(19) RÉPUBLIQUE FRANÇAISE

INSTITUT NATIONAL
DE LA PROPRIÉTÉ INDUSTRIELLE

COURBEVOIE

(11) **N° de publication :** ***3 131 647***

(à n'utiliser que pour les commandes de reproduction)

(21) **N° d'enregistrement national :** **21 14743**

(51) Int Cl[8] : ***G 06 F 21/55*** *(2022.01), G 06 F 11/07, G 06 F 11/30, G 06 F 11/34, G 06 F 9/30, G 05 B 17/02, G 05 B 23/02*

(12) **BREVET D'INVENTION** **B1**

(54) Procédé de détection d'une anomalie dans un système électronique en fonctionnement.

(22) **Date de dépôt :** 31.12.21.

(30) **Priorité :**

(43) **Date de mise à la disposition du public de la demande :** 07.07.23 Bulletin 23/27.

(45) **Date de la mise à disposition du public du brevet d'invention :** 26.01.24 Bulletin 24/04.

(56) **Liste des documents cités dans le rapport de recherche :**

Se reporter à la fin du présent fascicule

(60) **Références à d'autres documents nationaux apparentés :**

(○) **Demande(s) d'extension :**

(71) **Demandeur(s) :** *THALES Société anonyme* — FR.

(72) **Inventeur(s) :** GIRBAL Sylvain, LE RHUN Jimmy Alain Daniel, FAURA David José et GRACIA PÉREZ Daniel.

(73) **Titulaire(s) :** THALES Société anonyme.

(74) **Mandataire(s) :** Lavoix.

FR 3 131 647 - B1

Description

Titre de l'invention : Procédé de détection d'une anomalie dans un système électronique en fonctionnement

[0001] La présente invention porte sur un procédé de détection d'une anomalie dans un système électronique en fonctionnement, en particulier une attaque informatique contre le système électronique.

[0002] L'invention s'applique en particulier au domaine des calculateurs embarqués critiques tels que ceux présents dans l'avionique, le spatial, le ferroviaire, l'automobile, le maritime, le médical ou l'IIoT (« Industrial Internet of Things » en anglais ou « Internet industriel des objets » en français), traditionnellement confrontés à des problématiques de sûreté de fonctionnement, dont le rôle est de protéger ces systèmes et leurs utilisateurs des défaillances et des pannes.

[0003] De nombreux standards de certification liés à ces industries tels que la norme DO178C/DO254 pour le domaine avionique, la norme IEC61508 pour le secteur industriel, la norme EN5012x pour le secteur ferroviaire, la norme ECSSx pour le domaine spatial permettent d'obtenir les garanties de sûreté nécessaires sur la fiabilité et la disponibilité de ces systèmes. Cependant les systèmes deviennent de plus en plus connectés, ce qui introduit de nouvelles problématiques liées à la cyber-sécurité et à la résistance aux attaques malveillantes.

[0004] A titre d'exemple, dans le domaine avionique, il est prévu la possibilité que le contrôle aérien, qui a une meilleure vue d'ensemble, puisse piloter les avions depuis le sol lors des phases d'approche des aéroports, introduisant de nouvelles problématiques d'authentification, de confidentialité et d'intégrité.

[0005] L'IIoT reprend les principes de l'internet des objets (ou « IoT » pour « Internet of Things » en anglais) pour les appliquer au secteur industriel, partageant les capteurs de données des différents systèmes, et maximisant les communications machines-machines sans intervention humaine. L'IIoT favorise la communication constante entre les différents systèmes pour augmenter l'efficacité opérationnelle, mais il est confronté à la nécessité d'une part de sécuriser ces communications, et d'autre part de s'assurer de la sécurité de chacun des systèmes le constituant, afin qu'il ne puisse pas servir de porte dérobée pour entrer dans le système de manière malveillante.

[0006] Le domaine des ordinateurs portatifs (« wearable computing » en anglais) et des systèmes médicaux implantés, en plus des problèmes de sécurité classiques, est également confronté à des problématiques de confidentialité et d'intégrité des données.

[0007] En outre, les systèmes sécurisés sont traditionnellement mis à jour dès qu'une faille de sécurité est découverte et corrigée. Cependant, les systèmes à haut niveau de sûreté

de fonctionnement sont rarement mis à jour, car ceci implique généralement de certifier à nouveau tout le système ce qui peut avoir de très lourds coûts financiers. Une sécurisation efficace des systèmes embarqués critiques ne peut donc pas uniquement recourir à des défenses ad-hoc à chaque type d'attaque nécessitant chacune une mise à jour, mais doit également permettre la détection proactive de nouvelles attaques malveillantes.

[0008] On connait en cyber-sécurité des systèmes de détection d'intrusion (IDS pour « Intrusion Detection System » en anglais). Ces systèmes sont classés généralement en deux catégories, à savoir les NIDS (pour « Network Intrusion Detection System » ou « Système de détection d'intrusion réseau » en français) et les HIDS (pour « Host Intrusion Detection System » ou « Système de détection d'intrusion de l'hôte » en français). Les NIDS se focalisent sur l'analyse des communications réseau. Il s'agit de capturer le trafic en un point particulier d'un réseau, et d'analyser les paquets en les comparant à une base de données de motifs d'attaque. Si un ensemble de paquets correspond à un motif d'attaque connu, une alerte est levée. Les HIDS se focalisent sur un ordinateur, au niveau de son système d'exploitation. Il s'agit de détecter des comportements anormaux, en surveillant les processus en cours d'exécution, les allocations de mémoire, les utilisateurs connectés, etc. Une alerte est générée quand une des grandeurs surveillées s'éloigne d'une norme prédéfinie.

[0009] Tous ces systèmes se focalisent sur des problèmes de sécurité soit au niveau logiciel, soit au niveau des communications. Toutefois, les attaques les plus récentes ont commencé à exploiter des vulnérabilités matérielles au niveau du processeur, qui sont bien plus complexes et coûteuses à corriger.

[0010] Par exemple, l'attaque Spectre exploite une vulnérabilité matérielle de certaines implémentations de la prédiction de branchement de microprocesseurs dotés d'exécution spéculative. Alors qu'au niveau fonctionnel, une erreur de prédiction est annulée dès qu'elle est constatée, l'exécution spéculative peut cependant, comme effet secondaire, permettre d'accéder arbitrairement à des emplacements de la mémoire vive, les données utilisées par le code ayant été exécuté "au cas où" restant stockées dans le cache.

[0011] L'attaque Meltdown exploite également une vulnérabilité liée à la spéculation utilisée pour l'exécution dans le désordre des instructions dans le processeur, incluant les mémoires caches et le TLB (de l'anglais « Translation Lookaside Buffer » ou « mémoire tampon d'anticipation de traduction » en français). Contrairement à l'attaque Spectre, l'attaque Meltdown utilise des mécanismes plus spécifiques à l'architecture du processeur, et permet d'une part d'accéder à des données sans les droits d'accès correspondants, mais également de faire de l'escalade de privilèges.

[0012] L'attaque Rowhammer exploite un effet secondaire imprévu des mémoires dy-

namiques à accès aléatoire (DRAM pour « Dynamic Random Access Memory » en anglais) provoquant une fuite de charge électrique dans des cellules de mémoire adjacentes, ce qui permet de modifier le contenu mémorisé dans ces cellules voisines sans disposer du droit d'accès.

[0013] Ainsi, les méthodes connues de détection sont devenues insuffisantes devant cette nouvelle génération d'attaques informatiques.

[0014] Il existe donc un besoin pour un procédé capable de mieux détecter et de se prémunir contre des anomalies affectant le système électronique, en particulier des attaques informatiques.

[0015] A cet effet, il est proposé un procédé de détection d'une anomalie dans un système électronique en fonctionnement, en particulier une attaque informatique contre le système électronique, le système électronique comprenant un ou plusieurs processeurs composés chacun d'une pluralité de blocs matériels , le processeur étant adapté pour exécuter au moins une application par interactions de l'application avec lesdits blocs matériels, le procédé de détection comprenant au moins les étapes suivantes :

[0016] - mesure, lors du fonctionnement du système électronique, d'au moins un paramètre représentatif des interactions de l'application avec l'un des blocs matériels pour obtenir des mesures dudit paramètre représentatif,

[0017] - pour chaque mesure, comparaison de ladite mesure avec un ensemble de données de référence associé pour détecter d'éventuelles incohérences entre la mesure et l'ensemble de données de référence, ledit ensemble de données de référence étant représentatif du fonctionnement du système électronique sans anomalies, et

[0018] - émission d'une alerte selon un critère d'alerte portant sur la ou les incohérences détectées.

[0019] Suivant d'autres modes de réalisation, le procédé comprend une ou plusieurs des caractéristiques suivantes, prise(s) isolément ou suivant toutes les combinaisons techniquement possibles :

– le procédé de détection comprend, en outre, au moins une étape choisie parmi le groupe constitué de :

[0020] - analyse de la nature du bloc matériel associé à chaque incohérence,

[0021] - analyse de la fréquence temporelle des incohérences, et

[0022] - analyse de l'écart de chaque mesure associée aux incohérences à l'ensemble de données de référence ;

[0023] le procédé de détection comprenant une étape de détection, en fonction de la ou chaque analyse d'une anomalie, en particulier une attaque informatique ou une défaillance matérielle de l'un des blocs matériels.

– lorsqu'au moins une alerte associée à l'un des blocs matériels est émise, une nouvelle itération des étapes du procédé de détection à partir de l'étape de

mesure est mise en œuvre, l'ensemble des blocs matériels dont une mesure de paramètre représentatif est effectuée étant, à chaque itération, inclus dans l'ensemble des blocs matériels de la précédente itération, l'ensemble des blocs matériels étant notamment inclus dans l'ensemble des blocs matériels associés à la ou les alertes.

– chaque critère d'alerte dépend d'au moins une condition sur les paramètres représentatifs, le procédé de détection comprenant, lorsqu'une alerte associée à l'un des blocs matériels est émise, une nouvelle itération des étapes du procédé de détection à partir de l'étape de mesure, le critère d'alerte comprenant, à chaque itération, un nombre croissant de conditions.

– chaque paramètre représentatif associé à l'un des blocs matériel est choisi parmi le groupe constitué de :

[0024] - le type de données traitées par le bloc matériel;

[0025] - le type d'instructions exécutées par le bloc matériel;

[0026] - le nombre d'accès au bloc matériel;

[0027] - le nombre de calcul exécutés et/ou le temps de calcul du bloc matériel;

[0028] - le nombre de dysfonctionnements du bloc matériel, et

[0029] - le nombre d'interactions sur un réseau d'interconnexion interne au processeur impliquant le bloc matériel, et

[0030] - le nombre d'interactions avec l'extérieur du processeur par le bloc matériel.

