

# Frage 1

## Kurzfragen Microcontroller Basics

Bitte beurteilen Sie die folgenden Aussagen:

Bei einem Half-Word Schreibzugriff auf dem 32bit-Systembus sind genau zwei NBL[x]-Signale aktiv.

Bei einer synchronen Datenübertragung verwendet ein Slave das Clock-Signal vom Master.

Die CPU kann Slaves am Systembus mittels Control-Bits konfigurieren.

Der Systembus besteht aus den zwei Bestandteilen Datenbus und Kontrollsignale.

Die CPU kann Slaves am Systembus mittels Status-Bits konfigurieren.

Der Systembus übermittelt unter anderem die Kontrollsignale.

# Frage 1 Lösung

## Kurzfragen Microcontroller Basics

Bitte beurteilen Sie die folgenden Aussagen:

Bei einem Half-Word Schreibzugriff auf dem 32bit-Systembus sind genau zwei NBL[x]-Signale aktiv.

Wahr

Bei einer synchronen Datenübertragung verwendet ein Slave das Clock-Signal vom Master.

Wahr

Die CPU kann Slaves am Systembus mittels Control-Bits konfigurieren.

Wahr

Der Systembus besteht aus den zwei Bestandteilen Datenbus und Kontrollsignale.

Falsch

Die CPU kann Slaves am Systembus mittels Status-Bits konfigurieren.

Falsch

Der Systembus übermittelt unter anderem die Kontrollsignale.

Wahr

# Frage 2

## Partielle Dekodierung

Gegeben ist ein System mit einem **8bit-Adressbus**. Sie untersuchen eine Peripherie und stellen fest, dass sie genau auf den Adressen **0x6D**, **0x7F**, **0x6F** und **0x7D** selektiert ist; offensichtlich ein Fall von partieller Adressdekodierung.

Bitte beantworten Sie die folgenden Fragen:

**Wieviele der 8 Adressleitungen werden nicht dekodiert bzw. ignoriert?**

**Geben Sie die Nummern der ignorierten Adressleitung(en) an!**

*Wenn es mehrere Leitungen sind, geben Sie die Nummern aufsteigend, getrennt durch Leerschläge an, z.B. 0 1 2 3 4; Leitung 0 ist wie üblich das LSB.*

# Frage 2 Lösung

## Partielle Dekodierung

Gegeben ist ein System mit einem **8bit-Adressbus**. Sie untersuchen eine Peripherie und stellen fest, dass sie genau auf den Adressen **0x6D**, **0x7F**, **0x6F** und **0x7D** selektiert ist; offensichtlich ein Fall von partieller Adressdekodierung.

Bitte beantworten Sie die folgenden Fragen:

