

ARM-basierte, echtzeitfähige Hardwareplattform für Datenerfassung, Signalverarbeitung und eHealth

Lutz Krätzer¹, Hermann Lorenz¹, Christian Berthold² und Frank Sonntag¹

¹Fraunhofer-Institut für Werkstoff- und Strahltechnik IWS, Dresden, Deutschland

²SBU Schirmer + Dr. Berthold Umwelttechnik GmbH, Waldheim, Deutschland

Kontakt: frank.sonntag@iws.fraunhofer.de

Einleitung

Einplatinencomputer wie Raspberry Pi, CubieTruck, PandaBoard oder BeagleBone ermöglichen in Kombination mit echtzeitfähigen Betriebssystemen die Realisierung zahlreicher komplexer, medizintechnischer Applikationen und stellen dafür gleichzeitig die Funktionalitäten Echtzeit-Datenerfassung, Signalverarbeitung und nutzbare Remote-Dienste für eHealth bereit. Die eingesetzten Prozessoren, welche auf der ARM-Architektur basieren, zeichnen sich durch eine geringe Leistungsaufnahme aus, was sie für den Einsatz in Eingebetteten Systemen / *Embedded Systems* prädestiniert.

Grundvoraussetzung für eine sensitive Messung ist die äquidistante Abtastung, damit überlagerte, periodische Störsignale kompensiert werden können. Die damit verbundene harte Echtzeitfähigkeit ist eine wesentliche Anforderung an die eingesetzten Hardwareplattformen.

Optische Messmethoden verbinden den Vorteil der Nicht-Invasivität mit hoher Sensitivität und sind von besonderem Interesse [1-5].

Methoden und Materialien

Fluoreszenzmesssystem

Für die Fluoreszenzmessung wurde das von Fraunhofer-Institut für Angewandte Optik und Feinmechanik IOF und Fraunhofer-Institut für Werkstoff- und Strahltechnik IWS gemeinsam entwickelte Sensormodul verwendet, welches ohne mechanisch bewegte Teile die Fluoreszenzanalyse bei zwei Anregungswellenlängen ermöglicht [5]. Dabei wird das Licht zweier unterschiedlich farbiger Hochleistungsleuchtdioden überlagert und in einer mikroskop-ähnlichen Anordnung auf die Probe abgebildet. Die spektrale Charakteristik wird durch die Wahl von Lichtquellen und Filtersatz bestimmt. Mit diesem System ist es möglich, durch sequenzielle Beleuchtung, d. h. Schalten der beiden unterschiedlichen Leuchtdioden, die Fluoreszenz zweier verschiedener Farbstoffe zu bestimmen. Da die Fluoreszenz in erster Näherung isotrop in alle Raumrichtungen emittiert wird, muss ein empfindliches System einen großen Raumwinkelbereich erfassen. Diese Forderung entspricht einer großen numerischen Apertur (NA) des optischen Systems, weshalb als Frontlinse eine asphärische Linse (NA~0.5) eingesetzt wird. Die optische Abbildung ist so dimensioniert, dass Kanäle mit etwa 0,5 mm Breite ausgeleuchtet werden. Als Detektor wird ein

Photomultiplier (PM) H10723-20 (Hamamatsu) verwendet.

Ansteuerung Fluoreszenzmesssystem

Für die Anbindung des Fluoreszenzmesssystems wurde in enger Kooperation mit der SBU Schirmer + Dr. Berthold Umwelttechnik GmbH eine universelle Steuerelektronik entwickelt, welche einen Analog-Digital-Umsetzer (ADC), zwei Digital-Analog-Umsetzer (DAC) sowie einen 2-Kanal-LED-Treiber umfasst, siehe Abbildung 1. Die Kommunikation zwischen Einplatinencomputer und Steuerelektronik erfolgt über elektronische Standardschnittstellen: Serial Peripheral Interface (SPI), Inter-Integrated Circuit (I²C)-Bus sowie vier General Purpose Input/Output (GPIO)-Kanäle.

