

Gestion de l'Exécution de Phases de Production par Interface PLI

Patrice Caulier

Université Polytechnique Hauts-de-France, LAMIH, CNRS, UMR 8201, Valenciennes, France
INSA Hauts-de-France, Valenciennes, France
patrice.caulier@uphf.fr

Résumé

La quatrième révolution industrielle en marche permet de basculer de l'ère de la « *mass-production* » vers celle, attendue par les consommateurs de biens comme de services, de « *mass-customization* ». Elle concerne l'introduction, au niveau des équipements de production comme de leur pilotage, des innovations liées à l'Internet des objets et aux technologies du numérique (cobotique, réalité augmentée, fabrication additive, *cloud computing*) mais aussi l'exploitation de données issues du *big data* et du jumeau numérique. En effet, un tel changement d'ère nécessite la mise en place de systèmes de production « intelligents » (« *smart manufacturing* ») capables, d'une part, d'une flexibilité, d'une réactivité et d'une adaptabilité de la production accrues et, d'autre part, d'une allocation multicritère et optimale des ressources face aux dysfonctionnements internes et aux perturbations externes subies. La maîtrise de telles infrastructures passe par le développement d'architectures de pilotage cyber-physiques décentralisées ou modulaires. Par exemple, le niveau contrôle/commande des équipements de production doit, désormais, être capable de gérer, en autonomie, des phases, élémentaires et indépendantes, de la production. Selon cette approche, « l'orchestration » des phases d'une gamme de production donnée est rejetée au niveau supérieur de l'architecture de pilotage modulaire, celui de l'exécution de la production ou « *Manufacturing Execution System* » (MES).

L'article se situe à ces deux niveaux. Il se propose de présenter et d'illustrer une interface de dialogue entre, d'une part, les phases d'équipement prescrites des gammes de production définies et exécutées au niveau MES et, d'autre part, les phases d'équipement exécutives de ces gammes définies et exécutées au niveau contrôle/commande : l'interface PLI (*Phase Logic Interface*) introduite par la norme ANSI/ISA-S88.01 (ou CEI/IEC 61512). L'illustration est réalisée et validée sur la cellule de production flexible du GIS S-mart, pôle Nord-Pas de Calais, plateforme de Valenciennes. Enfin, au-delà du modèle d'exécution cyclique classique de la norme CEI/IEC 61131-3 sur lequel s'appuie les parties précédentes, l'article propose, en perspective, d'étendre l'interface PLI aux blocs fonctionnels de la norme CEI/IEC 61499. Cette dernière norme définit l'architecture de pilotage des CPPS (*Cyber-*

Physical Production System) selon un modèle d'exécution, en plein essor, novateur, objet, événementiel, distribué et interopérable.

1 Introduction

Notre article s'inscrit dans le contexte général des systèmes cyber-physiques (*Cyber-Physical Systems, CPS*) formés d'entités collaboratives, dotées de capacités de calcul, en connexion intensive avec le monde physique, fournissant et utilisant à la fois les services de mise à disposition et de traitement de données disponibles sur les réseaux (Monostori, 2014 ; Qi *et al.*, 2020). Plus particulièrement, il se situe au niveau, d'une part, contrôle/commande des équipements physiques de production individuels et, d'autre part, de l'exécution de la production ou « *Manufacturing Execution System* » (MES). Chaque équipement physique de production doit être capable de gérer, en autonomie, une ou plusieurs phases, élémentaires et indépendantes, de la production. La gestion de ces phases d'une gamme de production donnée est, désormais, rejetée au niveau supérieur de MES. L'objet principal de cet article est d'exposer puis d'illustrer l'interface indispensable au dialogue efficace entre ces deux niveaux : l'interface PLI (*Phase Logic Interface*) et le protocole de communication qu'elle permet.

L'article est structuré en quatre parties. Nous commençons par exposer le contexte opérationnel de notre travail. Il est composé de la cellule de production flexible (CPF) du GIS S-mart, pôle Nord-Pas de Calais, plateforme de Valenciennes et de son architecture de pilotage en cinq niveaux complémentaires. La deuxième partie décrit les niveaux 3, MES, et 1, contrôle/commande, de cette architecture. La partie suivante présente l'interface de dialogue PLI entre ces deux niveaux. La mise en œuvre opérationnelle de cette interface est illustrée dans la partie quatre. Enfin, au-delà du modèle d'exécution cyclique classique de la norme CEI/IEC 61131-3 sur lequel s'appuie la partie précédente, l'article propose, en perspective, d'étendre l'interface PLI aux blocs fonctionnels de la norme CEI/IEC 61499. Cette dernière norme définit l'architecture de pilotage des CPPS (*Cyber-Physical Production System*) selon un modèle d'exécution, en plein essor, novateur, objet, événementiel, distribué et interopérable.

2 Contexte

Dans cette première partie, nous posons le cadre dans lequel le travail synthétisé dans cet article a été mené. Il est composé, d'une part, de la CPF et, d'autre part, de l'architecture de pilotage de cette dernière.

2.1 La Cellule de Production Flexible

Le travail relaté dans cet article a été développé dans le cadre de la CPF du GIS S-mart, pôle Nord-Pas de Calais, plateforme de Valenciennes, figure 1 (aip, 2021). Cette CPF se situe au niveau 0 de l'architecture de pilotage présentée au paragraphe suivant. Elle est constituée d'un réseau de convoyage monorail flexible. La flexibilité est apportée par la présence, d'une part, de plusieurs boucles de circulation et, d'autre part, d'aiguilles rotatives commandées. Sur ce réseau de convoyage peuvent circuler des *shuttles* motorisés *Montratec* capables d'accueillir les pièces composant le produit fini objet de la production. Une fois démarré, un *shuttle* se déplace par tronçons selon un sens unique de circulation et dessert sept unités de traitement :

- a. Quatre robots manipulateurs STÄUBLI TX2-60, six axes anthropomorphes RRRRRR, de montage de pièces.

- b. Un manipulateur cartésien PPR de chargement/déchargement des supports de pièces du produit fini (unité de démarrage).
- c. Un poste optique de contrôle qualité.
- d. Un poste de traitement manuel des produits finis non qualifiés.

