(19) RÉPUBLIQUE FRANÇAISE

INSTITUT NATIONAL DE LA PROPRIÉTÉ INDUSTRIELLE

COURBEVOIE

(11) **N° de publication :** ***3 138 535***

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

(21) **N° d'enregistrement national :** **23 07844**

(51) Int Cl[8] : ***G 06 F 11/30*** *(2023.01), G 06 N 3/044, 20/00*

(12) DEMANDE DE BREVET D'INVENTION A1

(22) **Date de dépôt :** 21.07.23.

(30) **Priorité :** 28.07.22 DE 102022207776.8.

(43) **Date de mise à la disposition du public de la demande :** 02.02.24 Bulletin 24/05.

(56) **Liste des documents cités dans le rapport de recherche préliminaire :** *Ce dernier n'a pas été établi à la date de publication de la demande.*

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

( ) **Demande(s) d'extension :**

(71) **Demandeur(s) :** *ROBERT BOSCH GmbH GMBH* — DE.

(72) **Inventeur(s) :** SCHOLZ Bjoern, KULICKE David, WALTER Harald, TRINH Hoang, KAHLE Holger, KISHOR K B Mohan, DE MENEZES Marcio Jose Junior et FRUEHBERGER Peter.

(73) **Titulaire(s) :** ROBERT BOSCH GmbH GMBH.

(74) **Mandataire(s) :** CABINET HERRBURGER.

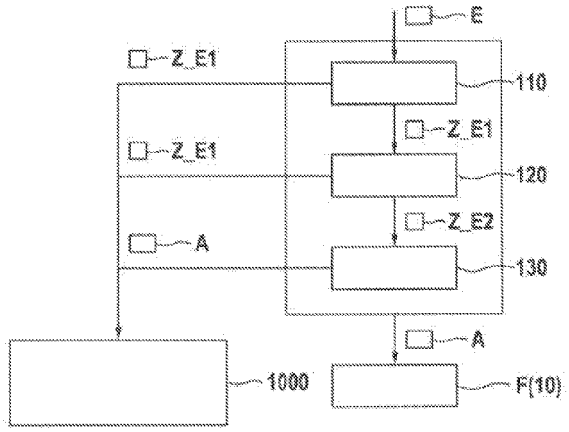
(54) **« Procédé et système de surveillance d'un modèle fonctionnel ».**

(57) TITRE : Procédé et système de surveillance d'un modèle fonctionnel

Procédé implémenté par ordinateur pour surveiller un modèle fonctionnel (100) pour fournir des données pour au moins une fonction d'une machine (10) commandée par ordinateur, notamment d'un algorithme de reconnaissance d'image,

- le modèle (100) définissant au moins un résultat intermédiaire (Z_E1, Z_E2) fondé sur les données d'entrée (E) et à partir du résultat intermédiaire (Z_E1, Z_E2) donne une sortie (A). Le résultat intermédiaire (Z_E1, Z_E2) et la sortie (A) du modèle fonctionnel (100) sont fournis à un modèle de surveillance (1000) pour la reconnaissance d'anomalies et suivie d'une étape de validation d'une fonctionnalité du modèle fonctionnel (100) et/ou d'une fonctionnalité du modèle de surveillance (1000).

Figure 1



FR 3 138 535 - A1

Description

Titre de l'invention : « Procédé et système de surveillance d'un modèle fonctionnel »

DOMAINE DE L'INVENTION

[0001] La présente invention se rapporte à un procédé et un système de surveillance d'un modèle fonctionnel.

ETAT DE LA TECHNIQUE

[0002] Les modèles d'apprentissage automatique sont utilisés de plus en plus par l'utilisation très étendue d'algorithme de vision d'ordinateur, par exemple, dans les véhicules automobiles. Un nombre croissant de caméras à l'intérieur et à l'extérieur du véhicule observent à tout instant l'environnement intérieur et extérieur du véhicule.

[0003] Les modèles d'apprentissage automatiques sont, par exemple, installés sur des systèmes sur une puce SoC puissants qui utilisent en général différents types d'accélérateurs matériels pour décharger les noyaux CPU.

[0004] En principe, il faut s'assurer qu'à la fois le modèle de l'apprentissage automatique et aussi le matériel qui l'applique répondent aux exigences de fonctionnement quant à la sécurité des fonctionnalités prévues comme, par exemple, le prévoit la norme ISO 21448. De plus, il faut respecter les conditions de sécurité fonctionnelle selon la norme ISO 26262 ; Cela est, par exemple, important si des systèmes d'exploitation ne sont pas sécurisés au sens de la norme ISO 26262 ou si l'on utilise des accélérateurs de matériel. Il est rare qu'un système complexe composé d'un système sur une puce SoC et d'autres composants techniques tels que, par exemple, des mémoires de travail, des alimentations, etc. se compose globalement de composants développés selon ISO 26262. C'est pourquoi, pour de tels systèmes, on définit fréquemment, de manière explicite un chemin critique particulièrement sécurisé.

[0005] EXPOSE ET AVANTAGES DE L'INVENTION

[0006] La présente invention a pour objet un procédé implémenté par ordinateur pour surveiller un modèle fonctionnel pour fournir des données pour au moins une fonction d'une machine commandée par ordinateur, notamment d'un algorithme de reconnaissance d'image, le modèle définissant au moins un résultat intermédiaire en se fondant sur les données d'entrée dans au moins une première étape de traitement et, le modèle fonctionnel, se fondant sur le résultat intermédiaire détermine, dans au moins une autre étape de traitement, une sortie du modèle fonctionnel, le résultat intermédiaire et la sortie du modèle fonctionnel sont fournis à un modèle de surveillance pour la reconnaissance d'anomalies et, en se fondant sur le résultat intermédiaire et la sortie du modèle fonctionnel on effectue une reconnaissance d'anomalies et, on

effectue au moins une étape de validation d'une fonctionnalité du modèle fonctionnel et/ou d'une fonctionnalité du modèle de surveillance.

[0007] L'expression modèle fonctionnel signifie dans le cas de la présente description, un modèle, notamment un algorithme permettant de fournir des données pour au moins une fonction d'une machine commandée par ordinateur.

[0008] Un exemple d'un tel modèle est un algorithme de reconnaissance d'images assistées par du matériel ou un programme et qui fournit des données de sortie, à partir de données d'entrée, qui, dans ce cas sont des données d'images, notamment numériques. Un tel algorithme est, par exemple, un algorithme de classification. Mais, il peut également être avantageux, en liaison avec la reconnaissance ou détection d'anomalies que le modèle soit un algorithme de régression. Mais cela n'exclut pas que le modèle ou une application en aval du modèle effectue une classification définitive.

