



Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Federal Department of the Environment, Transport,
Energy and Communications DETEC

Swiss Federal Office of Energy SFOE
Energy Research and Cleantech Division

Délivable 3 du 30/09/2022

COSTAM Project

D3 – Report on modified STATCOM



Haute école d'ingénierie et d'architecture Fribourg
Hochschule für Technik und Architektur Freiburg

HEIG VD | **IESE**
Institut d'Énergie
et Systèmes
électriques

SIL

SERVICES INDUSTRIELS LAUSANNE

Date: 30/09/2022

Lieu: Yverdon-les-Bains

Prestataire de subvention:

Office fédéral de l'énergie OFEN
Section Recherche énergétique et cleantech
CH-3003 Bern
www.ofen.admin.ch

Bénéficiaires de la subvention:

HEIA-FR
Bd. De Pérolles 80, CH-1700 Fribourg
<https://www.heia-fr.ch>

HEIG-VD
Route de Cheseaux 1, CH-1401 Yverdon-les-Bains
<https://heig-vd.ch>

Auteur(s):

Guillaume Courteau, HEIG-VD, guillaume.courteau@heig-vd.ch
Simon Kissling, HEIG-VD, simon.kissling@heig-vd.ch
Mokhtar Bozorg, HEIG-VD, mokhtar.bozorg@heig-vd.ch
Mauro Carpita, HEIG-VD, Mauro.Carpita@heig-vd.ch

Suivi du projet à l'OFEN:

Dr. Michael Moser, michael.moser@bfe.admin.ch

Numéro du contrat de l'OFEN: SI/502069-01

Les auteurs sont seuls responsables du contenu et des conclusions du présent rapport.



TABLE DES MATIÈRES

1	INTRODUCTION.....	7
2	CONCEPTION HARDWARE.....	7
2.1	STRATÉGIE DE PILOTAGE	8
2.2	SINGLE STAR BRIDGE CELL (SSBC)	8
2.2.1	Armoire du SSBC.....	9
2.3	DOUBLE STAR CHOPPER-CELLS (DSCC)	11
2.3.1	Armoire du DSCC	12
3	CONCEPTION FIRMWARE	14
3.1	CARTE CLÉOPÂTRE.....	15
3.1.1	Structures	15
3.1.1.1	S_ArmsControl	16
3.1.1.2	S_Measures	17
3.1.1.3	S_Pll.....	18
3.1.1.4	S_Command.....	19
3.1.1.5	S_ContactorStatus.....	20
3.1.1.6	S_Battery.....	20
3.1.1.7	S_UserCtrl.....	20
3.1.1.8	S_FlagError	21
3.1.1.9	Régulateurs	21
3.1.2	Machine d'état	24
3.1.3	Interruption ADC	25
3.1.3.1	Régulation	26
3.2	CARTE ALEXANDRE.....	27
3.2.1	Structures	27
3.2.1.1	S_Slave.....	27
3.2.1.2	S_Measures	28
3.2.1.3	S_FlagError	28
3.2.1.4	S_Control	29
3.2.2	Interruption de contrôle (XINT4)	30
3.2.3	Interruption TIMER0 (led rouge).....	30
3.2.4	Stratégie de gestion des erreurs.....	31
3.2.4.1	Mesures de tension.....	31
4	CONCLUSION.....	32
5	BIBLIOGRAPHIE.....	32



LISTE DES FIGURES

FIGURE 2-1 : PROTOTYPES DES CONVERTISSEURS BASSE TENSION	7
FIGURE 2-2 : STRATÉGIE HARDWARE DE PILOTAGE DES CONVERTISSEURS.....	8
FIGURE 2-3 : SCHÉMA ÉLECTRIQUE SIMPLIFIÉ DU SSBC	9
FIGURE 2-4 : POSITIONNEMENT MÉCANIQUE DES DIFFÉRENTES CARTES DANS L'ARMOIRE SSBC.....	9
FIGURE 2-5 : CÂBLAGE DE L'ARMOIRE SSBC.....	10
FIGURE 2-6 CÂBLAGE DE LA PLAQUE DE FOND SSBC	11
FIGURE 2-7 : SCHÉMA ÉLECTRIQUE SIMPLIFIÉ DU DSCC.....	11
FIGURE 2-8 : POSITIONNEMENT MÉCANIQUE DES DIFFÉRENTES CARTES DANS L'ARMOIRE DSCC	12
FIGURE 2-9 : CÂBLAGE DE L'ARMOIRE DSCC	13
FIGURE 2-10 CÂBLAGE DE LA PLAQUE DE FOND DSCC.....	14
FIGURE 3-1 : STRATÉGIE DE FONCTIONNEMENT AVEC LA MODULATION PSPWM.....	15
FIGURE 3-2 : STRUCTURES DES DIFFÉRENTS OBJETS DANS LE FIRMWARE CLÉOPÂTRE 1/3	22
FIGURE 3-3 : STRUCTURES DES DIFFÉRENTS OBJETS DANS LE FIRMWARE CLÉOPÂTRE 2/3	23
FIGURE 3-4 : STRUCTURES DES DIFFÉRENTS OBJETS DANS LE FIRMWARE CLÉOPÂTRE 3/3	23
FIGURE 3-5: MACHINE D'ÉTAT DÉCRIVANT LE COMPORTEMENT GÉNÉRAL	24
FIGURE 3-6 : STRUCTOGRAMME DE L'INTERRUPTION ADC.....	25
FIGURE 3-7: STRUCTOGRAMME DE LA FONCTION DE RÉGULATION.....	26
FIGURE 3-8: STRUCTURES DES DIFFÉRENTS OBJETS DANS LE FIRMWARE ALEXANDRE.....	29
FIGURE 3-9 : STRUCTOGRAMME DE L'INTERRUPTION DE CONTRÔLE	30
FIGURE 3-10: MACHINE D'ÉTAT DE LA CARTE ALEXANDRE	31



LISTE DES TABLEAUX

TABLEAU 3-1 : STRUCTURE S_ARMSCONTROL.....	16
TABLEAU 3-2 : STRUCTURE S_ARMCTRL.....	16
TABLEAU 3-3 : STRUCTURE S_MESSAGE_TOSLAVE	16
TABLEAU 3-4 : STRUCTURE S_MESSAGE_FROMSLAVE	16
TABLEAU 3-5 : STRUCTURE S_CONTROL	17
TABLEAU 3-6 : STRUCTURE S_MEASURES	17
TABLEAU 3-7 : STRUCTURE S_ARMS.....	17
TABLEAU 3-8 : STRUCTURE S_GRID.....	18
TABLEAU 3-9 : STRUCTURE S_GRIDVALUE	18
TABLEAU 3-10 : STRUCTURE S_DCPN.....	18
TABLEAU 3-11 : STRUCTURE S_PLL.....	18
TABLEAU 3-12 : STRUCTURE S_PLLVALUES	18
TABLEAU 3-13 : STRUCTURE S_PLLREG	19
TABLEAU 3-14 : STRUCTURE S_COMMAND	19
TABLEAU 3-15 : STRUCTURE S_ENERGYCTRL	19
TABLEAU 3-16 : STRUCTURE S_ZSVCCTRL.....	19
TABLEAU 3-17 : STRUCTURE S_CONTACTORSTATUS.....	20
TABLEAU 3-18 : STRUCTURE S_BATTERY	20
TABLEAU 3-19 : STRUCTURE S_USERCTRL	20
TABLEAU 3-20 : STRUCTURE S_PWRCTRL.....	20
TABLEAU 3-21 : STRUCTURE S_FLAGERROR DU SSBC.....	21
TABLEAU 3-22 : STRUCTURE S_FLAGERROR DU DSCC.....	21
TABLEAU 3-23 : STRUCTURE S_REGULATOR.....	22
TABLEAU 3-24 : STRUCTURE S_REGULATOR3.....	22
TABLEAU 3-25 : STRUCTURE S_ENERGYREG3.....	22
TABLEAU 3-26 : STRUCTURE S_SLAVE	27
TABLEAU 3-27 : STRUCTURE S_MESSAGE_TOMASTER.....	27
TABLEAU 3-28 : STRUCTURE S_MESSAGE_FROMMASTER	27
TABLEAU 3-29 : STRUCTURE S_SMVALUE	28
TABLEAU 3-30 : STRUCTURE S_FLAGERROR (ALEXANDRE).....	28
TABLEAU 3-31 : STRUCTURE S_CONTROL	29
TABLEAU 3-32 : EXPLICATIONS DES ERREURS POSSIBLES DANS LE FIRMWARE ALEXANDRE	31



ABREVIATIONS

ADC :	Analog to Digital Converter
CMPA/B :	Compare register A/B (microcontrôleur)
DSCC :	Double Star Chopper Cell
I2C :	Inter-Integrated Circuit (communication bus)
SOCA :	Start Of Conversion A (microcontrôleur)
SSBC :	Single Star Bridge Cell
STATCOM :	STAtic synchronous COMpensator
PSPWM :	Phase-Shift Pulse Width Modulation
PWM :	Pulse Width Modulation
TBCTR :	Time Base CounTeR register (microcontrôleur)
TBPRD :	Time Base PeRioD register (microcontrôleur)
XINT :	eXternal INTerrupt (microcontrôleur)



1 INTRODUCTION

Deux topologies de STATCOM ont été sélectionnées pour les tests pratiques du projet. Ces topologies sont le SSBC et le DSCC. Ces structures sont capables de fonctionner sur un réseau moyenne et haute tension et de continuer leur opération lorsque le réseau est déséquilibré ou lorsqu'un défaut apparaît dans le réseau.

Ce rapport présente en détail la conception hardware et firmware des 2 prototypes. Il ne contient pas de mesures de fonctionnement car elles seront présentées dans le prochain livrable. A noter que les 2 prototypes sont conçus pour fonctionner sur un réseau basse tension afin de pouvoir effectuer différents tests en laboratoire.

2 CONCEPTION HARDWARE

Ce chapitre présente la conception hardware des 2 prototypes. Pour chaque topologie, la stratégie de pilotage des composants de puissance est présentée. Puis le schéma électrique simplifié est rappelé. Finalement, le détail du câblage ainsi que la disposition des différentes cartes électroniques sont présentés.

Les 2 prototypes sont chacun intégrés dans une armoire comme le montre la Figure 2-1 :



Figure 2-1 : Prototypes des convertisseurs basse tension



2.1 STRATÉGIE DE PILOTAGE

Le pilotage des composants de puissance se fait en passant par plusieurs cartes électroniques comme cela est visible sur la Figure 2-2.

