NDAC Not Data Accepted

DAV Data Valid

GPIB cont.

The interface management lines

ATN Attention

EOI End or Identify

IFC Interface Clear

REN Remote Enable

SRQ Service Request

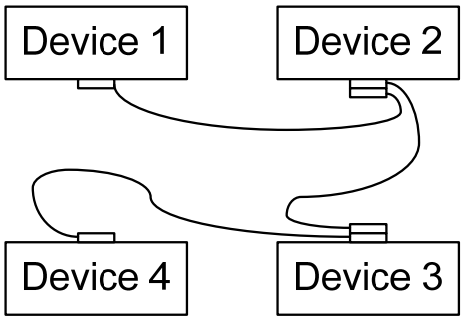
GPIB cont.

Pin	*Abbriviation*	*Name*
1	DIO1	Data input/output bit 1
2	DIO2	Data input/output bit 2
3	DIO3	Data input/output bit 3
4	DIO4	Data input/output bit 4
5	EIO	End or Identify
6	DAV	Data Valid
7	NRFD	Not Ready for Data
8	NDAC	Not Data Accepted
9	IFC	Interface Clear
10	SRQ	Service Request
11	ATN	Attention
12		Shield
13	DIO5	Data input/output bit 5
14	DIO6	Data input/output bit 6
15	DIO7	Data input/output bit 7
16	DIO8	Data input/output bit 8
17	REN	Remote Enable
18		Shield
19		Shield
20		Shield
21		Shield
22		Shield
23		Shield
24		Single GND

GPIB cont.

Bus configuration

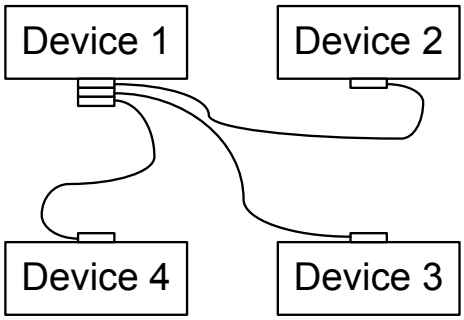
Linear



GPIB cont.

Bus configuration cont.

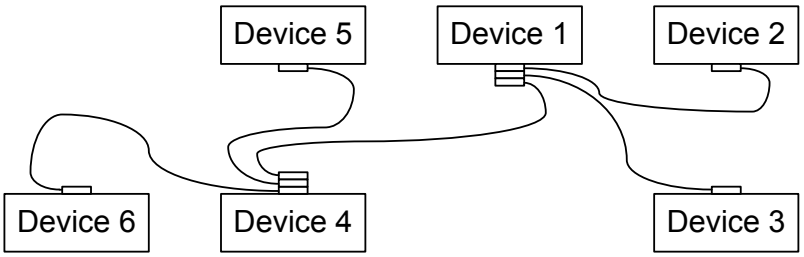
Star



GPIB cont.

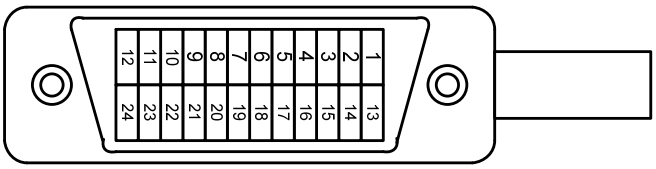
Bus configuration cont.

Combined linear and star



GPIB cont.

