Université du Québec
Institut national de la recherche scientifique
Énergie, Matériaux et Télécommunications

Technique d'encryption utilisant un guide d'onde de type "fishbone".
Transmissions simultanées de données sécurisées à l'aide de fréquences variables.
Caractéristiques de matériaux intelligents.

Vincent Audet - Ménard
menard@emt.inrs.ca
15 Janvier 2005

*** www.ground418.org pre-release.

Cet article présente une technique d'encryption et de transmission de données utilisant un résonateur "*fishbone*". Notre travail de collaboration avec une équipe de recherche et ce, à titre de stagiaire, nous permet ici de clarifier quelques notions de base sur les résonateurs et guides d'ondes coplanaires. Plus particulièrement, il nous semble intéressant d'explorer les issues d'encryption et de transmission de données possibles avec ces filtres.

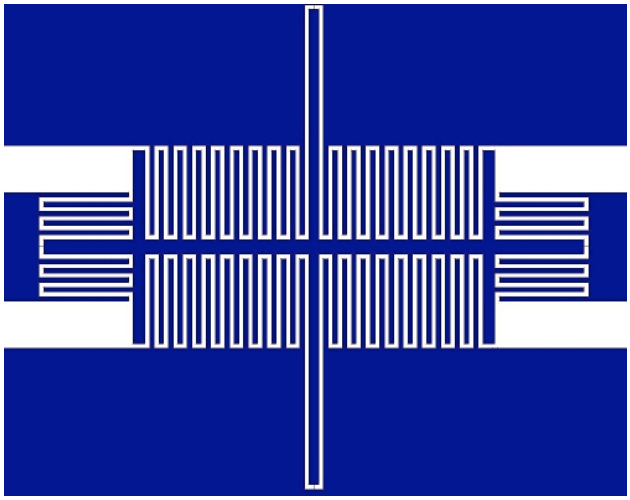
Les mesures et données mentionnées dans cet article sont fictifs, ils n'ont aucune valeur scientifique, le but est simplement d'en exposer les principes d'utilisation. En conséquence, nous utiliserons des ordres de grandeur fictifs appliquées à des notions simplifiées. L'utilisation de la technologie discutée ci-dessous est présentement possible et un projet expérimental pourrait facilement être entrepris afin de de mettre en application les concepts discutés dans cet article.

Qu'est-ce qu'un résonateur *fishbone* ?

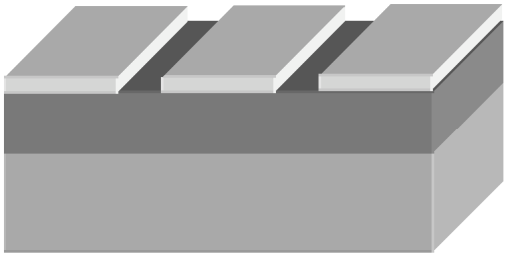
Ce type de filtre micro-ondes a été étudié pour répondre aux besoins grandissants de transmission de données. Sur un cellulaire, il servira à gérer la réception simultanée de données sur des fréquences différentes. Sur un sans fil traditionnel, on utilise habituellement toujours la même fréquence de réception. Lors de la navigation sur *internet* avec un cellulaire, s'ouvre une connexion (utilisant le protocole GSM ou GPRS) avec un fournisseur sans fil. Il est présentement impossible d'ouvrir une deuxième connexion avec un autre fournisseur en utilisant une deuxième fréquence. Il serait souhaitable alors d'avoir un dispositif qui nous permet de changer la fréquence de réception afin d'ouvrir une deuxième connexion. Nous demandons , dès lors, à notre résonateur d'être intelligent, de s'adapter à la fréquence de nos connections.

