Susceptibilité des mémoires embarquées aux injections de fautes

Ziling Liao
Doctorante en 1ère année
LIRMM
Encadrants : Philippe Maurine, Florent Bruguier

Contexte

Sécurité matérielle - Injection par fautes

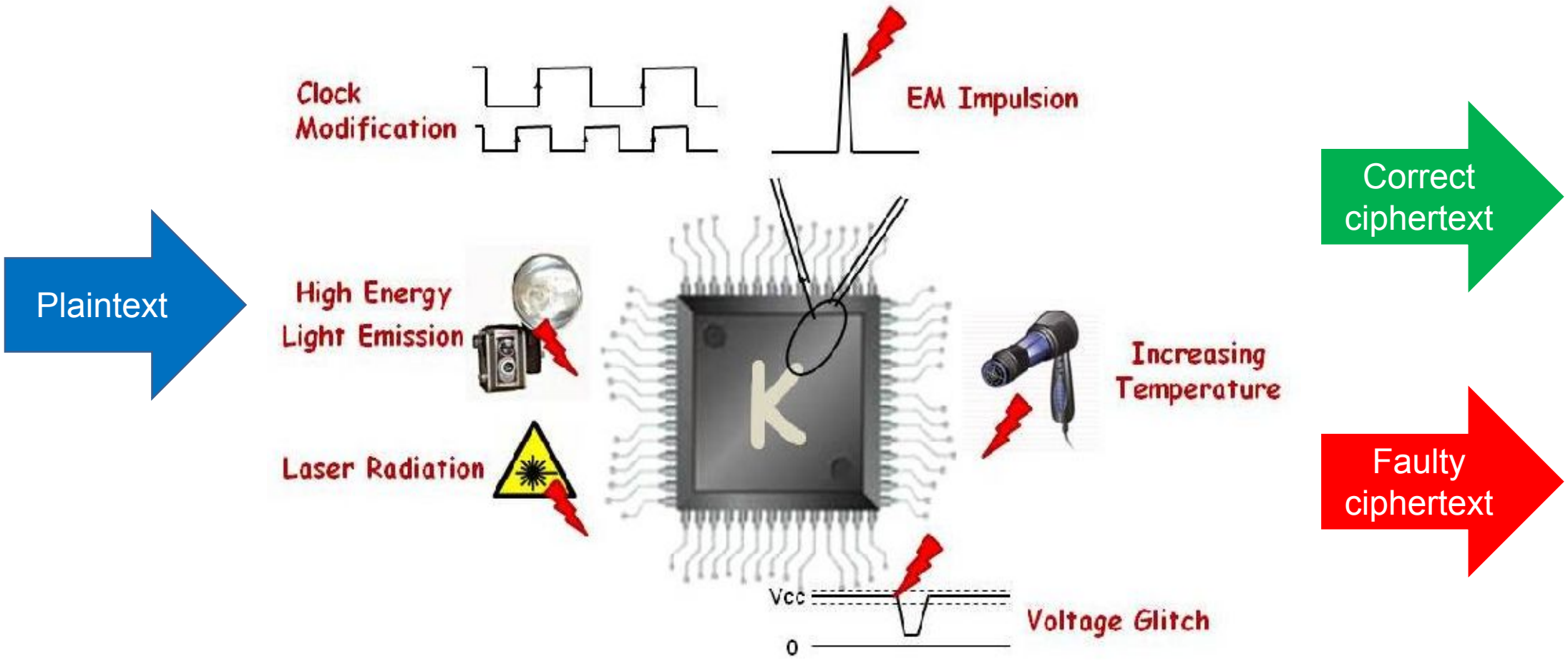


Figure 1. Attaques par injection de fautes (Source : https://www.semanticscholar.org/paper/Etude-de-la-vuln%C3%A9rabilit%C3%A9-des-circuits-l%27injection-Mirbaha/6833981d13b4fdd26e26567a18e927f36979a575)

Qu'est ce que la BBI (Body Bias Injection) ?

Principe : Impulsion de tension dans le substrat

=> Perturbations sur :

- **Les réseaux d'alimentation**
- **L'horloge**
- **Les signaux de contrôle**
- **…**

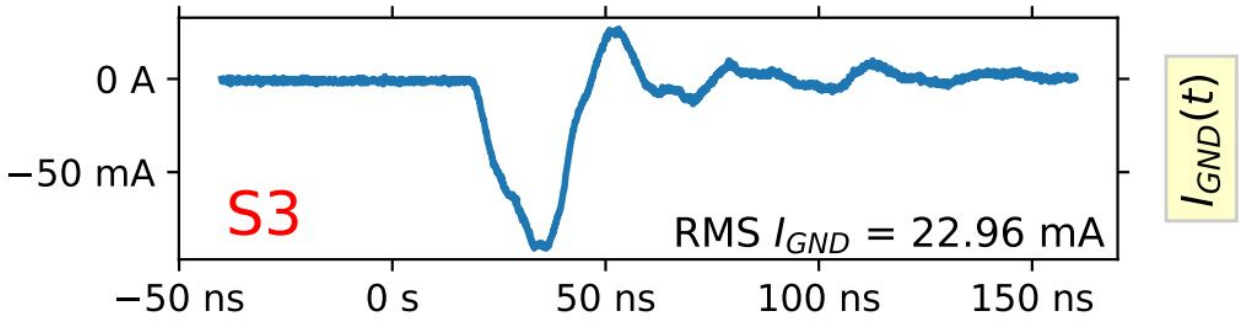


Figure 2. Fluctuations de courant causées par BBI (Source : https://www.computer.org/csdl/proceedings-article/fdtc/2023/425200a048/1W5bh1IiFzi)

