(19) CONFÉDÉRATION SUISSE
INSTITUT FÉDÉRAL DE LA PROPRIÉTÉ INTELLECTUELLE

(11) CH 712 947 A2

(51) Int. Cl.: H04L 9/10 (2006.01)
H04L 12/22 (2006.01)

Demande de brevet pour la Suisse et le Liechtenstein
Traité sur les brevets, du 22 décembre 1978, entre la Suisse et le Liechtenstein

(12) DEMANDE DE BREVET

(21) Numéro de la demande: 01241/16

(22) Date de dépôt: 23.09.2016

(43) Demande publiée: 29.03.2018

(71) Requérant:
Pierino Vidoni, Rue de Büren 84
2504 Bienne (CH)
Marta Dorothea Altenkamp, Rue du Village 20B
1273 Arzier-Le Muids (CH)

(72) Inventeur(s):
Pierino Vidoni, 2504 Bienne (CH)
Marta Dorothea Altenkamp, 1273 Arzier-Le Muids (CH)

(74) Mandataire:
LEMAN CONSULTING S.A., Chemin de Précossy 31
1260 Nyon (CH)

(54) Dispositif et méthode de sécurisation de transmission de données entre un émetteur et un récepteur.

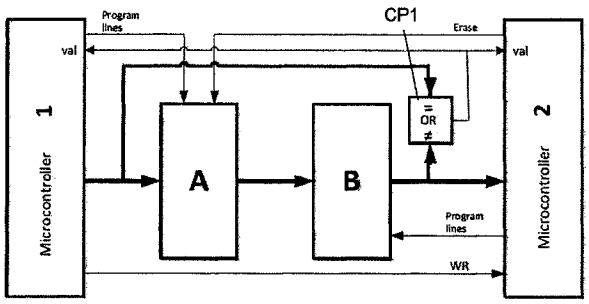
(57) La présente invention concerne un dispositif de sécurisation de transmission de données entre un émetteur et un récepteur, ce dispositif comprenant un circuit d'interface connecté entre l'émetteur et le récepteur. Ce dispositif est caractérisé en ce que le circuit d'interface comprend:

un premier circuit d'entrée (A) agencé pour recevoir des données à transmettre, ce premier circuit d'entrée (A) comprenant de la logique programmable pour transformer lesdites données à transmettre, cette logique programmable étant construite dans le premier circuit d'entrée par un premier microcontrôleur (1);

un premier circuit de sortie (B) agencé pour recevoir les données transformées par le premier circuit d'entrée (A), ce premier circuit de sortie (B) comprenant de la logique programmable pour retransformer lesdites données transformées, cette logique programmable étant construite dans le premier circuit de sortie (B) par un deuxième microcontrôleur (2); et

un premier comparateur (CP1) agencé pour comparer lesdites données retransformées par le premier circuit de sortie (B) et les données à transmettre, la logique programmable du premier circuit d'entrée (A) étant inverse et complémentaire à la logique programmable du premier circuit de sortie (B).

L'invention concerne également une méthode mise en œuvre par un dispositif tel que décrit ci-dessus.



Description

DOMAINE TECHNIQUE

[0001] La présente invention concerne un dispositif de sécurisation de transmission de données entre un émetteur et un récepteur. Elle concerne également une méthode destinée à sécuriser la transmission de données entre un émetteur et un récepteur.

[0002] Plus précisément, cette invention concerne la protection de l'interconnexion entre un émetteur et un récepteur, cette protection étant destinée à éviter toute lecture ou modification par une personne non autorisée, de données échangées entre un émetteur et un récepteur.

TECHNIQUE ANTERIEURE

[0003] Dans les dispositifs conventionnels permettant la transmission ou l'échange de données entre un émetteur et un récepteur, les données peuvent être chiffrées avec des algorithmes plus ou moins puissants pour garantir le secret de l'information lorsque c'est nécessaire. Les données sont chiffrées au moyen d'un algorithme utilisant par exemple une clé jetable ou une paire de clés publique/privée selon le système et le degré de protection souhaité du message.

[0004] Il n'existe pas de système matériel ou hardware empêchant l'accès au dispositif de transmission ou à un objet connecté.

[0005] Dans les dispositifs de transmission standards, les données sont simplement transmises du récepteur à une logique d'interface de sortie ou de la logique d'interface d'entrée à l'émetteur sans système de contrôle.

[0006] Dans ces dispositifs de l'art antérieur, l'identification et l'authentification du récepteur est faite uniquement par un protocole défini de façon logicielle, qui peut être plus ou moins facile à attaquer. A partir du moment où l'adresse du récepteur est connue, on peut continuellement lui adresser des messages, allant parfois jusqu'à le saturer.

[0007] Dans les dispositifs de l'art antérieur, il n'y a pas de protection physique ou hardware dédiée à l'interconnexion elle-même. Toute la protection, y compris l'identification et l'authentification, est assurée par le software.

EXPOSÉ DE L'INVENTION

[0008] Les problèmes auxquels les dispositifs de l'art antérieur sont confrontés sont notamment les suivants: une fois que l'on dispose de l'adresse du récepteur, par exemple l'adresse IP dans le cas d'une connexion par Internet, on peut envoyer des messages et remplir ou saturer le récepteur de données de messages spam ou autres. Si l'on ajoute une couche matérielle dédiée uniquement à l'interconnexion, ceci ne peut arriver.

