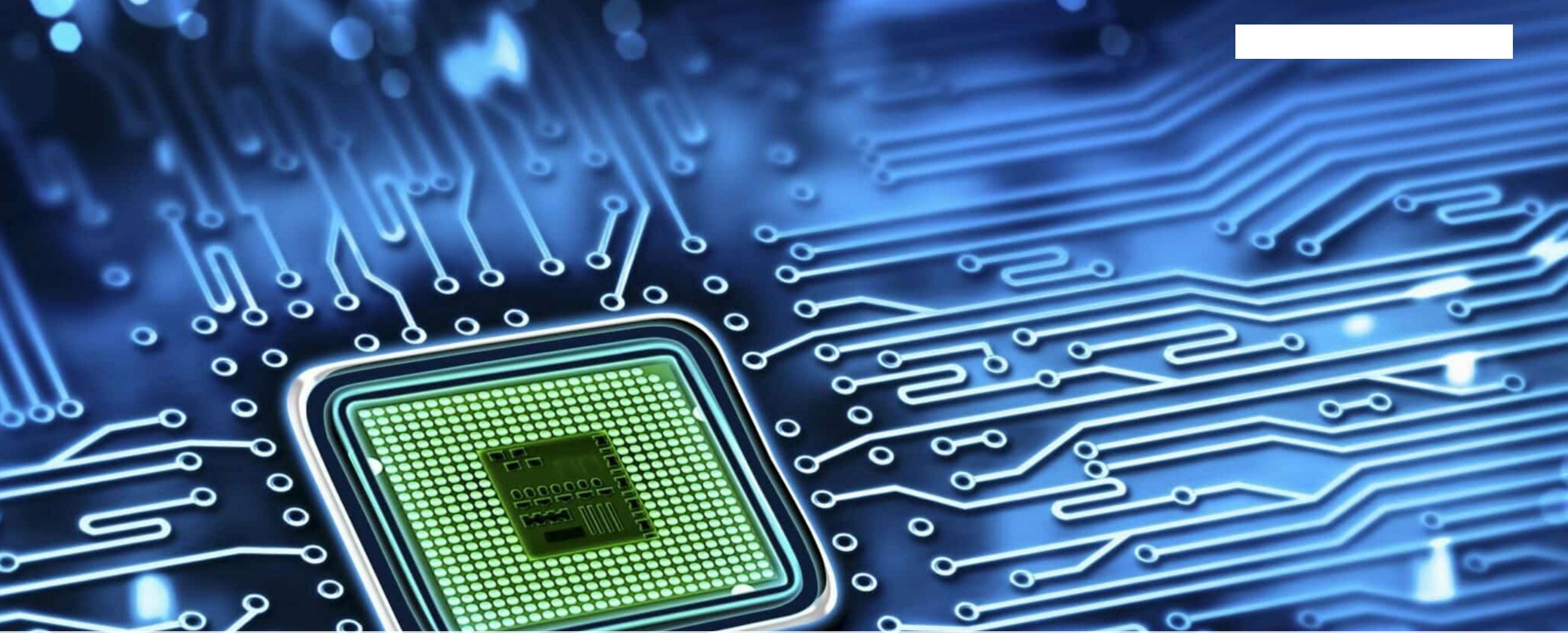
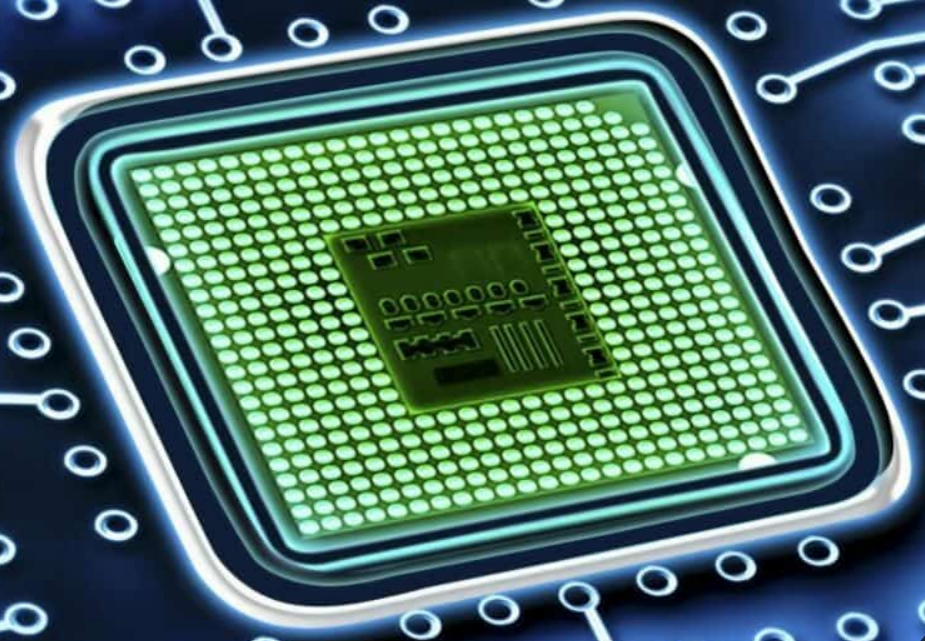


