



Julian Bärtschi

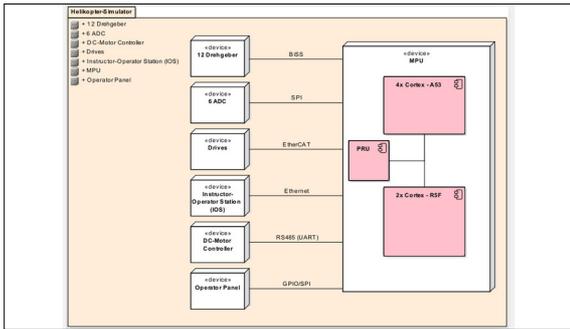


Dominic Hüppi

Diplomanden	Julian Bärtschi, Dominic Hüppi
Examinator	Prof. Reto Boderer
Experte	Urs Reidt, Hamilton Medical AG, Bonaduz, GR
Themengebiet	Embedded Software Engineering
Projektpartner	VRMotion AG, Dübendorf, ZH

Enhanced Motion Controller Platform

Embedded Linux Platform mit Hard Real-time Komponenten

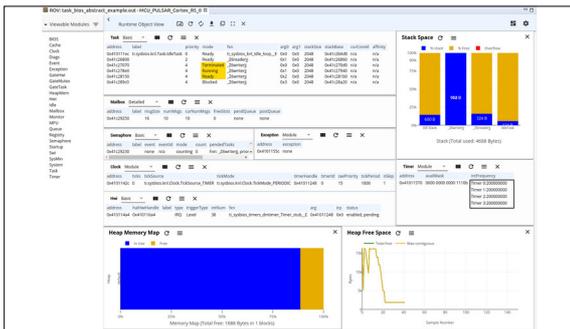


Ein Deployment-Diagramm der Sitara AM6548 MPU als Bestandteil eines Helikopter-Simulators
Eigene Darstellung

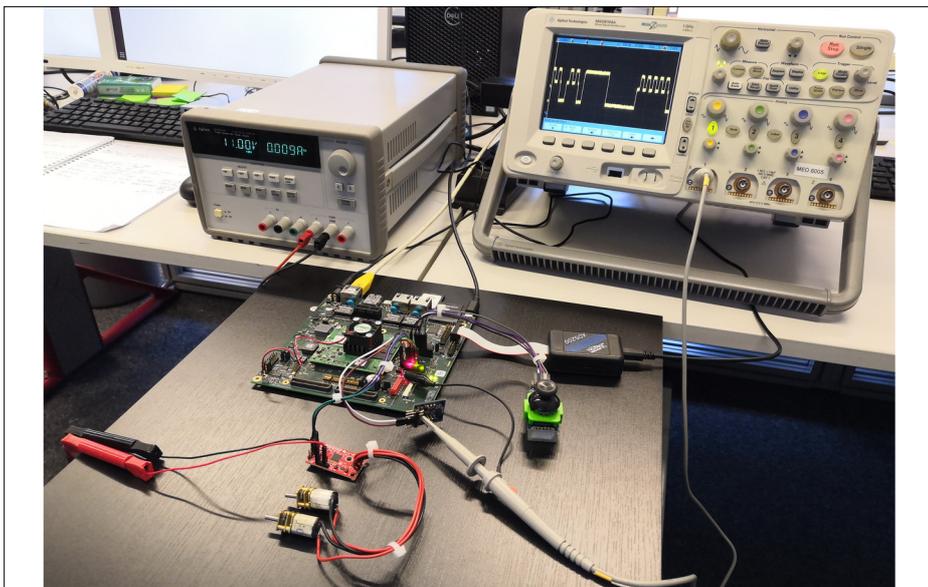
Ausgangslage: Die VRMotion AG entwickelt Helikopter-Simulatoren, um möglichst sichere, effiziente und umweltfreundliche Flugtrainings zu ermöglichen. Ein solcher Simulator besteht unter anderem aus einer Motion Platform und einem Control Loading System, welche heute mit einer Dual-Core Arm Cortex-R5f Mikrocontroller Unit realisiert werden. Um in Zukunft die Leistungsfähigkeit eines Simulators zu erhöhen, soll ein leistungsfähigerer Prozessor eingesetzt werden. Mit dieser Arbeit soll nun ein Sitara AM6548 Prozessor analysiert werden. Dieser Prozessor verfügt über 4 Arm Cortex A53 Kerne, einem Dual-Core Arm Cortex R5f Prozessor und einer Programmable Realtime Unit (PRU). Um diesen Prozessor zu testen, wurde das phyCORE-AM6548 Alpha Development Kit zur Verfügung gestellt.

Ziel der Arbeit: Um zu beurteilen, ob dieser Prozessor für die Anwendung im Helikopter-Simulator angemessen ist, sollen diverse Kommunikationsschnittstellen untersucht werden. Insbesondere soll die Interprozesskommunikation zwischen den Cortex A53- und den Cortex R5f Kernen sowie der PRU untersucht werden. Ausserdem soll die Kommunikation mit unterschiedlichen peripheren Schnittstellen untersucht werden. Dazu gehören GPIOs, UART, SPI und weitere. Schlussendlich sollen diese Resultate in Form eines Demonstrators aufgezeigt werden. Ein weiteres Ziel dieser Arbeit ist es auch, den Einstieg in eine Entwicklung auf dieser Plattform zu erleichtern.

Ergebnis: Die Interprozesskommunikation konnte erfolgreich umgesetzt und getestet werden. Die Kommunikation mit den peripheren Schnittstellen konnte wegen Problemen mit der Anpassung des Embedded Linux Kernels nur teilweise umgesetzt werden. Die resultierte Dokumentation ermöglicht es einem Entwickler, die Plattform aufzusetzen und in Betrieb zu nehmen. Zudem ist ersichtlich, wie die Interprozesskommunikation zu realisieren ist und wie neue Geräte in das Gesamtsystem einzubinden sind. Zudem wurde ein Demonstrator realisiert, welcher die Erkenntnisse der Arbeit umsetzt und veranschaulicht.



Die Runtime Object View bietet ein Instrument, um das Verhalten von Applikationen auf dem TI RTOS zu untersuchen.
Eigene Darstellung



An der phyCORE-AM6548 Plattform kann diverse Peripherie in Betrieb genommen werden. Die unterschiedlichen Kommunikationskanäle wurden in der Arbeit genutzt und analysiert.
Eigene Darstellung