[0009] Si l'on ajoute par exemple dans un modem une couche matérielle qui autorise ou pas la connexion, une attaque du type «refus de service» (déniai of service) ne peut arriver. En effet, le modem lui-même coupe la connexion si cette couche matérielle est ajoutée.

[0010] Cette couche matérielle est particulièrement intéressante si l'on veut s'adresser à un appareil qui doit avoir un très haut degré de protection, en particulier dans le cas où l'on souhaite isoler complètement une application connectée de façon à ne pas pouvoir l'attaquer par attaque software.

[0011] Le meilleur moyen d'isoler un appareil si l'on veut être sûr qu'il ne subisse pas d'attaque de l'extérieur serait de couper la connexion lorsqu'on ne l'utilise pas. A ce moment là, aucun type d'attaque ne peut se produire sur le système ainsi déconnecté.

[0012] La présente invention se propose de réaliser un dispositif dans lequel une protection efficace est apportée, en particulier de façon à éviter que ce dispositif puisse être attaqué et qu'il délivre des informations confidentielles.

[0013] Le but de l'invention est atteint par un dispositif de sécurisation de transmission de données entre un émetteur et un récepteur, comprenant un circuit d'interface connecté entre l'émetteur et le récepteur, caractérisé en ce que le circuit d'interface comprend:
un premier circuit d'entrée agencé pour recevoir des données à transmettre, ce premier circuit d'entrée comprenant de la logique programmable pour transformer lesdites données à transmettre, cette logique programmable étant construite dans le premier circuit d'entrée par un premier microcontrôleur;
un premier circuit de sortie agencé pour recevoir les données transformées par le premier circuit d'entrée, ce premier circuit de sortie comprenant de la logique programmable pour retransformer lesdites données transformées, cette logique programmable étant construite dans le premier circuit de sortie par un deuxième microcontrôleur; et
un premier comparateur agencé pour comparer lesdites données retransformées par le premier circuit de sortie et les données à transmettre, la logique programmable du premier circuit d'entrée étant inverse et complémentaire à la logique programmable du premier circuit de sortie.

[0014] Les buts de l'invention sont également atteints par une méthode de sécurisation de transmission de données entre un émetteur et un récepteur au moyen d'un circuit d'interface connecté entre l'émetteur et le récepteur, ce circuit d'interface comprenant:

un premier circuit d'entrée agencé pour recevoir des données à transmettre, ce premier circuit d'entrée comprenant de la logique programmable pour transformer lesdites données à transmettre, cette logique programmable étant construite dans le premier circuit d'entrée au moyen d'un premier microcontrôleur;
un premier circuit de sortie agencé pour recevoir les données transformées par le premier circuit d'entrée, ce premier circuit de sortie comprenant de la logique programmable pour retransformer lesdites données transformées, cette logique programmable étant construite dans le premier circuit de sortie au moyen d'un deuxième microcontrôleur; et
un premier comparateur agencé pour comparer lesdites données retransformées par le premier circuit de sortie avec les données à transmettre, cette méthode étant caractérisée en ce qu'elle comprend les étapes suivantes:
dans ledit premier circuit d'entrée, construction par ledit premier microcontrôleur, d'une logique programmable au moyen d'informations de programmation;
construction par ledit deuxième microcontrôleur, d'une logique programmable dans ledit premier circuit de sortie, ladite logique programmable dans ledit premier circuit d'entrée et dans ledit premier circuit de sortie étant inverses et complémentaires;
transmission de données de l'émetteur au premier circuit d'entrée, ce premier circuit d'entrée transformant les données d'une manière dépendant de la logique programmable du premier circuit d'entrée;
transmission desdites données transformées au premier circuit de sortie et retransformation des données d'une manière dépendant de la logique programmable du premier circuit de sortie;
comparaison des données de l'émetteur avec les données retransformées et activation d'une contremesure si la comparaison des données de l'émetteur avec les données retransformées indique une différence.

[0015] De façon plus générale, le but de la présente invention est d'ajouter une couche matérielle pouvant être appliquée à tout système de transfert d'informations. Cette couche s'occupe uniquement de la protection de l'accès lui-même.

[0016] Il s'agit donc d'une couche matérielle ou hardware, indépendante des couches logicielles, pouvant être appliquée en plus sur n'importe quel système de transmission de données dans lequel il y a identification de l'émetteur et du récepteur. Cette couche matérielle est indépendante de la protection logicielle. Elle peut être appliquée dans un système dans lequel les données sont chiffrées ou pas.

[0017] Selon les objets connectés, la protection apportée par le dispositif de l'invention est très importante, spécialement dans certains domaines où les données peuvent être stratégiques.

[0018] Ce dispositif est spécialement pratique s'il est intégré à un modem de communication, pour isoler tout ou partie d'un système. L'invention trouve un intérêt partout où il faut accéder à des systèmes de télécommande, bidirectionnels ou unidirectionnels importants. Comme ce dispositif est un élément indépendant, il peut être ajouté dans toute sorte d'objets existants, selon leur importance.

