



## Chargement dynamique d'extensions pour le pilote du GPU Mali

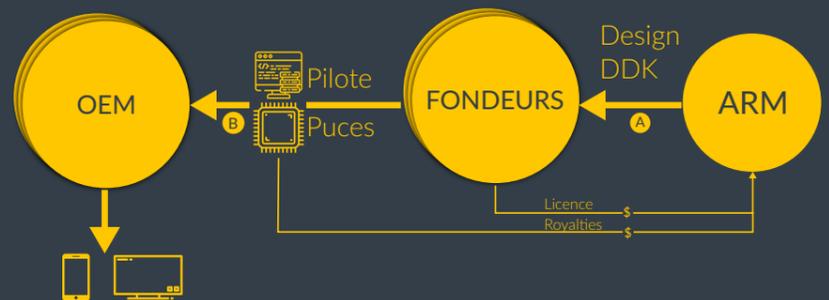
### MOTIVATION DU STAGE

#### A CODE SOURCE VEROUILLÉ

En plus du design de la puce, ARM vend aussi le DDK permettant aux fondeurs de compiler un pilote à fournir aux OEM.

#### B BESOIN DE FLEXIBILITÉ POUR LES CLIENTS INDIRECT

Les OEM achètent les puces, et obtiennent le pilote sous forme binaire. Si jamais ils souhaitent utiliser un système non prévu par ARM, ils doivent passer par le fondeur pour négocier le support. Ainsi ils ne peuvent pas utiliser facilement leurs implémentations maison pour les systèmes de fenêtres.



Fonctionnement du modèle de licence d'ARM

### SUJET

#### SYSTÈME DE PLUGIN

Création d'une couche permettant de charger dynamiquement l'implémentation du système de fenêtres. Modification du système de compilation pour tester cette fonctionnalité.

#### REMANIEMENT DU CODE

Les modifications apportées doivent permettre de retirer du code devenu inutile afin d'alléger la maintenance du pilote.

#### APPLICATION SUR UN CAS RÉEL

Modification du système de fenêtre Wayland pour permettre son chargement en tant que plugin. Remaniement du code pour utiliser des API natives au noyau Linux plutôt que le protocole propriétaire d'ARM.

#### INTÉGRATION DANS LE PILOTE

Intégration dans le tronc du pilote après approbation par les ingénieurs. Qualité du code assurée par la relecture et les tests.

### BÉNÉFICES DU STAGE

#### EXPERTISE EN C

Développement de méthodes de travail utilisables sur des bases de code volumineuses et peu outillée. Insertion dans un projet important : 300k lignes de c, 70k lignes de header et 150k lignes de commentaires.

#### MÉTHODES D'ORGANISATION INDUSTRIELLES

Découverte des processus permettant d'assurer la qualité du code : review, intégration continue, analyse statique, etc. Introduction à la méthode agile (scrum).

Apprentissage des méthodes et moyens permettant d'assurer la circulation de l'information dans une grande organisation.

#### ACCÈS AUX CONFÉRENCES DE L'ENTREPRISE

Initiation à la micro-architecture des GPU Mali : sous-composants, chemin de données, et fonctionnement global du GPU.

Aperçu des failles majeures dans les micro-architectures (Rowhammer, Spectre, Meltdown).

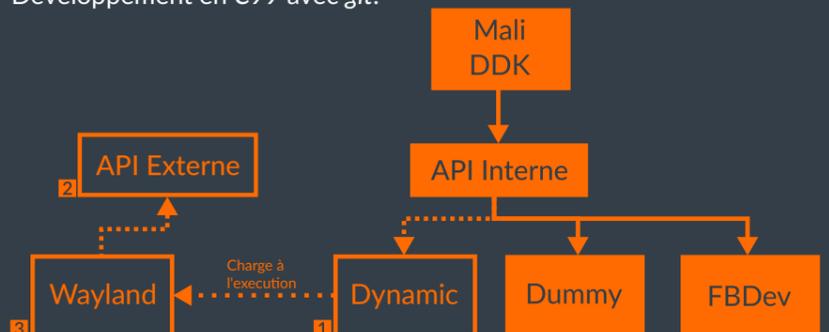
Sécurisation et contrôle d'accès sur les bus CAN.

#### AFFINAGE DE MON PROJET PROFESSIONNEL

Découverte du monde industriel et du fonctionnement d'une grande organisation me permettant d'écarter la grande industrie de mon projet professionnel.

### TRAVAUX RÉALISÉS

Développement en C99 avec git.



#### 1 MODIFICATIONS DU PILOTE

Modification du driver pour permettre le chargement d'un système de fenêtre à l'exécution. Chargement d'une librairie partagée à l'exécution et versionnement de l'interface utilisée pour exposer des fonctionnalités.

#### 2 OUVERTURE DE CODE

Création d'une API permettant d'implémenter son propre système de fenêtre pouvant être diffusée sous une licence plus permissive que le reste du Driver Development Kit (DDK)

#### 3 DÉMONSTRATION SUR UN SYSTÈME EXISTANT

Modification du système de fenêtre Wayland pour valider le fonctionnement du système de plugin. Migration sur CMake pour assurer un découplage maximal entre le DDK et l'implémentation du plugin afin de représenter le cas d'un client sans le code source du DDK.

**ARM** : Advanced Risc Machine

**API** : Application Programming Interface

**GPU** : Graphical Processing Unit

**OEM** : Original Equipment Manufacturer

**DDK** : Driver Development Kit

**CAN** : Controller Area Network