– chaque bloc matériel est choisi parmi le groupe constitué de :

[0031] - une ou plusieurs unités de calcul,

[0032] - une ou plusieurs unités de prédiction de branchement,

[0033] - un ou plusieurs registres de mémoire interne du processeur,

[0034] - une ou plusieurs mémoires caches,

[0035] - une ou plusieurs mémoires vives,

[0036] - une ou plusieurs chaines de traitement,

[0037] - une ou plusieurs unités de protection mémoire ou de traduction d'adresse,

[0038] - un ou plusieurs bus ou réseaux d'interconnexion,

[0039] - une ou plusieurs unités d'entrée-sortie

[0040] chaque paramètre représentatif du fonctionnement de l'un des blocs matériels étant choisi parmi le groupe constitué de :

[0041] - le nombre de données d'un même type traitées par la ou les unités de calcul, le ou les réseaux d'interconnexion ou la ou les mémoires,

[0042] - le nombre d'instructions d'un même type exécutées par la ou les unités de calcul,

[0043] - la consommation d'énergie de la ou des unités de calcul,

[0044] - la température de la ou des unités de calcul,

[0045] - le nombre de réussites et/ou d'erreurs de prédiction de l'unité de prédiction de

branchement,

[0046] - le nombre d'accès aux mémoires,

[0047] - le nombre de réussites et/ou d'erreurs de sollicitations de la mémoire cache,

[0048] - le nombre de réussites et/ou d'erreurs de sollicitations de l'unité de protection mémoire,

[0049] - le temps d'exécution d'une instruction dans la chaine de traitement,

[0050] - le nombre d'échanges d'information via le ou les réseaux d'interconnexion interne(s) au processeur,

[0051] - le nombre d'échanges d'information via le ou les réseaux d'interconnexion obtenus par une source donnée,

[0052] - le nombre d'échanges d'information via le ou les réseaux d'interconnexion envoyés à une destination donnée, et

[0053] - le nombre d'échanges d'informations via l'unité d'entrée-sortie.

– le système électronique comprend une pluralité de blocs matériels de comptage configurés chacun pour mesurer l'un des paramètres représentatifs, l'étape de mesure comprenant la mesure d'une pluralité de paramètres représentatifs, notamment entre quatre et six, par exemple les paramètres représentatifs mesurés par une partie des blocs matériels de comptage.

– le procédé de détection comprend, en outre, une étape de :

[0054] - comparaison de la ou de chaque incohérence associée à l'alerte émise avec un ensemble d'anomalies connues prédéterminées, et

[0055] - catégorisation de l'alerte lorsque la ou chaque incohérence correspond à l'une des anomalies connues dudit ensemble d'anomalies.

– le système électronique est un calculateur embarqué dans une plateforme de transport.

– le procédé de détection est mis en œuvre par un système choisi dans le groupe constitué de :

  – un circuit logique programmable dédié formant l'un desdits blocs matériels du processeur ;

  – un logiciel d'un système d'exploitation du système électronique configuré pour interagir avec une mémoire protégée par une unité de protection mémoire, et

  – une application propre à être mise en œuvre par le système électronique par adjonction d'un pilote contrôlant les blocs matériels.

[0056] Il est aussi proposé un produit programme d'ordinateur comportant un support lisible d'informations, sur lequel est mémorisé un programme d'ordinateur comprenant des instructions de programme, le programme d'ordinateur étant chargeable sur une unité de traitement de données et adapté pour entraîner la mise en œuvre d'un procédé de

détection tel que défini ci-dessus lorsque le programme d'ordinateur est mis en œuvre sur l'unité de traitement des données.

[0057] L'invention concerne également un support lisible d'informations sur lequel est mémorisé un produit programme d'ordinateur tel que défini ci-dessus.

[0058] Il est proposé en outre un calculateur embarqué dans une plateforme de transport configuré pour mettre en œuvre le procédé de détection tel que défini ci-dessus, le système électronique étant le calculateur embarqué.

[0059] L'invention concerne également une plateforme de transport comprenant le calculateur tel que défini ci-dessus.

[0060] L'invention concerne également un procédé de génération d'un jeu de référence, le jeu de référence étant représentatif du fonctionnement d'un système électronique en fonctionnement sans anomalies,

[0061] le système électronique comprenant un ou plusieurs processeurs composés chacun d'une pluralité de blocs matériels, les blocs matériels comprenant des blocs matériels d'exécution et des blocs matériels de comptage, le processeur étant adapté pour exécuter au moins une application par interactions de l'application avec lesdits blocs matériels d'exécution, chaque bloc matériel de comptage étant adapté pour mesurer un paramètre de fonctionnement représentatif des interactions de l'application avec l'un des blocs matériels d'exécution pour obtenir des mesures dudit paramètre représentatif,

[0062] le procédé de génération comprenant au moins les étapes suivantes :

[0063] - exécution de l'application sur un banc de test en fonction d'une pluralité de données d'entrée prédéterminées, lesdites données d'entrée étant représentatives du fonctionnement en exploitation du système électronique sans anomalies,

[0064] - mesures par au moins l'un des blocs matériels de comptage d'au moins un paramètre de fonctionnement représentatif des interactions de l'un des blocs matériels d'exécution lors de l'étape d'exécution avec l'application, pour obtenir des mesures,

[0065] - traitement des mesures de l'au moins un paramètre de fonctionnement, pour obtenir des données traitées, et

[0066] - génération du jeu de référence à partir des données traitées.

[0067] Suivant d'autres modes de réalisation, le procédé comprend une ou plusieurs des caractéristiques suivantes, prise(s) isolément ou suivant toutes les combinaisons techniquement possibles :

- lors de l'étape de mesures, les mesures obtenues sont des mesures d'une pluralité de paramètres de fonctionnement, notamment d'un nombre de paramètres de fonctionnement compris entre quatre et six, l'ensemble des mesures obtenues étant, par exemple, uniquement les paramètres de fonctionnement mesurés par une partie des blocs matériels de comptage.
- au moins une nouvelle itération des étapes du procédé de génération est mise

en œuvre, l'étape de mesures d'une itération étant mise en œuvre par au moins un bloc matériel de comptage différent de l'au moins un bloc matériel de comptage mettant en œuvre l'étape de mesures d'une autre itération.

– chaque paramètre de fonctionnement représentatif des interactions de l'un des blocs matériel d'exécution avec l'application est choisi parmi le groupe constitué de :

[0068] - le type de données traitées par le bloc matériel considéré,

[0069] - le type d'instructions exécutées par le bloc matériel considéré;

[0070] - le nombre d'accès au bloc matériel considéré,

[0071] - le nombre de calcul exécutés et/ou le temps de calcul du bloc matériel considéré,

[0072] - le nombre de dysfonctionnements du bloc matériel considéré,

[0073] - le nombre d'interactions sur un réseau d'interconnexion interne au processeur impliquant le bloc matériel, et

[0074] - le nombre d'interactions avec l'extérieur du processeur par le bloc matériel d'exécution considéré.

– chaque bloc matériel d'exécution est choisi parmi le groupe constitué par :

[0075] - une ou plusieurs unités de calcul,

[0076] - une ou plusieurs unités de prédiction de branchement,

[0077] - un ou plusieurs registres de mémoire,

[0078] - une ou plusieurs mémoires caches,

[0079] - une ou plusieurs mémoires vives,

[0080] - une ou plusieurs chaînes de traitement,

[0081] - une ou plusieurs unités de protection mémoire ou de traduction d'adresse,

[0082] - un ou plusieurs bus ou réseaux d'interconnexion, et

[0083] - une ou plusieurs unités d'entrée-sortie,

[0084] chaque paramètre représentatif du fonctionnement de l'un des blocs matériels d'exécution étant choisi parmi le groupe constitué par :

[0085] - le nombre de données d'un même type traitées par la ou les unités de calcul, le ou les réseaux d'interconnexion ou la ou les mémoires,

[0086] - le nombre d'instructions d'un même type exécutées par l'unité de calcul,

[0087] - la consommation d'énergie de la ou des unités de calcul,

[0088] - la température de la ou des unités de calcul,

[0089] - le nombre de réussites et/ou d'erreurs de prédiction de l'unité de prédiction de branchement,

[0090] - le nombre d'accès aux mémoires,

[0091] - le nombre de réussites et/ou d'erreurs de sollicitations de la mémoire cache,

[0092] - le nombre de réussites et/ou d'erreurs de sollicitations de l'unité de protection mémoire,

[0093] - le temps d'exécution d'une instruction dans la chaîne de traitement,

[0094] - le nombre d'échanges d'informations sur le ou les réseaux d'interconnexions interne du processeur,

[0095] - le nombre d'échanges d'information via le ou les réseaux d'interconnexion par source,

[0096] - le nombre d'échanges d'information via le ou les réseaux d'interconnexion par destination, et

[0097] - le nombre d'échanges d'informations via l'unité d'entrée-sortie.

- l'exécution de l'application est mise en œuvre selon au moins deux phases temporelles successives, l'étape de traitement étant mise en œuvre par application d'opérations respectives sur les mesures de chaque phase
- l'application est configurée pour mettre en œuvre au moins deux fonctions, chaque fonction étant mise en œuvre durant une phase temporelle propre.
- chaque phase temporelle correspond à un intervalle de temps prédéterminé d'exécution de l'application, notamment un intervalle de temps inférieur à 300 millisecondes.
- l'étape de traitement comprend une agrégation des mesures pour chaque paramètre de fonctionnement pour obtenir une base de données propre à chaque paramètre de fonctionnement.
- l'ensemble de mesures et l'ensemble des données traitées occupent une taille respective sur une mémoire, l'étape de traitement comprenant l'application d'une opération permettant que la taille de l'ensemble des données traitées soit inférieure à la taille de l'ensemble de mesures.
- l'opération est mise en œuvre par une technique d'apprentissage automatique comprenant un entraînement à partir de l'ensemble de mesures.
- la technique d'apprentissage automatique comprend l'entraînement d'un réseau de neurones à partir de l'ensemble de mesures.
- l'opération est une opération d'analyse statistique telle que l'ensemble des données traitées est une distribution statistique de l'ensemble de mesures.
- un nombre prédéterminé d'itérations de l'étape de mesure est mis en œuvre afin d'obtenir un premier ensemble de mesures et un premier ensemble de données traitées, le procédé de génération comprenant, en outre, les étapes suivantes :
- deuxième mise œuvre du nombre prédéterminé d'itérations de l'étape de mesure pour obtenir un deuxième ensemble de mesures et un deuxième ensemble de données traitées,
- comparaison du premier ensemble de données traitées et du deuxième ensemble de données traitées en fonction d'un critère de convergence pré-

déterminé,

- lorsque le critère de convergence n'est pas respecté, mise en œuvre d'un nombre d'itérations de l'étape de mesure supérieur au nombre d'itérations prédéterminé, et
- lorsque le critère de convergence est respecté, génération du jeu de référence à partir des premier et deuxième ensembles de données traitées.

