

UNIVERSITÉ PIERRE ET MARIE CURIE

PSESI

Projet Centrale DCC

Document Définition Interface et
Logique Enclenchement

Étudiant :

Maxime AYRAULT
3203694

Encadrant :

Julien DENOULET

25 mars 2017



1 Introduction

Présentation des 3 interfaces :

- Interface Homme Machine
 - Interface Centrale DCC et le circuit ferroviaire
 - Interface interne Centrale DCC - Moteur IXL
- et présentation des règles de la logique d'enclenchement.

2 Interface Homme Machine

Sur la carte

- Bouton gauche -> changement du paramètre
- Bouton droit -> changement de la valeur du paramètre

Dans le design

1. en sortie :
 - les 7 selecteurs de segment Ca -> Cg
 - les 8 selecteurs de digit An (7 down to 0)
 - octet adresse (*addr or aigui*) dans ADD
 - octet speed (*speed*) dans SPEED
 - octet fonction (*feat*) dans FCTN
2. en entrée :
 - une horloge à 100 MHz
 - un reset synchrone actif haut
 - la valeur du bouton gauche BTNL
 - la valeur du bouton droit BTNR

2.1 Interface commande train

8 vitesses

1. vitesse 0 -> "01100000"
2. vitesse 1 -> "01100010"
3. vitesse 2 -> "01100100"
4. vitesse 3 -> "01101000"
5. vitesse 4 -> "01101010"

6. vitesse 5 -> "01110000"
7. vitesse 6 -> "01110100"
8. vitesse 7 -> "01111111"

6 adresses de train

1. adresse 0 -> "00000000"
2. adresse 1 -> "00000001"
3. adresse 2 -> "00000010"
4. adresse 3 -> "00000011"
5. adresse 4 -> "00000100"
6. adresse 5 -> "00000101"
7. adresse 6 -> "00000110"

8 adresses de d'aiguillage

1. elles ne sont pas encore définies. mais de 10 à 18 pour faire simple.

X fonctions

1. phares on -> "10010000"
2. bruit moteur -> "10000001"
3. klaxon 1 long -> "10000100"
4. klaxon 1 bref -> "10100100"
5. bruit du vent -> "10100001"

2.2 Interface itinéraire

bla bla bla...

3 Interface Centrale DCC - Train, aiguillage et capteurs

3.1 Interface Train

Trames envoyées 4 par 4 séparées chacune par 400 us avec valeur 0

1er	trame	IDLE
2eme	trame	Vitesse
3eme	trame	Action
4eme	trame	IDLE

Une fois une trame envoyé, la repeter toutes les 55 ms
si aucune trame de chargée envoyer des trames idle

adresse	=>	"11111111"
donnée	=>	"00000000"
CRC	=>	"11111111"

Parametre à envoyer stocké dans *Master*
-> à chaque pression sur le bouton central changement de la valeur
des paramètres à envoyer, Les paramètre viennent de l'IHM et
remplacent la valeur IDLE et sont envoyés en continue tant que pas de
nouvelle prssion sur bouton C.

3.2 Interface Aiguillage

bla bla bla

3.3 Interface Capteurs

bla bla bla...

4 Interface Centrale DCC - Moteur IXL

La logique d'enclenchement est basée sur les notions de signaux, de circuit de voie et d'itinéraire ferroviaire.

Un *signal* est le moyen de communication entre le conducteur du train et la logique d'enclenchement. Il permet d'indiquer au conducteur l'état de la voie se trouvant devant le train (par définition un train ne recule jamais, il est nécessaire de changer de cabine de conduite pour changer la direction du train). Il existe beaucoup de signaux ferroviaire. Les trois plus importants sont :

- signal *rouge* qui signifie que le train doit s'arrêter devant le signal car la voie devant lui n'est pas sûre (présence d'un autre train, aiguillage en train d'être commandé, panne matériel...)
- signal *jaune* qui signifie qu'il y a un danger à proximité. Le train peut passer le signal mais à vitesse réduite seulement et doit être capable de s'arrêter sur une courte distance. Ceci permet de fluidifier le trafic.
- signal *vert* qui signifie que la voie devant le train est libre de tout danger et que le train peut poursuivre sa route en respectant les vitesses limites.

L'objectif des opérateurs consiste à toujours présenter un signal vert devant un train de façon à ce que le train n'est jamais à s'arrêter à cause d'un danger.

Le projet implémente le signal rouge et le signal vert. Le signal jaune ne peut pas être implémenté sans conducteur dans le train. Il n'y a pas de signaux sur le circuit ferroviaire, mais la notion de signaux sera quand même implémentée. Un signal *virtuel* sera créé pour chaque circuit de voie. Il autorisera ou pas l'entrée d'un train dans le circuit de voie auquel il est associé. Ceci permettra de commander les trains.