Présentation BE : Pilote Barre Franche

Présenté par : DJAROUD Mohamed

Encadré par : Pedro CARVALHO, Thierry PERISSE | M2 SME | 2022/2023

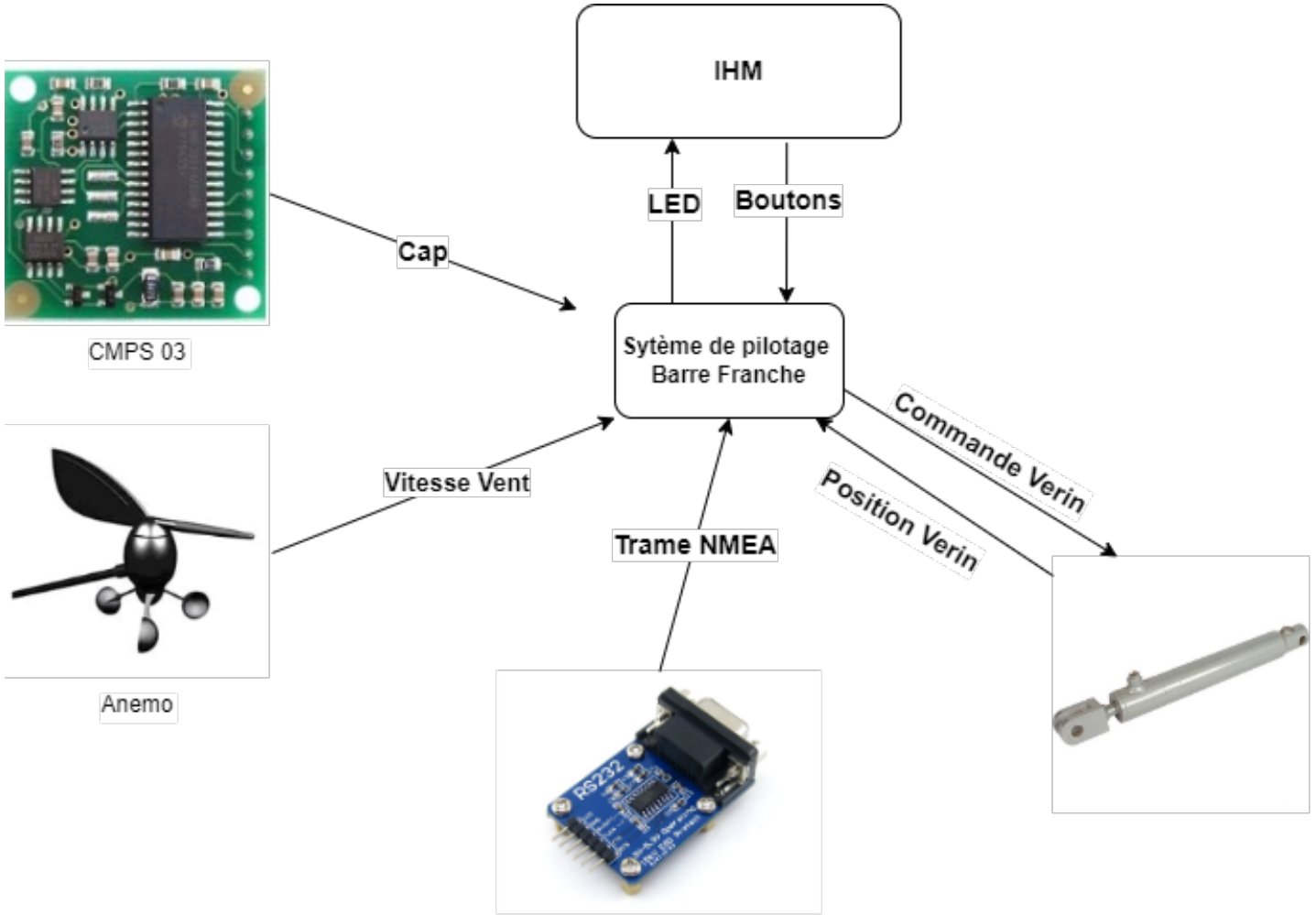
Plan



- INTRODUCTION
- PRESENTATION GENERALE DU PROJET
- FONCTION SIMPLE
- FONCTION COMPLEXE
- CONCLUSION

00 Introduction

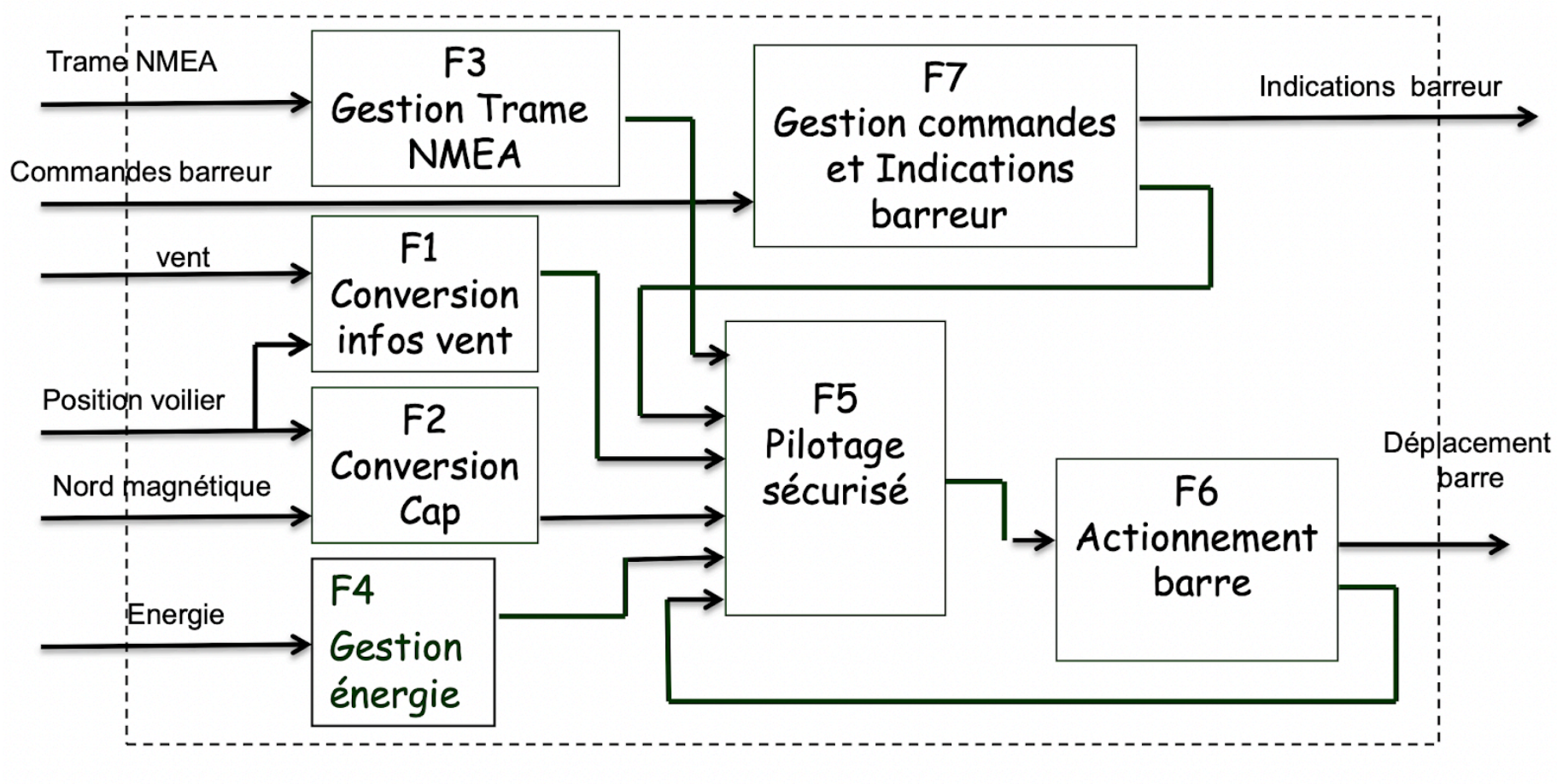
Diagramme de contexte



01

Présentation générale du projet

Exploration architecturale



Architecture de projet

02 Fonction simple

F1 : Conversion Info Vent

Blocs :