Die äquidistante Konvertierung des PM-Ausgangssignals erfolgt über einen ADC vom Typ ADS8328 (low power, 16-bit, 500-kSPS ADC, Texas Instruments). Dieser wird über zwei GPIO-Kanäle und SPI-Bus angesteuert.

Die analogen Spannungen zur Steuerung von LED-Intensität und PM-Sensitivität werden mit zwei DAC vom Typ LTC2606 (16-Bit Rail-to-Rail DAC, Linear Technology) generiert. Die Ansteuerung erfolgt über I²C-Bus.

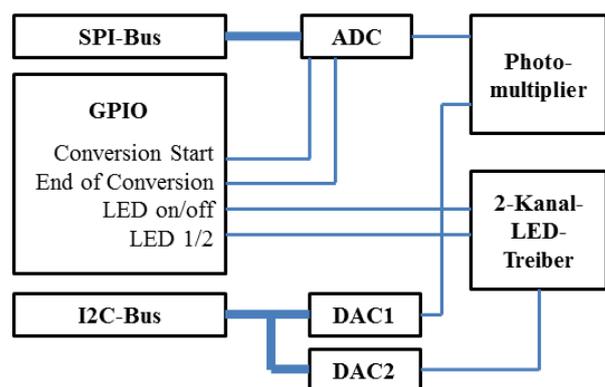


Abb. 1: Blockschaubild der Steuerelektronik

Echtzeiterweiterung Xenomai

Harte Echtzeitanforderungen unter Linux werden im vorliegenden Ansatz über das Framework Xenomai realisiert.

Xenomai basiert auf dem Dual-Kernel-Ansatz. Der Linux-Kernel wird nur so weit modifiziert, dass er mit dem Xenomai-Mikrokern zusammenarbeitet.

Damit wird eine klare Trennung der Verantwortungsgebiete für die Bereitstellung der unterschiedlichen Services des Systems erreicht. Diese Bereiche sind als Echtzeit-Domäne und Linux-Domäne organisiert. Für das Zusammenwirken beider Domänen ist eine Interrupt-Pipeline (I-Pipe) verantwortlich, siehe Abbildung 2. Eine Nutzung gemeinsamer Hardwareressourcen wird somit möglich. Ereignisse, welche üblicherweise von Systemkomponenten wie beispielsweise GPIO-Schnittstellen oder SPI-Kommunikationsbausteine ausgelöst werden, können von Handlern in beiden Domänen grundsätzlich verarbeitet werden. Entscheidend für das Erreichen harter Echtzeitanforderungen ist eine festgelegte Priorität zwischen den Domänen bei der Zustellung von Ereignissen. Echtzeittasks haben exklusiv und vorrangig die Kontrolle über Interrupts. Dies ist bei einer äquidistanten Messwertfassung (siehe Abbildung 3) eine notwendige Bedingung. Eingehende elektrische Signale können über einen Interrupt-Handler von der implementierten Logik erkannt und verarbeitet werden. Damit das System korrekt arbeitet, müssen die Verarbeitungsschritte innerhalb von bestimmten Zeitschranken abgearbeitet werden. Diese Zeitschranken sind von Parametern der systemischen Spezifikationen abhängig, beispielsweise von der Sampling-Rate. Des Weiteren sieht das Xenomai-Konzept einen Mechanismus vor, welcher verhindert, dass harte Echtzeit-Tasks durch einen Interrupt unterbrochen werden und somit ihre festgelegten Zeitschranken verletzen. Ein optionaler Interrupt-Shield setzt diese Logik um. Der Vorteil besteht darin, dass Echtzeit-Tasks auch in der Linux-Domäne ausgeführt werden dürfen. Zugriff besteht dann auch auf die bekannten Systemrufe von Linux und auf die API des Xenomai-Frameworks. Der Interrupt-Shield realisiert eine Verbindung zwischen der Echtzeit-Domäne und der Linux-Domäne ohne Verlust der geforderten harten Echtzeitfähigkeit.

