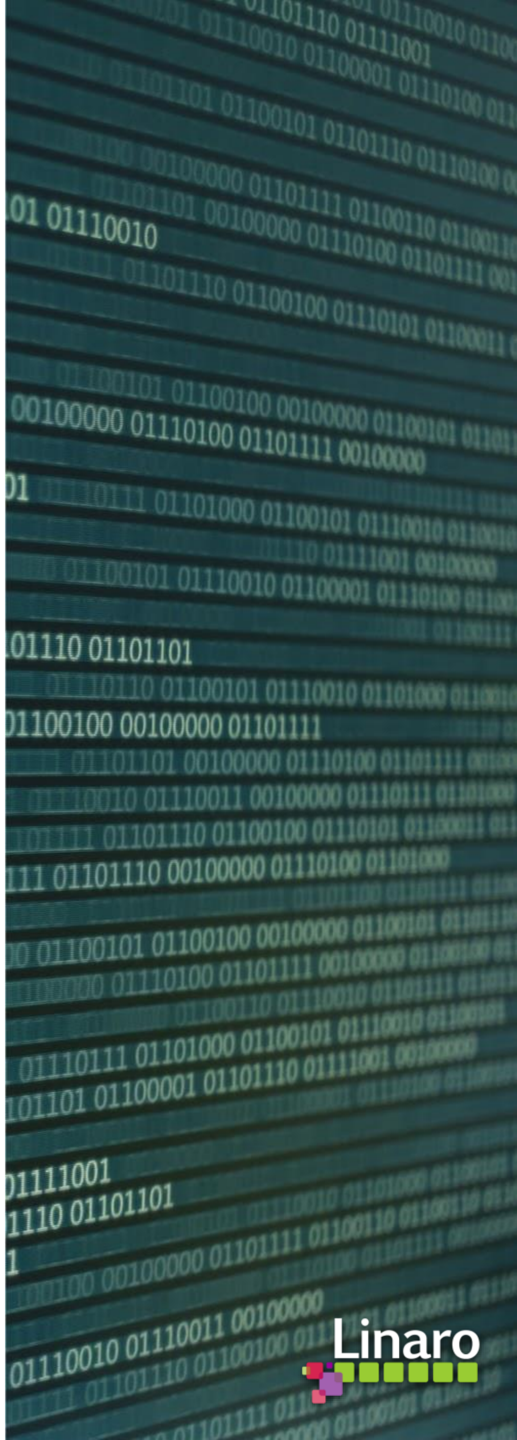
# Open-CMSIS-Pack

Technical Project Meeting 2022-04-12

This meeting is recorded !