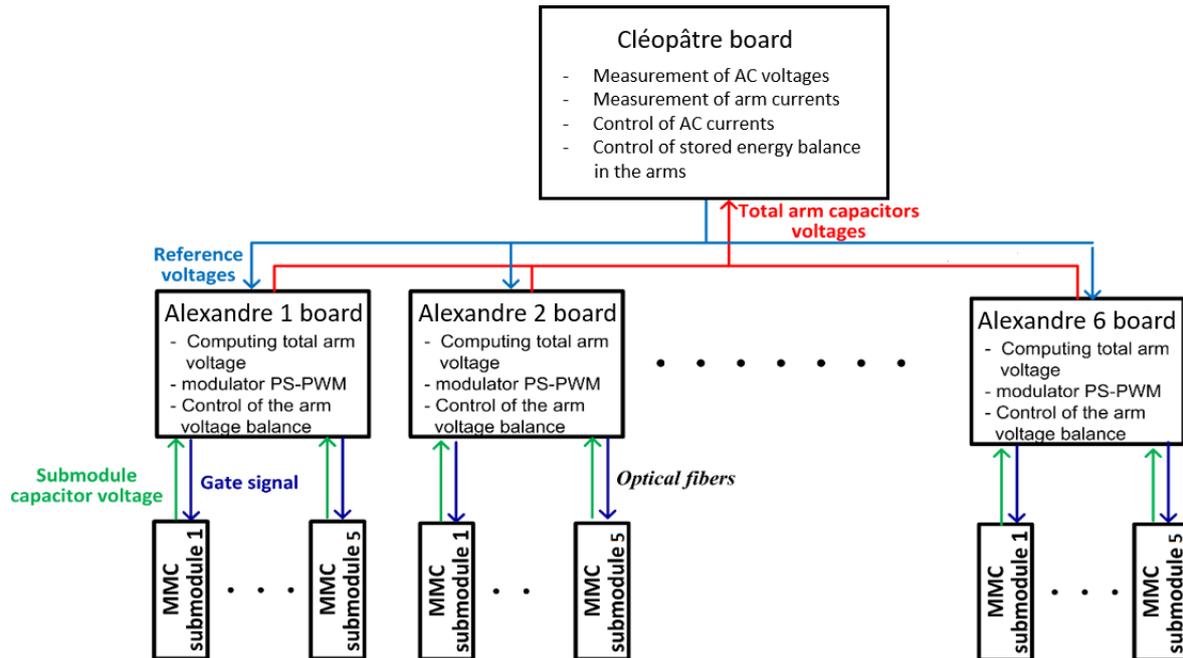


Figure 2-2 : stratégie hardware de pilotage des convertisseurs

La carte Cléopâtre est le maître qui mesure les tensions réseau et les courants de branche. Cette carte effectue le contrôle haut niveau des convertisseurs et communique avec les différentes cartes Alexandre (3 pour le SSBC et 6 pour DSCC). Chaque carte Alexandre effectue le contrôle bas niveau du convertisseur en pilotant les 5 sous-modules de la branche correspondante. Les sous-modules sont composés soit d'une cellule « bridge cell » (pour le SSBC) soit d'une « chopper cell » (pour le DSCC). Ils sont pilotés avec des fibres optiques et remontent la mesure de tension du sous-module avec la même technologie.

2.2 SINGLE STAR BRIDGE CELL (SSBC)

Pour rappel, la Figure 2-3 montre le schéma électrique simplifié de la topologie SSBC. Le développement mathématique et les simulations de son fonctionnement sont disponibles dans [1].

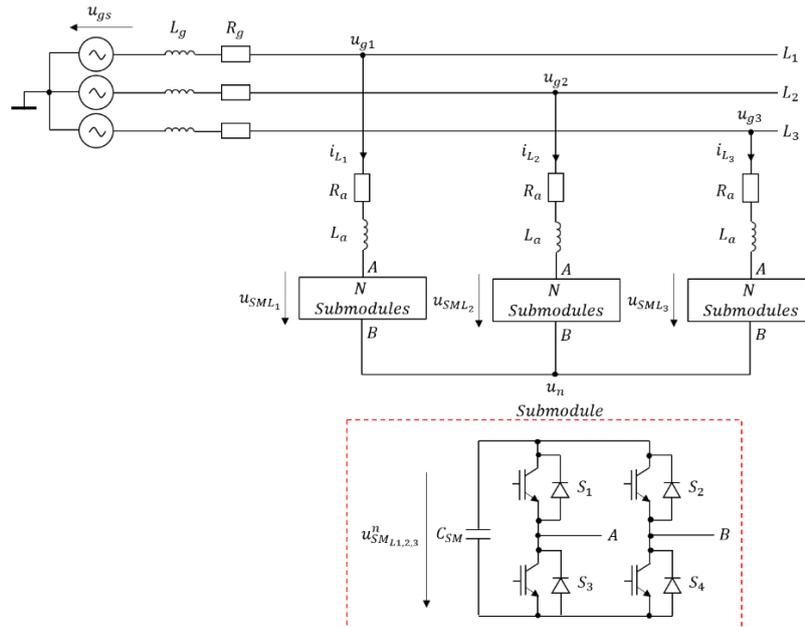


Figure 2-3 : schéma électrique simplifié du SSBC

2.2.1 Armoire du SSBC

La Figure 2-4 montre l'intégration mécanique des différentes cartes électroniques dans l'armoire SSBC.

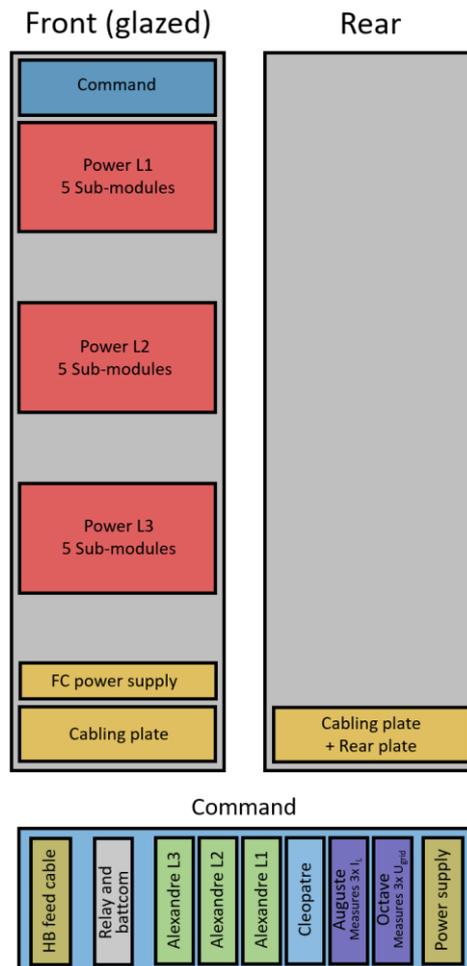


Figure 2-4 : positionnement mécanique des différentes cartes dans l'armoire SSBC



Le détail du câblage entre les différents éléments de l'armoire est visible sur la Figure 2-5.

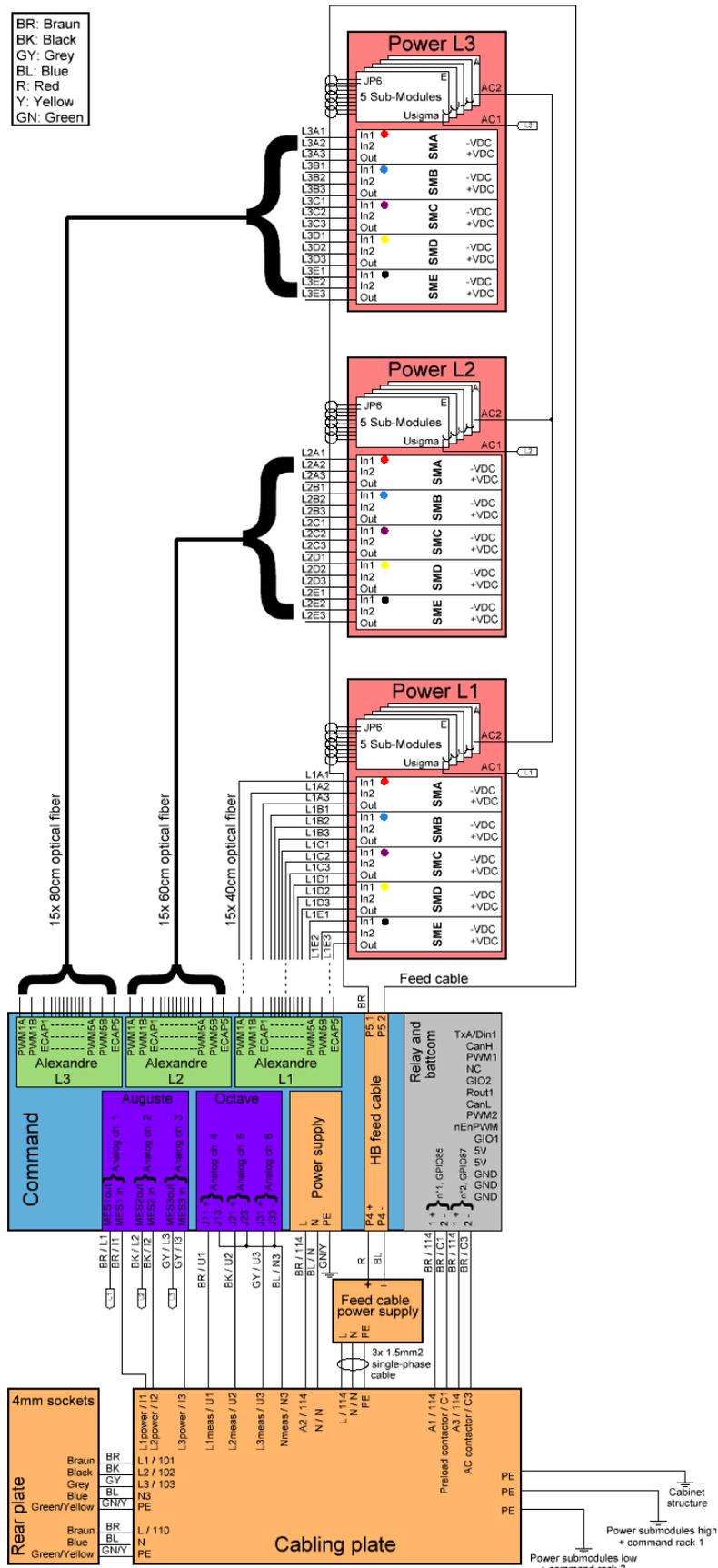


Figure 2-5 : câblage de l'armoire SSBC



Finalement, le détail du câblage de la plaque de fond est montré sur la Figure 2-6.

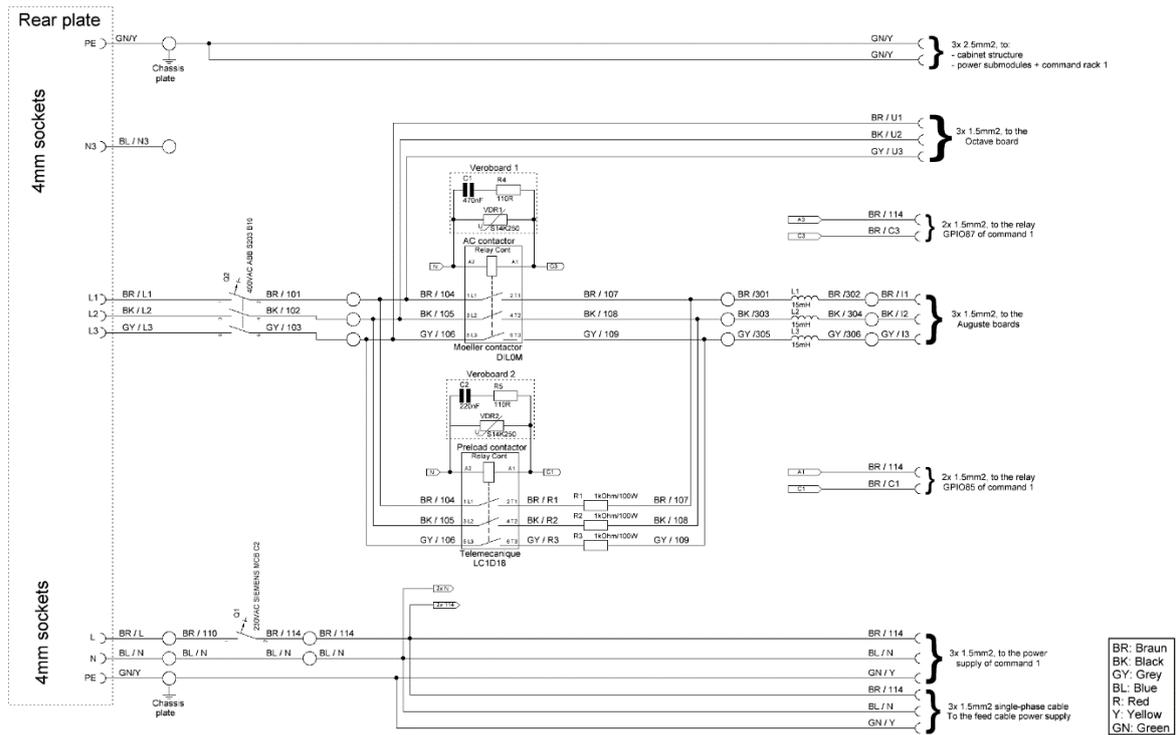


Figure 2-6 câblage de la plaque de fond SSBC

2.3 DOUBLE STAR CHOPPER-CELLS (DSCC)

Pour rappel, la Figure 2-7 montre le schéma électrique simplifié de la topologie DSCC. Le développement mathématique et les simulations de son fonctionnement sont disponibles dans [1].

