Projet SOC

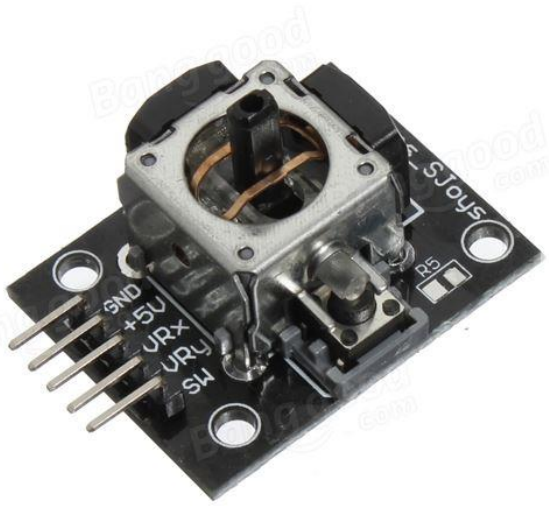
A/D on SPI
Analog to Digital Converter on Synchronous Peripheral Interface
Case study MCP3204

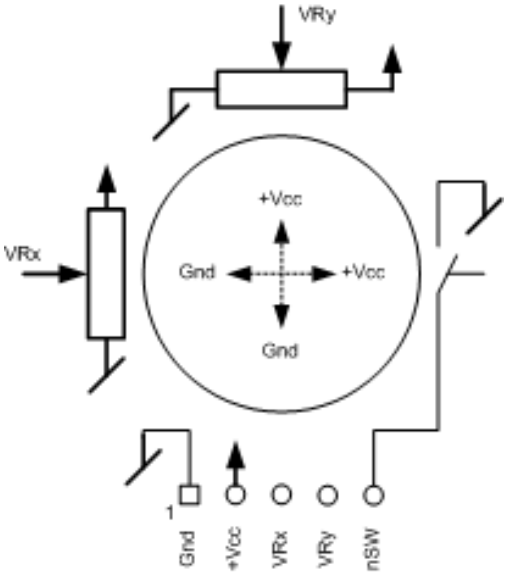
René Beuchat
Laboratoire d'Architecture des Processeurs