[0009] Il est prévu que le modèle fonctionnel s'appuyant sur les données d'entrée détermine, dans au moins une première étape de traitement, par exemple, en s'appuyant sur l'inférence, au moins un résultat intermédiaire. Dans au moins une autre étape de traitement, par exemple, dans une étape de post-traitement, le modèle fonctionnel s'appuie sur le résultat intermédiaire pour déterminer la sortie du modèle fonctionnel.

[0010] A la fois le résultat intermédiaire et aussi la sortie sont soumis à un modèle de surveillance d'une détection d'anomalies. La détection d'anomalies vérifie les résultats du modèle fonctionnel surveillé quant aux éventuelles anomalies, par exemple, les données aberrantes. Le défit de la détection d'anomalies consiste à distinguer entre les vraies « données aberrantes » qui proviennent d'une erreur d'exécution d'une fonction de consigne et les fausses « données aberrantes » selon lesquelles des événements rares avec des variations brusques des entrées conduisent à de telles données aberrantes. Toute donnée aberrante ne provient pas nécessairement d'un défaut d'exécution d'une fonction de consigne. C'est pourquoi la seule détection des données aberrantes n'est pas suffisante pour conclure à une exécution erronée d'une fonction de consigne du modèle fonctionnel. Tout au contraire, les données aberrantes ne sont, en général, aucune preuve qu'il y a eu effectivement un événement d'erreur à reconnaître.

[0011] Selon l'invention, il est, pour cela, prévu de compléter la détection d'anomalies par au moins une étape de validation d'une fonctionnalité du module de fonction et/ou d'une fonctionnalité et du modèle de surveillance.

[0012] Pour valider la fonctionnalité du modèle de surveillance, on propose, par exemple, d'exécuter un autodiagnostic BIST.

[0013] Il est prévu que le procédé consiste à :

[0014] - déterminer une sortie de référence fondée sur une entrée de référence avec le modèle fonctionnel et de vérifier la sortie de référence, notamment comparer la sortie de référence à des données de vérité de terrain avec le modèle de surveillance. Une

entrée de référence est, par exemple, enregistrée dans une mémoire appropriée pour être ainsi fournie au modèle fonctionnel.

[0015] La dénomination donnée de « vérité de terrain » signifie que toute donnée représente une référence qui décrit de manière suffisamment précise la réalité des données de référence pour l'objectif respectif. En d'autres termes, les données de vérité de terrain sont des données observées ou mesurées qui ont été analysées de manière objective.

[0016] La vérification de la sortie de référence par le modèle de surveillance permet de vérifier, à l'aide des données de référence connues, si le modèle fonctionnel est apte à fonctionner. Au cas où le modèle fonctionnel comprend un algorithme de classification, il est avantageux que les entrées de référence couvrent différentes classes, notamment toutes les classes.

[0017] Il peut être prévu de déterminer une sortie de référence en se fondant sur une entrée de référence avec le modèle fonctionnel et de comparer la sortie de référence avec les données de vérité de terrain, périodiquement avec le modèle de surveillance. Cette exécution peut être, par exemple, déclenchée par la mise à disposition d'au moins une entrée de référence.

[0018] On peut également prévoir de déterminer la sortie de référence en se fondant sur l'entrée de référence avec le modèle fonctionnel pendant le fonctionnement normal, c'est-à-dire un fonctionnement approprié du modèle fonctionnel. Mais, il peut également arriver que le modèle fonctionnel commute sur le mode de diagnostic. Cela est, par exemple, avantageux si le modèle fonctionnel comprend un algorithme dont le fonctionnement aurait été influencé pendant un mode de fonctionnement normal par l'utilisation de données de référence. A titre d'exemple, il peut s'agir de réseaux de neurones récurrents RNN mais qui implémentent un état interne. Une entrée de référence pourrait influencer l'état interne et ainsi influencer également les prévisions qui en résultent.

[0019] En complément ou en variante, on peut prévoir d'autres étapes pour valider la fonctionnalité du modèle fonctionnel. Le procédé comprend, par exemple, les étapes suivantes : fournir une valeur de hachage avec le modèle de surveillance, signer la valeur de hachage avec le modèle fonctionnel, fournir la valeur de hachage signée au modèle de surveillance et vérifier la valeur de hachage signée, notamment comparer la valeur de hachage à la valeur de hachage signée en utilisant le modèle de surveillance.

[0020] Le modèle de surveillance génère, par exemple, un nombre aléatoire et aussi une valeur de hachage. La valeur de hachage est ensuite fournie au modèle fonctionnel. La valeur de hachage est une fonction mathématiquement non réversible pour garantir que le modèle fonctionnel ne rétablit pas de lui-même la valeur d'origine.

[0021] Signer la valeur de hachage par le modèle fonctionnel consiste, par exemple, à appliquer une fonction réversible à la valeur de hachage. Le modèle de surveillance

vérifie la valeur de hachage signée par l'application de l'inverse et compare la valeur de hachage avec la valeur de hachage générée initialement.

[0022] Selon une forme de réalisation, signer la valeur de hachage consiste à :

[0023] ajouter une signature à la valeur de hachage dans la première étape de traitement du modèle fonctionnel et ajouter une autre signature dans au moins une autre étape de traitement du modèle fonctionnel. Dans le cas où le modèle fonctionnel comprend d'autres étapes de traitement, il est prévu d'ajouter également des signatures dans les autres étapes de traitement. Les étapes de traitement suivantes du modèle fonctionnel appliquent alors leur signature, c'est-à-dire la fonction inverse appliquée à la valeur de hachage signée provenant de l'étape de traitement respective précédente, pour finalement générer une valeur de hachage signée définitivement qui est alors fournie au modèle de surveillance.

[0024] Il est prévu, de manière avantageuse, que le modèle fonctionnel détermine le résultat intermédiaire dans la première étape de traitement et génère une signature pour la valeur de hachage. Le résultat intermédiaire et la valeur de hachage signée sont transmis à au moins une autre étape de traitement. Dans cette autre étape de traitement, le modèle fonctionnel s'appuyant sur le résultat intermédiaire, détermine la sortie du modèle fonctionnel et aussi une signature pour la valeur de hachage signée de l'étape précédente. Il est également prévu que, pour chaque étape de traitement, de transmettre un résultat intermédiaire généré ainsi que la valeur de hachage signée à l'étape de traitement suivante.

[0025] La valeur de hachage signée définitivement est finalement transmise au modèle de surveillance avec la sortie.

[0026] Selon une autre forme de réalisation, le procédé consiste en outre à :

[0027] - surveiller la communication entre une première instance qui applique le modèle fonctionnel et une autre instance qui applique le modèle de surveillance. La surveillance permet, par exemple, de détecter des retards dans la communication. Si certains retards dépassent un seuil donné, il n'est pas certain que la surveillance fonctionne encore correctement.