- Compteur de front montant
- Machine a état
- Bascule D
- Timer 1s

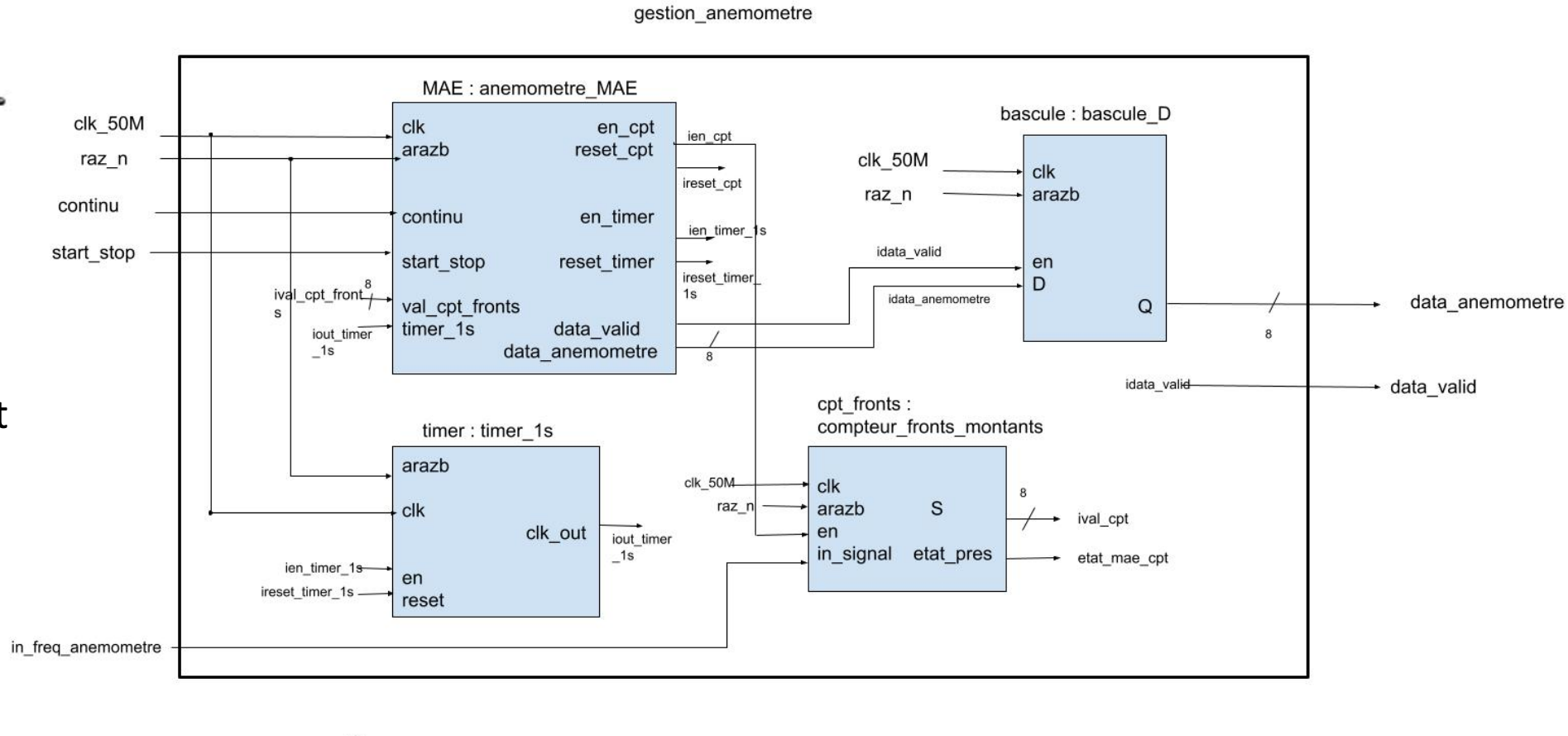
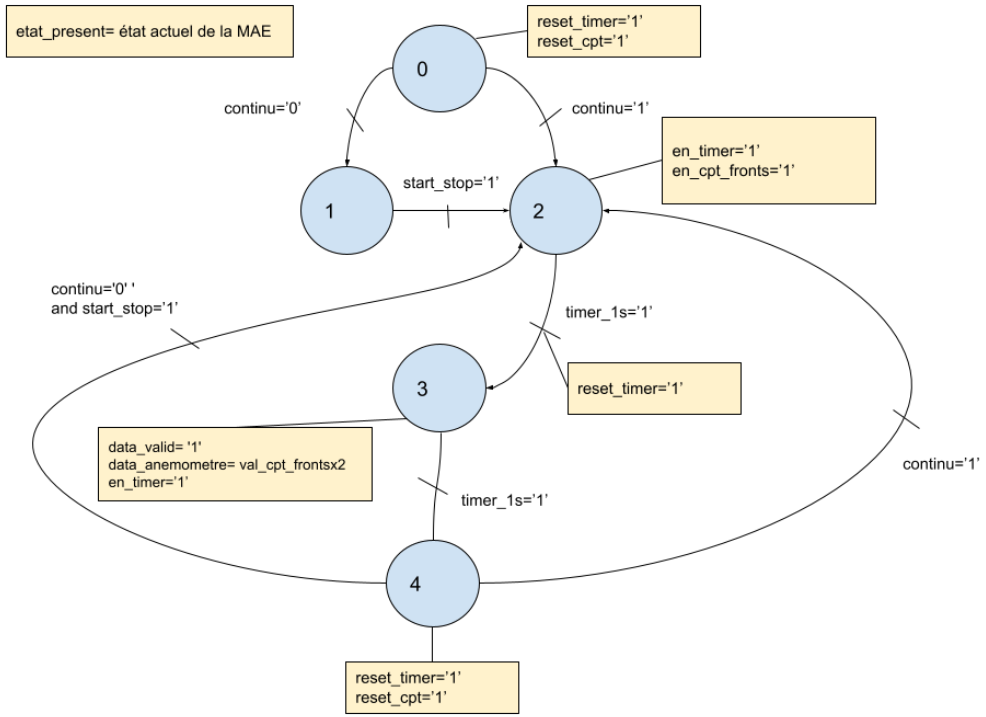
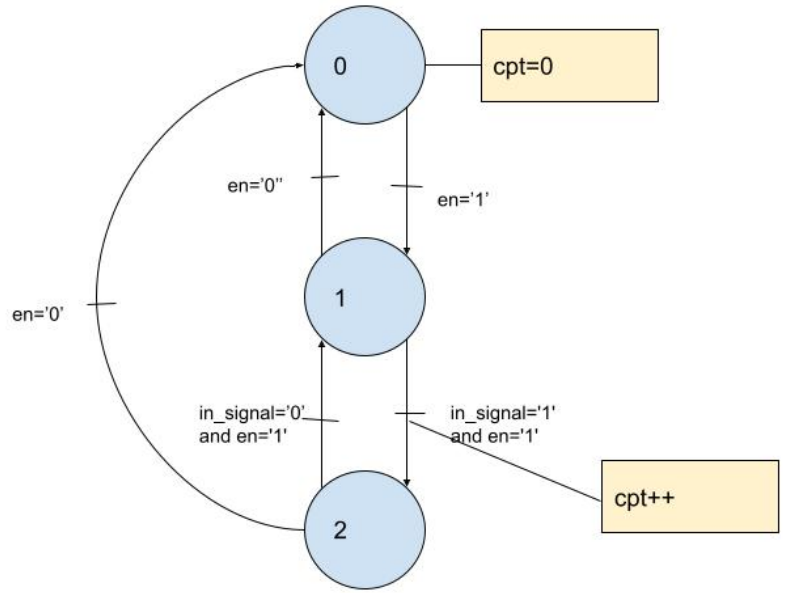


Schéma fonctionnel de l'Anémomètre

Détails des blocs



Machine a état de l'anémomètre



Machine a état du compteur de fronts montants

Implémentation du schéma bloc sur Quartus

- Création de plusieurs fichiers .vhd pour chaque bloc.
- Création d'un fichier BDF

