



ECLIPSE
2020 CON

Eclipse CDT 10.0 and beyond

Alexander Fedorov, ArSysOp

Who am I?

Alexander Fedorov @ ArSysOp



 Eclipse Platform Committer

 Eclipse CDT Committer

 Eclipse Passage Project Lead

Leading IDE effort
at Software Task Group



A person is working at a desk in a dimly lit office. They are wearing glasses and have their hand on their chin, looking at a large monitor. There are other monitors and papers on the desk. The background is a brick wall.

Why Eclipse CDT is the best choice?

Project Maturity

- 18 years of active development
- About 35k of commits
- About 100 contributors
- 2.3M LOC in the main repo

and Functionality

- Configure and Build
- Navigate and Refactor
- Analyse and Fix
- Debug and Run
- Extend and Integrate

High Configurability

The image displays the Eclipse IDE interface with three windows open, illustrating the configuration of a RISC-V toolchain.

Left Window: Properties for rtc_func_riscv_rv32m1_vega

- Settings** (Configuration: debug [Active])
 - Tool Settings
 - Toolchains**
 - Name: GNU MCU RISC-V GCC (riscv-none-embed-gcc)
 - Architecture: RISC-V
 - Prefix: riscv32-unknown-elf-
 - Suffix:
 - C compiler: gcc
 - C++ compiler: g++
 - Archiver: ar
 - Hex/Bin converter: objcopy
 - Listing generator: objdump
 - Size command: size
 - Devices
 - Container Settings
 - Build Settings

Right Window: Properties for rtc_func_riscv_rv32m1_vega

- Preprocessor Include Paths, Macros etc.** (Configuration: debug [Active])
 - Entries
 - Setting Entries
 - CDT User Setting Entries
 - Exported Entries from Referenced Projects [Shared]
 - CDT Managed Build Setting Entries [Shared]
 - CDT RISC-V Cross GCC Built-in Compiler Settings**
 - /home/jonah/riscv/vega/boards/rv32m1_vega/demo_apps
 - /home/jonah/riscv/vega/boards
 - /home/jonah/riscv/vega/boards/rv32m1_vega/demo_apps/rtc_func
 - /home/jonah/riscv/toolchain/riscv32-unknown-elf-gcc/lib/gcc/riscv32
 - /home/jonah/riscv/toolchain/riscv32-unknown-elf-gcc/lib/gcc/riscv32
 - /home/jonah/riscv/toolchain/riscv32-unknown-elf-gcc/riscv32-unknown-elf-gcc
 - /home/jonah/riscv/toolchain/riscv32-unknown-elf-gcc/riscv32-unknown-elf-gcc
 - # CPU_RV32M1_riscy=1

Bottom Window: Preferences

- Global RISC-V Toolchains Paths**
 - Configure the locations where various GNU RISC-V toolchains are installed. The values are stored within Eclipse. Unless redefined more specifically, they are used for all projects in all workspaces.
 - Default toolchain: GNU MCU RISC-V GCC
 - Toolchain name: GNU MCU RISC-V GCC

Bottom Left Window: Global ARM Toolchains Paths

- Global ARM Toolchains Paths
- Global Build Tools Path
- Global Jumper Path
- Global OpenOCD Path
- Global pyOCD Path
- Global QEMU Path

Easy Extensibility

Eclipse Platform

- Workspace hooks
- Natures and builders
- Annotations and markers
- Actions and controls
- View and perspectives
- Debug model
- Launch configurations

Eclipse CDT

- Project templates
- Build definitions
- Error parsers
- Code formatters
- Doc comment tools
- Debug info visualizers
- Indexer policies

SimRel Components

- Eclipse Modeling Framework
- Eclipse Orbit and Eclipse Platform
- Eclipse Parallel Tools Platform (PTP)
- Eclipse EGit and Eclipse Mylyn
- Eclipse TM4E and Eclipse Wild Web Developer
- Eclipse LSP4J and Eclipse LSP4E
- Eclipse Linux Tools

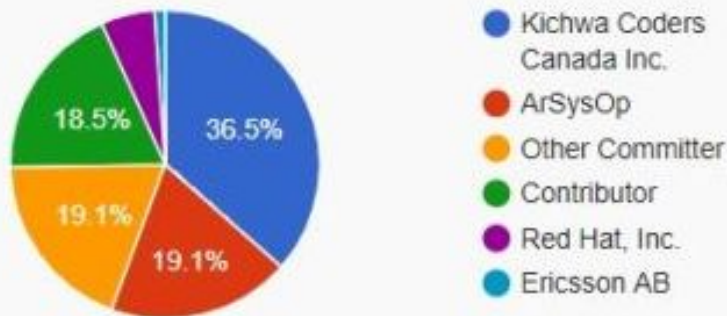
Who is using Eclipse CDT?

Nios II IDE 	MCUXpresso 	Code Warrior 	TrueStudio 	Sloeber (for Arduino) jantje	Kalray 	iDev 
VX Software 	Artik IDE 	e2 studio 	Momentics 	Code Composer 	Sourcery CodeBench 	Simplicity Studio 
DAVE 	Xtensa Xplorer 	DS-5 	CrossCore (CCES) 	eGui 	Ascet Developer 	Cevelop 
XSDK 	Luminosity 	SoftConsole 	Snapdragon Debugger 	Wind River Workbench 	System Workbench 	COSIDE® 

Who is actively contributing?

Organization Contribution Activity:

Commits on this project by supporting organization over the last three months.