**Wieviele der 8 Adressleitungen werden nicht dekodiert bzw. ignoriert?**

2

**Geben Sie die Nummern der ignorierten Adressleitung(en) an!**

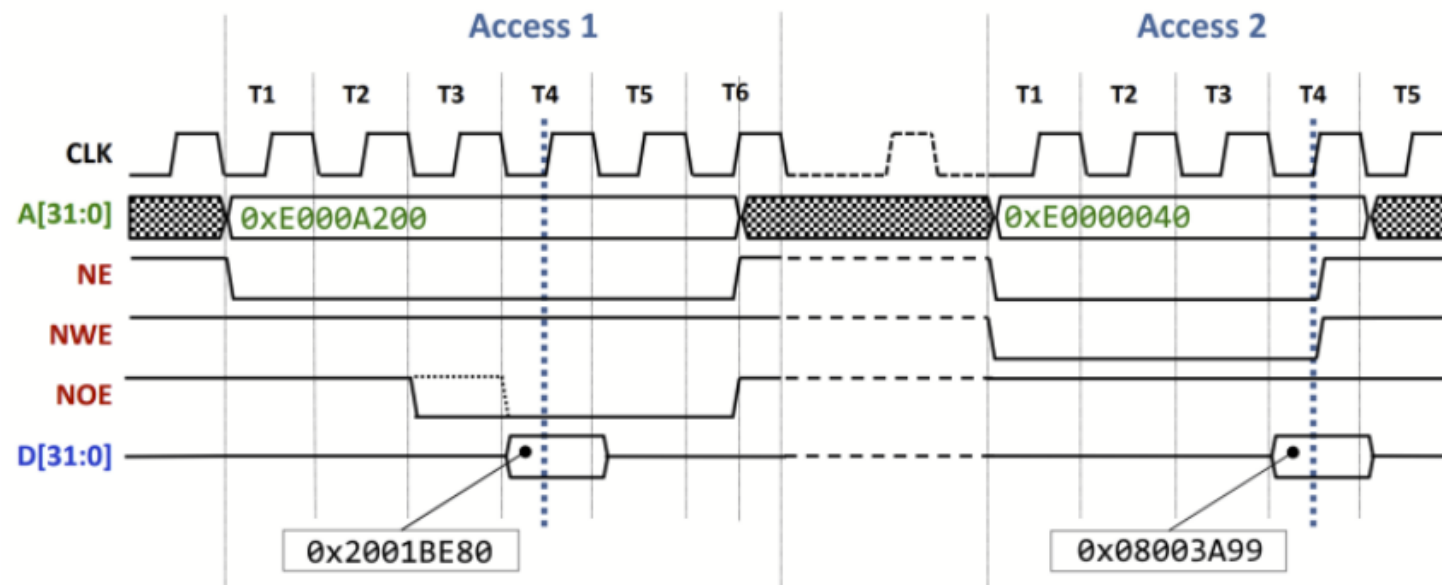
*Wenn es mehrere Leitungen sind, geben Sie die Nummern aufsteigend, getrennt durch Leerschläge an, z.B. 0 1 2 3 4; Leitung 0 ist wie üblich das LSB.*

1 4

# Frage 3

## Buszugriff

Gegeben ist das folgende Diagramm eines Buszugriffs:



Adresse (aufsteigend) Daten-Byte


Tragen Sie alle Bytes des **Write-Zugriffs** in die untenstehende Tabelle ein. Geben Sie für jedes Byte die Adresse und den geschriebenen Wert an; der Prozessor ist **little endian**.

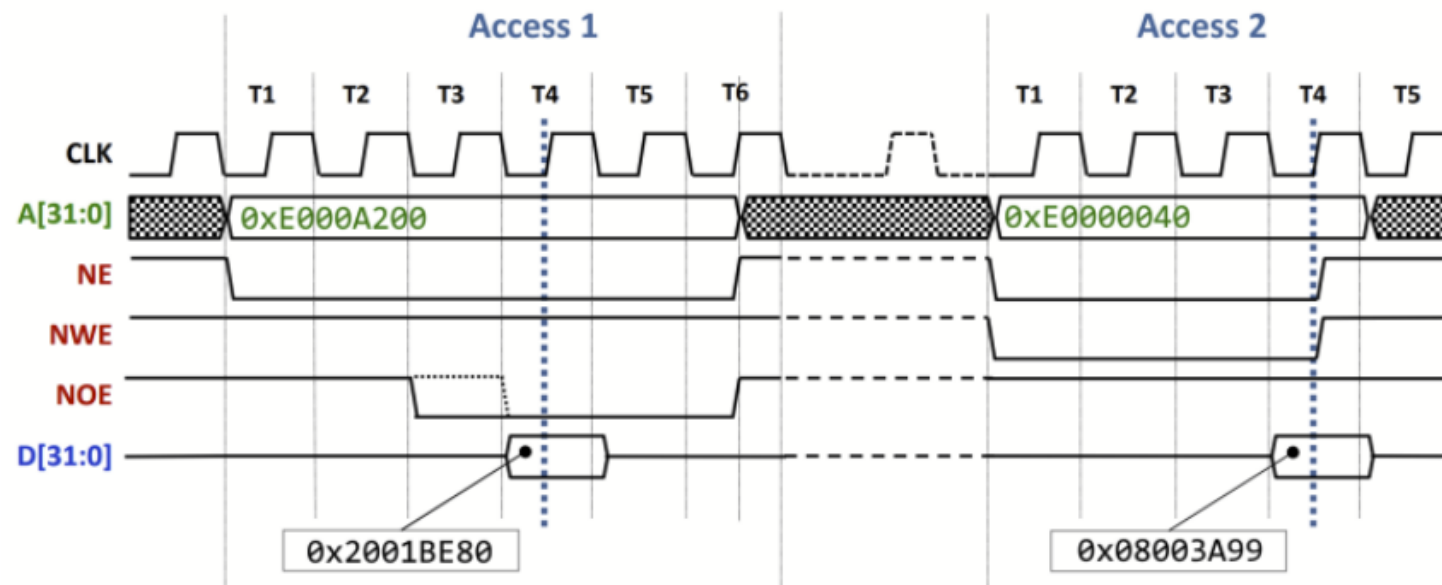
Verwenden Sie für Adressen die folgende Form: 0x..... (achtstellige Hexzahl, Zeichen 0 bis F)

