Linker definition file Improvement Investigation

For an easier integration

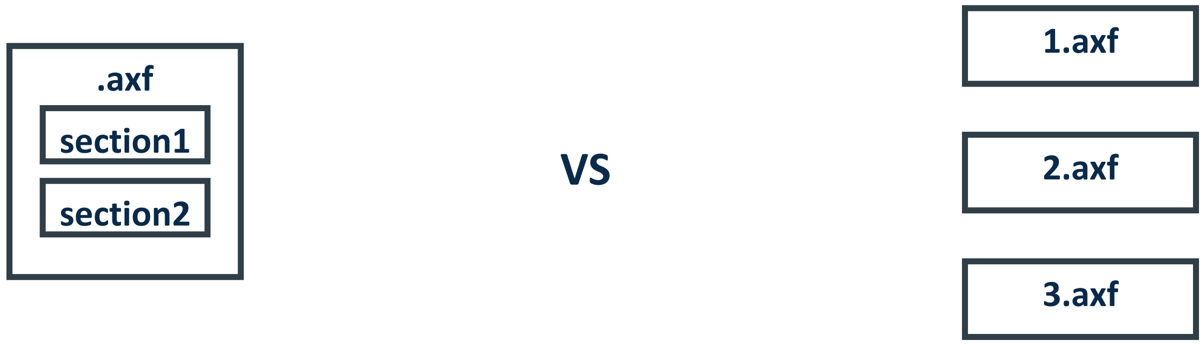
Ken Liu
May 13

Introduction

- 3 supported linker definition file (.ld, .sct and .ict).
- Most of the platform uses the common linker definition, so this common files contains some specific items.
- Isolation level 1 and 2 are using wildcards collection to collect partition symbols – no templating is needed, good for straight integration.
- Isolation level 3 is still using templating for per-partition isolation boundaries.

Why per-partition isolation boundary needed

- For isolation level 3 requirement.
- We are now assembling all components into one image.
  - Multiple image needs to involve PIE (Position Independent Execution) mechanism.
    - Pure PIE increases RAM/ROM size.
    - Para-PIE can work well but may be not easy for all toolchains.
    - Para-PIE also requires assembler tool support.
- So only choice is to use toolchain sections for these boundaries.



.axf

section1

section2

VS

1.axf

2.axf

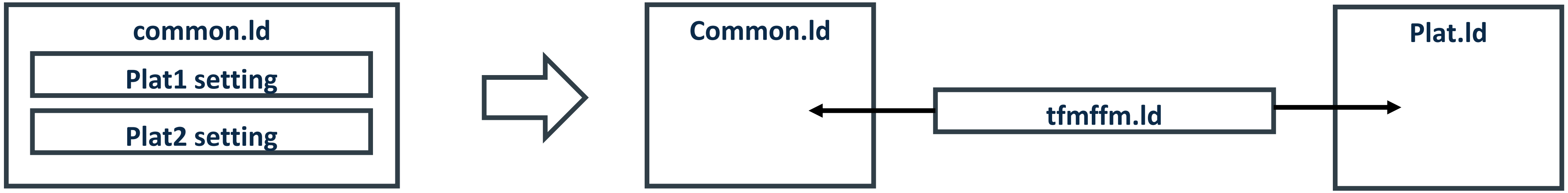
3.axf

Needs templating

Needs image assembling tool

Why the existing solution is not perfect?

- Common linker definition file provides the simplest case.
  - It is a reference.
  - It is not welcome to put platform specific items into it – platform complexity.
- Ideally, each complex platform has its its own linker definition file.
  - Include additional TF-M sections when necessary, including sub-items.
  - Inclusion may not be supported by a toolchain or has a complicated use.
- TF-M can introduce new tools but think about the supply chain:
  - Involving other 3rd party tools may lead to supply chain concern, python for example.
  - IDE can call programs but may not be easy for python scripts.



Example

```
; <lbtp leadingskip=2 suppressed=\/*,\*/> suppress the C comment tags.
; ?? +0 ALIGN32 {
;     *?tfm_app_rot_partition*?.* (+RW +ZI)
; }
; </lbtp>
; <lbtp skipall>
tfm_app_rot_domain_priv +0 ALIGN32 {
    *tfm_app_rot_partition*.* (+RW +ZI)
}
; </lbtp>
```

Command line:

```
parser ./generated/platform/ext/common/gcc/tfm_isolation.ld
          ./generated/platform/ext/common/gcc/tfm_isolation_out.ld
           --lbtpmore="@"
          ./secure_fw/partitions/protected_storage/libtfm_app_rot_partition_ps.a -o tfm_s.axf
```

Dependencies

- Using a python would involve python and its packages.
  - Risks?

- TF-M tool in source code and pre-built executable file?
  - How do you guarantee the integrity?
  - Efforts on maintaining such a tool.

- For those HOST ENV is not windows/linux, they can build it.
  - But needs the HOST complier – is this another dependency?

Summary

- It is at very beginning.
  - A specification is going to be public.
- Everything is open.
  - Provide your idea and file a proposal.
- Feedbacks and Q&A

Thank You
Danke
Gracias
谢谢
ありがとう
Asante
Merci
감사합니다
धन्यवाद
Kiitos
شكرًا
ধন্যবাদ
תודה