Active Member Companies:

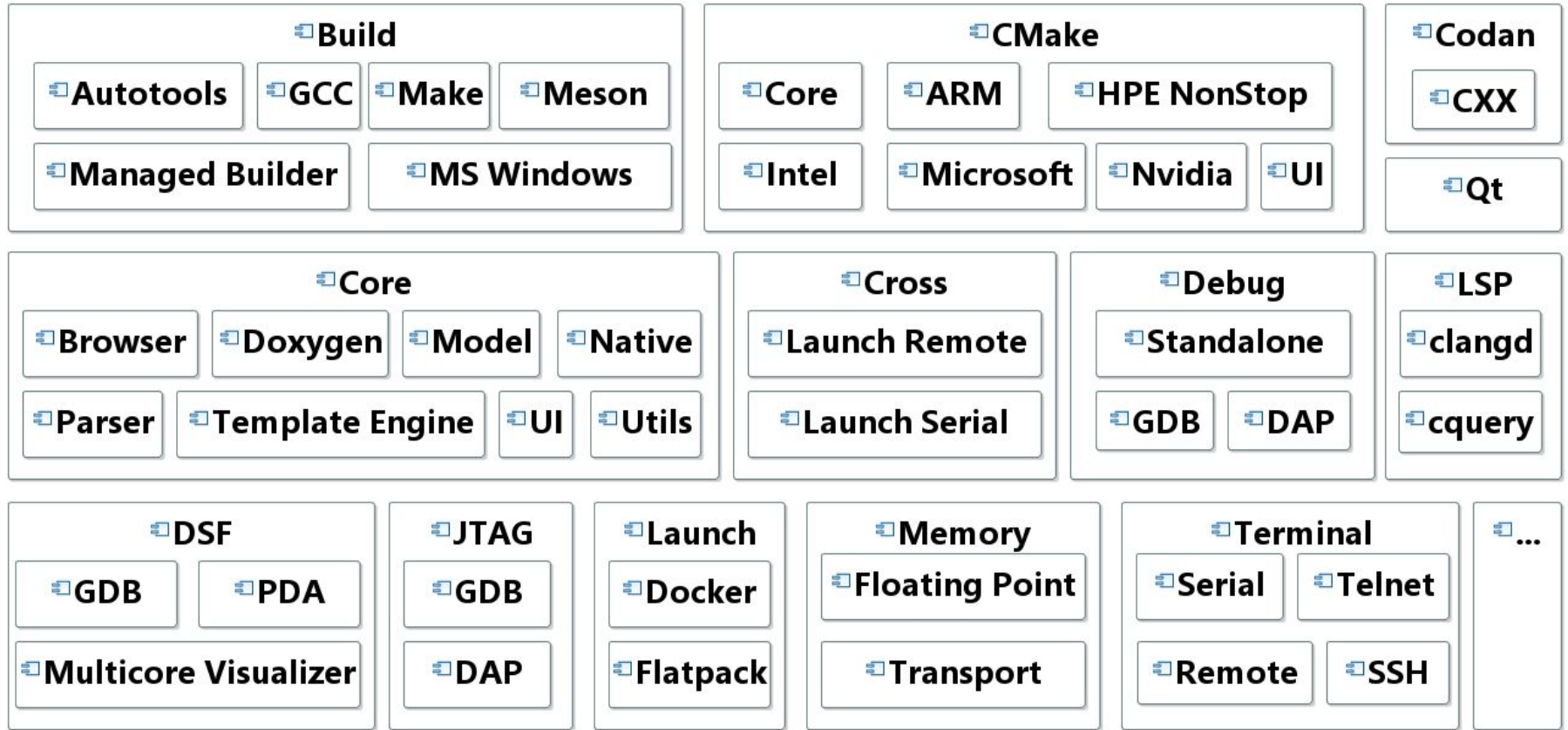
Member companies supporting this project over the last three months.



A woman with dark hair and glasses is sitting at a desk, looking at a laptop screen. Her hands are on the keyboard. The desk has some papers and a glass of water. The background is blurred, showing shelves with books or files. The overall lighting is dim, with a warm orange glow from the laptop screen.

What's changed in Eclipse CDT 10.0?