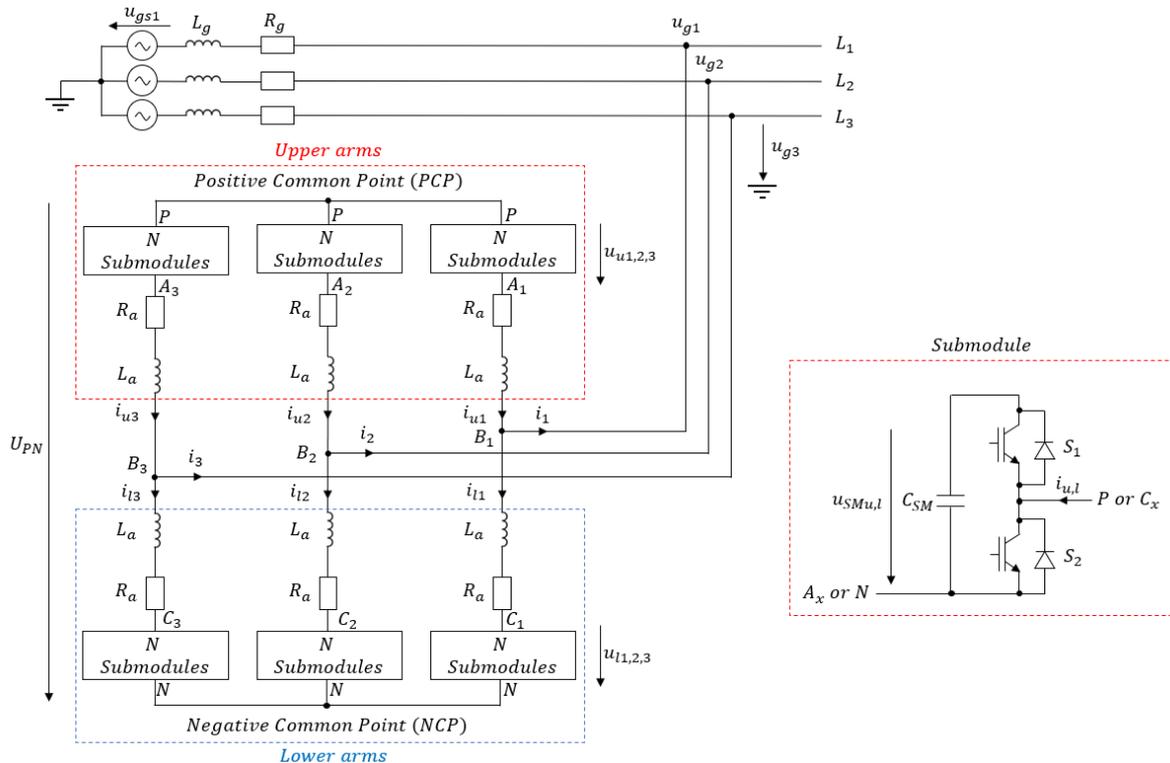


Figure 2-7 : schéma électrique simplifié du DSCC



2.3.1 Armoire du DSCC

La Figure 2-8 montre l'intégration mécanique des différentes cartes électroniques dans l'armoire DSCC.

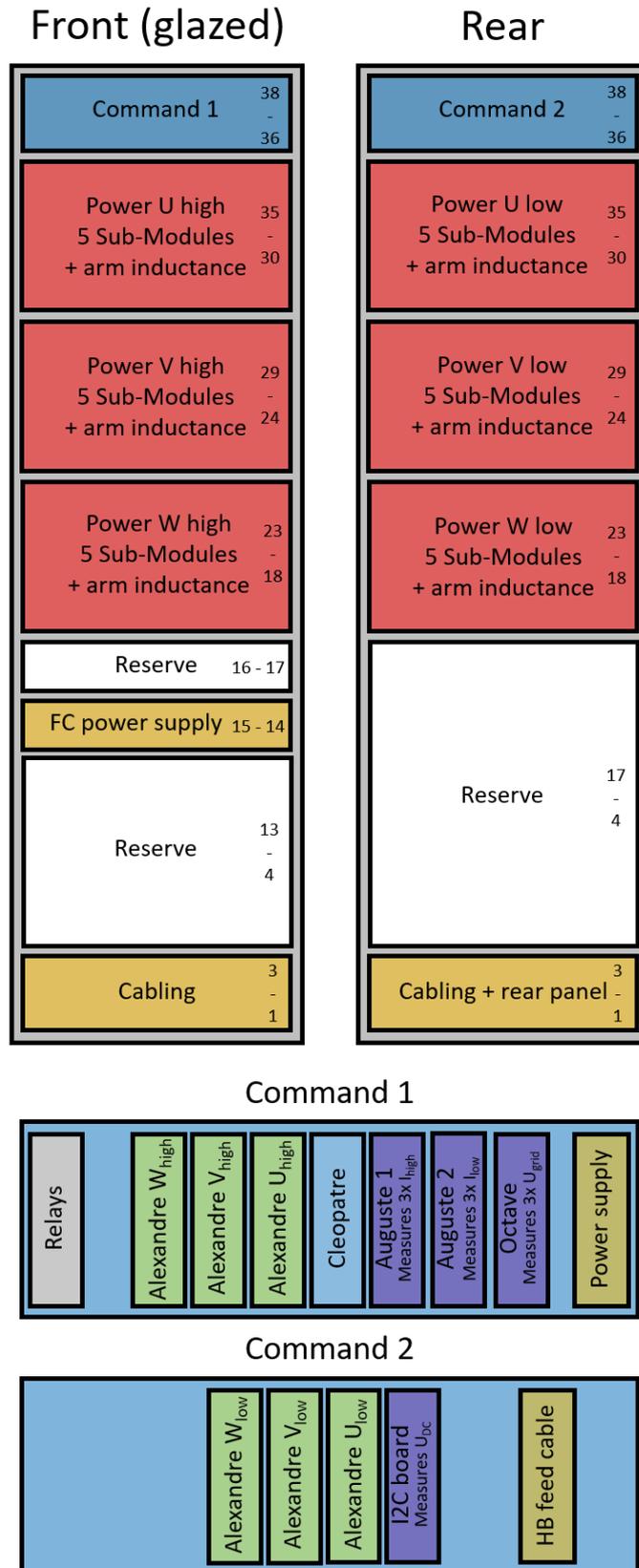


Figure 2-8 : positionnement mécanique des différentes cartes dans l'armoire DSCC



Le détail du câblage entre les différents éléments de l'armoire est visible sur la Figure 2-9.

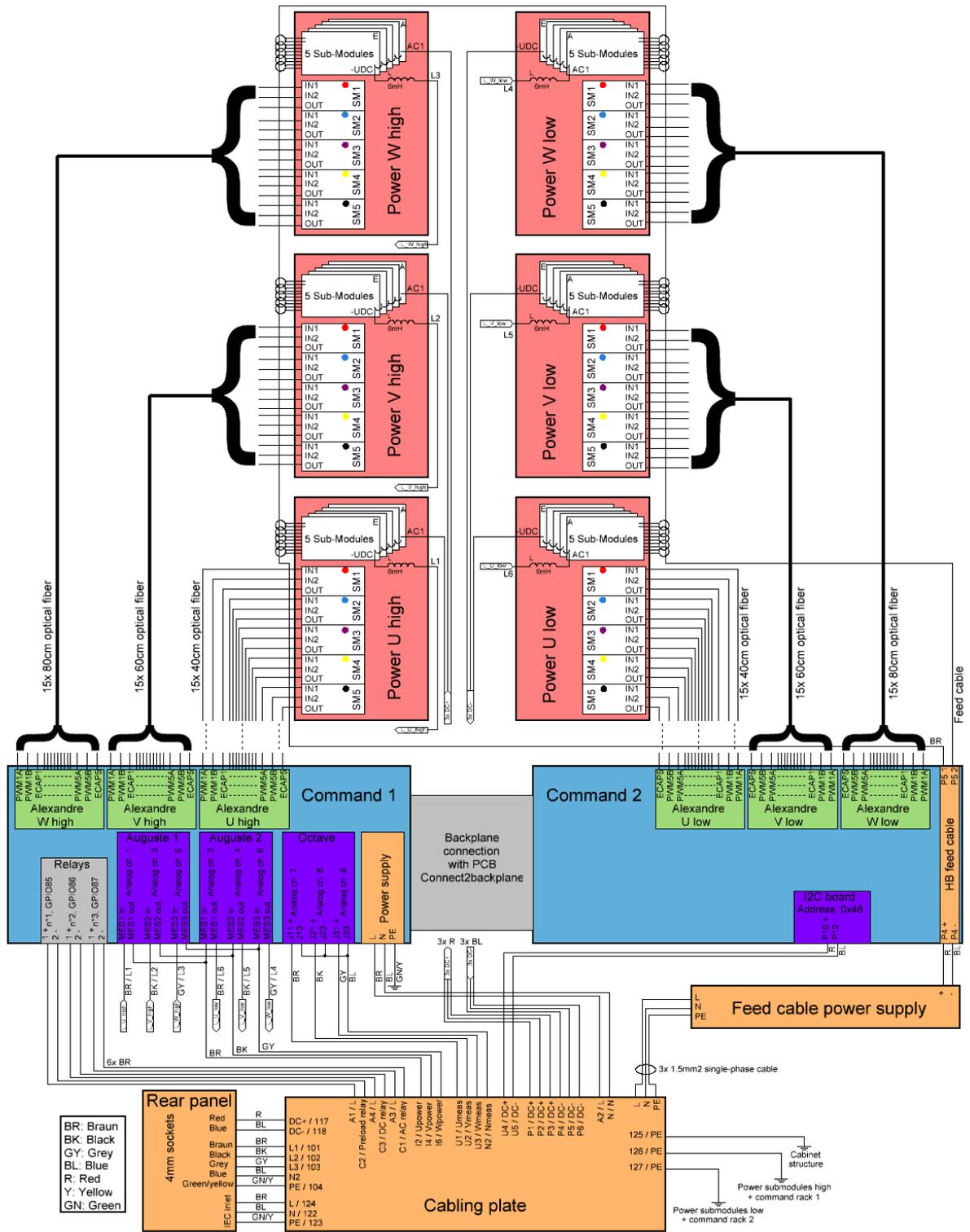


Figure 2-9 : câblage de l'armoire DSCC



Finalement, le détail du câblage de la plaque de fond est montré sur la Figure 2-10.