Connector



Stackable

RS-232-C

Unbalanced

Synchronious or asynchronious

Logic one ~ -12 Volts

Logic zero ~ +12 Volts

Up to 20 Kbps in standard but used for higher bit rates, for example 115,2 Kbps

Distances up to 15 meter

RS-232-C signals 25 pin DSUB

Pin	*Abbriviation*	*Name*	*Direction*
1	GND	Protective ground	Both ways
2	TD	Transmitted data	TDE to DCE
3	RD	Received data	TCE to DTE
4	RTS	Request to send	TDE to DCE
5	CTS	Clear to send	TCE to DTE
6	DSR	Data set ready	TCE to DTE
7	SG	Signal ground	Both ways
8	DCD	Data carrier detect	TCE to DTE
9		Positive test voltage	TCE to DTE
10		Negative test voltage	TCE to DTE
11		Unassigned	
12	SDCD	Secondary data carrier detect	TCE to DTE
13	SCTS	Secondary clear to send	TCE to DTE
14	STD	Secondary transmitted data	TDE to DCE
15	TC	Transmit clock	TCE to DTE
16	SRD	Secondary received data	TCE to DTE
17	RC	Receive clock	TCE to DTE
18		Unassigned	
19	SRTS	Secondary request to send	TDE to DCE
20	DTR	Data terminal ready	TDE to DCE
21	SQ	Signal quality detect	TCE to DTE
22	RI	Ring indicator	TCE to DTE
23	DRS	Data rate select	Either way
24	XTC	External transmit clock	TDE to DCE
25		Unassigned	

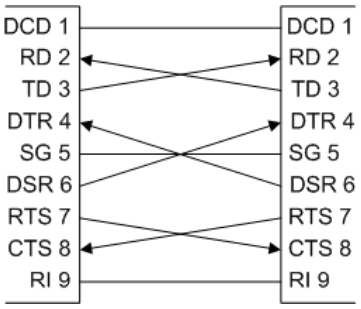
RS-232-C signals 9 pin DSUB

Pin	*Abbriviation*	*Name*	*Direction*
1	DCD	Data carrier detect	TCE to DTE
2	RD	Received data	TCE to DTE
3	TD	Transmitted data	TDE to DCE
4	DTR	Data terminal ready	TDE to DCE
5	SG	Signal ground	Both ways
6	DSR	Data set ready	TCE to DTE
7	RTS	Request to send	TDE to DCE
8	CTS	Clear to send	TCE to DTE
9	RI	Ring indicator	TCE to DTE

SCI – Asynchronous Communication Interface

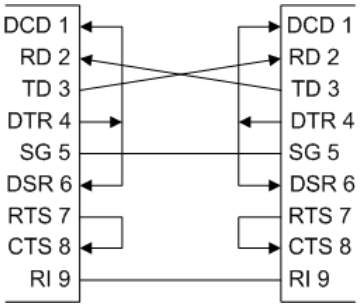
Connecting a DTE unit to a DTE unit

DTE1		*DTE2*	
Pin	*Abbriviation*	*Pin*	*Abbriviation*
1	DCD	1	DCD
2	RD	3	TD
3	TD	2	RD
4	DTR	6	DSR
5	SG	5	SG
6	DSR	4	DTR
7	RTS	8	CTS
8	CTS	7	RTS
9	RI		RI



Zero modem

Pin	*Output*	*Pin*	*Input*
4	DTR	1	DCD
		6	DSR
7	RTS	8	CTS

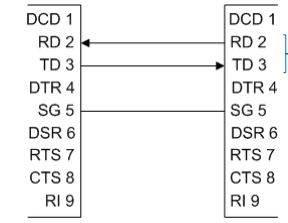


Simplified connection

Sometimes only Rx and Tx are implemented

This could be done

straight



DTE DCE

In this case the RD and TD connections must be reversed in one end.
In most cases at the DCE

