


Ostfalia Hochschule für angewandte Wissenschaften Fakultät Fahrzeugtechnik M.Sc. Torsten Wylegala Institut für Fahrzeugsystem- und Servicetechnologien		Modulprüfung	Name:.....
		Mikroprozessortechnik BPO 2011	Vorname.....
		WS 2020/2021 03.03.2021	Matr.Nr.:.....
			Unterschrift.....

Zugelassene Hilfsmittel:

- **Einfacher Taschenrechner**
- **Folien und Notizen aus Vorlesung und Labor**

Zeit:

- 5 Minuten Vorbereitung
- 60 Minuten Bearbeitungszeit
- 10 Minuten Abgabe

Prüfungsform:

- Bei der Prüfungsform handelt es sich um eine schriftliche Onlineklausur, die in Einzelarbeit auf Papier vor Ihrem Computer bearbeitet werden muss.
- Für die Dauer der Prüfung muss mit eingeschalteter Webcam an einer Big Blue Button (BBB) Session teilgenommen werden. Dies dient der Kontrolle, um sicherzustellen, dass Sie die Klausur alleine bearbeiten (es wird keine Aufzeichnung angefertigt).
- Wie bei einer normalen Klausur können Fragen zur Aufgabenstellung gestellt werden.
- Jegliche Kommunikation untereinander oder mit Dritten ist während des Bearbeitungszeitraums untersagt.
- Das Internet ist als Hilfsmittel nicht zugelassen.
- Die Klausur wird Ihnen zu Beginn der Prüfung über das Stud.IP als PDF bereitgestellt.
- Die Bearbeitung der Klausuraufgaben erfolgt auf Papier. Sie müssen die Klausur nicht ausdrucken. Achten Sie aber bitte darauf die Nummern der Aufgaben klar ersichtlich zu notieren.
- Für die Abgabe der Aufgaben wird ein Hausaufgabenordner im Stud.IP angelegt.
- Für die Digitalisierung Ihrer Arbeitsblätter stehen Ihnen 10 extra Minuten am Ende der Klausur zur Verfügung. Sie können einen Scanner oder eine Digitalkamera/Smartphone für die Digitalisierung verwenden (achten Sie bitte darauf, dass die Aufnahmen scharf und leserlich sind).

Punkte:

1 (6)	2 (22)	3 (6)	4 (12)	5 (14)	Punktsumme (max. 60)	Prozente	Note

Tabelle HEX-Ziffern – Binärcode

F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
1111	1110	1101	1100	1011	1010	1001	1000	0111	0110	0101	0100	0011	0010	0001	0000

Aufgabe 1 (6 Punkte) – Kurzfragen

Σ	
----------	--

Kreuzen Sie an, ob die folgenden Aussagen richtig oder falsch sind. **Falsche** Antworten führen zu einem **Punktabzug**. (Die Aufgabe ergibt aber keine negative Gesamtpunktzahl.)

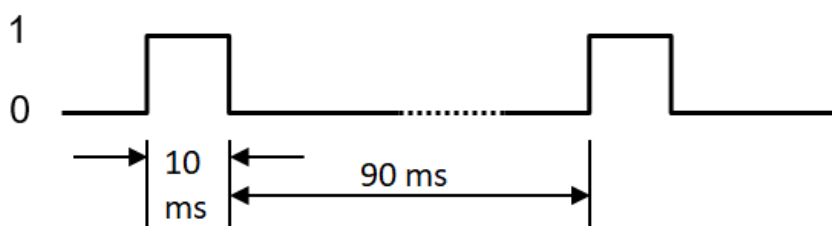
Aussage	richtig	falsch
Beim I2C-Bus gibt es nur einen Master.		
Das Dirty-Bit eines Cache-Eintrags zeigt an, dass der Eintrag von mehreren Prozessen genutzt wird.		
Die Datenübertragung beim SPI-Bus ist vergleichsweise schnell, weil über die Busleitungen keine Protokollinformationen übertragen werden.		
Statisches RAM speichert die Informationen auch im ausgeschalteten (stromlosen) Zustand		
Ein Watchdog-Timer setzt den Prozessor im Fehlerfall zurück.		
Die Bits im Statusregisters eines Prozessors werden durch Rechenoperationen der ALU beeinflusst.		

