AUFGABE 1: HALLO WELT

Diese Aufgabe dient dem initialen Einrichten Ihrer Entwicklungsumgebung und dem ersten Kontakt mit dem Echtzeitbetriebssystem *eCos* und dem STM32F411-Board. Ziel ist es einen ersten Einblick in die Möglichkeiten der Entwicklungsumgebung zu erhalten.

Hinweis: Die Bedienung von Oszilloskopen ist Bestandteil einer folgenden Tafelübung.

1 Aufgabenstellung

1.1 Vorbereitung:

Teilaufgabe 1.

Kopieren und entpacken Sie die Vorgabe in ein beliebiges Arbeitsverzeichnis und setzen Sie die nötigen Umgebungsvariablen durch den Aufruf von `source ecosenv.sh` Nun können Sie die Makefiles generieren und die noch funktionslose Anwendung erstmals kompilieren. Sie können die erzeugte Anwendung mit Hilfe des Debuggers in den Flash-Speicher des Boards laden und starten.

☞ `source ecosenv.sh`
☞ `cd build`
☞ `cmake ..`
☞ `make`
☞ `make flash`

1.2 Fadensystem:

Die Erzeugung eines geeigneten Threadsystems ist Bestandteil der folgenden Aufgaben.

Teilaufgabe 2.

Implementieren Sie in `hello.c` die `cyg_user_start()`-Funktion, in der Sie mit Hilfe von `cyg_thread_create()` einen Thread erzeugen. Die Threadfunktion soll zunächst periodisch, jede Sekunde, mittels `ezs_printf()` die Zeichenkette „`Hallo Welt!\n`“ auf der seriellen Schnittstelle ausgeben. Sobald Sie das Board mit dem Mini-USB-Kabel an Ihrem PC angeschlossen haben, können Sie sich die serielle Ausgabe mit Hilfe von `cutecom` anzeigen lassen (`/dev/ttyACM1`, 115200 Baud, 8 Datenbits, ein Stopbit, keine Parität, kein Handshake).

Um einen periodischen Thread zu simulieren, verwenden Sie folgende Funktion:

`ezs_delay_us()`

Dieser Funktion können ganzahlige Werte übergeben werden, welche als Mikrosekunden interpretiert werden und eine entsprechende Verzögerung der Programmausführung bewirken.

Teilaufgabe 3.

Sehen Sie Nachteile, periodische Ausführung auf diese Art umzusetzen? Haben Sie eine Idee für einen sinnvolleren Ansatz? Halten Sie sich hier nicht mit diesen Fragen auf, sondern behalten Sie sie im Hinterkopf. Beantworten Sie die Frage im Laufe dieser Übungsaufgabe.

Antwort:

1.3 Debugging:

Teilaufgabe 4.

Für das Echtzeitverhalten ist neben der eigenen Implementierung immer das Gesamtsystem zu betrachten. Um ein Gefühl hierfür zu bekommen, setzen Sie im Debugger einen Breakpoint auf die Funktion

```
stm32_serial_putc_polled()
```

und setzen Sie die Ausführung fort. Sobald die Ausführung an dem gesetzten Breakpoint gestoppt hat, können Sie die Aufrufhierarchie im Backtrace-Fenster betrachten. Erkunden Sie den Aufrufgraph der Funktion `..._putc_polled()`. Wieso ist diese Funktion für Ihr Programm relevant?

Antwort:

Teilaufgabe 5.

Lassen Sie sich den Quellcode von `..._putc_polled()` im Debugger anzeigen. Welchen Zweck erfüllt die Funktion? Wie erbringt sie diese Leistung?

Antwort:

Teilaufgabe 6.

Welche aus Echtzeitsicht kritischen Stellen fallen in der Funktion besonders auf? Die Verwendung welcher Art von Funktionen ist also aus Sicht eines Echtzeitsystems im Allgemeinen problematisch?

Antwort:

Teilaufgabe 7.

Betrachten Sie nun die Funktionen `ezs_dac_write()` und `timer_set_oc_value()` im Debugger. Wie unterscheidet sie sich von `stm32_serial_putc_polled()` im Bezug auf das Echtzeitverhalten?

Antwort:

1.4 Signalerzeugung:

Mittels des periodischen Threads kann nun ein erstes Signal erzeugt werden. Sie können einen Timer des STM32F411-Prozessors mit Hilfe der Funktion `ezs_dac_write()` als Digital-Analog-Wandler verwenden (Wertebereich 0 bis $2^8 - 1$) und sich das erzeugte Signal am Oszilloskop anschauen. Verbinden Sie hierzu die Masseklemme des Tastkopfes mit dem Massepin der Filterplatine. Nun können sie mit Hilfe der Klemmprüfspitze das ungefilterte und das gefilterte Signal an den entsprechnden Pins ansehen.

Teilaufgabe 8.

Erzeugen Sie ein Sinussignal und verwenden Sie für diesen Zweck die Funktion `sinf()`. ☞ `math.h`
Verfolgen Sie bei Ihrer Implementierung einen systematischen Ansatz bei der Erzeugung des Signals und sorgen Sie dafür, dass die relevanten Parameter der Sinusschwingung flexibel einstellbar sind. Welche Parameter sind bei einer sinusförmigen Schwingung von Interesse?[1]

[1] `https://de.wikipedia.org/wiki/Schwingung#Harmonische_Schwingung`

Antwort:

Teilaufgabe 9.

Verringern Sie nun die Periode des Fadens drastisch (z. B. 2 ms). Was beobachten Sie? Wie müssen Sie Abtastrate und Parameter des Sinus wählen um ein Signal einer bestimmten Frequenz zu erzeugen?

Antwort:

Hinweise

- Bearbeitung: Gruppe mit je drei Teilnehmern.
- Abgabezeit: bis 04.11.2016
- Fragen bitte an `i4ezs@lists.cs.fau.de`