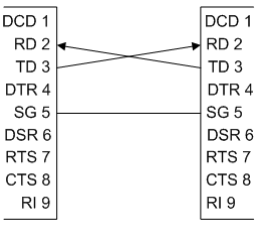
Reversed function

Simplified connection

Sometimes only Rx and Tx are implemented

This could also be done

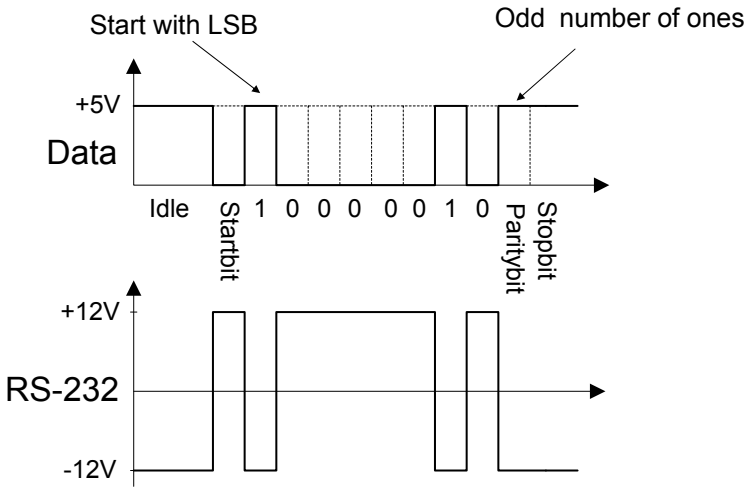
crossed-over



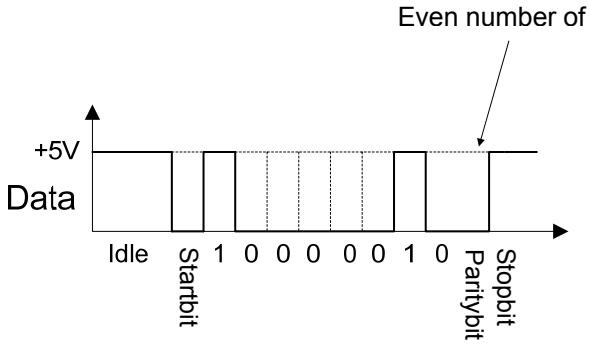
This type of connection is necessary when two master units are connected together

Make sure that you use the correct type of cable

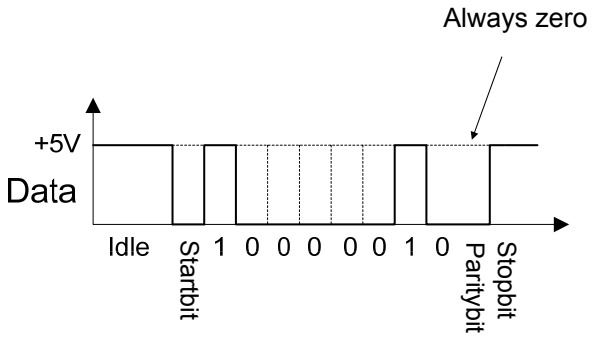
Transfer of the letter ’A’ ($65_{10}=41_{16}$) odd parity