rene.beuchat@epfl.ch

Joystick: 2 x Analog signals

- VRx, VRy : analog values between 0..Vcc



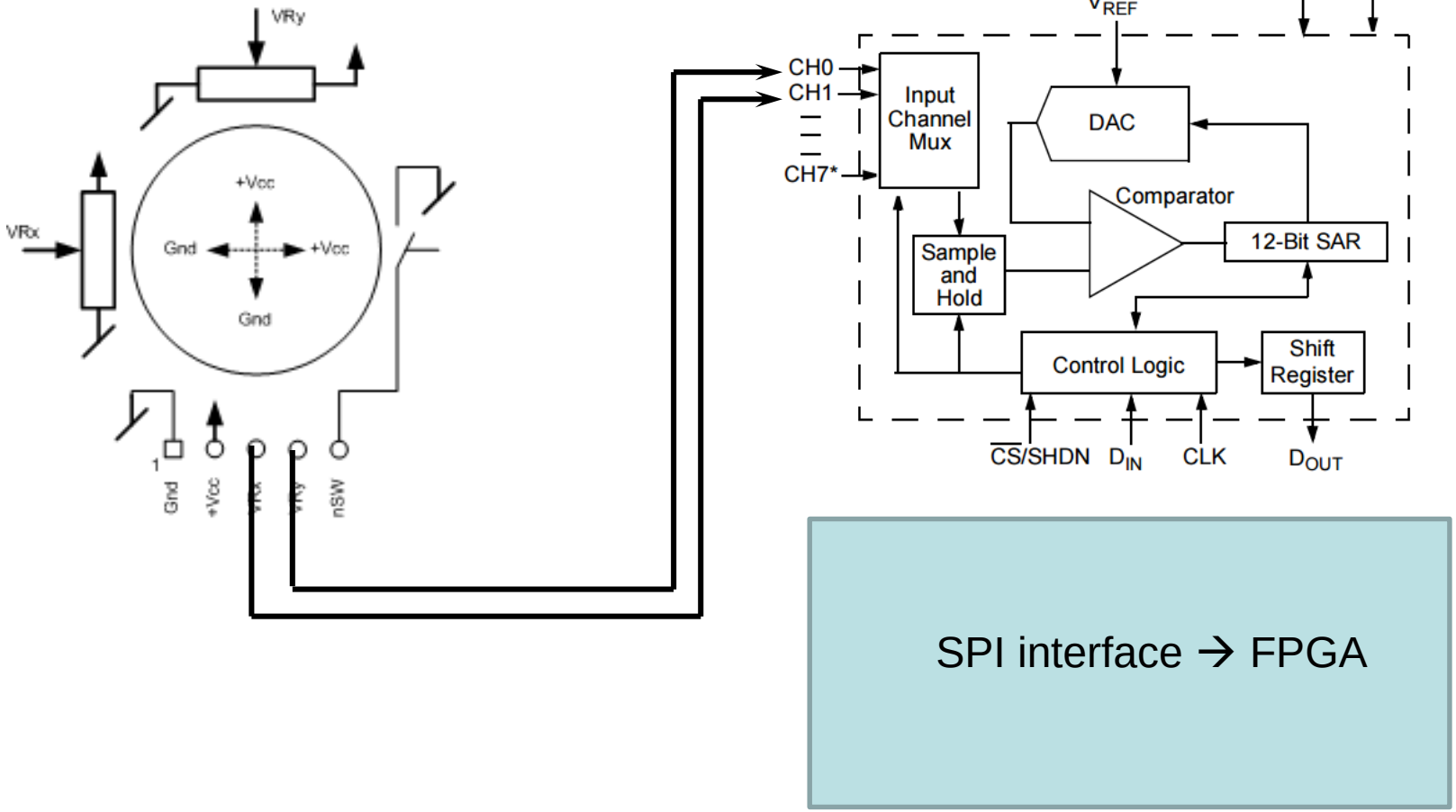


- Use an ADC converter

General system: Joystick to ADC, ADC to FPGA

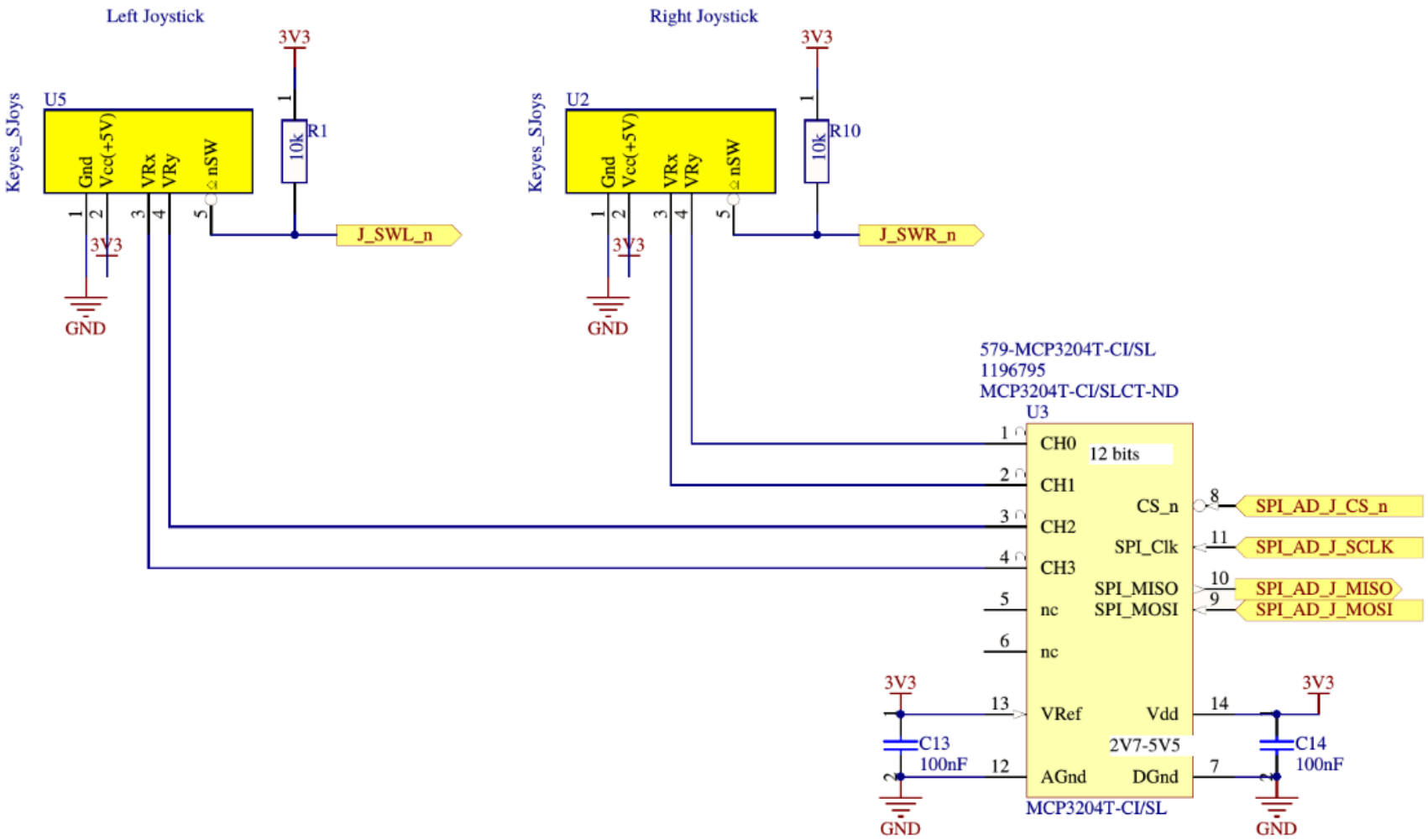
Analog Inputs

CH0 → CH3
Only on
MCP3204

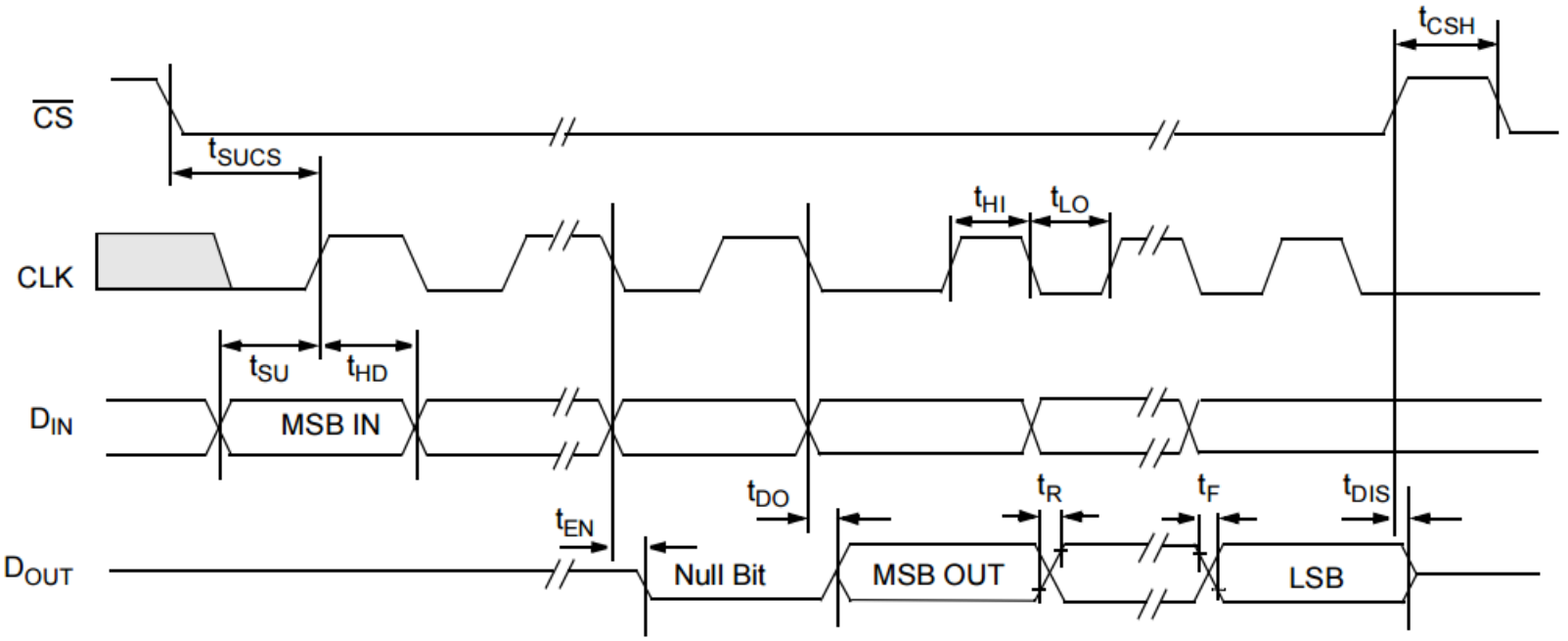


SPI interface → FPGA

Schematic Joystick → ADC on SPI