[0028] Selon une forme de réalisation, en fonction d'un résultat de détection d'anomalies et/ou en fonction d'un résultat de surveillance de la communication, on applique au moins une des étapes suivantes :

[0029] a. vérifier le résultat de la comparaison de la sortie de référence avec les données de vérité de terrain en utilisant le modèle de surveillance,

[0030] b. vérifier le résultat de la comparaison de la valeur de hachage avec la valeur de hachage signée en utilisant le modèle de surveillance,

[0031] c. fournir un signal de commande pour commander la machine commandée par ordinateur, notamment au moins une partie de la machine commandée par ordinateur et/

ou une fonction de la machine commandée par ordinateur,

[0032] d. transférer un état défini à la machine commandée par ordinateur, notamment à au moins une partie de la machine commandée par ordinateur et/ou à une fonction de la machine commandée par ordinateur,

[0033] e. transférer un résultat de a) et/ou b) à la machine commandée par ordinateur notamment au moins une partie de la machine commandée par ordinateur et/ou à la fonction de la machine commandée par ordinateur dans l'état défini en fonction d'un résultat selon a) et/ou b).

[0034] Un état défini est, par exemple, un état sécurisé. Un état sécurisé signifie que la machine commandée par ordinateur et/ou la fonction de la machine commandée par ordinateur est mise dans un état dans lequel exécuter la fonction n'est pas fondé sur le modèle fonctionnel, par exemple, la coupure ou l'interruption de la fonction. En plus, il est prévu avantageusement, que la coupure ou l'interruption de la fonction notamment des composants qui exécutent cette fonction et/ou de la machine commandée par ordinateur sont signalées à un autre système, par exemple, un système E/E d'un véhicule automobile et/ou d'un utilisateur du système. Un exemple d'état sécurisé pour certaines fonctions et/ou des machines commandées par ordinateur dans un véhicule automobile est l'affichage visuel sur un écran, par exemple, également sous la forme d'un écran noir, d'un voyant de signalisation et/ou d'un signal acoustique.

[0035] D'autres formes de réalisation concernent le système de surveillance d'un modèle fonctionnel pour fournir des données pour au moins une fonction d'une machine commandée par ordinateur notamment d'un algorithme de reconnaissance d'image, le système étant réalisé pour appliquer des étapes d'un procédé selon les formes de réalisation décrites avec un modèle de surveillance, à savoir au moins une étape du procédé exécuté avec le modèle de surveillance, dans au moins une première instance du système et que le modèle fonctionnel, pour fournir les données pour au moins une fonction de la machine commandée par ordinateur, notamment l'algorithme de reconnaissance d'images soit appliqué dans au moins une seconde instance du système.

[0036] Pour respecter les conditions de sécurité selon la norme ISO 26262 on peut, par exemple, prévoir que le modèle fonctionnel soit appliqué selon la norme ISO 26262 dans un processeur d'application sécurisé avec un système de fonctionnement sécurisé et que le modèle de surveillance se trouve dans le même système. Dans un tel scénario, le modèle fonctionnel et le système périphérique sont déjà sûrs du point de vue de la sécurité fonctionnelle car cela est fait selon la norme ISO 26262. Une telle implémentation n'est toutefois pas toujours prévue, par exemple, pour des raisons d'économie.

[0037] De manière avantageuse, notamment pour la première instance, il suffit d'un niveau

de sécurité d'intégrité, par exemple, le niveau d'intégrité ASIL selon la norme ISO 26262.

[0038] De façon avantageuse, la première et la seconde instance sont réalisées comme une instance d'un système sur une puce SoC commun, le système d'exploitation qui est associé à la première instance étant exécuté sur un noyau de calcul séparé du système sur une puce. De cette manière, on arrive à l'absence d'interférence.

[0039] Il est prévu, de manière avantageuse, que la première instance et la seconde instance sont appliquées par un système réparti. On peut également prévoir que la première instance soit appliquée par un premier processeur et la seconde instance, par un co-processeur distinct. La communication entre ces instances doit avoir une sécurité appropriée qui garantit l'intégrité des informations ainsi que l'authenticité des informations et la complétude de toutes les informations transmises.

[0040] De façon avantageuse, un hyperviseur certifié, approprié, virtualise un plan de matériel et la première instance est le domaine de l'hyperviseur et la seconde instance est un autre domaine de l'hyperviseur.

[0041] D'autres formes de réalisation concernent l'application du procédé selon les formes de réalisation et/ou le système selon les formes de réalisation dans une machine commandée par ordinateur, par exemple, un système E/E d'un véhicule automobile, notamment pour fournir des fonctions de conduite autonome, de conduite partiellement autonome et/ou de fonctions d'assistance de conduite, un robot, un appareil électroménager, un outil électroportatif, une machine de fabrication, un dispositif d'inspection optique automatique ou un système d'accès, un modèle fonctionnel étant assuré dans la machine commandée par ordinateur en s'appuyant sur des données d'entrée, notamment des données d'images numériques fournies par un capteur d'images, notamment un capteur vidéo, radar, lidar, à ultrasons, un capteur de mouvements ou un capteur d'images thermiques. Pour avoir des données pour au moins une fonction de la machine commandée par ordinateur et en se fondant sur les données, pour avoir au moins un signal de commande pour exécuter la fonction de la machine commandée par ordinateur, le modèle fonctionnel est surveillé selon le procédé et/ou la machine commandée par ordinateur, notamment au moins une partie de la machine commandée par ordinateur et/ou une fonction de la machine commandée par ordinateur étant mise dans un état défini.

Brève description des dessins

[0042] La présente invention sera décrite ci-après de manière plus détaillée à l'aide de modes de réalisation de l'invention représentés dans les dessins annexés dans lesquels :

[0043] [Fig.1] vue d'ensemble schématique des étapes d'un procédé de surveillance d'un modèle fonctionnel avec un modèle de surveillance selon un premier mode de réa-

lisation,

[0044] [Fig.2] vue d'ensemble schématique des étapes d'un procédé de surveillance d'un modèle fonctionnel avec un modèle de surveillance selon un autre mode de réalisation,

[0045] [Fig.3] vue d'ensemble schématique des étapes d'un procédé de surveillance d'un modèle fonctionnel avec un modèle de surveillance selon un autre mode de réalisation,

[0046] [Fig.4] vue d'ensemble schématique d'un système de surveillance d'un modèle fonctionnel avec un modèle de surveillance selon un premier mode de réalisation, et

[0047] [Fig.5] vue d'ensemble schématique d'un système de surveillance d'un modèle fonctionnel avec un modèle de surveillance selon un second mode de réalisation.

[0048] DESCRIPTION DE MODES DE REALISATION DE L'INVENTION

[0049] La [Fig.1] montre schématiquement un modèle fonctionnel 100. Le modèle fonctionnel 100 est, par exemple, un modèle d'apprentissage automatique, notamment un algorithme pour fournir des données pour au moins une fonction F d'une machine 10 commandée par ordinateur.