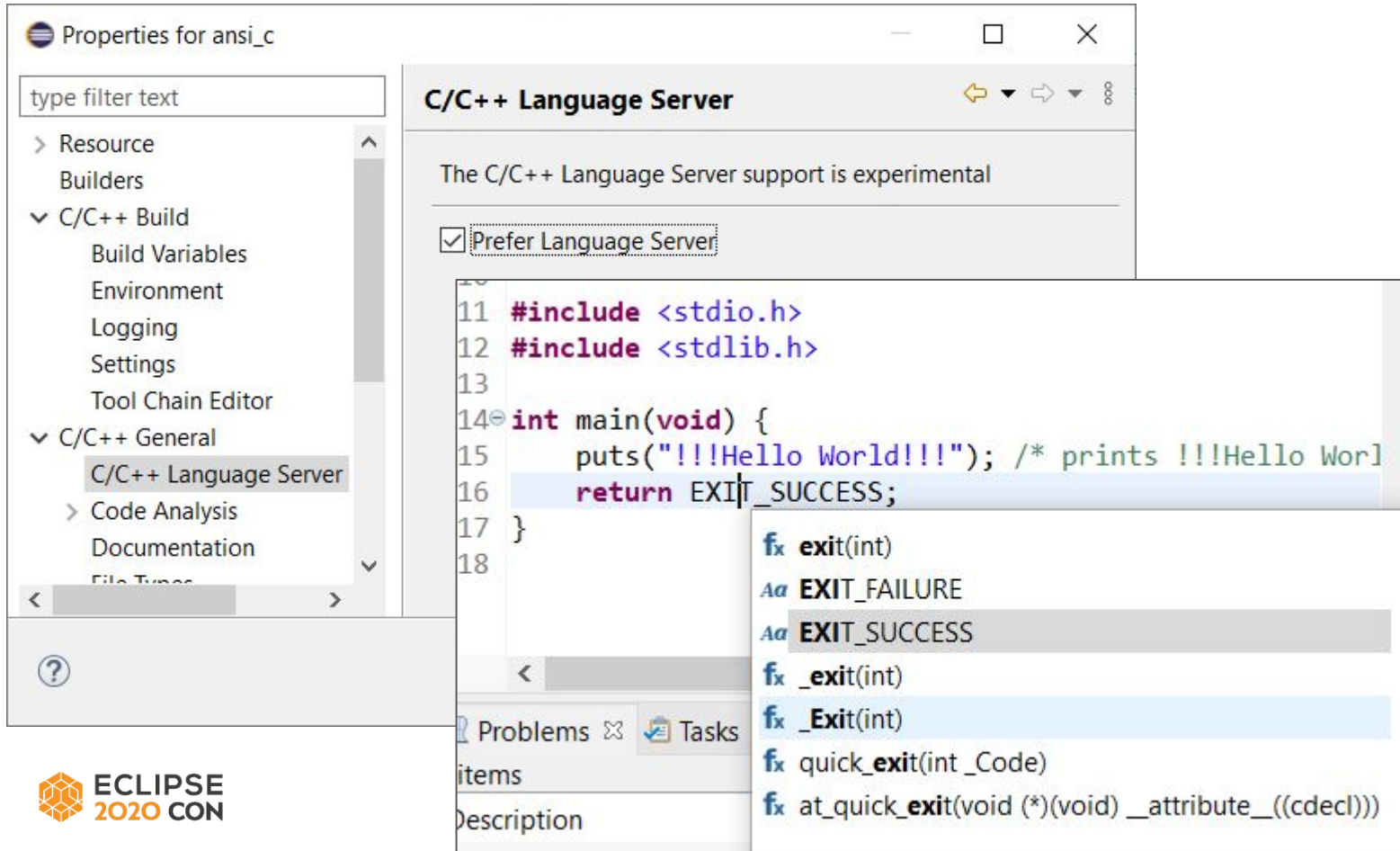
Eclipse CDT 10.0 Overview



CMake CDT Indexer Support

- Origin: <https://github.com/15knots/cmake4eclipse>
- Better user experience in C/C++ editors:
 - a. code completion
 - b. jump to declaration
 - c. folding of inactive `#ifdef`'s
- Employs compilation database (`compile_commands.json`)
- Compiler built-in detection
- Best for gcc, gcc based cross-compilers and clang
- Extension point to support commercial compilers

Language Server Support (experimental)



The screenshot shows the Eclipse IDE interface. On the left, the 'Properties for ansi_c' dialog is open, with the 'C/C++ General' tab selected. Under 'C/C++ General', the 'C/C++ Language Server' option is checked. The main editor displays a C program snippet:

```
11 #include <stdio.h>
12 #include <stdlib.h>
13
14 int main(void) {
15     puts("!!!Hello World!!!"); /* prints !!!Hello World! */
16     return EXIT_SUCCESS;
17 }
18
```

A tooltip is visible over the `EXIT_SUCCESS` constant, showing a list of related symbols:

- `exit(int)`
- `EXIT_FAILURE`
- `EXIT_SUCCESS` (highlighted)
- `_exit(int)`
- `_Exit(int)`
- `quick_exit(int _Code)`
- `at_quick_exit(void (*)(void)) __attribute__((cdecl))`

Important!

Use
Generic
Editor



Eclipse CDT 10.0 Breaking Changes

- Requires Java 11 as a minimum.
- Environment Variables are always case sensitive in CDT.
- Environment variables no longer support `\${` to avoid expanding.
- Arduino plug-ins and features removed.
- LRParser, XLC and UPC removed.

Learn more

https://help.eclipse.org/2020-09/index.jsp?topic=%2Forg.eclipse.cdt.doc.isv%2Fguide%2Fdeprecated_API_removals.html