[0019] Ceci est particulièrement intéressant dans le cadre de l'Internet des objets (Internet of Things – IOT), une partie des objets connectés nécessitant en effet un haut niveau de protection. L'invention est intéressante dans le cas de communications avec des commandes, des voitures, des télécommandes de paramètres ou des interfaces permettant des actions à distance sur des bâtiments, et plus généralement dans les cas où il y a des effets qui pourraient être dommageables si l'on arrive à se connecter et à agir sur ces objets.

[0020] Le niveau de protection atteint par le dispositif de l'invention étant suffisamment élevé, même en utilisant seulement cette protection de connexion, il n'est pas nécessaire, dans un grand nombre de cas, de chiffrer les données, en particulier dans des application concernant NOT. Un tel chiffrement est toutefois possible sans changer le fonctionnement du dispositif de l'invention. Comme le dispositif de l'invention est indépendant, ajouté aux applications, le trafic normal des données, chiffrées ou pas, peut passer à très haute vitesse, sans retard une fois que la connexion est autorisée.

[0021] La présente invention permet de créer un système en logique programmable dont le circuit est vide au début de la transmission, cette logique étant chargée au début de la transmission. Cette logique est maintenue dans une mémoire temporaire modifiable de type RAM.

[0022] Ce circuit de protection est intercalé entre les circuits émetteurs/récepteurs et les circuits d'interface entrée/sortie. Ce circuit est piloté par deux microcontrôleurs à savoir un microcontrôleur d'entrée et un microcontrôleur de sortie.

DESCRIPTION SOMMAIRE DES DESSINS

[0023] La présente invention et ses avantages seront mieux compris en référence aux figures annexées et à la description détaillée de modes de réalisation particuliers, dans lesquelles:

la fig. 1	est un schéma d'un dispositif selon l'invention, selon un mode de réalisation unidirectionnel;
la fig. 2	représente un dispositif selon l'invention, selon un mode de réalisation bidirectionnel;
les fig. 3a à 3d	illustrent des types de cellules qui peuvent être utilisées dans un dispositif selon l'invention;
la fig. 4	représente un circuit utilisé dans un dispositif selon l'invention pour transformer des données; et

la fig. 5 représente un exemple de message pour la construction de la logique programmable.

MANIERES DE REALISER L'INVENTION

[0024] La fig. 1 illustre un mode de réalisation d'un dispositif selon l'invention, dans un mode de réalisation unidirectionnel. En référence à cette figure, le dispositif selon l'invention comporte un premier circuit d'entrée A et un premier circuit de sortie B, ces deux circuits comportant une logique programmable, ces circuits étant réversibles et les logiques programmables étant complémentaires. Ces circuits peuvent comporter des cellules réversibles, qui peuvent facilement être implémentées en logique programmable. Ces circuits peuvent utiliser des cellules telles que des cellules de type NOT, des cellules de Feynmann, de Toffoly ou de Fredkin notamment. Des cellules de ce type sont illustrées par les fig. 3a à 3d.

[0025] Le dispositif de l'invention comporte en outre deux microcontrôleurs, l'un des microcontrôleurs 1 étant connecté au premier circuit d'interface A d'une part et à un émetteur (non représenté) d'autre part. L'autre microcontrôleur 2 est connecté au premier circuit de sortie B d'une part et à un récepteur (non représenté) d'autre part. A l'enclenchement du dispositif, le deuxième microcontrôleur 2 va télécharger un contenu qu'il possède en mémoire fixe, flash, EEPROM et construire la logique du premier circuit de sortie B.

[0026] Dans l'état actuel, le circuit logique du premier circuit d'entrée A est totalement vide et peut être mis en haute impédance puisqu'aucune donnée permettant la création d'un circuit n'ont été téléchargées depuis le premier microcontrôleur 1 qui pilote ce premier circuit d'entrée A.

[0027] Le premier microcontrôleur 1 est relié aux interfaces émetteur/récepteur qui peuvent utiliser une grande variété de protocoles selon les tâches à accomplir. Ces protocoles peuvent être de type Wifi, Bluetooth, Zigbee, Ethernet, protocoles IOT ou autres ...

[0028] Dans notre cas, il n'est pas possible de transmettre des données aux interfaces avant d'avoir transmis un bloc de données nécessaire à la création de la logique du premier circuit d'entrée A au moyen d'un protocole sécurisé.

[0029] Pour la création de cette logique du premier circuit d'entrée A, le premier microcontrôleur 1 reçoit un ordre de la part de l'émetteur. Si cet ordre est reçu au bon format, le premier microcontrôleur 1 va implémenter une logique programmable dans le premier circuit d'entrée A. Cette logique programmable doit être complémentaire à la logique implémentée dans le premier circuit de sortie B qui a été implémentée lors la mise sous tension du dispositif.

[0030] Seules les personnes connaissant la logique implémentée dans le premier circuit de sortie B peuvent implémenter la logique complémentaire dans le premier circuit d'entrée A et ainsi communiquer avec le premier circuit de sortie B. Cette logique programmable correspond a une clé qui peut avoir plusieurs milliers de bits qu'il est nécessaire d'envoyer pour la création du circuit logique dans le premier circuit d'entrée A.