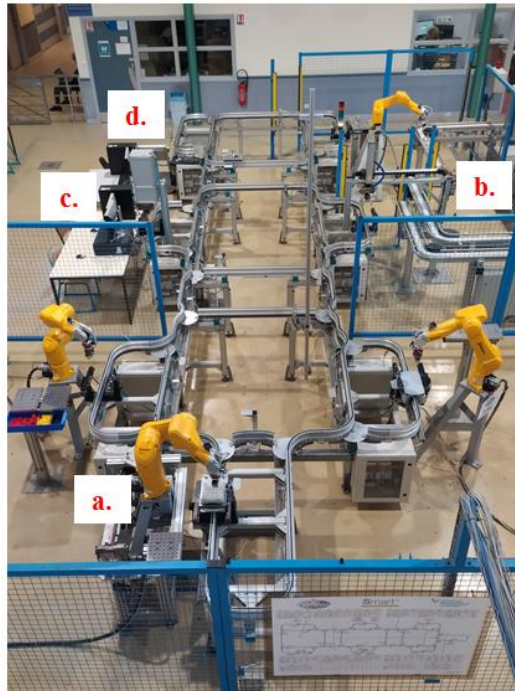


Figure 1 : La CPF du GIS S-mart, pôle Nord-Pas de Calais, plateforme de Valenciennes

La présence des *shuttles* aux divers points de démarrage des tronçons est identifiable, en lecture mais aussi en écriture, via des tags RFID 13.66 MHz *Schneider Electric (SE) OsiSense*, fixes et embarqués. Ils permettent un routage dynamique des *shuttles* motorisés sur le réseau de convoyage. Les baies de contrôle/commande des robots, l'automate de contrôle/commande du manipulateur cartésien, la caméra IP *Cognex* de l'unité d'inspection et les tags RFID sont sur réseau *Modbus Ethernet*. Les entrées et sorties du réseau de convoyage monorail flexible sont sur réseaux *AS-i* reliés aux quatre automates de contrôle/commande. Les unités de traitement permettent la réalisation de différentes gammes de produits finis constituées de diverses pièces assemblées et inspectées.

Le pilotage de cette CPF est assuré par une architecture multi-couches exposée dans le paragraphe suivant.

2.2 L'Architecture de Pilotage

L'architecture de pilotage de la CPF est structurée en cinq niveaux fonctionnels depuis le niveau 0 à sa base jusqu'au niveau 4, figure 2. Ce type d'architecture est actuellement le plus abouti (Caulier, 2021). Précisons toutefois que seuls les deux premiers niveaux 0 et 1 sont indispensables à un mode de marche automatique de la CPF.

Le niveau 0 est le niveau *process* ou opératoire. Il se compose de l'ensemble des équipements physiques constituant la CPF telle que décrite dans la section précédente. Ces équipements apportent les valeurs ajoutées nécessaires à l'obtention du produit fini.

Au niveau 1, la CPF est contrôlée et commandée, de façon distribuée, par quatre automates *SE Modicon M580* certifiés ANSSI-CSPN-2020/20, en réseau *Modbus Ethernet* identique à celui évoqué au paragraphe précédent. Les variables d'E/S des automates sont sur réseaux *AS-i* raccordés via des coupleurs *AS-i SE TSX SAY 1000*. Pour sa part, le manipulateur cartésien PPR est contrôlé et commandé par un automate *WAGO 750-842* également relié au réseau *Modbus Ethernet*. Lorsque la commande nécessite des échanges de variables entre ces automates, ils utilisent le service *I/O Scanning*. Les antennes RFID sont-elles adressées via des requêtes en lecture/écriture. En amont, cette couche de contrôle/commande est interfacée avec, d'une part, la couche supervision et, d'autre part, l'application de MES.

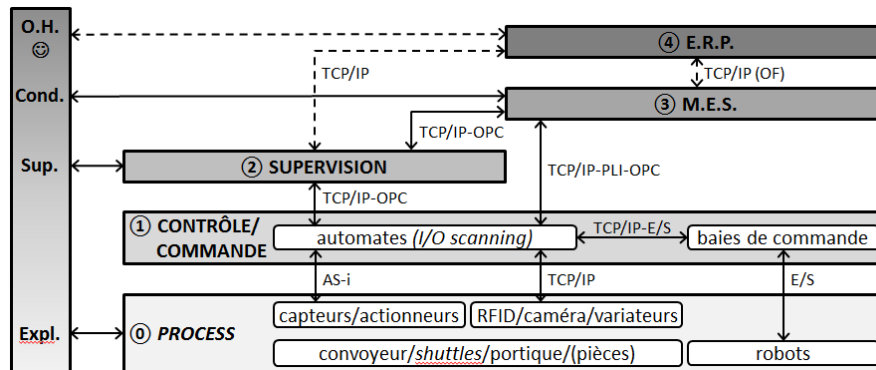


Figure 2 : Architecture de pilotage

L'application de supervision, ou SCADA (*Supervisory Control And Data Acquisition*), constitue le niveau 2. Au-delà de la conduite de la production en fonctionnement nominal et à l'aide de vues synoptiques, l'application de supervision est, aussi et surtout, consacrée à la surveillance, au pronostic et au diagnostic, d'une part, des défaillances (internes) des moyens de production et, d'autre part, des perturbations (externes) subies (Delépine *et al.*, 2008, Caulier et Vanderhaegen, 2011). Elle est développée à l'aide du progiciel de supervision PcVue ® (éditeur ARC Informatique ; Pcvue, 2021). Elle est connectée aux automates du niveau 1 par autant d'alias clients d'un serveur OPC (Caulier et Békrar, 2017).

La définition, l'exécution et la conduite des gammes ou procédures de production sont gérées par le niveau 3 de « *Manufacturing Execution System* » (MES). *In fine*, les phases d'équipement qui composent une gamme ou une procédure de production sont exécutées au niveau 1 par autant de phases de commandes élémentaires. Pour cela et comme pour l'application de supervision, l'application de MES est connectée aux automates du niveau 1 par autant d'alias clients d'un serveur OPC. Toutefois, ici, le contrôle de l'exécution des phases d'équipement respecte l'interface PLI objet de cet article. Ainsi, le paragraphe 3 revient sur l'objet des niveaux 3 et 1.