Aufgabe 2 (22 Punkte) – Timer

Gegeben sei ein mit 10,5 MHz getakteter Mikroprozessor, der über eine 16-Bit-Timereinheit verfügt. Der Timer verfügt über ein Zählerstandsregister TCNT und ein ladbares Vergleichsregister OCR. Bei Erreichen des Werts in OCR wird das Überlauf-Bit OVF gesetzt und TCNT auf 0 zurückgesetzt.

- (1 P) Berechnen Sie die Periodendauer des Mikrocontrollers!
- (1 P) Wie groß ist die Periodendauer des Timers ohne Vorteiler?
- (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:



- (4 P) Wählen Sie den/die Vorteilerwert(e) so, dass Sie die vorgegebenen Zeiten möglichst genau einhalten! (Mögliche Vorteilerwerte: (1), 4, 8, 32, 128, 256, 1024)

- e) (4 P) Welche(n) Vergleichswert(e) für OCR ergibt/ergeben sich?
- f) (2 P) Welche absoluten Abweichungen zum geforderten Signal erhalten Sie? Durch welche Modifikation des Timers könnten Sie die Abweichung verringern?
- g) (4 P) Skizzieren Sie den Verlauf des Zählerstands für die Erzeugung des Synchronsignals bis zur Zeit $t = 110\text{ms}$ unter Angabe von Zahlenwerten für die „Eckpunkte“!
- h) (4 P) Stellen Sie den Ablaufplan/Pseudocode 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 als „PB=0“ (Portbit auf ,0' setzen) bzw. „PB=1“ (Portbit auf ,1' setzen) darstellen.

Aufgabe 3 (6 Punkte) – Rechnerarchitektur/Speicherhierarchien

Σ	
----------	--

- a) (2 P) Worin besteht der Unterschied zwischen vollassoziativem, mehrfach-assoziativen und direkten Cachespeichern bezogen auf den Hardwareaufwand?
- b) (4 P) Ein Mikrorechnersystem mit einem Adressraum von 16 MByte verfügt über einen 4-fach-assoziativen Cachespeicher mit 8 Sätzen und einer Block(Satz)-Größe von 16 Byte . Aus welchen Adressbits wird der Cache-Satz bestimmt und welche Adressbits dienen als Tag?

Aufgabe 4 (12 Punkte) – Adressierdekodierung/Bussysteme

Σ	
----------	--

Ein **16-Bit**-Mikrorechner verfügt über einen Adressraum von **386kByte**, der mit folgenden Bausteinen belegt ist:

- ROM-Baustein **ROM** mit **32kByte** Größe bei Adresse **0x0000**
- RAM-Baustein **RAM** mit **128kByte** Größe
- I/O-Baustein **I/O** mit **16** Registern bei Adresse **0x4FFF0**

a) (1 P) Wie viele Adressleitungen umfasst der Adressbus des Mikrorechners?

b) (1 P) Wie viele Adresseingänge besitzt der **ROM**-Baustein?

c) (5 P) Platzieren Sie den RAM-Baustein so im noch verfügbaren Adressbereich, dass die Dekodierung möglichst einfach ist!
Welchen Adressbereich belegt dann der **RAM**-Baustein?
Wie lautet dann die CS-Logik für den **RAM**-Baustein?

d) (5 P) Bestimmen die CS-Logik für den **I/O**-Baustein

Aufgabe 5 (14 Punkte) – Assemblerprogrammierung mit Modellrechner

Σ	
----------	--

Befehlssatz des Beispielprozessors:

Beschreibung	Mnemonic (Assembler)	Maschinencode
Registers Rr <- 0	clr Rr	000r xxxx
Register Rr <- Speicher(m)	ldm Rr,m	001r mmmm
Speicher(m) <- Register Rr	stm m,Rr	010m mmmr
Rr <- Rr + Rs	add Rr,Rs	011r sxxx
Rr <- Rr - Rs	sub Rr,Rs	100r sxxx
Rr <- Rr * Rs	mul Rr,Rs	101r sxxx
Prozessor anhalten	halt	110x xxxx

- a) (1 P) Schreiben Sie ein Assemblerprogramm, das mit dem obenstehenden Befehlssatz, die folgende Berechnung durchführt:

$$E = (A + B) * (C + D)$$