[0098] L'invention concerne également un jeu de référence susceptible d'être obtenu par un procédé de génération tel que défini ci-dessus.

[0099] L'invention concerne également l'utilisation du jeu de référence tel que défini ci-dessus pour détecter une anomalie dans le système électronique en fonctionnement, en particulier une attaque informatique contre le système électronique, par comparaison d'au moins une mesure d'au moins l'un des paramètres de fonctionnement lors du fonctionnement du système électronique et par comparaison de ladite mesure avec le jeu de référence pour détecter d'éventuelles incohérences entre la mesure et ledit jeu de référence.

[0100] L'invention concerne également un produit programme d'ordinateur comportant un support lisible d'informations, sur lequel est mémorisé un programme d'ordinateur comprenant des instructions de programme, le programme d'ordinateur étant chargeable sur une unité de traitement de données et adapté pour entraîner la mise en œuvre d'un procédé de génération tel que défini ci-dessus lorsque le programme d'ordinateur est mis en œuvre sur l'unité de traitement des données.

[0101] L'invention concerne également un support lisible d'informations sur lequel est mémorisé un programme d'ordinateur comprenant des instructions de programme, le programme d'ordinateur étant chargeable sur une unité de traitement de données et adapté pour entraîner la mise en œuvre d'un procédé de génération tel que défini ci-dessus lorsque le programme d'ordinateur est mis en œuvre sur l'unité de traitement des données.

[0102] L'invention sera mieux comprise à l'aide de la description qui va suivre, donnée uniquement à titre d'exemple non limitatif et faite en se référant aux dessins annexés, sur lesquels :

[0103] [Fig.1] la [Fig.1] est une représentation schématique d'un système électronique selon l'invention embarqué dans un système de transport,

[0104] [Fig.2] la [Fig.2] est un organigramme d'un procédé, selon l'invention, de génération d'un jeu de référence,

[0105] [Fig.3] la [Fig.3] est un organigramme de l'étape de découpage du procédé de génération,

[0106] [Fig.4] la [Fig.4] est un organigramme des étapes de mesures et de traitement du procédé de génération,

[0107] [Fig.5] la [Fig.5] est un organigramme d'un procédé, selon l'invention, de détection d'une anomalie,

[0108] [Fig.6] la [Fig.6] est une représentation schématique d'un processeur embarqué dans le système électronique de la [Fig.1].

[0109] La [Fig.1] représente un système électronique 10.

[0110] Le système électronique 10 est embarqué dans une plateforme de transport 12. La plateforme de transport 12 est notamment propre à transporter des passagers et/ou des marchandises.

[0111] La plateforme de transport 12 est, en particulier, une plateforme de transport avionique telle qu'un avion, un hélicoptère ou un drone.

[0112] En variante, la plateforme de transport 12 est une plateforme de transport spatiale comme par exemple un satellite ou une navette spatiale ; une plateforme de transport ferroviaire, notamment un train ; une plateforme de transport maritime telle qu'un navire ou un sous-marin ou une plateforme de transport automobile comme par exemple une voiture autonome ou un bus.

[0113] En variante encore, l'invention s'applique à tout domaine dans lequel sont présents des calculateurs embarqués critiques. Notamment, en variante, le système électronique 10 est embarqué dans un dispositif médical, dans un ordinateur portatif, dans un système domotique ou dans un système industriel.

[0114] La plateforme 12 comprend éventuellement une pluralité de systèmes électroniques 10. Par exemple, lorsque la plateforme de transport 12 est un aéronef, la plateforme comprend un système de décollage, un système de pilotage, un système de communication, un système d'alerte, etc.

[0115] Comme visible sur la [Fig.1], le système électronique 10 comprend un ou plusieurs processeurs 14, un système d'exploitation 16 et une mémoire 18.

[0116] Le système d'exploitation 16 (souvent appelé OS de l'anglais « Operating System ») est un ensemble de programmes qui dirige l'utilisation des ressources du système électronique 10 par des applications.

[0117] Chaque processeur 14 est composé d'une pluralité de blocs matériels.

[0118] Les blocs matériels comprennent des blocs matériels d'exécution 20 et des blocs matériels de comptage 22.

[0119] Un bloc matériel d'exécution 20 est un composant électronique configuré pour exécuter une fonction élémentaire du processeur 14 telle que le calcul, le stockage mémoire ou le transfert de données.

[0120] En particulier, le processeur 14 est adapté pour exécuter au moins une application par interactions de l'application avec les blocs matériels d'exécution 20.

[0121] L'application est un programme propre à réaliser une tâche, ou un ensemble de tâches élémentaires. L'application s'exécute en utilisant les services du système

d'exploitation pour utiliser les ressources matérielles fournies par les blocs matériels d'exécution 20.

[0122] En particulier, chaque bloc matériel d'exécution 20 est choisi parmi le groupe constitué de :

- une ou plusieurs unités de calcul,
- une ou plusieurs unités de prédiction de branchement,
- un ou plusieurs registres de mémoire interne du processeur 14,
- une ou plusieurs mémoires caches,
- une ou plusieurs mémoires vives,
- une ou plusieurs chaînes de traitement,
- une ou plusieurs unités de protection mémoire ou de traduction d'adresse,
- un ou plusieurs bus ou réseaux d'interconnexion, et
- une ou plusieurs unités d'entrée-sortie.

[0123] Ces différents blocs matériels d'exécution 20 permettent de fournir les ressources matérielles nécessaires à l'exécution des différentes applications.

[0124] Chaque bloc matériel de comptage 22 est adapté pour mesurer un paramètre de fonctionnement représentatif des interactions de l'application avec l'un des blocs matériels d'exécution 20 avec l'application pour obtenir des mesures dudit paramètre représentatif.

[0125] Avantageusement, le système électronique 10 comprend une pluralité de blocs matériels de comptage 22 propres à mesurer chacun un paramètre de fonctionnement propre.

[0126] Le système électronique 10 comprend notamment entre quatre et six blocs matériels de comptage 22.

[0127] En variante, le système électronique 10 comprend plus de six blocs matériels de comptage 22.

[0128] Chaque paramètre représentatif associé à l'un des blocs matériel d'exécution 20 est choisi parmi le groupe constitué de :

- le type de données traitées par le bloc matériel d'exécution 20 ;
- le type d'instructions exécutées par le bloc matériel d'exécution 20;
- le nombre d'accès au bloc matériel d'exécution 20;
- le nombre de calcul exécutés et/ou le temps de calcul du bloc matériel d'exécution 20;
- le nombre de dysfonctionnements du bloc matériel d'exécution 20, et
- le nombre d'interactions sur un réseau d'interconnexion interne au processeur impliquant le bloc matériel d'exécution 20, et
- le nombre d'interactions avec l'extérieur du processeur 14 par le bloc matériel d'exécution 20.

[0129] Les différents types de données sont par exemple un type booléen, un type entier, un type réel à virgule flottante, un type caractère, etc.

[0130] Les différentes instructions sont par exemple une instruction de calcul, une instruction d'envoi de données, une instruction de mise en mémoire, etc.

[0131] Chaque réseau d'interconnexion interne du processeur 14 forme une passerelle permettant la communication entre différentes entités du processeur, notamment entre les différents blocs matériels.

[0132] Plus spécifiquement, chaque paramètre représentatif du fonctionnement de l'un des blocs matériels d'exécution 20 est choisi parmi le groupe constitué de :

- le nombre de données d'un même type traitées par la ou les unités de calcul, le ou les réseaux d'interconnexion ou la ou les mémoires,
- le nombre d'instructions d'un même type exécutées par la ou les unités de calcul,
- la consommation d'énergie de la ou des unités de calcul,
- la température de la ou des unités de calcul,
- le nombre de réussites et/ou d'erreurs de prédiction de l'unité de prédiction de branchement,
- le nombre d'accès aux mémoires,
- le nombre de réussites et/ou d'erreurs de sollicitations de la mémoire cache,
- le nombre de réussites et/ou d'erreurs de sollicitations de l'unité de protection mémoire,
- le temps d'exécution d'une instruction dans la chaîne de traitement,
- le nombre d'échanges d'information via le ou les réseaux d'interconnexion interne(s) au processeur 14,
- le nombre d'échanges d'information via le ou les réseaux d'interconnexion interne(s) obtenus par une source donnée,
- le nombre d'échanges d'information via le ou les réseaux d'interconnexion interne(s) envoyés à une destination donnée, et
- le nombre d'échanges d'informations via l'unité d'entrée-sortie.

[0133] La mémoire 18 stocke un ensemble de données de référence, ou aussi appelé jeu de référence par la suite.

[0134] L'ensemble de données de référence est représentatif du fonctionnement du système électronique 10 sans anomalies.

[0135] On entend par « fonctionnement sans anomalies », un fonctionnement du système électronique 10 tel que prévu et envisagé lors de sa conception et de son exploitation. Un fonctionnement anormal est donc un fonctionnement s'écartant de ce qui était envisagé durant l'exploitation du système électronique 10, dû par exemple à une attaque informatique ou à une défaillance matérielle et/ou logicielle. En particulier, il

est défini une marge de fonctionnement sans anomalie autour des valeurs de fonctionnement prévues lors de la conception du système. A titre d'exemple, un intervalle de température de fonctionnement sans anomalies des unités de calcul est déterminé à la conception. Un fonctionnement anormal est détecté quand la température est par exemple 10°C plus chaud que la borne maximale de l'intervalle de fonctionnement sans anomalies.

[0136] Comme cela sera expliqué par la suite, l'ensemble de données de référence, ou jeu de référence, permet de détecter d'éventuelles incohérences entre la mesure d'un paramètre de fonctionnement et l'ensemble de données de référence.

[0137] Un procédé de génération 100 d'un tel jeu de référence va maintenant être décrit, en référence à la [Fig.2] représentant un organigramme dudit procédé de génération 100.

[0138] Le procédé de génération 100 comprend une étape d'exécution 120 de l'application A sur un banc de test en fonction d'une pluralité de données d'entrée E prédéterminées.

[0139] On entend par banc de test, un environnement d'essais permettant de mettre le système électronique 10 en conditions d'utilisation paramétrables et contrôlées afin d'observer et mesurer son comportement. L'exécution sur banc de test est donc différente de l'utilisation opérationnelle en exploitation par la suite du système électronique 10.

[0140] Les données d'entrée E sont représentatives du fonctionnement en exploitation du système électronique 10 sans anomalies.

[0141] Ces données d'entrées E sont déterminées en fonction de l'utilisation envisagée pour le système électronique 10 en exploitation. Le retour d'expérience issu de l'exploitation de systèmes électroniques semblables est également utilisé. Ces données d'entrées sont choisies afin de couvrir une grande variété d'évènements susceptibles d'être rencontrés par le système électronique 10 lors de son exploitation en fonctionnement normal, sans anomalies.