Enfin, le niveau ultime de notre architecture de pilotage est celui de « *l'Enterprise Resource Planning* » (ERP). Ce progiciel de gestion intégrée de l'ensemble des processus d'une entreprise est pour nous, ici, la source des Ordres de Fabrication (OF) reçus et gérés par l'application de MES à laquelle il est directement connecté. Un OF précise, essentiellement, la recette selon laquelle sera exécutée telle ou telle gamme de production d'un lot.

Ces différents niveaux de l'architecture de pilotage se distinguent, non seulement par leurs fonctionnalités, mais aussi par leurs échelles temporelles. Alors que l'ordonnancement de la production, au niveau de l'ERP, travaille, au mieux, à la journée ou à la demi-journée, le MES doit, lui, être capable de réagir dans des durées de quelques minutes. Par ailleurs, chaque niveau est relié, au moins, à son niveau immédiatement supérieur par les divers protocoles cités figure 2. Compte tenu

de l'existence du niveau 3 de MES dans notre architecture de pilotage, les commandes développées au niveau 1 sont purement élémentaires et leur enchaînement décidé et contrôlé, selon l'interface PLI, par le niveau 3. Enfin, en dehors du niveau 1 de contrôle/commande, chaque niveau est interfacé avec un profil d'opérateur humain qui en assure la bonne gestion : l'opérateur humain d'exploitation est interfacé avec le niveau opérationnel 0, l'opérateur humain de supervision a en charge la surveillance et, le cas échéant, le traitement des alarmes, enfin, l'opérateur humain en relation avec l'application de MES s'intéresse à l'exécution et au contrôle de la production (Vanderhaegen et Caulier, 2011, 2013).

Le niveau 4 ERP et le niveau 2 supervision, centré, essentiellement, sur les modes de dysfonctionnement, n'entrent pas dans le champ de la problématique relatée dans notre article. Ils ne seront donc pas plus développés. La partie suivante décrit, plus particulièrement, l'objet des niveaux 3 de MES et 1 de contrôle/commande intéressés, comme le montre la figure 2, directement par l'interface PLI.

3 Niveaux MES et Contrôle/Commande

Après avoir décrit le contexte applicatif de notre travail, nous détaillons dans ce paragraphe l'objet du niveau 3 de MES puis, en conséquence, celui du niveau 1 de contrôle/commande. En effet, l'existence du niveau 3 de MES dans une architecture de pilotage amène à reconsidérer la finalité du niveau 1. Ce dernier n'est plus le lieu de la détermination des scénarios de contrôle/commande mais, plus simplement, celui de la définition et de l'exécution de phases de commandes élémentaires des actionneurs du niveau 0 décidées au niveau 3.

3.1 Niveau MES

Si l'on considère le niveau 4 ERP comme plutôt lié au système d'information de l'entreprise, le niveau 3 de MES se comporte alors comme le véritable « chef d'orchestre », côté système de production. En effet, alimenté par les OF de l'ERP, le MES est avant tout un système de pilotage et de suivi de la production. Il détermine les séquences d'opérations à réaliser et transmet ses instructions de production sous la forme de phases. En cours de production, par le biais du niveau 1 de contrôle/commande et des opérateurs humains d'exploitation, le MES récupère, au fil de l'eau, ce qui a été fabriqué, les conditions de fabrication et ce qui est en cours. Il peut aussi transmettre au système d'information de l'entreprise (ERP) les productions de produits finis, de co-produits et de déchets, les consommations de matières premières, de consommables et d'énergies, les prédictions de production ou, encore, les états des stocks (Jaskó *et al.*, 2020 ; Caulier, 2021).

Le niveau 3 de MES constitue, donc, un niveau maître et multifonctionnel. Son organisation s'appuie sur deux standards internationaux et complémentaires évoqués maintenant.

3.1.1 Normes

Le premier standard de la fonction MES de pilotage et de gestion de l'exploitation des équipements physiques de production est défini par la norme ANSI/ISA-95 (ou CEI/IEC 62264). Comme l'illustre le modèle fonctionnel de la figure 3, ce standard définit et structure neuf fonctionnalités du périmètre MES autour de la fonctionnalité centrale de définition, d'exécution et de contrôle de la production (Isa, 2021 ; Iec, 2021). Par exemple, le service d'assurance qualité va fournir au contrôle de la production les spécifications clients. En retour, ce dernier fournira au service d'assurance qualité les résultats de ses audits qualité. Ou bien encore, l'ordonnancement fournit au contrôle de la production le plan de production, lequel fournira, en réponse, la capacité de production restante.