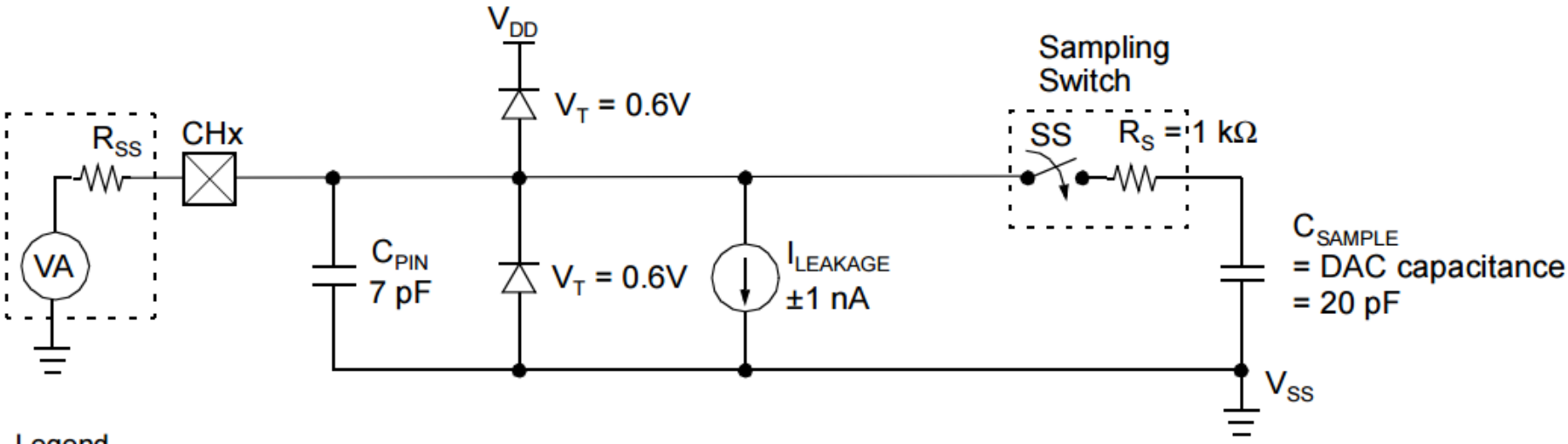
SPI serial protocol



- SClk → fMax 1MHz
- DIn → MOSI
- DOut →MISO

http://ww1.microchip.com/downloads/en/DeviceDoc/21298c.pdf

ADC Input equivalent



Legend

VA	=	Signal Source	$I_{leakage}$	=	Leakage Current At The Pin Due To Various Junctions
R_{ss}	=	Source Impedance	SS	=	Sampling switch
