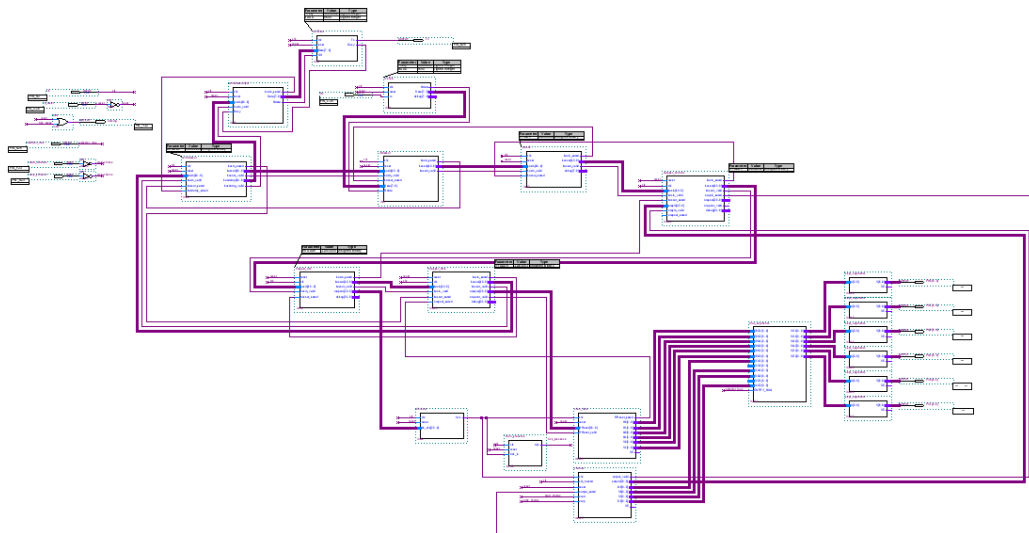




Rapport de Projet

Micro Architecture



L. Melchior - T. Gennuso

ENSIIE
Ivan Augé

2018

Table des Matières

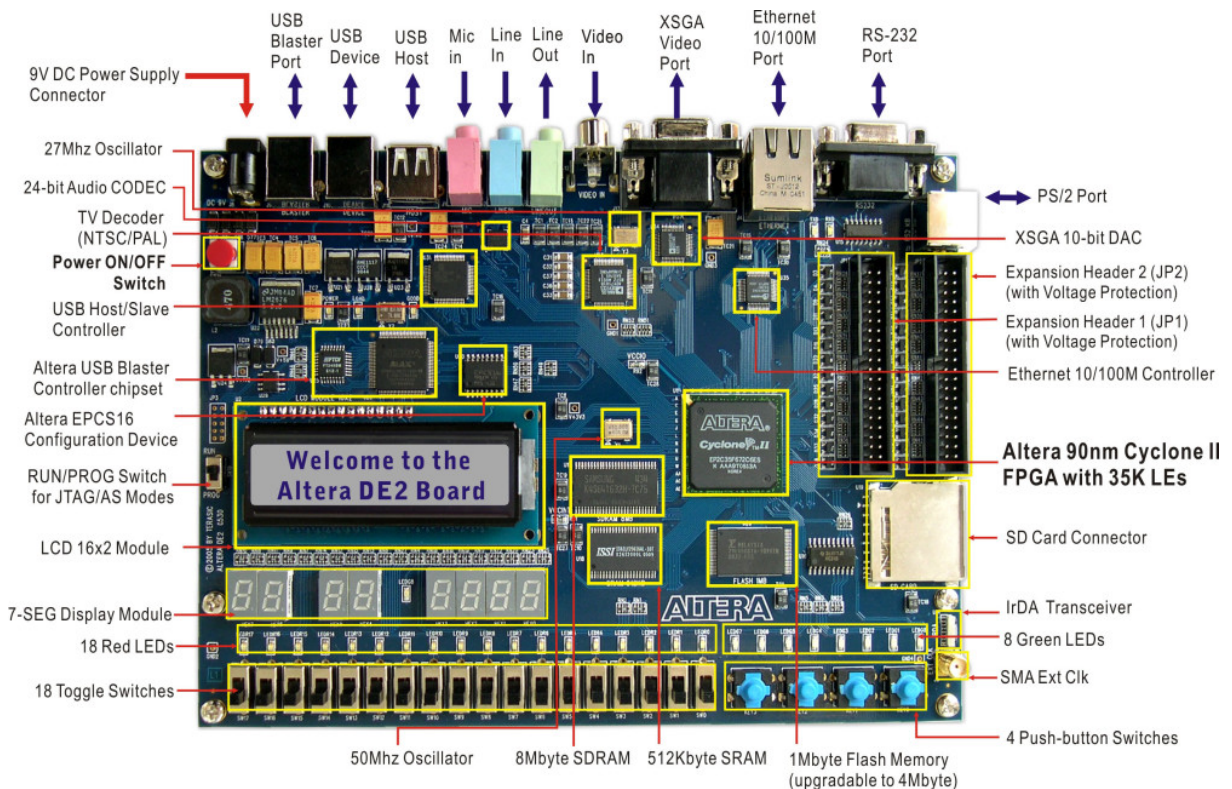
Table des Matières	3
1 Introduction	5
2 Manuel d'utilisation	7
2.1 Généralités	7
2.2 Fonctionnalités	7
2.2.1 Sur ordinateur	8
2.2.2 Sur la carte FPGA	8
2.3 Communication	8
2.3.1 Ordinateur	9
2.3.2 Carte FPGA	9
2.4 Connexion	9
3 Fonctionnement	11
3.1 Carte FPGA	11
3.2 Format des messages	11
3.2.1 Description matérielle	11
3.3 Détail du Chronomètre	12
3.3.1 Partie Opérative	12
3.3.2 Partie Contrôle	12