Eclipse Embedded CDT

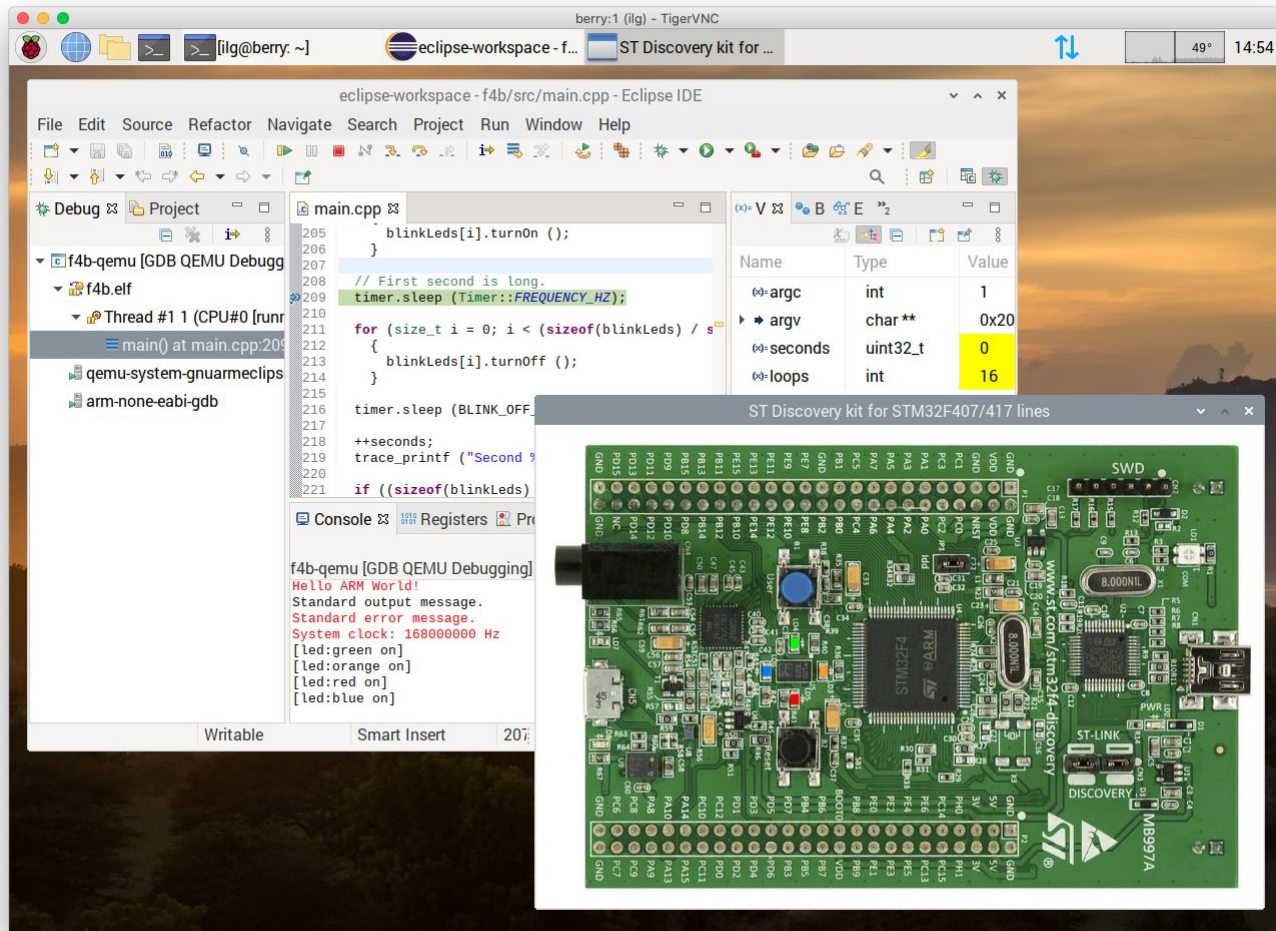


Eclipse Embedded CDT

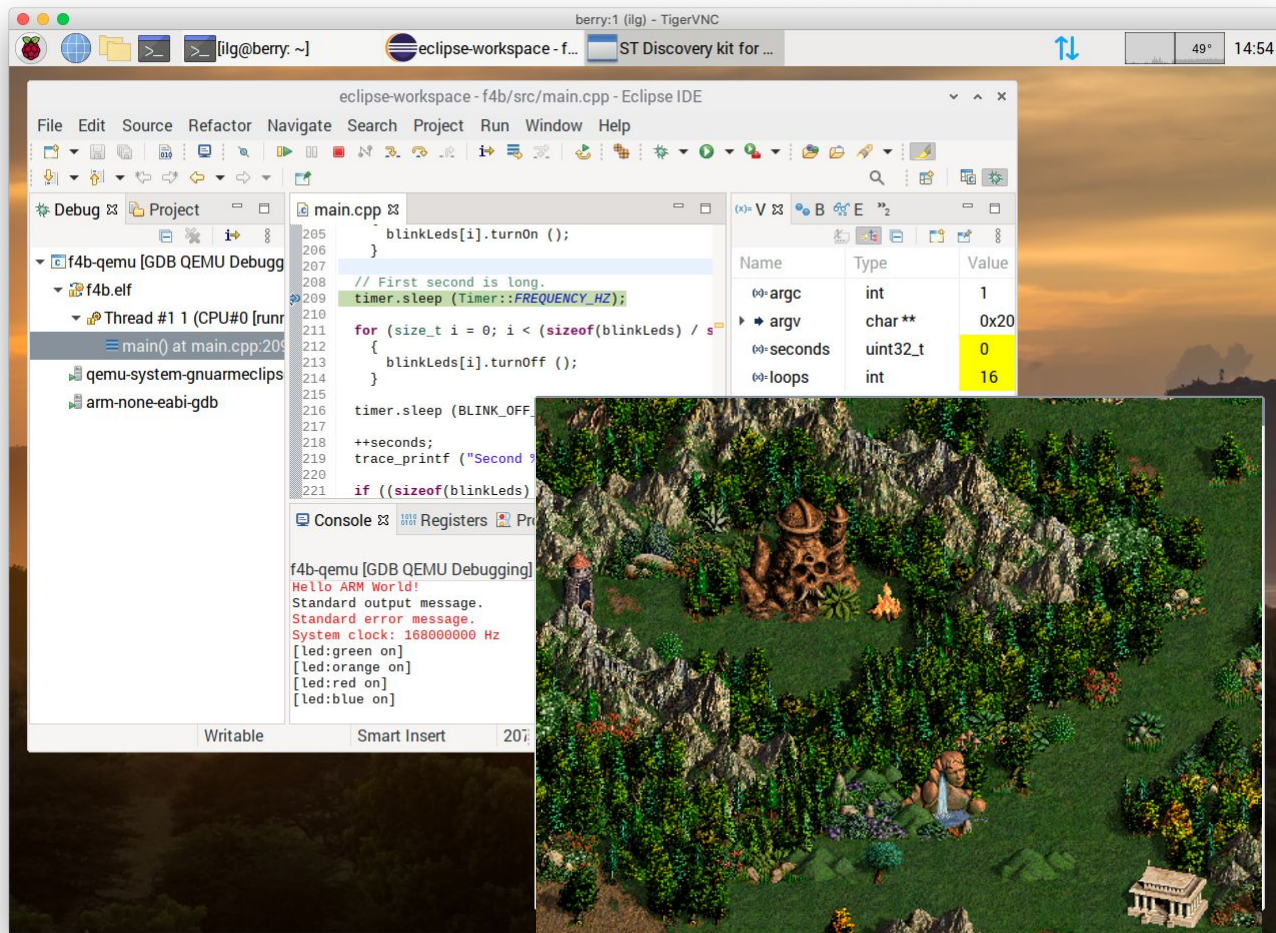
- create/build/manage embedded ARM/AArch64/RISC-V applications
- ready to run templates for some ARM Cortex-M processors
- debugging support via JTAG/SWD
- examine and modify peripheral registers during debug sessions
- supports a wide range of 32 and 64-bit toolchains