[0142] L'application A est configurée pour mettre en œuvre au moins deux fonctions, chaque fonction étant mise en œuvre durant une phase P temporelle propre.

[0143] A titre d'exemple, l'application A comprend une phase initiale d'acquisition des données, une phase de traitement des données, une phase de mise en mémoire et une phase d'envoi des données.

[0144] Ainsi, l'exécution de l'application A est mise en œuvre selon au moins deux phases P temporelles successives.

[0145] Le procédé de génération 100 comprend donc une étape préalable de découpage 110 en phases P de l'application A.

[0146] En effet, comme cela sera expliqué plus en détail par la suite, un traitement global de l'ensemble d'une application dont le comportement varierait significativement au cours

de l'exécution de l'application risquerait d'entraîner la création d'un jeu de référence insuffisamment sélectif, ce qui rendrait difficile la détection de comportements anormaux.

[0147] Selon le type d'application, le découpage est un découpage temporel ou un découpage fonctionnel.

[0148] En particulier, la [Fig.3] présente un arbre de décision permettant de choisir le type de découpage de l'application A.

[0149] Ainsi, le découpage temporel est choisi lorsque le code source de l'application A n'est pas connu ou maitrisé et que l'identification des phases n'est pas aisée.

[0150] A l'inverse, quand le code source est maîtrisé et que le comportement de l'application A n'est pas dépendant des données d'entrées, il est possible d'identifier des phases P de l'application A et le découpage fonctionnel sera privilégié.

[0151] En effet, la réalisation d'un découpage fonctionnel et une identification des phases P comportementales d'une application requiert une certaine maîtrise de l'application, ce qui n'est pas toujours le cas lorsque l'on intègre des applications tierces. Dans ce contexte « boîte-noire », des solutions de profilage classique (comme par exemple en utilisant le logiciel connu Gprof) ou d'analyse statique de code peuvent aider à identifier les phases applicatives, et permettre un tel découpage fonctionnel. Ce profilage recherche notamment les boucles régulières, c'est-à-dire des ensembles d'instructions dont les conditionnelles et les bornes de boucles peuvent s'exprimer sous la forme de fonctions affines des itérateurs englobants.

[0152] Toutefois, si le comportement de l'application A dépend fortement des données, comme cela peut être le cas par exemple pour des applications de cryptographie, un découpage fonctionnel comportemental n'est pas possible, car non répétable d'une exécution à l'autre.

[0153] Dans ce cas, le découpage est un découpage temporel qui requiert une étape supplémentaire de sélection de la fréquence d'échantillonnage.

[0154] Ce choix a un impact sur le nombre total de phases P et est déterminé en cherchant un compromis entre la finesse de la caractérisation de ces phases et la taille du jeu de référence obtenu.

[0155] Par exemple, dans le cadre des applications embarquées périodiques, le découpage est effectué sur la base de cette période.

[0156] Toutefois, si l'activation d'une tâche est elle-même composée de phases comme par exemple une phase de récupération des données, une phase de calcul et une phase de fourniture du résultat, la fréquence d'échantillonnage est plus importante.

[0157] En particulier, chaque phase P temporelle correspond à un intervalle de temps prédéterminé d'exécution de l'application, notamment un intervalle de temps inférieur à 300 millisecondes, typiquement 200 millisecondes.

[0158] Le procédé de génération 100 comprend une étape de mesures 130 par au moins l'un des blocs matériels de comptage 22 d'au moins un paramètre de fonctionnement représentatif des interactions de l'un des blocs matériels d'exécution 20 lors de l'étape d'exécution avec l'application, pour obtenir des mesures. Les mesures obtenues sont donc des mesures d'une pluralité de paramètres de fonctionnement tels que définis ci-dessus.

[0159] La [Fig.4] illustre notamment cette étape de mesures 130.

[0160] Ces mesures sont réalisées phase par phase afin d'obtenir un ensemble de mesures des paramètres de fonctionnement par phase.

[0161] Au moins une nouvelle itération I des étapes du procédé de génération 100 est mise en œuvre, l'étape de mesures 130 d'une itération étant mise en œuvre par au moins un bloc matériel de comptage 22 différent de l'au moins un bloc matériel de comptage 22 mettant en œuvre l'étape de mesures d'une autre itération I.

[0162] Un ensemble de blocs matériels de comptage 22 mesurant les paramètres de fonctionnement lors d'une itération I est appelé configuration C des blocs matériels de comptage 22.

[0163] Comme expliqué ci-dessus, il n'est pas possible de déterminer *a priori* quels événements matériels seront visés et donc sollicités par les éventuelles futures attaques informatiques, et ne disposant que d'un nombre de blocs matériels de comptage 22 réduit par rapport au grand nombre d'événements, il est donc intéressant de parcourir équitablement l'ensemble des événements matériels et de varier les configurations des blocs matériels de comptage 22.

[0164] De manière à collecter des données statistiquement significatives, l'application A est exécutée sur un nombre important d'itérations I, avec des données d'entrées différentes, pour capturer des distributions des valeurs possibles des différents paramètres de fonctionnement en fonctionnement nominal.

[0165] Le procédé de génération 100 comprend alors une étape de traitement 140 des mesures des paramètres de fonctionnement, pour obtenir des données traitées.

[0166] En particulier, l'étape de traitement 140 est mise en œuvre par application d'opérations respectives sur les mesures de chaque phase.

[0167] L'étape de traitement 140 comprend notamment une agrégation des mesures pour chaque paramètre de fonctionnement pour obtenir une base de données propre à chaque paramètre de fonctionnement.

[0168] En particulier, le traitement 140 comprend une agrégation des mesures par phase P, des mesures par configuration C, et des mesures par itération I pour obtenir un ensemble des données statistiques collectées. À la fin de chacune de ces étapes, les valeurs collectées sont agrégées comme suit.

[0169] Lors d'une phase, l'ensemble des couples (évènement, valeur) sont collectés tel que

l'évènement appartienne à la configuration de blocs matériels de comptage 22 courante.

[0170] L'agrégation par phase P consiste à regrouper dans un tableau indexé par phase l'ensemble des valeurs mesurées.

[0171] L'agrégation par configuration C étend l'ensemble des évènements au-delà de la configuration courante pour couvrir l'ensemble des configurations nécessaire pour couvrir tous les évènements matériels envisagés en fonctionnement normal.

[0172] L'agrégation par itération I diffère de l'agrégation précédente en remplaçant les couples (évènement, valeur) par des couples (évènement, ensemble des valeurs observées au cours des différentes itérations).

[0173] L'ensemble des mesures collectées pour l'application A est susceptible d'être ainsi très important et représenter typiquement plusieurs giga-octets de données collectées.

[0174] Il est alors utile de faire une réduction de ces données pour obtenir un jeu de référence J de taille réduite, intégrable dans la mémoire 18 du système électronique 10.

[0175] L'étape de traitement 140 comprend alors l'application d'une opération permettant que la taille en mémoire de l'ensemble des données traitées soit inférieure à la taille en mémoire de l'ensemble de mesures.

[0176] En particulier, l'opération est mise en œuvre par une technique d'apprentissage automatique comprenant un entraînement à partir de l'ensemble de mesures.

[0177] La technique d'apprentissage automatique est notamment implémentée par l'entraînement d'un réseau de neurones à partir de l'ensemble de mesures. Le jeu de référence deviendrait alors ce réseau une fois entrainé. Des techniques d'apprentissage symbolique ou de segmentation sont utilisées en variante.

[0178] En variante, l'opération de réduction est une opération d'analyse statistique telle que l'ensemble des données traitées est une distribution statistique de l'ensemble de mesures.

[0179] Cette opération est similaire à celle couramment réalisée pour permettre la visualisation des données statistiques avec des histogrammes ou des boîtes à moustaches pas exemple.

[0180] La distribution statistique comprend une pluralité de valeurs donnant des informations clefs sur la distribution statistique telles que le minimum, le maximum, la moyenne, la médiane, l'écart-type, des déciles, des quartiles, etc. Par exemple, ces valeurs sont les différents intervalles (« bins » en anglais) d'un histogramme ou les quartiles pour les boîtes à moustaches.

[0181] Cette pluralité de valeurs devient alors le jeu de référence J.

[0182] Il est donc nécessaire de réduire la taille des données collectées mais il est également important de s'assurer que les données collectées sont statistiquement représentatives de l'application. Il est ainsi nécessaire d'évaluer la couverture statistique des données

collectées.

[0183] Ainsi, un nombre prédéterminé d'itérations de l'étape de mesure 130 est mis en œuvre afin d'obtenir un premier ensemble de mesures et un premier ensemble de données traitées.

[0184] Le procédé de génération 100 comprend alors en outre une deuxième mise œuvre du nombre prédéterminé d'itérations de l'étape de mesure 130 pour obtenir un deuxième ensemble de mesures et un deuxième ensemble de données traitées.

[0185] Puis, le premier ensemble de données traitées est comparé au deuxième ensemble de données traitées en fonction d'un critère de convergence prédéterminé.

[0186] Le critère de convergence permet de déterminer si ces deux distributions sont statistiquement similaires. La littérature propose différentes solutions à ce problème comme par exemple la comparaison d'histogrammes en calculant la distance de Bhattacharyya, la divergence de Kullback-Leiber ou la distance de Hellinger.

[0187] Lorsque le critère de convergence n'est pas respecté, le procédé de génération 100 comprend la mise en œuvre d'un nombre d'itérations de l'étape de mesure 130 supérieur au nombre d'itérations prédéterminé.

[0188] Autrement dit, les deux distributions statistiques collectées sont concaténées et on réitère la collection d'une distribution de taille supérieure, par exemple d'une taille double.

[0189] Cette étape est réitérée jusqu'à obtenir une convergence de la distribution statistique.

[0190] Lorsque le critère de convergence est respecté, le procédé de génération 100 comprend la génération du jeu de référence J à partir des premier et deuxième ensembles des données traitées.

[0191] On obtient ainsi un jeu de référence J à partir des données traitées. Comme cela sera expliqué par la suite, ce jeu de référence est notamment utilisé pour détecter une anomalie dans le système électronique 10 en fonctionnement, en particulier une attaque informatique contre le système électronique 10, par comparaison d'au moins une mesure d'au moins l'un des paramètres de fonctionnement lors du fonctionnement du système électronique 10 et par comparaison de ladite mesure avec le jeu de référence pour détecter d'éventuelles incohérences entre la mesure et le jeu de référence.

[0192] Toutefois, d'autres applications du jeu de référence J ainsi généré sont possibles.