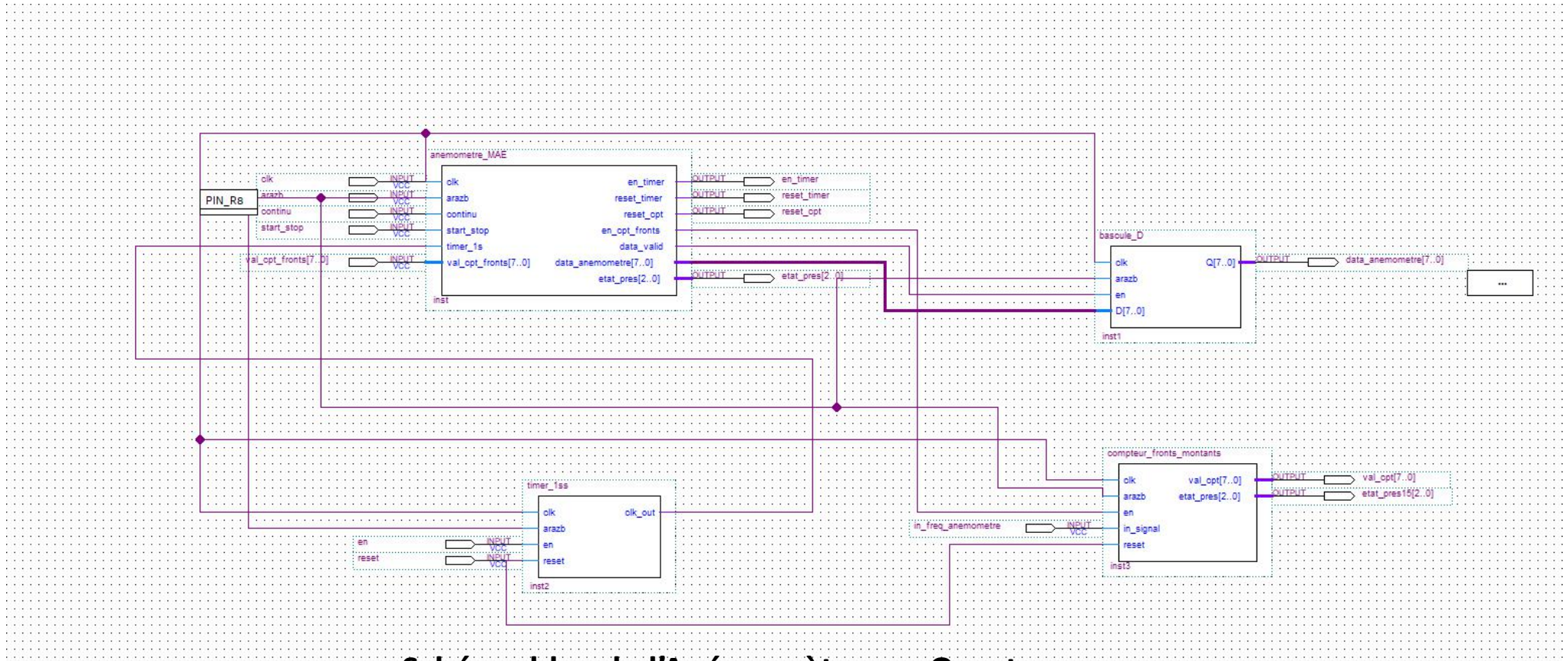
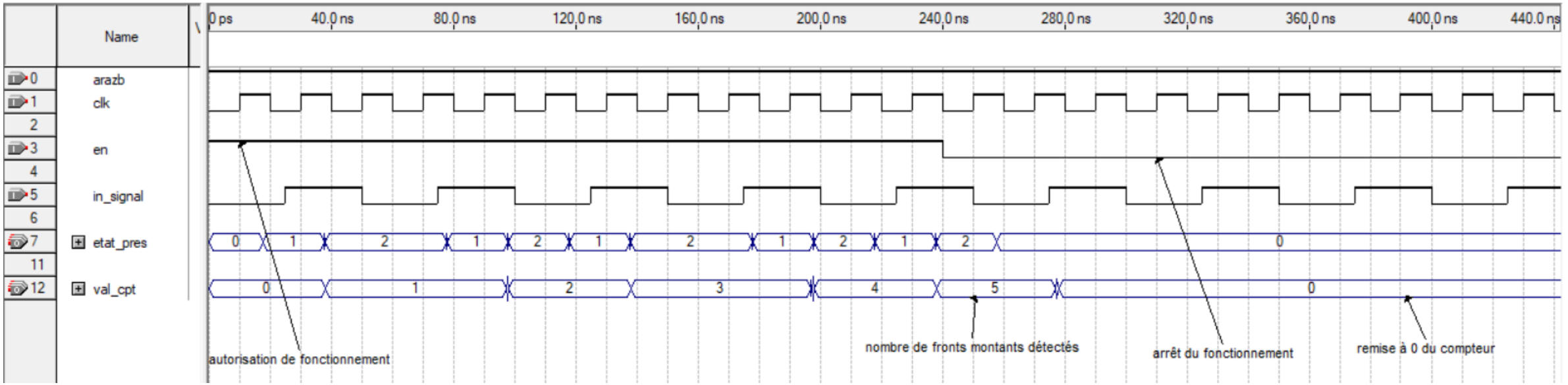