[0031] Les interfaces d'entrée/sortie sont alors prêtes à recevoir des ordres de transfert de données ou des messages contenant des données ou des informations. Dans le cas de la fig. 1, seules les lignes de sortie sont activables pour donner des commandes.

[0032] Pour transmettre des données aux interfaces depuis un récepteur, il y donc toujours trois phases:

Phase 1: Réception de données de construction de la logique programmable

Phase 2: Transmission de paquet de données si la construction est valide

Phase 3: Effacement de la logique en fin de transmission

[0033] Selon une variante, il est possible d'ajouter une commande pour le deuxième microcontrôleur 2 lui disant qu'après un certain temps d'inactivité de transmission, la phase 3 d'effacement de la logique est automatiquement déclenchée.

[0034] Lors de la première phase, à savoir la phase de réception de données permettant la construction de la logique programmable, plusieurs variantes sont possibles si les données reçues par le premier microcontrôleur 1 ne sont pas conformes. Selon une première variante, le premier microcontrôleur 1 ne répond pas et la logique n'est pas construite dans le premier circuit d'entrée A. Selon une deuxième variante, le premier microcontrôleur 1 répond selon un protocole déterminé. Ce protocole pourrait par exemple être:

- blocage après 3 essais, déblocage pas un système similaire à ce qui est utilisé en téléphonie portable, du type code PUK; ou
- blocage après un nombre d'essais défini, par exemple 3, puis admission de nouveaux essais selon une fenêtre temporelle, incrémentale ou pas, par exemple après 1 seconde, 10s, 1 minute, 10 min, etc.

[0035] Il est clair que de nombreux autres protocoles sont envisageables.

[0036] La fig. 5 illustre un exemple d'ordre envoyé au premier circuit d'entrée A pour construire la logique. Cet exemple est décrit en détail ci-dessous.

[0037] Envoi d'un caractère de début STX, puis d'une commande CMD, puis d'un nombre d'octets NB indiquant la taille ou la longueur du message. L'ordre se poursuit par un certain nombre de données Data 0, Data 1, ... correspondant à la «partie utile» du message, soit par exemple la partie permettant de construire la logique à implémenter dans le circuit d'entrée. L'ordre se poursuit par un ou plusieurs octets de contrôle CHKSO, CHKS1, ... et se termine par un caractère de fin ESC. Ceci n'est qu'un exemple de protocole d'envoi pour les commandes au premier microcontrôleur 1. Le protocole est structuré selon les applications.

[0038] Si le message reçu pour construire la logique est conforme, le circuit logique correspondant à ces données est chargé et fonctionnel.

[0039] Le nombre de commandes à disposition est variable. Il est possible d'en ajouter selon les besoins. Chaque ordre CMD correspond à un certain nombre d'octets qui peuvent être différents d'un ordre à un autre. En effet, certains ordres sont très courts alors que d'autres, comme par exemple la construction de la logique, sont beaucoup plus longs.

[0040] Dans la phase 2, à savoir la phase de transmission de paquet de données, chaque paquet provenant de l'émetteur est transféré au premier circuit d'entrée A. Les paquets sont ensuite transformés en fonction de la logique programmable implémentée dans ce premier circuit d'entrée A. Les données transformées sont ensuite transmises au premier circuit de sortie B dans lequel les données sont retransformées en fonction de la logique programmable implémentée dans ce premier circuit de sortie B.

[0041] Chaque paquet de bits 1 .. n, entré dans le premier circuit d'entrée A est comparé en sortie avec le résultat de la retransformation du paquet correspondant par le premier circuit de sortie B. Comme le premier circuit de sortie B effectue l'opération inverse de celle réalisée dans le premier circuit d'entrée A, le résultat de la comparaison indique que les valeurs sont égales si la logique du premier circuit de sortie est inverse et complémentaire à la logique du premier circuit d'entrée A ou en d'autres termes, si les circuits sont correctement initialisée.

[0042] Le circuit d'interface selon l'invention comporte en outre une ligne d'écriture (WR). C'est cette ligne qui informe le deuxième microcontrôleur 2, pilotant le premier circuit de sortie B, que le premier circuit d'entrée A a reçu des données du premier microcontrôleur 1 et que le deuxième microcontrôleur 2 peut lire le résultat du comparateur indiquant si la transmission est valide ou pas.

[0043] Selon une variante, si la transmission n'est pas valide, le deuxième microcontrôleur 2 peut réinitialiser (effacer) la logique dans les premiers circuits d'entrée et/ou de sortie.

[0044] De nombreuses variantes peuvent être utilisées pour la logique programmable. En effet, tout système de logique réversible symétrique peut être utilisé pour créer un circuit logique dans les circuits A et B. Les cellules telles qu'illustrées par la fig. 3 sont faciles à implémenter en logique programmable et permettent de créer des réseaux avec des lignes permutées, inversées ou non, suivant la valeur des données. Si, à la place de cellules, on implémente des émetteurs-récepteurs asynchrones universels (Universal Asynchronous Receiver Transmitter – UART) spéciaux avec nombre de bits variables par exemple et que de part et d'autre, on recopie le même circuit, cela fonctionne.