[0193] En effet, dans les domaines critiques tels que l'automobile, le ferroviaire, l'avionique ou le spatial, la tendance actuelle des systèmes électronique/informatiques embarqués est de s'orienter vers l'utilisation de plateformes de calcul génériques assurant à la fois la fonctionnalité et les propriétés de sûreté de fonctionnement et de sécurité.

[0194] Ces plateformes de calcul possèdent un ensemble de ressources matérielles limitées et doivent exécuter un ou plusieurs logiciels applicatifs, de niveaux de criticité différents et provenant de fournisseurs différents.

[0195] La complexité des nouveaux systèmes embarqués pour les domaines critiques et le partage du marché entre les différents acteurs économiques a nécessité la modification du processus industriel en redéfinissant les rôles et responsabilités entre les différents intervenants lors de la conception d'un système. Chacun de ces acteurs est responsable de fournir les briques technologiques qui lui sont attribuées et d'appliquer les contraintes qui lui sont imposées par les exigences de fonctionnement du système.

[0196] Le fournisseur de plateforme matérielle livre le calculateur ainsi que toute l'électronique nécessaire au bon fonctionnement des applications logicielles qui seront susceptibles de lui être attribuées.

[0197] Le fournisseur du système d'exploitation fournit le système d'exploitation qui va permettre d'assurer la gestion de l'exécution des différents logiciels applicatifs en fonction de leurs besoins en terme de priorité, de temps et de fréquence d'exécution, d'espace mémoire, etc.

[0198] Le fournisseur de logiciels applicatifs est responsable du bon fonctionnement de ses applications tout en respectant les consignes fournies par l'intégrateur du système concernant les règles d'utilisation des ressources matérielles du système.

[0199] Enfin, l'intégrateur système se voit confier le rôle central. Il assemble les différentes briques technologiques, à la fois matérielles et logicielles du système tout en allouant l'ensemble des ressources nécessaires aux fournisseurs des applications logicielles. De plus, il est responsable de la consolidation des exigences de fonctionnement et de la prise en compte des interactions des différentes briques technologiques vis-à-vis des ressources du système. Il est celui qui doit garantir la sureté de fonctionnement et sécurité globale du système.

[0200] La maitrise du système implique tous les participants chacun à son niveau de responsabilité mais c'est à l'intégrateur système qu'incombe la définition et la validation de l'ensemble du système en cours de construction en s'assurant que les applications s'exécutant sur les plateformes de calcul du système respectent à tout instant les spécifications non fonctionnelles tels que le temps, l'énergie et le partage des ressources nécessaire au calcul, la sûreté, la sécurité, etc.

[0201] Dans ce contexte, le jeu de référence J est avantageusement utilisé pour transmettre les exigences de l'intégrateur système vers les autres acteurs et la transmission des propriétés non-fonctionnelles des applications vers l'intégrateur comme le besoin effectif en termes de ressources matérielles partagées.

[0202] En effet, en fournissant avec son application le jeu de référence correspondant, le fournisseur de logiciel fournit bien des informations liées à l'utilisation des ressources matérielles par son application, et donc un taux d'utilisation des différentes ressources partagées.

[0203] Le jeu de référence J est également un moyen pour l'intégrateur système de traduire

ses exigences sous forme d'une enveloppe maximale d'utilisation des ressources pour chaque application, de manière à garantir leur coexistence dans de système intégré final.

[0204] Le jeu de référence J représentant le comportement matériel des applications a donc plusieurs domaines d'applicabilité.

[0205] Le jeu de référence J peut être utilisé dans un cadre cyber-sécurité, comme cela sera expliqué par la suite, pour se prémunir de cyber-attaques externes.

[0206] Dans un cadre sûreté de fonctionnement, le jeu de référence J peut être également utilisé pour participer à la surveillance du système, détecter les cas anormaux liés aux défaillances et déclencher les mesures visant à un retour à un état stable.

[0207] Le jeu de référence J peut être également utilisé comme moyen de transmission des exigences non-fonctionnelles. En effet, dans le cadre des processeurs multi-cœurs, eux-mêmes des systèmes complexes provenant de fournisseurs tiers et possédant le plus souvent des spécifications partielles, la technique usuelle du partitionnement temporel est devenue insuffisante.

[0208] Le jeu de référence J peut être également utilisé pour faciliter l'intégration des applications en prenant en compte leur spécificité d'utilisations des ressources matérielles partagées. La technique usuelle se basant sur des modèles abstraits n'étant plus adaptée aux multi-cœurs.

[0209] Le jeu de référence J peut être également utilisé pour construire une bibliothèque d'applications représentatives d'un domaine en les caractérisant via la distance entre leurs jeux de référence respectifs.

[0210] Dans un cadre de certification incrémentale, le jeu de référence J peut être également utilisé pour anticiper l'impact de l'intégration d'une nouvelle application dans le système connaissant son jeu de référence.

[0211] Un procédé de détection 200 d'une anomalie dans un système électronique 10 en fonctionnement va maintenant être décrit en référence à la [Fig.5] représentant un organigramme dudit programme de détection 200.

[0212] Le procédé de détection 200 est réalisé en utilisant un jeu de référence tel que généré ci-dessus.

[0213] En particulier, le procédé 200 vise à détecter une attaque informatique contre le système électronique 10.

[0214] Toutefois, le procédé de détection 200 est également propre à détecter une défaillance matérielle et/ou logicielle du système électronique 10.

[0215] Plusieurs modes de réalisation sont possibles pour la mise en œuvre du procédé de détection, comme illustré sur la [Fig.6].

[0216] Selon un premier mode de réalisation, le procédé de détection est mis en œuvre par un circuit logique programmable dédié formant l'un desdits blocs matériels

d'exécution 20 du processeur 14.

[0217] Ainsi, cette mise en œuvre est réalisée au niveau matériel sous forme d'un composant dédié, comme représenté schématiquement sur la [Fig.6] avec la référence numérique 1. Ce composant pourrait ainsi reprogrammer et exploiter les blocs matériels de comptage 22 sans se soucier des niveaux de privilèges nécessaires au niveau logiciel. Un composant matériel permet également de disposer facilement d'une mémoire dédiée pour stocker les jeux de référence au sein même du composant.

[0218] Ce mode de réalisation offre donc une solution hautement sécurisée.

[0219] Selon un deuxième mode de réalisation, le procédé de détection est mis en œuvre par un logiciel d'un système d'exploitation 16 du système électronique 10 configuré pour interagir avec une mémoire protégée par une unité de protection mémoire.

[0220] Ce mode de réalisation est représenté schématiquement sur la [Fig.6] avec la référence numérique 2.

[0221] Dans ce cas, le jeu de référence est stocké au niveau de la mémoire principale. Le logiciel bénéficiera des niveaux de privilège élevés du système d'exploitation pour configurer et lire les blocs matériels de comptage 22. La zone de stockage des jeux de références est protégée via l'unité de gestion mémoire (ou MMU de l'anglais « memory management unit »).

[0222] Selon un troisième mode de réalisation, le procédé de détection est mis en œuvre par une application propre à être mise en œuvre par le système électronique 10 par adjonction d'un pilote contrôlant les blocs matériels, comme représenté schématiquement sur la [Fig.6] avec la référence numérique 3.

[0223] Ainsi, le procédé est mis en œuvre sous la forme d'une application tournant au même niveau que les applications à surveiller.

[0224] Ce mode de réalisation est le plus facile à implémenter.

[0225] En variante encore, le procédé de détection est mis en œuvre via une implémentation hybride, à la fois logicielle et matérielle, comme par exemple une réalisation au niveau logiciel qui stocke les jeux de référence dans une mémoire flash matérielle dédiée, pour protéger l'espace mémoire des jeux de références.

[0226] Dans l'état initial du procédé de détection 200, le système électronique 10 est en fonctionnement opérationnel.

[0227] Par exemple, le système électronique 10 est embarqué dans un aéronef en train de voler à destination un aéroport. Le système électronique 10 est alors par exemple un système avionique de communication, de pilotage, etc.

[0228] Le procédé de détection 200 comprend une première étape de mesure 210, lors du fonctionnement du système électronique 10, d'au moins un paramètre représentatif des interactions de l'application A avec l'un des blocs matériels d'exécution 20 pour obtenir des mesures dudit paramètre représentatif.

[0229] Les paramètres représentatifs sont tels que définis ci-dessus. Ils correspondent en particulier à ceux pour lesquels des mesures ont été effectuées lors du procédé de génération du jeu de référence associé.

[0230] Pour chaque mesure, le procédé de détection 200 comprend la comparaison 220 de la mesure avec l'ensemble de données de référence associé, ou jeu de référence, pour détecter d'éventuelles incohérences entre la mesure et l'ensemble de données de référence.

[0231] Une incohérence est un écart statistique de la mesure par rapport au jeu de référence. Par exemple, la mesure n'est pas comprise dans l'enveloppe formée par le jeu de référence.

[0232] Tout écart indique un comportement anormal de l'application qui pourrait être lié à une cyber-attaque.

[0233] Ainsi, le procédé de détection 200 comprend une étape d'émission 230 d'une alerte W selon un critère d'alerte portant sur la ou les incohérences détectées.

[0234] Chaque critère d'alerte dépend d'au moins une condition sur les paramètres représentatifs. Chaque critère d'alerte est par exemple une valeur seuil du paramètre mesuré.

[0235] Lors de la phase de génération 100 du jeu de référence J, il est possible de construire un jeu de référence exhaustif, au prix de nombreuses exécutions de l'application, avec un jeu d'événements différents. Au contraire, lors de la phase de détection 200, il n'est pas possible d'exécuter plusieurs fois l'application dans le but de détecter une attaque. La détection doit avoir lieu directement, lors de l'exécution.

[0236] Ainsi, seul un nombre limité de paramètres est mesuré à partir de blocs matériels de comptage 22 afin de minimiser le temps de détection.

[0237] Selon un mode de réalisation, les paramètres mesurés sont alternés rapidement afin de couvrir dans une courte période de temps l'ensemble des événements significatifs par échantillonnage.

[0238] En variante, les paramètres mesurés sont choisis au moyen d'une sélection hiérarchique des événements surveillés.

[0239] Lorsqu'au moins une alerte W associée à l'un des blocs matériels d'exécution 20 est émise, une nouvelle itération des étapes du procédé de détection 200 à partir de l'étape de mesure 210 est mise en œuvre.

[0240] L'ensemble des blocs matériels d'exécution 20 dont une mesure de paramètre représentatif est effectuée étant, à chaque itération, inclus dans l'ensemble des blocs matériels d'exécution 20 de la précédente itération, l'ensemble des blocs matériels d'exécution 20 étant notamment inclus dans l'ensemble des blocs matériels d'exécution 20 associés à la ou les alertes.