CHx	=	Input Channel Pad	R_s	=	Sampling switch resistor
C_{pin}	=	Input Pin Capacitance	C_{sample}	=	Sample/hold capacitance
V_t	=	Threshold Voltage			

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,*

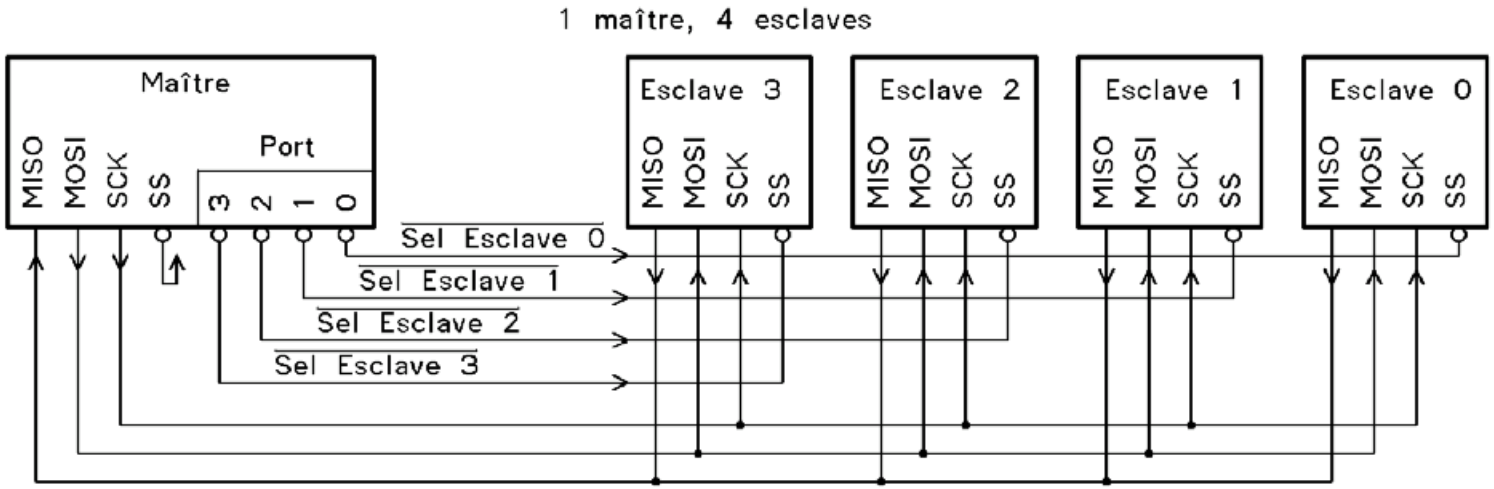
SCK *Serial ClocK*
Provided by the master unit

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

SPI, Synchronous Peripheral Interface (Motorola)