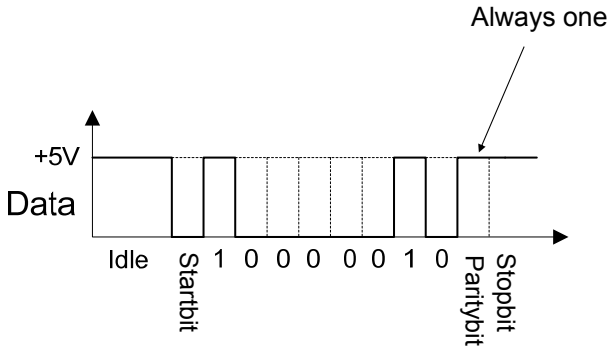
Transfer of the letter ’A’ even parity



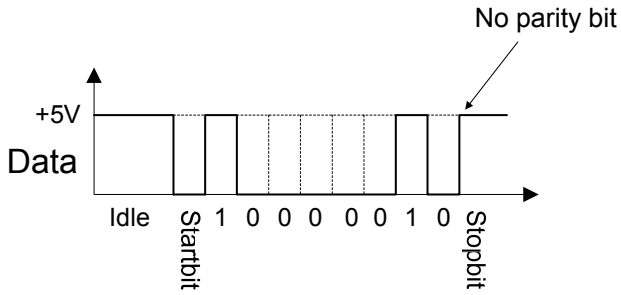
Transfer of the letter ’A’ space parity



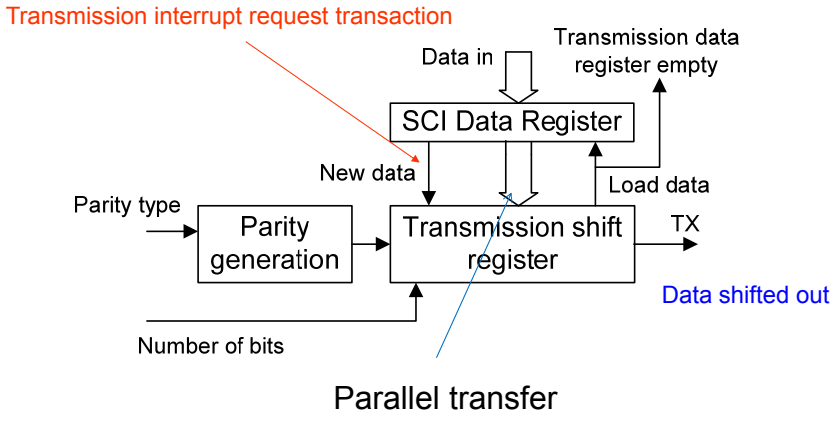
Transfer of the letter ’A’ mark parity



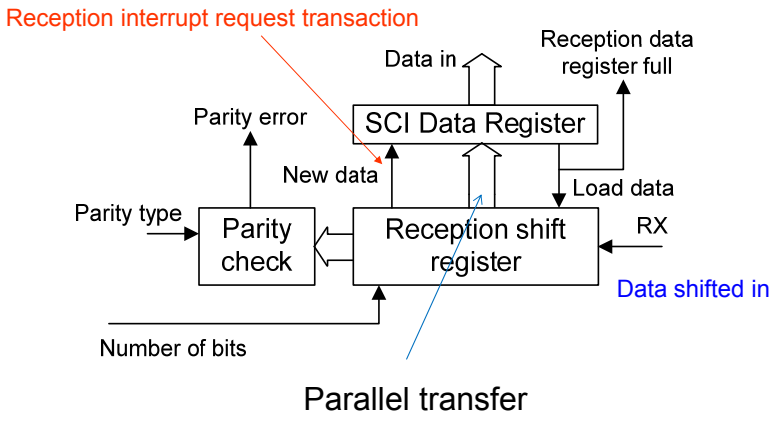
Transfer of the letter ’A’ no parity



Typical asynchronous transmitter



Typical asynchronous receiver



Baud rate and symbol rate

Baud rate is the number of signal changes per second

Symbol rate is the number of symbols per second

In simple transmission the two are the same

In modern transmission techniques (modulation) more than one symbol can be sent in each bit so the symbol rate might be higher than the baud rate

Hamming coding

Example

4 data bits

3 parity bits

$D_3D_2D_1P_2D_0P_1P_0$

Hamming coding CONT.

Generation of parity bits

$$\begin{cases} P_2 = D_3 \oplus D_2 \oplus D_1 \\ P_1 = D_3 \oplus D_2 \oplus D_0 \\ P_0 = D_2 \oplus D_1 \oplus D_0 \end{cases}$$

$\oplus$ exclusive-OR

Hamming coding CONT.

Generation of status bits

$$\begin{cases} S_2 = D_3 \oplus D_2 \oplus D_1 \oplus P_2 \\ S_1 = D_3 \oplus D_2 \oplus D_0 \oplus P_1 \\ S_0 = D_2 \oplus D_1 \oplus D_0 \oplus P_0 \end{cases}$$

$S_2S_1S_0$ give error position

Position 7 MSB (to the left)
Position 1 LSB (to the right)

Hamming coding CONT.

Example

We are to transmit the data sequence

$D_3D_2D_1D_0 = 1011$

We get the control bits

$$\begin{cases} P_2 = D_3 \oplus D_2 \oplus D_1 = 1 \oplus 0 \oplus 1 = 0 \\ P_1 = D_3 \oplus D_2 \oplus D_0 = 1 \oplus 0 \oplus 1 = 0 \\ P_0 = D_2 \oplus D_1 \oplus D_0 = 0 \oplus 1 \oplus 1 = 0 \end{cases}$$

We send the total sequence

$D_3D_2D_1P_2D_0P_1P_0 = 1010100$

Hamming coding CONT.

Example cont.

In the receiver we decode the check bits

$$\begin{cases} S_2 = D_3 \oplus D_2 \oplus D_1 \oplus P_2 = 1 \oplus 0 \oplus 1 \oplus 0 = 0 \\ S_1 = D_3 \oplus D_2 \oplus D_0 \oplus P_1 = 1 \oplus 0 \oplus 1 \oplus 0 = 0 \\ S_0 = D_2 \oplus D_1 \oplus D_0 \oplus P_0 = 0 \oplus 1 \oplus 1 \oplus 0 = 0 \end{cases}$$

The result

$$S_2 S_1 S_0 = 000$$

indicates that the transfer is correct

Hamming coding CONT.

Example cont.