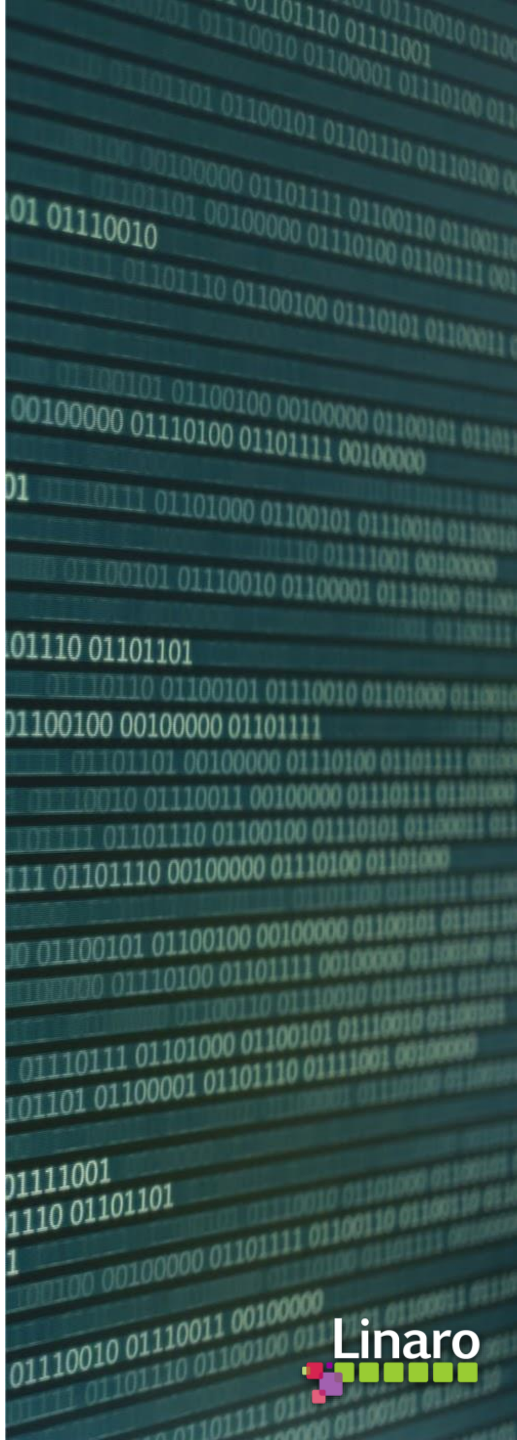
# Agenda

- Welcome
- Request for review
- Next Steps – CMSIS-Toolbox
- Project Template
- Component Selection vs. Configuration
- API, Components, and Interfaces
- Wrap Up



# Request for review:

- [projmgr] Structure of Projects that use *.clayer.yml #113
  - Proposal introduction
  - Indirectly addresses #77 #80 #65 #30
- [projmgr] Using existing .cprj as lockfile #119 (ST)
- [projmgr] Source patterns have to be supported #82 (ST)
- [projmgr] Source type/language specific include #276
- [spec] Component level configuration #26
  - To be discussed (see slide 6)



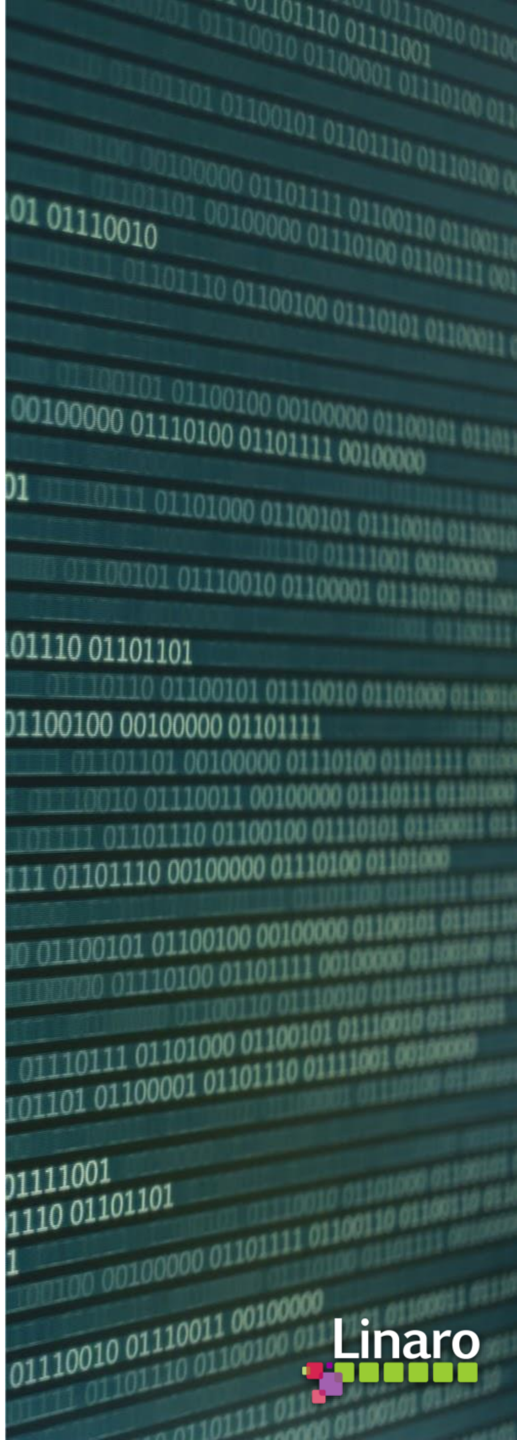
# Next Steps – CMSIS-Toolbox

CMSIS-Toolbox 0.10.0 (targeting April 19th 2022) containing:

- separated archives for every supported platform (Windows, Linux and Mac) replacing the cbuild_install.sh bash installer (deprecated)
- adding packchk (#231)
- CMSIS-Build Manager 0.11.0
  - cbuild binaries replacing the cbuild.sh bash script (deprecated)
- CMSIS-Project Manager 0.9.4
  - rework file references in generated CPRJs, allowing in-source conversion of solutions with layers
  - Bugfixes

**Roadmap:**

- Feature freeze version of CMSIS Toolbox – stabilized until end of May 2022
- During May-July changes/enhancements to PDSC and csolution will be defined
  - https://github.com/Open-CMSIS-Pack/devtools/tree/main/tools/projmgr/docs/Q3Features
- These new features will get implemented in Q3'22



# Project Template

https://github.com/Open-CMSIS-Pack/devtools/tree/main/tools/projmgr/docs/Template

Proposed Structure:

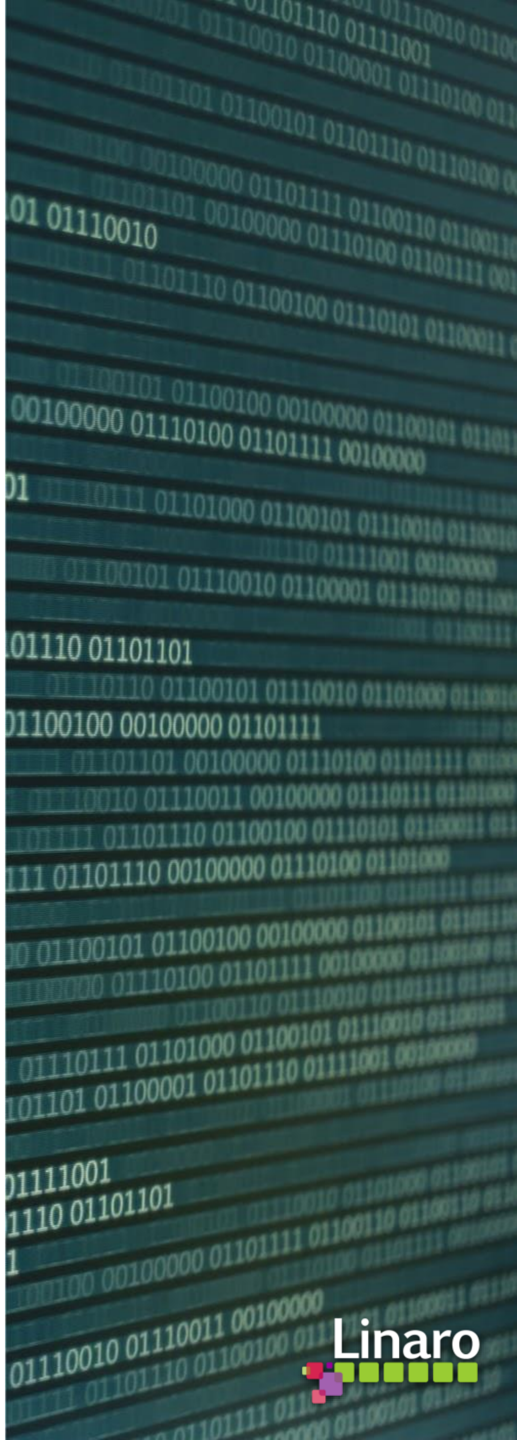
*.cdefault.yml – defines the compiler environment (GCC, AC6, IAR, …)

- ○ This file is not yet there, but the content is in MySolution.csolution.yml (at the top)
- ○ Location of this file tdb, could be in the ctools\etc directory and automatically read by csolution

MySolution.csolution.yml – defines built-types and target-types

- ○ All packs that are used are specified to enable reproducible builds

MyProject.cproject.yml – the actual project content



# Component Selection vs. Configuration – two different steps

https://open-cmsis-pack.github.io/Open-CMSIS-Pack-Spec/main/html/pdsc_components_pg.html#Component_Files