Un *Circuit de Voie* (CdV) représente la plus petite partie de voie sur lequel il est possible de détecter la présence d'un train. Pour un train réel, la détection de la présence d'un train s'effectue par les essieux du train. En effet, les essieux du train sont conducteur (métal), ils peuvent servir comme un *interrupteur* entre les 2 rails de la voie. Si une partie du train est présent alors les 2 rails sont au même potentiel (le circuit est dit fermé), sinon les rails sont a des potentiels différents (le circuit est dit ouvert).

Pour le projet, un Circuit de voie sera délimité par 2 capteurs de passage(sauf en zone d'aiguille où le circuit de voie sera délimité par les capteurs de chaque

branche). L'état d'occupation du CdV sera fourni pas le sens de parcours identifié par le capteur pour chaque train.

Un *itinéraire ferroviaire* représente une portion de voie entre 2 signaux. Il contient les mouvements d'aiguilles nécessaires pour que le train parcourt le chemin entre le signal de départ et le signal de sortie de l'itinéraire. Dans le système d'enclenchement, un itinéraire est représenté par une machine à états. Les états possibles sont :

- *commandé*. L'opérateur demande au train de parcourir l'itinéraire choisi. Il a par conséquent demandé la formation de l'itinéraire au système d'enclenchement. Si les conditions sur les aiguilles à manoeuvrer pour former l'itinéraire sont sûres, le système d'enclenchement passe à l'état de formation de l'itinéraire sinon il est rejeté.
- *formé*. Les aiguilles sont positionnées dans la bonne direction. Il n'est plus possible de bouger les aiguilles tant que le train se situe dans une certaine zone.
- *détruit*. Le train a parcouru l'itinéraire (il a passé le signal de sortie), l'itinéraire peut être détruit. Ou l'opérateur a demandé la destruction de l'itinéraire (en cas d'erreur par exemple) Dans ce cas, si le train ne se trouve pas trop proche du signal d'entrée alors l'itinéraire peut être détruit sinon la demande de destruction est rejetée. Ceci permet d'autoriser le parcours sur les aiguilles et la zone d'un autre train.

L'interface entre la Centrale DCC et le Moteur IXL est bi-directionnelle.

Dans le sens Centrale vers IXL, les informations envoyées sont :

- les commandes d'itinéraire
- l'information des capteurs de passage
- l'état des aiguilles (gauche/droite)

Dans le sens IXL vers Centrale, les informations envoyées sont :

- les commandes des aiguilles
- l'état des itinéraires (commandé/formé/détruit)
- l'état des signaux (rouge/vert)

4.1 Commande itinéraire

bla bla bla...

4.2 Information capteur de passage

La déclaration VHDL

```
type train_addr is 7 downto 0 ;
type train_dir is bit ;

type sensor is
  record
    addr : train_addr;
    dir  : train_dir;
  end record;

type SE_state is array (31 downto 0) of sensor;
```

4.3 État des aiguillages

L'état d'une aiguille est déterminé par la combinaison entre les 2 capteurs d'aiguilles. Une aiguille peut avoir 3 états :

- *gauche* lorsque le contact gauche est à l'état 1
- *droite* lorsque le contact droit est à l'état 1
- *en mouvement* lorsqu'aucun des 2 contacts ne sont à 1

Il n'y a pas de capteur fournissant l'état d'une aiguille sur la maquette. Par conséquent, l'état d'une aiguille correspondra à l'état de la dernière commande envoyée à l'aiguille. L'état *en mouvement* ne sera pas considéré.

L'état d'une aiguille sera représenté par 1 bit avec la convention

- 0 = droite
- 1 = gauche

Il y a 4 aiguilles sur le circuit. L'indice du tableau correspond au numéro de l'aiguille.

La déclaration VHDL

```
type Sw_state is array (7 downto 0) of bit;
```

4.4 État des itinéraires

Les états possibles pour un itinéraire sont ; commandé, formé, détruit et rejeté. Il sera représenté par 2 bits avec la convention

- 00 = commandé
- 01 = formé
- 10 = détruit
- 11 = rejeté

Il y a 32 itinéraires possibles. L'indice dans le tableau correspond au numéro d'itinéraire.

La déclaration VHDL

```
type Ro_state_t is 3 downto 0 ;

type Ro_state is array (31 downto 0) of Ro_state_t;
```

4.5 Commande des aiguillages

bla bla bla...

4.6 État des signaux

bla bla bla...

4.7 État des Circuits de Voie

Chaque circuit de voie possède 2 états; *Libre* ou *Occupé*. De façon à pouvoir effectuer la gestion des trains, il est nécessaire d'identifier quel train occupe le circuit de voie. Il y a 6 trains au maximum. Les états sont codés come suit :

- libre : 0x07
- occupé : numéro du train 0x01 à 0x06

Il y a 20 circuits de voie sur la maquette de train. L'indice du tableau correspond au numéro du Circuit de Voie.