Say that there is a transmission error in bit D_1 and
we receive the incorrect sequence

$$D_3 D_2 D_1 P_2 D_0 P_1 P_0 = 1000100$$

We decode the check bits

$$\begin{cases} S_2 = D_3 \oplus D_2 \oplus D_1 \oplus P_2 = 1 \oplus 0 \oplus 0 \oplus 0 = 1 \\ S_1 = D_3 \oplus D_2 \oplus D_0 \oplus P_1 = 1 \oplus 0 \oplus 1 \oplus 0 = 0 \\ S_0 = D_2 \oplus D_1 \oplus D_0 \oplus P_0 = 0 \oplus 0 \oplus 1 \oplus 0 = 1 \end{cases}$$

$$S_2 S_1 S_0 = 101 = 5_{10}$$

indicates an error in bit 5 counted from the right, that is in D_1

We can indicate and correct errors in both data and parity bits

but only one bit error

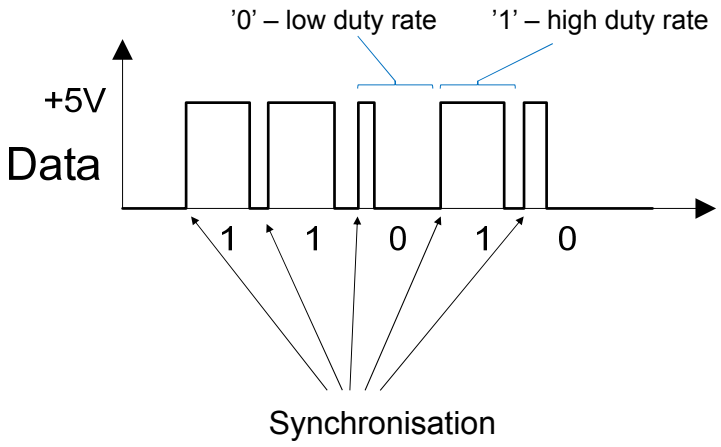
RS-422, RS-423 and RS-485

RS-423 unbalanced, faster than RS-232
but signal compatible, half duplex

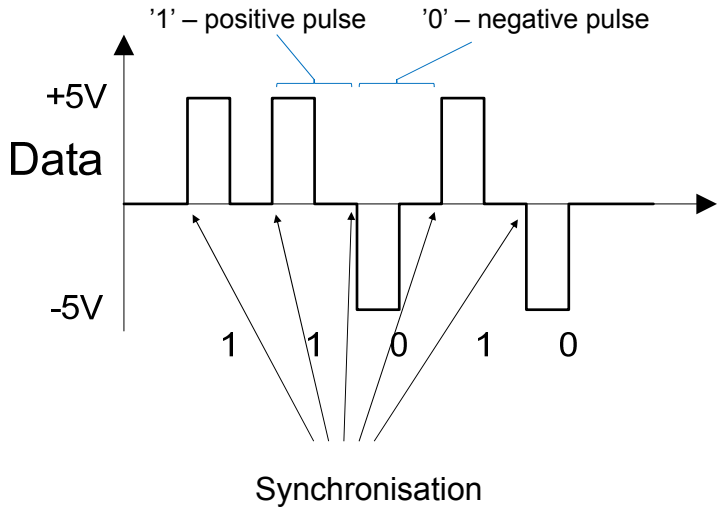
RS-422 balanced version of RS-423, half duplex

RS-485, balanced, multi master

Return to zero protocols



Return to zero protocols cont.



Serial peripheral interface, SPI

Synchronious

Separate clock line

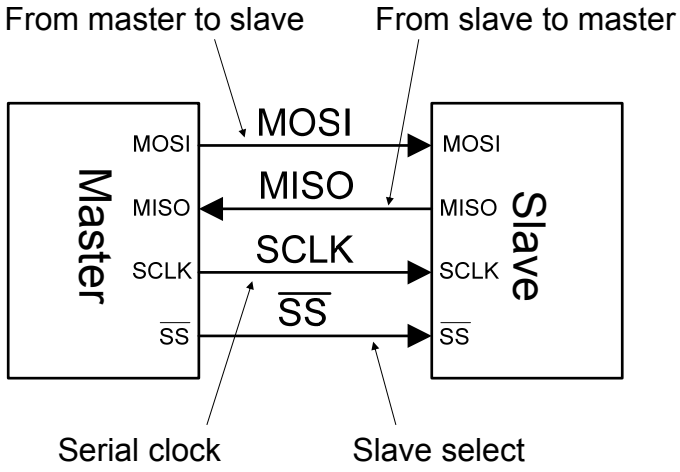
Separate transfer lines in the two directions