[0050] Un exemple d'un tel modèle est un algorithme de reconnaissance d'images pour déterminer des données de sortie fondées sur des données d'entrée qui, dans ce cas, sont notamment des données d'images numériques. Un tel algorithme est, par exemple, un algorithme de classification. Mais, il peut également être avantageux, en liaison avec une reconnaissance d'anomalies que le modèle soit un algorithme par régression. Mais, cela n'exclut pas que le modèle ou une application en aval du modèle applique une classification définitive.

[0051] Les modèles d'apprentissage automatique sont utilisés dans de multiples domaines, par exemple, dans le domaine des véhicules automobiles, par exemple, dans le domaine de la conduite autonome ou partiellement autonome ou encore dans le domaine des fonctions d'assistance de conduite qui doivent répondre à des exigences de sécurité poussées. Dans le domaine des véhicules automobiles, il y a également des conditions concernant la sécurité fonctionnelle selon la norme ISO 26262 et des conditions de sécurité de la fonction de consigne (norme SOTIF selon ISO 21448 (SOTIF signifie « Sécurité des fonctionnalités prévues » ou « sécurité de la fonction attendue »).

[0052] La sécurité fonctionnelle selon la norme ISO 26262 est une sécurité intrinsèque vis-à-vis des fonctions de défaut, c'est-à-dire la sécurisation contre des fonctions de défaut autoprovoquées. Il s'agit, par exemple, d'aspects spécifiques à des circuits et des implémentations de programmes. La norme SOTIF selon ISO 21448 peut se définir de façon générale comme non existence d'un risque non mesuré à cause d'un danger occasionné par une incohérence dans la spécification ou des limitations de puissance dans l'implémentation. Il s'agit, par exemple, de la fonction d'implémentation de programme.

[0053] Selon les figures 1 à 3 on décrira tout d'abord un procédé de surveillance du modèle fonctionnel qui remplit à la fois les exigences concernant la sécurité fonctionnelle selon la norme ISO 26262 et aussi les exigences SOTIF selon la norme ISO 21448.

[0054] Selon les figures 1 à 3, on suppose appliquer au moins un modèle de surveillance, une sorte de superviseur pour surveiller le modèle fonctionnel 100 dans un environnement sécurisé, notamment dans du matériel pour répondre aux exigences selon les normes ISO 26262 et ISO 21448 spécifiques au matériel.

[0055] Le modèle fonctionnel 100 se fondant sur les données d'entrée E, par exemple des données d'image, donne dans au moins une première étape de traitement 110 au moins un résultat intermédiaire Z_E1.

[0056] Selon l'exemple, le modèle fonctionnel 100 comprend une seconde étape de traitement 120 et une troisième étape de traitement 130. Dans la seconde étape de traitement 120, le modèle fonctionnel 100 s'appuyant sur le premier résultat intermédiaire Z_E1 donne un second résultat intermédiaire Z_E2. Dans la troisième étape de traitement 130, le modèle fonctionnel 100 s'appuyant sur le second résultat intermédiaire Z_E2 détermine une sortie A. La sortie A est fournie, par exemple, comme donnée d'au moins une fonction F d'une machine commandée par ordinateur 10, par exemple, une fonction d'un système E/E d'un véhicule automobile.

[0057] Selon l'exemple, il est prévu qu'à la fois les résultats intermédiaires Z_E1, Z_E2 et la sortie A du modèle fonctionnel 100 soient fournis par un modèle de surveillance 1000 pour la détection d'anomalies.

[0058] Le modèle de surveillance 1000 se fondant sur les résultats intermédiaires Z_E1, Z_E2 et la sortie A du modèle fonctionnel 100 fait une détection d'anomalies. Dans cette détection ou recherche d'anomalies, on vérifie à la fois les résultats intermédiaires du modèle fonctionnel surveillé 100 quant aux anomalies, par exemple, ce qui est appelé « une donnée aberrante ». Un défit de la reconnaissance d'anomalies est celui de distinguer entre une vraie « donnée aberrante » qui provient d'une exécution erronée d'une fonction de consigne et d'une fausse « donnée aberrante » pour laquelle des événements rares, avec des variations brusques des entrées, conduisent également à de telles données aberrantes. En conséquence, chaque donnée aberrante ne correspond pas nécessairement à un problème. C'est pourquoi, la détection de données apparentes en elle-même n'est pas suffisante pour conclure à l'exécution erronée d'une fonction de consigne du modèle fonctionnel.

[0059] Il est ainsi prévu en complément de la détection d'anomalies d'effectuer au moins une étape pour valider une fonctionnalité du modèle fonctionnel 100 et/ou une fonctionnalité du modèle de surveillance 1000.

[0060] Pour valider la fonctionnalité du modèle fonctionnel 100, on effectue, par exemple, un « autodiagnostic intégré » désigné sous l'acronyme BIST. Comme cela sera décrit

en liaison avec la [Fig.2].

[0061] Par exemple, le procédé consiste à :

[0062] - déterminer une sortie de référence R_A en se fondant sur une entrée de référence R_E à l'aide du modèle fonctionnel 100 et,

[0063] - vérifier la sortie de référence R_A notamment comparer la sortie de référence R_A à des données de vérité de terrain avec le modèle de surveillance 1000. Une entrée de référence R_E se trouve, par exemple, dans une mémoire et sera fournie au modèle fonctionnel 100. Par la vérification de la sortie de référence R_A avec le modèle de surveillance 1000 on peut, avec les données de référence, connues, vérifier si le modèle fonctionnel 100 est apte à fonctionner. Si le modèle fonctionnel 100 comprend un algorithme de classification, il est avantageux que les données de référence R_E couvrent différentes classes, notamment toutes les classes.

[0064] Il peut être prévu de déterminer périodiquement une sortie de référence R_A en se fondant sur une entrée de référence R_E avec le modèle fonctionnel 100 et comparer la sortie de référence R_A avec les données de vérité de terrain avec le modèle de surveillance 1000. L'exécution peut être déclenchée, par exemple, par la fourniture de l'entrée de référence R_E.

[0065] Comme le montre la [Fig.1] et la description correspondante, le modèle fonctionnel traite l'entrée E pour la sortie A, se compose de plusieurs étapes de traitement. Les étapes de traitement sont, par exemple, effectuées de manière linéaire l'une après l'autre, dans le sens d'un procédé « pipeline » linéaire. Dans d'autres cas d'application non représentés, les étapes de traitement peuvent être reliées entre elles sous la forme d'un graphe acyclique, orienté.

[0066] Dans la suite, on décrira des étapes qui peuvent être effectuées pour valider la fonctionnalité du modèle fonctionnel. On propose des étapes pour vérifier si une quelconque étape de traitement a été effectuée et participe ainsi à la sortie A. Ces étapes peuvent également être regroupées dans une validation pipeline.