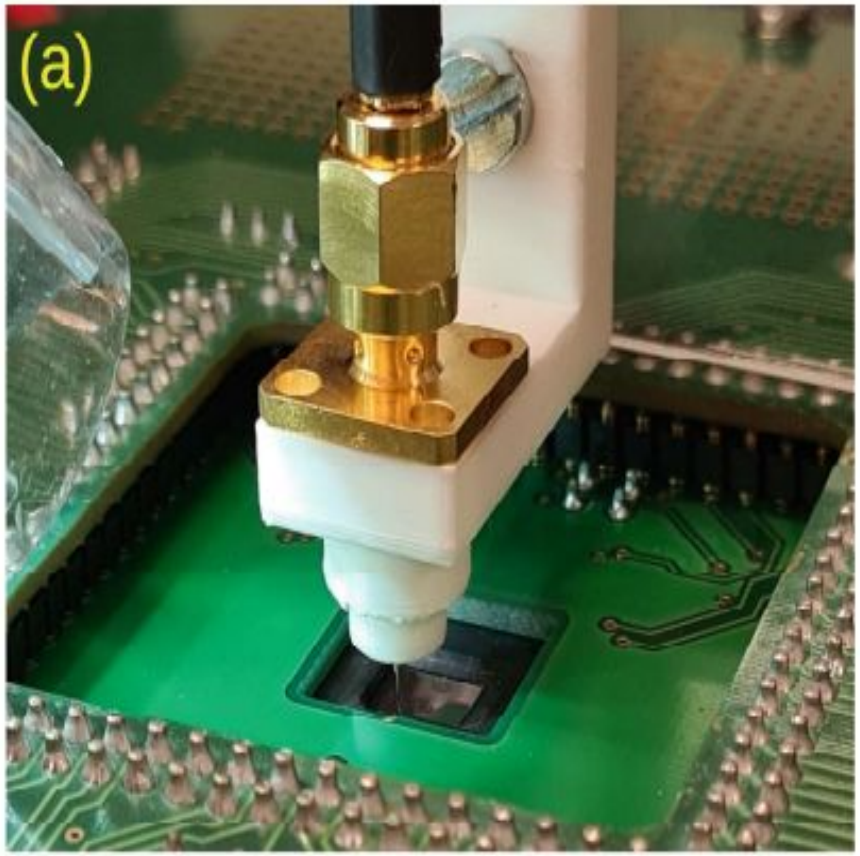


Figure 3. Plate-forme d'injection

Que sait on sur la BBI ?

2011 — Yet Another Fault Injection Technique : by Forward Body Biasing Injection

Invention de cette technique - Attaque de Bellcore sur RSA

2012 — Voltage spikes on the substrate to obtain timing faults

Nature des fautes sur la logique

2016 — Body biasing injection attacks in practice (Lumped model for dual-well substrates)

Premier modèle explicatif

2020 — Low-cost body biasing injection (BBI) attacks on WLCSP devices

Attaque sur AES

2022 — Breaking a Recent SoC's Hardware AES Accelerator Using Body Biasing Injection

Attaque sur AES

2023 — A better practice for Body Biasing Injection

Amélioration de la pratique d'injection

Les objectifs ?

Objectif 1 : Comprendre l'impact de la BBI sur les mémoires

Pourquoi ?

Dans le domaine de BBI, il manque des études sur les mémoires.

Les mémoires sont essentielles pour l'exécution des programmes embarqués, c'est l'endroit où les instructions et les données sont stockées.

Par où commencer ?

On commence par observer les attaques sur une opération écriture dans SRAM d'un circuit STM32.

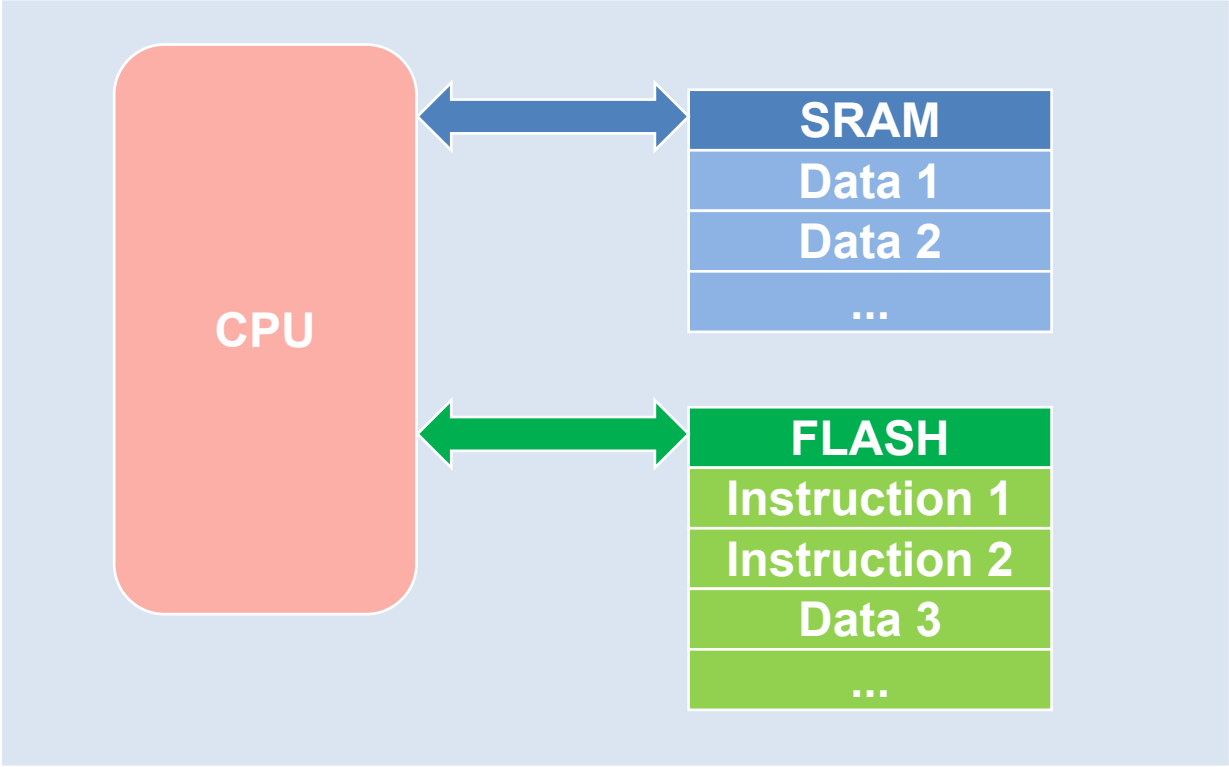


Figure 4. Rôle des mémoires

Les objectifs ?

