Open-CMSIS-Pack

Technical Project Meeting 2023-11-07

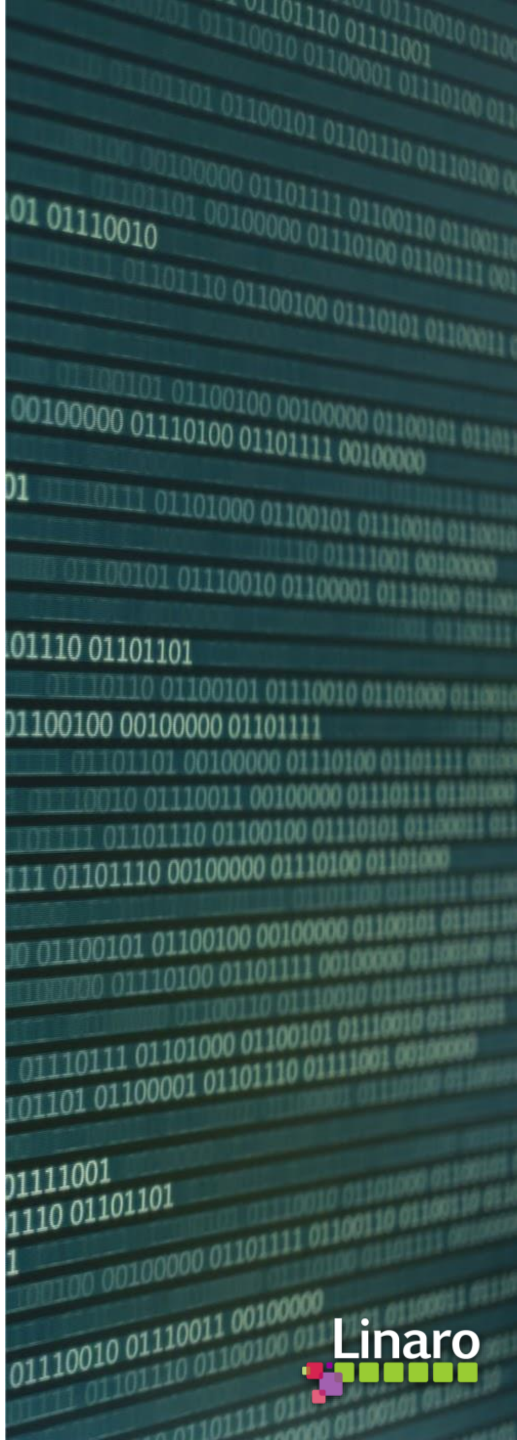
This meeting is recorded !

Open-CMSIS Pack

Linaro

Agenda

- Welcome
- Project Boards
- Issues for Review
- Attract 3rd party software vendors
- Wrap Up



Boards:

- Open-CMSIS-Pack Specification Change Board

- CMSIS-Toolbox 2.2 Project Board
  - See progress and issues in scope for version 2.2.0
  - Please review and provide feedback in case you see topics missing
  - Add issues or comment on existing issues that you think should be added to 2.2.0

- CMSIS-Toolbox 2.3 Project Board

Issues to Review

- [csolution] `list generators` command display directories and `contexts` info #1171
- [csolution] Explicit section of Automatic Linker Script Generator #1175
- Implement <csolution-name>.cbuild-pack.yml #1122 (ST) (PR #1143)
    - Document cbuild-pack.yml PR #76 - in progress
- A `pack:` specified with `path:` in YAML input takes precedence over regularly installed pack version #1098
- [csolution] Should we change extension to `*.yaml` of generated files? #1186
- [csolution] Ensure that *.csolution.yml projects are usable #1177
- [csolution] output for component selection is misleading #1187

Issues to Review (cont’d)

- [csolution] Using packs via local_repository.pidx #1172
- [RTE] provide access to component version from the application #1154
- [csolution] add support for toolchain agnostic `pre-include` files in YAML #808

Defects:

- [csolution] access sequence `$Dname$` undefined if only a board is specified #1184
- [csolution] $Dname$/$Bname$ differ from RTE converted file and path names (‘-’ replaced by ‘_’) #1185

What is required to get 3rd party software in packs

- Better Guidance for 3rd party vendors; https://github.com/Open-CMSIS-Pack/SW-Pack-HandsOn is insufficient as 3rd party vendors do not understand the workflow
  - ✓ Scripts for pack creation and tutorials are there
- Understanding how a software components are used to create an application
- To achieve flexibility for 3rd party vendors, alignment on API interfaces that can be used
- Cclass Taxonomy Management to ensure compatibility
- Pack datasheet generator that shows how the software appears to the user
- Better IDE workflows, potentially leveraging pack datasheets (i.e. via a web service)

Cclass Taxonomy Management

- github.com/Open-CMSIS-Pack/Open-CMSIS-Pack-Taxonomy
- New Cclass issue template:
  - https://github.com/Open-CMSIS-Pack/Open-CMSIS-Pack-Taxonomy/issues/new?assignees=&labels=new+Cclass&projects=&template=cclass_request.yml
- Example: Cclass CMSIS
  - https://github.com/Open-CMSIS-Pack/Open-CMSIS-Pack-Taxonomy/issues/1

Create applications from components

- New Section in Documentation
- Could we get feedback on what is missing? #1189
- Can we work on tutorials and examples?
  - What should be used, i.e. based on LwIP, FreeRTOS, AWS components?
- Can we get help on pack datasheets? #262
- How should complex examples be published?
  - Via links to github? Similar to https://github.com/Arm-Examples
  - Could we change the approach to examples by leveraging clayers for boards?