Started to supply “Embedded C/C++” package for 2020-12!

Learn more <https://projects.eclipse.org/projects/iot.embed-cdt>



Eclipse Embedded CDT on Raspberry Pi 4 (based on Eclipse Platform for Aarch64)



Eclipse Embedded CDT on Raspberry Pi 4 (not a toy!)

Eclipse CDT and Open Hardware Group

CORE-V IDE



OPENHW GROUP
— PROVEN PROCESSOR IP —



Eclipse Modeling Framework



Eclipse Platform



Eclipse CDT



Eclipse Embedded CDT
(GNU MCU/ARM
Eclipse Plug-ins)



CORE-V IDE: planned items

- Integrate GCC-based toolchain from Embecosm
- Integrate LLVM-based toolchain from Thales
- Add “Hello World” sample project
- Provide project templates
- Publish binaries to be a foundation for downstream solutions

<https://github.com/openhwgroup/core-v-ide-cdt>

Eclipse CDT and Platform IO



Embedded development

PlatformIO is a professional collaborative platform for embedded development that support multiple IDE including Eclipse

- **800+ target boards** (development kits)
- **20+ software frameworks**
(Arduino, ARM mbed, CMSIS, ESP-IDF, FreeRTOS, STM32Cube, Zephyr RTOS, and others)
- **30+ semiconductor architectures and development platforms**
(ARM, AVR, Espressif 8266/32, MCS-51, MSP430, PIC32, STM8, RISC-V, and others)
- **Over 10,000 libraries**
- **All famous operating systems**
(Windows, macOS, Linux, FreeBSD, Linux ARMv6+, card-sized PCs)

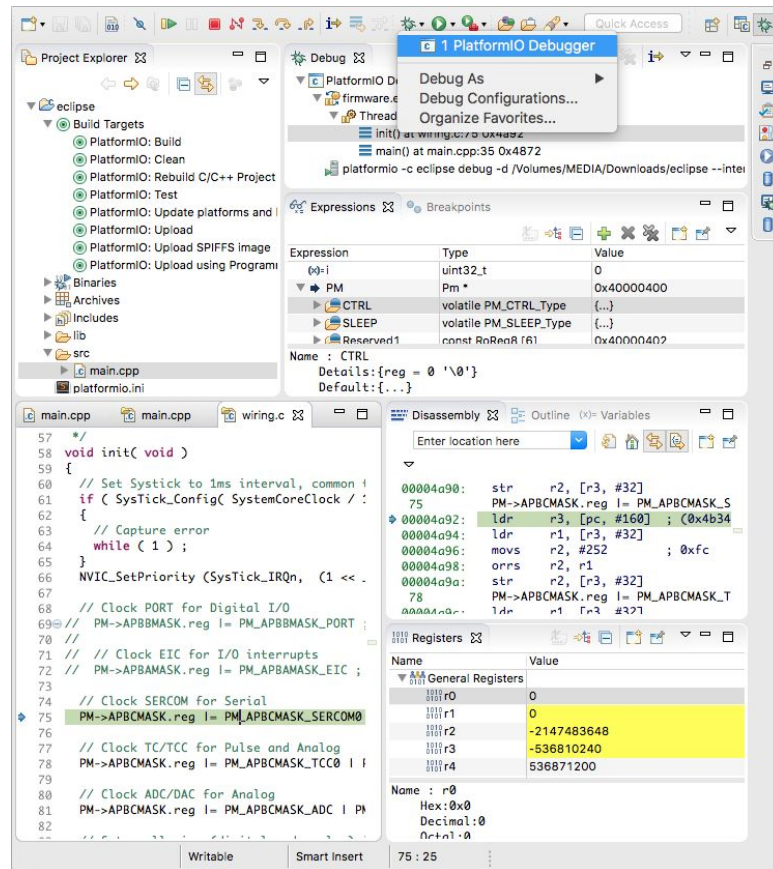


PlatformIO



- Multi-board and Multi-architecture programming experience
- Debugging, Unit Testing, Static Analysis, Firmware Inspection, and Remote Development out-of-the-box
- Developers can work simultaneously on the same embedded project using different development environments and the favourite operating system
- Code for any supported framework can be compiled and uploaded to a target platform in minutes
- Developers no longer have to manually find and assemble an environment of toolchains

Eclipse integration



The screenshot displays the Eclipse IDE interface with PlatformIO integration. The Project Explorer on the left shows a project structure with 'main.cpp' selected. The Debug console at the top right shows the PlatformIO Debugger interface with a menu for 'Debug As' and 'Debug Configurations...'. The Expressions panel shows variables like 'PM' and 'CTRL'. The Disassembly panel shows assembly code for the selected line. The Registers panel shows the current state of registers, with r0 highlighted.

