

Miniprojekt Echtzeit-Betriebssystem

1 Allgemeines

Ziel des Miniprojektes

Während diesem Miniprojekt wird der Stoff des Vertiefungsmoduls "Echtzeit-Betriebssysteme" durch praktisches Arbeiten mit einem RTOS vertieft.

Das Miniprojekt soll zudem die Kompetenz der Studierenden für die Arbeit in Kleingruppen fördern. Das Projekt wird deshalb in Teams von zwei bis maximal vier Personen durchgeführt.

Bewertung

Das Miniprojekt wird bewertet. Bewertungskriterien sind:

- Design (30%)
- Code-Qualität (30%)
- Funktionalität / Zielerreichung (40%)

Themen

Für die Projektarbeit stehen drei mögliche Modelle zur Verfügung:

- **Stellaris Robotermodell:** Ansteuerung der Greifer und Förderbänder über einen CAN-Bus zum Transport von ECTS (Extreme Complex Transport Stone; farbiges, zylinderförmiges Element).
- **Getränkroboter:** Ansteuerung des Greifers, Waage, Drehtisch und Förderband über einen CAN-Bus zum Einfüllen und ausgeben eines Getränkes.
- **Echtzeitmodell Carme-M4 Kit:** Einlesen von Strings auf der seriellen Schnittstelle und darstellen des Strings mit Hilfe der Drehscheibe und Strobo.

2 Aufgabenstellung

Stellaris Robotermodell

Implementieren Sie eine Steuerung für das Robotersystem auf einem CARME-M4 Kit. Folgende Anforderungen sind zu erfüllen:

- Das CARME-Kit steuert den Ablauf des ganzen Robotersystems. Ihre Steuerungssoftware koordiniert die Transportbänder, den Verteiler und die beiden Roboterarme. Es sind gleichzeitig 3 ECTS auf dem Robotersystems unterwegs.
- Die Bedienung der Anlage wird mit Hilfe der Taster, Schalter und Poti auf dem Kit ausgeführt. Welche Funktionalität Sie implementieren ist Ihnen überlassen. Beispielsweise können Sie durch die Position eines Schalters veranlassen, dass die ECTS immer denselben Weg nehmen, oder ob Sie eine „∞“ fahren. Oder Sie können die Geschwindigkeit der Förderbänder mit dem Potentiometer einstellen.
- Geben Sie den Zustand der Anlage sinnvoll auf dem LCD aus.
- Verwenden Sie auch die UART für die Ausgabe von Programmabläufen, Fehlern usw.

Getränkroboter

Implementieren Sie eine Steuerung für den Getränkeroboter auf dem CARME-M4 Kit. Folgende Anforderungen sind zu erfüllen:

- Über den CAN-Bus können die einzelnen Komponenten angesprochen werden. So kann der Roboterarm ein Glas aus dem Regal auf die Waage stellen. Mit einer Flasche vom Drehteller wird das Glas aufgefüllt. Der aktuelle Füllzustand kann jeweils aus dem Knoten der Bearbeitungsstation (Waage) ausgelesen werden. Ist das Glas gefüllt kann es der Roboterarm auf das Förderband stellen, wo es dann zum Benutzer transportiert wird.
- Die Bedienung der Anlage wird mit Hilfe der Taster, Schalter und Poti auf dem Kit ausgeführt. Welche Funktionalität Sie implementieren ist Ihnen überlassen. Mit den Tastern kann beispielsweise die Auswahl für das gewünschte Getränk durchgeführt werden und die Ausgabe gestartet werden.
- Geben Sie den Zustand der Anlage sinnvoll auf dem LCD aus.
- Verwenden Sie auch die UART für die Ausgabe von Programmabläufen, Fehlern usw.

Echtzeitmodell Carme-M4 Kit

Implementieren Sie eine Steuerung für den Getränkeroboter auf dem CARME-M4 Kit. Folgende Anforderungen sind zu erfüllen:

- Der Benutzer übergibt dem System über die serielle Schnittstelle einen einfachen String (z.B. "FreeRTOS"). Das System stellt anschliessend den String mit Hilfe des Drehscheibe und dem Strobo dem Benutzer dar.
- Drehzahl der Drehscheibe sowie Blitzdauer des Strobo sollen eingestellt werden können.
- Die Bedienung der Anlage wird mit Hilfe der Taster, Schalter und Poti auf dem Kit ausgeführt. Welche Funktionalität Sie implementieren ist Ihnen überlassen. So könnte das Poti Beispielsweise die Geschwindigkeit der Drehscheibe steuern.
- Geben Sie den Zustand der Anlage sinnvoll auf dem LCD aus.
- Optional: Führen Sie Latenzzeitmessung durch: Messen Sie die Zeit welche vergeht wenn der Index-Interrupt ausgelöst wird bis Sie in der ISR oder dem verarbeitenden Task sind.

3 Vorschlag für das Vorgehen

- Arbeiten Sie sich zuerst in die Problemstellung ein. Zu jedem Modell ist eine detaillierte Beschreibung verfügbar.
- Machen Sie sich einen Zeitplan und aktualisieren Sie diesen laufend.
- Erstellen Sie ein Design für die Software. Dieses Design soll so dokumentiert sein, dass sich eine aussenstehende Person gut in ihren Code einarbeiten kann. In einem ersten Schritt werden Sie ein Grobdesign erstellen. Aktualisieren Sie dieses laufen!
- Spezifizieren Sie insbesondere auch die Schnittstellen.
- Teilen Sie die verschiedenen Aufgaben innerhalb ihrer Gruppe sinnvoll auf.
- Schreiben Sie den Code.
- Führen Sie laufend Tests durch.



4 Abgabe

Abzugebende Unterlagen

- Dokumentation Design: Dateiübersicht, Diagramme, alles was mir den Einstieg in Ihre Arbeit erleichtert (siehe auch Bewertungskriterien Design). Maximal 10 Seiten A4. Das Dokument soll keine Funktions- oder Code-Beschreibungen enthalten, diese erwarte ich als Doxygen-Kommentare im Source-Code.
- Kurze Bedienungsanleitung, 1 Seite A4
- Source-Code in elektronischer Form (exportiertes Atollic-Projekt).

Abgabetermin

Bitte alle Unterlagen in elektronischer Form auf CD oder per E-Mail an mich abgeben. Der Abgabetermin wird im Unterricht bekannt gegeben.