Verwenden Sie für Daten die folgende Form: 0x.. (zweistellige Hexzahl, Zeichen 0 bis F)

# Frage 3 Lösung

## Buszugriff

Gegeben ist das folgende Diagramm eines Buszugriffs:



Tragen Sie alle Bytes des **Write-Zugriffs** in die untenstehende Tabelle ein. Geben Sie für jedes Byte die Adresse und den geschriebenen Wert an; der Prozessor ist **little endian**.

Verwenden Sie für Adressen die folgende Form: 0x..... (achtstellige Hexzahl, Zeichen 0 bis F)

Verwenden Sie für Daten die folgende Form: 0x.. (zweistellige Hexzahl, Zeichen 0 bis F)

Adresse (aufsteigend) Daten-Byte

0xE0000040	0x99
0xE0000041	0x3A
0xE0000042	0x00
0xE0000043	0x08

# Frage 4

## GPIO Treiberstufen

Wir betrachten zwei GPIO-Pins. Einer ist als Output-Pin ('out') konfiguriert, der andere als Input-Pin ('in'). Die beiden Pins sind über eine Leitung miteinander verbunden.



Der Output-Pin 'out' ist als **push-pull-Treiberstufe ohne pull-Widerstand** konfiguriert.

Geben Sie an, welcher Wert am Input-Pin 'in' erkannt wird, je nachdem wie der pull-Widerstand am Input-Pin konfiguriert ist:

Pull-Widerstand am Input-Pin	Output '0': Gelesener Input-Wert	Output 'floating': Gelesener Input-Wert
kein	<input type="text"/>	<input type="text"/>
pull-up	<input type="text"/>	<input type="text"/>
pull-down	<input type="text"/>	<input type="text"/>

# Frage 4 Lösung

## GPIO Treiberstufen

Wir betrachten zwei GPIO-Pins. Einer ist als Output-Pin ('out') konfiguriert, der andere als Input-Pin ('in'). Die beiden Pins sind über eine Leitung miteinander verbunden.



Der Output-Pin 'out' ist als **push-pull-Treiberstufe ohne pull-Widerstand** konfiguriert.

Geben Sie an, welcher Wert am Input-Pin 'in' erkannt wird, je nachdem wie der pull-Widerstand am Input-Pin konfiguriert ist:

Pull-Widerstand am Input-Pin	Output '0': Gelesener Input-Wert	Output 'floating': Gelesener Input-Wert
kein	0	undefiniert
pull-up	0	1
pull-down	0	0



# Frage 5

## GPIO

Sie sollen einen GPIO-Port des STM32F429 konfigurieren. Die notwendigen Informationen finden Sie in den Folien zu GPIO oder im Reference Manual.