Fernzugriff

Für Anwendungen im Bereich eHealth wird der vollständig implementierte und erprobte Netzwerk-Stack des Linux-Kernels verwendet. Darüber hinaus bieten die Einplatinenrechner Zugriff auf die physikalische Übertragungsschicht über etablierte Schnittstellen wie Ethernet, WiFi und Bluetooth. Realisierbar sind somit beispielsweise Telemonitoring-Applikationen auf Basis eines Web-servers über Hypertext Transfer Protocol (HTTP) oder Hypertext Transfer Protocol Secure (HTTPS), welche die Messwerte von außerklinisch durchgeführten, fluoreszenzbasierten Assays zur Verfügung stellen. Fernwartungszugriffe und Datenübertragungen werden über Nut-

zerprogramme, welche die Protokolle Secure Copy (SCP) oder Secure Shell (SSH) implementieren, sicher und ressourceneffizient realisiert.

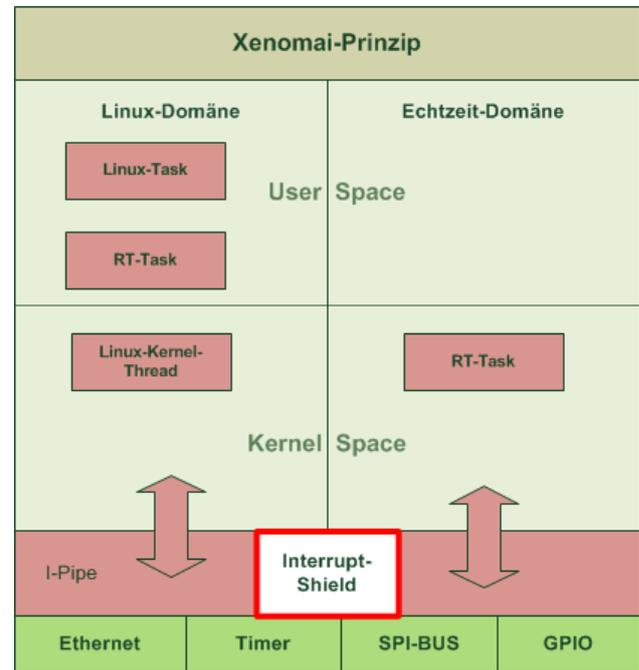


Abb. 2: Schematische Darstellung des Xenomai-Prinzips

Raspberry Pi

Der Raspberry Pi (Revision B) ist ein Einplatinencomputer, der auf System-on-a-Chip BCM2835 (Broadcom) basiert. Dieser gehört zur ARM11-Familie (ARM1176JZFS) und wurde mit einer Taktfrequenz von 700 MHz betrieben [6]. Das System verfügt über 512 MB Arbeitsspeicher. Die wichtigsten Schnittstellen sind ein SD-Card-Slot, zwei USB 2.0-Ports, ein 10/100 MBit-Ethernet-Port, ein HDMI-Ausgang sowie ein Expansionsport, welcher unter anderem sieben frei verwendbare GPIO-Ports sowie SPI-, I²C- und UART-Schnittstellen bereitstellt. Die Stromversorgung wird über einen Micro-USB-Anschluss realisiert (5 V, 1000 mA).

Ergebnisse

Konzeption Messwertfassung

Die vier Schritte zur Messung der Fluoreszenz sind die Initialisierung und Konfiguration der Hardware, die Aktivierung der Lichtquelle über eine vorgegebene Zeitspanne, die zyklische Messwertfassung sowie das Abschalten der Hardware. Systemisch bedingt sind die Initialisierung, das Abschalten der Hardware und die Lichtquellenaktivierung als zeitunkritisch zu beurteilen. Als zeitkritischer Pfad ist die zyklische Messwertfassung zu betrachten (siehe Abbildung 3).