1 | Introduction

Ce document a pour but de familiariser le lecteur à l'utilisation de l'application *TimeMachine* développée par Louis Melchior & Tàzio Gennuso pour le projet de Micro Architecture de l'ENSIIE - S3.

L'application décrite fonctionne sur carte FPGA et communique avec un ordinateur exécutant un binaire issu de sources C fournies par Ivan Augé. Ce document tentera également d'éclaircir certains points techniques et solutions adoptées pour l'application.

2 | Manuel d'utilisation



2.1 Généralités

Notre application *TimeMachine* est développée sous langages vhdl et bdf. Elle est capable de communiquer sur le port RS232 d'une puce FPGA.

Un pilote, fourni par Ivan Augé, permet de communiquer avec l'application sur la carte FPGA depuis un ordinateur. Pour établir une connexion viable entre la carte FPGA et l'ordinateur, il est impératif « d'initialiser » la carte à l'aide du bouton 0 (KEY0) avant de d'initier la connexion avec l'ordinateur. Pour ce faire, garder ce bouton appuyé pendant environ 2 secondes. La carte est alors prête à initier une connexion avec l'ordinateur.

2.2 Fonctionnalités

TimeMachine propose une fonctionnalité d'horloge et de chronomètre connectés à un ordinateur.

Les prochaines sections décrivent leurs fonctionnements sur la carte FPGA et sur ordinateur.

2.2.1 Sur ordinateur

Les sources fournies doivent être compilées sur votre station de travail. Vous devez compiler uniquement les modules que vous souhaitez utiliser.

La ligne de commande typique est :

```
gcc busia.c ping-pong.c plus12.c set_tick.c set_clock.c chronos.c -o  
TimeMachine.bin -L/pub/readline/lib -lreadline -lhistory -lcurses -lrt
```

Une fois chose faite, vous pouvez commencer à communiquer avec *TimeMachine* via le port RS232 de la carte FPGA. Pour ce faire, référez vous à la section [connexion](#).

Cette compilation permet l'utilisation des modules de :

- gestion d'horloge,
- de réception des temps chronométré,
- d'addition calculée sur carte,
- de modification de la fréquence de l'horloge,
- et du chronomètre.

2.2.2 Sur la carte FPGA

L'affichage principal de « *TimeMachine* » est situé sur la carte FPGA, sur les afficheurs digitaux 7-segments.

Affichage

Le choix de l'affichage sur les 7 segments se fait à l'aide du switch 0 sur la carte FPGA. L'horloge s'affiche sur les 6 afficheurs 7-segments de gauche selon le schéma suivant : HH MM SS. Le chronomètre quant à lui utilise les afficheurs 3,4,5,6. L'affichage s'effectue selon le schéma suivant : M SS D.