Konfigurieren Sie **GPIO Port A.5** als **low speed digitalen Output** mit **open-drain** und **pull-up**.

Geben Sie die Basisadresse der Control- und Statusregister des GPIO Ports A in Hexadezimal-Schreibweise an (0x...):

Geben Sie in der folgenden Tabelle die Offsets der Kontrollregister an, sowie die zu setzenden Bitmuster und um wie viele Stellen die Bitmuster geschoben werden müssen.

**Formatvorgabe:** Für die zu setzenden Bits wählen Sie aus dem Auswahlmene die richtige Maske aus. Für die Shifts geben Sie die Anzahl Stellen an.

Register	Offset	Bits: Bitmaske in binär	Shift um:
Beispiel:	0x1F	01	7
MODER	<input type="text"/>	<input type="text"/>	<< <input type="text"/>
OTYPER	<input type="text"/>	<input type="text"/>	<< <input type="text"/>
PUPDR	<input type="text"/>	<input type="text"/>	<< <input type="text"/>
OSPEEDR	<input type="text"/>	<input type="text"/>	<< <input type="text"/>

# Frage 5 Lösung

## GPIO

Sie sollen einen GPIO-Port des STM32F429 konfigurieren. Die notwendigen Informationen finden Sie in den Folien zu GPIO oder im Reference Manual.

Konfigurieren Sie **GPIO Port A.5** als **low speed digitalen Output** mit **open-drain** und **pull-up**.

Geben Sie die Basisadresse der Control- und Statusregister des GPIO Ports A in Hexadezimal-Schreibweise an (0x...):

Geben Sie in der folgenden Tabelle die Offsets der Kontrollregister an, sowie die zu setzenden Bitmuster und um wie viele Stellen die Bitmuster geschoben werden müssen.

**Formatvorgabe:** Für die zu setzenden Bits wählen Sie aus dem Auswahlmeneu die richtige Maske aus. Für die Shifts geben Sie die Anzahl Stellen an.

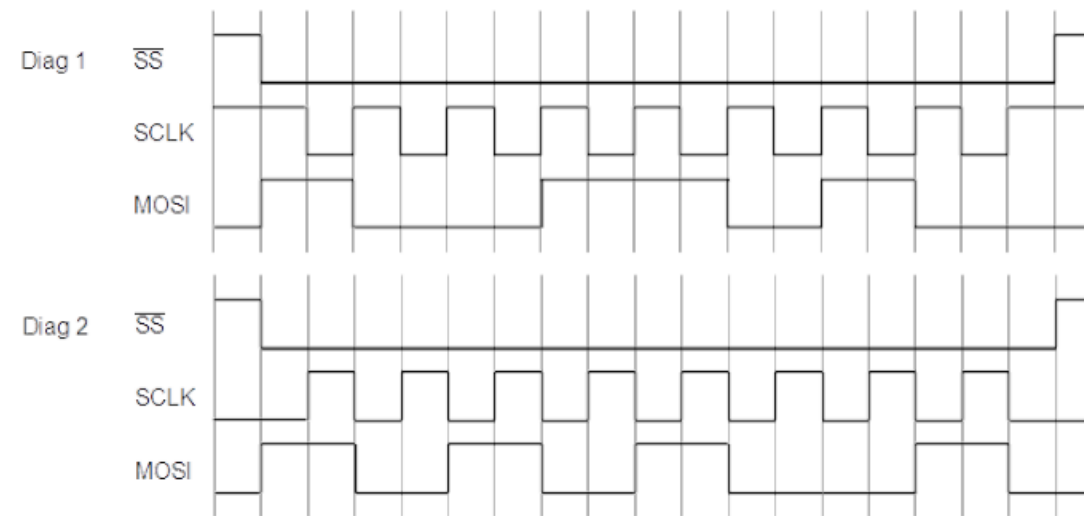
Register	Offset	Bits: Bitmaske in binär	Shift um:
Beispiel:	0x1F	01	7
MODER	<input type="text" value="0x00"/>	<input type="text" value="01"/>	<< <input type="text" value="10"/>
OTYPER	<input type="text" value="0x04"/>	<input type="text" value="1"/>	<< <input type="text" value="5"/>
PUPDR	<input type="text" value="0x0C"/>	<input type="text" value="01"/>	<< <input type="text" value="10"/>
OSPEEDR	<input type="text" value="0x08"/>	<input type="text" value="00"/>	<< <input type="text" value="10"/>