Objectif 2 : Comprendre l'impact de la BBI sur l'exécution des programmes embarqués dans un sens général

Par où commencer ?

On commence par observer les impacts de BBI sur le Program Counter (PC).

Pourquoi PC ?

Le PC joue un rôle clé dans l'exécution des programmes. Il dirige le CPU vers l'instruction à exécuter.

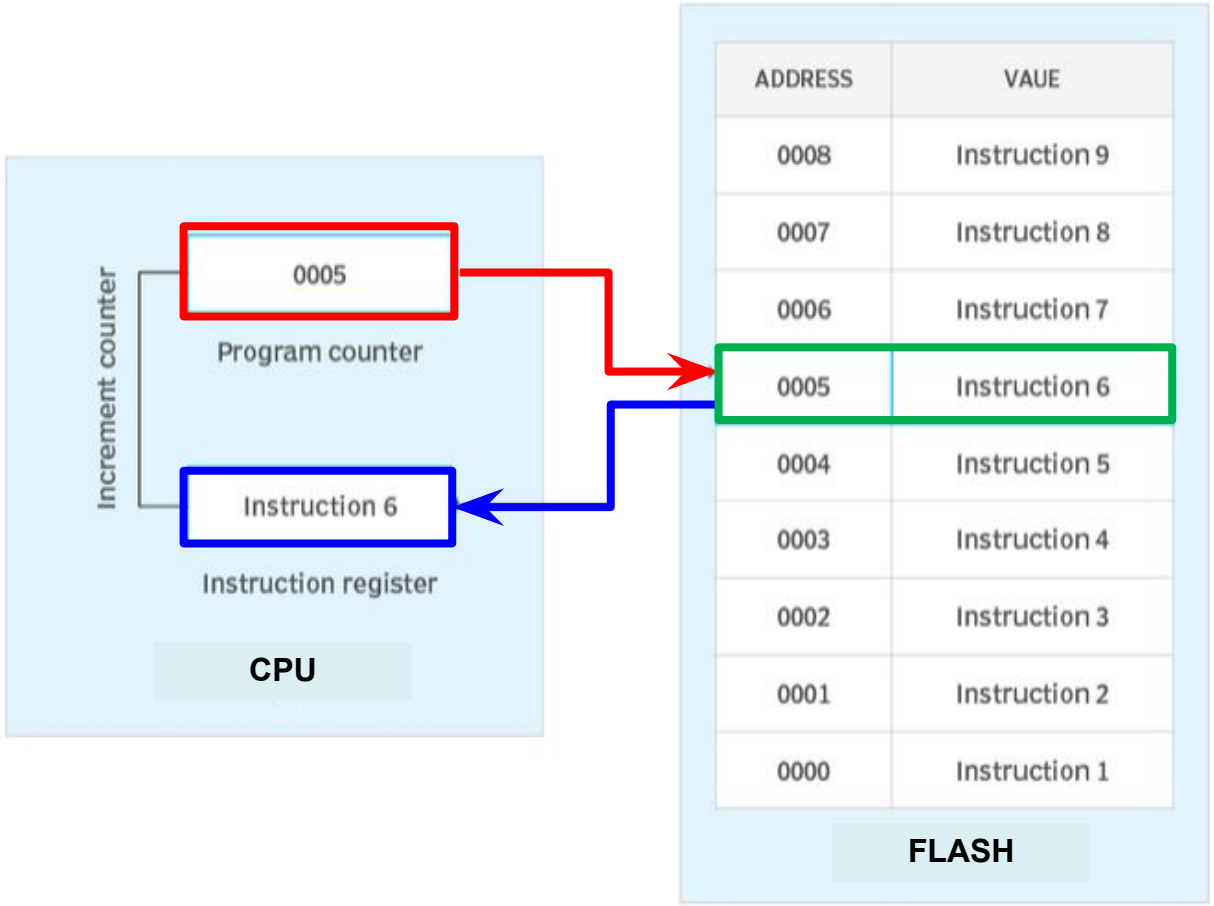


Figure 5. Rôle de Program Counter (Source:https://www.techtarget.com/whatis/definition/program-counter)

Premières observations

Extrait de la fonction cible :

```
...
NOP
NOP
NOP
NOP
NOP
STR R4, [R3]  (Objectif1)
NOP
NOP
NOP
NOP
NOP
...
observation de PC  (Objectif2)
...
```

Injections pendant ces cycles

Commentaires :
NOP => instruction vide
STR R4, [R3] => écrire une donnée(R4) depuis CPU dans une adresse(R3)