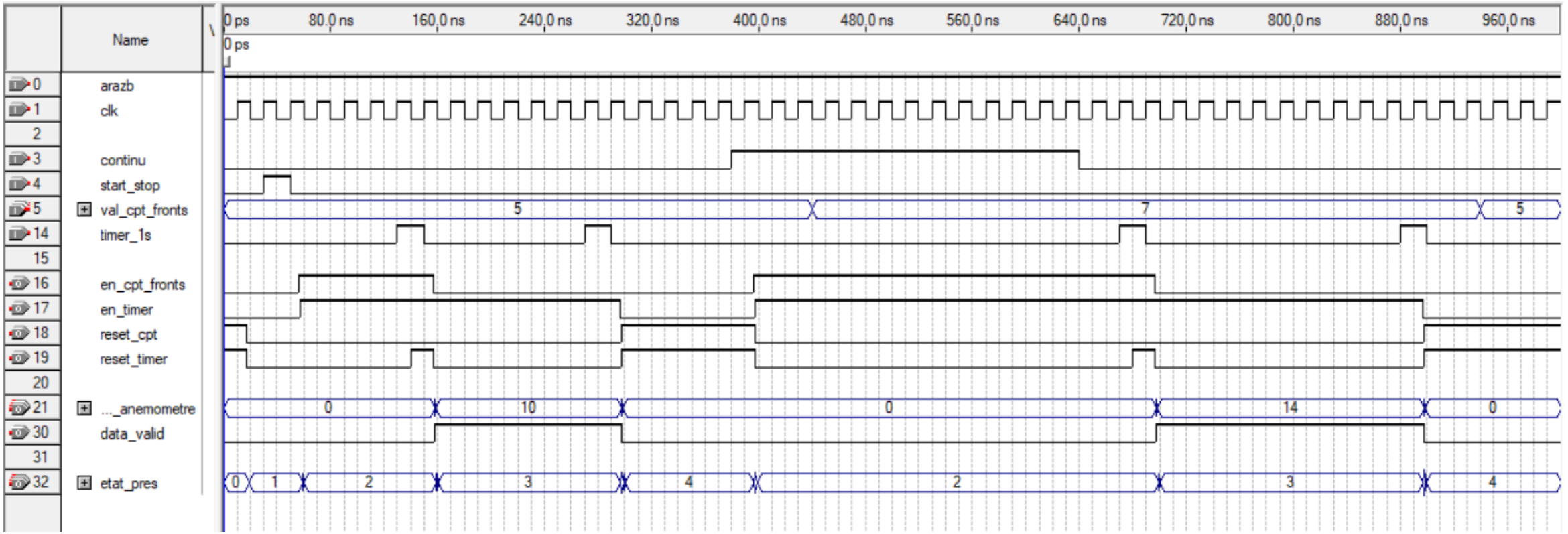
Schéma bloc de l'Anémomètre sur Quartus

Partie Simulation



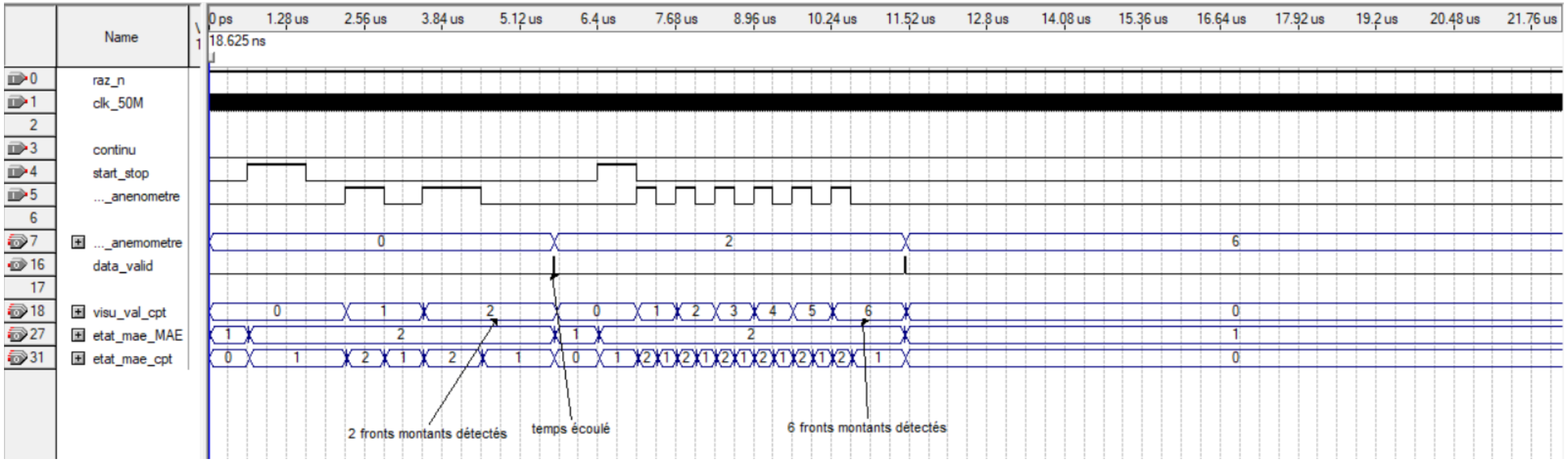
Simulation sur Quartus 9 du bloc compteur de fronts montants