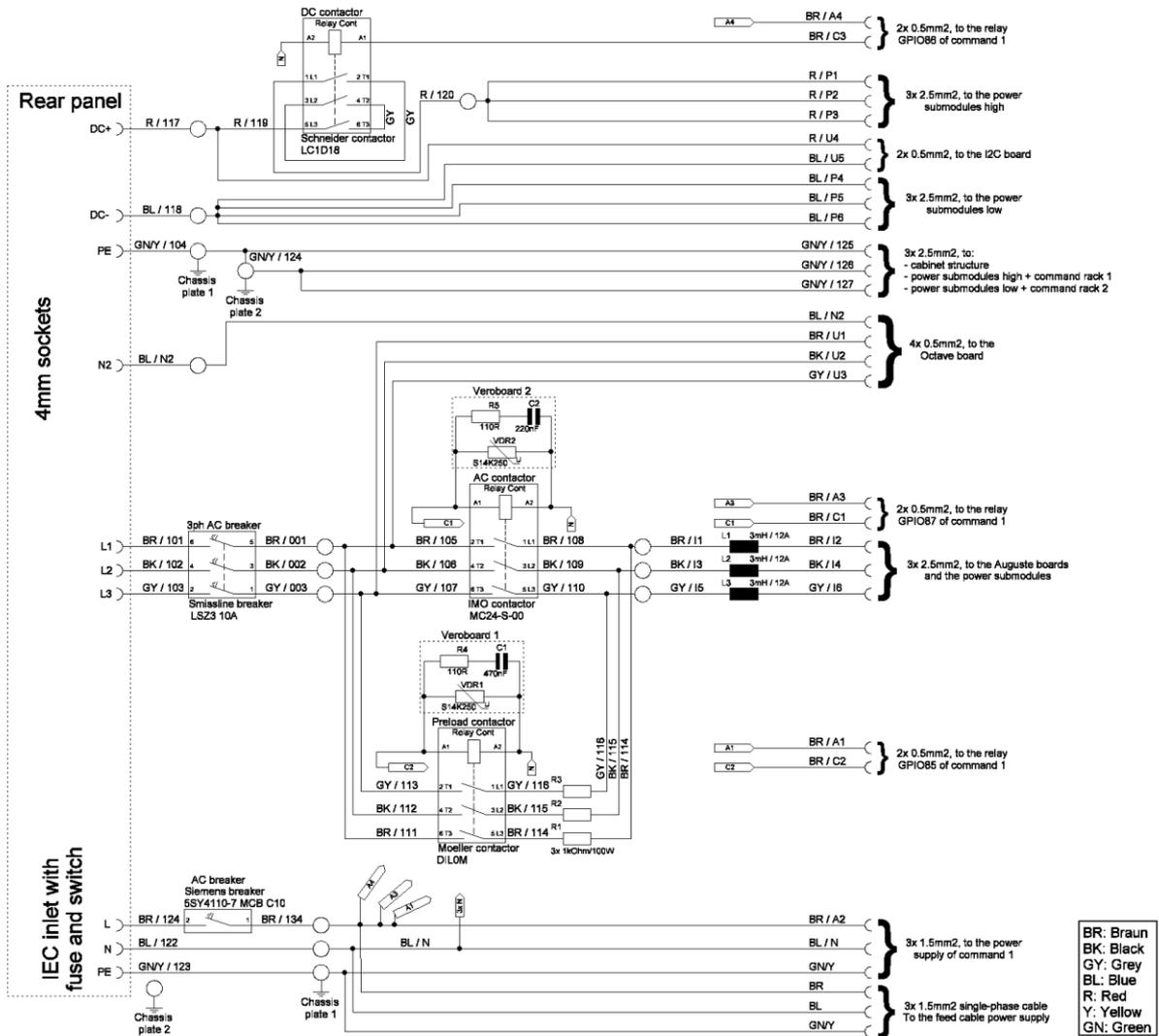


Figure 2-10 câblage de la plaque de fond DSCC

3 CONCEPTION FIRMWARE

La section 2.1 permet de comprendre au mieux l'interconnexion entre les différentes cartes électroniques et facilite la compréhension des firmwares sur les cartes Cléopâtre et Alexandre.

La stratégie de fonctionnement avec la modulation phase-shift pulse width modulation (PSPWM) est représentée sur la Figure 3-1. Le déroulement général du firmware de la carte Cléopâtre se trouve en bleu sur la gauche. Celui du firmware des cartes Alexandre se trouve en vert sur la droite. La partie inférieure de l'image montre le timer de la carte Cléopâtre en bleu qui permet de générer un signal de synchronisation pour toutes les cartes Alexandre. Les timers des cartes Alexandre sont en vert et permettent la génération des signaux PWMs pour le contrôle des sous-modules de puissance.

Pour des explications supplémentaires de la stratégie de synchronisation et de fonctionnement général, les sections 3.1 et 3.1.2 présentent en détail ces fonctionnements.



Firmware strategy for the PSPWM modulation

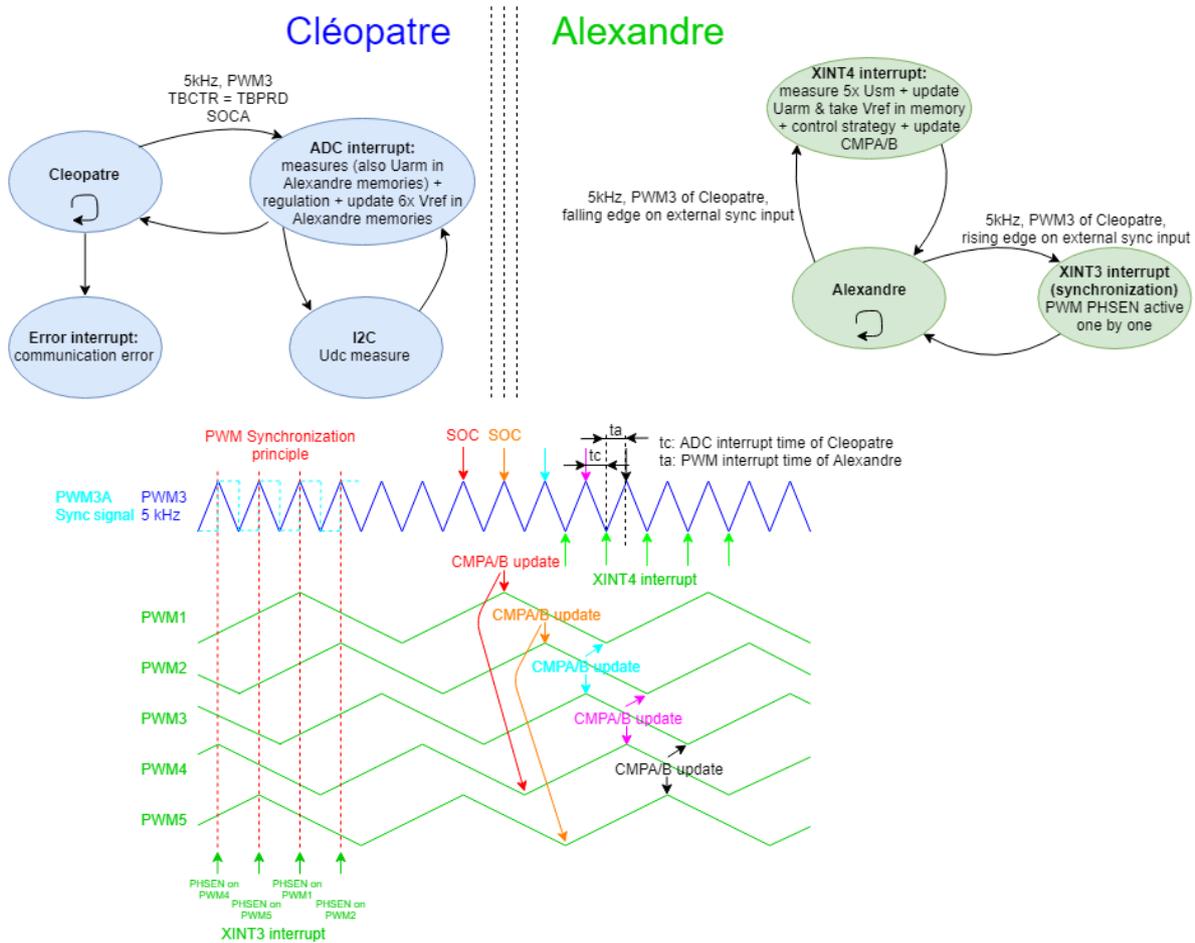


Figure 3-1 : stratégie de fonctionnement avec la modulation PSPWM

3.1 CARTE CLÉOPÂTRE

Selon la Figure 3-1, la carte Cléopâtre a une interruption ADC qui est déclenchée par le timer PWM3. Ce timer permet d'effectuer les mesures à une fréquence de 5 [kHz] ainsi que de générer un signal de synchronisation à la même fréquence pour le pilotage des cartes Alexandre. Plus de détails sur cette interruption ADC sont donnés dans la section 3.1.3. La section 3.1.2 présente le fonctionnement général du firmware avec la machine d'état intégrée dans la carte Cléopâtre.

A noter que les firmwares des 2 prototypes (SSBC et DSCC) sont très proches. Ainsi, il n'y a qu'une seule description pour les 2 prototypes. Dans la suite de ce chapitre, les légères différences entre les 2 topologies sont indiquées par un code de couleur. En bleu pour ce qui ne concerne que le SSBC et en rouge pour le DSCC.

3.1.1 Structures

Avant d'expliquer le fonctionnement du firmware, il est important de présenter les structures qui composent le code. En effet, la compréhension de ces structures permet une lecture simplifiée du code. Bien que le langage de programmation C n'est pas un langage orienté objet, on parlera d'objet pour les structures de haut niveau.



3.1.1.1 S_ArmsControl

L'objet « armsControl » est déclaré de ce type-là, il permet de contrôler les différentes cartes Alexandre du système. La communication est comprise dans cette objet. Il est composé de :

Nom	Type	Description
arm[NB_ARM]	S_ArmCtrl	Contrôle des cartes Alexandre de manière indépendante
mode	E_MODE	Mode de fonctionnement de l'armoire
ma	float	Profondeur de modulation en mode OPEN_LOOP

Tableau 3-1 : structure S_ArmsControl

Le type « E_MODE » est un type enum qui peut prendre les valeurs suivantes :

- DISABLE : désactivation des cartes Alexandre.
- OPEN_LOOP : Activation des cartes Alexandre en boucle ouverte. La profondeur de modulation ma est prise en compte pour générer les consignes.
- REGULATION : Activation des cartes Alexandre avec la régulation.

S_ArmCtrl

Cette structure est comprise dans les objets de type « S_ArmsControl ». La structure est composée de :

Nom	Type	Description
messTo	S_MessageToSlave	Message transmis à l'esclave (Alexandre)
messFrom	S_MessageFromSlave	Message provenant de l'esclave
address	Uint16	Adresse de l'esclave
control	S_Control	Contrôle de l'esclave

Tableau 3-2 : structure S_ArmCtrl

S_MessageToSlave

Cette structure est comprise dans les structures de type « S_ArmCtrl ». La structure est composée de :

Nom	Type	Description
rawMessage	Uint16	Message brut à transmettre à l'esclave
nRef	float	Référence de l'indice d'insertion pour l'esclave
armCurrentDir	int	Direction du courant dans le bras que contrôle l'esclave
ack_disUndervoltage	Uint16 :1	Quittancement/désactivation de l'erreur de sous-tension pour les cartes Alexandre
enable	Uint16 :1	Autorisation de fonctionnement de l'esclave
parityBit	Uint16 :1	Bit de parité pour la détection d'erreur de communication

Tableau 3-3 : structure S_MessageToSlave

S_MessageFromSlave

Cette structure est comprise dans les structures de type « S_ArmCtrl ». La structure est composée de :

Nom	Type	Description
rawMessage	Uint16	Message brut provenant de l'esclave
uSigma	float	Tension totale du bras (somme des différents sous-modules) contrôlé par l'esclave
capUnbalanced	Uint16 :1	Actif si la tension d'un sous-module contrôlé par l'esclave est trop éloignée de la moyenne
error	Uint16 :1	Actif si une erreur de l'esclave a lieu
parityBit	Uint16 :1	Bit de parité pour la détection d'erreur de communication

Tableau 3-4 : structure S_MessageFromSlave



S_Control

Cette structure est comprise dans les structures de type « S_ArmCtrl ». La structure est composée de :

Nom	Type	Description
enable	Uint16	Autorisation du fonctionnement de l'esclave
ack_disUndervoltage	Uint16	Quittancement/désactivation de l'erreur de sous-tension pour les cartes Alexandre
nRef	float	Indice d'insertion de référence à appliquer par l'esclave

