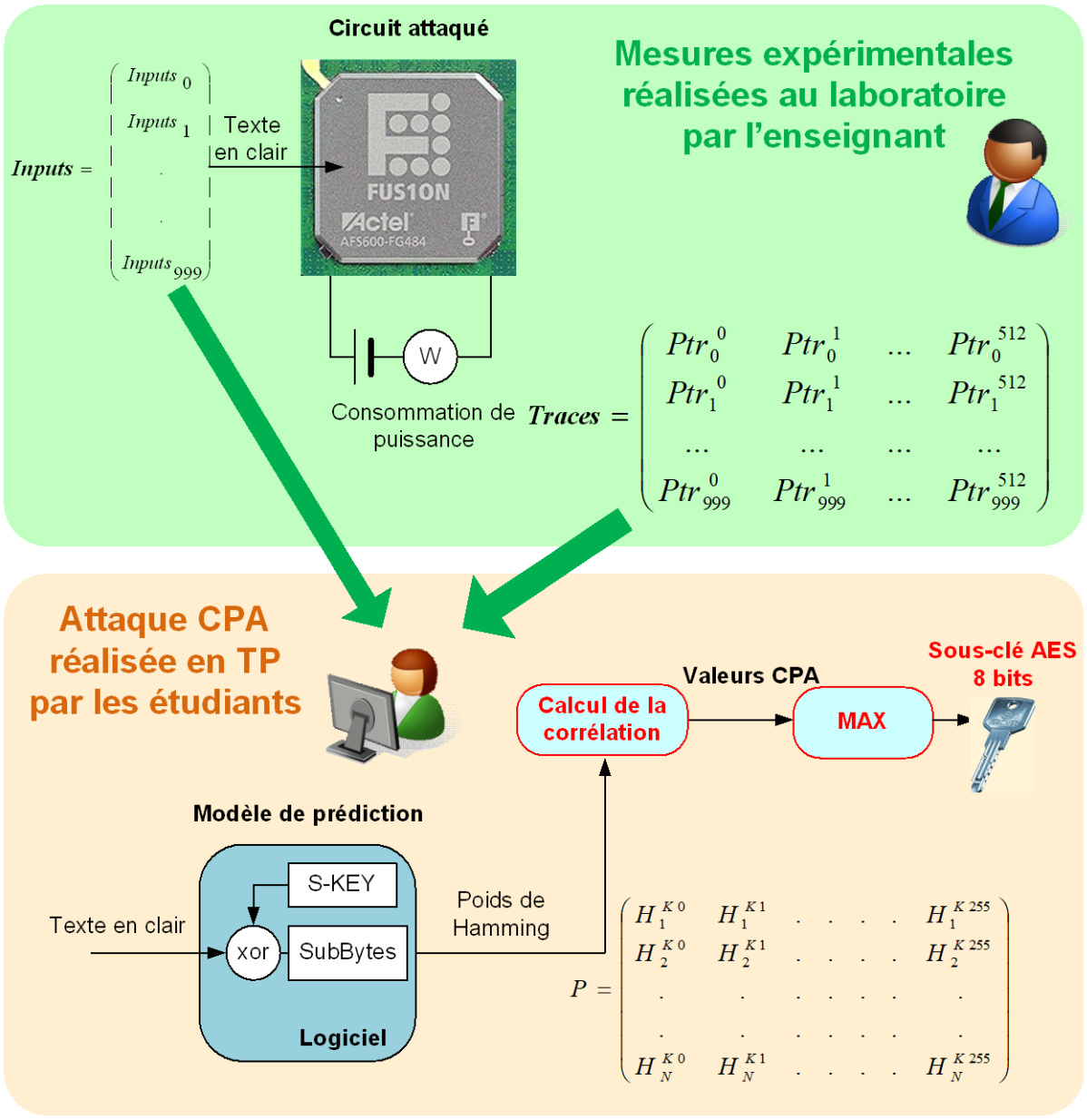
Architectures matérielles sécurisées - TP CPA

L'objectif de ce TP est de réaliser l'attaque CPA d'un chiffrement AES implanté dans un FPGA à partir de traces fournies. L'attaque ne cible que le premier octet de la clé secrète. Le code de l'attaque sera développé avec MATLAB.

Schéma de principe de l'attaque



Vous devez donc modéliser en langage Matlab les deux premières étapes de l'algorithme AES (*AddRoundKey* et *SubBytes*), puis vous devez modéliser le calcul de la matrice *P* d'estimation du poids de Hamming de la sortie de l'opération *SubBytes* pour toutes les sous-clés possibles

(sur 1 octet). Enfin un calcul de corrélation doit vous permettre d'extraire la valeur de la clé secrète, comme celle qui donne le maximum de corrélation. Vous devrez présenter dans un compte rendu une visualisation graphique en 2D et 3D de la corrélation afin de mettre en évidence les points d'intérêts (fuites d'information) lors du chiffrement.

Attention : La matrice Inputs ne contient que l'octet de poids faible de chaque bloc de 128 bits du texte en clair. Les 512 points de mesure obtenu pour chaque trace de consommation de puissance sont codés sur 8 bits (à cause de l'oscilloscope utilisé).

SubBytes : Pour la réalisation de cette opération, un vecteur de 256 valeurs contenant les valeurs de substitution des octets est donné.

Aide MATLAB

Pour réaliser le calcul de corrélation avec Matlab il est possible d'utiliser la fonction `corrcoef`

Pour obtenir des informations sur des fonctions Matlab il faut utiliser la commande `help`

Quelques fonctions Matlab qui peuvent être utiles pour ce TP (liste non exhaustive) :

- `load`
- `bitxor`
- `bin2dec`
- `dec2bin`
- `sum`
- `length, size`
- `bitget`
- `max`
- `plot`
- `surf`

Dans le programme .m :

% (% => commentaire) pour charger les vecteurs **Inputs1**, **SubBytes** et **traces** (attention les fichiers.mat doivent être dans le même dossier que le programme.m) :

```
load('Inputs.mat');

load('SubBytes.mat');

load('traces1000x512.mat');
```

%pour créer un vecteur pour les 256 valeurs de sous-clé :

```
subb_key = 0:255;
```

%pour céer un matrice de 1000 ligne de 256 colonnes avec que des zéro (initialisation)

```
ma_matrice = zeros(1000,256);
```

% pour avoir la valeur à la ligne i (valeur entre 1 et 1000) et la colonne j (valeur entre 1 et 256) de ma_matrice

```
valeur = ma_matrice(i,j);
```

% pour avoir la ligne i de ma_matrice

```
vecteur_ligne = ma_matrice(i,:);
```

% pour avoir la colonne j de ma_matrice

```
vecteur_ligne = ma_matrice(:,j);
```

```
% pour avoir la valeur à la ligne i et la colonne j de ma_matrice
valeur = ma_matrice(i,j);


%boucle for d'un indice i qui va par exemple de 1 à 1000
for i= 1:1000
  % instructions du for
end
```