[0067] A titre d'exemple, on prévoit les étapes suivantes :

[0068] Le modèle de surveillance 1000 fournit une valeur de hachage h. Cela résulte, par exemple, de la génération d'un nombre aléatoire z et de la génération d'une valeur de hachage h(z). La valeur de hachage h(z) sera fournie ensuite au modèle fonctionnel 100. La valeur de hachage h(z) est une fonction non réversible mathématiquement pour garantir que le modèle fonctionnel 100 ne risque de fournir, lui-même, la valeur initiale. Comme une transmission d'erreur dans le modèle fonctionnel ou entre les étapes dans le modèle fonctionnel ne peut être exclue, il n'est pas certain que la valeur de hachage h(z) fournie correspond à la valeur reçue dans le modèle fonctionnel et c'est pourquoi on utilise la dénomination h'(z) ou simplement h' comme notation dans le modèle fonctionnel et qui, dans le bon cas, est identique à h(z) ou à h.

[0069] Dans la première étape de traitement 110 du modèle fonctionnel 100, en s'appuyant sur l'entrée E on génère le résultat intermédiaire Z_E1. Ensuite, on signe la valeur de hachage h'(z), par exemple, en appliquant une fonction réversible s1, par exemple h'os1(z).

[0070] Le résultat intermédiaire Z_E1 obtenu dans la première étape de traitement est fourni avec la valeur de hachage signée h'(z)os1 à la seconde étape de traitement 120.

[0071] Dans la seconde étape de traitement 120, le modèle fonctionnel 100 s'appuyant sur le premier résultat intermédiaire Z_E1 fournit un second résultat intermédiaire Z_E2. Ensuite, on signe une nouvelle fois la valeur de hachage signée h'(z)os1, par exemple, en appliquant une fonction réversible s2, par exemple h'(z)os1os2.

[0072] Le résultat intermédiaire Z_E2 obtenu dans la seconde étape de traitement 120 est fourni avec la valeur de hachage signée h'(z)os1os2 à la troisième étape de traitement 130.

[0073] Dans la troisième étape de traitement 130, le modèle fonctionnel 100 s'appuyant sur le second résultat intermédiaire Z_E2 détermine une sortie A. Ensuite, on signe une nouvelle fois la valeur de hachage signée h'(z)os1os2, par exemple, en appliquant une fonction réversible s3, par exemple, h'(z)os1os2os3. Dans la dernière étape de traitement on génère en quelque sorte la signature définitive S(z)=h'(z)os1os2os3o…

[0074] La signature définitive S(h'(z)) est fournie avec la sortie A au modèle de surveillance 1000. Le modèle de surveillance 1000 vérifie la signature finale ou définitive S(h'(z)). Le modèle de surveillance 1000 utilise, par exemple, l'inverse des signatures réversibles s'n…s'1 successivement, par exemple S(h'(z))os'n…os'1=h''(z). La valeur de hachage h''(z) ainsi obtenue est alors comparée à la valeur de hachage h(z) générée à l'origine. Lors de la transmission initiale, il y a pu y avoir une transmission avec erreur de h(z) et c'est pourquoi le modèle fonctionnel n'exclut pas une valeur h'(z) différente de la valeur h(z). En outre, dans des étapes avec erreur, la signature a pu ne pas être appliquée correctement et c'est pourquoi, après des calculs inverses dans le modèle de surveillance, on calcule une troisième valeur h''(z) qui ne correspond ni à h(z), ni à h'(z). Dans un cas correct h=h'=h''.

[0075] Grâce à cette extension, on est assuré que la transmission initiale de la valeur de hachage h(z) est également prise en compte.

[0076] Le modèle de surveillance 1000 diagnostique également la valeur de hachage déterminée h''(z) quant à l'identité avec la valeur initiale h(z) ; z n'est connu que du modèle de surveillance 1000. Si les deux valeurs de hachage coïncident, cela valide que chaque étape de traitement du modèle fonctionnel 100 a bien fonctionné.

[0077] Il peut être avantageux que la validation pipeline décrite soit effectuée dans chaque itération, c'est-à-dire pour chaque passage du modèle fonctionnel en se fondant sur une entrée E.

[0078] Il peut être avantageux dans chaque itération de générer une nouvelle valeur du nombre aléatoire z. Cela garantit que les différentes étapes de traitement dans chaque itération sont demandées de nouveau et interdit d'utiliser de signature constante. Une fonction simple, mais néanmoins difficilement réversible est, par exemple, une simple opération binaire XOR désignée par l'opérateur $\otimes$ car il s'agit d'une fonction auto inverse :

$$(s \circ s)(h) = s(s(h)) = h \text{ et } h \otimes s \otimes s = h.$$

[0079] Pour éviter que les fonctions réversibles de chaque étape de traitement ne soient codées de manière définitive dans le modèle de surveillance, on peut prévoir que pour une quelconque étape de traitement, on a, comme signature sn sur le fondement de la signature d'origine h(z) et de l'étape de traitement, par exemple sn(z) = h(z) + h(n). Dans ce cas, il faut que le modèle de surveillance connaisse uniquement le nombre d'étapes pipeline n pour l'inversion à partir de sn.

[0080] Il s'est avéré comme avantageux que la validation pipeline reste activée pour un diagnostic BIST en cours, c'est-à-dire que l'on effectue en même temps un diagnostic BIST et une validation pipeline. Cela garantit que chaque étape de traitement du modèle fonctionnel 100 est effectivement exécuté dans le diagnostic BIST.

[0081] Un exemple de déroulement d'un procédé 300 de surveillance du modèle fonctionnel 100 sera décrit ci-après, en référence à la [Fig.3].

[0082] En se fondant sur une entrée I, par exemple, comprenant des entrées E et des valeurs de hachage h, le modèle fonctionnel 100 se traduit par des étapes de traitement 110, 120, 130 et génère, le cas échéant, comme précédemment décrit, les signatures correspondantes pour les valeurs de hachage. Cela est, par exemple, regroupé dans l'étape 310.

[0083] Dans l'exemple, on a représenté un déclencheur périodique T qui déclenche l'exécution du diagnostic BIST. L'exécution du diagnostic BIST est, par exemple, regroupée dans l'étape 320.

[0084] Le modèle de surveillance 1000 effectue une détection d'anomalies comme cela est, par exemple, regroupé dans l'étape 330.

[0085] Si une anomalie est détectée, le modèle de surveillance 1000 effectue d'autres étapes pour distinguer entre une vraie donnée « aberrante » qui provient d'une exécution erronée d'une fonction de consigne et une fausse « donnée aberrante » dans le cas d'événements rares avec des variations brusques des entrées aboutissant à de telles données aberrantes. Mais, toute donnée aberrante ne signifie pas qu'il s'agit nécessairement d'une exécution erronée d'une fonction de consigne du modèle fonctionnel.