Tableau 3-5 : structure S_Control

3.1.1.2 S_Measures

L'objet « mesures » est déclaré de ce type-là, il permet d'avoir toutes les mesures accessibles facilement au même endroit. Il est composé de :

Nom	Type	Description
arms	S_Arms	Mesures des différents bras du système
grid	S_Grid	Mesures du réseau triphasé
dcPN	S_DcPN	Mesures du bus DC (seulement pour le DSCC)

Tableau 3-6 : structure S_Measures

S_Arms

Cette structure est comprise dans les objets de type « S_Measures ». La structure est composée de :

Nom	Type	Description
offsetCurrent[NB_ARM]	float	Offset des mesures de courant
current[NB_ARM]	float	Mesures des courants dans chaque bras
uSigma[NB_ARM]	float	Mesures des tensions totales de bras (vient de la communication avec l'esclave)
uSigmaAverage[2]	float	Moyenne des tensions
voltageTot[NB_PH]	float	Tension totale de branche (up + low) (seulement pour le DSCC)
energyTot[NB_PH][2]	float	Energie totale de branche (seulement pour le DSCC)
energyTotFiltered[NB_PH][2]	float	Energie totale de branche filtrée (seulement pour le DSCC)
voltageDiff[NB_PH]	float	Différence de tension de branche (up - low) (seulement pour le DSCC)
energyDiff[NB_PH][2]	float	Différence d'énergie de branche (seulement pour le DSCC)
energyDiffFiltered[NB_PH][2]	float	Différence d'énergie de branche filtrée (seulement pour le DSCC)
iCirc[NB_PH]	float	Mesures des courants circulants (seulement pour le DSCC)
energy[NB_ARM][2]	float	Energie de branche (seulement pour le SSBC)
energyFiltered[NB_ARM][2]	float	Energie de branche filtrée (seulement pour le SSBC)
energyAverage[2]	float	Energie moyenne de branche
energyAverageFiltered[2]	float	Energie moyenne de branche filtrée

Tableau 3-7 : structure S_Arms



S_Grid

Cette structure est comprise dans les structures de type « S_Measures ». La structure est composée de :

Nom	Type	Description
current	S_GridValue	Mesures des courants du réseau triphasé
voltage	S_GridValue	Mesures des tensions du réseau triphasé

Tableau 3-8 : structure S_Grid

S_Gridvalue

Cette structure est comprise dans les structures de type « S_Grid ». La structure est composée de :

Nom	Type	Description
uvw[NB_PH]	float	Valeurs triphasées
posSeq[NB_PH]	float	Valeurs de la séquence positive
negSeq[NB_PH]	float	Valeurs de la séquence négative
zerSeq	float	Valeur de la séquence homopolaire
dqPos[2]	float	Valeurs de l'axe d et q de la séquence positive/négative ([0] -> d, [1] -> q)
dqNeg[2]	float	
dqPosFiltered[2]	float	Valeurs de l'axe d et q filtrée de la séquence positive

Tableau 3-9 : structure S_GridValue

S_DcPN

Cette structure est comprise dans les objets de type « S_Measures ». Elle n'est utilisée que pour le DSCC. La structure est composée de :

Nom	Type	Description
voltage	float	Tension entre le pôle positif et négatif
voltageFiltered	float	Tension filtrée
invVoltageFiltered	float	Inverse de la tension filtrée

Tableau 3-10 : structure S_DcPN

3.1.1.3 S_PII

L'objet « pll » est déclaré de ce type-là, il permet d'avoir tout ce qui concerne la synchronisation au réseau triphasé au même endroit. Il est composé de :

Nom	Type	Description
value	S_PIIValues	Mesures de la PLL
regul	S_PIIReg	Valeurs de régulation de la PLL

Tableau 3-11 : structure S_PII

S_PIIValues

Cette structure est comprise dans les objets de type « S_PII ». La structure est composée de :

Nom	Type	Description
theta	float	Angle actuel du réseau
ud[2]	float	Tension de l'axe d du réseau
uq[2]	float	Tension de l'axe q du réseau
uHomo	float	Tension homopolaire du réseau
w[2]	float	Pulsation actuelle du réseau

Tableau 3-12 : structure S_PIIValues



S_PllReg

Cette structure est comprise dans les objets de type « S_Pll ». La structure est composée de :

Nom	Type	Description
Alpha	float	Paramètre du régulateur de la PLL
Ki	float	Gain intégral standard
Kp	float	Gain proportionnel standard
B0_s	float	Gain proportionnel sous forme numérique
B1_s	float	Gain intégral sous forme numérique

Tableau 3-13 : structure S_PllReg

3.1.1.4 S_Command

L'objet « command » est déclaré de ce type-là, il permet d'avoir tout ce qui concerne les consignes des différentes grandeurs de régulation. Il est composé de :

Nom	Type	Description
idqPos[2]	float	Commande de courant d'axe d et q de la séquence positive/négative ([0] -> d, [1] -> q)
idqNeg[2]	float	
udqPos[2]	float	Commande de tension d'axe d et q de la séquence positive/négative ([0] -> d, [1] -> q)
udqNeg[2]	float	
uabc[NB_PH]	float	Commandes de tensions triphasées
zsvc	S_ZsvcCtrl	Contrôle de la stratégie ZSVC (seulement pour le SSBC)
nRef[NB_ARM]	float	Commandes des indices d'insertion pour chaque esclave
internalVoltage	S_EnergyCtrl	Commande de tension interne pour chaque bras
activePower	float	Commande de puissance active
reactivePower	float	Commande de puissance réactive
iCirc[NB_PH]	float	Commandes de courants circulants finales pour régler l'énergie dans le système complet (seulement pour le DSCC)
uCirc[NB_PH]	float	Commandes de tensions pour les courants circulants (seulement pour le DSCC)

Tableau 3-14 : structure S_Command

S_EnergyCtrl

Cette structure est comprise dans les objets de type « S_Command ». La structure est composée de :

Nom	Type	Description
strategy	E_ENERGY_CTRL	Stratégie de contrôle pour le réglage de l'énergie
ref	float	Référence de tension pour le réglage de l'énergie

Tableau 3-15 : structure S_EnergyCtrl

Le type « E_ENERGY_CTRL » est un type enum qui peut prendre les valeurs suivantes :

- CLOSED_LOOP : stratégie en fermant la boucle de réglage de l'énergie.
- AVERAGE: stratégie moyennée qui permet d'effectuer la précharge.

S_ZsvcCtrl

Cette structure est comprise dans les objets de type « S_Command ». La structure est composée de :

Nom	Type	Description
enable	Uint16	Autorisation pour le contrôle de la ZSVC
u0	float	Commande de tension homopolaire provenant de la stratégie ZSVC (seulement pour le SSBC)

Tableau 3-16 : structure S_ZsvcCtrl



3.1.1.5 S_ContactorStatus

L'objet « contactorStatus » est déclaré de ce type-là, il permet de voir l'état des différents contacteurs du système. Il est composé de :

Nom	Type	Description
dc	Uint16	Etat du contacteur du bus DC (seulement pour le DSCC)
ac	Uint16	Etat du contacteur du réseau AC
preload	Uint16	Etat du contacteur de la précharge

Tableau 3-17 : structure S_ContactorStatus

3.1.1.6 S_Battery

L'objet « battery » est déclaré de ce type-là, il permet de voir l'état de la batterie. Elle n'est utilisée que pour le SSBC. Il est composé de :

Nom	Type	Description
isUsed	Uint16	Indique si la batterie est utilisée
available	Uint16	Indique si la batterie est disponible
soc	float	Etat de charge de la batterie

Tableau 3-18 : structure S_Battery

3.1.1.7 S_UserCtrl

L'objet « userCtrl » est déclaré de ce type-là, il permet d'effectuer le contrôle de tout le système depuis un seul endroit. Il est composé de :

Nom	Type	Description
ackSystem	Uint16	Permet de quittancer le système
frcSwError	Uint16	Permet de forcer une erreur software
turnOffSystem	Uint16	Permet d'éteindre le système
internalVoltage	S_EnergyCtrl	Permet de contrôler la tension (niveau d'énergie) interne et la stratégie pour l'effectuer
power	S_PwrCtrl	Permet de contrôler la puissance réactive
zsvcEnable	Uint16	Autorisation pour le contrôle avec la ZSVC.

Tableau 3-19 : structure S_UserCtrl

S_PwrCtrl

Cette structure est comprise dans les objets de type « S_UserCtrl ». La structure est composée de :

Nom	Type	Description
enable	Uint16	Autorisation pour le réglage de la puissance réactive
reactiveRef	float	Référence de puissance réactive

Tableau 3-20 : structure S_PwrCtrl



3.1.1.8 S_FlagError

L'objet « flagsError » est déclaré de ce type-là, il permet de regrouper toutes les erreurs possibles dans une seule structure. Dans le cas du SSBC, l'objet est composé de :

Nom	Type	Description
Software	Uint16 :1	Actif si une erreur software a eu lieu
acOverCurrent	Uint16 :1	Actif si un surcourant a eu lieu dans le réseau AC
acVoltage	Uint16 :1	Actif si la tension du réseau AC est hors spécification
Preload	Uint16 :1	Actif si la tension des condensateurs évolue hors de la plage attendue durant la précharge
armVoltageImbalance	Uint16 :1	Actif si les tensions de bras sont trop déséquilibrées
communication	E_COM_ERROR	Actif si une erreur de communication a eu lieu
armU	Uint16 :1	Actif si l'esclave de la branche U est en erreur
armV	Uint16 :1	Actif si l'esclave de la branche V est en erreur
armW	Uint16 :1	Actif si l'esclave de la branche W est en erreur

Tableau 3-21 : structure S_FlagError du SSBC

Dans le cas du DSCC, l'objet est composé de :

Nom	Type	Description
software	Uint16 :1	Actif si une erreur software a eu lieu
armOverCurrent	Uint16 :1	Actif si un surcourant a eu lieu dans un bras
dcOverCurrent	Uint16 :1	Actif si un surcourant a eu lieu dans le bus DC
dcVoltage	Uint16 :1	Actif si la tension du bus DC est hors spécification
acOverCurrent	Uint16 :1	Actif si un surcourant a eu lieu dans le réseau AC
acVoltage	Uint16 :1	Actif si la tension du réseau AC est hors spécification
preload	Uint16 :1	Actif si la tension des condensateurs évolue hors de la plage attendue durant la précharge
armVoltageImbalance	Uint16 :1	Actif si les tensions de bras sont trop déséquilibrées
communication	E_COM_ERROR	Actif si une erreur de communication avec un esclave a eu lieu
armUup	Uint16 :1	Actif si l'esclave de la bras Uup est en erreur
armUdown	Uint16 :1	Actif si l'esclave de la bras Udown est en erreur
armVup	Uint16 :1	Actif si l'esclave de la bras Vup est en erreur
armVdown	Uint16 :1	Actif si l'esclave de la bras Vdown est en erreur
armWup	Uint16 :1	Actif si l'esclave de la bras Wup est en erreur
armWdown	Uint16 :1	Actif si l'esclave de la bras Wdown est en erreur

Tableau 3-22 : structure S_FlagError du DSCC

Le type « E_COM_ERROR » est un type enum qui peut prendre les valeurs suivantes :

- NO_ERROR_COM : il n'y a pas d'erreur de communication.
- AT_READ : une erreur de communication a été détectée à la lecture.
- AT_WRITE : une erreur de communication a été détectée à l'écriture.

Ces structures d'erreurs sont construites de manière à ce qu'elles valent zéro si aucune erreur n'est détectée.

3.1.1.9 Régulateurs

Les régulateurs utilisés sont différents entre le système SSBC et le DSCC. Dans les 2 cas il y a des régulateurs de courant « idPosReg », « iqPosReg », « idNegReg » et « iqNegReg » ainsi qu'un régulateur d'énergie « energyReg » de type « S_Regulator ». Ensuite, pour le SSBC il y a :

- Deux régulateurs de cluster « L1clusterReg » et « L2clusterReg » de type « S_Regulator ».