Circuit cible
STM32F439 ARM Cortex M4, 2MB FLASH, 256KB SRAM

Injections dans la région SRAM

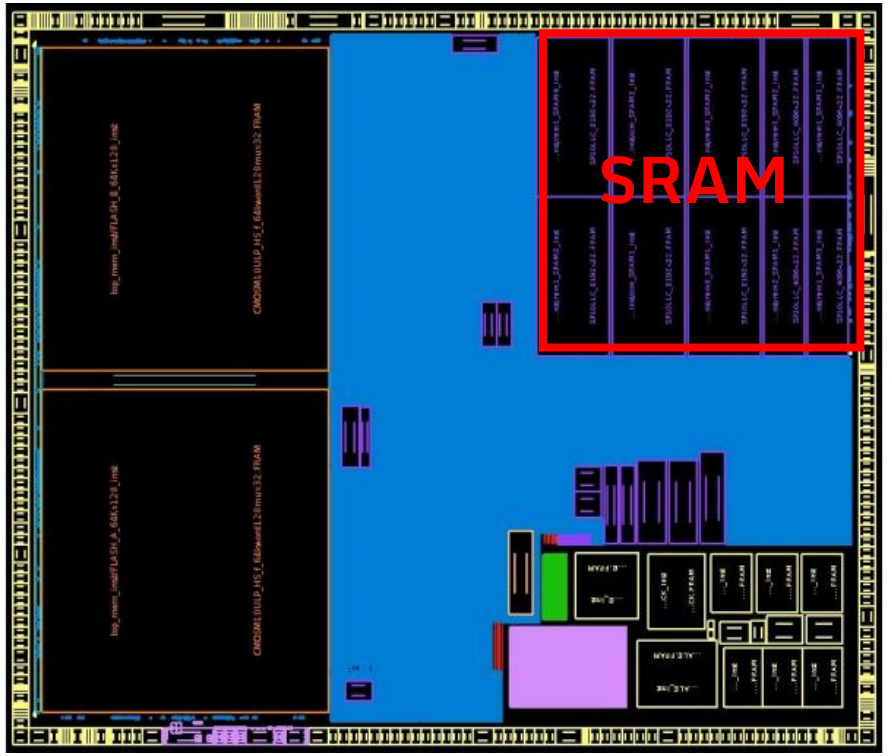


Figure 6. Layout du circuit cible

Résultats d'observation - Fautes de mémoire SRAM

Fautes de donnée

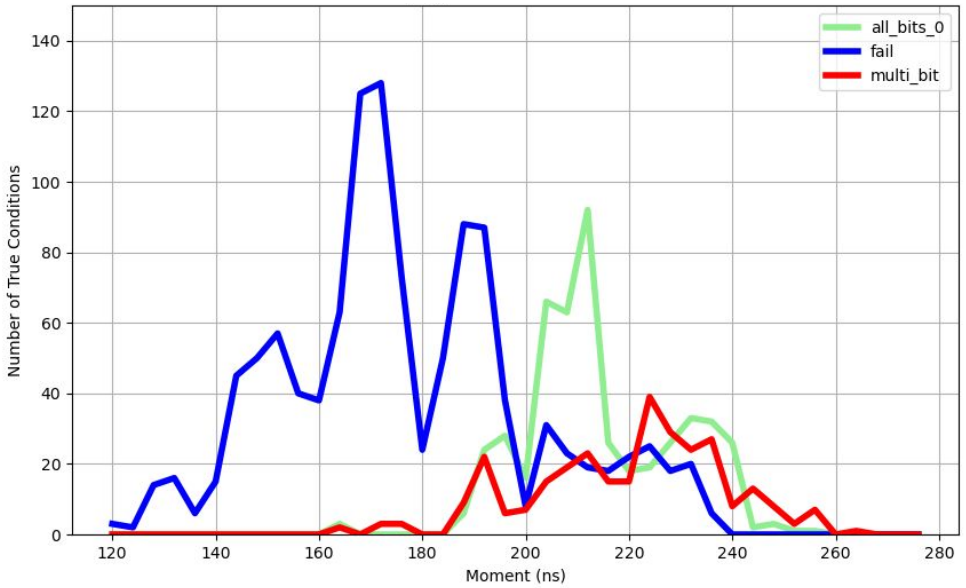


Figure 7. Distribution de tous les types de fautes de donnée dans le temps

Fautes d'adresse

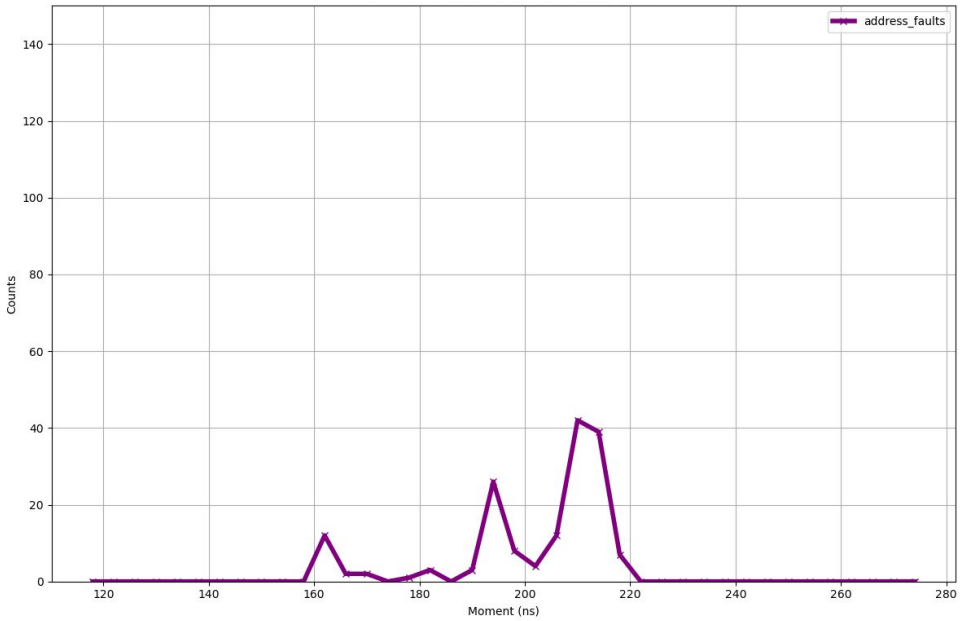


Figure 8. Distribution de tous les fautes d'adresse dans le temps

Bleu : empêchement d'écriture
Vert : mise à 0 du contenu de l'adresse cible
Rouge : mauvaise donnée non 0 écrite dans l'adresse cible
Violet : fautes d'adresse