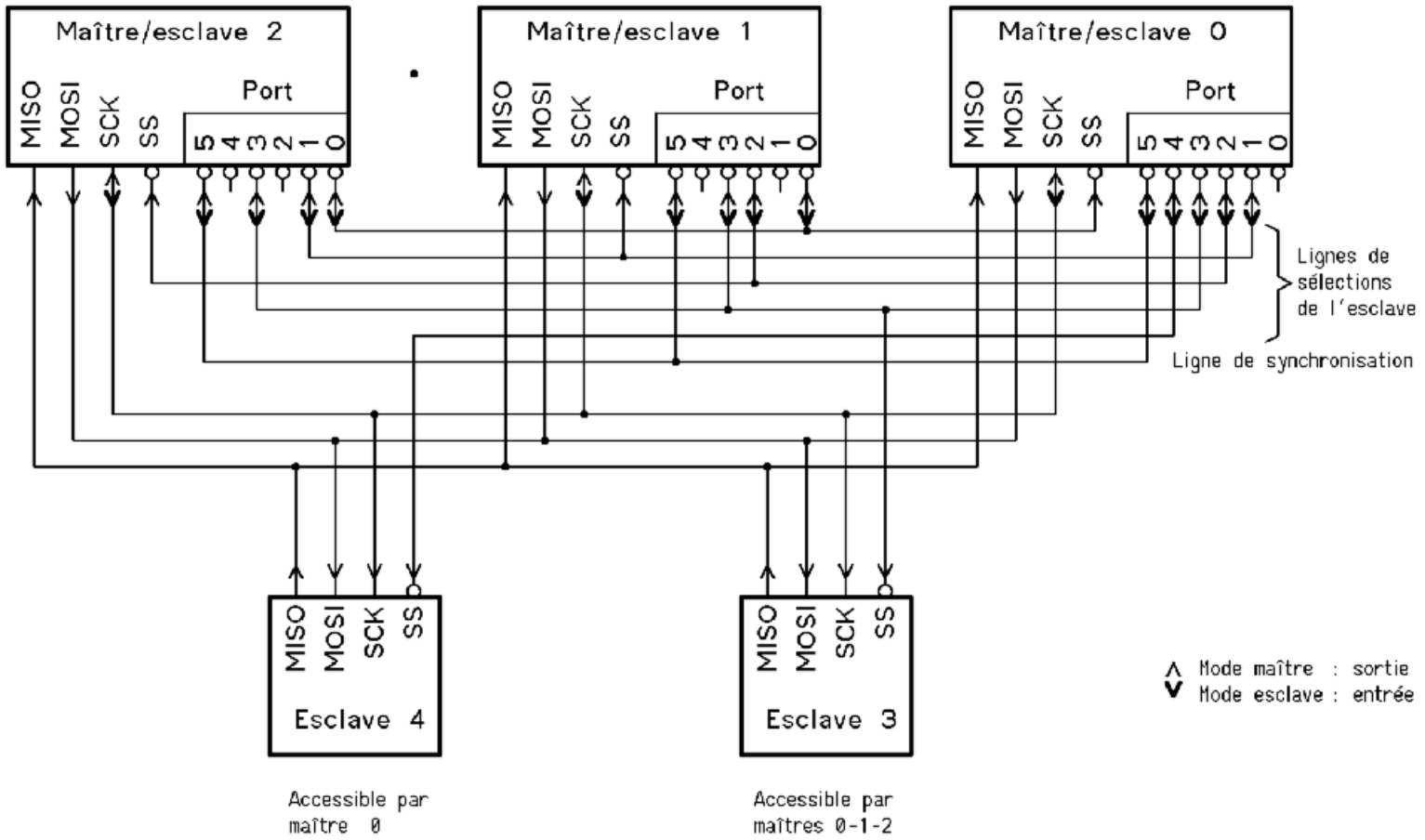
Example :

- 1 master
- 4 slaves



SPI, Synchronous Peripheral Interface (Motorola)

