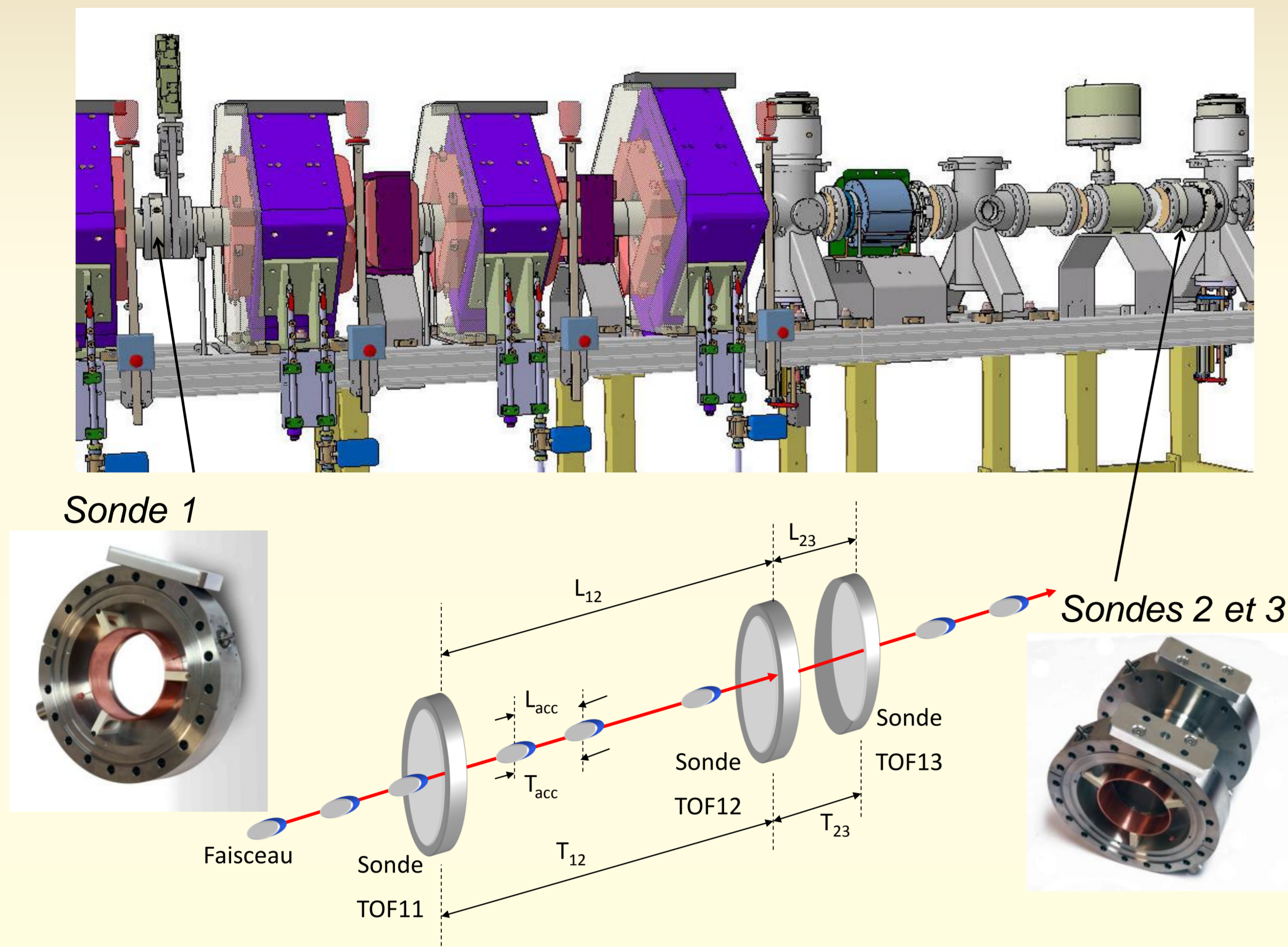


RÉSUMÉ

Un système de mesure et de contrôle de l'énergie du faisceau [1] est situé à la sortie du LINAC de l'accélérateur SPIRAL2. Ce dispositif est nécessaire, pour le réglage et le suivi du faisceau, pour assurer la protection thermique de l'accélérateur et pour contrôler le respect du domaine de fonctionnement de l'accélérateur. Ce système, en cours de conception, doit répondre à des exigences de sûreté de fonctionnement. Suite à une Analyse des Modes de Défaillances et de leurs Effets (AMDE) effectuée par un prestataire extérieur, un certain nombre de recommandations ont été