Résultats d'observation - Décalage de valeur PC

Valeur de décalage = Valeur de PC observée - Valeur de PC normale (fixée)

Caractéristiques du décalage :

- **Les valeurs de décalage peuvent être positives ou négatives**
- **Les valeurs de décalage positives/négatives correspondent à un temps d'exécution de fonction plus long/court**
- **Les valeurs de décalage dépendent du type d'instruction**

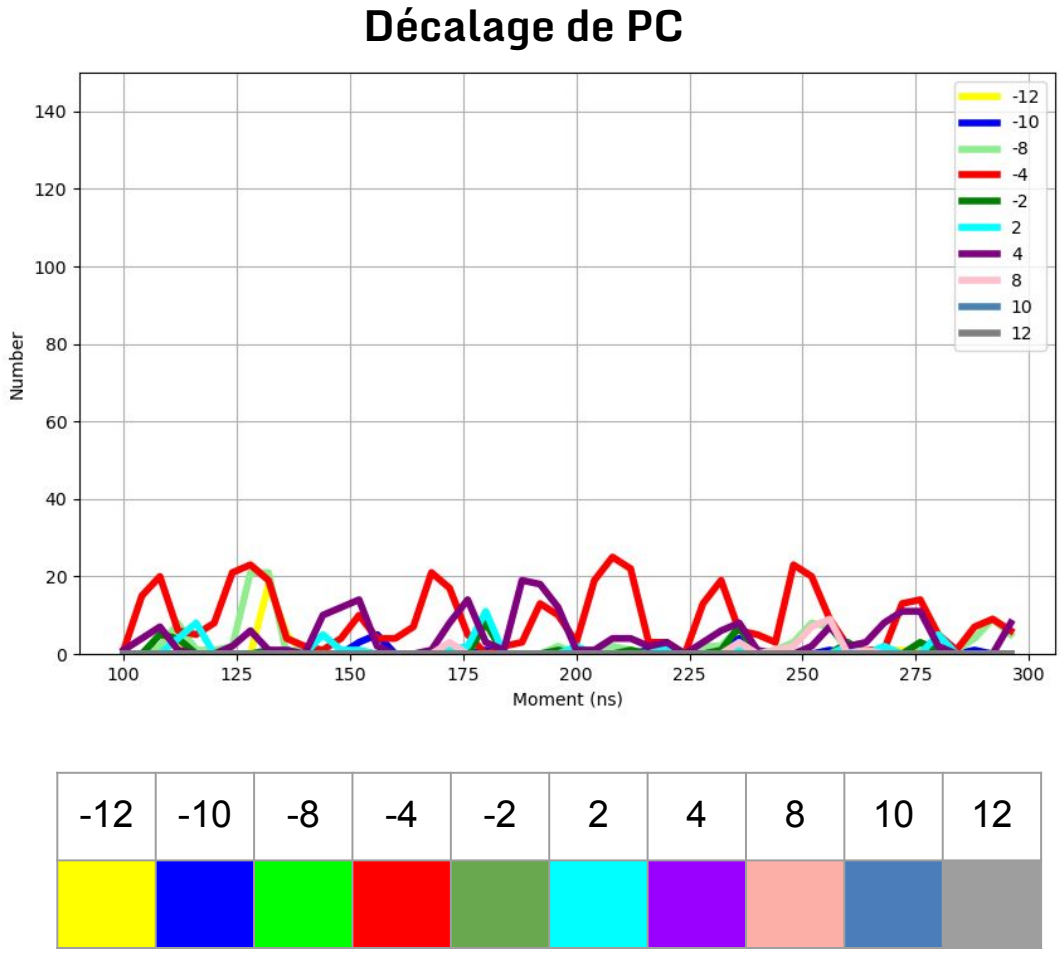


Figure 9. Distribution de tous les valeurs de décalage PC dans le temps

Conséquence de décalage de PC - adressage relatif au PC

Exemple : Ecrire 0x12345678 dans l'adresse 0x40040214

```
...
instruction 1 : ldr r1, = 0x40040214
instruction 2 : ldr r2, = 0x12345678
instruction 3 : str r2, [r1]
...
```

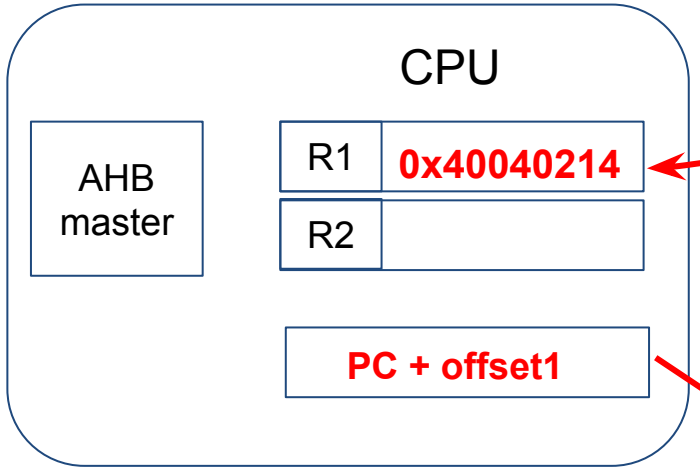
Compilation

Adressage relatif au PC

```
...
instruction 1 : ldr r1, [pc, offset1]
instruction 2 : ldr r2, [pc, offset2]
instruction 3 : str r2, [r1]
...
```

FLASH	
Address	Instruction/data
...	...
@ins 1	instruction 1
@ins 2	instruction 2
@ins 3	instruction 3
...	...
@data 1	0x40040214
@data 2	0x12345678
@data 3	0x4005315
@data 4	0x87654321
...	...