[0045] Dans l'exemple illustré, les circuits logiques sont représentés comme utilisant des cellules telles qu'une porte NOT, une cellule de Feynmann, de Toffoli et de Fredkin (voir fig. 3a à 3d pour les symboles utilisés).

[0046] Le comparateur CP1 est prévu pour comparer les données introduites dans le premier circuit d'entrée A et les données retransformées par le circuit de sortie B, cette comparaison induisant un temps de retard très faible. La comparaison donne uniquement la valeur «juste» ou «faux». Dans le cas où la comparaison indique que les données comparées sont différentes, la communication est arrêtée et la logique contenue dans les circuits d'entrée et/ou de sortie est effacée. D'autres contremesures peuvent bien entendu être implémentées. Il est possible de comptabiliser le nombre de tentatives infructueuses pour statistiques si l'on veut, pour connaître les tentatives d'accès non abouties.

[0047] Les fig. 3a à 3d illustrent quatre exemples de cellules réversibles utilisables dans la présente invention. Ces cellules sont une cellule NOT, une cellule de Feynmann, une cellule de Toffoli et une cellule de Fredkin.

[0048] La cellule NOT (fig. 3a) est l'exemple le plus simple de cellule réversible, elle consiste simplement en un inverseur logique.

[0049] La cellule CN, CONTROLLED NOT (fig. 3b) ou cellule de Feynmann, consiste en une cellule NOT dont l'inverseur logique est commandé, la façon la plus simple de l'implémenter étant avec un simple circuit XOR.

[0050] La cellule CCN, CONTROLLED CONTROLLED NOT (fig. 3c) ou cellule de Toffoli, consiste en une cellule NOT dont l'inverseur logique est commandé par le résultat du ET de deux commandes. Il faut que les lignes C1 et C2 soient à 1 pour que le circuit active le NOT, sinon, la sortie est directe. La façon la plus simple de l'implémenter est avec un simple circuit XOR dont une des branches est connecté sur un circuit AND.

[0051] La cellule de Fredkin (3d), permet selon l'état de la ligne de commande C, d'inverser ou pas les lignes A et B. Cette cellule utilise un peu plus de place en logique programmable que les circuits très simples précédents. Elle est aussi appelée SWAP ou CSWAP vu qu'elle permet de permuter des lignes.

[0052] La fi. 4 illustre un exemple d'implémentation de cellules dans les circuits d'entrée A et de sortie B. Cette fig. 4 illustre également un exemple de remplissage partiel sur quelques bits.

[0053] Les données reçues à l'entrée du premier circuit d'entrée A s'inversent, croisant des bits ou inversent des résultats de bits selon les valeurs. C'est comme si les valeurs étaient cryptées par le circuit d'entrée A sans attente, avec uniquement le temps de transition à travers ce circuit suivi du délai de transition dans le circuit de sortie B.

[0054] La comparaison des données initiales avec les données retransformées, obtenues après le passage dans les premiers circuits d'entrée et de sortie, valide la transmission lors de l'impulsion sur la ligne WR (write).

[0055] Si le signal n'est pas validé, on efface totalement le contenu du circuit d'entrée A, ce qui coupe toute possibilité de transmission erronée et l'émetteur doit réinitialiser le circuit.

REALISATION D'UN SYSTEME BIDIRECTIONNEL

[0056] La fig. 2 illustre un mode de réalisation d'un système bidirectionnel.

[0057] Dans l'explication du fonctionnement d'un dispositif de construction/destruction d'accès donnée ci-dessus, le système était unidirectionnel. En pratique, on utilise plutôt des systèmes bidirectionnels entrée/sortie. Pour ce faire, un système bidirectionnel tel qu'illustré par la fig. 2 comporte un deuxième circuit d'entrée C et un deuxième circuit de sortie D, le deuxième circuit d'entrée C étant piloté par le deuxième microcontrôleur 2 et le deuxième circuit de sortie D étant piloté par le premier microcontrôleur 1. Ainsi, le dispositif selon l'invention comporte un «ensemble de communication» pour chaque sens.

[0058] Le premier microcontrôleur 1 dans ce cas a deux circuits à initialiser et à programmer selon les données reçues. Il doit en effet gérer la logique programmable du premier circuit d'entrée A et du deuxième circuit de sortie D. Le deuxième microcontrôleur 2 charge le premier circuit de sortie B et le deuxième circuit d'entrée C avec ses données internes lors de la mise sous tension.

[0059] Chaque ensemble de communication formé d'un circuit d'entrée et du circuit de sortie correspondant a un comparateur CP1, CP2 et une ligne d'écriture (write) ainsi que des moyens d'effacement des logiques programmables.

[0060] Le système selon la présente invention a les avantages suivants:

- Très haute fiabilité, impossible de contacter la cible et y transmettre ou y lire des données sans que le paquet de bits nécessaire à l'initialisation soit exact. Si un seul bit est différent, le circuit d'entrée ne fonctionne pas.
- Pas de retard de calculs dans la transmission (très hautes vitesses possible), juste un temps de transition dans la logique. Ceci est un grand avantage pour des systèmes, par exemple pour l'IOT qui aurait une très grande vitesse de l'information à passer sans les crypter.
- faible consommation
- Système d'audit (feed-back à un centre) possible indiquant par exemple le nombre d'accès réussis, le nombre d'accès refusés, etc.

