

Qt for Microcontroller – A Competitive Comparison

Im Dezember 2019 veröffentlichte The-Qt-Company das erste Release von Qt-for-Microcontroller [\[1\]](#) [\[2\]](#). Diese Technologie basiert auf einer völlig neuen Qt-Architektur, welche es erlaubt, einfache Qt-Quick Anwendungen für eine Auswahl an Microcontroller zu kompilieren (Bare Metal ohne Betriebssystem). Diese neue Möglichkeit, Qt-ähnlichen Code in Zukunft auch auf Microcontroller laufen zu lassen beflügelt aktuell die Phantasie der Community. Doch wo liegen die Grenzen dieser Technologie? Wie gut funktioniert Qt-for-Microcontroller im Vergleich zu etablierten Herstellern auf diesem Gebiet? Welche Vor- und Nachteile ergeben sich bei der Verwendung von Qt-for-Microcontroller für ein Unternehmen, wenn es bereits Produkte gibt, welche mit Open-Source-Qt auf embedded Linux umgesetzt sind?



[1] <https://www.qt.io/blog/qt-for-mcus-1.0>
 [2] <https://www.qt.io/product/develop-software-microcontrollers-mcu>

Aufgabenstellung:

- Inbetriebnahme STM, NXP und/oder Renesas Hardware für die Qt-on-Microcontroller Evaluierung.
- Evaluierung aktueller alternativer Werkzeuge und Tools im Umfeld Bare-Metal-Grafik, welche einen vergleichbaren Funktionsumfang zu Qt-on-Microcontroller haben.
- Portierung der Sequality Bottling-Demo (<https://www.sequality.at/loesungen/touch-display-software-hmis/>) auf Qt-on-Microcontroller.
- Gegenüberstellung von Qt-on-Microcontroller vs. Qt-Open-Source und μ C-Software-Alternativen hinsichtlich
 - Ressourcenverbrauch
 - Features
 - Hardware-Stückkosten

Technologie-Umfeld:

- C, C++, Qt, QML
- STM und NXP Microcontroller
- Linux, embedded Linux

Kontakt:

Stefan Larndorfer, Stefan.Larndorfer@sequality.at
 sequality software engineering
 Softwarepark 26, A-4232 Hagenberg
www.sequality.at