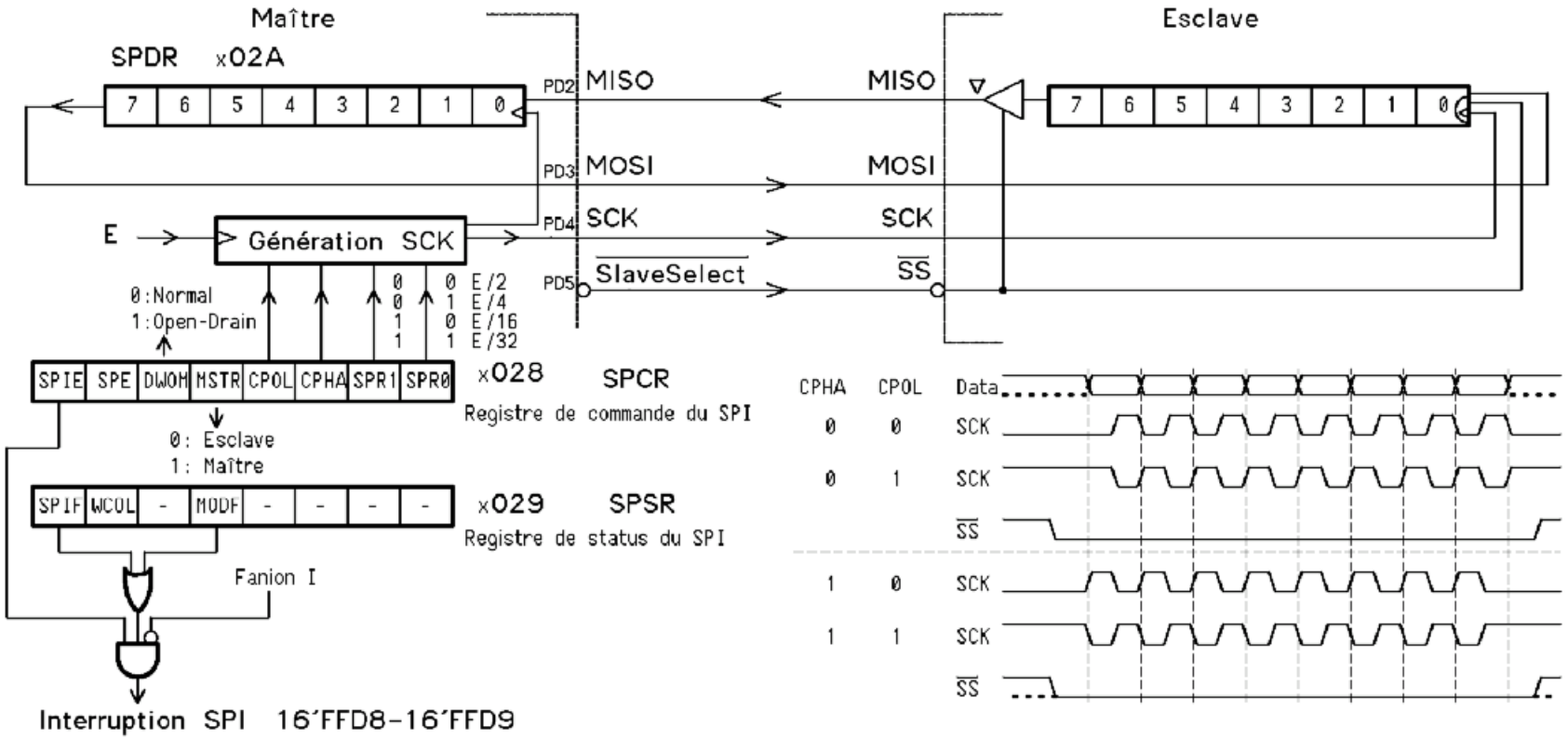
Example : - 3 masters/slaves - 2 slaves



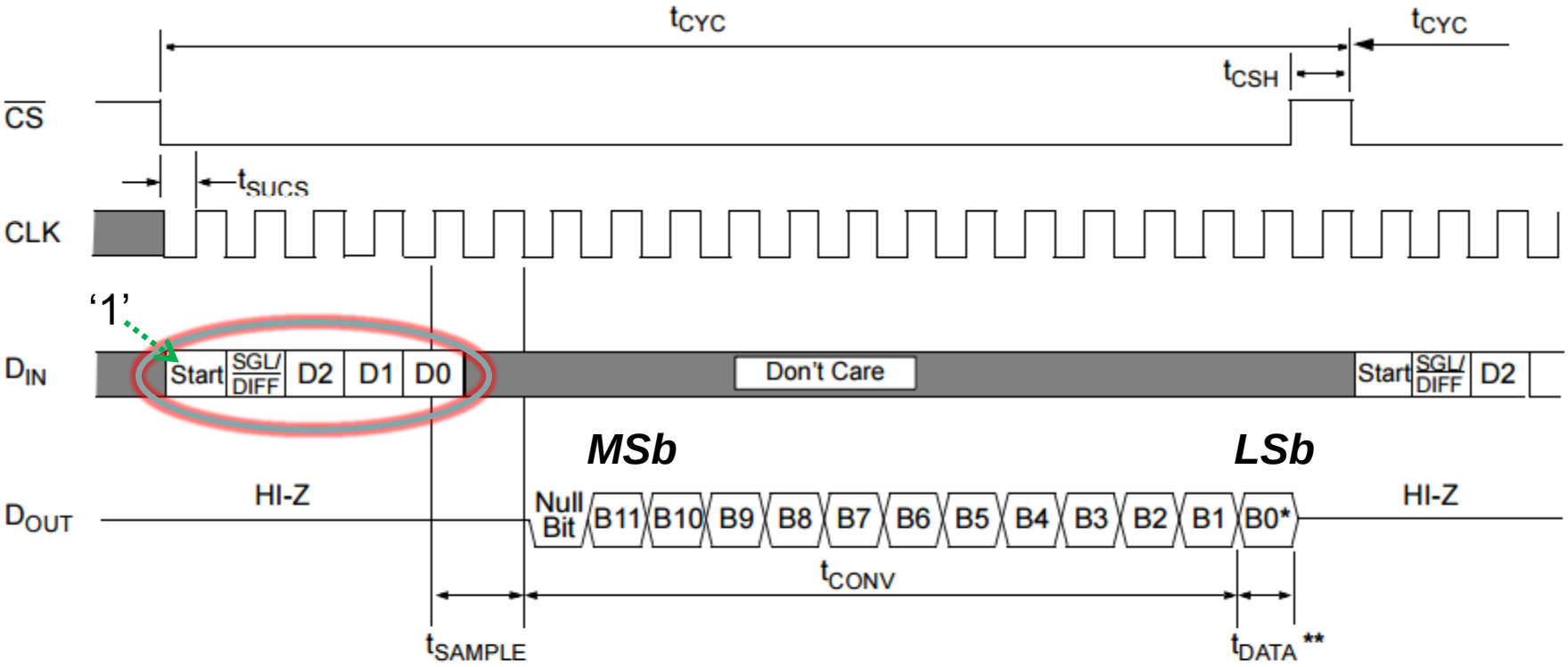
SPI, Synchronous Peripheral Interface (Motorola)

Implementation example on a 68HC11 uC

2 simples registers who exchange their data



MCP3204, read transfer



* After completing the data transfer, if further clocks are applied with $\overline{CS}$ low, the A/D converter will output LSB first data, followed by zeros indefinitely (see Figure 5-2 below).