[0241] Ainsi, un petit nombre d'événements est choisi, correspondant au nombre de blocs matériels de comptage 22 disponibles et couvrant les différents blocs matériels

d'exécution 20 du processeur 14, afin de constituer un premier niveau de surveillance.

[0242] En cas de dépassement d'un ou plusieurs de ces événements par rapport à des seuils issus du jeu de référence, une suspicion d'attaque est levée et les paramètres mesurés sont reconfigurés pour se focaliser sur le bloc matériel d'exécution 20 concerné afin de confirmer ou infirmer la suspicion.

[0243] Les étapes de mesure 210 sont ainsi répétées avantageusement sur plusieurs niveaux hiérarchiques. Cette méthode permet d'améliorer la fiabilité de la détection malgré le nombre limité de blocs matériels de comptage 22.

[0244] Le procédé de détection 200 comprend, lorsqu'une alerte W associée à l'un des blocs matériels d'exécution 20 est émise, une nouvelle itération des étapes du procédé de détection 200 à partir de l'étape de mesure 210, le critère d'alerte comprenant, à chaque itération, un nombre croissant de conditions.

[0245] Ainsi, il est possible d'affiner l'alerte et de caractériser plus précisément la suspicion d'attaque.

[0246] Par exemple, une première condition d'alerte est le dépassement d'un seuil d'accès mémoire d'un bloc matériel. A la prochaine itération, la condition d'alerte comprend en plus de ce seuil, le dépassement d'un seuil de calculs exécutés par ce bloc matériel.

[0247] Le procédé 200 comprend ensuite l'analyse 240 de la nature du bloc matériel d'exécution 20 associé à chaque incohérence, l'analyse de la fréquence temporelle des incohérences, et/ou l'analyse de l'écart de chaque mesure associée aux incohérences à l'ensemble de données de référence.

[0248] Le procédé de détection 200 comprend alors une étape de détection 250, en fonction de la ou chaque analyse d'une anomalie, en particulier une attaque informatique ou une défaillance matérielle de l'un des blocs matériels.

[0249] En effet, en fonction de la fréquence, du type et de la gravité des évènements anormaux, il est possible de distinguer une attaque informatique d'une défaillance matérielle. En particulier, les évènements transitoires liés à la sûreté de fonctionnement sont essentiellement ponctuels, et se traduisent au niveau des blocs matériels de comptage 22 par un pic.

[0250] Les évènements liés à la cyber-sécurité au contraire durent le temps de l'attaque, et les attaques ont généralement un effet plus durable sur les blocs matériels de comptage 22.

[0251] En comparant l'incohérence associée à l'alerte émise avec un ensemble d'anomalies connues prédéterminées, le procédé comprend la catégorisation de l'alerte lorsque la ou chaque incohérence correspond à l'une des anomalies connues dudit ensemble d'anomalies.

[0252] Il est ainsi possible de connaitre la nature de l'anomalie rencontrée, notamment le type d'attaque informatique, et de réagir en fonction.

[0253] Toutefois, le procédé de détection 200 ne dépend pas du type d'attaque. Le procédé permet de détecter également des nouvelles classes d'attaques ciblant le matériel ou le logiciel.

[0254] Finalement, lors du vieillissement des composants, la fréquence des évènements transitoires augmente progressivement jusqu'à une défaillance permanente. Le procédé de détection, via cette analyse, permet d'observer cette variation de fréquence, pour participer à la maintenance préventive des composants et prévenir ce type de défaillances.

Revendications

[Revendication 1] Procédé de détection (200) d'une anomalie dans un système électronique (10) en fonctionnement, en particulier une attaque informatique contre le système électronique (10), le système électronique (10) comprenant un ou plusieurs processeurs (14) composés chacun d'une pluralité de blocs matériels (20, 22), le processeur (14) étant adapté pour exécuter au moins une application (A) par interactions de l'application (A) avec lesdits blocs matériels (20), le procédé de détection (200) comprenant au moins les étapes suivantes :

- mesure (210), lors du fonctionnement du système électronique (10), d'au moins un paramètre représentatif des interactions de l'application (A) avec l'un des blocs matériels (20) pour obtenir des mesures dudit paramètre représentatif,
- pour chaque mesure, comparaison (220) de ladite mesure avec un ensemble de données de référence (J) associé pour détecter d'éventuelles incohérences entre la mesure et l'ensemble de données de référence (J), ledit ensemble de données de référence (J) étant représentatif du fonctionnement du système électronique (10) sans anomalies, et
- émission (230) d'une alerte (W) selon un critère d'alerte portant sur la ou les incohérences détectées, le procédé de détection (200) comprenant, en outre, au moins une étape choisie parmi le groupe constitué de :
- analyse (240) de la nature du bloc matériel (20) associé à chaque incohérence,
- analyse (240) de la fréquence temporelle des incohérences, et
- analyse (240) de l'écart de chaque mesure associée aux incohérences à l'ensemble de données de référence (J) ;

le procédé de détection (200) comprenant une étape de détection (250), en fonction de la ou chaque analyse d'une anomalie, en particulier une attaque informatique ou une défaillance matérielle de l'un des blocs matériels (20).

[Revendication 2] Procédé de détection (200) selon la revendication 1, dans lequel lorsqu'au moins une alerte (W) associée à l'un des blocs matériels (20) est émise, une nouvelle itération des étapes du procédé de détection

(200) à partir de l'étape de mesure (210) est mise en œuvre, l'ensemble des blocs matériels (20) dont une mesure de paramètre représentatif est effectuée étant, à chaque itération, inclus dans l'ensemble des blocs matériels (20) de la précédente itération, l'ensemble des blocs matériels (20) étant notamment inclus dans l'ensemble des blocs matériels (20) associés à la ou les alertes (W).

[Revendication 3] Procédé de détection (200) selon l'une quelconque des revendications 1 à 2, dans lequel chaque critère d'alerte dépend d'au moins une condition sur les paramètres représentatifs, le procédé de détection (200) comprenant, lorsqu'une alerte (W) associée à l'un des blocs matériels (20) est émise, une nouvelle itération des étapes du procédé de détection (200) à partir de l'étape de mesure (210), le critère d'alerte comprenant, à chaque itération, un nombre croissant de conditions.

[Revendication 4] Procédé de détection (200) selon l'une quelconque des revendications 1 à 3, dans lequel chaque paramètre représentatif associé à l'un des blocs matériel (20) est choisi parmi le groupe constitué de :

- le type de données traitées par le bloc matériel (20);
- le type d'instructions exécutées par le bloc matériel (20);
- le nombre d'accès au bloc matériel (20);
- le nombre de calcul exécutés et/ou le temps de calcul du bloc matériel (20);
- le nombre de dysfonctionnements du bloc matériel(20), et
- le nombre d'interactions sur un réseau d'interconnexion interne au processeur (14) impliquant le bloc matériel (20), et
- le nombre d'interactions avec l'extérieur du processeur par le bloc matériel (20).

[Revendication 5] Procédé de détection (200) selon l'une quelconque des revendications 1 à 4, dans lequel chaque bloc matériel (20) est choisi parmi le groupe constitué de :

- une ou plusieurs unités de calcul,
- une ou plusieurs unités de prédiction de branchement,
- un ou plusieurs registres de mémoire interne du processeur,
- une ou plusieurs mémoires caches,
- une ou plusieurs mémoires vives,
- une ou plusieurs chaines de traitement,
- une ou plusieurs unités de protection mémoire ou de

traduction d'adresse,
- un ou plusieurs bus ou réseaux d'interconnexion,
- une ou plusieurs unités d'entrée-sortie

chaque paramètre représentatif du fonctionnement de l'un des blocs matériels (20) étant choisi parmi le groupe constitué de :

- le nombre de données d'un même type traitées par la ou les unités de calcul, le ou les réseaux d'interconnexion ou la ou les mémoires,
- le nombre d'instructions d'un même type exécutées par la ou les unités de calcul,
- la consommation d'énergie de la ou des unités de calcul,
- la température de la ou des unités de calcul,
- le nombre de réussites et/ou d'erreurs de prédiction de l'unité de prédiction de branchement,
- le nombre d'accès aux mémoires,
- le nombre de réussites et/ou d'erreurs de sollicitations de la mémoire cache,
- le nombre de réussites et/ou d'erreurs de sollicitations de l'unité de protection mémoire,
- le temps d'exécution d'une instruction dans la chaine de traitement,
- le nombre d'échanges d'information via le ou les réseaux d'interconnexion interne(s) au processeur,
- le nombre d'échanges d'information via le ou les réseaux d'interconnexion obtenus par une source donnée,
- le nombre d'échanges d'information via le ou les réseaux d'interconnexion envoyés à une destination donnée, et
- le nombre d'échanges d'informations via l'unité d'entrée-sortie.

[Revendication 6] Procédé de détection (200) selon l'une quelconque des revendications 1 à 5, dans lequel le système électronique (10) comprend une pluralité de blocs matériels de comptage (22) configurés chacun pour mesurer l'un des paramètres représentatifs, l'étape de mesure (210) comprenant la mesure d'une pluralité de paramètres représentatifs, notamment entre

quatre et six, par exemple les paramètres représentatifs mesurés par une partie des blocs matériels de comptage (22).

[Revendication 7] Procédé de détection (200) selon l'une quelconque des revendications 1 à 6, dans lequel le procédé de détection (200) comprend, en outre, une étape de :

- comparaison de la ou de chaque incohérence associée à l'alerte (W) émise avec un ensemble d'anomalies connues prédéterminées, et
- catégorisation de l'alerte (W) lorsque la ou chaque incohérence correspond à l'une des anomalies connues dudit ensemble d'anomalies.

[Revendication 8] Procédé de détection (200) selon l'une quelconque des revendications 1 à 7, dans lequel le système électronique (10) est un calculateur embarqué dans une plateforme de transport (12).

[Revendication 9] Procédé de détection (200) selon l'une quelconque des revendications 1 à 8, dans lequel le procédé de détection (200) est mis en œuvre par un système choisi dans le groupe constitué de :

- un circuit logique programmable dédié formant l'un desdits blocs matériels (20) du processeur ;
- un logiciel d'un système d'exploitation (16) du système électronique (10) configuré pour interagir avec une mémoire protégée par une unité de protection mémoire, et
- une application propre à être mise en œuvre par le système électronique (10) par adjonction d'un pilote contrôlant les blocs matériels (20).