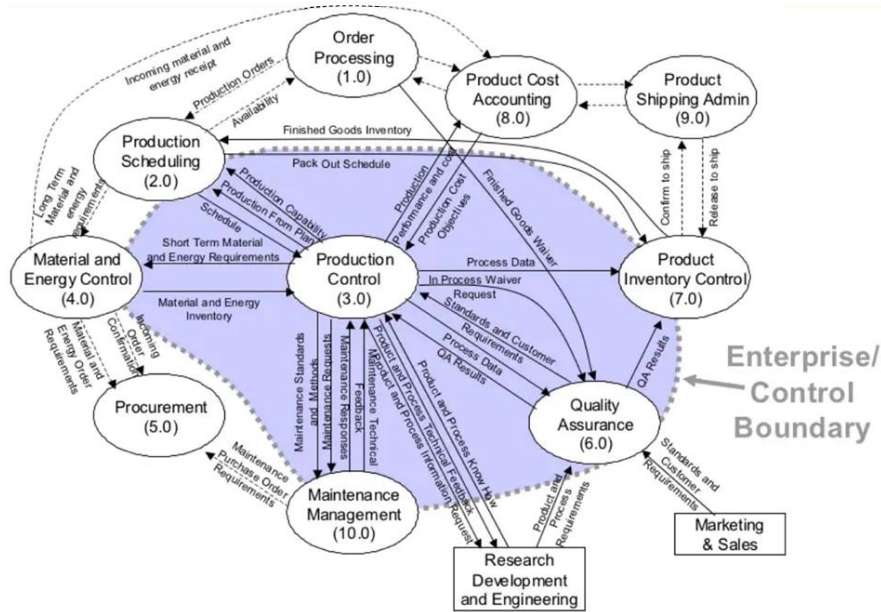


Figure 3 : Modèle fonctionnel de l'ANSI/ISA-95

A un niveau plus opérationnel, le second standard mis en œuvre par les applications de MES est précisé par la norme ANSI/ISA-S88.01 (ou CEI/IEC 61512). Il comporte essentiellement deux modèles : le modèle physique et le modèle procédural. Le modèle physique, figure 4, agrège l'organisation de l'entreprise puis des ressources, ou moyens de production, selon sept niveaux. Côté organisation des ressources (Isa, 2021 ; Iec, 2021) :

- Une cellule constitue l'espace de réalisation des procédures ou des gammes de production.
- Une unité est, le cas échéant, le regroupement fonctionnel de modules d'équipement.
- Un module d'équipement est composé de modèles d'équipement physique au niveau 0.
- Un module de contrôle/commande concerne un équipement physique réel.

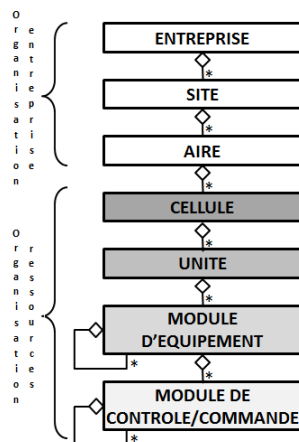


Figure 4 : Modèle physique de la norme ANSI/ISA-S88.01

Pour ce qui concerne la modélisation physique de notre CPF, nous verrons, au prochain paragraphe, que nous ne distinguons pas d'unités mais directement des modèles d'équipement.

Pour sa part, le modèle procédural concerne l'exploitation des ressources définies par le modèle physique. De la même façon, il agrège les éléments d'exploitation selon 4 niveaux. Il sera présenté au paragraphe suivant.

Par la maîtrise et l'optimisation de toutes les étapes de la production, depuis les OF jusqu'au produit fini, le MES participe à la transformation numérique de l'industrie. Il apporte à ses moyens intelligents et communicants de production une agilité accrue (Caulier, 2021). Dans la suite, nous limitons la présentation de notre application de MES à la fonction centrale de définition de la production. En effet, c'est elle qui est concernée par l'exécution et le contrôle de phases de production par l'intermédiaire de l'interface PLI objet de cet article.

3.1.2 Définition de la production

La fonction centrale d'une application de MES concerne la définition, l'exécution et le contrôle de la production. Sa mise en œuvre nécessite l'enchaînement des quatre étapes suivantes :

1. Modélisation des équipements physiques de la CPF.
2. Définition des procédures de production.
3. Programmation des tâches d'exécution élémentaires côté automates.
4. Intégration de l'interface PLI côtés MES et contrôle/commande.

Les deux premières étapes concernent la définition de la production par l'application de MES et sont décrites dans ce paragraphe. L'étape suivante traite de l'exécution proprement dite de la production et intéresse, donc, le niveau 1 de contrôle/commande. Elle sera abordée dans le paragraphe suivant. Enfin, la dernière étape est relative au contrôle de la production, côtés MES et contrôle/commande, selon les modalités définies par l'interface PLI. Elle est définie et illustrée dans la seconde partie de l'article. Les développements nécessaires à la réalisation de ces étapes ont été menés, côté MES, à l'aide du progiciel COOX[®] (éditeur ORDINAL Software ; Ordinal, 2021) et, côté automates *SE Modicon M580*, à l'aide de l'environnement de développement *ExoStruxure Control Expert*[®] (éditeur Schneider Electric). Le progiciel COOX comporte deux exécutables : le *Builder* et le *Manager*. Le *Builder* est utilisé pour le développement de l'application MES. L'exécution de l'application est traitée par le *Manager*. Elle peut aussi l'être via un client web. Enfin, un serveur OPC *SE OFS* a aussi été configuré et utilisé pour gérer la communication entre les deux niveaux fonctionnels 3 et 1.

Conformément au modèle physique de la norme ANSI/ISA-S88.01, la définition de la production débute par la création des modèles des équipements physiques du niveau 0 *process*. Pour notre application, les modèles d'équipement suivants ont été conçus et réalisés : Aiguille, Position shuttle simple, Position shuttle indexé, Position shuttle RFID, Poste VA et Pupitre. Le modèle Aiguille permet d'orienter les *shuttles* sur le réseau de convoyage. Le modèle Position shuttle simple identifie des positions discrètes du *shuttle* sur le réseau de convoyage. Le modèle Position shuttle indexé correspond à des positions aux unités de traitement de la CPF où le *shuttle* peut être indexé ou bridé en position. Le modèle Position shuttle RFID permet d'identifier le *shuttle* par la lecture et/ou l'écriture d'un tag RFID. Le modèle Poste VA correspond aux unités de traitement ou poste à valeur ajoutée. Enfin, le modèle Pupitre autorise la commande en mode manuel des équipements de la CPF et affiche, alors, un pupitre de commande par automate. Chacun de ces modèles recense ses propriétés, ses visuels, animés ou non, et la (ou les) phase(s) qu'il est capable d'exécuter. Une fois créés, les modèles d'équipement sont instanciés en équipements particuliers composant effectivement la CPF. Les visuels de ces instances sont utilisés pour construire un synoptique de conduite et de suivi de la production.