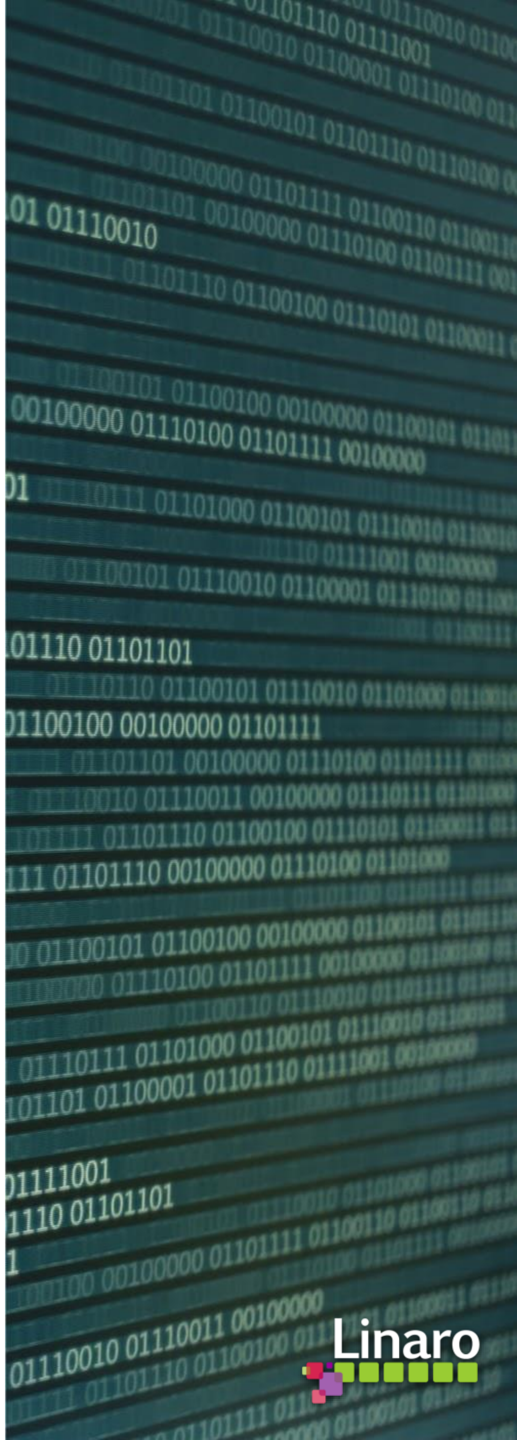
**Component Selection:**

- A set of files defined in the *.pdsc that are added to the project
- Selection is done in IDEs using the RTE window, in csolution using `components:` list
- Variants of a component refer to library or variants with additional debug info/tests
- Configuration files allow to configure the component and are always local to a project

**Component Configuration:** is a separate step

- Configuration files are customized using an editor
- May have Config Wizard annotations
  https://open-cmsis-pack.github.io/Open-CMSIS-Pack-Spec/main/html/configWizard.html
- RTE_Components.h may be used to provide additional config information

- Generators are another option Configuration files are customized using an editor #104
  - For Generators, additional files will be provided in the PDSC file