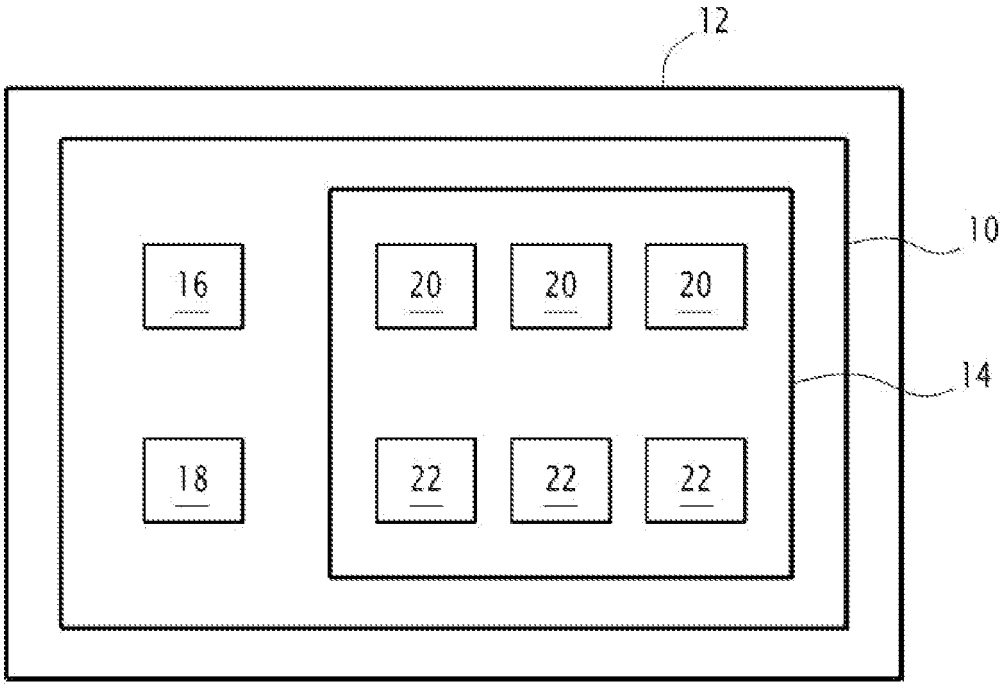
[Revendication 10] Produit programme d'ordinateur comportant un support lisible d'informations, sur lequel est mémorisé un programme d'ordinateur comprenant des instructions de programme, le programme d'ordinateur étant chargeable sur une unité de traitement de données et adapté pour entraîner la mise en œuvre d'un procédé de détection (200) selon l'une quelconque des revendications 1 à 9 lorsque le programme d'ordinateur est mis en œuvre sur l'unité de traitement des données.

[Revendication 11] Support lisible d'informations sur lequel est mémorisé un produit programme d'ordinateur selon la revendication 10.

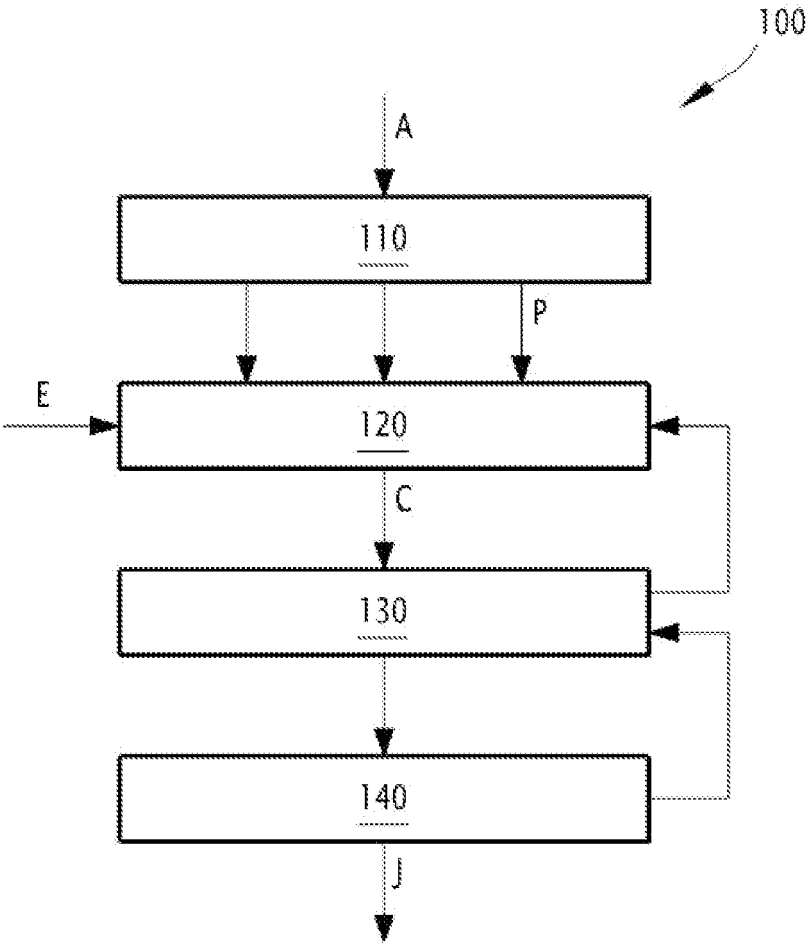
[Revendication 12] Calculateur embarqué dans une plateforme de transport (12) configuré pour mettre en œuvre le procédé de détection (200) selon l'une quelconque des revendications 1 à 11, le système électronique étant le calculateur embarqué.

[Revendication 13] Plateforme de transport (12) comprenant le calculateur selon la revendication 12.

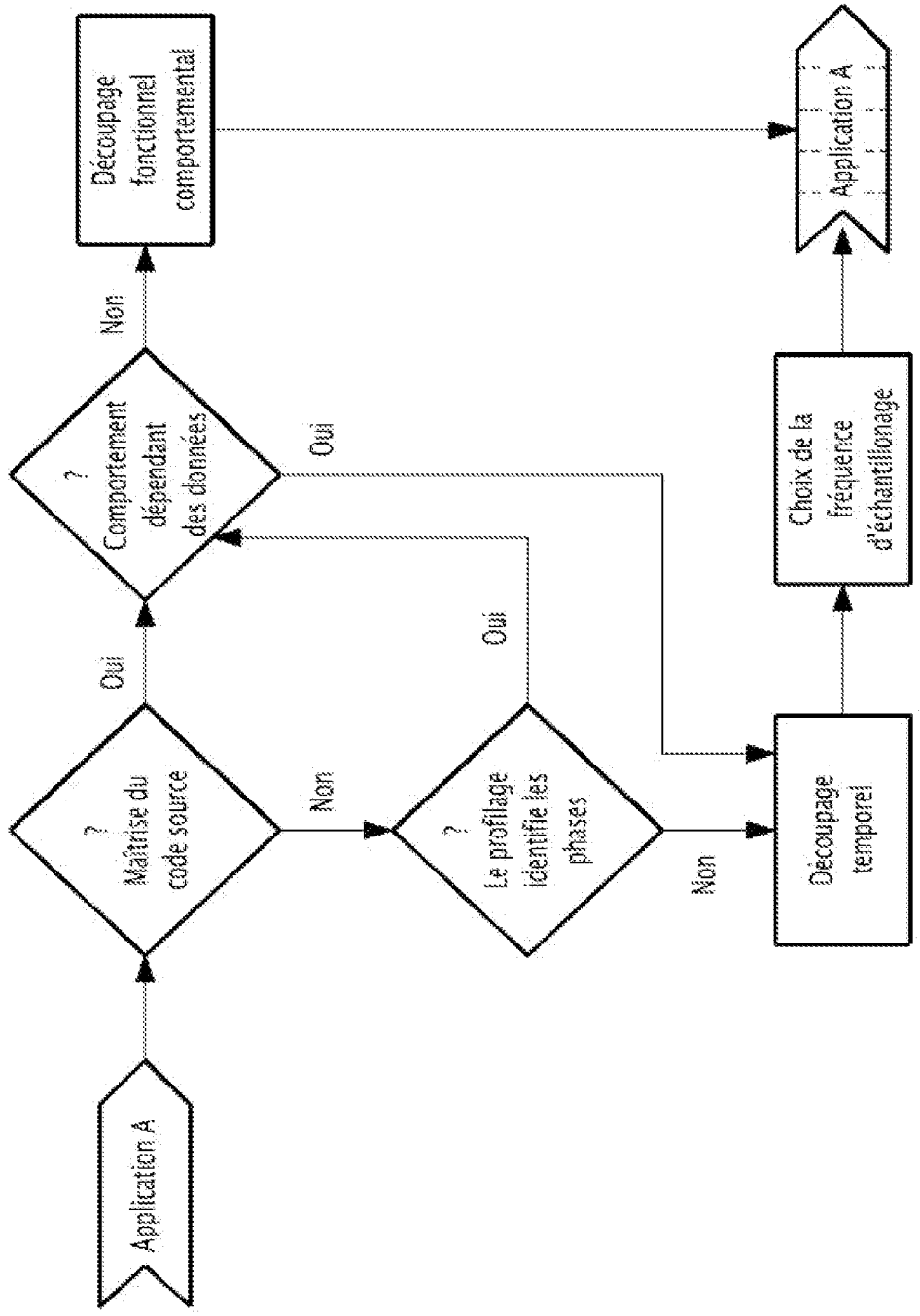
[Fig. 1]