# Frage 6

## SPI Timing Diagramm

Eine SPI Schnittstelle ist wie folgt konfiguriert: CPOL=1, CPHA=0, MSB first.

Der Master sendet das Byte 0x59. Bildet eines der untenstehenden Diagramme den Verlauf korrekt ab, und wenn ja, welches?



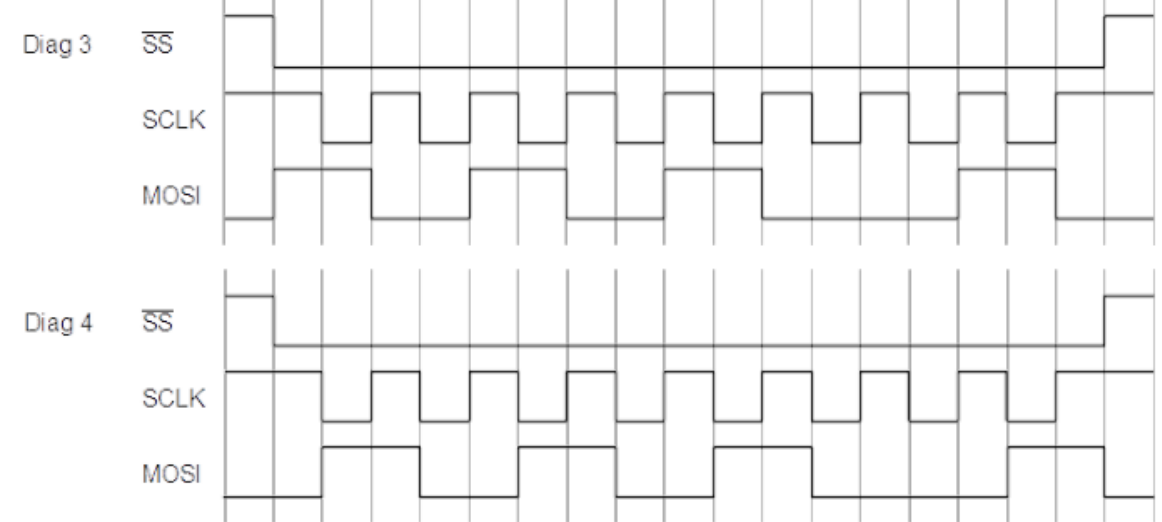
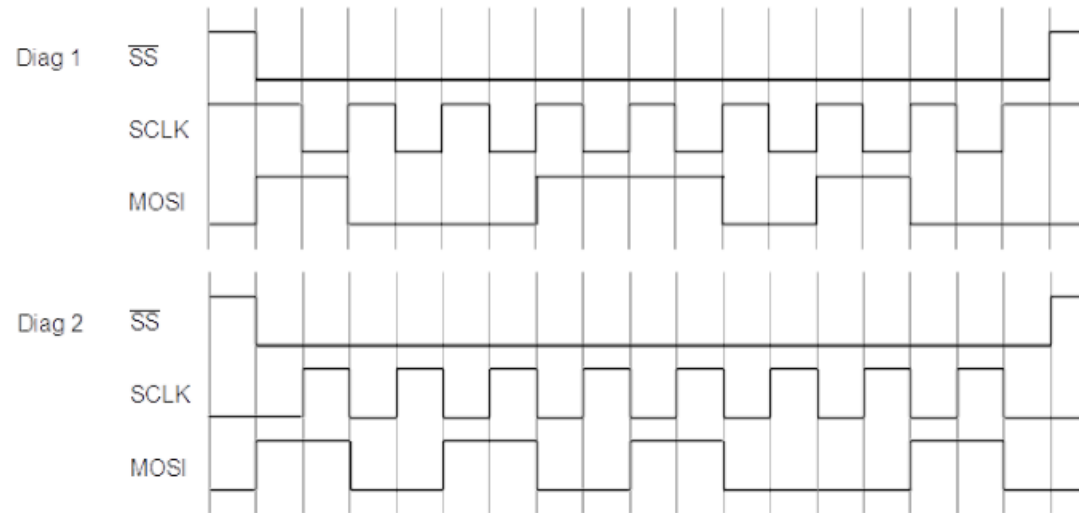
# Frage 6 Lösung