# API, Components, and Interfaces - better definition #114

API and Components are defined at PDSC (software pack) level

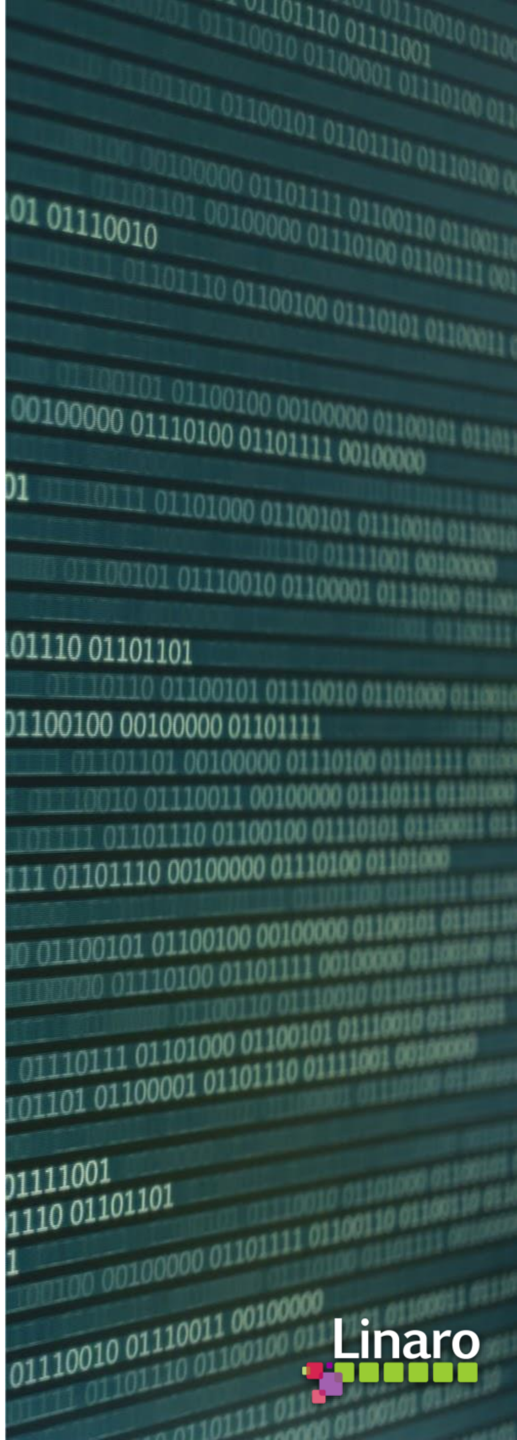
- An **API** is an interface template provided as header file.
  - Example: CMSIS-Driver API    CMSIS-Driver:I2C
- A **Component** is a set of files that implement a functionality
  - A **Component** may be based on an **API** -> API element provides header files of component interface.
  - Example: CMSIS-Driver:I2C:I2C   or CMSIS-Driver:I2C:Custom
  - Going forward, with Instances this may relate to peripherals CMSIS-Driver:I2C:FlexCom1

NEW: Interfaces are defined and requested at cproject/clayer level

- Could refer to Components or APIs in software packs

```
interface:
  provides:
    - component: CMSIS-Driver:I2C:Custom   // could resolve with PDSC conditions
  requires:
    - component: CMSIS-Driver:I2C          // could create a new PDSC requirement
  provides:
    - type: Heap=100000                    // pragmatic interface resource
  requires:
    - type: Heap>50000                     // layer needs pragmatic resource
```

- Together with component selection this creates conditions that select a compatible layer, i.e. {Board}



# Open-CMSIS-CDI: Technical Introduction

**Invitation to attend**

- Arm will host a one-hour public meeting as a technical introduction to Open-CMSIS-CDI

**Topics to cover include**

- Goals of the project
- Proposed Open-CMSIS-CDI APIs
- Evolution of the PSA Firmware Update API
- Development of a common reference implementation: Open IoT SDK
- Tie in to Open-CMSIS-Pack project
- Compatibility and promotion campaign

**EMEA / US Timezone**

- Thursday 21 April 2022
- 17:00 – 18:00 UK BST
- 09:00 – 10:00 PDT

**APAC / EMEA Timezone**

- Friday 22 April 2022
- 16:00 – 17:00 CST
- 09:00 – 10:00 UK BST

People on this call have the option to join this meetings and are automatically invited

# Wrap Up

- Action all:
  - Your feedback on the proposals as well as the questions raised are essential
- Due to Easter-holidays there will be no Open-CMSIS-Pack Technical Meeting next week 19th April 2022
- Join Open-CMSIS-CDI Technical Introduction on April 21 or April 22
- Next Open-CMSIS-Pack meeting: 26th April 2022 @ 16:00 CET (15:00 UK)

# Thank you

Linaro