[0086] Lorsqu'une anomalie est détectée, le modèle de surveillance 1000 vérifie, dans

l'étape 340, dans l'exemple, le résultat du diagnostic BIST.

[0087] Si un défaut BIST est détecté, on en conclut qu'il y a une exécution défectueuse du modèle fonctionnel 100. On peut prévoir de dépasser dans un état sécurisé décrit ensuite.

[0088] S'il n'y a pas de défaut de diagnostic BIST (défaut BIST) détecté, le modèle de surveillance 1000 vérifie dans l'étape 350, par exemple, le résultat de la validation pipeline.

[0089] Si on détecte un défaut de validation pipeline PV, on conclut qu'il y a une exécution défectueuse du modèle fonctionnel 100. On peut prévoir de passer directement dans un état de sécurité.

[0090] Si aucun défaut de validation pipeline n'a été détecté, on en conclut qu'il s'agit d'une fausse « donnée aberrante » et qu'il n'y a pas eu d'exécution défectueuse du modèle fonctionnel 100.

[0091] Selon une forme de réalisation, le procédé consiste en outre à :

[0092] - surveiller la communication entre une première instance dans laquelle le modèle fonctionnel 100 est appliqué et une autre instance dans laquelle le modèle de surveillance 1000 est appliqué. Par la surveillance, on peut, par exemple, détecter des retards dans la communication. Si certains retards dépassent un seuil, il n'est pas garanti que la surveillance continue de fonctionner correctement. On peut alors, par exemple, déclencher également un autre état sécurisé.

[0093] L'expression « état sécurisé » signifie que la machine 10 commandée par ordinateur et/ou la fonction F de la machine 10 commandée par ordinateur est passée dans un état dans lequel l'exécution de la fonction F ne s'appuie pas sur le modèle fonctionnel 100, par exemple, à cause de la coupure ou de l'interruption de la fonction F. En plus, il est prévu avantageusement que la coupure ou l'interruption de la fonction F notamment du composant qui exécute cette fonction et/ou la machine commandée par ordinateur est informée à un autre système, par exemple, un système E/E d'un véhicule automobile et/ou l'utilisateur du système.

[0094] Les fonctions de surveillance décrites du superviseur remplissent à la fois les conditions de la norme ISO 26262 et aussi celles de SOTIF de la norme ISO 21448.

[0095] La détection de l'anomalie vérifie si le modèle fonctionnel 100 a fonctionné comme prévu pour une entrée réelle E et passe ainsi sous la norme SOTIF.

[0096] Le diagnostic BIST est fondé sur une entrée de référence R_E et ne peut ainsi vérifier aucun aspect de SOTIF. Le diagnostic BIST est utilisé pour vérifier si le modèle fonctionnel est encore en activité et susceptible de fonctionner. Le diagnostic BIST répond ainsi à la norme ISO 26262 concernant la sécurité de fonctionnement.

[0097] La validation de pipeline se fait en combinaison avec les entrées réelles E. La validation pipeline vérifie également si le modèle fonctionnel est encore actif et apte à

fonctionner. C'est pourquoi la validation pipeline applique à la fois la norme SOTIF et la norme ISO 26262.

[0098] La surveillance des retards de la communication se fait pour des entrées réelles E. On vérifie néanmoins également si le modèle fonctionnel est encore actif et apte à fonctionner. C'est pourquoi, la surveillance de la communication concernant les retards applique à la fois la norme SOTIF et la norme ISO 26262.

[0099] A l'aide des figures 4 et 5 on décrira différentes configurations matérielles pour différentes applications.

[0100] La [Fig.4] montre un système 400 conçu pour effectuer les étapes du procédé décrit avec un modèle de surveillance 1000. Il est prévu d'exécuter les étapes du procédé par le modèle de surveillance 1000 qui ont été exécutées dans au moins une première instance 410 du système 400 et que le modèle fonctionnel 100 fournit des données pour au moins une fonction F de la machine commandée par ordinateur 10, notamment un algorithme de reconnaissance d'images, exécuté dans une seconde instance 420 du système 400.

[0101] La seconde instance 420 est, par exemple, une instance qui n'est pas considérée comme sûre dans le sens de la norme de la sécurité fonctionnelle ISO 26262, par exemple, par l'application d'un système de fonctionnement qui n'est pas sûr ou d'un accélérateur de matériel qui n'est pas sûr. Par exemple, le modèle fonctionnel 100 est implémenté dans un système d'exploitation Linux. Toutefois, le noyau Linux ne remplit pas la certification de sécurité selon ISO 26262. Pour des raisons de sécurité et aussi d'autres raisons telles que le temps de démarrage ou encore l'assistance CAN, Linux est souvent utilisé avec d'autres domaines qui s'appliquent à un coprocesseur distinct ou un système hôte d'un hyperviseur. Ce domaine intègre, en général, un petit système de fonctionnement en temps réel (RTOS) avec des temps de démarrage rapide et qui émet et reçoit sur un bus de connectivité de véhicule automobile tel que CAN, Ethernet ou autres. Du fait de leur simplicité, de tels domaines peuvent être certifiés pour des applications concernant la sécurité. La première instance est, par exemple, associée à des domaines sécurisés. Il est, pour cela, prévu qu'au moins la première instance répond au niveau de sécurité ASIL selon la norme ISO 26262.

[0102] La première et la seconde instances 410, 420 peuvent se trouver sur des puces distinctes. On peut également prévoir que la première et la seconde instance 410, 420 soient réalisées comme une seule instance sur une puce commune (SoC) ; un système de gestion associé à la première instance peut être exécuté sur un noyau de calcul distinct du système sur une puce SoC et garantir ainsi la sécurité.

[0103] On peut également réaliser que la première et la seconde instance 410, 420 sur un système partagé. Également la première instance 410 est appliquée dans un premier processeur et la seconde instance 420 dans un second coprocesseur distinct. La com-

munication entre ces instances doit disposer d'une sécurité appropriée qui garantit l'intégrité ainsi que l'authenticité des informations et la complétude de toutes les informations transmises.

[0104] Le système 400 peut également être un système virtualisé 500 (voir par exemple [Fig.5]).

[0105] Selon le mode de réalisation représenté, le système 500 comprend deux plateformes de matériel 502a, 502b qui forment le plan matériel 502.

[0106] Les plateformes de matériel 502a, 502b comprennent, par exemple, des unités de matériel telles que, par exemple, un microcontrôleur, des ressources de matériel, garanties, des puces «hardware-routing ».

[0107] Le système 500 comprend un hyperviseur 506. L'hyperviseur 506 visualise les plateformes de matériel 502a, 502b. La première instance 410 est, par exemple, un premier domaine 508a fourni par l'hyperviseur et la seconde instance 420 est, par exemple, un autre domaine 508b fourni par l'hyperviseur 506.

[0108] Un domaine respectif désigne un certain domaine d'une fonctionnalité globale fournie par le système 500.