Le développement de l'application MES se poursuit avec la définition des modèles d'exploitation des ressources définies par le modèle physique. Cette approche se base sur le modèle procédural de la norme ANSI/ISA-S88.01, figure 5. Ce modèle structure une gamme de production d'une cellule, selon quatre éléments d'exploitation :

1. La procédure.
2. La procédure d'unité.
3. L'opération.
4. La phase.

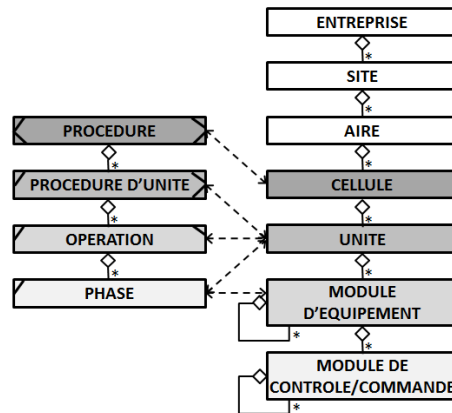


Figure 5 : Modèle procédural de la norme ANSI/ISA-S88.01

Une procédure est une stratégie d'exécution d'une production. Le domaine physique attaché à la procédure est la cellule avec ses équipements. La procédure occupe le plus haut niveau dans la hiérarchie des éléments procéduraux. Elle est définie sous la forme d'un ensemble ordonné de procédures d'unité, dont l'exécution s'effectue de manière séquentielle ou concurrente, ou de phases. Une fois associée à une recette, une procédure constitue une gamme exécutable suite à un OF issu de l'application d'ERP pour constituer, alors, un lot de production.

Une procédure d'unité est une stratégie pour dérouler un procédé dans une unité donnée. Une procédure d'unité est attachée à une unité spécifique ou à une catégorie d'unités. Elle est définie sous la forme d'un ensemble ordonné d'opérations s'exécutant de manière séquentielle ou concurrente. Plusieurs procédures d'unités peuvent s'exécuter concurrentement dans des unités différentes.

Une opération est un ensemble ordonné de phases. Elle est attachée à une catégorie d'unité disposant des phases d'équipements nécessaires pour la mise en œuvre des phases de cette opération. Les phases d'une opération peuvent s'exécuter de manière séquentielle ou concurrente.

Enfin, la phase est l'élément procédural le plus élémentaire. Elle est attachée à un équipement physique du *process* commandé. La commande de la production, un automate, ne peut exécuter que des phases. Dans notre contexte, des exemples de phase sont : démarrer un *shuttle* depuis une position, orienter une aiguille à droite ou bien, encore, désindexer un *shuttle* d'une position liée à une unité de traitement robotisée.

Nous avons commencé par recenser, analyser et définir l'ensemble des modèles de phase de notre application. Huit modèles de phases ont été identifiés :

1. Ph_ST_Shuttle_PsX : démarrage d'un *shuttle* depuis une position quelconque PsX.
2. Ph_Index_PsX : indexation d'un *shuttle* à une position quelconque PsX d'une unité de traitement.

3. Ph_Desindex_Psx : désindexation d'un *shuttle* présent à une position quelconque PSx d'une unité de traitement.
4. Ph_Rot_D_Ax : rotation à droite d'une aiguille quelconque.
5. Ph_Rot_G_Ax : rotation à gauche d'une aiguille quelconque.
6. Ph_Lect_RFID_Psx : lecture d'un tag RFID d'un *shuttle* à une position quelconque PSx.
7. Ph_Ecrit_RFID_Psx : écriture d'un tag RFID d'un *shuttle* à une position quelconque PSx.
8. Ph_Valeur_Ajoutée_Psx : exécution d'une valeur ajoutée à une position quelconque PSx d'une unité de traitement.

Ces modèles de phase sont ensuite associés aux modèles d'équipement précédents. Ainsi, par exemple, au modèle d'équipement *Position shuttle indexé*, nous associons les phases Ph_ST_Shuttle_Psx, Ph_Index_Psx et Ph_Desindex_Psx. Les modèles d'opération et de procédures d'unité de l'application sont également créés. À l'aide d'instances des modèles d'exécution et du langage PFC (*Procedure Function Chart*), des procédures peuvent alors être construites. Par exemple, la figure 6 présente la procédure Proc_PS1a2 permettant à un *shuttle* d'aller de la position PS1 à la position PS2 (sur notre CPF, un *shuttle* se déplace par tronçon délimité par une position de départ PSd et une position d'arrivée PSa). Elle est composée de deux procédures d'unité. Tout d'abord, la procédure teste la présence d'un *shuttle* en PS1. Puis, une sélection de deux séquences exclusives permet, le cas échéant, d'orienter l'aiguille 1 vers la gauche à l'aide d'une instance de la procédure d'unité Pu_Rotation_A1 (laquelle fait appel à l'instance de phase d'équipement : Ph_Rot_G_A1). De façon similaire, le *shuttle* est alors démarré. La procédure s'arrête lorsque le *shuttle* arrive en PS2.

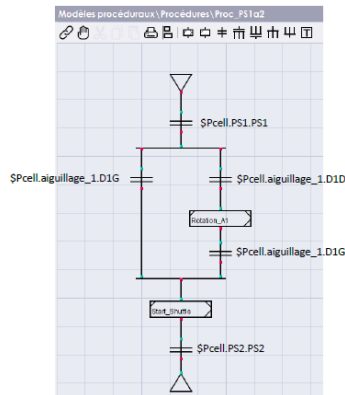


Figure 6 : Procédure PS1 vers PS2

Avant de pouvoir exécuter et contrôler une procédure à l'aide de l'application de MES, deux étapes demeurent. La première consiste à programmer les phases au niveau des automates de commande du niveau 1. La seconde vise à gérer la communication entre une phase de prescription MES et une phase d'exécution côté automate à l'aide de l'interface PLI. La suite de l'article décrit ces deux étapes.

3.2 Niveau Contrôle/Commande

Les phases activées par les procédures de l'application de MES sont exécutées au niveau 1 de contrôle/commande. Ce niveau est constitué de quatre automates *SE Modicon M580* en réseau *Modbus Ethernet*.