## SPI Timing Diagramm

Eine SPI Schnittstelle ist wie folgt konfiguriert: CPOL=1, CPHA=0, MSB first.

Der Master sendet das Byte 0x59. Bildet eines der untenstehenden Diagramme den Verlauf korrekt ab, und wenn ja, welches?

Keines, Diag 1 ist LSB nicht MSB



# Frage 7

## Kurzfragen SPI

Bitte beurteilen Sie die folgenden Aussagen:

SPI wird auch 2-wire bus genannt.

SPI ist für Onboard-Verbindungen geeignet.

Bei SPI müssen neben den Datenbits sogenannte Synchronisationsbits übertragen werden.

SPI ist eine synchrone Verbindung.

Bei SPI braucht es zu jedem Slave eine separate Slave-Select-Leitung.

# Frage 7 Lösung

## Kurzfragen SPI

Bitte beurteilen Sie die folgenden Aussagen:

SPI wird auch 2-wire bus genannt.

Falsch

SPI ist für Onboard-Verbindungen geeignet.

Wahr

Bei SPI müssen neben den Datenbits sogenannte Synchronisationsbits übertragen werden.

Falsch

SPI ist eine synchrone Verbindung.

Wahr

Bei SPI braucht es zu jedem Slave eine separate Slave-Select-Leitung.

Wahr

# Frage 8

## I2C Adressierung

Der Master sendet zur Initialisierung der Kommunikation mit einem Slave folgende 8 Bit: 0100'0110 (MSB first). Interpretieren Sie diese:

Slave-Adresse

Read/Write

# Frage 8 Lösung

## I2C Adressierung

Der Master sendet zur Initialisierung der Kommunikation mit einem Slave folgende 8 Bit: 0100'0110 (MSB first). Interpretieren Sie diese:

Slave-Adresse

Read/Write



# Frage 9

## I2C Ende

Der Master signalisiert das **Ende** einer I2C-Kommunikation durch folgende Bedingung:

Flanke auf  während   ist.

# Frage 9 Lösung

## I2C Ende

Der Master signalisiert das **Ende** einer I2C-Kommunikation durch folgende Bedingung:

Steigende

Flanke auf

SDA

während

SCL

High

ist.

# Frage 10

## Timer/Counter

Gegeben sei ein 16-bit Counter, der mit einem 40 MHz Clock-Signal verbunden ist. Mit Hilfe dieses Counters soll nun alle 20 ms ein Interrupt ausgelöst werden. Durch welchen Wert aus der Auswahlliste muss der Prescaler die Clock-Frequenz teilen, damit der Wertebereich des Auto-Reload-Registers (ARR) möglichst gut ausgeschöpft wird?

# Frage 10 Lösung

## Timer/Counter

Gegeben sei ein 16-bit Counter, der mit einem 40 MHz Clock-Signal verbunden ist. Mit Hilfe dieses Counters soll nun alle 20 ms ein Interrupt ausgelöst werden. Durch welchen Wert aus der Auswahlliste muss der Prescaler die Clock-Frequenz teilen, damit der Wertebereich des Auto-Reload-Registers (ARR) möglichst gut ausgeschöpft wird?