Expression	Type	Value
PM	uint32_t	0
CTRL	volatile PM_CTRL_Type	{...}
SLEEP	volatile PM_SLEEP_Type	{...}
Reserved1	const RnRnR16	0x40000402

Name	Value
r0	0
r1	0
r2	-2147483648
r3	-536810240
r4	536871200

A dark, moody photograph of a person sitting at a desk in an office. The person is wearing glasses and has their hand near their chin, looking at a computer monitor. The desk is cluttered with papers, a keyboard, and a coffee cup. The background shows a brick wall and another monitor.

Creating Safe Products with Eclipse CDT

Compliant Safety Plan & Safety Case

ISO 26262-2: 6.4.5-13

Safety Plan (Process)

- System Safety Concept
- Hardware Safety Concept
- Software Safety Concept
- Tools: ISO 26262-2: 6.4.6.5.i

**Tool
Classification
Report**

**Tool
Safety
Manual**

**Tool
Qualification
Plan**

**Tool
Qualification
Report**



Create
Safe
Product



Safe Product
ISO 26262-2: 6.4.8

Safety Case

- System SC
- Hardware SC
- Software SC
- Tools:

**Safe Tool
Usage Report**

TCR

TSM

TQP

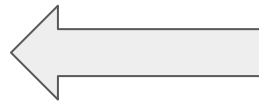
TQR



Qualification Kit



IDE Integration



Project Metadata



The screenshot shows the Eclipse IDE interface. On the left is the Project Explorer with a project named 'safe'. The main area displays the Problems view, which lists 2 errors and 2 infos. The 'safe' project is highlighted in blue. A 'Safe Tool Usage Report' dialog is open, showing the report for the 'safe' project.

Description	Resource
✖ Errors (2 items)	
✖ The used option '-O3' of tool 'gcc' was not qualified, please follow you Safety Manual	not_yet_safe
✖ The used version '4.9.4' of tool 'gcc' was not qualified, please follow you Safety Manual	not_yet_safe
ℹ Infos (2 items)	
ℹ The used option '-O2' of tool 'gcc' was qualified successfully	safe
ℹ The used version '4.9.3' of tool 'gcc' was qualified successfully	safe

The used version '4.9.3' of tool 'gcc' was qualified successfully

Safe Tool Usage Report
Go to Resource



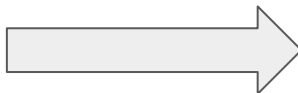
Safe Tool Usage Report

A woman with dark hair and glasses is sitting at a desk, looking at a laptop screen. Her hands are on the keyboard. The desk has some papers and a glass of water. The background is blurred, showing shelves with books or files. The overall lighting is dim, with a warm orange glow from the laptop screen.

Eclipse CDT needs your contribution

Step 1: Eclipse Account

1. Create account at <https://www.eclipse.org/>
2. Sign Eclipse Contributor Agreement (electronically)
3. Specify your GitHub id in your Eclipse profile



Email address 🌐 *

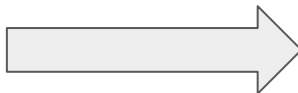
alexander.fedorov@arsysop.ru

The Eclipse community operates in an open and transparent manner. All community activities should be visible to others who use our services. Some users p

Eclipse Foundation username *

afedorov3un

This is your committer ID, which is also your username



STATUS

Eclipse Contributor Agreement 



Social Media Links

GitHub Username 🌐

ruspl-afed

Your GitHub Username makes it easier projects hosted on GitHub.

