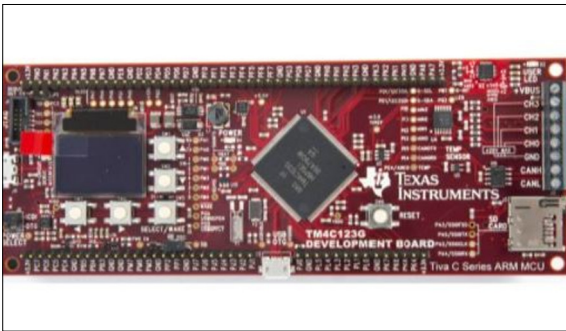
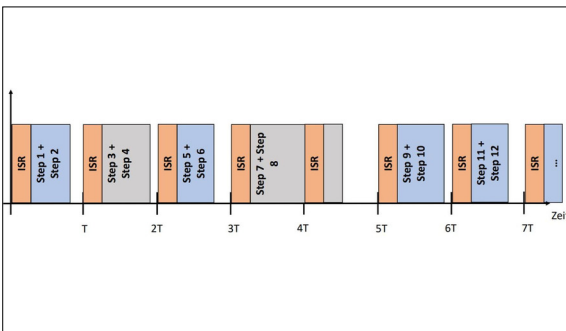


Student	Pascal Engeler
Examinatoren	Prof. Dr. Markus Kottmann, Bruno Vollenweider
Themengebiet	Regelungstechnik

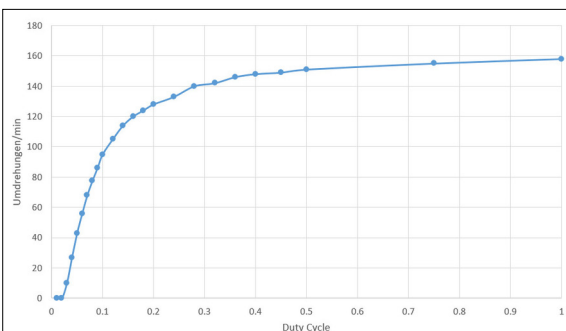
## Realtime-System mit Simulink auf einem Cortex M4



Development Kit mit dem Mikroprozessor (Mitte) und dem darin eingebetteten Cortex-M4 Prozessorkern



Beispiel eines Zeitr Rahmens, bei der 2 Simulink-Schritte in der vorgegebenen Zeit T durchgeführt werden sollen



Ausgemessene (stationäre) Drehzahlen des DC-Motors in Abhängigkeit des Verhältnisses Pulsweite/Periode

**Aufgabenstellung:** Regelalgorithmen werden heutzutage zunehmend mithilfe von Software-Paketen wie MATLAB oder Simulink entworfen. Um einen neuen Regler möglichst schnell testen zu können, kann in Simulink C-Code erstellt werden. Je nach Zielhardware müssen noch gewisse Anpassungen implementiert werden. Daher soll in dieser Arbeit ein Realtime Frame auf einem Tiva DK-TM4C123G von Texas Instrument erarbeitet werden. Dieses Development Kit basiert auf dem häufig verwendeten Cortex M4 Prozessor. Dabei wird das Interface sowie das Realtime Frame direkt in C/C++ implementiert und anschliessend wird der in Simulink erzeugte Code in diesem Frame eingebettet. Somit müssen nur jeweils die von Simulink erstellten Files ausgetauscht werden, um einen neuen Regler zu implementieren. Um das System zu testen, soll am Ende ein Regler für einen DC Motor erarbeitet und umgesetzt werden.

**Vorgehen:** Um aus dem Simulink-Modell C-Code zu erstellen, wurde der "Embedded Coder", ein Plug-In von Simulink, gewählt. Dieser erzeugt einen effizienten, für den Cortex-M4 geeigneten Code.

Danach wurden verschiedene Modelle für das Realtime-Frame erarbeitet und das geeignetste ausgewählt. Das ausgewählte Modell stellt eine bestimmbare Zeit zur Verfügung, bei der eine wiederum bestimmbare Anzahl "Simulink-Schritte" ausgeführt werden können (siehe mittleres Bild). Wird diese Zeit überschritten, wird eine Variable inkrementiert und der Rest der "Simulink-Schritte" wird noch ausgeführt. Durch die Wahl der Anzahl "Simulink-Schritte" kann die Sensitivität des Frames angepasst werden.

Der Mikrocontroller enthält neben dem Cortex auch etliche Peripheriemodule, unter anderem PWM's, ADC's, Quadratur-Encoder-Interface, UART-Schnittstellen etc. Die wichtigsten dieser Module wurden ins Frame implementiert.

Am Schluss wurde ein kleiner DC-Motor als Demo-Anwendung ins Projekt integriert. Dazu steuerte man den Motor mittels PWM-Signalen und einer Treiberstufe an. Die Signale des Quadratur-Encoders im Motor konnten direkt ins oben genannte QEI-Modul geleitet werden, welches die Position des Rotors wiedergeben hat. Mit einem Simulink-Modell wurde schlussendlich ein Positionsregler implementiert.

**Fazit:** Die Einarbeitung in die zur Verfügung stehenden Bibliotheken des Mikrocontrollers dauerte einiges länger als zuerst geplant war. Trotzdem konnte ein erster Entwurf des Frames erstellt werden, welcher die wichtigsten Funktionen ausführen kann.

Das Wunschziel der Verwendung vom "external mode" in Simulink, welches während des Betriebs Änderungen von Parametern des Modells ermöglichen würde, konnte aus Zeitgründen nicht realisiert werden. Der Positionsregler konnte noch kurzfristig realisiert werden, dieser ist jedoch nicht unter mechanischer Belastung geeignet und hat viel Spielraum für Verbesserungen.