Conséquence de décalage de PC - adressage relatif au PC



ldr r1, [pc, offset1]

FLASH	
Address	**Instruction/data**
…	…
@ins 1	instruction 1
@ins 2	instruction 2
@ins 3	instruction 3
…	…
@data 1	0x40040214
@data 2	0x12345678
@data 3	0x40050315
@data 4	0x87654321
…	…

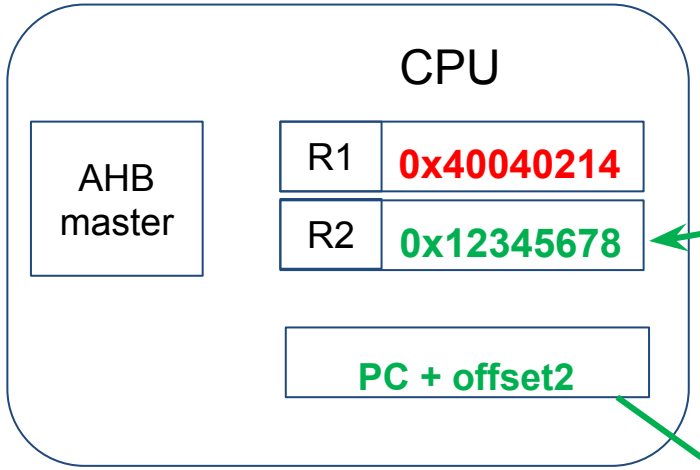
Module destinataire correct

Register	
Addr	0x40040214
Data	

Module destinataire faux

Register	
Addr	0x40050315
Data	

Conséquence de décalage de PC - adressage relatif au PC



ldr r2, [pc, offset2]

FLASH	
Address	**Instruction/data**
…	…
@ins 1	instruction 1
@ins 2	instruction 2
@ins 3	instruction 3
…	…
@data 1	0x40040214
@data 2	0x12345678
@data 3	0x40050315
@data 4	0x87654321
…	…