Le nom *fishbone* est en fait un abbréviation pour « bilateral interdigitated capacitor-loaded CPW ». Ce type de structure fût d'abord étudié par A. Görür en 1998 [1].Il s'agit d'une couche de doigts métalliques déposées sur un composé ferroélectrique. L'onde se propage dans notre guide et rencontre des changements de milieux répétitifs, des trous d'air. Cet espacement entre les doigts est normalement très petit, pas plus de 10 microns. Le composé ferroélectrique doit également être de très faible épaisseur afin de minimiser les pertes. En ce sens, l'épaisseur de la surface métallique doit être adaptée selon la gamme de fréquence à laquelle nous travaillons. Cette couche métallique est déposée sur un composant nous permettant d'avoir un bonne propagation des champs magnétiques entre les doigts ainsi qu'une bonne *accordabilité*. Je ne discuterai pas ici des caractéristiques des matériaux, mais simplement vous exposer le principe général. Les deux figures ci-dessous nous présentent une vue de dessus d'un résonateur (1) et une coupe transversale des différentes couches (2).

(1)



(2)



Le mot *accordabilité* représente le déplacement de la fréquence de résonance en fonction de la différence de potentiel appliquée entre la partie centrale et les plans de masse (grounds) situés en haut et bas de la figure 1. En d'autres termes, si on place une couche du circuit sous tension, la fréquence de résonance augmentera. La différence de potentiel modifie les propriétés du composant ferroélectrique ce qui a pour effet de déplacer la position de la fréquence de résonance. Une bonne *accordabilité* est donc un grand déplacement de la position de la résonance pour une faible différence de potentiel.

Utilisations en cryptographie

Il me faudrait préciser que l'exposé qui suit ne réfère pas à l'encryption conventionnelle. L'encryption utilise habituellement les mathématiques, tandis que cette technique représente plutôt une méthode de transmission de données utilisant une clé physique. Rien ne vous empêche de transmettre de l'information préalablement encryptée. L'analyse qui suivera fait référence aux caractéristiques d'un filtre de type fishbone mais le concept d'encryption peut s'appliquer à n'importe quel filtre accordable.

Pour commencer, il nous faut déterminer une table d'*accordabilité* pour un résonateur *fishbone* fictif:

Différence de pot. (ΔV)	Fréq. de Résonance (ghz)
0	2
2	2.2
4	2.4
6	2.6
8	2.8
10	3
...	...

Considérons que Alice et Bob sont des amis qui se connaissent depuis très longtemps. Ils ont chacun un résonateur identique qui répond avec les résonances de la table ci-dessus. Ils connaissent cette table et ils la gardent secrète. Ils s'agit en fait de leur clé. Nous verrons plus tard qu'une partie de cette table pourrait être rendue publique.

Comme c'est dans son habitude, Alice voudrait envoyer un message binaire à Bob. Le message binaire d'Alice est: 1010011010.
Alice et Bob s'entendent sur un mode de lecture ; si l'onde résonne, c'est "1", sinon c'est "0". Elle détermine ensuite le (ΔV) que Bob devra utiliser pour lire le message avec son résonateur.

Clés de Alice et Bob: 2V-10V-8V-0V-6V.

Pour chaque binaire que Alice envoie, Bob doit appliquer cet ordre de différences de potentiels. Ils doivent se synchroniser (Δt = 0.1 secondes) puis, quand Alice change de fréquence d'émission, Bob doit changer son (delta V) de lecture. Le message codé est donc une série de fréquences différentes. Quand Alice désire envoyer un 0, elle sait qu'elle peut envoyer n'importe quelle fréquence, sauf celle qui résonne.
Nous obtenons donc:

```
Clé (V)
2V -  10V - 8V -  0V -  6V -  2V - 10V -  8V -  0V -  6V

msg clair d'Alice (binaires):
1  -  0  -  1  -  0  -  0  -  1  -  1  -  0  -  1  -  0

msg codé (gHz):
2.2 - 2.6 - 2.8 - 2.8 - 3.0 - 2.2 - 3.0 - 2.4 - 2.2 - 2.2

résonance de Bob (o:oui n:non):
o  -  n  -  o  -  n  -  n  -  o  -  o  -  n  -  o  -  n

Bob sait ce qui résonne : oui : 1
et ce qui ne résonne pas : non : 0
```

Ce schéma est simplifié et nous savons tous qu'il y a plus d'une étape entre la réception de l'onde et la lecture en binaire. L'onde elle-même est décodée d'une manière différente et contient aussi des données. Dans l'optique de vulgariser uniquement certains principes en lien avec la notion d'*accordabilité,* nous ne nous y intéresserons pas. Par ailleurs, il faut aussi penser que pendant ce temps Charles, qui est un ami de Bob peut très bien communiquer avec Bob avec une clé différente. Il suffit de diminuer le temps (Δt) de réception à 0.05 secondes.