Revendications

[Revendication 1] Procédé implémenté par ordinateur pour surveiller un modèle fonctionnel (100) pour fournir des données pour au moins une fonction d'une machine (10) commandée par ordinateur, notamment d'un algorithme de reconnaissance d'image,
- le modèle (100) définissant au moins un résultat intermédiaire (Z_E1, Z_E2) en se fondant sur les données d'entrée (E) dans au moins une première étape de traitement (110, 120) et,
- le modèle fonctionnel, en se fondant sur le résultat intermédiaire (Z_E1, Z_E2) détermine, dans au moins une autre étape de traitement (120), une sortie (A) du modèle fonctionnel (100) et,
- le résultat intermédiaire (Z_E1, Z_E2) et la sortie (A) du modèle fonctionnel (100) sont fournis à un modèle de surveillance (1000) pour la reconnaissance d'anomalies et,
en se fondant sur le résultat intermédiaire (Z_E1, Z_E2) et la sortie (A) du modèle fonctionnel (100) on effectue une reconnaissance d'anomalies et,
- on effectue au moins une étape de validation d'une fonctionnalité du modèle fonctionnel (100) et/ou d'une fonctionnalité du modèle de surveillance (1000).

[Revendication 2] Procédé selon la revendication 1,
consistant à :
- déterminer une sortie de référence (R_A) fondée sur une entrée de référence (R_E) avec le modèle fonctionnel (100) et,
- vérifier la sortie de référence (R_A), notamment comparer la sortie de référence (R_A) à des données de vérité de terrain avec le modèle de surveillance (1000).

[Revendication 3] Procédé selon la revendication 2,
selon lequel
- on détermine une sortie de référence (R_A) en se fondant sur une entrée de référence (R_E) avec le modèle fonctionnel (100) et on compare périodiquement la sortie de référence (R_A) aux données de vérité de terrain avec le modèle de surveillance (1000).

[Revendication 4] Procédé selon l'une des revendications précédentes,
selon lequel le procédé consiste à :
- fournir une valeur de hachage (h) avec le modèle de surveillance (1000),

- signer la valeur de hachage avec le modèle fonctionnel (100),
- fournir une valeur de hachage signée (h'os) au modèle de surveillance (1000) et,
- vérifier la valeur de hachage signée (h'os) notamment comparer la valeur de hachage (h) avec la valeur de hachage (h') calculée à partir de la valeur de hachage signée (h'os), à l'aide du modèle de surveillance (1000).

[Revendication 5] Procédé selon la revendication 4,
selon lequel signer la valeur de hachage (h) consiste à :
- ajouter une signature (s) à la valeur de hachage dans la première étape de travail (110) du modèle fonctionnel (100) et,
- ajouter une autre signature (s) dans au moins une autre étape de traitement (120, 130) du modèle fonctionnel (100).

[Revendication 6] Procédé selon l'une des revendications précédentes,
selon lequel le procédé consiste à :
- surveiller la communication entre une première instance qui applique le modèle fonctionnel (100) et une autre instance qui applique le modèle de surveillance (1000).

[Revendication 7] Procédé selon l'une des revendications précédentes,
selon lequel
- en fonction d'un résultat de la reconnaissance d'anomalies et/ou en fonction d'un résultat de la surveillance de la communication, on applique au moins l'une des étapes suivantes consistant à :
a) vérifier le résultat de la comparaison de la sortie de référence (R_A) aux données de vérité de terrain avec le modèle de surveillance (1000),
b) vérifier le résultat de la comparaison de la valeur de hachage (h) avec la valeur de hachage (h') calculée à partir de la valeur de hachage signée (h'os) avec le modèle de surveillance (1000),
c) fournir un signal de commande pour commander la machine commandée par ordinateur, notamment au moins une partie de la machine commandée par ordinateur et/ou une fonction de la machine commandée par ordinateur,
d) faire passer la machine commandée par ordinateur, notamment au moins une partie de la machine commandée par ordinateur et/ou une fonction de la machine commandée par ordinateur dans un état défini,
e) faire passer la machine commandée par ordinateur, notamment une partie de la machine commandée par ordinateur et/ou la fonction de la machine commandée par ordinateur dans l'état défini en fonction d'un

résultat de a) et/ou b).

[Revendication 8] Système (400) de surveillance d'un modèle fonctionnel (100) pour fournir des données pour au moins une fonction (F) d'une machine commandée par ordinateur (10), notamment d'un algorithme de reconnaissance d'image,
- le système (400) est conçu pour effectuer avec un modèle de surveillance (1000) les étapes d'un procédé selon l'une des revendications 1 à 7,
- au moins les étapes du procédé qui sont exécutées avec le modèle de surveillance (1000) dans au moins une première instance (410) du système et le modèle fonctionnel (100) pour fournir des données pour au moins une fonction de la machine commandée par ordinateur, notamment de l'algorithme de reconnaissance d'images dans au moins une seconde instance (420) du système.

[Revendication 9] Système (400) selon la revendication 8,
selon lequel
au moins la première instance (410) répond à un niveau d'intégrité de sécurité, par exemple, le niveau d'intégrité de sécurité automobile, niveau ASIL, selon ISO 26262.

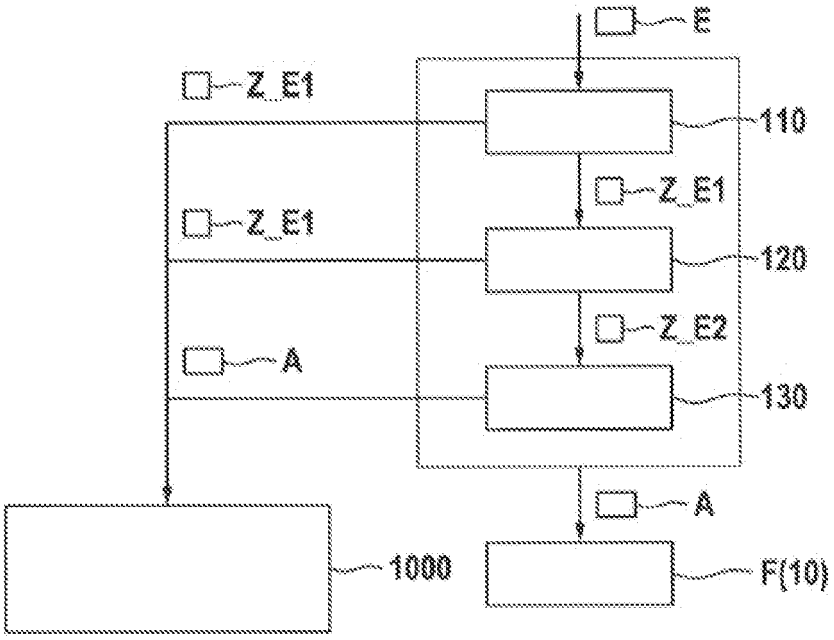
[Revendication 10] Système (400) selon l'une des revendications 8 ou 9,
selon lequel
la première et la seconde instances (410, 420) sont réalisées respectivement comme une instance d'un système sur une puce SoC commun, un système de fonctionnement associé à la première instance (410) étant exécuté sur un noyau de recherche distinct du système sur une puce SoC.