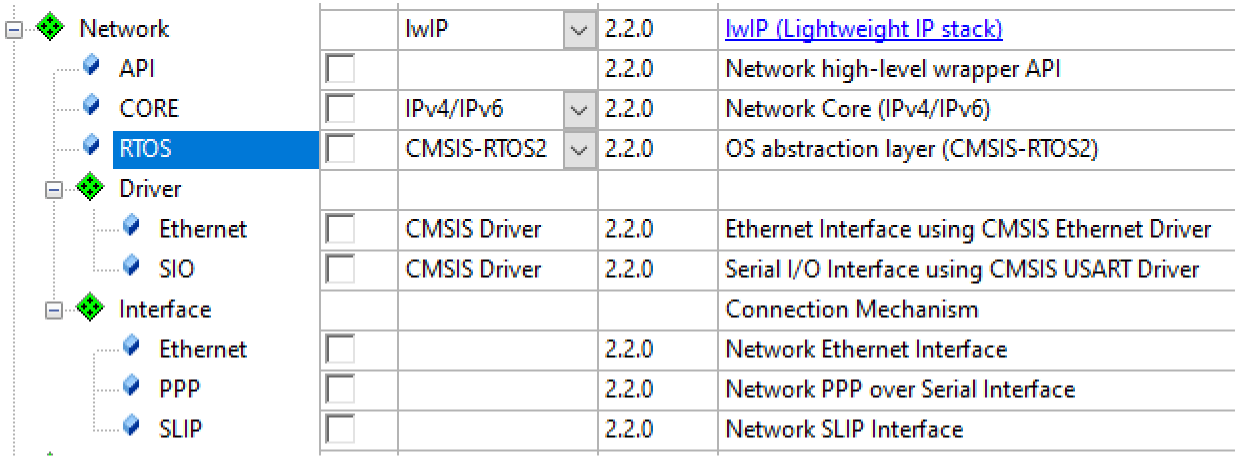
Pack Datasheet #260: github.com/Open-CMSIS-Pack/lwIP

- Generated “Pack Datasheet” based on *.PDSC XML data

- Possible “datasheet” content based on PDSC file
  - Pack
  - Link to documentation
  - License
  - Keywords (for search)
  - Components
  - External dependencies (required components)
  - Exposed header files (API)
  - Configuration files, user code template files, examples
  - Release history

- Not directly possible (based on PDSC file) but potential useful
  - Mandatory components (that a user must select)
  - Provided interfaces (i.e. BSD socket)
    - Maybe exposed header files is sufficient?
  - Overview diagram (could be part of documentation)
  - Related components (i.e. Crypto, IoT socket, Cloud stacks, RTOS kernels)
    - Maybe by listing packs that require components from this pack

Pack Datasheet example *needs more work*: https://github.com/ReinhardKeil/lwIP/tree/patch-1



Network		lwIP	2.2.0	lwIP (Lightweight IP stack)
API	□		2.2.0	Network high-level wrapper API
CORE	□	IPv4/IPv6	2.2.0	Network Core (IPv4/IPv6)
RTOS	□	CMSIS-RTOS2	2.2.0	OS abstraction layer (CMSIS-RTOS2)
Driver				
Ethernet	□	CMSIS Driver	2.2.0	Ethernet Interface using CMSIS Ethernet Driver
SIO	□	CMSIS Driver	2.2.0	Serial I/O Interface using CMSIS USART Driver
Interface				Connection Mechanism
Ethernet	□		2.2.0	Network Ethernet Interface
PPP	□		2.2.0	Network PPP over Serial Interface
SLIP	□		2.2.0	Network SLIP Interface

Pack cross reference

Used by:

- pack: MDK-Packs::IoT_Socket

lwIP::lwIP

uses:

- component: CMSIS:RTOS2
- component: RTOS&FreeRTOS:Core
- component: CMSIS Driver:Ethernet
- component: CMSIS Driver:Ethernet MAC
- component: CMSIS Driver:Ethernet PHY
- component: CMSIS Driver:USART

Wrap Up

Is anyone preparing/working on a topic to present and discuss in the coming weeks?

- Please contact Joachim.Krech@arm.com ahead of the meeting

Next Open-CMSIS-Pack meeting: 14th Nov 2023 @ 16:00 CET (15:00 UK)

Thank you

Linaro