Analyse du schéma d'encryption

Prenons d'abord comme exemple une conversation téléphonique entre Alice et Bob. Puisque Alice veut téléphoner à Bob, elle décroche le combiné en sachant très bien que Bob possède un téléphone. Elle connaît de mémoire le numéro de Bob et sait que le numéro ne changera pas durant la conversation. Si le numéro de Bob changeait durant la conversation, il faudrait qu'elle le sache et recompose le nouveau numéro. Mais normalement, si Bob ne dit pas à Alice son nouveau numéro, elle ne saura pas comment rejoindre Bob.

Le changement de fréquence utilisé dans notre protocole correspond un peu à un changement de numéro brusque de la part de Bob. À l'aide de la table d'*accordabilité*, Alice sait quelle fréquence utiliser (ou ne pas utiliser) pour que Bob puisse lire le message correctement. Ils utilisent deux clés pour leur communication. La première est la suite de différences de potentiels utilisés, dans l'exemple, elle était 2V-10V-8V-0V-6V. La seconde partie de la clé est l'ensemble des caractéristiques physiques du résonateur qui définissent la table d'*accordabilité*. Il s'agit ici d'une clé physique.

En variant la forme du résonateur nous modifions les caractéristiques de réception. Par exemple, Bob change son résonateur sans le dire à Alice. Il utilise un modèle plus long, avec plus de doigts. Il utilise la même clé pour lire les fréquences qu'il reçoit. Le fait que Bob utilise un résonateur avec plus de doigts modifie sa table d'*accordabilité*. C'est en utilisant ce concept que l'on peut élaborer un schéma de transmission sécurisée. Mais avant tout, nous devons étudier de plus près les variations d'*accordabilité* en fonction des modifications physiques du résonateur.

Caractéristiques de la clé physique

Vous utilisez tous les jours une clé physique pour débarrer votre maison, voiture, bureau, etc.. Une clé de maison conventionnelle possède environ cinq encoches avec 10 positions possibles pour chaque encoche. Il y a donc 100 000 clés possibles. (10^5). Plusieurs données changent notre position de résonance ainsi que notre table d'*accordabilité*. C'est sur ces variables que repose l'entière sécurité de notre transmission. Comme une clé de maison, nous pouvons modifier les caractéristiques physiques du résonateur pour calculer notre quantité de clés possibles. Voici quelques caractéristiques:

1. Nombre de doigts,
2. Longueur des doigts,
3. Largeur des doigts,
4. Largeur des gaps,
5. Longueur totale du résonateur.

Rapidement, nous trouvons cinq caractéristiques, qui analogiquement correspondent aux encoches d'une clé de maison. Contrairement à une clé conventionnelle, nous avons beaucoup plus de "positions" possibles. On peut énumérer les positions possibles pour chaque caractéristique modifiable de notre résonateur.

1. Nombre de doigts, 10 à 1010
2. Longueur des doigts, 10 à 1010microns
3. Largeur des doigts, 10 à 1010microns
4. Largeur des gap, 10microns à 10nm
5. Longueur totale du résonateur. 0.1mm à 10mm

Il s'agit ici d'une grossière approximation que nous développerons plus loin dans le texte. Il y a un minimum de 1000 positions possibles pour chaque caractéristique physique modifiable. Nous avons donc 1000^5 positions possibles, 1,000,000,000,000,000 clés possibles. L'ordre de grandeur augmente énormément le nombre de clés possibles. Les grandeurs nous imposent aussi quelques contraintes; il serait improbable et inutile d'utiliser des doigts larges de 1000 microns et longs de 10 microns. De plus, il est difficile (faisable mais très long) de prédire quelle sorte de

résonance l'on obtiendra avec des caractéristiques physiques aléatoires. Peut-être qu'il n'y aura pas de résonance. Peut-être il y en aura deux, trois. Il a été démontré que ce type de résonateur possède des harmoniques. La résonance revient donc à intervalle régulier (ex.: 4GHz, 8GHz, 12GHz...)