Pour le DSCC, il y a :

- Un régulateur d'énergie « energyReg3 » de type « S_EnergyReg3 ».
- Un triple régulateur de courant « iCircReg » de type « S_Regulator3 ».

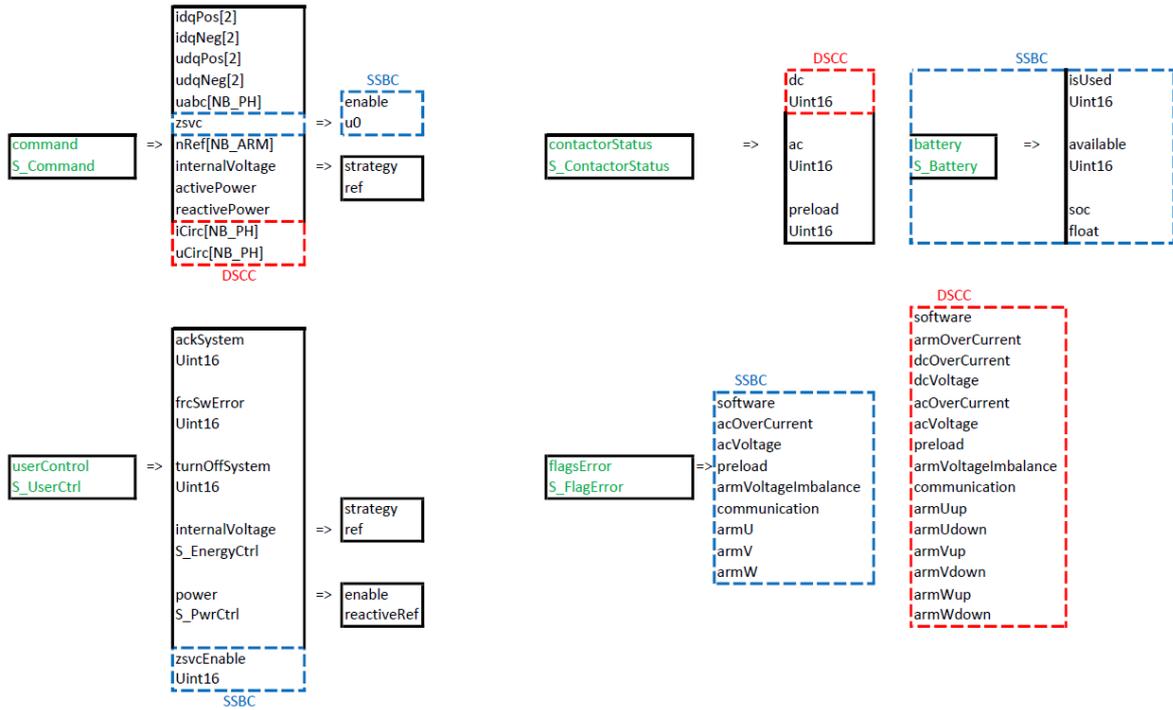


Figure 3-3 : structures des différents objets dans le firmware Cléopâtre 2/3

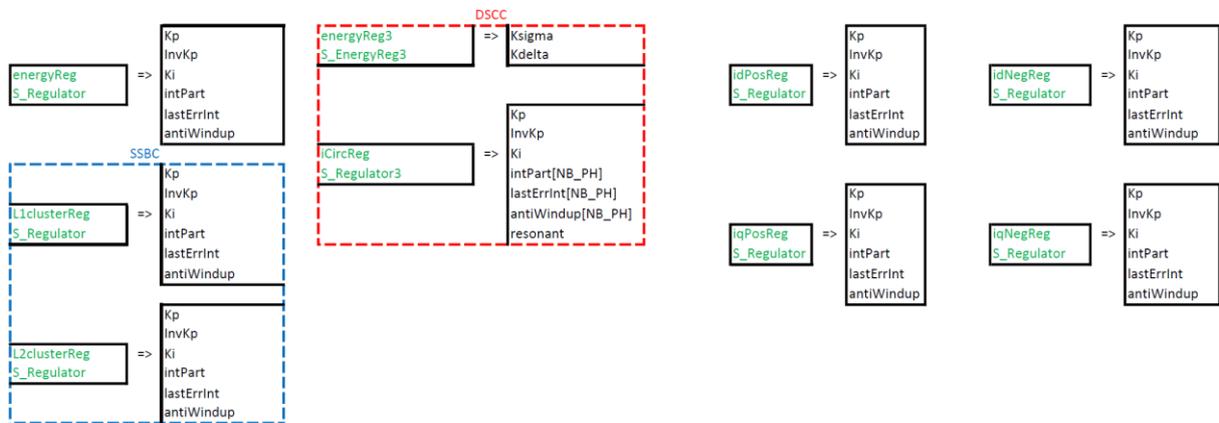


Figure 3-4 : structures des différents objets dans le firmware Cléopâtre 3/3



3.1.2 Machine d'état

La machine d'état visible sur la Figure 3-5 permet de comprendre le comportement général du système. La description de chaque état est :

- **Fault** : c'est l'état au démarrage. Le système est en erreur et désactive tout le contrôle du système. L'état est quitté lorsque l'utilisateur quitte l'erreur au moins 1 seconde après que l'erreur soit apparue.
- **Acknowledge** : le quittancement efface les erreurs de la carte Cléopâtre ainsi que les erreurs des cartes Alexandre. Le passage au prochain état est automatique.
- **Preload** : la précharge s'effectue en 2 étapes. La première se fait au travers de résistances de précharge en mode boucle ouverte et une profondeur de modulation constante. Lors de la deuxième étape, le branchement direct au réseau AC est effectué, le mode de régulation de puissance est activé. Lorsque la tension désirée est atteinte, le contacteur au réseau DC se ferme et la machine d'état passe à l'état suivant après 1 seconde d'attente.
- **Operating** : les commutations sont autorisées et le réglage complet est effectué. L'utilisateur peut choisir de passer à l'état suivant afin d'éteindre le système.
- **Turn off** : les commutations sont toujours actives car elles permettent de décharger les condensateurs plus rapidement.

A noter que dès l'état « Preload », si une erreur intervient, la machine d'état passe à « Fault ». De plus, pour le DSCC, le contacteur DC est toujours enclenché afin que la mesure entre le pôle positif et négatif soit toujours active.

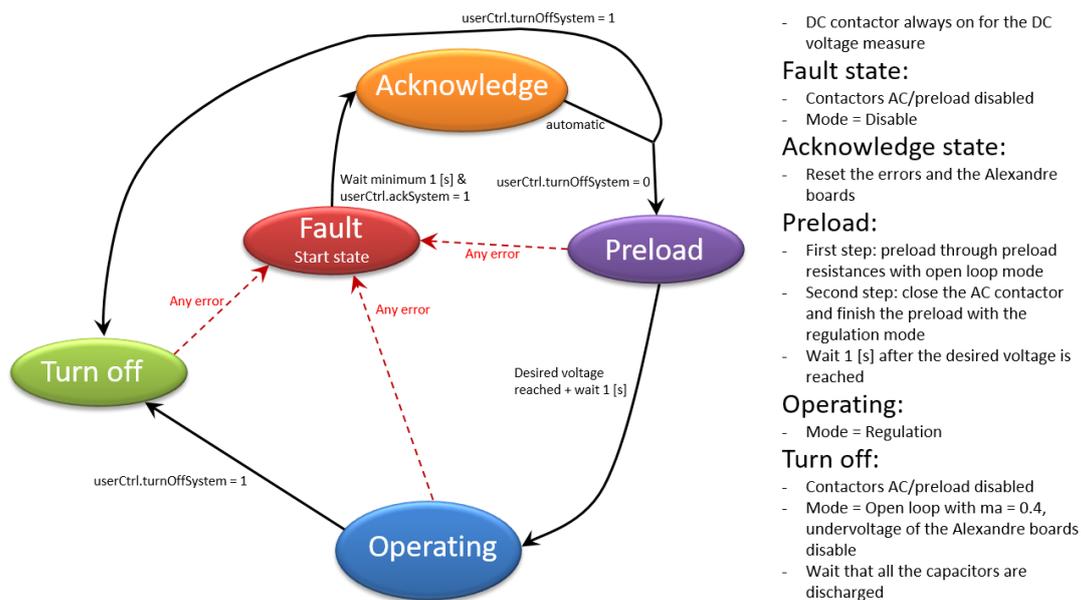


Figure 3-5: Machine d'état décrivant le comportement général.



3.1.3 Interruption ADC

La Figure 3-6 présente le fonctionnement de l'interruption ADC. La présentation générale est assez sommaire, cependant un descriptif est donné pour chaque action du firmware.

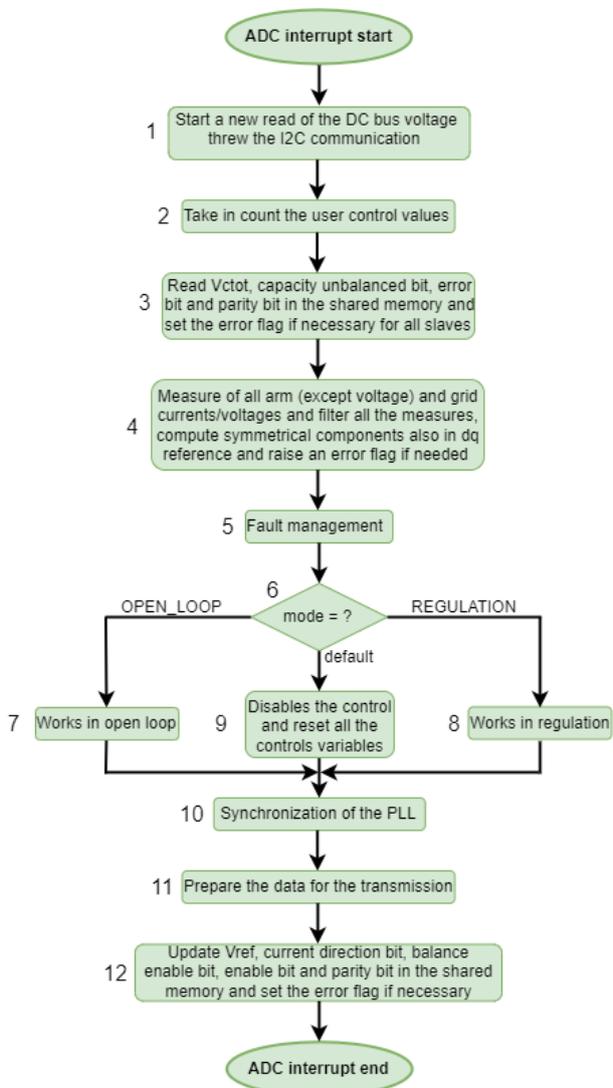


Figure 3-6 : Structogramme de l'interruption ADC.

1 : Démarre une nouvelle communication I2C afin de mesurer la tension entre le pôle positif et négatif. La communication fonctionne en parallèle à l'interruption ADC et ne l'interrompt pas. Le résultat de la mesure est disponible au prochain passage dans cette interruption.

2 : Prend en compte les consignes données par l'utilisateur (puissance et tension interne).

3 : Sample & hold des messages provenant des esclaves. Lit et sauvegarde les informations. Le flag d'erreur communication « AT_READ » est levé si l'un des bits de parité est faux.

4 : Mesure les courants de bras et les courants/tensions du réseau AC. Filtre toutes les mesures, y compris les tensions de bras lues dans les messages des esclaves, calcule les composantes symétriques (aussi dans le référentiel dq). Active le flag d'erreur correspondant pour chaque mesure.

5 : Gestion des erreurs. Si une erreur est active, la machine d'état générale est mise dans l'état d'erreur.