5 Logique Enclenchement

Cette section présente la logique d'enclenchement, sa justification et les conversions de nommage utilisées dans les équations.

5.1 Circuit de Voie

Un Circuit de Voie est la section de voie se trouvant entre 2 capteurs de présence consécutifs. Un capteur de présence permet de connaître, le numéro du train et son sens de déplacement. Le sens de déplacement va permettre de savoir quel Circuit de voie occupé par le train. Deux directions sont arbitrairement déclarées pour chaque capteur ; Up et Down.

Les convention de nommage des éléments :

- SE_Up_nn, 1 lorsqu'un train vient de passer le capteur numéro nn dans le sens Up
- SE_Do_nn, 1 lorsqu'un train vient de passer le capteur numéro nn dans le sens Down
- TC_nn, 1 lorsque le Circuit de Voie est libre (aucun train présent sur le CdV), 0 lorsque le circuit de voie est occupée. Cette logique va dans le sens de la sécurité, en cas de perte de courant ou de dysfonctionnement, il n'est pas possible de savoir si le Circuit de Voie est occupé ou libre, donc on choisit la valeur la plus sécuritaire ; Occupé même si aucun train n'est présent.

À l'initialisation, tous les Circuits de Voie sont libres.

Le Circuit de voie TC_nn est occupé si une des condition suivantes est vérifiée :

- le capteur SE_Up_nn passe à 1, un train vient de rentrer dans le sens Up
- le capteur SE_Do_nn passe à 1, un train vient de rentrer dans le sens Down
- le Circuit de Voie est occupé et le train n'est pas sorti dans le sens Up ou Down

Il faut inverser cette logique car lorsque le Circuit de voie est occupé, on cherche à avoir une valeur 0.

L'équation générale d'occupation d'un Circuit de Voie est :

$$\begin{aligned} \text{TC_nn} &\leftarrow \text{NOT SE_Up_nn AND NOT SE_Do_nn} \\ &\quad \text{AND (TC_nn OR SE_Do_nn OR SE_Up_nn)}; \end{aligned}$$

5.2 Signaux

Il existe deux types de signaux ; les signaux d'espacement qui garantissent que 2 trains ne vont pas se percuter et les signaux de manoeuvre qui garantissent qu'un train ne va pas parcourir une aiguille qui n'est pas contrôlée.

Les convention de nommage des éléments :

- SI_nn, 1 lorsque le signal est Vert, 0 pour afficher un signal Rouge
- À l'initialisation, tous les signaux sont au Vert.

Un Signal d'espacement passe :

- au Rouge, lorsque le signal d'espacement ou le signal de manoeuvre est au Rouge
- au Vert, lorsque le signal d'espacement et le signal de manoeuvre sont au Vert

L'équation générale d'un signal est :

$$SI_nn \leftarrow SI_SP_nn \text{ AND } SI_IX_nn ;$$

5.3 Signal d'espacement des trains

Il est dangereux d'avoir 2 trains dans le même Circuit de Voie car rien ne permet de garantir que les 2 trains ne vont pas se percuter. Il n'est pas certain qu'un conducteur puisse arrêter le train devant un signal car la distance de freinage entre le signal et le Circuit de Voie à protéger n'est pas toujours suffisante. C'est pourquoi, il est nécessaire de toujours avoir un Circuit de Voie libre entre 2 trains. Ce Circuit de Voie est nommé, *CdV tampon*. Dans le cas de 2 trains qui se suivent, il est nécessaire que la longueur du Circuit de Voie soit plus grande que la distance d'arrêt du train suiveur. Dans le cas de 2 trains en sens inverse, il est nécessaire de s'assurer que la longueur du Circuit de Voie tampon est égale à 2 fois la longueur d'arrêt d'un train.

L'entrée dans un Circuit de Voie est protégée par un signal dit *d'espacement*.

Les convention de nommage des éléments :

- SI_SP_nn, 1 lorsque le signal est Vert, 0 pour afficher un signal Rouge
- À l'initialisation, tous les signaux sont au Vert.

Un Signal d'espacement passe :

- au Rouge, lorsque le TC_nn est occupé ou le TC_mm est occupé
- au Vert, lorsque les TC_nn et TC_mm sont libres.

L'équation générale d'un signal d'espacement est :

SI_SP_nn <- TC_nn AND TC_mm;

5.4 Signal de manoeuvre

Un signal de manoeuvre garantit qu'un train peut traverser une zone d'aiguillage sans danger. C'est à dire que les aiguilles de la zones sont toutes enclenchées.

5.5 Zone d'aiguille

Une aiguille peut être commandée si elle n'est pas enclenchée et si il n'y a pas de train sur la zone

5.6 Itinéraire

Bla bla bla...

5.7 zone d'aiguille

Bla bla bla...

Références