Certaines personnes qui étudient les filtres électromagnétiques me diront, avec raison, que mon nombre de clés est gonflé. Puisqu'un filtre possédant des caractéristiques aléatoires risque de ne pas (ou très peu) résonner, il faudra donc considérer un maximum de 10^5 clés possibles. Comme le nombre de clés possibles vient de diminuer dramatiquement, il est possible de placer plusieurs résonateurs en parallèle. La résonance observée sera la résultante de chaque résonateur. Notre nombre de clés augmente suivant la formule $(10\text{^}5)^n$, où n est le nombre de résonateurs en parallèle. Il faut donc considérer les caractéristiques physiques de chaque résonateur, le nombre de dispositifs en parallèle, l'ordre de grandeur ainsi que les matériaux utilisés et l'épaisseur de métal pour évaluer le nombre de clés possibles.

Il est possible de simuler un résonateur à l'aide de logiciels afin de trouver la position de sa fréquence de résonance. Une telle simulation prend entre 30 minutes et 4 heures sur un pentium 4, 2.8gHz. Il est présentement impossible de simuler un résonateur avec 1000 doigts de 10 microns en largeur puisque la matrice de calcul des effets électromagnétiques est exponentielle. De plus, ces logiciels présentent une très grande incertitude (plus de 0.1gHz) qui influenceraient beaucoup dans les cas d'utilisations parallèles. Une attaque par force brute (calcul de toutes les clés) serait impossible pour deux filtres placés parallèlement.

Principe clés aléatoires dynamiques

Si un observateur donné qui ne possède pas les deux parties de la clé, écoute la conversation entre Alice et Bob, il entendra une série de fréquences. Il sait que les fréquences correspondent à des 0 et des 1. Il pourra après un certain temps, retrouver des répétitions de fréquences. L'observateur doué en *cryptanalyse* pourra déduire que s'il retrouve la même fréquence à intervalle régulier durant la transmission, celle-ci correspond à "1".

Comme pour la cryptographie conventionnelle, la sécurité de l'information transmise dépend de la longueur de la clé par rapport à la longueur du message. Si la clé est aussi longue que le message, le problème ne se pose pas. Néanmoins, le message codé comprend 50% de fréquences aléatoires. Quand Alice veut envoyer un "0", elle sait qu'elle peut envoyer n'importe quelle fréquence qui ne résonne pas pour Bob. Cet avantage permet de réduire les risques de répétitions qui compromettent souvent les algorithmes modernes.

De plus, comme nous observons une transmission de données et non un archivage, nous pouvons changer la clé durant la transmission. Il suffit que Bob sache quand Alice change de clé. Par exemple, Alice et Bob possèdent une banque de clés secrètes de grandeurs variables. Ils ne communiquent pas la clé durant la transmission mais plutôt un numéro de clé (id). Ils savent que s'ils utilisent toujours la même clé, la transmission pourrait être compromise. Au début de chaque transmission Alice transmet la suite de clés qu'elle utilisera. Cette suite de clé devra tenir compte de la grandeur du message ainsi que la réponse potentielle de Bob.