[0061] Il est à noter que ce système peut être utilisé de façon indépendante d'un système cryptographique. A ce titre, il est possible d'ajouter entre l'émetteur et le premier microcontrôleur 1, de même qu'entre le deuxième microcontrôleur 2 et le récepteur, un module cryptographique en charge de chiffrer/déchiffrer des données.

[0062] Selon différentes variantes, il est possible d'introduire un compteur en charge de compter des événements déterminés, par exemple le nombre d'accès réussis, le nombre de tentatives d'accès ayant échoué, etc. Le deuxième microcontrôleur 2 peut initialiser toujours la même logique programmable ou au contraire, utiliser une logique différente, selon une règle prédéterminée. Il est également possible de prévoir une ligne de communication entre les deux microcontrôleurs, ce qui permet de reconfigurer le système et d'échanger la configuration entre les deux microcontrôleurs.

Revendications

1. Dispositif de sécurisation de transmission de données entre un émetteur et un récepteur, comprenant un circuit d'interface connecté entre l'émetteur et le récepteur, caractérisé en ce que le circuit d'interface comprend:
   un premier circuit d'entrée (A) agencé pour recevoir des données à transmettre, ce premier circuit d'entrée (A) comprenant de la logique programmable pour transformer lesdites données à transmettre, cette logique programmable étant construite dans le premier circuit d'entrée par un premier microcontrôleur (1);
   un premier circuit de sortie (B) agencé pour recevoir les données transformées par le premier circuit d'entrée (A), ce premier circuit de sortie (B) comprenant de la logique programmable pour retransformer lesdites données transformées, cette logique programmable étant construite dans le premier circuit de sortie (B) par un deuxième microcontrôleur (2); et
   un premier comparateur (CP1) agencé pour comparer lesdites données retransformées par le premier circuit de sortie (B) et les données à transmettre, la logique programmable du premier circuit d'entrée (A) étant inverse et complémentaire à la logique programmable du premier circuit de sortie (B).
2. Dispositif selon la revendication 1, caractérisé en ce que le premier microcontrôleur (1) est configuré pour recevoir des informations de programmation permettant à ce premier microcontrôleur de construire la logique programmable du premier circuit d'entrée (A).

3. Dispositif selon la revendication 2, caractérisé en ce que le premier microcontrôleur (1) est configuré pour recevoir les informations de programmation par voie sécurisée.

4. Dispositif selon l'une quelconque des revendications précédentes, caractérisé en ce que ledit circuit d'interface comporte en outre un deuxième circuit d'entrée (C) et un deuxième circuit de sortie (D),
ledit deuxième circuit d'entrée (C) étant agencé pour recevoir des données à transmettre, ce deuxième circuit d'entrée (C) comprenant de la logique programmable pour transformer lesdites données à transmettre, cette logique programmable étant construite dans le deuxième circuit d'entrée (C) au moyen dudit deuxième microcontrôleur (2);
ledit deuxième circuit de sortie (D) étant agencé pour recevoir les données transformées par le deuxième circuit d'entrée (C), ce deuxième circuit de sortie (D) comprenant de la logique programmable pour retransformer lesdites données transformées, cette logique programmable étant construite dans le deuxième circuit de sortie (D) au moyen dudit premier microcontrôleur (1); et
un deuxième comparateur (CP2) agencé pour comparer lesdites données retransformées par le deuxième circuit de sortie (D) avec les données à transmettre.

5. Dispositif selon l'une quelconque des revendications précédentes, caractérisé en ce que le premier microcontrôleur (1) est en outre configuré pour assurer que la logique du circuit d'entrée (A, C) soit inverse et complémentaire à la logique du circuit de sortie (B, D) correspondant de sorte que les données à transmettre soient égales aux données retransformées.

6. Dispositif selon l'une quelconque des revendications précédentes, caractérisé en ce qu'il comporte des moyens d'inactivation de la logique programmable d'au moins l'un desdits circuits d'entrée (A, C) ou de sortie (B, D).

7. Dispositif selon l'une quelconque des revendications précédentes, caractérisé en ce qu'il comporte une ligne d'écriture (WR) agencée pour permettra à l'un desdits microcontrôleurs (1, 2) d'indiquer à l'autre microcontrôleur qu'une donnée a été transmise.