Master and slave(s)

Bit rate up to tens of Mbps

Peripherals at short distance

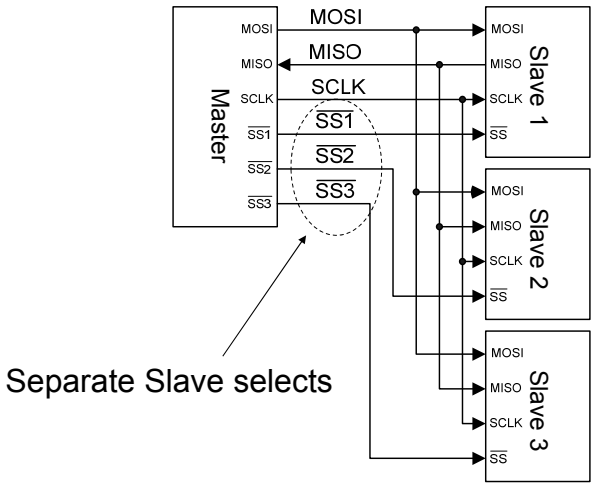
Serial peripheral interface, SPI one slave



Serial peripheral interface, SPI

Symbol	*Name*
MOSI	Master out, slave in
MISO	Master in, slave out
SCLK	Serial clock
SS	Slave select

Serial peripheral interface, SPI multiple slaves

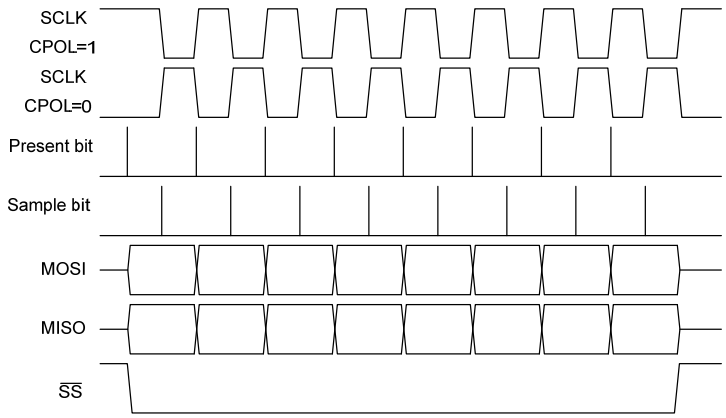


Serial peripheral interface, SPI clocking conditions

SPI mode	*CPOL*	*CPHA*	*Active edge*
0	*0*	*0*	*Rising*
1	*0*	*1*	*Falling*
2	*1*	*0*	*Falling*
3	*1*	*1*	*Rising*

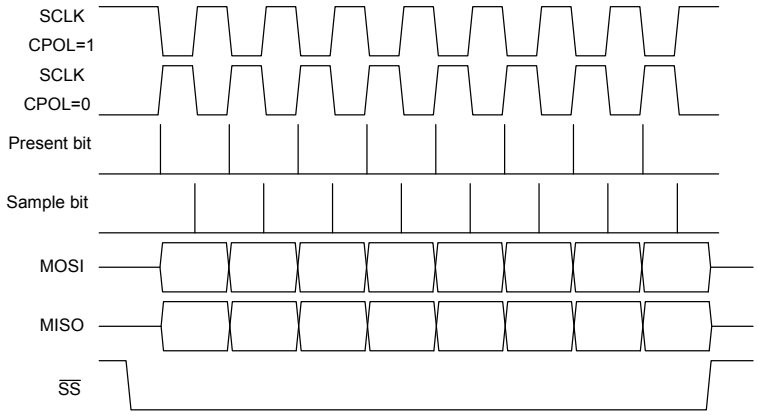
Serial peripheral interface, SPI clocking conditions cont.

CPHA=0



Serial peripheral interface, SPI clocking conditions cont.