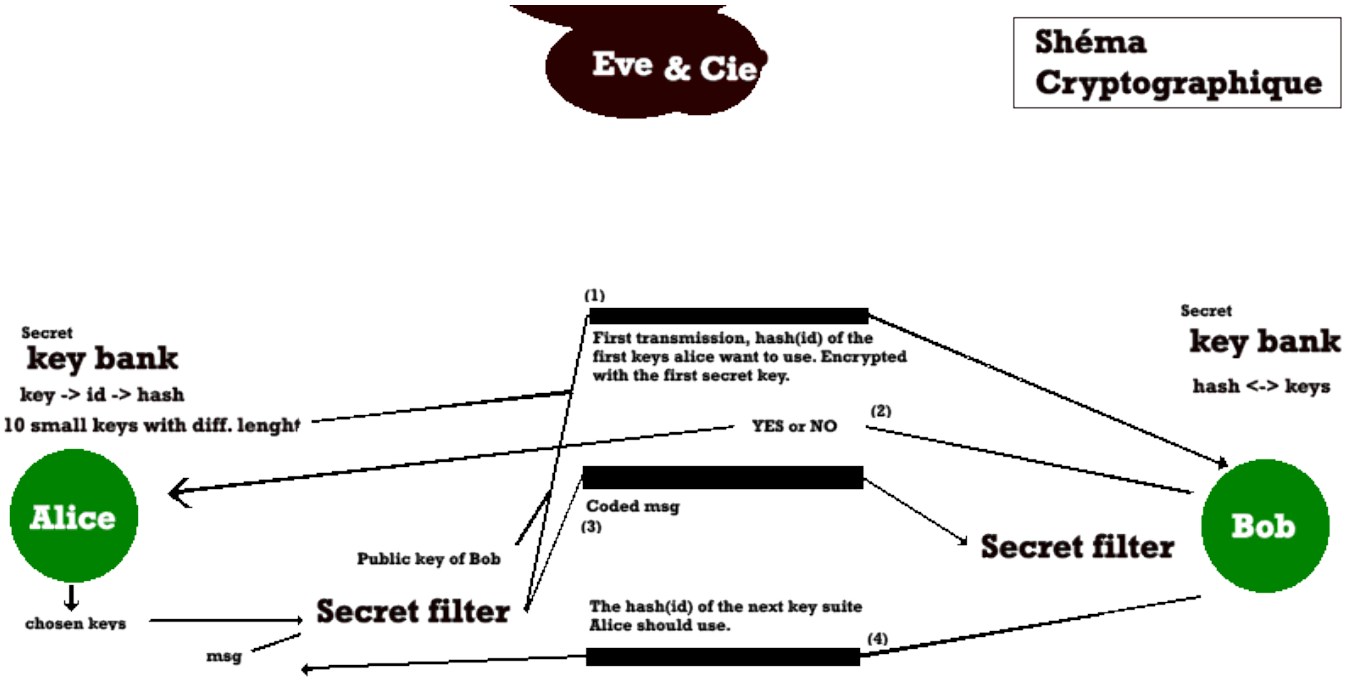
La réponse de Bob a toujours la même forme. Il s'agit de la prochaine suite de clés à utiliser. Si Bob n'a rien à transmettre, Alice saura avec quelle suite de clés entreprendre une autre

conversation. Un fonction de *hash* à sens unique permet à Bob et Alice de faire varier le nombre de clés dans leur suite en gardant la même "grandeur" de leur transmission. Une petite comparaison dans la banque de clés permet, avec la même fonction de *hashage*, de trouver les clés. Pour ceux qui ne sont pas familiers avec les fonctions de *hashage*, voici un exemple d'utilisation. Ceci montre comment la longueur du message codé de dépend pas de la longueur du message clair.

```
md5("Alice")          = 64489c85dc2fe0787b85cd87214b3810
md5("Alice loves Bob")= 5f3b147ae863e541701d7011e597e98b
```

Cette méthode utilise plusieurs avantages. L'entreposage d'une banque de clé énorme ne pose pas de problème puisque nous savons très bien que l'espace disque est beaucoup moins cher que la quantité de données transmises sur un réseau. Un abonnement d'un mois sur internet paie facilement un disque dur de 20go. De plus, la vitesse des processeurs nous permet de comparer rapidement les fonctions de *hash* dans notre banque de clés. La première clé qu'utilise Alice peut être publique, si l'on prend en considération que la banque de clé est secrète et que cette clé est utilisée une seule fois pour établir la suite de clés utilisées. Alice utilise donc la clé publique de Bob pour établir les clés futures.

Pour conclure, voici un petit schéma qui résume la conversation entre Alice et Bob suivant le protocole expliqué ci-dessus:



Références

[1] Adnan Gürür, Ceyhun Karpuz, and Mustafa Alkan,
"Characteristics of Periodically Loaded CPW Strutures",
IEEE Microwave and guided wave letters, Vol. 8, No. 8, August 1998

[2] M.Ouaddari, S. Delprat, F. Vidal, M. Chaker and Ke Wu, Fellow, IEEE, **"Microwave Characterization of Ferroelectric Thin-Film Meterials"**, IEEE transactions on microwave theory and technique.

[3] Bruce Schnier, **Secrets & Lies**, Digital Security in a Networked World, Wiley Publishing, Inc. 2000, chap 6, p. 85-101