

Thema: Timer und Hardware Abstraction Layer

Aufgabe 1

Bauen Sie Ihre Implementierung aus Übungszettel 2, Aufgabe 2 so um, dass die Taster nicht mehr über einen externen Interrupt abgefragt werden, sondern in einem 50ms-Takt von einem Timer-Interrupt. Auch das Runterzählen des 3-Bit Zählers im Sekundentakt soll von einem Timer-Interrupt aus geschehen. Verwenden Sie bitte nur einen Timer, vorzugsweise Timer0. Bitte halten Sie die Implementierung der ISRs so klein wie möglich und signalisieren ggf. Veränderungen über Events. Nutzen Sie dafür die Event-Implementierung, die Sie als Teil des „TimerHAL“-Projektes im GRIPS-Kurs finden (oder schreiben Sie diese selbst).

Abgabe Aufgabe 1: Vorführung

Aufgabe 2

- a) Entwerfen und implementieren Sie ein Konzept für Software-Timer (analog zu den Timern, die Sie im 2. Semester in RB implementiert hatten): Der Programmierer kann über einen Funktionsaufruf namens `declareTimer(...)` einen neuen Timer anlegen, der jedoch noch nicht gestartet wird. Der Timer wird mittels einer Funktion `startTimer(...)` gestartet und kann mittels `cancelTimer(...)` gestoppt werden. Die Timer sollen jeweils mit einer Callback-Funktion ausgestattet werden, die aufgerufen wird, sobald der Timer abgelaufen ist. Alle Timer sollen von einem Interrupt aus Timer1 abgearbeitet werden, die Zeitspannen, die für einen Timer festgelegt werden können, liegen im Bereich 1µs bis 1s. Sie können davon ausgehen, dass es nicht mehr als 5 Timer in einem System gibt. Achten Sie bei der Implementierung darauf, an welchen Stellen Sie volatile verwenden müssen und an welchen Stellen ggf. Interrupts unterbunden werden müssen. Erarbeiten Sie Ihr Konzept bitte so, dass keine Interna der Timer-Implementierung benutzt werden müssen/dürfen.
- b) Bauen Sie anschließend Ihre Implementierung aus Aufgabe 1 so um, dass für die zeit-gesteuerten Funktionalitäten die in a) implementierten Software-Timer verwendet werden.
- c) Überlegen Sie sich ein Konzept, wie Sie diese Timer in das HAL-Konzept und insbesondere in den HALGenerator integrieren können (nur Konzept, keine Implementierung gefordert). Gibt es Vorteile, wenn der HALGenerator die Timer und ihre Zeitspannen kennt? Wenn ja welche und wie würde man das umsetzen?

Abgabe Aufgabe 2: Vorführung + mündlich