

# Automatisches Testsystem für echtzeitfähige Linux-Images

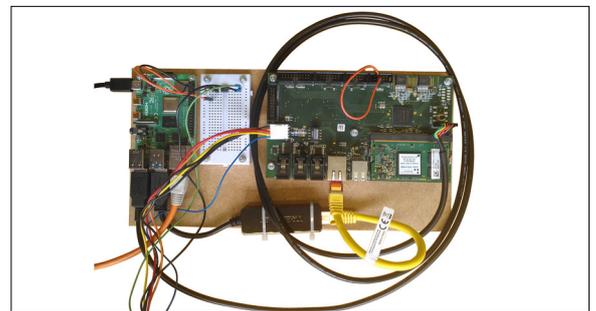
**Ausgangslage:** Die Fachhochschule Ostschweiz OST entwickelt eigene Linux-Images für verschiedene eingebettete Systeme. Die Images werden mit dem Yocto Buildsystem erstellt und dann an die Nutzer verteilt. Vor dieser Arbeit war das Testen der Images eine manuelle Aufgabe, die viel Zeit in Anspruch nahm. Im Rahmen dieser Bachelorarbeit wurde ein System zum automatischen Buiden und Testen von Linux-Echtzeitbetriebssystemen entwickelt. Eine Build- und Test-Pipeline erstellt automatisch alle erforderlichen Dateien. Die Pipeline kann mit verschiedenen Zielsystemen verwendet werden, zum Beispiel mit dem OST CB20 oder dem BeagleBone Blue.

**Vorgehen / Technologien:** Als Entwicklungsplattform wurde GitLab ausgewählt, weil es bereits für andere Projekte bei OST verwendet wird und viele Funktionen zur Verfügung stehen. Mit der Pipeline wird das Image zunächst auf einem Build-PC erstellt und dann in die GitLab-Instanz hochgeladen. Anschliessend wird das Image auf einer Testplattform bereitgestellt und gestartet. Gleichzeitig wird das SDK und die spezifischen Tests für das Zielsystem auf dem Build-PC erstellt. Um zu überprüfen, ob die Images korrekt funktionieren, werden die Tests auf dem Zielsystem ausgeführt und die Testergebnisse gespeichert. Aus diesen Ergebnissen wird nach jedem Testlauf ein Testbericht erstellt, um zu überprüfen, ob die Tests erfolgreich verlaufen sind. Im Falle eines gescheiterten Tests schlägt die Pipeline fehl und der Entwickler muss den Fehler beheben.

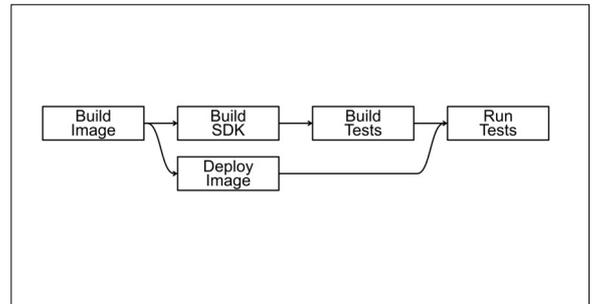
**Fazit:** Die Pipeline erleichtert die Entwicklung von Images deutlich. Für die CB20-Plattform konnte die komplette Pipeline umgesetzt und für die

Produktionsumgebung vorbereitet werden. Beim BeagleBone Blue konnten die einzelnen Schritte der Pipeline realisiert und getestet werden.

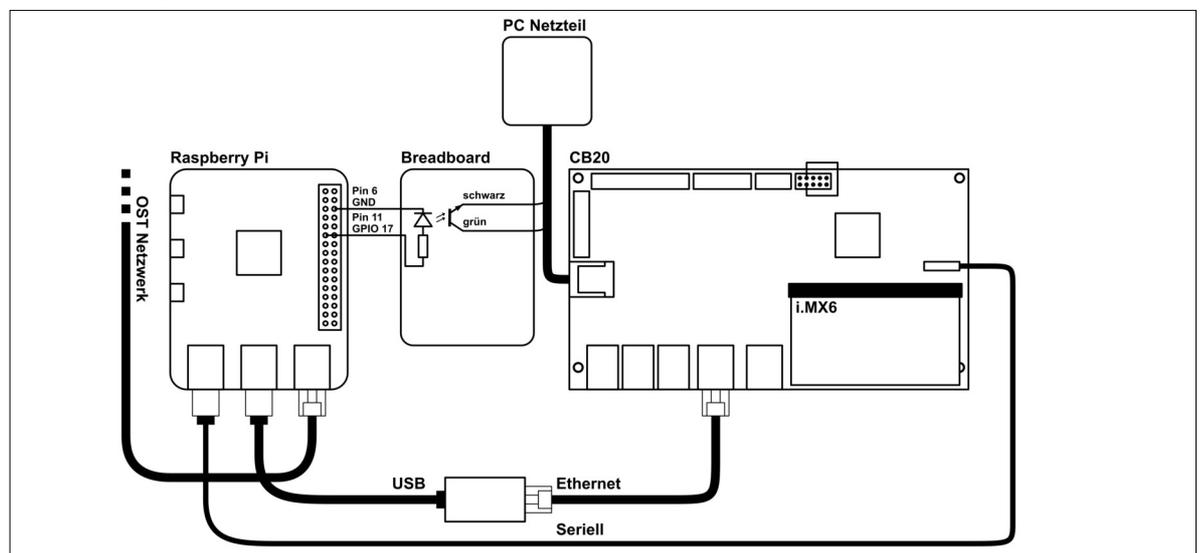
**Testaufbau CB20**  
Eigene Darstellung



**Pipeline zum Testen der Images**  
Eigene Darstellung



**Testaufbau CB20 Schema**  
Eigene Darstellung



Referent  
Prof. Dr. Urs Graf

Korreferent  
Prof. Rolf Grun

Themengebiet  
Ingenieurinformatik