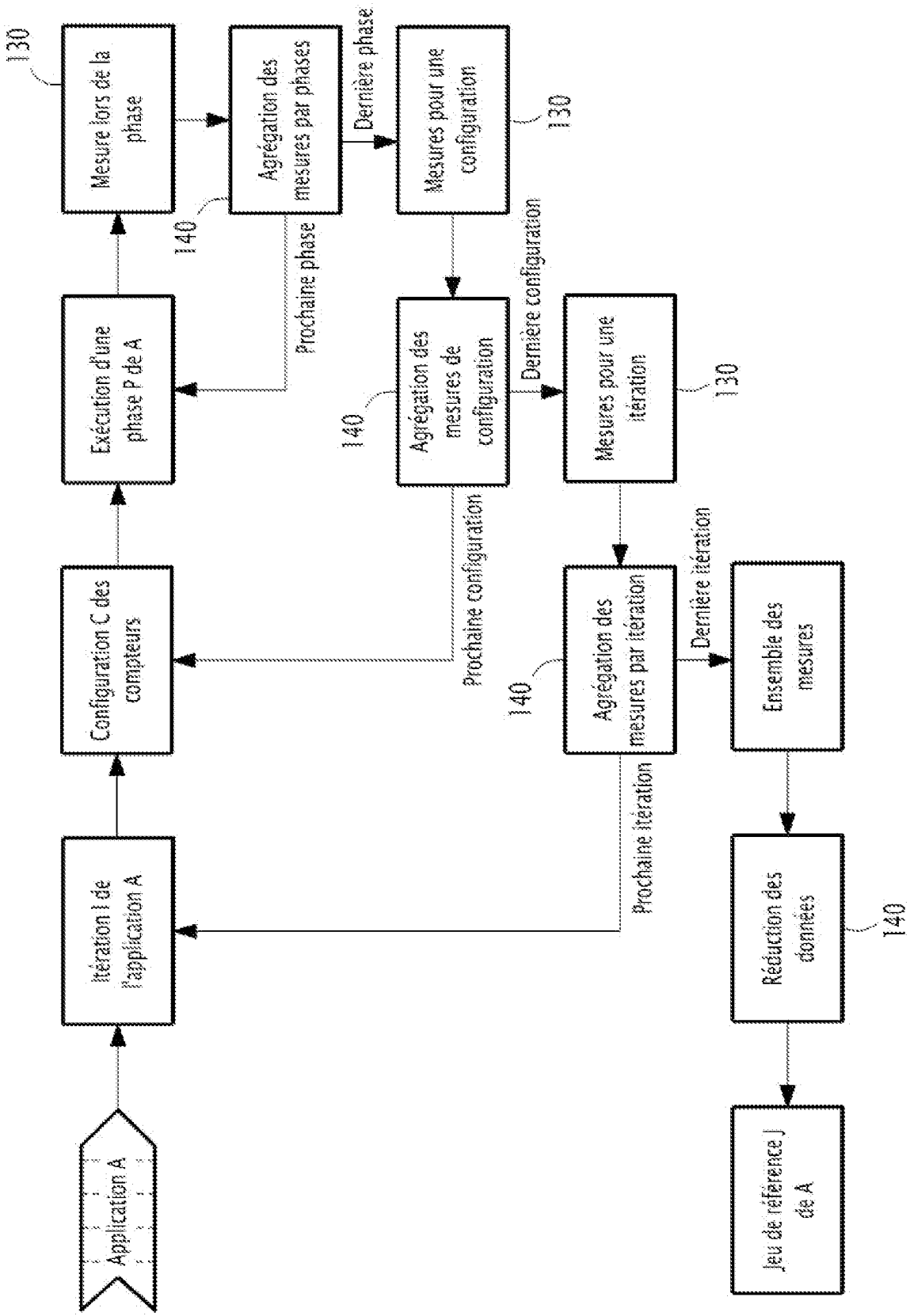
[Fig. 2]



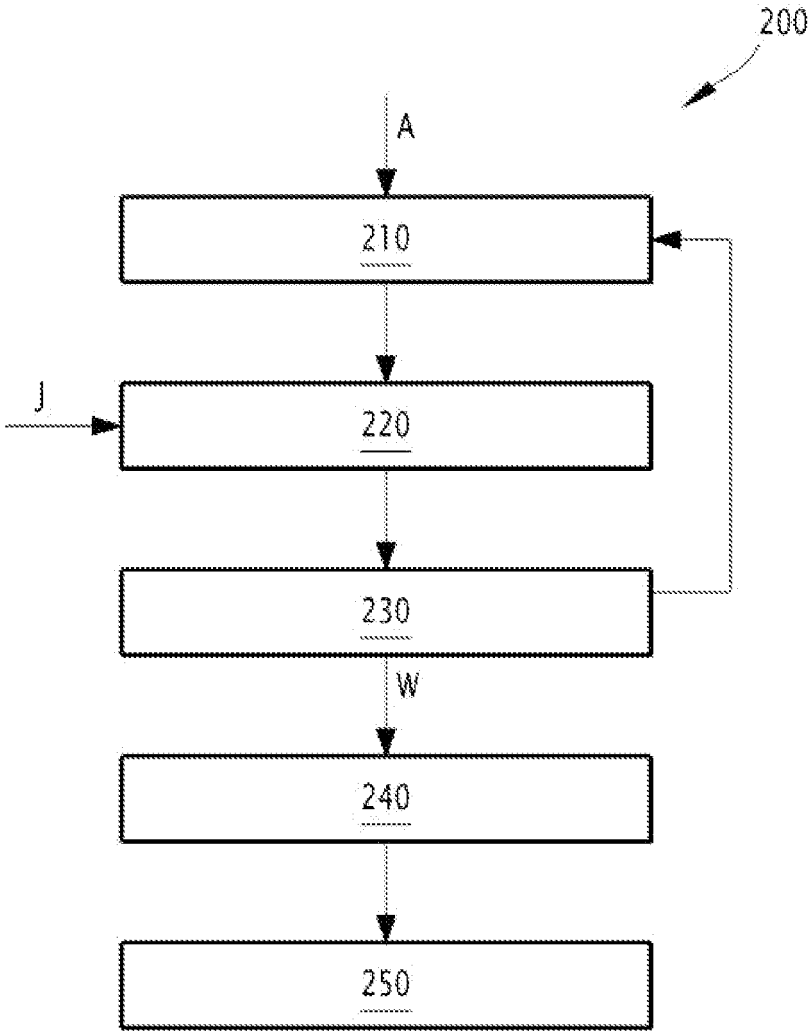
[Fig. 3]


Application A
? Maîtrise du code source
Oui
Non
? Le profilage identifie les phases
Oui
Non
? Comportement dépendant des données
Oui
Non
Découpage fonctionnel comportemental
Découpage temporel
Choix de la fréquence d'échantillonage
Application A

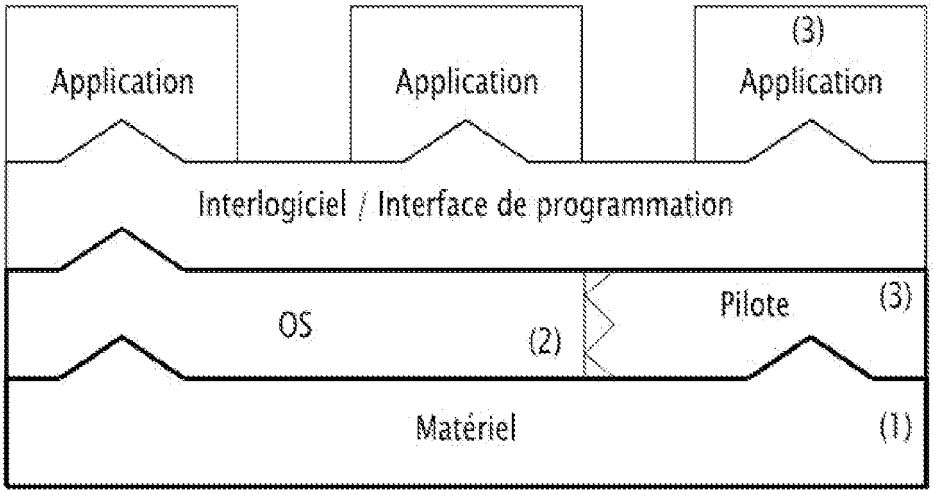
[Fig. 4]


Application A
Itération I de l'application A
Configuration C des compteurs
Exécution d'une phase P de A
Mesure lors de la phase
130
Agrégation des mesures par phases
140
Prochaine phase
Dernière phase
Mesures pour une configuration
130
Agrégation des mesures de configuration
140
Prochaine configuration
Dernière configuration
Mesures pour une itération
130
Agrégation des mesures par itération
140
Prochaine itération
Dernière itération
Ensemble des mesures
Réduction des données
140
Jeu de référence J de A

[Fig. 5]



[Fig. 6]


Application
Application
(3)
Application
Interlogiciel / Interface de programmation
OS
(2)
Pilote
(3)
Matériel
(1)

RAPPORT DE RECHERCHE

articles L.612-14, L.612-53 à 69 du code de la propriété intellectuelle

OBJET DU RAPPORT DE RECHERCHE

L'I.N.P.I. annexe à chaque brevet un "RAPPORT DE RECHERCHE" citant les éléments de l'état de la technique qui peuvent être pris en considération pour apprécier la brevetabilité de l'invention, au sens des articles L. 611-11 (nouveauté) et L. 611-14 (activité inventive) du code de la propriété intellectuelle. Ce rapport porte sur les revendications du brevet qui définissent l'objet de l'invention et délimitent l'étendue de la protection.
Après délivrance, l'I.N.P.I. peut, à la requête de toute personne intéressée, formuler un "AVIS DOCUMENTAIRE" sur la base des documents cités dans ce rapport de recherche et de tout autre document que le requérant souhaite voir prendre en considération.

CONDITIONS D'ETABLISSEMENT DU PRESENT RAPPORT DE RECHERCHE

[x] Le demandeur a présenté des observations en réponse au rapport de recherche préliminaire.

☐ Le demandeur a maintenu les revendications.

[x] Le demandeur a modifié les revendications.

☐ Le demandeur a modifié la description pour en éliminer les éléments qui n'étaient plus en concordance avec les nouvelles revendications.

☐ Les tiers ont présenté des observations après publication du rapport de recherche préliminaire.

☐ Un rapport de recherche préliminaire complémentaire a été établi.

DOCUMENTS CITES DANS LE PRESENT RAPPORT DE RECHERCHE

La répartition des documents entre les rubriques 1, 2 et 3 tient compte, le cas échéant, des revendications déposées en dernier lieu et/ou des observations présentées.

[x] Les documents énumérés à la rubrique 1 ci-après sont susceptibles d'être pris en considération pour apprécier la brevetabilité de l'invention.

☐ Les documents énumérés à la rubrique 2 ci-après illustrent l'arrière-plan technologique général.

☐ Les documents énumérés à la rubrique 3 ci-après ont été cités en cours de procédure, mais leur pertinence dépend de la validité des priorités revendiquées.

☐ Aucun document n'a été cité en cours de procédure.

1. ELEMENTS DE L'ETAT DE LA TECHNIQUE SUSCEPTIBLES D'ETRE PRIS EN CONSIDERATION POUR APPRECIER LA BREVETABILITE DE L'INVENTION

Aliénor Damien ET AL: "Anomaly Based Intrusion Detection for an Avionic Embedded System",
,
8 novembre 2018 (2018-11-08), page 1967646, XP055715503,
Warrendale, PA
DOI: 10.4271/2018-01-1941
Extrait de l'Internet:
URL:https://hal.laas.fr/hal-01967646/document
[extrait le 2019-01-01]

ZECHENG HE ET AL: "CloudShield: Real-time Anomaly Detection in the Cloud",
ARXIV.ORG, CORNELL UNIVERSITY LIBRARY, 201 OLIN LIBRARY CORNELL UNIVERSITY ITHACA, NY 14853,
20 août 2021 (2021-08-20), XP091036846,

US 2016/328561 A1 (TAMIR ELIEZER [IL] ET AL) 10 novembre 2016 (2016-11-10)

WO 2021/089357 A1 (ELECTRICITE DE FRANCE [FR]; CENTRE NAT RECH SCIENT [FR])
14 mai 2021 (2021-05-14)

2. ELEMENTS DE L'ETAT DE LA TECHNIQUE ILLUSTRANT L'ARRIERE-PLAN TECHNOLOGIQUE GENERAL

NEANT

3. ELEMENTS DE L'ETAT DE LA TECHNIQUE DONT LA PERTINENCE DEPEND DE LA VALIDITE DES PRIORITES

NEANT