CPHA=1



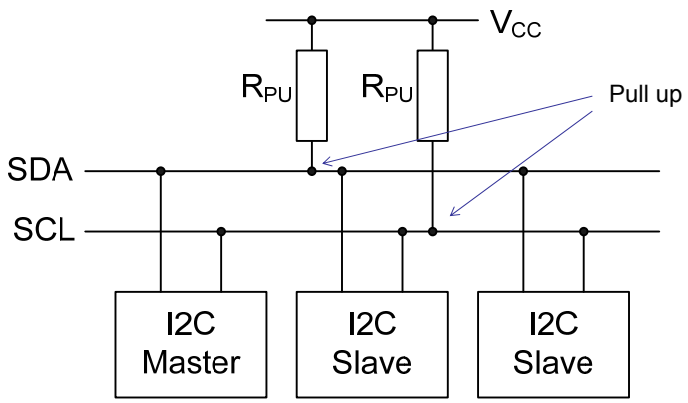
Inter-integrated circuit, I2C

Synchronious

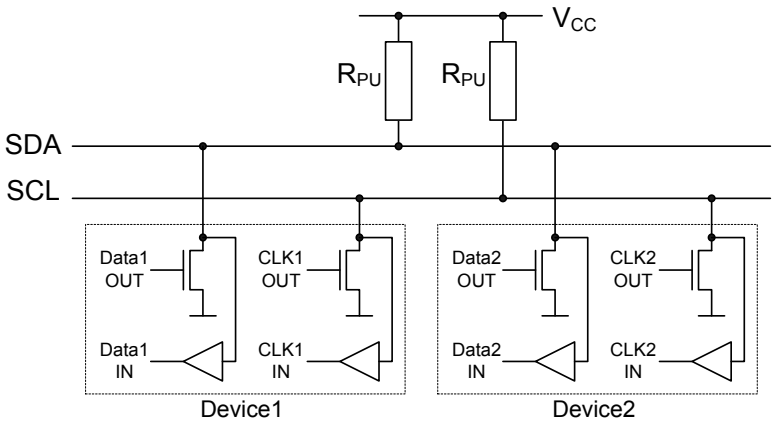
Bus topology

Bit rates up to Mbps

Inter-integrated circuit, I2C

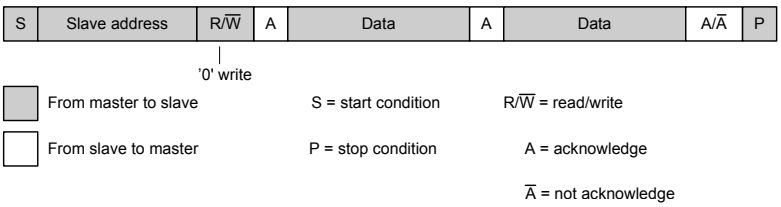


Inter-integrated circuit, I2C cont.



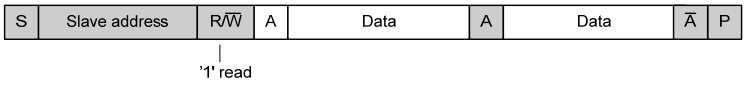
Inter-integrated circuit, I2C cont.

Master transmitting two bytes of data to slave



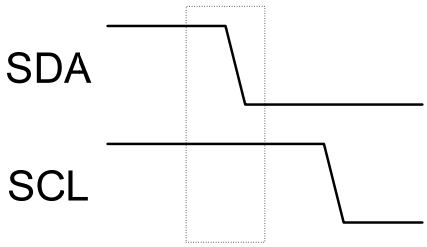
Inter-integrated circuit, I2C cont.

Master receiving two bytes of data from slave



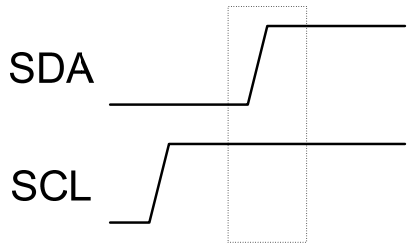
Inter-integrated circuit, I2C cont.

Start condition



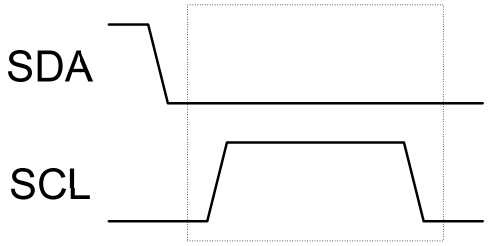
Inter-integrated circuit, I2C cont.