Chronomètre

Le chronomètre dispose de deux boutons : un bouton départ, et un bouton stop. Le bouton de départ [KEY3] lance le chronomètre ; l'affichage en mode « chronomètre » affiche le temps écoulé depuis le relâchement de ce bouton¹. Pour arrêter le chronomètre, il faut appuyer sur le bouton stop [KEY2]. Dès l'arrêt du chronomètre, la valeur atteinte est envoyée à l'ordinateur² et l'affichage est ré-initialisé à 0-00-0.

2.3 Communication

Le code fourni dans le cadre de l'application « *TimeMachine* » est personnalisable. Il est possible de changer, notamment, la valeur de référence en Bauds pour les communications entre l'ordinateur et la carte électronique.

¹Le chronomètre ne peut pas mesurer un temps supérieur à 9 minutes, 59 secondes et 9 dixièmes. Si ce temps est atteint, le chronomètre boucle : il revient à la valeur 0-00-0.

²Pour consulter la valeur atteinte par le chronomètre, il faut avoir au préalable correctement connecté l'ordinateur à la carte. La valeur atteinte est alors affichée dans le terminal "chronos".

La personnalisation de ces valeurs est à effectuer à vos risques et périls. Elles sont toutefois envisageable pour un utilisateur avancé.

2.3.1 Ordinateur

Pour changer la fréquence de communication, il suffit de modifier les valeurs des variables BAUD dans les fichiers rs232in et rs232out dans les sources vhd. Attention toutefois à garder la cohérence entre les sources vhd et C.

2.3.2 Carte FPGA

Dans les sources C, il faut modifier la définition TTY_BAUD dans le fichier busia.c. Attention toutefois à garder la cohérence entre les sources vhd et C.

Il est possible de choisir l'interface de communication sortant de l'ordinateur (USB ou RS232).

2.4 Connexion

Pour connecter la carte et l'ordinateur il faut, au préalable, avoir initialisé la carte comme indiqué dans la section [généralités](#).

Il suffit ensuite d'exécuter le binaire généré par la compilation (telle que vue dans la section [sur ordinateur](#)).

Il est nécessaire de passer en argument l'interface au binaire. Ainsi, l'exécution de l'application se fait de la manière suivante :

```
./TimeMachine.bin /dev/ttyUSB0
```

On note ici que l'on utilise une connexion de type USB <-> RS232.

L'exécution de ce binaire lance dans des terminaux différents les différentes fonctionnalités prévues à la compilation.

3 | Fonctionnement

3.1 Carte FPGA

Cette partie tente d'expliquer l'architecture des composants développés dans la carte.

3.2 Format des messages

Message de :	Format du message
chronos (m, s, ds)	00—0 m1(4 bits) s2(4 bits) s1(4 bits) d(4 bits)
set_clock (hh, mm, ss)	h2(4bits) h1(4bits) m2(4bits) m1(4bits) s2(4bits) s1(4bits)
set_tick (nb)	nb(24 bits)

3.2.1 Description matérielle

Les composants dans la carte ont été organisés en un BUS dans le but d'avoir une multitude d'applications pouvant fonctionner en même temps.

Le principaux composants utilisés par le bus ont été fournis par Ivan Augé. Ce bus est donc issu du bus Busiac.