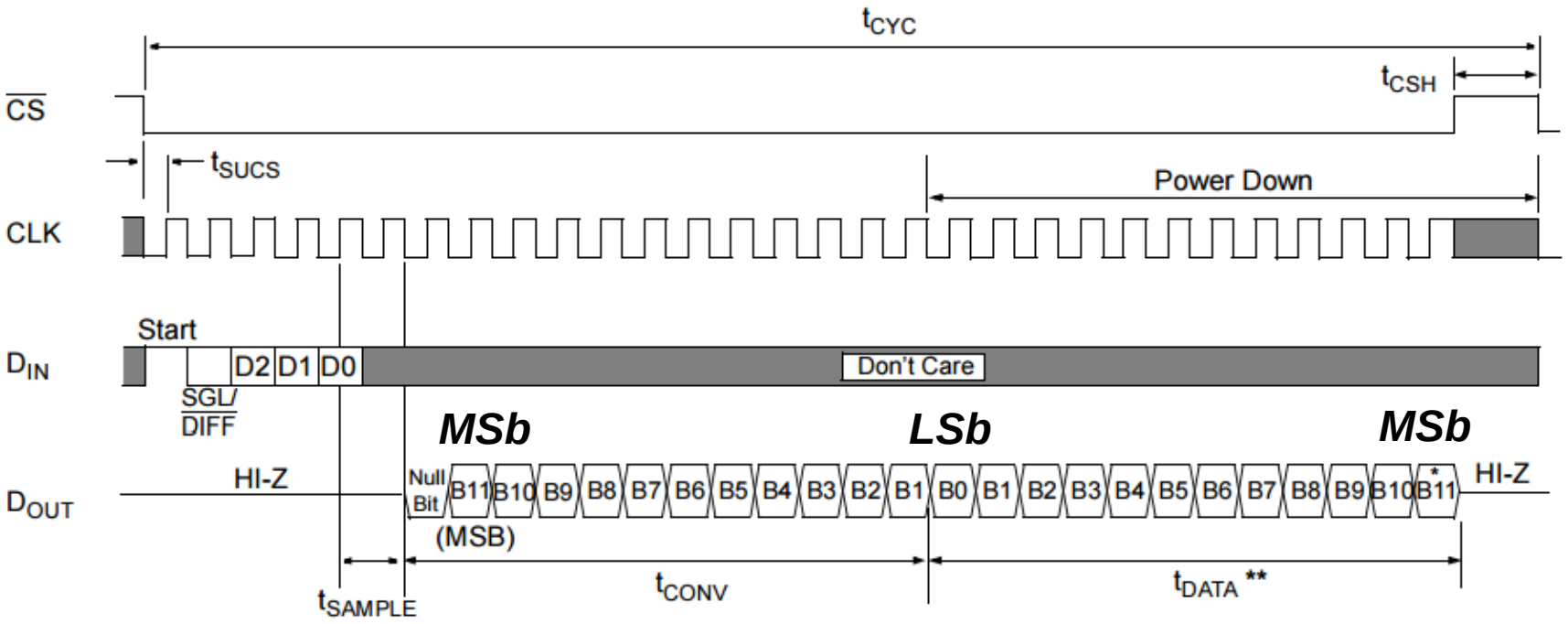
** t_{DATA}: during this time, the bias current and the comparator power down while the reference input becomes a high impedance node, leaving the CLK running to clock out the LSB-first data or zeros.

ADC, SPI protocol

- Selection of the channel to convert

Control Bit Selections				Input Configuration	Channel Selection
Single/ Diff	D2*	D1	D0		
1	X	0	0	single-ended	CH0
1	X	0	1	single-ended	CH1
1	X	1	0	single-ended	CH2
1	X	1	1	single-ended	CH3
0	X	0	0	differential	CH0 = IN+ CH1 = IN-
0	X	0	1	differential	CH0 = IN- CH1 = IN+
0	X	1	0	differential	CH2 = IN+ CH3 = IN-
0	X	1	1	differential	CH2 = IN- CH3 = IN+