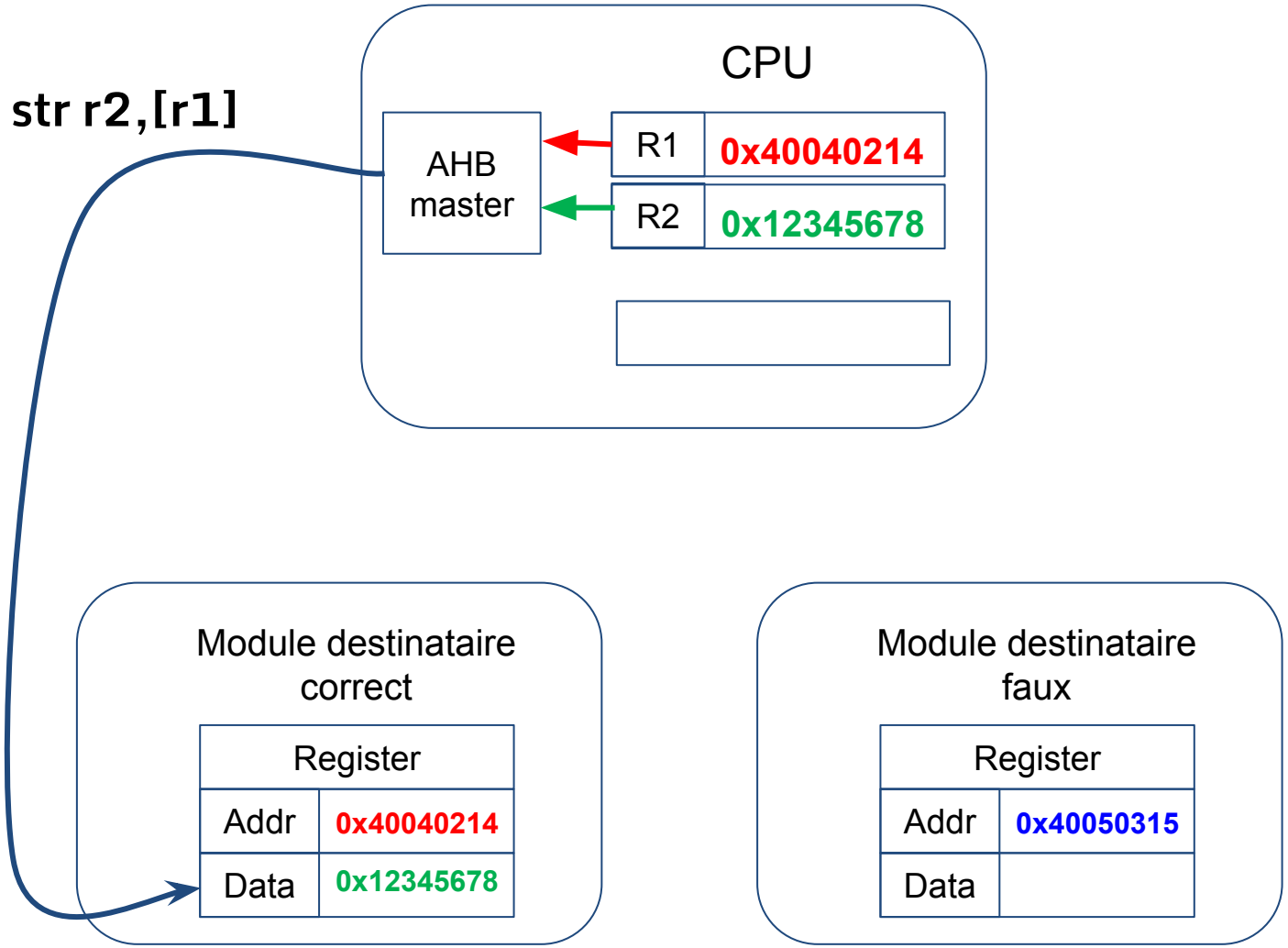
Module destinataire correct

Register	
Addr	0x40040214
Data	

Module destinataire faux

Register	
Addr	0x40050315
Data	

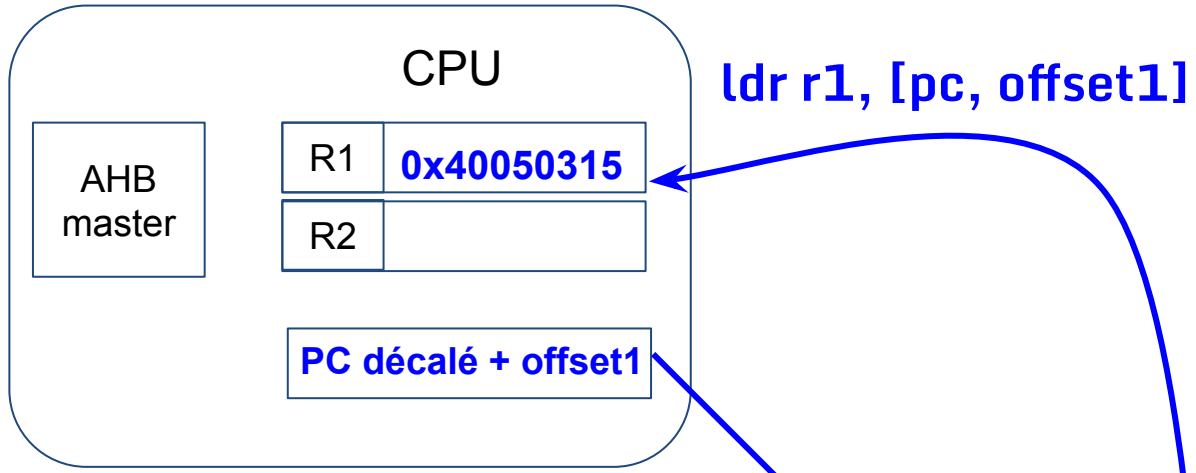
Conséquence de décalage de PC - adressage relatif au PC



FLASH	
Address	**Instruction/data**
…	…
@ins 1	instruction 1
@ins 2	instruction 2
@ins 3	instruction 3
…	…
@data 1	0x40040214
@data 2	0x12345678
@data 3	0x40050315
@data 4	0x87654321
…	…

Conséquence de décalage de PC - adressage relatif au PC

Si PC est décalé d'une certaine valeur



ldr r1, [pc, offset1]

FLASH	
Address	**Instruction/data**
…	…
@ins 1	**instruction 1**
@ins 2	**instruction 2**
@ins 3	**instruction 3**
…	…
@data 1	**0x40040214**
@data 2	**0x12345678**
@data 3	**0x40050315**
@data 4	**0x87654321**
…	…