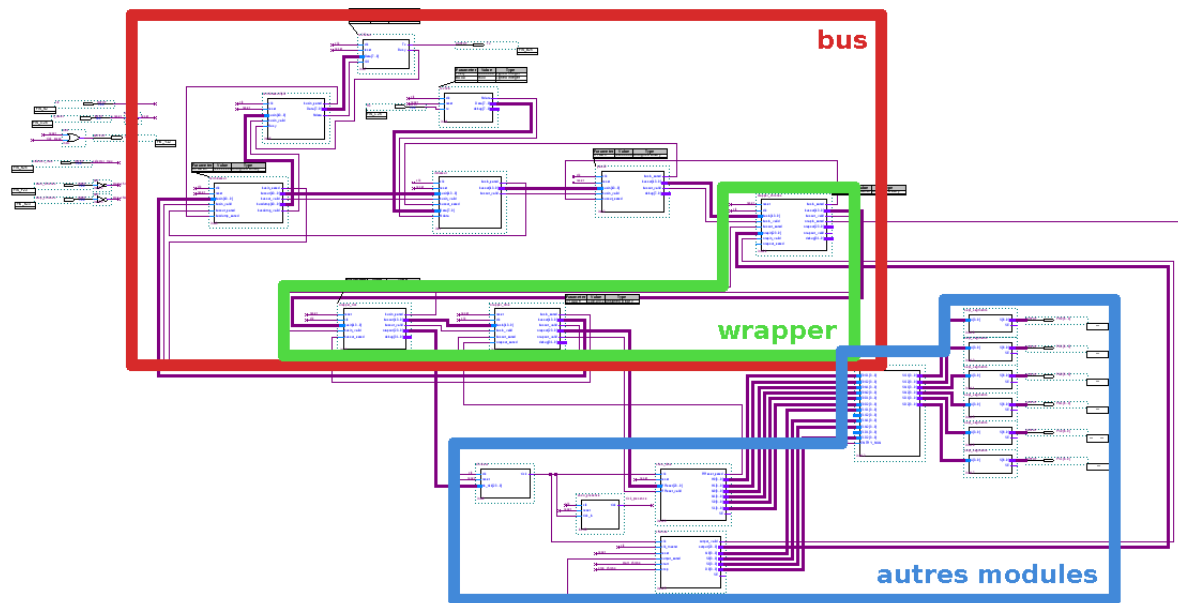


Figure 3.1: Organisation de Time Machine, avec les différents éléments mis en avant.

3.3 Détail du Chronomètre

Vous trouverez ici les schémas de la partie opérative du chronomètre ainsi que le schéma de l'automate décrivant la partie contrôle du chronomètre tel que nous l'avons implémenté dans la carte FPGA.

