Linux/GNU

évaluation des avantages et des inconvénients de Yocto par rapport à Buildroot

---

YOCTO

Colin Baumgard

Paul-Antoine Le Tolguenec

May 2, 2020

Contents

List of Figures

List of Tables

1 Introduction

Dans le monde de l'embarqué, il est souvent utile de créer son propre système d'exploitation. En effet, les contraintes imposées par le système sont diverses et variées en fonction du projet. La construction d'un OS peut être longue et délicate. C'est pourquoi différents framework sont nées pour facilité cette construction d'OS personnalisé. Le plus connus est Buildroot. Buildroot est un ensemble d'outil permettant le développement des principaux éléments d'un OS. Mais dans l'industrie, un nouveau projet a vu le jour. La création d'un système d'exploitation personnalisé sur Linux conçu pour les spécificités d'une carte embarquée et les exigences du futur produit peut s'avérer difficile, mais le projet Yocto a pour mission d'aider. Le projet Yocto est un projet collaboratif en exploitation libre géré par la Fondation Linux. Dans cet article, nous allons tenter d'évaluer les avantages et inconvénients de chacun de ces projet en fonction des différents systèmes. Puis nous allons synthétiser le déroulement de la construction de l'OS pour les deux projets.
Repo GitHub

`https://github.com/Paul-antoineLeTolguenec/Linux-GNU.git`

2 De Buildroot à Yocto

2.1 Buildroot

Buildroot est un outil qui simplifie et automatise le processus de construction d'un système Linux complet pour un système embarqué, en utilisant la compilation croisée.

Pour y parvenir, Buildroot est capable de générer une toolchain, un RFS (root file system), un kernel Linux et un bootloader pour la cible. Buildroot peut être utilisé pour toute combinaison de ces options, indépendamment (vous pouvez par exemple utiliser une chaîne de compilation croisée existante, et ne construire que votre système de fichiers racine avec Buildroot). Ce qui permet de simplifier le processus en laissant tout de même beaucoup de liberté.

Les systèmes embarqués utilisent souvent des processeurs qui ne sont pas les processeurs x86 habituels que tout le monde est habitué à avoir dans son PC. Créer son OS pour une architecture que peu de gens utilise s'avére très fastidieu.Il peut s'agir de processeurs PowerPC, de processeurs MIPS, de processeurs ARM, etc. Buildroot rend le dévellopement del'OS adapté à ces processeurs très accessible.

2.2 Yocto

Yocto intègre des parties développées conjointement pour OpenEmbedded, notamment BitBake, OpenEmbedded-Core et d'autres métadonnées. Les éléments développés dans le cadre du projet, appelés "meta-yocto" et "meta-yocto-bsp", comprennent l'intégration des éclipses. Ensemble, ils améliorent les outils d'OpenEmbedded, cette plateforme de référence pour la construction de systèmes avancés embarqués dans les HW est connue sous le nom de Poky.

Pour développer des logiciels, nous avons besoin d'une chaîne d'outils (croisés) : les fichiers sources et les instructions sur la façon de les compiler. C'est suffisant pour une source. Pour plus de composants et de dépendances dans la compilation et le temps d'exécution, il faut augmenter la complexité et des étapes supplémentaires. Bitbake est un agent ayant la

capacité d'interpréter et d'exécuter les recettes d'amélioration, il calcule la chaîne des tâches nécessaires pour développer l'objectif défini et exécuté.

3 Discussions et Conclusion

ecris ici

References