formulées. Certaines d'entre elles concernent le développement du logiciel du microcontrôleur utilisé dans l'électronique de mesure et de contrôle de l'énergie. Après une description du principe de mesure de l'énergie, ce poster explique la méthodologie et le formalisme utilisés pour développer le logiciel, en sûreté et en qualité.

PRINCIPE DE MESURE DE L'ENERGIE

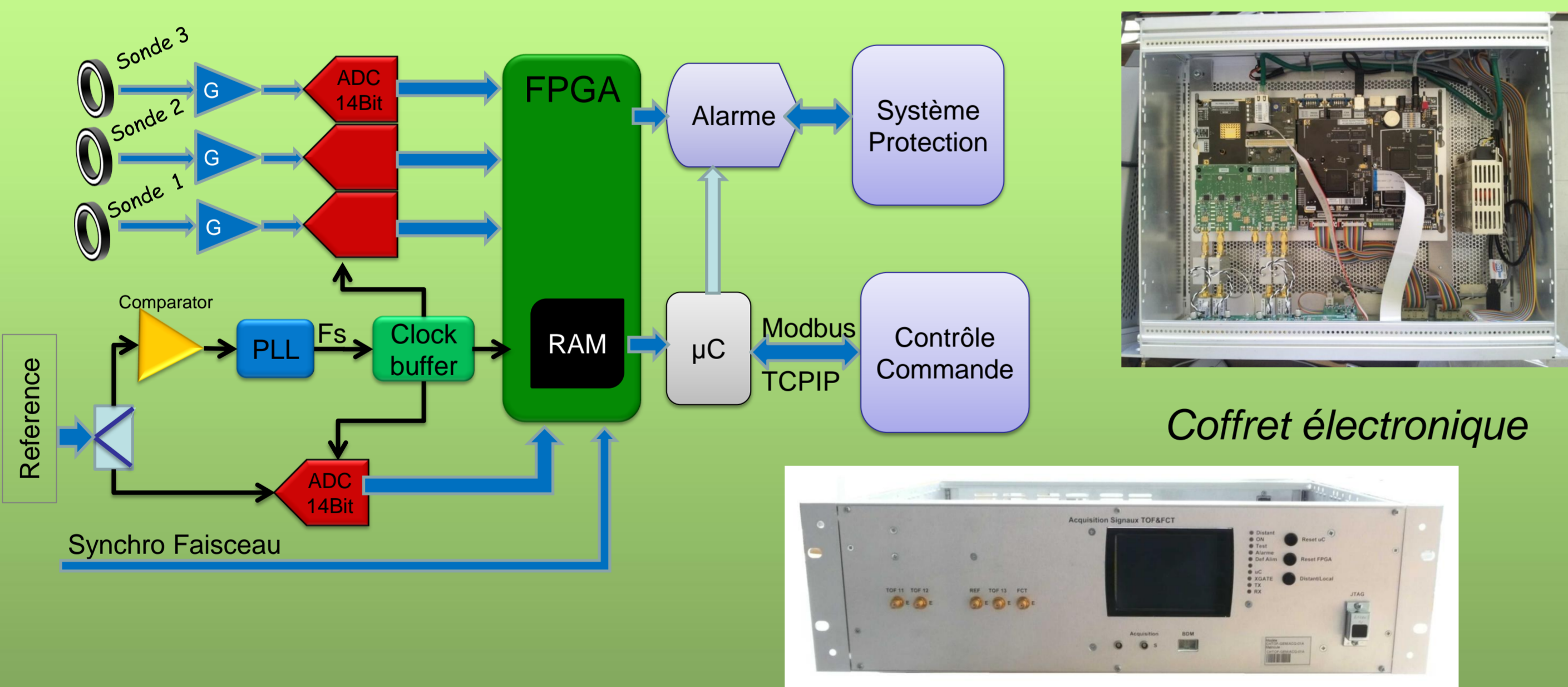
La détermination de l'énergie est basée sur le principe de la mesure du temps de vol (TOF : Time of Flight). Le faisceau traverse trois sondes capacitatives placées en sortie du Linac le long de la ligne haute énergie.