3.3.1 Partie Opérative

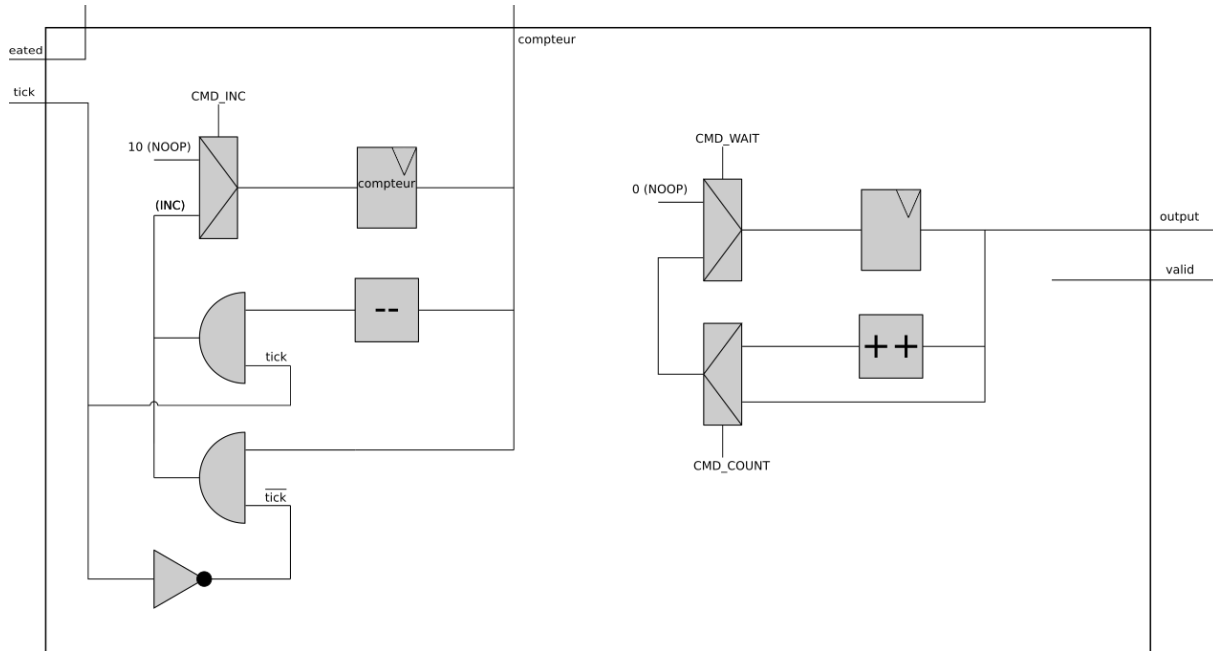


Figure 3.2: Partie opérative du chronomètre

3.3.2 Partie Contrôle

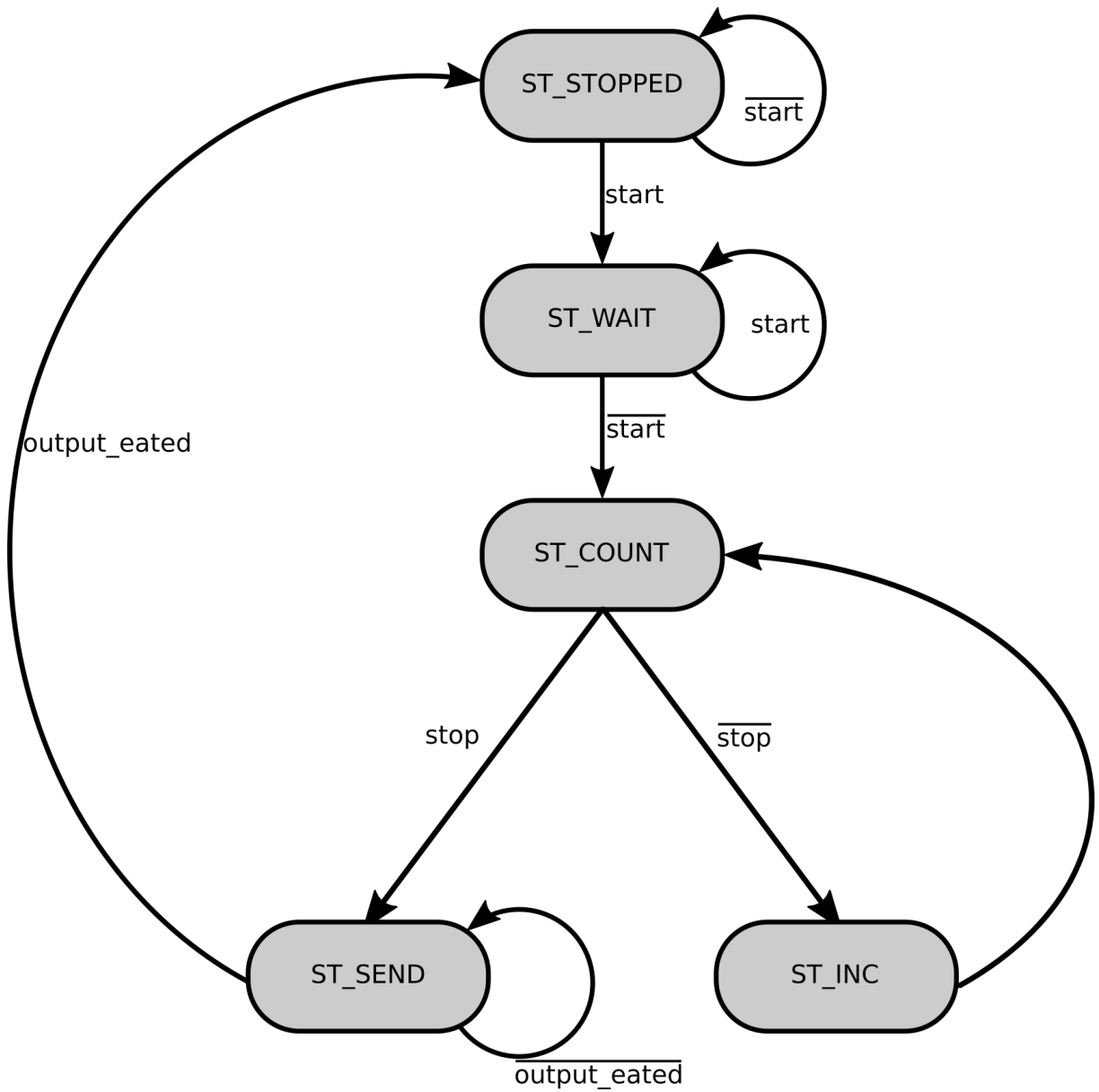


Figure 3.3: Partie contrôle du chronomètre