6 : Détection du mode de fonctionnement.

7 : Fonctionne en boucle ouverte selon la profondeur de modulation désirée. Les variables de contrôle des contrôleurs d'énergie et des courants réseau sont réinitialisées.

8 : Fonctionnement normal, la régulation du système est activée. Cette étape est expliquée plus en détail dans la section 3.1.3.1.

9 : Désactive le contrôle et réinitialise les variables de contrôle.

10 : Synchronisation de la PLL afin de déterminer l'angle du réseau électrique.

11 : Prépare les données à transmettre aux différents esclaves dans les messages prévus à cet effet (tension de référence, bit d'activation de l'équilibrage et bit d'activation de l'esclave).

12 : Construit les messages à transmettre aux esclaves selon le contenu des messages. Effectue les transmissions et les vérifie avec des lectures. Si l'une des écritures se passe mal, essaye d'écrire 5 fois puis essaye de transmettre encore une fois avec l'erreur mise à jour. L'erreur « AT_WRITE » est générée.



3.1.3.1 Régulation

La Figure 3-7 présente le fonctionnement de l'étape 8 de l'interruption ADC, c'est-à-dire la régulation. Les parties vertes avec les flèches noires concernent le DSCC et le SSBC. En revanche les parties bleues ne concernent que le SSBC et les parties rouges que le DSCC.

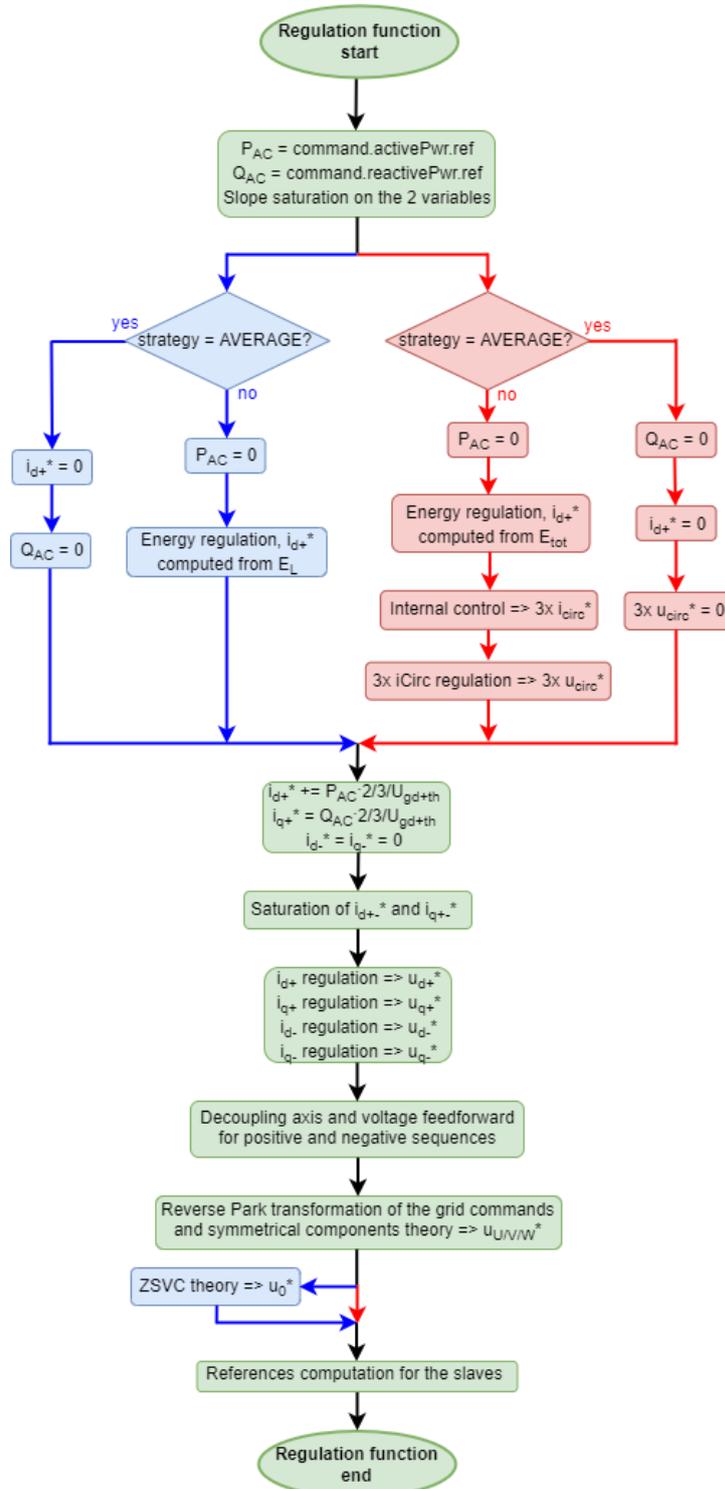


Figure 3-7: Structogramme de la fonction de régulation.



3.2 CARTE ALEXANDRE

Selon la Figure 3-1, la carte Cléopâtre fournit à toutes les cartes Alexandre un signal de synchronisation à une fréquence de 5 kHz et un rapport cyclique de 50%. La carte Alexandre a 2 interruptions :

- XINT3 : cette interruption est levée à chaque flanc montant du signal de synchronisation. Elle permet d'effectuer la synchronisation des différents PWMs sans risque de manquer une commutation PWM. En effet, la synchronisation est effectuée un PWM l'un après l'autre et toujours lorsque son compteur est égal à la valeur maximale.
- XINT4 : Cette interruption est levée à chaque flanc descendant du signal de synchronisation. Elle effectue tout le contrôle du bras, à savoir les mesures, la communication, la sécurité et la gestion des signaux PWMs. Sa durée est représentée par le temps t_a . Plus de détails sur cette interruption sont donnés dans la section 3.2.2.

A noter qu'une autre interruption, celle du timer 0, est utilisée. Elle permet la gestion de la led d'erreur (rouge). Plus de détails sont donnés dans la section 3.2.3.

3.2.1 Structures

A nouveau, il est important de présenter les structures qui composent le code avant d'expliquer le fonctionnement du firmware.

3.2.1.1 *S_Slave*

L'objet « slave » est déclaré de ce type-là, il permet de gérer la communication avec le master (carte Cléopâtre). Il est composé de :

Nom	Type	Description
messToMaster	S_MessageToMaster	Message transmis au master (carte Cléopâtre)
messFromMaster	S_MessageFromMaster	Message provenant du master (carte Cléopâtre)
address	Uint16	Adresse du slave (carte Alexandre)

Tableau 3-26 : structure S_Slave

S_MessageToMaster

Cette structure est comprise dans les objets de type « S_slave ». La structure est composée de :

Nom	Type	Description
rawMessage	Uint16	Message brut à transmettre au master
uSigma	float	Tension totale du bras (somme des différents sous-modules)
capUnbalanced	Uint16 :1	Actif si la tension d'une capacité est trop éloignée de la moyenne
error	Uint16 :1	Actif si une erreur a lieu dans le contrôle des sous-modules
parityBit	Uint16 :1	Bit de parité pour la détection d'erreur de communication

Tableau 3-27 : structure S_MessageToMaster

S_MessageFromMaster

Cette structure est comprise dans les objets de type « S_slave ». Elle contient la traduction du message envoyé par le master. La structure est composée de :

Nom	Type	Description
rawMessage	Uint16	Sample & hold du message brut reçu du master
nRef	float	Référence de l'indice d'insertion
armCurrentDir	int	Direction du courant dans le bras où sont les sous-modules
ack_disUndervoltage	Uint16 :1	Quittancement/désactivation de l'erreur de sous-tension
enable	Uint16 :1	Autorisation de fonctionnement général (seulement pour le DSCC)
parityBit	Uint16 :1	Bit de parité pour la détection d'erreur de communication

Tableau 3-28 : structure S_MessageFromMaster



3.2.1.2 S_Measures

L'objet « mesures » est déclaré de ce type-là, il permet d'avoir toutes les mesures accessibles facilement au même endroit. Actuellement il n'est composé que de la structure « voltage » de type « S_Sm_Value » car il n'y a que des tensions à mesurer. Cependant, de la sorte, il sera aisé d'ajouter d'autres mesures si besoin.

S_SmValue

Cette structure est comprise dans les objets de type « S_Measures ». La structure est composée de :

Nom	Type	Description
sm[2][NB_SM]	float	Tensions brutes des sous-modules. La 1 ^{ère} dimension est pour la mesure actuelle (0) et la mesure précédente (1). La 2 ^{ème} est pour les différents sous-modules.
sigma	float	Somme des tensions filtrées des sous-modules
average	float	Moyenne des tensions filtrées des sous-modules

Tableau 3-29 : structure S_SmValue

3.2.1.3 S_FlagError

L'objet « flagsError » est déclaré de ce type-là, il permet de regrouper toutes les erreurs possibles dans une seule structure. La structure est composée de :

Nom	Type	Description
synchronization	Uint16 :1	Actif si le signal de synchronisation n'est pas détecté
communication	E_COM_ERROR	Actif si une erreur de communication a eu lieu
voltageSM1	E_V_ERROR	Actif si une erreur de tension a eu lieu sur le sous-module 1
voltageSM2	E_V_ERROR	Actif si une erreur de tension a eu lieu sur le sous-module 2
voltageSM3	E_V_ERROR	Actif si une erreur de tension a eu lieu sur le sous-module 3
voltageSM4	E_V_ERROR	Actif si une erreur de tension a eu lieu sur le sous-module 4
voltageSM5	E_V_ERROR	Actif si une erreur de tension a eu lieu sur le sous-module 5

Tableau 3-30 : structure S_FlagError (Alexandre)

A noter que l'erreur de synchronisation ne détecte qu'une perte totale du signal de synchronisation. En effet, si le signal est présent mais que l'interruption n'est pas levée 1 fois sur 20, l'erreur n'interviendra pas. La vitesse de détection de cette erreur est lente (10 à 110 [ms]).

Le type « E_COM_ERROR » est un type enum qui peut prendre les valeurs suivantes :

- NO_ERROR_COM : il n'y a pas d'erreur de communication.
- AT_READ : une erreur de communication a été détectée à la lecture.
- AT_WRITE : une erreur de communication a été détectée à l'écriture.
- Le type « E_V_ERROR » est un type enum qui peut prendre les valeurs suivantes :
 - o NO_ERROR_V : il n'y a pas d'erreur de tension.
 - o OPTICAL : la fibre optique du sous-module est débranché ou présente un défaut.
 - o OVER_VOLTAGE : une surtension sur le sous-module a été détectée.
 - o UNDER_VOLTAGE : une sous-tension sur le sous-module a été détectée.
 - o VARIATION : la tension du sous-module a varié trop rapidement.

Cette structure est construite de manière à ce qu'elle vaille zéro si aucune erreur n'est détectée.



3.2.1.4 S_Control

L'objet « control » est déclaré de ce type-là, il permet le contrôle effectif des sous-modules. La structure est constituée de :

Nom	Type	Description
enableSW	Uint16 :1	Autorisation des sorties générée par la machine d'état
enableFromMaster	Uint16 :1	Autorisation des sorties générée par le master
ack_disUndervoltage	Uint16 :1	Quittancement/désactivation de l'erreur de sous-tension
armCurrentDir	int	Direction du courant dans le bras où sont les sous-modules
nRef	float	Référence de l'indice d'insertion
nRefA[NB_SM]	float	Référence de l'indice d'insertion de la branche A
nRefB[NB_SM]	float	Référence de l'indice d'insertion de la branche B complémentaire (seul pour le SSBC)

Tableau 3-31 : structure S_Control

La Figure 3-8 montre de manière visuelle les différents objets (en vert) et leur contenu (sous-structure dans les encadrés).