Partie Simulation



Simulation sur Quartus 9 de la machine a état de l'anémomètre

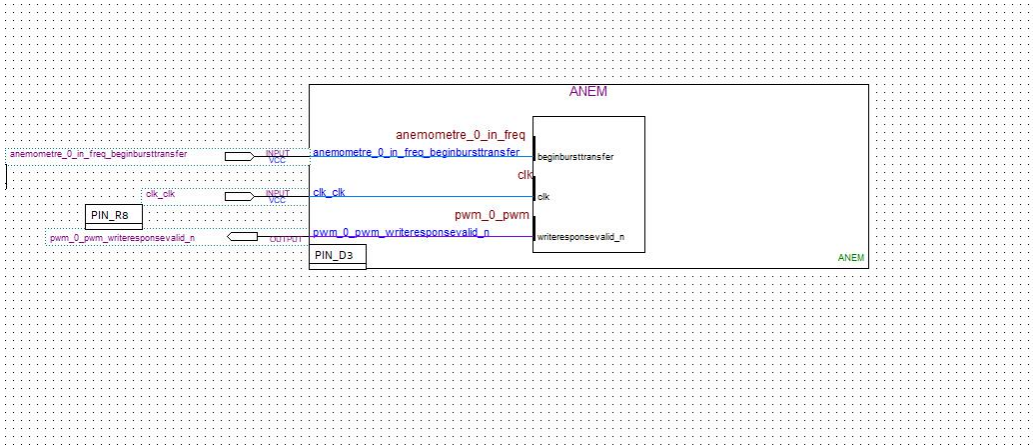
Partie Simulation



Simulation sur Quartus 9 du bloc de gestion de l'anémomètre

Conception du SOPC

☑		clk_0	Clock So…				
		clk_in	Clock Input	clk	exported		
		clk_in_reset	Reset Input	Double-click to export			
		clk	Clock Ou…	Double-click to export	clk_0		
		clk_reset	Reset Ou…	Double-click to export			
☑		nios2_gen2_0	Nios II Pr…				
		clk	Clock Input	Double-click to export	clk_0		
		reset	Reset Input	Double-click to export	[clk]		
		data_master	Avalon M…	Double-click to export	[clk]		
		instruction_master	Avalon M…	Double-click to export	[clk]		
		irq	Interrupt…	Double-click to export	[clk]	IRQ 0	
		debug_reset_requ…	Reset Ou…	Double-click to export	[clk]		
		debug_mem_slave	Avalon M…	Double-click to export	[clk]	0x0001_0800	0x0001_(
		custom_instruction…	Custom I…	Double-click to export			
☑		onchip_memory2…	On-Chip …				
		clk1	Clock Input	Double-click to export	clk_0		
		s1	Avalon M…	Double-click to export	[clk1]	0x0000_8000	0x0000_(
		reset1	Reset Input	Double-click to export	[clk1]		
☑		pwm_0	pwm				
		clock	Clock Input	Double-click to export	clk_0		
		avalon_slave_0	Avalon M…	Double-click to export	[clock]	0x0001_1030	0x0001_1
		reset	Reset Input	Double-click to export	[clock]		
		pwm	Conduit	pwm_0_pwm	[clock]		
☑		anemometre_0	anemom…				
		avalon_slave_0	Avalon M…	Double-click to export	[clock]	0x0001_1020	0x0001_1
		clock	Clock Input	Double-click to export	clk_0		
		reset	Reset Input	Double-click to export	[clock]		
		in_freq	Conduit	anemometre_0_in_freq	[clock]		
☑		jtag_uart_0	JTAG UA…				



Conception du SOPC sur Platform Designer

03 Fonction complexe

Fonction complexe – Gestion vérin partie VHDL

Le code vhdl de la fonction vérin comporte 4 fonctions principales :

- Bloc de génération PWM
- Bloc de Gestion butées
- Bloc de gestion du convertisseur MCP3201
- Bus Avalon