8. Méthode de sécurisation de transmission de données entre un émetteur et un récepteur au moyen d'un circuit d'interface connecté entre l'émetteur et le récepteur, ce circuit d'interface comprenant:
un premier circuit d'entrée (A) agencé pour recevoir des données à transmettre, ce premier circuit d'entrée (A) comprenant de la logique programmable pour transformer lesdites données à transmettre, cette logique programmable étant construite dans le premier circuit d'entrée (A) au moyen d'un premier microcontrôleur (1);
un premier circuit de sortie (B) agencé pour recevoir les données transformées par le premier circuit d'entrée (A), ce premier circuit de sortie (B) comprenant de la logique programmable pour retransformer lesdites données transformées, cette logique programmable étant construite dans le premier circuit de sortie (B) au moyen d'un deuxième microcontrôleur (2); et
un premier comparateur (CP1) agencé pour comparer lesdites données retransformées par le premier circuit de sortie (B) avec les données à transmettre, cette méthode étant caractérisée en ce qu'elle comprend les étapes suivantes:
dans ledit premier circuit d'entrée (A), construction par ledit premier microcontrôleur (1), d'une logique programmable au moyen d'informations de programmation;
construction par ledit deuxième microcontrôleur (2), d'une logique programmable dans ledit premier circuit de sortie (B), ladite logique programmable dans ledit premier circuit d'entrée (A) et dans ledit premier circuit de sortie (B) étant inverses et complémentaires;
transmission de données de l'émetteur au premier circuit d'entrée (A), ce premier circuit d'entrée (A) transformant les données d'une manière dépendant de la logique programmable du premier circuit d'entrée (A);
transmission desdites données transformées au premier circuit de sortie (B) et retransformation des données dune manière dépendant de la logique programmable du premier circuit de sortie (B);
comparaison des données de l'émetteur avec les données retransformées et activation d'une contremesure si la comparaison des données de l'émetteur avec les données retransformées indique une différence.

9. Méthode selon la revendication 8, caractérisé en ce que ladite contremesure comporte une étape d'arrêt de la transmission de données.

10. Méthode selon la revendication 8, caractérisé en ce que ladite contremesure comporte une étape de modification d'au moins l'une des logiques programmable des circuits d'entrée (A) et de sortie (B).

11. Méthode selon la revendication 8, caractérisé en ce que le microcontrôleur (1) recevant des données de l'émetteur envoie un signal au microcontrôleur (2) connecté au récepteur au moyen d'une ligne d'écriture (WR) lorsqu'une donnée a été transmise au premier circuit d'entrée (A).

12. Méthode selon la revendication 8, dans lequel des données sont échangées au moyen d'un dispositif selon la revendication 4, caractérisé en ce que des données sont émises par un émetteur connecté au premier dispositif d'entrée (A) et des données sont également émises par un émetteur connecté au deuxième dispositif d'entrée (C).

13. Méthode selon la revendication 8, caractérisée en ce qu'au moins l'une des logiques programmables des circuits d'entée (A) et de sortie (B) sont altérées à la fin de ladite transmission de données.

14. Méthode selon la revendication 13, caractérisée en ce que l'altération de la logique programmable comporte une étape d'effacement de cette logique programmable.

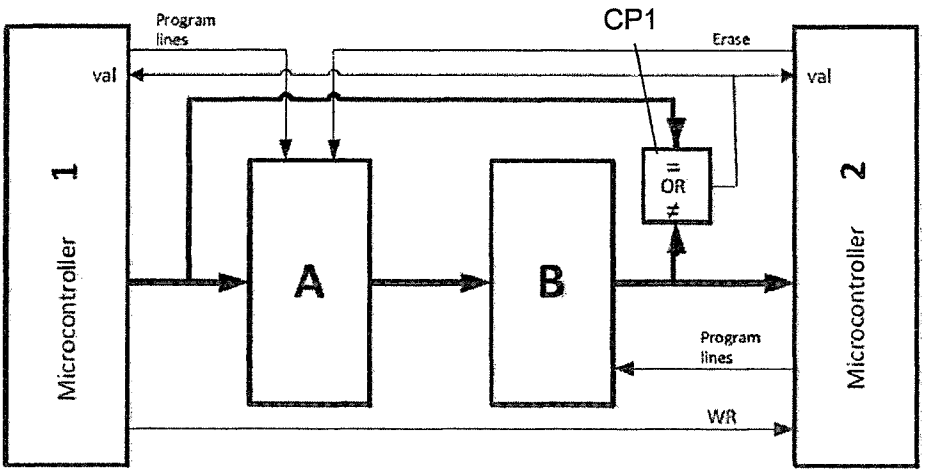

Program lines
CP1
Erase
val
val
1
Microcontroller
A
B
=
OR
≠
2
Microcontroller
Program lines
WR