[Revendication 11] Système (400, 500) selon l'une des revendications 8 à 10,
selon lequel
un hyperviseur virtualise un plan de circuit et la première instance est un premier domaine fourni par l'hyperviseur et la seconde instance est un autre domaine fourni par l'hyperviseur.

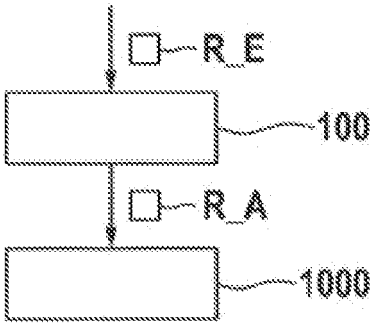
[Revendication 12] Application du procédé selon les formes de réalisation et/ou du système selon les formes de réalisation, à une machine commandée par ordinateur, par exemple, un système E/E d'un véhicule automobile, notamment pour des fonctions de conduite autonome, de conduite partiellement autonome et/ou de fonction d'assistance de conduite, un robot, un appareil électroménager, un outil électrique, une machine de fabrication, un dispositif d'inspection optique automatique ou d'un

système d'accès,
le modèle fonctionnel fournissant, dans la machine commandée par ordinateur, en se fondant sur les données d'entrée, notamment des données d'images numériques, d'un capteur d'images, notamment d'un capteur vidéo, radar, lidar, d'ultrasons, de données d'images mobiles ou d'images thermiques, des données pour au moins une fonction de la machine commandée par ordinateur et,
en se fondant sur les données, fournir au moins un signal de commande pour exécuter la fonction de la machine commandée par ordinateur, et
- le modèle fonctionnel étant surveillé selon le procédé et/ou
la machine commandée par ordinateur notamment une partie de la machine commandée par ordinateur et/ou
- une fonction de la machine commandée par ordinateur étant mise dans un état défini.

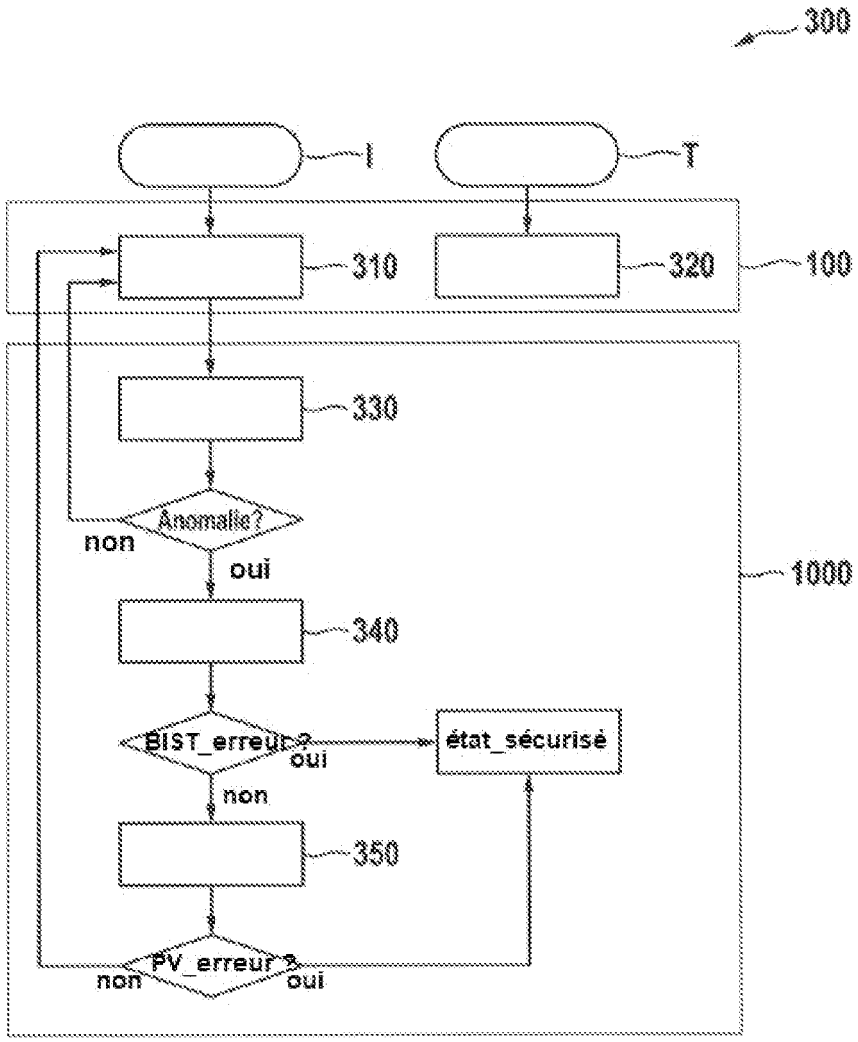
[Fig. 1]



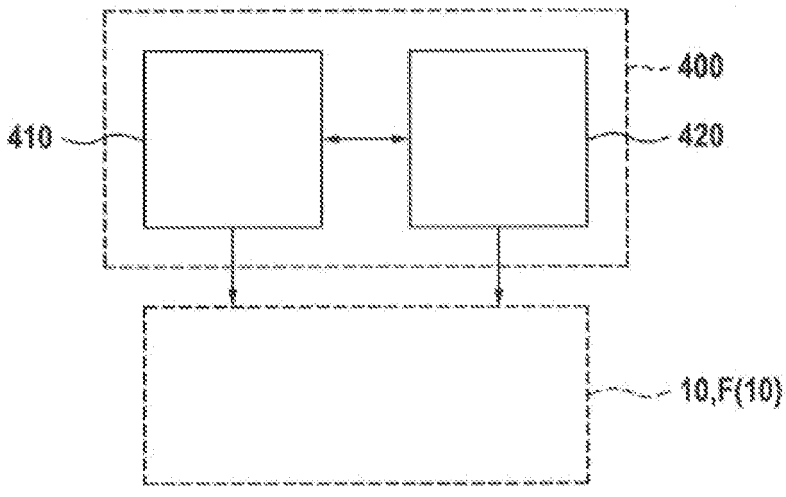
[Fig. 2]



[Fig. 3]


300
I
T
310
320
100
330
Anomalie?
non
oui
1000
340
BIST_erreur ?
oui
état_sécurisé
non
350
PV_erreur ?
non
oui

[Fig. 4]


400
410
420
10,F(10)

[Fig. 5]