Step 2: Bugzilla

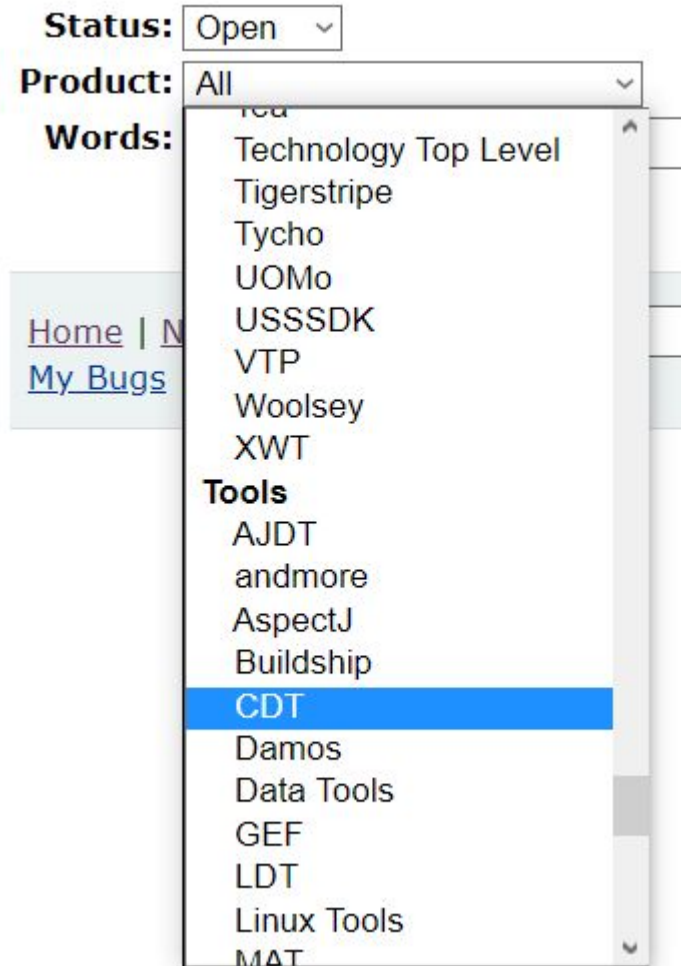
1. Login to Bugzilla

<https://bugs.eclipse.org/bugs>

2. Configure notifications

<https://bugs.eclipse.org/bugs/userprefs.cgi?tab=email>

3. Actualize existing and create new records



Step 3: Gerrit

1. Login to Gerrit <https://git.eclipse.org/r/>
2. Upload SSH keys
<https://git.eclipse.org/r/settings/#SSHKeys>
3. Configure notifications
<https://git.eclipse.org/r/settings/#Notifications>



Sign In to Gerrit

Username

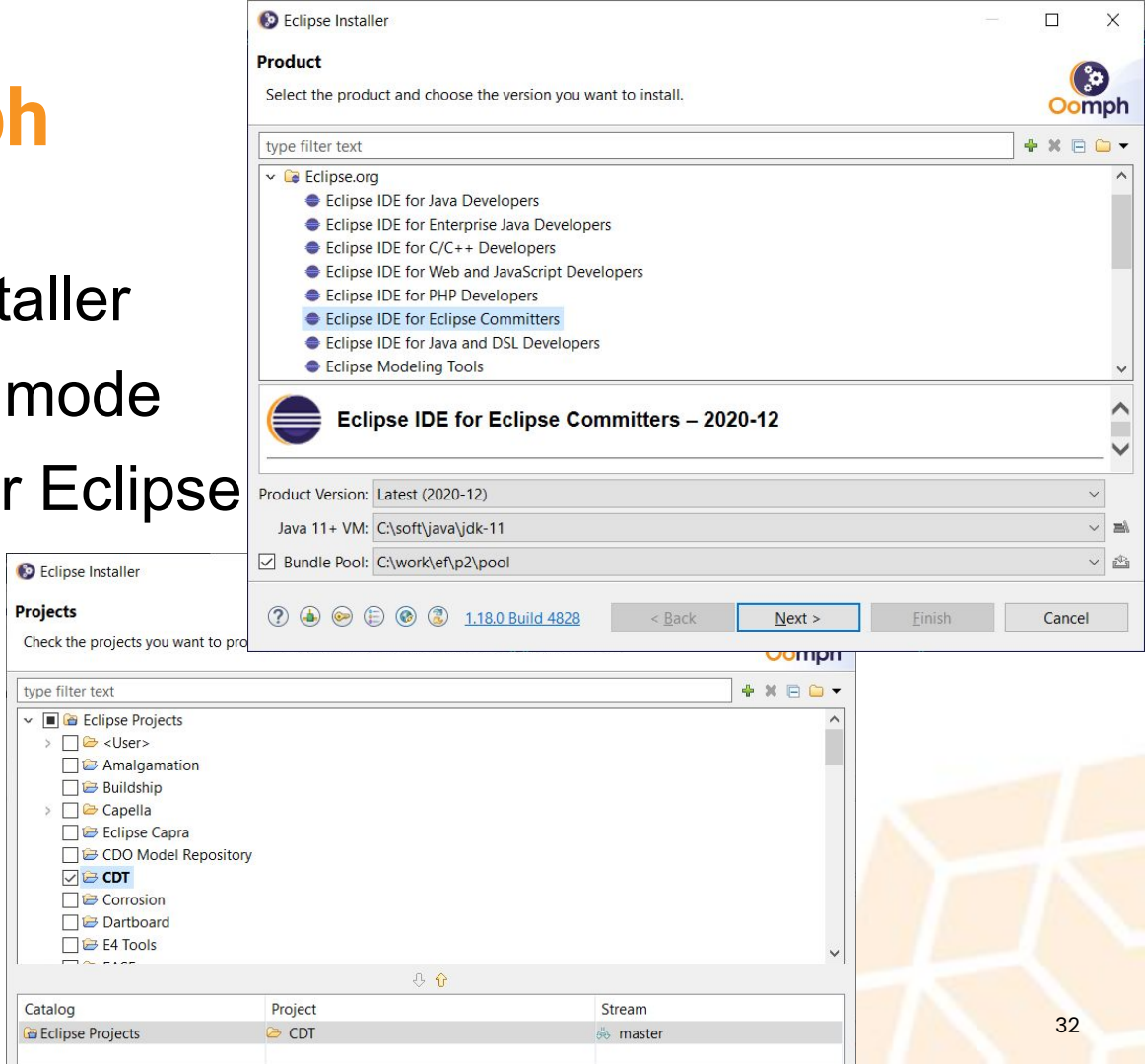
Password

☐ Remember me

[Cancel](#)