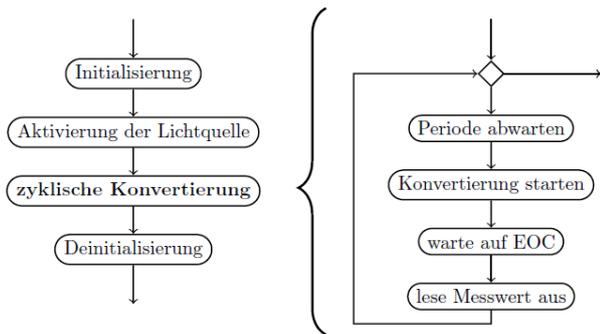


Abb. 3: Algorithmus der Messwernerfassung

Zur Erfassung der Messwerte ist eine zyklische Abtastung der Spannungswerte des PM durch den ADC erforderlich. Diese Spannungswerte repräsentieren die Fluoreszenz als Intensität über die Zeit. Die Messung wird durch ein 40 ns-Signal am Kanal *Conversion Start* (CONVST) gestartet. Der ADC signalisiert das Ende der Messwertkonvertierung am Kanal *End of Conversion* (EOC) mit einer steigenden Flanke. Anschließend kann der konvertierte Messwert über den SPI-Bus ausgelesen werden. Dieser Vorgang muss bis zum Beginn des nächsten Zyklus, welcher durch den Start der Konvertierung eingeleitet wird, beendet sein. Ansonsten kann ein fehlerbehafteter Wert im Ergebnisregister verbleiben, welcher die Korrektheit und Validität der Messreihe herabsetzt.

Die garantierbare, maximale Abtastrate f ergibt sich aus der Dauer für die Generierung des Auslösesignals zum Einleiten der Konvertierung auf dem Kanal CONVST (t_{start}), der Dauer der Konvertierung im ADC (t_{conv}), der Dauer, die der Raspberry Pi benötigt um eine steigende Flanke zu erkennen (t_{int}) und dem Auslesen des Messwertes über SPI (t_{spi}). Daraus folgt der Zusammenhang zwischen den einzelnen Zeiten und der garantierbaren, maximalen Abtastrate, siehe Gleichung 1.

$$\frac{1}{f} \geq t_{start} + t_{conv} + t_{int} + t_{spi} \quad (1)$$

Echtzeiterweiterung Xenomai

Die Inbetriebnahme des Xenomai-Kernels wurde analog zum Projekt LinuxCNC [7] durchgeführt. Dabei wurde der Linux-Kernel 3.2.27 der Raspbian-Distribution mit dem für einen Linux-Kernel der Version 3.2.21 vorgesehenen Xenomai-Patch 2.6.1 modifiziert.

Softwarearchitektur

In Voruntersuchungen wurde ermittelt, dass echtzeitkritische Prozesse im Xenomai-Kernelspace die geringste Latenz aufweisen [8].

Darauf aufbauend wurde die in Abbildung 4 dargestellte Softwarearchitektur entwickelt. Die Applikation realisiert die grafische Benutzerschnittstelle (GUI) und wird im Linux-Userspace ausgeführt. Der echtzeitkritische Teil ist als Kernelmodul implementiert. Der Zugriff der Applika-

tion auf das Kernelmodul erfolgt über das Öffnen einer Echtzeit-Gerätefile, welche die darunterliegende Funktionalität kapselt und über den üblichen Dateifunktionen analoge Echtzeitfunktionen der Xenomai-Umgebung zur Verfügung stellt (üblich ist `ioctl()`, verwendet werden muss `rt_dev_ioctl()`).

Für die Datenerfassung startet das Kernelmodul einen Echtzeittask im Xenomai-Kernelspace, welcher für die Anzahl der angeforderten Messwerte, die unter Konzeption der Messwernerfassung beschriebene, zyklische Konvertierung durchführt. Dabei wird ein Echtzeit-Timer gestartet, der die Flanke am Ausgang CONVST generiert. Sobald der ADC die Messwertkonvertierung abgeschlossen hat, wird über die steigenden Flanke am Eingang EOC ein Interrupt ausgelöst und der Messwert über den SPI-Bus ausgelesen. Nach abgeschlossener Datenerfassung wird der Echtzeit-Task beendet.