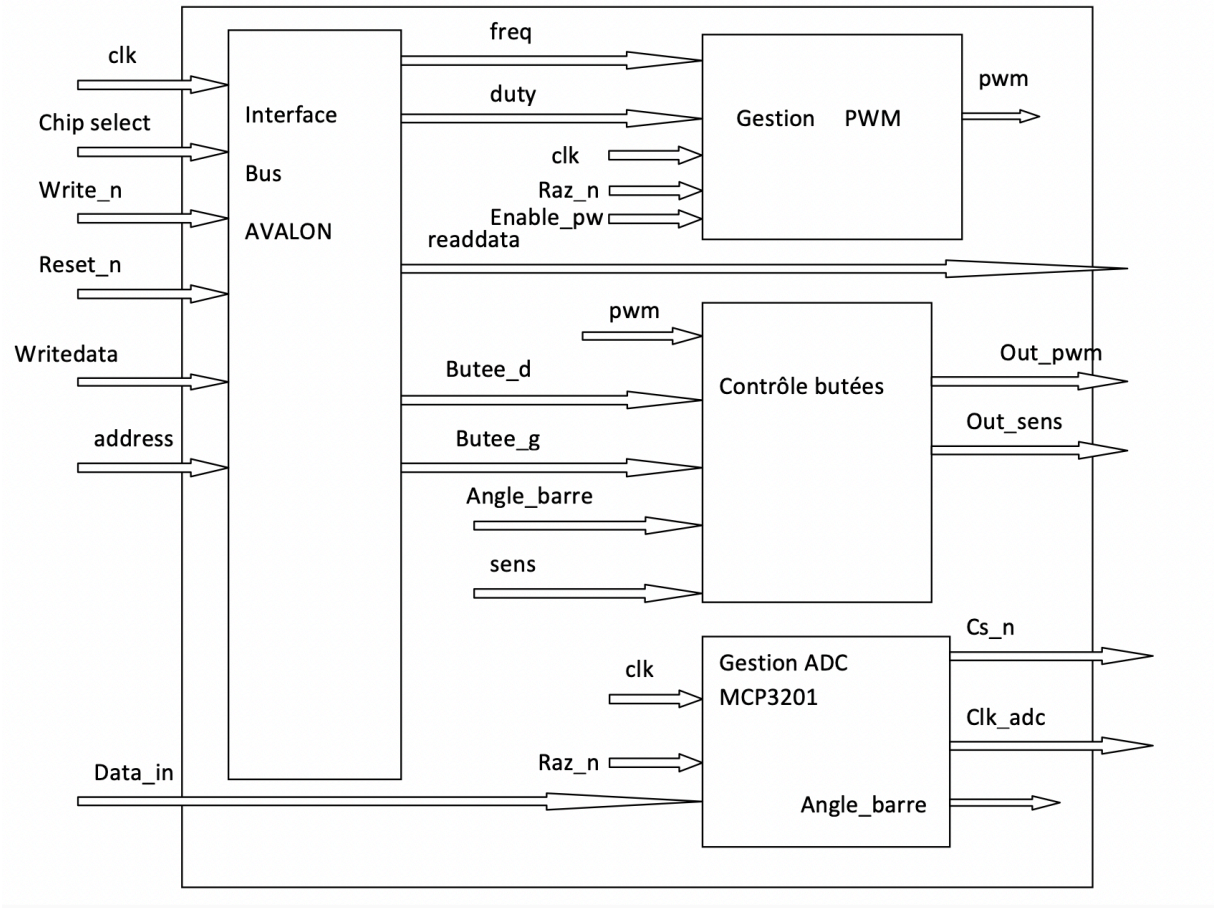


Schéma fonctionnel du Vérin

Gestion convertisseur MCP3201

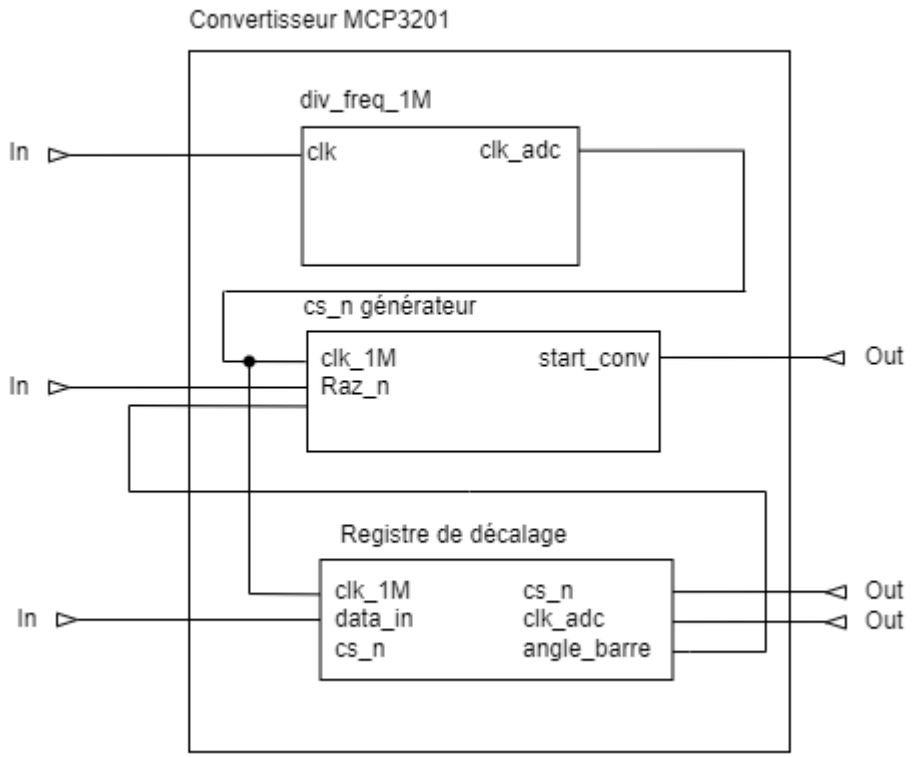


Schéma bloc de la partie gestion convertisseur MCP3201

Gestion convertisseur MCP3201

Simulation du convertisseur MCP 3201

```
process(clk_1M, raz_n)
variable cnt : integer range 0 to 16;
variable cnt2 : integer range 0 to 16;

begin

if raz_n = '0' then
   data_out <= x"000";
elsif falling_edge(clk_1M) then
   if start_conv ='0' and cs = '0' then
      cnt := cnt + 1;
      cnt2 := cnt2 + 1;
      if cnt >= 5 then
         if cnt2 <= 15 then
            decalage(16- cnt2) <= data_in;
         end if;
      end if;
   else
      cnt := 0;
      cnt2 := 0;
   end if;
end if;

data_out <= decalage;

end process;
```

Développement de l'interface Avalon

clk_0	Clock Source		
clk_in	Clock Input	clk	*exported*
clk_in_reset	Reset Input	Double-click to export	
clk	Clock Output	Double-click to export	clk_0
clk_reset	Reset Output	Double-click to export	
nios2_gen2_0	Nios II Processor		
clk	Clock Input	Double-click to export	**clk_0**
reset	Reset Input	Double-click to export	[clk]
data_master	Avalon Memory Mapped Master	Double-click to export	[clk]
instruction_master	Avalon Memory Mapped Master	Double-click to export	[clk]
