

<b>Ostfalia</b> Hochschule für angewandte Wissenschaften  Fakultät Fahrzeugtechnik Prof. Dr.-Ing. V. von Holt Institut für Fahrzeuginformatik und Fahrzeugelektronik		Modulprüfung	Name:.....
		Mess- und	Vorname:.....
		Mikroprozessortechnik	Matr.Nr.:.....
		WS 2011/12 09.01.2012	Unterschrift:.....

Zugelassene Hilfsmittel: **Einfacher Taschenrechner**  
Zeit: 120 Minuten (Modulprüfung)

**Punkte:**

1	2	3	Punktsumme (max. 50)	Prozente (max. 50%)

**Mikroprozessortechnik (50%)   Messtechnik / Sensorik (50%)      Gesamt**

--	--	--

### Aufgabe 1 (18 Punkte) – Assembler / Stack

Das folgende Listing enthält den Rahmen eines Assemblerprogramms für einen Atmel AVR-RISC-Prozessor. In einem Hauptprogramm „Main“ wird ein Unterprogramm „Subroutine“ zur Berechnung eines arithmetischen Ausdrucks aufgerufen. Die **Parameterübergabe** und die **Ergebnisrückgabe** sollen über den **Stack** erfolgen. Im Unterprogramm sollen keine Register dauerhaft verändert werden, d.h. die im Unterprogramm benutzten Register sind zu Beginn des Unterprogramms zu sichern und vor Verlassen des Unterprogramms wiederherzustellen.

a) (12 P) Ergänzen Sie das Listing derart, dass

1. im Hauptprogramm „Main“ die Parameter vor Aufruf des Unterprogramms „Subroutine“ aus den angegebenen Speicherstellen auf den Stack geladen werden,
2. die Parameter im Unterprogramm so benutzt werden, wie dies in den dort enthaltenen Kommentaren ersichtlich ist,
3. der Rückgabewert im Unterprogramm auf den Stack gelegt wird, und
4. der Rückgabewert im Hauptprogramm wieder vom Stack geholt und an der angegebenen Speicherstelle abgelegt wird.

b) (6 P) Skizzieren Sie den Aufbau/Inhalt des Stacks zu dem Zeitpunkt, der im Listing gekennzeichnet ist!

```

        .cseg
Main:
    ...
    ; Alle notwendigen Initialisierungen des Stacks sind bereits
    ; erfolgt. Fügen Sie hier nur die Befehle ein, welche die
    ; Parameter über den Stack übergeben!

    ; Aufruf des Unterprogramms zur Berechnung
    call Subroutine
    ; Fügen Sie hier nur die Befehle ein, um den Stack zu
    ; bereinigen!

    ...

        .dseg
    ; Die Eingabeparameter a und b können als initialisiert
    ; angenommen werden.
a:     .BYTE      1
b:     .BYTE      1
c:     .BYTE      1

        .cseg

    ; Unterprogramm Subroutine
    ; Berechnung von  $c = 1.5 * a + 3 * b$ 
    ; Eingangsparameter:
    ; a: 1 Byte
    ; b: 1 Byte
    ; Rückgabewert:
    ; c: 1 Byte
    ; Parameter sollen auf dem Stack übergeben werden.
    ; Register dürfen vom Unterprogramm nicht verändert werden.

```

Subroutine:

```
;
; Hier: Skizze vom Aufbau des Stacks an dieser Programmstelle
;
; An dieser Stelle wird
; Parameter a in r0 und
; Parameter b in r1 erwartet.
; Berechnung von  $c = 1.5 * a + 3 * b$ 
mov  r2,r0
asr  r2
add  r0,r2
mov  r2,r1
asl  r2
add  r1,r2
add  r0,r1
; An dieser Stelle steht
; das Ergebnis c in r0.
```

Stack

ret

## Aufgabe 2 (18 Punkte) – Timer

Gegeben sei ein mit 8 MHz getakteter Mikroprozessor, der über eine 16-Bit-Timereinheit verfügt.

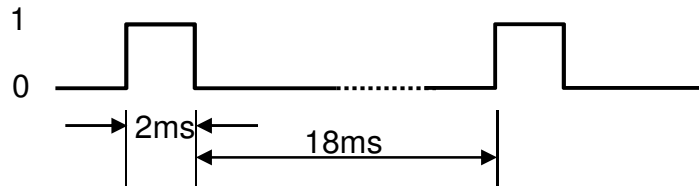
a) (5 P) Skizzieren Sie schematisch die Hauptkomponenten der Timereinheit eines Mikrocontrollers!

b) (1 P) Wie groß ist die Periodendauer des Timers ohne Vorteiler?

c) (2 P) Skizzieren Sie in nachfolgendem Diagramm den Verlauf des Zählerstandes über der Zeit für 2 Perioden!