Fig. 1

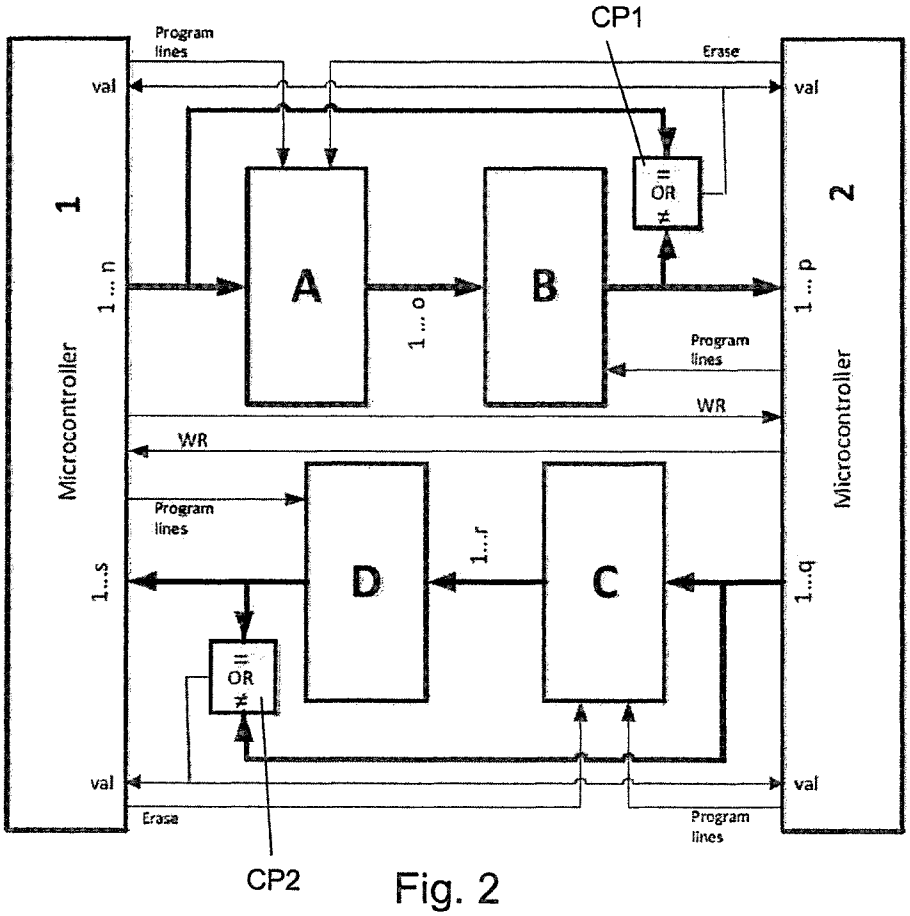

CP1
Program lines
Erase
val
val
1
Microcontroller
1 ... n
A
1 ... o
B
=
OR
≠
2
Microcontroller
1 ... p
Program lines
WR
WR
Program lines
D
1...r
C
1...s
1...q
=
OR
≠
val
val
Erase
Program lines
CP2

Fig. 2

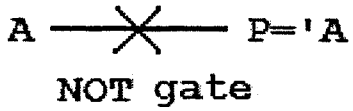


Fig. 3a

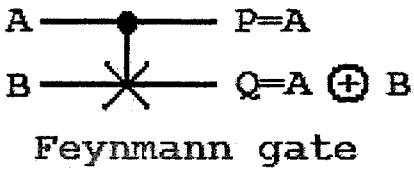


Fig. 3b

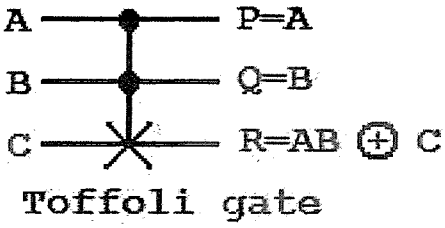


Fig. 3c

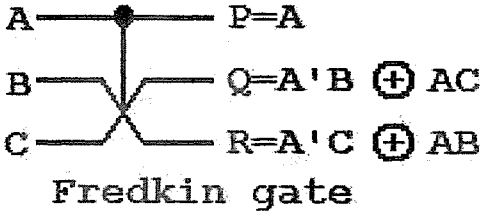


Fig. 3d

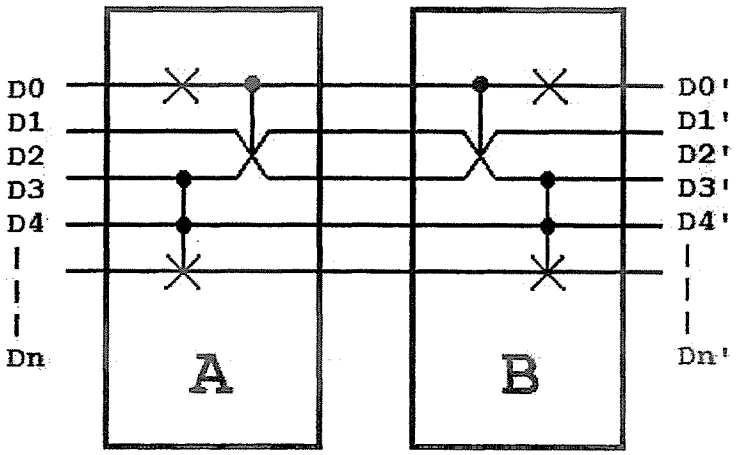


Fig. 4

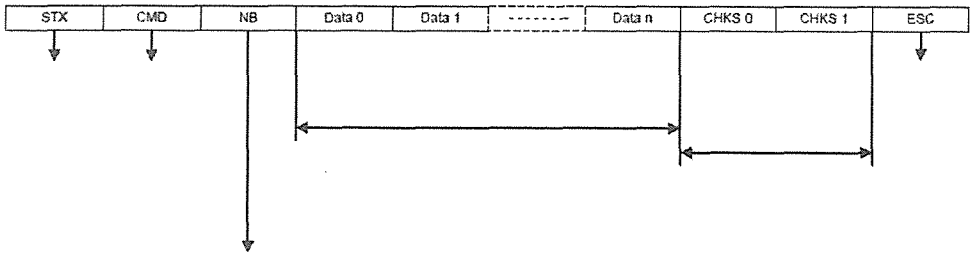


Fig. 5