irq	Interrupt Receiver	Double-click to export	[clk]
debug_reset_request	Reset Output	Double-click to export	[clk]
debug_mem_slave	Avalon Memory Mapped Slave	Double-click to export	[clk]
custom_instruction_m...	Custom Instruction Master	Double-click to export	
jtag_uart_0	JTAG UART Intel FPGA IP		
clk	Clock Input	Double-click to export	**clk_0**
reset	Reset Input	Double-click to export	[clk]
avalon_jtag_slave	Avalon Memory Mapped Slave	Double-click to export	[clk]
irq	Interrupt Sender	Double-click to export	[clk]
onchip_memory2_0	On-Chip Memory (RAM or ROM) Intel ...		
clk1	Clock Input	Double-click to export	**clk_0**
s1	Avalon Memory Mapped Slave	Double-click to export	[clk1]
reset1	Reset Input	Double-click to export	[clk1]
AVALON_VERIN_0	AVALON_VERIN		
clock	Clock Input	Double-click to export	**clk_0**
avalon_slave_0	Avalon Memory Mapped Slave	Double-click to export	[clock]
reset	Reset Input	Double-click to export	[clock]

Conception du SOPC sur Platform Designer

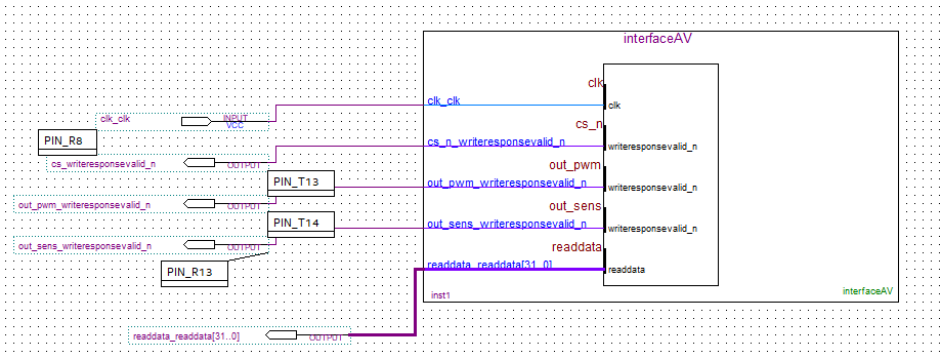


Schéma bloc de l'avalon Vérin

04 Conclusion

Conclusion

Bilan :

- Mise en pratique des compétences acquises durant la formation
- Pouvoir travailler sur un projet à partir du besoin
- Acquérir de nouvelles compétences en VHDL et C embarqué

Merci de votre attention