L'introduction de fonctions MES au sein d'une architecture de pilotage induit un point de vue nouveau et important sur le niveau contrôle/commande. Ce dernier n'est plus le lieu de la définition déterministe des modes de marche. Il devient un exécutant de phases de production élémentaires prescrites et orchestrées, désormais, au niveau MES. Cette approche impose de reconsidérer la commande des équipements de production :

- de façon élémentaire, fonctionnelle et modulaire,
- de respecter un protocole d'échanges rigoureux : l'interface PLI.

Ainsi, une analyse fonctionnelle de chaque type de phase d'exécution élémentaire sur le réseau de convoyage monorail flexible et les *shuttles* a été menée. Ensuite, à partir de ces types de phase, chaque instance de phase d'exécution particulière et élémentaire a été programmée. La programmation modulaire a été réalisée à l'aide du langage fonctionnel « *Function Block Diagram* » (FBD) de la norme ISO/CEI 61131-3 (Wcislik *et al.*, 2015). Chaque phase d'un équipement de production est donc définie par un bloc fonctionnel autonome dans le programme de tel ou tel automate. Si des variables doivent être échangées, en lecture et/ou en écriture, entre des automates, le service de communication *I/O Scanning* du protocole *Modbus Ethernet* est configuré et exploité. Par ailleurs, la gestion d'une variable *AutoManu* permet d'exécuter les phases en mode automatique ou en mode manuel. Chaque phase d'un équipement de production ne connaît que ses paramètres d'E/S et l'équipement sur lequel elle est exécutée. Elle ne sait pas à quel moment elle intervient dans la gamme de production. Ces instants sont définis par l'application de MES, « *manager* » de l'enchaînement des phases d'exécution. Cette approche de programmation fonctionnelle et modulaire permet la réutilisation des phases d'un équipement à l'autre et d'une gamme à l'autre et d'un projet à l'autre. Elle autorise aussi de « standardiser » le dialogue entre l'application de MES maître et les automates de commande esclaves. Ce standard constitue l'interface PLI présentée maintenant.

4 Présentation de l'Interface PLI

La description de procédures de production par une application de MES conduit à la définition de phases d'équipement. Ces phases sont de nature prescriptive. Afin d'assurer leur exécution effective par le *process*, les phases d'équipement prescriptives sont traduites côté application de contrôle/commande. On parle alors de phases d'équipement exécutives. La nécessaire coordination entre ces deux types de phase est assurée par l'interface PLI (*Phase Logic Interface*) initiée par la norme ANSI/ISA-S88.01, figure 7. Cette interface établit une couche logicielle de dialogue entre la gestion d'une phase d'équipement par le niveau 3 de MES et son exécution par le niveau 1 de contrôle/commande. Elle garantit une exécution contrôlée d'une phase d'équipement par échange de données synchronisé. L'interface PLI distingue plusieurs phases logiques, ou états, pour un équipement de production.

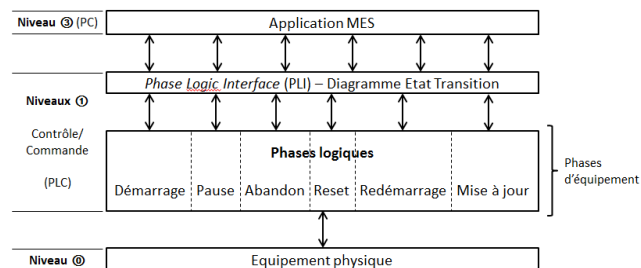


Figure 7 : Flux de communication de l'interface PLI

Le système séquentiel que constitue l'interface PLI se modélise sous la forme d'un diagramme états-transitions, figure 8. Sa gestion est partagée par les applications de MES et de contrôle/commande. Les états traduisent les modes de marche et d'arrêt que peut prendre une phase d'équipement. Ils sont au nombre de neuf numérotés, selon la norme, de 1 à 6 puis de 9 à 11 :

- <1-Initial> : c'est l'état de démarrage de toute phase d'équipement. Dans cet état, la phase est prête à recevoir le code commande {1-Démarrage}.
- <2-Exécution> : caractérise l'état d'une phase en cours d'exécution par l'automate.
- <3-Terminé> : la phase passe dans cet état une fois son exécution achevée.
- <4-Pause demandée> : une phase en cours d'exécution passe dans cet état après un code commande {2-Pause}.
- <5-Pause> : dans cet état, la pause demandée est obtenue.
- <6-Redémarrage> : à partir de l'état <5-Pause>, cet état est obtenu suite à un code commande {3-Redémarrage}.
- <9-Arrêt demandé> : à la réception d'un code commande {5-Arrêt}, une phase dans l'état <2-Exécution> ou <5-Pause>, évolue vers cet état.
- <10-Arrêt> : dans cet état, l'arrêt demandé est obtenu. Il nécessite un redémarrage en situation initiale par le code commande {6-Reset}.
- <11-Bloqué> : cet état permet à l'automate d'avertir le moteur d'exécution MES que la phase courante ne peut pas recevoir de code commande.

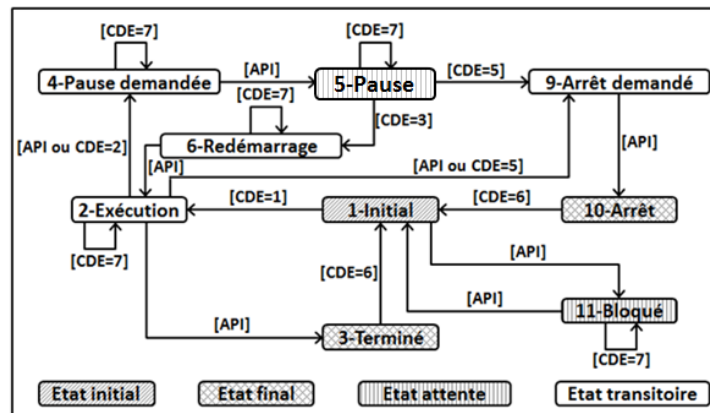


Figure 8 : Diagramme états-transitions de l'interface PLI

Quant à elles, les transitions d'états peuvent être à l'initiative, soit du niveau 1 de contrôle/commande (code commande [API] figure 8), soit du niveau 3 de MES par l'envoi de l'un des six codes commande suivants numérotés, selon la norme, de 1 à 3 puis de 5 à 7 :

- {1-Démarrage} : démarre l'exécution d'une phase d'équipement dans l'état initial.
- {2-Pause} : suspend la phase en cours d'exécution.
- {3-Redémarrage} : reprend l'exécution d'une phase suspendue.
- {5-Arrêt} : arrête une phase en cours d'exécution ou suspendue.
- {6-Reset} : provoque le retour en état initial d'une phase terminée ou arrêtée.
- {7-Mise à jour} : entraîne le test de certains états.

