

Portierbare Treiberbibliothek für IO-Link Master Shield/Hat

Studiengang: BSc in Elektrotechnik und Informationstechnologie | Vertiefung: Embedded Systems
Betreuer: Prof. Dr. Torsten Mähne
Experte: Daniel Kühni (Inetronic AG)
Industriepartner: Balluff AG, Biel/Bienne

Mittels IO-Link können Messdaten und zusätzliche Informationen von intelligenten Sensoren und Aktoren abgerufen sowie auch konfiguriert werden. Die Implementierung des IO-Link-Protokolls ist aufgrund seines grossen Funktionsumfangs zeitintensiv. Oft wird deshalb auf proprietäre Bibliotheken von Drittanbietern zurückgegriffen. Ziel des Projekts war eine Open-Source-Lösung, die die Realisierung eines IO-Link Masters durch eine frei verfügbare Bibliothek vereinfacht.

Ausgangslage

Der IO-Link-Standard definiert einheitliche Stecker/Kabel und ein flexibles Protokoll zur Anbindung von intelligenten Sensoren und Aktoren an Steuerungen. Er findet in der Automatisierungsindustrie immer grössere Verbreitung. Deshalb soll er auch in der Ingenieurs- und Techniker Ausbildung berücksichtigt werden. In einer vorangegangenen Arbeit wurde von der Berner Fachhochschule in Partnerschaft mit der Balluff AG eine Hardware und eine dazugehörige rudimentäre Software-Bibliothek entwickelt. Der sogenannte IO-Link Master Shield/Hat ist für den Einsatz mit einem Arduino Due oder einem Raspberry Pi vorgesehen und stiess bereits auf Interesse bei weiteren externen Partnern, die ihn für erste eigene Projekte einsetzen. Ein Portieren der vorhandenen Bibliothek auf eine andere Hardware war jedoch erschwert, da die Bibliothek sehr hardwareabhängig aufgebaut war. Ziel der Arbeit war es, die Bibliothek grundlegend zu überarbeiten, so dass die Implementierung möglichst unabhängig von der konkreten Hardware ist und somit künftig einfach an eine neue Hardware angepasst werden kann.

Konzept und Realisierung

In der Projektstudie wurde ein Konzept entworfen, welches die Bibliothek mehr von der Hardware abstrahiert. Mithilfe dieses Konzepts wurde die Bibliotheksarchitektur angepasst. Die Aufteilung in verschiedene Abstraktionsschichten ist in Abb. 1 ersichtlich.

Das CMake-basierte Build-System wurde ebenfalls überarbeitet. Es musste eine Toolchain für Raspberry Pi erstellt werden. Für Arduino wurde auf eine bereits vorhandene Lösung zurückgegriffen. Damit das Kompilieren und Debuggen einfach umzusetzen ist, wurde eine ausführliche Anleitung erstellt. Für Arduino und das Raspberry Pi kann die Bibliothek nun cross-kompiliert werden.

Um eine möglichst zuverlässige Funktion zu garantieren, wurde ein Unit-Testsystem auf Basis von Catch2 integriert. Mit dessen Hilfe können Tests erstellt werden, welche dann die Funktion der Bibliothek überprüfen.

Zielerreichung und Ausblick

Die Ordnerstruktur und Klassenhierarchie der Bibliothek wurden restrukturiert und reimplementiert, so dass eine bessere Trennung der Protokoll- und Hardwareebene erreicht wird. Mit dieser neuen Struktur können nun die Page-1-Daten von IO-Link-Geräten abgerufen werden und sie ist flexibler auf anderer Hardware einsetzbar. Durch die Arbeiten am Build-System kann die Bibliothek nun mit einer Entwicklungsplattform für Arduino und Raspberry Pi kompiliert werden, was bei der Weiterentwicklung einen erheblichen Vorteil bedeutet. Mit diesem Stand wurde die Bibliothek in der Organisation openiolink unter der Apache-2.0-Lizenz auf GitHub (<https://openiolink.github.io/>) veröffentlicht. Damit wird eine Weiterentwicklung durch die Community und im Rahmen weiterer studentischer Arbeiten angestrebt. In einem nächsten Schritt kann nun das Auslesen der Prozess- und ISDU-Daten sowie die Konfiguration



Janik Lehmann
janik.lehmann@swissonline.ch

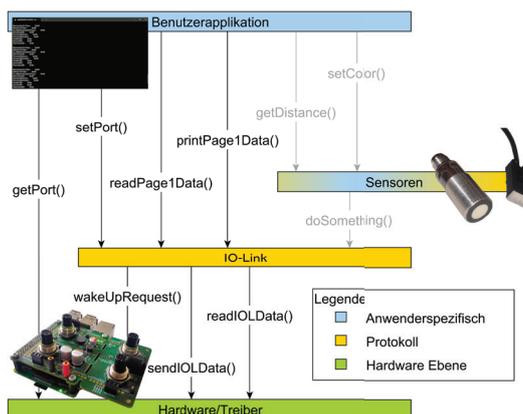


Abb. 1: Architekturkonzept für die quelloffene IO-Link-Master-Bibliothek