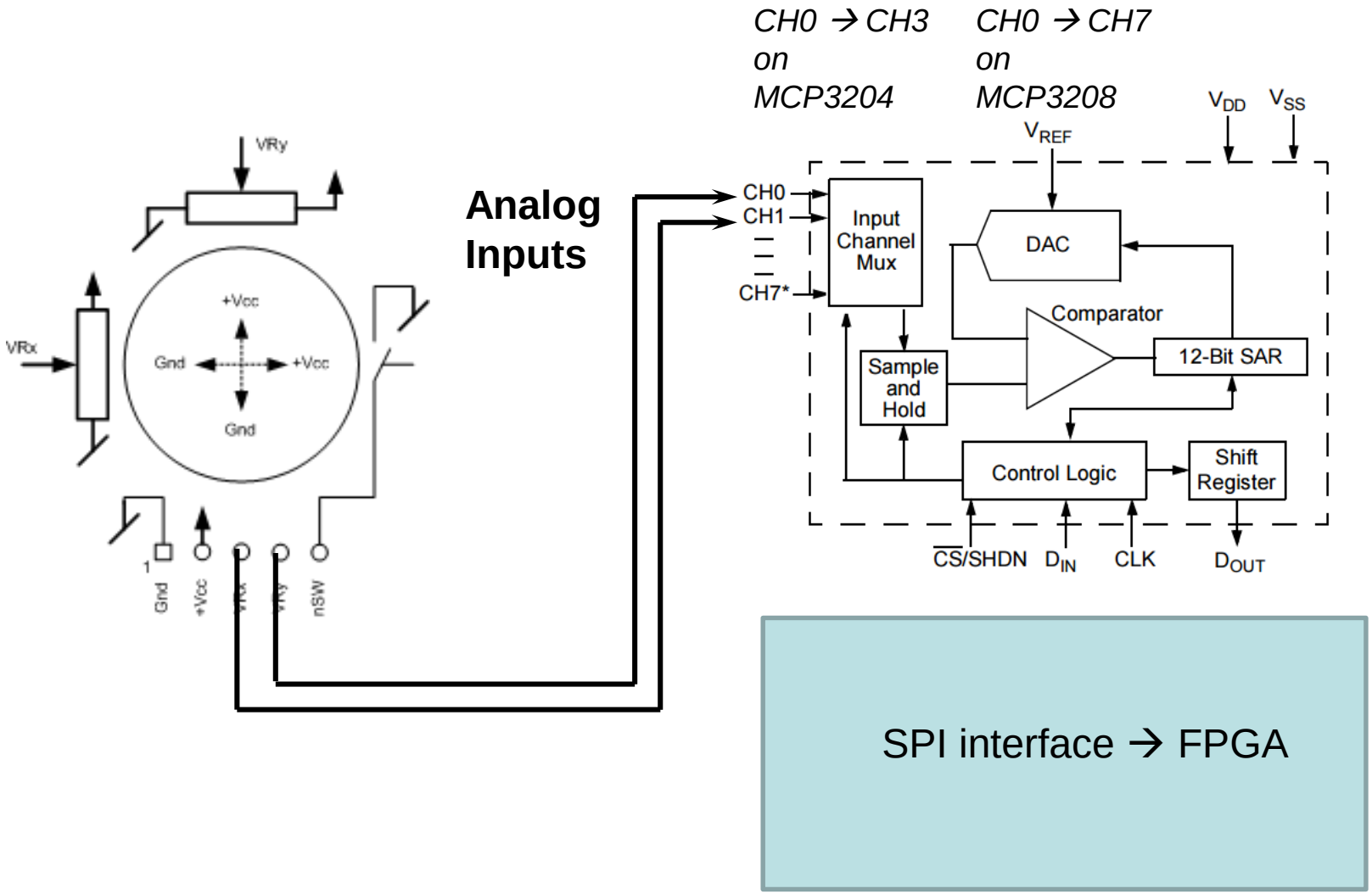
MCP3204, read transfer, clk not stopped



* After completing the data transfer, if further clocks are applied with $\overline{CS}$ low, the A/D converter will output zeros indefinitely.

** t_{DATA}: During this time, the bias circuit and the comparator power down while the reference input becomes a high impedance node, leaving the CLK running to clock out LSB first data or zeroes.

General system: Joystick to ADC, ADC to FPGA



SPI interface → FPGA

Programmable Interface: *SPI*_MCP3204

- Register's Model
  - Continuous access by the IP to read all the analog inputs through the SPI protocol
  - 4 registers for data values
  - 1 command register to start/stop
  - 1 status register
  - Accessed by Avalon bus
  - 16 bits data bus → 2 Bytes

Programmable Interface: Register's model

Reg n°	Master Addr	Name	Size	Function
0	….0	AD0	16 bits	Result of input 0
1	….2	AD1	16 bits	Result of input 1
2	….4	AD2	16 bits	Result of input 2
3	….6	AD3	16 bits	Result of input 3
4	….8	*ComReg*	16 bits	Command (not implement)
5	….A	*ComStat*	16 bits	Status (not to implement)
6	….C	-	-	-
7	….E	-	-	-

- To access to IP, the ***Base address of the IP*** needs to be added to the ***Master Addr.***
- The ***Base address of the IP*** is defined in **Qsys** at the implementation level
- The symbol of it is provide in **SBT** in ***system.h*** file

Programmable Interface: Black Box→ entity