Advanced Embedded Systems

Zusammenfassung

Joel von Rotz & Andreas Ming 23.05.2023

Quelldateien

Inhaltsverzeichnis

- ☐ FreeRTOS
  - ☐ Task
  - ☐ Scheduler
  - ☐ Priority
  - ☐ Interrupts
  - ☐ Timer
  - ☐ Queue
  - – [ ]

1. Entwicklung

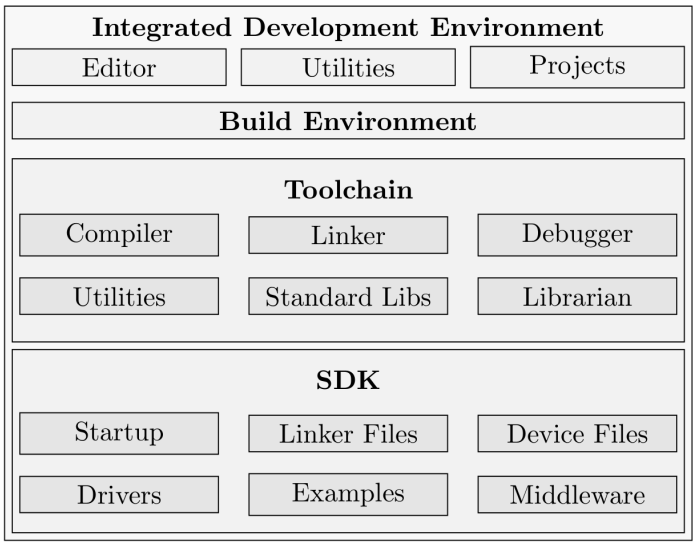
1.1 Entwicklungsumgebung

1.2 Cross-Development

Cross-Development bedeutet die Entwicklung einer Firmware auf einem **Host** für einen **Target**. Grund dafür ist, dass das *embedded* Target nicht genügend Ressourcen (CPU Leistung, Speicher) für die direkte Entwicklung hat.

> **i** Target & Host
>
> **Target** (wofür): Zielsystem, für das man entwickelt.
> **Host** (womit): bezeichnet die Umgebung , auf der man die Entwicklung vornimmt.

1.2.1 Integrated Development Environment



Toolchain: Kollektion von Tools wie Compiler, Linker, Debugger, etc. → einzelne Werkzeuge zum Zusammensetzen der Firmware

Build Environment: Steuert die Toolchain und den Übersetzungsvorgang → *make*, *Makefiles*

IDE: "Fancy Editor", beinhaltet Tools für bessere Produktivität → Intellisense, Workspace, Projekt

1.3 Eclipse (Open Source IDE)

2. Systeme

> **!** Was ist ein *embedded* System?
>
> Ein Rechner (CPU, MCU, etc.) integriert in ein System. Für eine Aufgabe/Zweck optimiert und meistens **kein** normaler Computer! Meistens von aussen nicht direkt zugreifbar, anders als beim Computer.
> Anwendung: Echtzeitsystem, Wetterstation, Steuerung für Roboterarm, etc.

3. Mikrocontroller

3.1 ARM Cortex Familie

4. FreeRTOS

4.1 Echtzeit

> **!** Was ist Echtzeit?

4.2 FreeRTOS & Interrupts

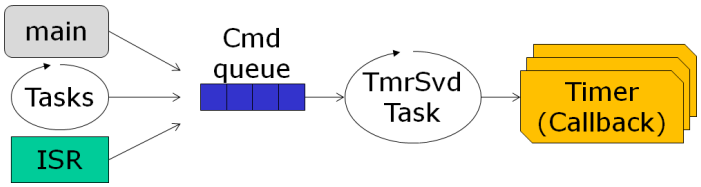
4.3 Kernel

4.4 Task

4.4.1 IDLE-Task

4.5 Timer

4.5.1 afd



Timer Service Daemon

Mit `configUSE_TIMERS` wird die Timer-Funktionen aktiviert und aktiviert automatisch die *Timer Service Daemon*

4.6 Queue

4.7 Semaphore & Mutex

4.8 Prioritäten

5. C

5.1 Konzepte

5.1.1 Synchronisation - Realtime

5.1.2 Synchronisation - Gadfly

5.1.3 Synchronisation - Interrupt

5.2 Bibliotheken