Stop condition

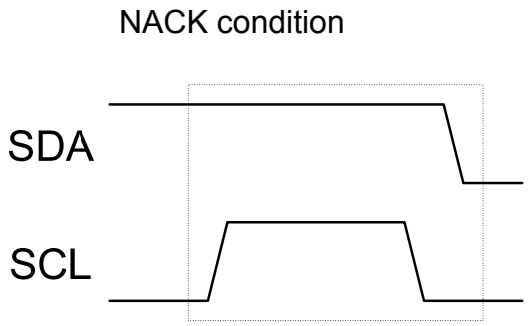


Inter-integrated circuit, I2C cont.

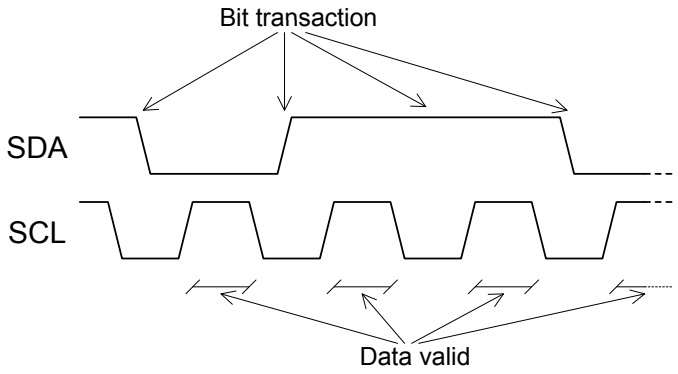
ACK condition



Inter-integrated circuit, I2C cont.



Inter-integrated circuit, I2C cont.



1-wire bus

Dallas Semiconductor

- Bus interface
- One master and slaves

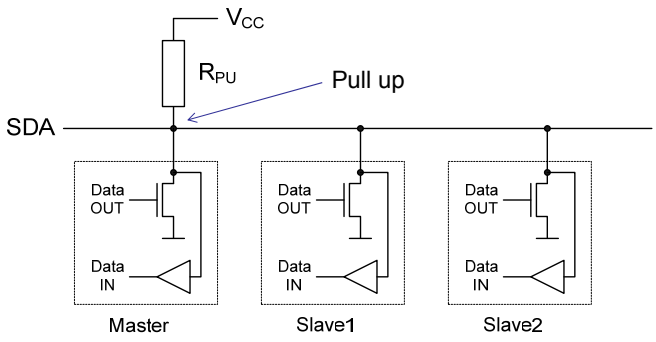
Twisted pair

Bus and ground

Power supply through the bus

1-wire bus

Structure



1-wire bus

Signaling

Bus high when idle

Reset (from master)

Bus low more than 480 µseconds

1-wire bus

Bit transfer

Start of bit transfer

Master pulls bus low for a short while

Sending node (master or slave)

'1' keeps the bus low for more than 60 µseconds

'0' keeps the bus low for less than 15 µseconds

Bits are transmitted LSB first

1-wire bus

Each device has a unique serial number

64 bits

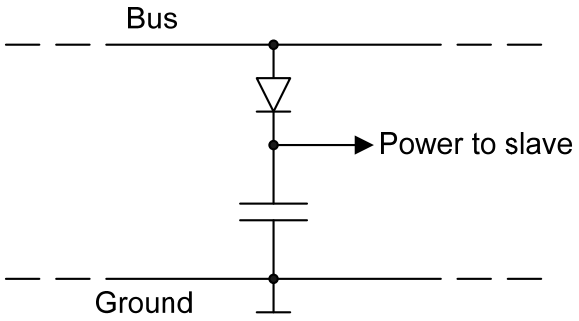
8 bits family code identify the device type

48 bits unique device code

8 CRC check sum

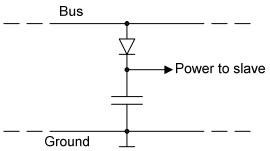
1-wire bus

Power to slave through the bus



1-wire bus

Power to slave through the bus cont.



Capacitor charged when bus is high

Capacitor isolated from bus when bus is low

Capacitor supplies the power