Le temps de vol des particules est déterminé par la mesure de la phase des signaux issus de chaque électrode par rapport à une référence commune. Connaissant la distance entre chaque capteur, la vitesse est quantifiée. L'énergie en est déduite à partir du type de la particule accélérée.

DESCRIPTION DE L'ELECTRONIQUE

La mesure de la phase est basée sur un système électronique qui numérise directement les signaux HF issus des sondes. Le traitement du signal est réalisé par un FPGA. Les valeurs traitées sont mises à disposition du microcontrôleur (μ C) par l'intermédiaire d'une mémoire RAM partagée entre les deux dispositifs.



Coffret électronique

DEVELOPPEMENT LOGICIEL

Un cycle de vie sur le modèle du 'Cycle en V', préconisé par la norme générique CEI 61508 est utilisé pour le développement du logiciel. La conception est réalisée suivant un plan qualité listant les activités et les procédures pour garantir un système sûr et exempt de défaillances systématiques. Les différentes phases de développement sont appliquées : cahier des charges, spécifications des exigences, architecture, conception détaillée du code, codage en langage C, test unitaire, test d'environnement et test de validation.

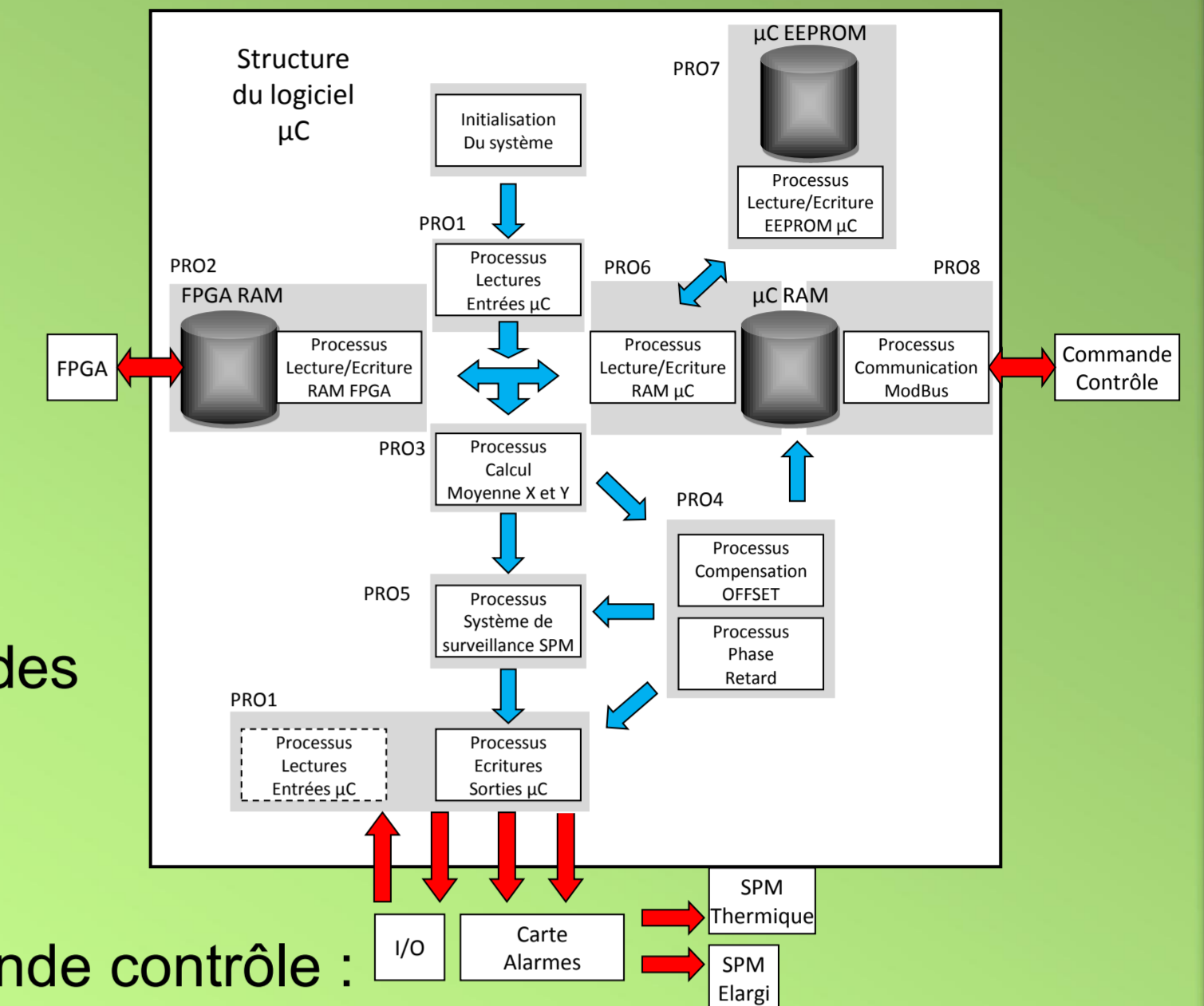
MICROCONTRÔLEUR

Le μ C de type HC12 à double processeurs de la société Freescale est basé sur une technologie éprouvée, performante et rapide. Le logiciel de développement CodeWarrior utilisé, offre de nombreux outils permettant le contrôle du code (MISRA-C, analyse statique, analyse des dépendances, ...)

RÔLE DU LOGICIEL μ C

Le μ C et son logiciel associé réalisent :

- ❑ des traitements de :
 - moyennage,
 - déduction,
 - calcul des amplitudes,
 - calcul des phases,
 - calcul de la vitesse,
- ❑ le contrôle et la surveillance :
 - alarme de dispersion de la vitesse,
 - alarme de la dispersion des amplitudes des signaux sondes,
 - alarme de dysfonctionnement des éléments constituant l'électronique,
- ❑ l'échange des données avec le commande contrôle :
 - mise à disposition : des acquisitions issues du FPGA, des résultats des calculs et des états du système.
 - les demandes spécifiques : enregistrements des offsets, des retards, des seuils de contrôle pour les Systèmes de Protection Machine (SPM).