Step 4: Oomph

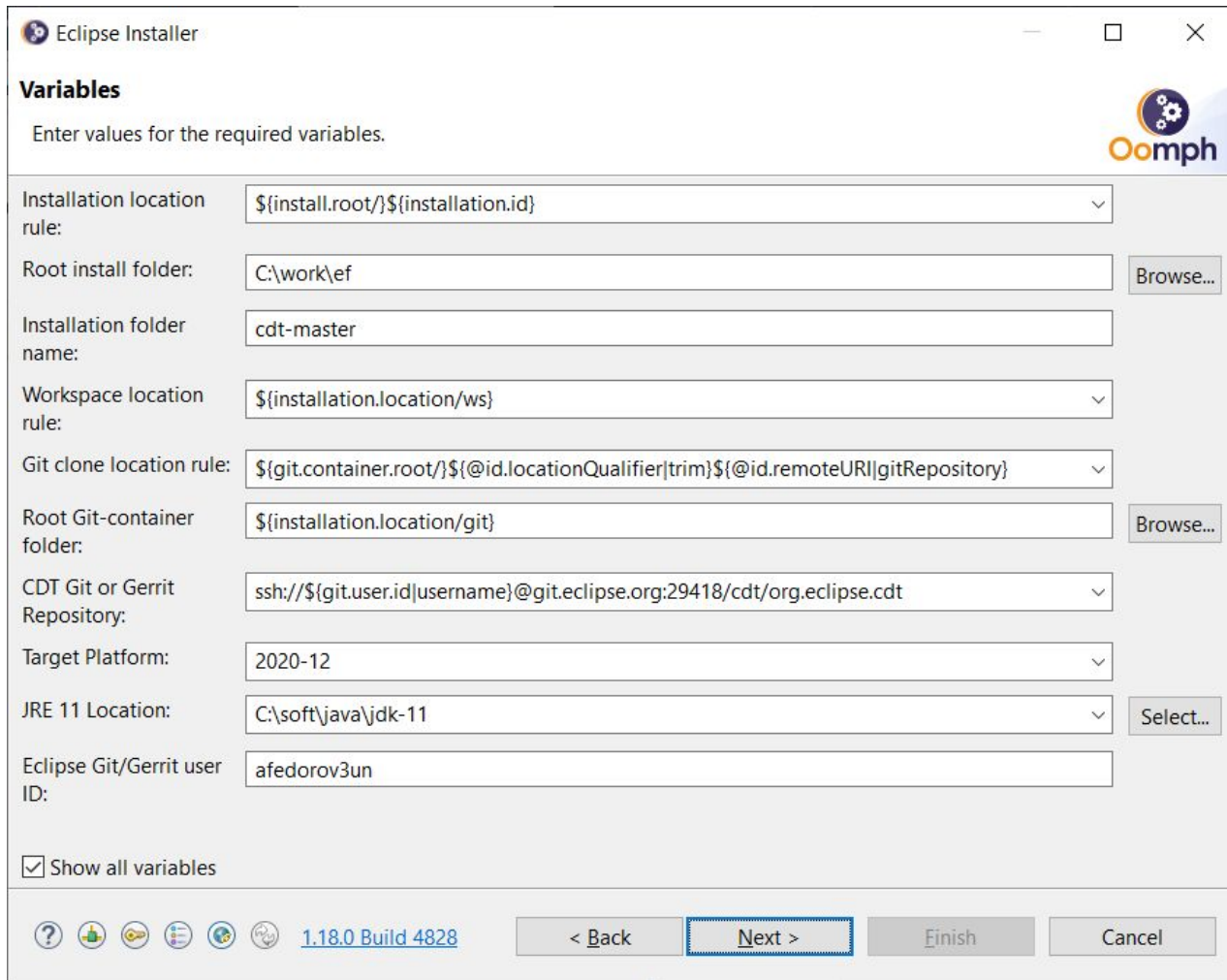
1. Download Eclipse Installer
2. Switch to “Advanced” mode
3. Select “Eclipse IDE for Eclipse Committers” package
4. Select “CDT” project



Step 4: Oomph (2)

Gerrit →

User ID →



The screenshot shows the 'Eclipse Installer' window with the 'Variables' tab selected. The window title is 'Eclipse Installer'. In the top right corner, there is an 'Oomph' logo. The main area is titled 'Variables' with the instruction 'Enter values for the required variables.' Below this, there are several configuration fields:

- Installation location rule:** A dropdown menu showing the value `${install.root}/${installation.id}`.
- Root install folder:** A text field containing `C:\work\ef` and a 'Browse...' button to its right.
- Installation folder name:** A text field containing `cdt-master`.
- Workspace location rule:** A dropdown menu showing the value `${installation.location/ws}`.
- Git clone location rule:** A dropdown menu showing the value `${git.container.root}/${@id.locationQualifier|trim}/${@id.remoteURI|gitRepository}`.
- Root Git-container folder:** A text field containing `${installation.location/git}` and a 'Browse...' button to its right.
- CDT Git or Gerrit Repository:** A dropdown menu showing the value `ssh://${git.user.id|username}@git.eclipse.org:29418/cdt/org.eclipse.cdt`.
- Target Platform:** A dropdown menu showing the value `2020-12`.
- JRE 11 Location:** A dropdown menu showing the value `C:\soft\java\jdk-11` and a 'Select...' button to its right.
- Eclipse Git/Gerrit user ID:** A text field containing `afedorov3un`.

At the bottom left, there is a checkbox labeled 'Show all variables' which is checked. At the bottom of the window, there is a status bar with several icons (help, download, keys, settings, network, refresh) and the text '1.18.0 Build 4828'. To the right of the status bar are four buttons: '< Back', 'Next >' (which is highlighted with a blue border), 'Finish', and 'Cancel'.

Step 5: Patch

Important!



1. Select a bug
2. Prepare a patch
3. Compose commit message.
4. Commit and Push to Gerrit
5. Follow the review comments

[illegible]

Contact us: cdt-dev@eclipse.org

Thank you!

Join the conversation:

 [@EclipseCon](https://twitter.com/EclipseCon) | [#EclipseCon](https://twitter.com/EclipseCon)



ECLIPSE
2020 CON

Evaluate the Sessions

Sign in and vote at Eclipsecon.org:

-1 0 +1



ECLIPSE
2020 CON