Pico Work Setup

1 History

1.1 Start

Start new rpi pico development based on the official plugin in vscode

Takeover dev1 V 0.1.2 24.aug.2025

2 Intro

2.1 Goal

How to setup the work environment for raspberry pi pico c/c++ development

3 Basic first client setup

3.1 Setup Windows

Create main subdir

\pico

Download from

https://code.visualstudio.com/

Use non installer packet

https://code.visualstudio.com/Download

Create install dirs

bin\vscode

Create work dirs

projects\

extract vscode to

bin\vscode

Install VS Extension

Raspberry Pi Pico

von

raspberrypi.com

Git is not auto installed

Install GIT from

https://git-scm.com/install/windows

Use Portable ("thumbdrive edition") and install in

bin\git

On same computer extensions and last project is still on

fwehfhw

3.1.1 Directory structure

Update pico probe binary

Connect *.sh with git-bash.cmd

3.2 Setup ChromeOS Flex

4 Preparation of a new project

4.1 New project at windows

Click on pico symbol and use extension created

New C/C++ Project

As location

<drive>:\pico\projects

Project subdir will be created

At Problems with cmake; delete build dir; open a terminal

```
cmake -S . -B build -G "Ninja"
ninja -C build
```

4.2 New project at RaspiOS

4.3 New project at ChromeOS Flex

5 Reactivate project from git

5.1 Reactivate at windows

5.2 Reactivate at RaspiOS

```
"gdbPath": "gdb-multiarch",
```

5.3 Reactivate at ChromeOS Flex

5.4 All

5.4.1 Execute programm generator

Edit launch configuration

Move source files to src

5.4.2 Configuration for external OpenOCD debugger

5.5

6 Dirty Work

6.1 got native compiler

7 Questions

7.1 Which build type

Execute full build with

```
cmake -S src -B . -G "Unix Makefiles" -D CMAKE_BUILD_TYPE=Debug
make
```

```
cmake -S src -B build_pico -G "Ninja" -D CMAKE_BUILD_TYPE=Debug
ninja
```

ninja geht nur mit -B .

```
cmake -S src -B . -G "Eclipse CDT4 - Ninja" -D CMAKE_BUILD_TYPE=Debug -D
PICO_BOARD=pico_w
make
```

8 Problems

9 Links

9.1

9.2 Profiles

9.3 Other