Eine eigene Implementierung eines SPI-Treibers realisiert die Kommunikation mit dem ADC. Die API von Xenomai beinhaltet nativ keine Unterstützung für das SPI.

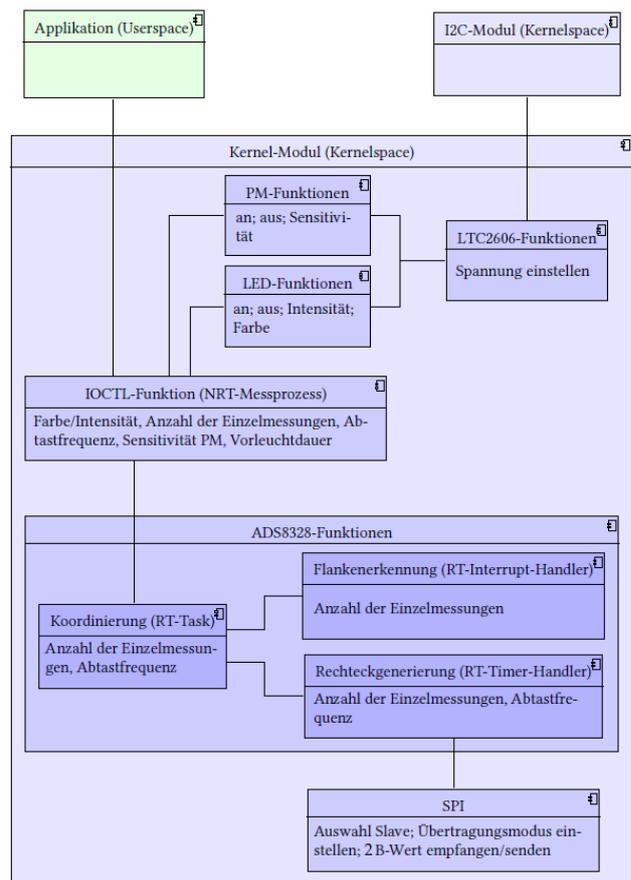


Abb. 4: Komponentendiagramm der Softwarearchitektur

Demonstrationssystem

Als Demonstrationssystem wurde eine Raspberry Pi verwendet und für die Interaktion mit den Benutzern über HDMI und USB mit einem Touchscreen (LS-7T, EETI eGalax_eMPIA Technology Inc.) gekoppelt. Abbildung 5 zeigt den Raspberry Pi mit aufgesteckter Steuerelektronik.

Das Anwendungsprogramm mit der GUI basiert auf der Qt-Klassenbibliothek zur plattformübergreifenden Entwicklung grafischer Benutzeroberflächen.

Für den Remotezugriff wurde ein SSH-Server mit X-Server-Unterstützung installiert und der Zugriff von Windows- und Linux-Systemen auf die Applikation erfolgreich demonstriert.

Für den Remotezugriff durch Windows-Systeme wurde XRDP (ein Server für das Remote Desktop Protocol) installiert und der Zugriff über Windows-Remote-Desktop erfolgreich nachgewiesen.

Weiterhin wurden ein SCP- sowie ein Webserver (Apache) installiert und der Datenaustausch mit Windows- und Linux-Systemen erfolgreich demonstriert.