MECANISMES DE SÛRETÉ DU CODE

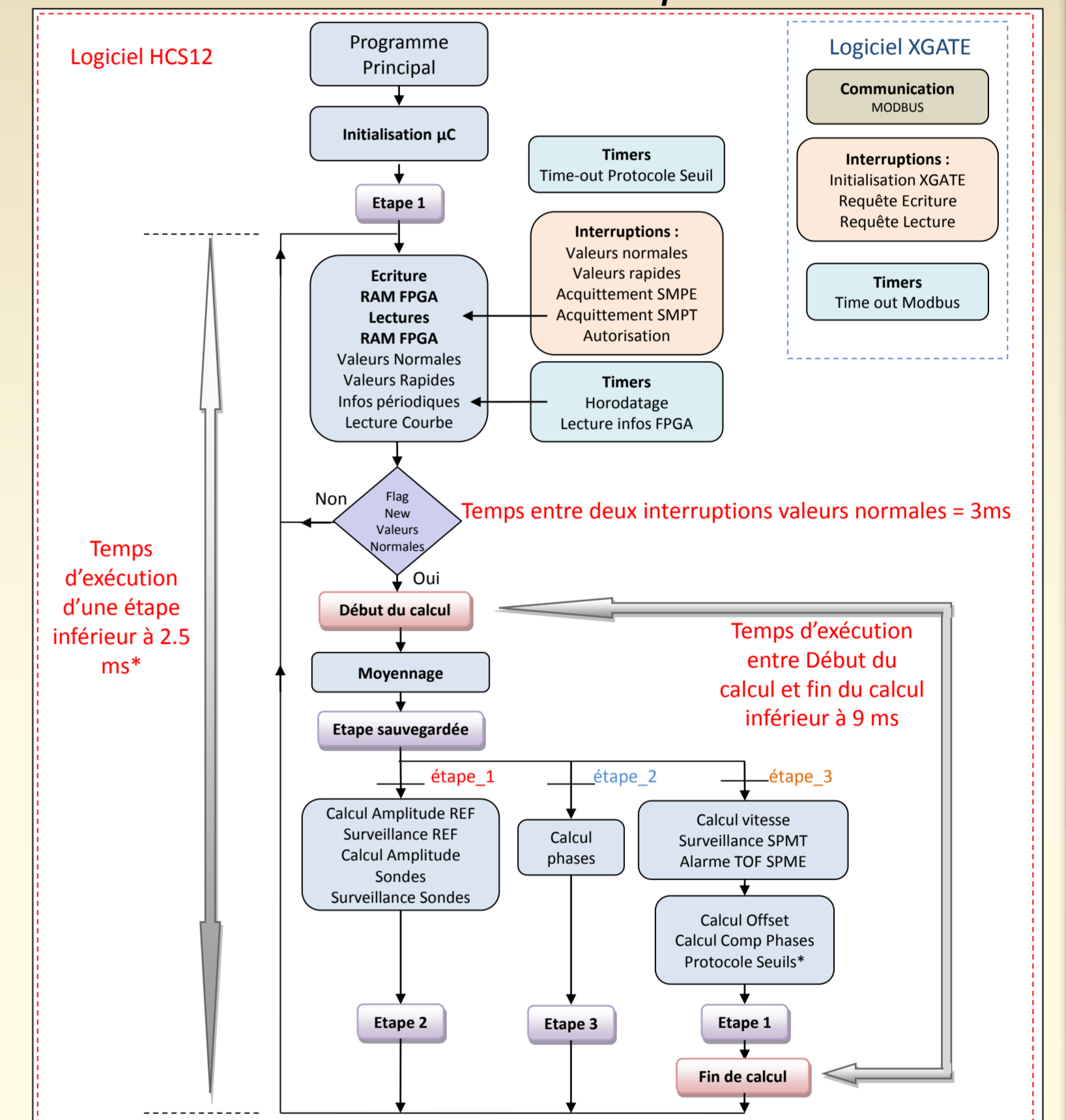
❑ Mécanisme de gestion de la qualité :

- Utilisation d'un plan de développement, du cycle en V.
- Traçabilité des exigences dans tous les documents.
- Tests et PV de tests.
- Gestion des évolutions du programme.

❑ Mécanisme lié à l'architecture du programme :

- Utilisation des deux coprocesseurs : un cœur est dédié aux traitements et alarmes, l'autre cœur gère la communication entre μ C et le Commande Contrôle.
- Ordonnement du programme en séquence pour exécuter l'intégralité des fonctions participant aux fonctions de sécurité.
- Contrôle temporel du cycle de déroulement du programme pour réaliser une surveillance extérieure de l'activité du programme.