Der Mikroprozessor soll nun dazu dienen, einen Synchronimpuls für ein analoges Videosignal zu erzeugen. Das zu erzeugende Synchronsignal soll den folgenden Verlauf haben:



- d) (4 P) Stellen Sie den Ablaufplan zur Realisierung des Synchronsignals mithilfe des 16-Bit-Timers dar. (Das Synchronsignal wird an einem digitalen Port ausgegeben.) Achten Sie insbesondere auf die Steuerung des Timers! Die Portausgabe können Sie abstrakt als „Setze Portbit auf 0 bzw. 1“ darstellen.

- e) (3 P) Welche(n) Vorteiler und welche(n) Startwert(e) wählen Sie? (Mögliche Vorteilerwerte: 2, 4, 8, 16, 32, 64, 128, 256)

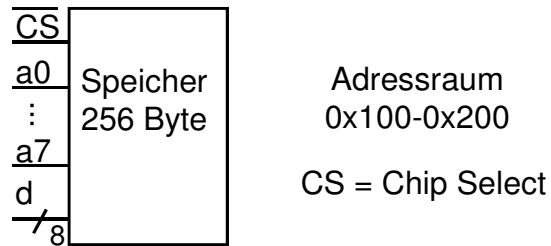
- f) (3 P) Skizzieren Sie den Verlauf des Zählerstands für die Erzeugung des Synchronsignals über 2 Perioden dar unter Angabe von Zahlenwerte für die „Eckpunkte“!



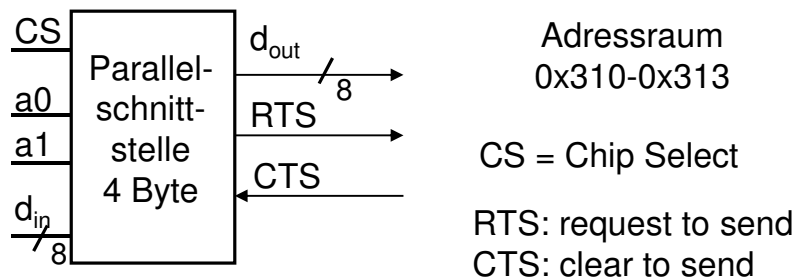
### Aufgabe 3 (14 Punkte) – Adressdekodierung und Datenübertragung

Ein 8-Bit-Mikroprozessorsystem ist mit einem RAM-Speicher von 256 Byte im Adressbereich 0..255 bestückt. Das System soll um einen Speicher im Adressbereich 256..511 (0x100..0x200) sowie um einen Parallelschnittstellenbaustein im Adressbereich 784..787 (0x310..0x313) erweitert werden. Das Mikroprozessorsystem verfügt über einen Adressbereich von 4096 Byte, entsprechend einem 12-Bit-Adressbus.

- a) (4 P) Entwerfen Sie die Adressdekodierung für den u.a. Speicherbaustein mit 256 Byte, der im Adressbereich 256..512 angesprochen werden soll! Geben Sie die Boolesche Formel für das Signal #CS an!



- b) (4 P) Entwerfen Sie analog zu Aufgabenteil a) die Dekodierung für den Parallelschnittstellenbaustein. Dieser belegt 4 Adressen und soll im Adressbereich 784..787 angesprochen werden. Geben Sie die logische Formel für das CS-Signal an!



- c) (2 P) Der Parallelschnittstellenbaustein verfügt über 2 Leitungen (RTS - Request-to-Send / CTS – Clear-to-Send) für ein Hardware-Handshaking. Erläutern Sie anhand einer Skizze das Zusammenspiel des Parallelschnittstellenbausteins mit einem Peripheriegerät zur Datenübertragung mit Handshaking!
- d) (2 P) Unter der Annahme, dass die Übertragung eines Datums inklusive des Handshaking 0,25ms dauert, wie viele Daten können dann in einer Sekunde übertragen werden?
- e) (2 P) Wenn anstelle der Parallelschnittstelle eine Serielle RS 232-Schnittstelle verwendet wird, wie lange dauert dann die Übertragung eines Datums unter der Annahme, dass 1 Startbit, 1 Paritätsbit und 2 Stopbit verwendet werden und der Übertragungstakt dem der Parallelschnittstelle entspricht? Wie viele Daten können dann in einer Sekunde übertragen werden?