Le paragraphe suivant illustre maintenant, côté application de contrôle/commande ou automate, la mise en œuvre de cette interface PLI.

5 Mise en œuvre de l'Interface PLI

L'interface PLI permet un échange de données structuré avec contrôle de l'exécution d'une phase d'équipement. Cet échange est partagé entre l'application de MES et l'application de contrôle/commande. Il utilise quatre registres d'informations. Les registres $\$Validation$, à valeurs $[0,1]$, et $\$Commande$, à valeurs $[1,2,3,5,6,7]$, sont gérés par le moteur d'exécution MES de la phase. Les registres $\$Acquittement$, à valeurs $[0,1]$, et $\$Status$, à valeurs $[1,2,3,4,5,6,9,10,11]$ sont contrôlés par l'application de contrôle/commande. Au démarrage, la valeur des registres $\$Validation$ et $\$Acquittement$ est identique et celle du registre $\$Status$ égale à 1. La gestion d'une commande par chacune des parties suit alors le scénario suivant :

1. Côté MES : $\$Commande \leftarrow \langle N^\circ \rangle$.
2. Côté MES : $\$Validation \leftarrow \overline{\$Acquittement}$.
3. Côté contrôle/commande : à la réception de la mise à jour des registres $\$Commande$ et $\$Validation$ précédente, la commande demandée est traitée.
4. Côté contrôle/commande : $\$Status \leftarrow \langle N^\circ \rangle$.
5. Côté contrôle/commande : $\$Acquittement \leftarrow \$Validation$.

Ce scénario se répète à chaque nouvelle commande. Suite à des codes commande [API], le registre $\$Status$ est mis à jour, de façon autonome, par l'automate. C'est le cas, par exemple, entre les états $\langle 2-Exécution \rangle$ et $\langle 3-Terminé \rangle$, figure 8.

Côté application de contrôle/commande et pour la phase d'équipement exécutive Ph_ST_Shuttle_PS1, la figure 9 décrit, pour des raisons de lisibilité sous la forme d'un grafcet, la gestion de la boucle d'exécution : $\langle 1-Initial \rangle \rightarrow \langle 2-Exécution \rangle \rightarrow \langle 3-Terminé \rangle \rightarrow \langle 1-Initial \rangle$ de l'interface PLI figure 8.

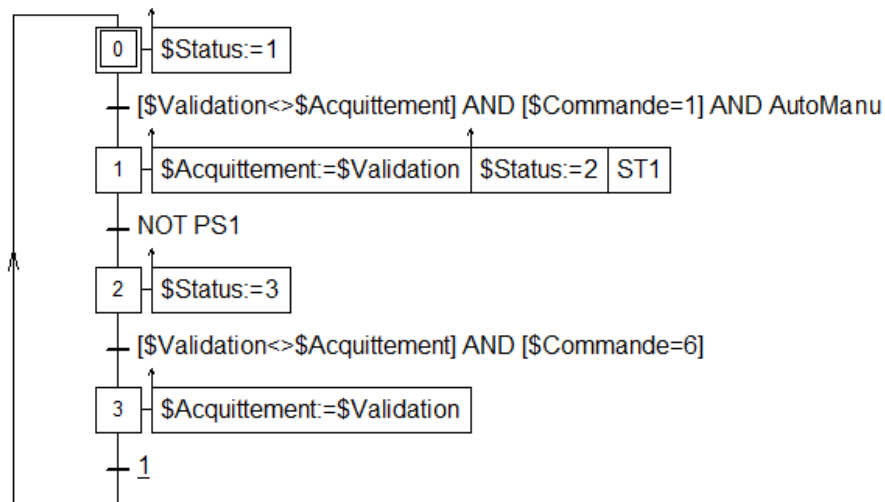


Figure 9 : Phase Ph_ST_Shuttle_PS1

Le grafcet peut être étendu à l'ensemble du diagramme états-transitions de la figure 8. La description opérationnelle de chaque phase d'équipement est ainsi enrichie de la gestion de l'interface PLI. La mise en œuvre des échanges nécessite la configuration et l'utilisation d'un serveur OPC entre l'application de MES et chaque automate du niveau contrôle/commande. Au terme de cette dernière

étape de l'intégration et suite à la réception d'un OF issu du niveau 4 *Enterprise Resource Planning* (ERP), l'application de MES est à même de faire exécuter et de contrôler avec succès, selon telle recette, une gamme d'assemblage sur la CPF.

6 Conclusion et Perspectives

Apporter une réponse industrielle à une demande de produits personnalisés nécessite la mise en place de systèmes de production de plus en plus flexibles, réactifs et adaptatifs. Cette nécessité conduit à développer des phases de contrôle/commande des équipements physiques de production élémentaires, fonctionnelles et indépendantes. Selon cette approche, l'ordonnancement des phases utiles à la réalisation d'une gamme de production donnée est rejeté à un système supérieur indépendant, celui de l'exécution de la production ou « *Manufacturing Execution System* » (MES).

Dans cet article, après avoir présenté les fonctionnalités du MES, nous illustrons, pour la fonctionnalité définition, exécution et contrôle de la production, les applications de MES et de contrôle/commande développées pour une CPF. La définition des procédures de production, côté MES, passe par l'instanciation de phases d'équipement prescriptives préalablement modélisées et associées aux modèles d'équipement. Pour sa part, l'exécution de la production, côté automate, est assurée par la programmation fonctionnelle de phases d'équipement élémentaires et exécutives à l'aide du langage FBD. Enfin, le contrôle de l'exécution est partagé entre les phases d'équipement prescriptives et exécutives à l'aide de l'interface PLI. Cette interface est également définie puis mise en œuvre pour un exemple de phase d'équipement.