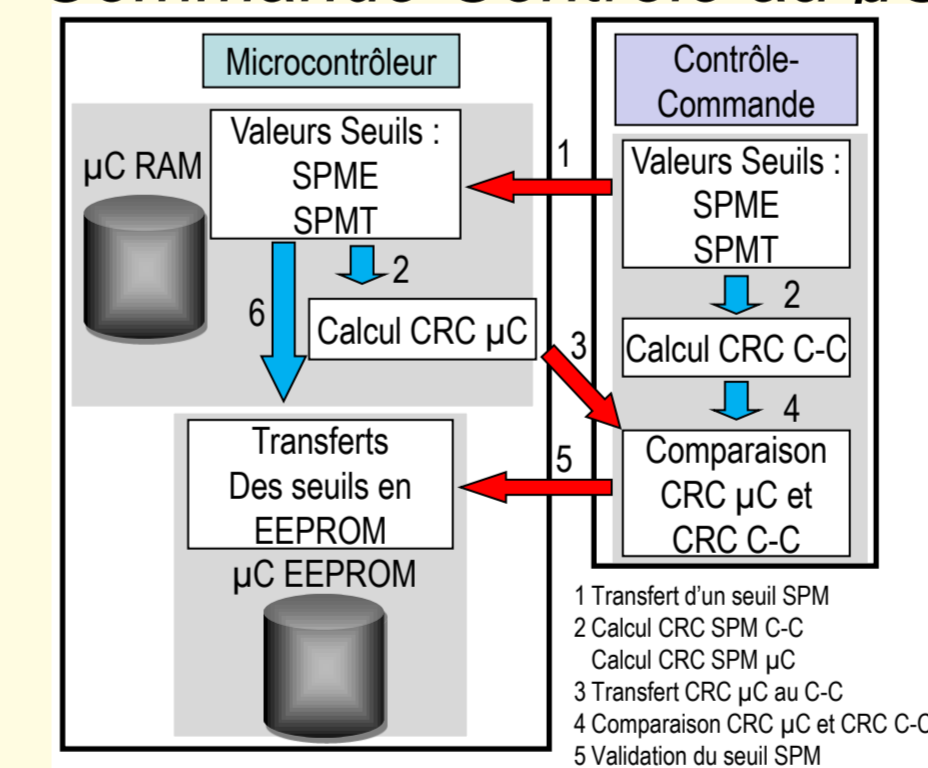
Architecture simplifiée



❑ Mécanisme de contrôle des données utilisées pour les fonctions alarmes :

- Description du protocole d'échange entre le Commande Contrôle et le μ C.
- Couverture du test unitaire à 100% de ces fonctions.
- Utilisation de contrôle de redondance cyclique (CRC) pour vérifier l'intégrité :
 - des données sauvegardées en mémoire non volatile.
 - des seuils échangés entre le commande contrôle au μ C.
 - des trames des requêtes ModBUS.
- Création d'une fonction gestion d'erreur, permettant la détection d'anomalies du système dans le cas d'un mauvais fonctionnement.

Protocole de transfert d'un seuil de Commande Contrôle au μ C



❑ Mécanisme de codage en langage C :

- Application d'un référentiel interne pour la codification des fonctions, variables, constantes, commentaires...
- Ecriture du code dans le respect des règles de codage en langage C issues du standard MISRA-C:2004.
- Minimisation de l'utilisation de la conversion en virgule flottante avec recours prépondérant à la conversion en virgule fixe pour les calculs.
- Application d'un codage défensif : contrôle de la taille des données pour supprimer les débordements et les dénominateurs nuls.

CONCLUSION

Le principe de fonctionnement de ce système repose sur la numérisation directe des signaux issus des sondes avec un traitement rapide des données effectué par un FPGA. En plus des contraintes de mesures liées à un système d'instrumentation vient s'ajouter des notions de qualité et de sûreté. Ce dispositif est un Élément Participant à la Sûreté (EPS) et doit répondre à des exigences fortes en terme de sûreté de fonctionnement. Cela a imposé, au niveau des programmes, un développement et une écriture du code en langage C suivant des normes de sécurité fonctionnelle, l'emploi de mécanismes de contrôle d'intégrité et l'utilisation d'une architecture à double processeurs. Cette démarche mise en place sera désormais un référentiel pour les prochains applicatifs créés par le groupe électronique machine du GANIL.