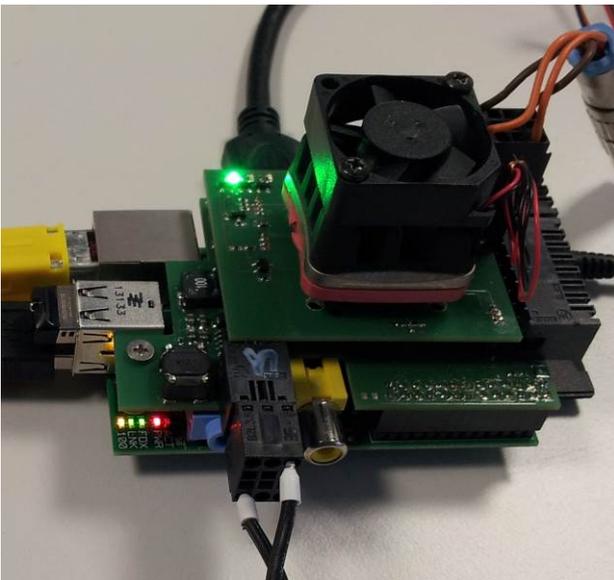


Abb. 5: Raspberry Pi mit aufgesteckter Steuerelektronik

Schlussfolgerungen

ARM-basierte Einplatinencomputer ermöglichen in Kombination mit echtzeitfähigen Betriebssystemen die Realisierung zahlreicher komplexer, medizintechnischer Applikationen und stellen dafür gleichzeitig die Funktionalitäten Echtzeit-Datenerfassung, Signalverarbeitung und nutzbare Remote-Dienste für eHealth bereit.

Basierend auf dem Raspberry Pi konnte erfolgreich ein System aufgebaut werden, welches Echtzeit-Fluoreszenzmessungen mit Abtastraten bis 20 kHz, Signalverarbeitung und Fernnutzung sicher realisieren kann.

Literatur

- [1] Schimek, K.; Busek, M.; Brincker, S.; Groth, B.; Hoffmann, S.; Lauster, R.; Lindner, G.; Lorenz, A.; Menzel, U.; Sonntag, F.; Walles, H.; Marx, U.; Horland, R.: Integrating biological vasculature into a multi-organ-chip microsystem. In: Lab on Chip, 2013, DOI: 10.1039/C3LC50217A
- [2] Wagner, I.; Materne, E.-M.; Brincker, S.; Süßbier, U.; Frädlich, C.; Busek, M.; Sonntag, F.; Sakharov, D. A.; Trushkin, E. V.; Tonevitsky, A. G.; Lauster, R.; Marx, U.: A dynamic multi-organ-chip for long-term cultivation and substance testing proven by 3D human liver and skin tissue co-culture. In: Lab on Chip, 2013, DOI: 10.1039/C3LC50234A.
- [3] Nötzel, M.; Busek, M.; Sonntag, F.: Nicht-invasive optische Druckmessung in Lab-on-a-Chip Systemen. In: 11. Dresdner Sensor Symposium, 2013, ISBN: 978-3-9813484-5-3, DOI: 10.5162/11dss2013/H2.
- [4] Ruhnow, M.; Schmieder, F.; Schmieder, S.; Danz, N.; Eger, R.; Sonntag, F.: Universelle Plattform für die automatisierte Vitalfärbung und fluoreszenzbasierte Charakterisierung eukariotischer Zellen. In: Dresdner Beiträge Medizintechnik (2012), S. 193-196, ISBN 978-3-942710-96-1.
- [5] Schmieder, F.; Schmieder, S.; Eger, R.; Friedrich, S.; Werner, A.; Danz, N.; Marx, U.; Sonntag, F.: Automated universal chip platform for fluorescence based cellular assays. In: Biomed Tech 2012; 57 (Suppl. 1), DOI: 10.1515/bmt-2012-4319.
- [6] Raspberry Pi Foundation, Hrsg. FAQs Raspberry Pi. url: <http://www.raspberrypi.org/faqs> (besucht am 12. 09. 2014).
- [7] Creating a minimal Raspbian based LinuxCNC image for Raspberry Pi; <http://wiki.linuxcnc.org/cgi-bin/wiki.pl?RaspbianXenomaiBuild> (abgerufen am 12.09.2014).
- [8] Lorenz, H.; Sonntag, F.; Krätzer, L.; Berthold, Ch.; Baumgartl, R.: Der Raspberry Pi als Plattform für Fluoreszenzmessungen unter Echtzeitbedingungen. In: Informatik aktuell, 2014, im Druck

Danksagung

Die Autoren danken dem Freistaat Sachsen und der Europäischen Union (SAB Projekt UNILOC) für die finanzielle Unterstützung.