Module destinataire correct

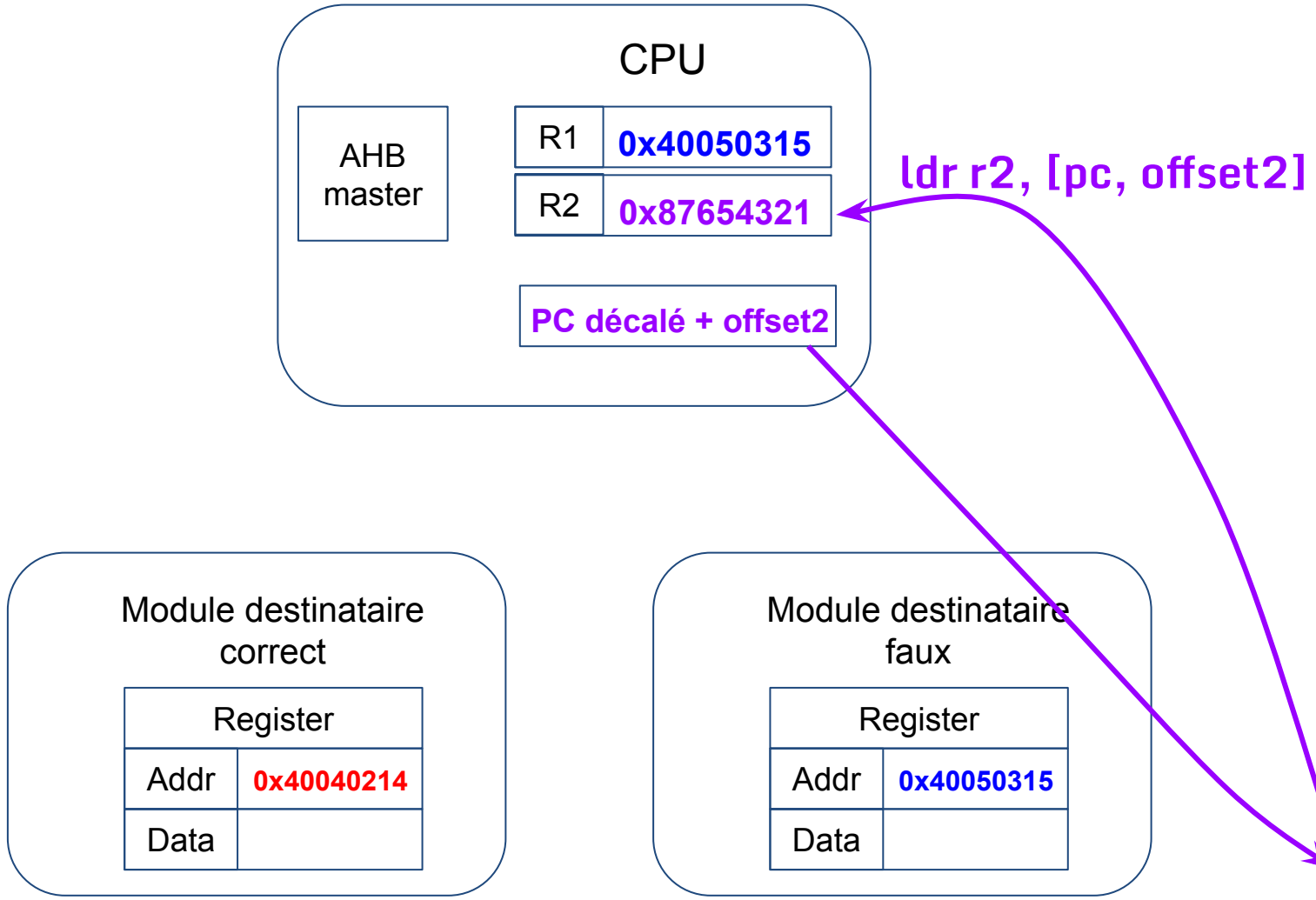
Register	
Addr	0x40040214
Data	

Module destinataire faux

Register	
Addr	0x40050315
Data	

Conséquence de décalage de PC - adressage relatif au PC

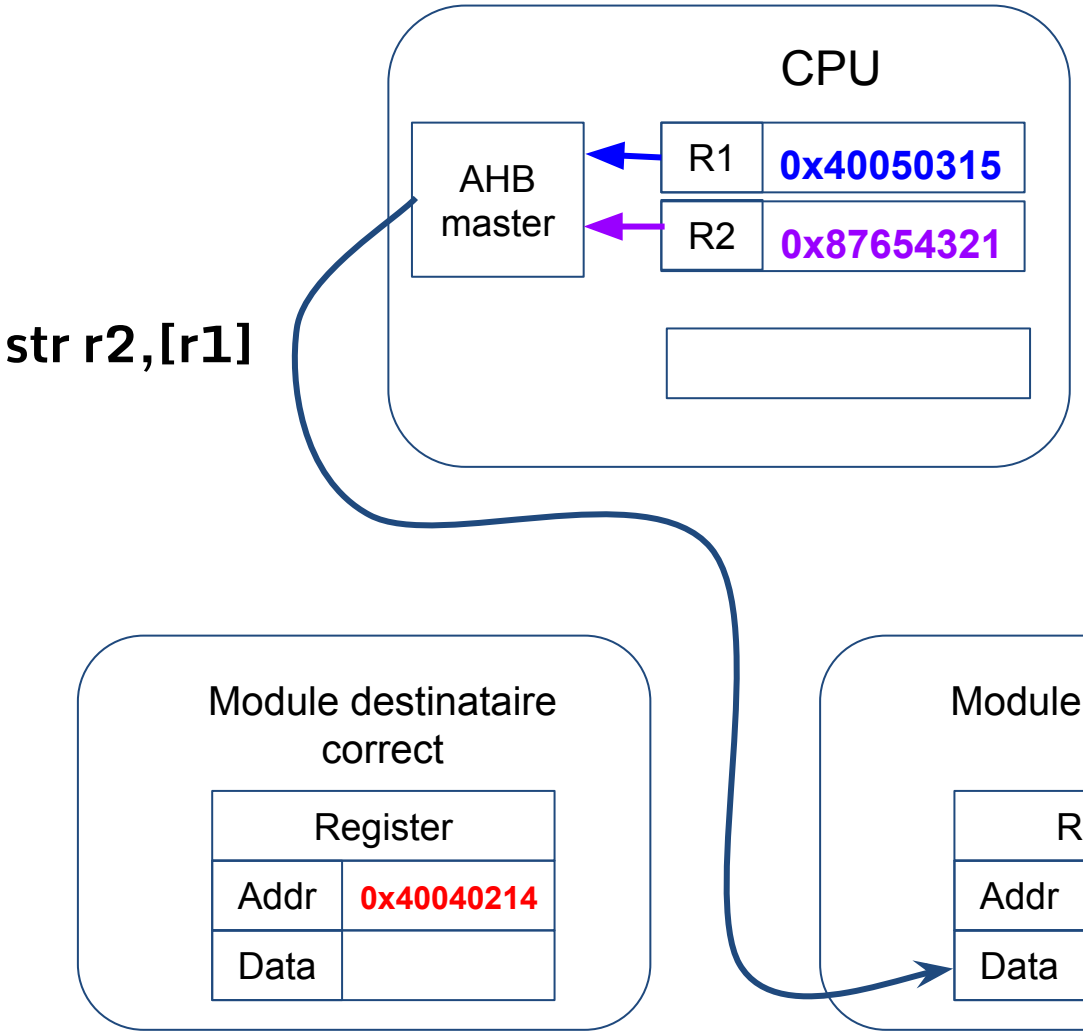
Si PC est décalé d'une certaine valeur



FLASH	
Address	**Instruction/data**
…	…
@ins 1	instruction 1
@ins 2	instruction 2
@ins 3	instruction 3
…	…
@data 1	0x40040214
@data 2	0x12345678
@data 3	0x40050315
@data 4	0x87654321
…	…

Conséquence de décalage de PC - adressage relatif au PC

Si PC est décalé d'une certaine valeur



Les instructions à l'adressage relatif au PC sont vulnérables au décalage de PC

FLASH	
Address	**Instruction/data**
…	…
@ins 1	**instruction 1**
@ins 2	**instruction 2**
@ins 3	**instruction 3**
…	…
@data 1	**0x40040214**
@data 2	**0x12345678**
@data 3	**0x40050315**
@data 4	**0x87654321**
…	…

Conclusions techniques de l'année

1. Il est facile de perturber le Program Counter par BBI

2. Les instructions à adressage relatif au Program Counter sont vulnérables au décalage de PC

=> Il ne faut pas utiliser ce type d'instruction dans des applications de sécurité.

3. Il est facile de perturber l'écriture dans SRAM

Perspectives et formations réalisées

Que va-t-on faire dans la suite ?

- Expliquer les mécanismes des fautes découvertes

- Communiquer ces résultats à la communauté

- Généraliser à EMFI / attaque par glitch d'alimentation

- Démonstration d'attaque (bypass vérification de pin code)

- Proposer des contre-mesures

ICO
Institut Cybersécurité Occitanie

Merci !