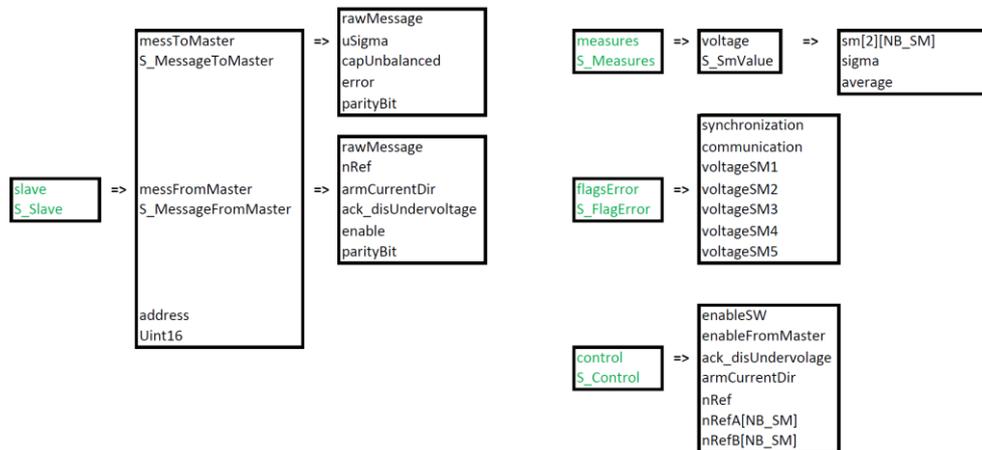


Figure 3-8: structures des différents objets dans le firmware Alexandre



3.2.2 Interruption de contrôle (XINT4)

La Figure 3-9 présente le fonctionnement de l'interruption de contrôle.

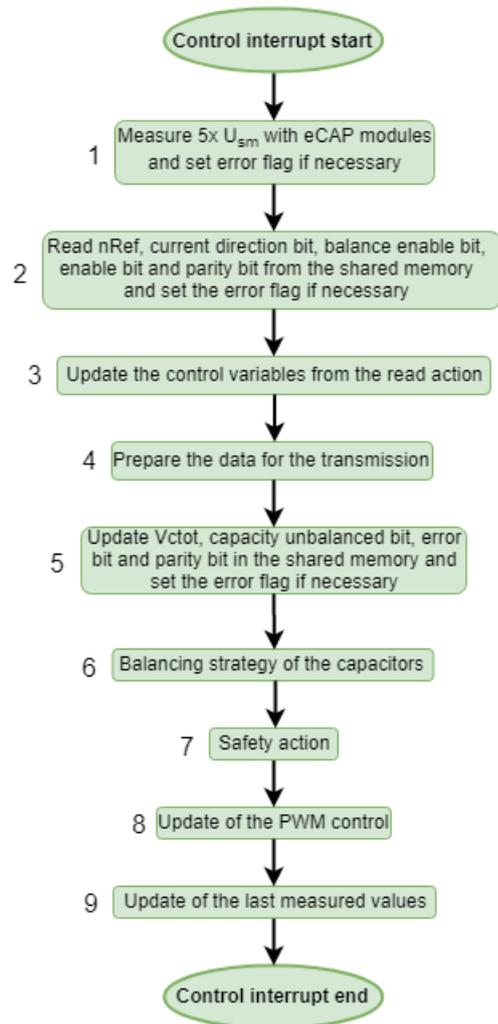


Figure 3-9 : structogramme de l'interruption de contrôle

1 : Mesure la tension des 5 sous-modules via les modules ECAP, filtre ces mesures, fait la somme et la moyenne de ces mesures.

Active le flag d'erreur correspondant pour chaque sous-modules si nécessaire dès que les sorties sont actives. Attention l'erreur « OPTICAL » est prioritaire. L'erreur « UNDER_VOLTAGE » est désactivée lors d'une erreur et au démarrage, la tension du sous-module doit passer 5V au-dessus de la valeur d'erreur pour activer cette erreur.

2 : Fait le sample & hold du message provenant du master. Lit et sauvegarde les informations du message. Le flag d'erreur communication « AT_READ » est levé si le bit de parité est faux.

3 : Met à jour les variables de contrôle à partir de la lecture.

4 : Prépare les données à transmettre au master dans le message prévu à cet effet (vSmTotal, capUnbalanced et error).

5 : Construit le message à transmettre au master selon le contenu du message. Fait la transmission et la vérifie avec une lecture. Si l'écriture se passe mal, essaie d'écrire 5 fois puis essaie de transmettre encore une fois avec l'erreur mise à jour. L'erreur « AT_WRITE » est générée.

6 : Effectue la stratégie d'équilibrage des condensateurs si cela est demandé et donne les références pour les signaux PWM.

7 : Gère le contrôle PWM en fonction des erreurs. Pour le DSCC, désactivation des PWM s'il y a n'importe quelle erreur.

8 : Mise à jour du contrôle des PWM.

9 : Mise à jour des mesures précédentes par les mesures actuelles.

3.2.3 Interruption TIMER0 (led rouge)

Cette interruption est levée toutes les 100 [ms] et permet de vérifier la bonne prise en compte du signal de synchronisation ainsi que de contrôler la led rouge afin d'effectuer un diagnostic d'erreur.

Afin de vérifier la synchronisation, une variable est incrémentée à chaque passage dans l'interruption de synchronisation (XINT3). Lors du passage dans l'interruption timer0, si cette variable est incrémentée de moins de 90% de ce qu'elle devrait, l'erreur est générée.

Lorsqu'une erreur apparaît, la led rouge clignote le nombre de fois qu'est le numéro de l'erreur à une fréquence de 2.5 [Hz] suivi d'une pause (éteinte) de 800 [ms]. Au démarrage ou en cas d'erreur inconnue, la led rouge reste allumée.

Le diagnostic d'erreur ne permet pas de savoir quel sous-module de puissance est en erreur ou s'il y en a plusieurs. De plus, si plusieurs erreurs sont actives en même temps, c'est toujours celle ayant le numéro le plus bas qui sera affichée.

Les numéros d'erreur sont répertoriés dans le Tableau 3-32.



N°	Type	Nom	Description
1	Synchronisation	Synchronisation	Le signal de synchronisation n'est pas détecté.
2	Communication avec le master	Lecture	La lecture du message provenant du master est erronée.
3		Ecriture	L'écriture du message pour le master est erronée.
4	Tension des sous-modules de puissance	Optique	Une fibre optique est déconnectée ou défectueuse.
5		Surtension	Une tension trop élevée a été mesurée.
6		Sous-tension	Une tension trop faible a été mesurée durant le fonctionnement.
7		Variation	Une tension a varié trop rapidement. Il y a un problème avec le module eCAP du DSP ou la fibre optique.

Tableau 3-32 : explications des erreurs possibles dans le firmware Alexandre

3.2.4 Stratégie de gestion des erreurs

Lorsqu'une erreur intervient, les variables de contrôle, notamment l'autorisation de fonctionnement générale, sont désactivées. De plus, l'information d'erreur est remontée à la carte Cléopâtre qui désactive l'autorisation de fonctionnement mais ne peut pas connaître le type d'erreur. Le quittancement des erreurs de la carte Alexandre est autorisé au plus tôt 1 seconde après que l'erreur soit intervenue. Ce quittancement permet de passer l'esclave dans l'état ready où les sorties sont désactivées et attend l'autorisation de la carte Cléopâtre. Ce fonctionnement est représenté par la machine d'état de la Figure 3-10.

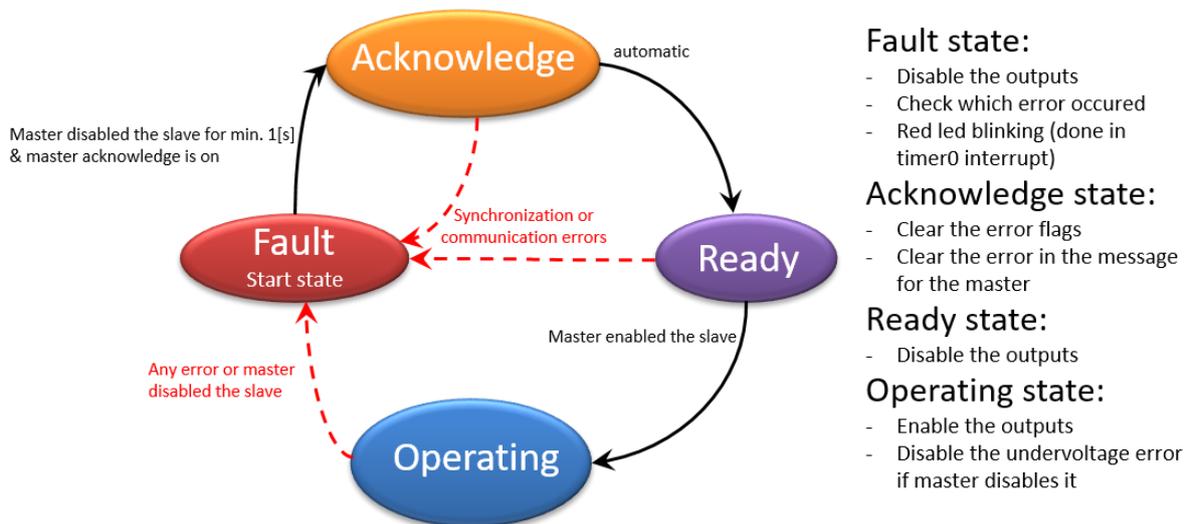


Figure 3-10: machine d'état de la carte Alexandre

3.2.4.1 Mesures de tension

Les mesures de tensions posent certains problèmes dus au hardware. En effet, lorsque la tension à mesurer est faible, le signal de mesure est aléatoire. Il n'est donc pas possible de savoir si la fibre optique est débranchée (erreur optique) ou si la valeur mesurée est correcte. Afin de contourner ce problème, les mesures de tension sont activées lorsque les critères suivants sont remplis :

- Atteindre au moins une fois l'état READY. Cela permet d'éviter tout problème lié à l'enclenchement non synchronisée des alimentations entre les différentes cartes.
- La mesure dépasse 3 [V] de manière continue durant 15 [ms]. Cela permet d'exclure la majorité des fausses mesures :
 - o Si, durant les 15 [ms], la mesure est plus élevée que 250 [V], le compteur temps est remis à zéro. Cela permet d'exclure les mesures soudaines à pleine échelle.

Les mesures sont à nouveau inactives lorsque la tension redescend en-dessous de 2 [V]. Lorsque les mesures de tensions sont inactives, les mesures sont forcées à zéro tandis que l'erreur optique n'est pas prise en compte.



Concernant l'erreur de sous-tension, elle est disponible lorsque la mesure de tension passe 15 [V] au-dessus de la valeur d'erreur. Cela permet d'éviter des erreurs intempestives lors du démarrage avec des tensions faibles. Cette erreur est désactivée dans l'état « Fault » et « Acknowledge » ainsi que dans l'état « Operating » si la carte Cléopâtre est dans l'état « Turn off ». Elle doit à nouveau être réactivée selon la condition ci-dessus.

4 CONCLUSION

Après avoir montré l'allure physique réelle des prototypes basse tension des convertisseurs SSBC et DSCC, le détail de la conception a été présenté. La partie hardware montre l'interconnexion entre les différentes cartes électroniques composant les convertisseurs. Elle montre aussi le détail du câblage des différents éléments et permet de se rendre compte de la complexité de conception de tels systèmes. Par la suite, la conception du firmware est expliquée et la structure du code est expliquée.

Dans l'état actuel, les prototypes sont prêts à être utilisés pour les essais en laboratoire afin de tester leur comportement réel et leur apport dans un réseau déséquilibré ou présentant un défaut.

5 BIBLIOGRAPHIE

- [1] G. Grosjean, G. Courteau, S. Kissling, M. Bozorg et M. Carpita, «D1.2 - Report on Modular STATCOM structures: Simulations of the selected structure,» COSTAM Deliverable, Yverdon-les-Bains, Switzerland, 2021.