Pour chacun de ces aspects, une approche à base de modèles d'équipement, de procédure et de commande a été suivie. Elle permet une réflexion conceptuelle sur ce que sont les éléments des applications et de constituer une bibliothèque de modèles. Ces résultats représentent une quantité de travail importante, côtés applications MES et automates, tant en conception et en programmation, qu'en essais et mises au point.

Au terme de cette première étape importante, nous poursuivons les développements selon deux perspectives majeures. La première consiste à remplacer le modèle d'exécution cyclique classique de la norme CEI/IEC 61131-3 par un modèle d'exécution piloté par un événement. Ce dernier fait l'objet de la norme CEI/IEC 61499 et permet une spécification explicite de l'ordre d'exécution de blocs fonctionnels. Pour sa part, un bloc fonctionnel spécifie une interface et une implémentation. Une interface contient des E/S d'événements et de données. La figure 10 présente le bloc fonctionnel de la phase d'équipement *Ph_ST_Shuttle_PS1* où les E/S d'événements reprennent les registres d'informations de l'interface PLI. L'environnement de développement *ExoStruxure Automation Expert*® (éditeur Schneider Electric), conforme à la norme CEI/IEC 61499, permet de développer cette nouvelle approche.

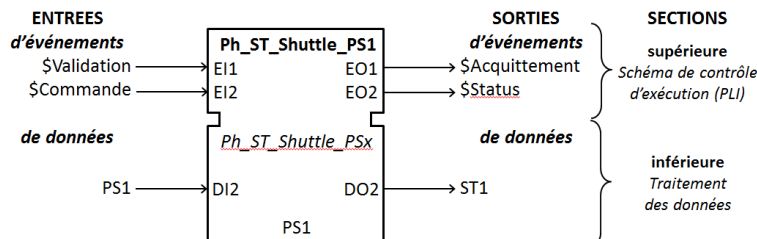


Figure 10 : Bloc fonctionnel CEI/IEC 61499 de la phase d'équipement *Ph_ST_Shuttle_PS1*

Une application, de contrôle/commande par exemple, est alors définie par un réseau de blocs

fonctionnels interconnectés. Elle est ensuite distribuée sur les équipements de production concernés. La norme CEI/IEC 61499 permet de développer l'architecture de pilotage des CPPS (*Cyber-Physical Production System*) selon un modèle d'exécution, en plein essor, novateur, objet, événementiel, distribué et interopérable (Lyu *et al.*, 2020). Il est indispensable aux applications portables. Il donne naissance à une approche « *plug and produce* » de l'automatisation où des composants logiciels sont connectés pour constituer les applications de pilotage.

Enfin, la seconde voie de développement traite de l'évaluation de méthodes de commande, de reconfiguration et d'accommodation adaptatives (Caulier et Békrar, 2017). L'objectif est ici d'adapter le pilotage de la CPF aux contraintes (1) de production, (2) de disponibilité des unités de traitement (évolutions alarmantes, défaillances et/ou perturbations) et (3) de maîtrise des diverses énergies.

Références

- Aip (2021). <https://www.uphf.fr/aipnpsc/>
- Caulier, P. (2021). Intégration de fonctions MES à l'architecture de pilotage d'une cellule de production flexible. *17^{ème} Colloque National S-mart*, 31 mars-2 avril, Valenciennes, France.
- Caulier, P. & Békrar, A. (2017). Pilotage adaptatif et durable d'une cellule de production de biens "intelligents". *15^{ème} Colloque National du GIS AIP-PRIMECA*, 12-14 avril, La Plagne, France.
- Caulier, P. & Vanderhaegen, F. (2011). Analyse de risques et résilience des systèmes sociotechniques en situation de crise. *4^{ème} Workshop du GIS 3SGS*, Octobre, Valenciennes, France.
- Delépine, O., Caulier, P. & Vanderhaegen, F. (2008). A method to improve human prognosis in supervision of complex systems. *Proceedings of the 17th IFAC World Congress*, 6-11 July, Seoul, Korea.
- Iec (2021). <https://www.iec.ch/homepage>
- Isa (2021). <https://www.isa.org/>
- Jaskó, S., Skrop, A., Holczinger, T., Chován, T. & Abonyi, J. (2020). Development of manufacturing execution systems in accordance with industry 4.0 requirements: a review of standard and ontology based methodologies and tools. *Computers in industry*, Volume 123.
- Lyu, G., Fazlirad, A. & Brennan, R.W. (2020). Multi-agent modeling of cyber-physical systems for IEC 61499 based distributed automation. *Procedia Manufacturing*, Volume 51, pp. 1200-1206.
- Monostori, L. (2014). Cyber-physical Production Systems: Roots, Expectations and R&D Challenges. *Proceedings of CIRP*, Volume 17, pp. 9-13.
- Ordinal (2021). <https://www.ordinal.fr/>
- Pcvue (2021). <https://www.pcvuesolutions.com/france/>
- Qi, Y., Mao, Z., Zhang, M. & Guo, H. (2020). Manufacturing practices and servitization: the role of mass customization and product innovation capabilities. *International Journal of Production Economics*, Volume 228.
- Vanderhaegen, F. & Caulier, P. (2011). A multi-viewpoint system to support abductive reasoning. *Information Sciences*, Vol. 181 (24), pp. 5349-5363, December.
- Vanderhaegen, F. & Caulier, P. (2013). Improving proactive human behavior in supervision of manufacturing systems using chronicles. *Proceedings of the 12th IFAC/IFIP/IFORS/IEA Symposium on Analysis, Design and Evaluation of Human-Machine Systems*, Vol. 46 (15), pp. 82-89, August, Las Vegas, Nevada, USA.
- Wcislik, M., Suchenia, K. & Laskawski, M. (2015). Programming of sequential control systems using functional block diagram language. *IFAC Papers on line*, Volume 48, Issue 4, pp. 330-335.