

Intelligenter Stromzähler

Heutige Smart Meter erfassen alle 15 Minuten einen Messpunkt. Andere, schnellere Geräte sind auf Systeme eines bestimmten Herstellers abgestimmt und nicht frei konfigurierbar. Zur Sicherstellung der Kontrolle über die gesamte Messkette kam die Idee auf, ein neues Messgerät zu konstruieren. Auf Basis einer vorangehenden Projektarbeit erfolgte die Weiterentwicklung.

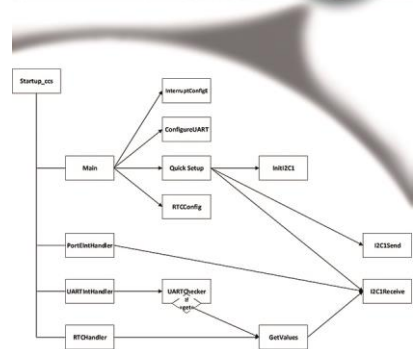
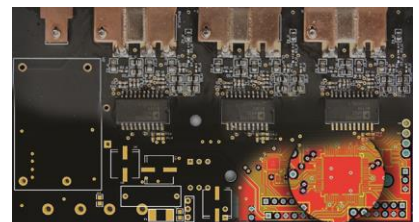
Die Arbeit beinhaltet die Inbetriebnahme der Messelektronik, die Entwicklung von Software für den enthaltenen Mikrocontroller sowie Messungen, Simulationen und Berechnungen zur Bestimmung der Latenz und der maximalen Abtastrate. Ausserdem sollten die digitalen und analogen Eingangsfiler charakterisiert werden. Die Erfassung der Latenz ergibt für eine Testmessung eine Verzögerung von 3.3 ms. Dieser Wert entspricht den Erwartungen durch die vorgängige Modellierung in Simulink, deren Ergebnis bei 3.2 ms lag. Für die Weiterleitung an die nächsthöhere Hardware-Ebene ist von einer Verdopplung dieser Zeit auszugehen. Unter Berücksichtigung eines Sicherheitsfaktors folgt daraus eine Abtastrate von 100 Abtastpunkten pro Sekunde. Bei einer Messkampagne über eine Woche ergibt sich dabei ein Datenvolumen von 7 GB. Ein Test erfolgte mit einer Abtastrate von zehn Messungen pro Sekunde. Jeder Messpunkt beinhaltet ein Paket mit den Parametern Strom, Spannung, Leistung und Energie. Die analogen und digitalen Filter am Eingang ergeben zusammen eine Grenzfrequenz von 3.3 kHz. Das analoge RC-Filter erster Ordnung am Eingang unterdrückt Aliasing-Effekte und hat eine Grenzfrequenz von 4.8 kHz. Um den Signal-Rauschabstand im betrachteten Bereich zu vergrössern, wird Oversampling und Noise Shaping eingesetzt.

In einem weiteren Schritt könnte noch die Benutzerebene der Hard- und Software entstehen. Ausserdem sollte das Layout der Leiterplatte einigen Anpassungen und Optimierungen unterzogen werden. Das zugrunde liegende Konzept hat sich jedoch bewährt und es empfiehlt sich eine weitere Verfolgung.



Diplomand
Simon Oehy

Dozent
Andreas Heinzlmann



Oben: Hardware Design.
Unten: Programmstruktur mit
Funktionsaufrufen.