1 Sekunde / 40 MHz = 0.000000025 Sekunden/Cycle

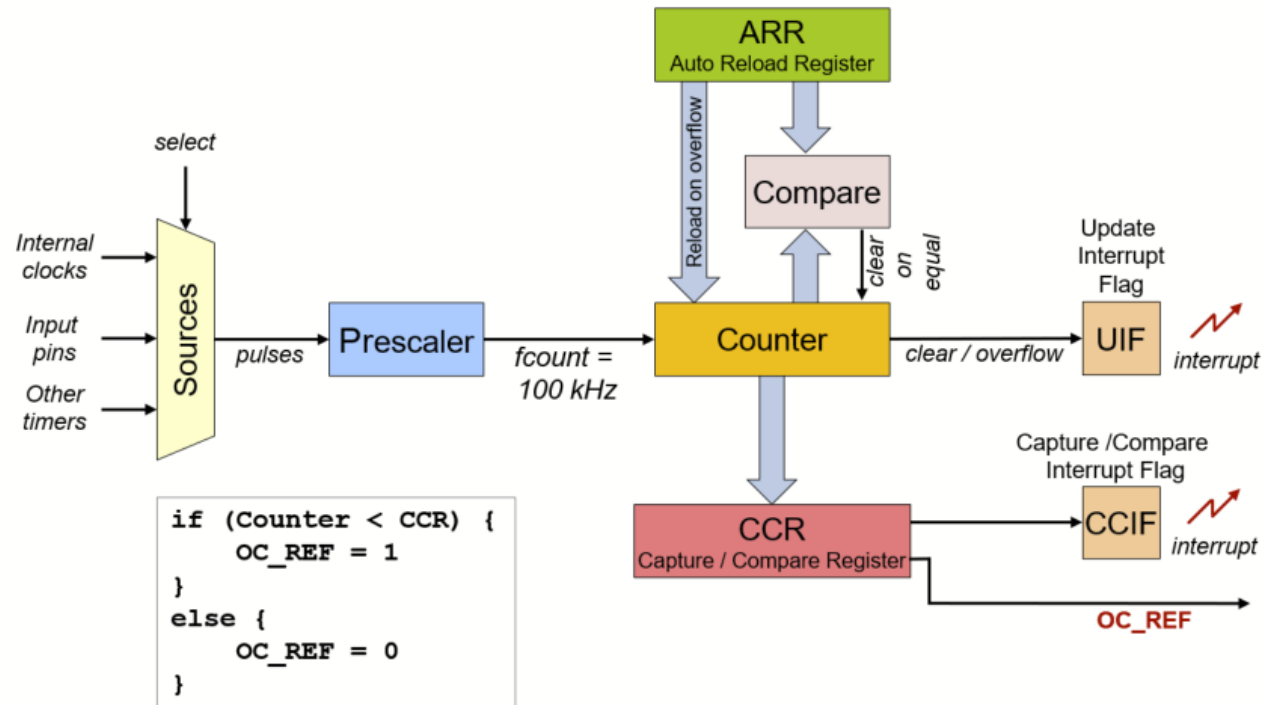
Bei  $2^{16}$  Counter =  $0.000000025 * 2^{16} = 0.0016384$  Sekunden = Alle 1.6 ms ein Interrupt

Für 20 ms Clock mindestens durch  $\sim 13$  Teilen

# Frage 11

## PWM - Periode und Duty Cycle

Gegeben ist der folgende Counter. Dieser ist als Up-counter konfiguriert und inkrementiert mit einer Frequenz  $f_{\text{count}} = 100 \text{ kHz}$ . Alle Register sind 16-bit breit.



Mit welchen Werten müssen das ARR Register und das CCR Register programmiert werden, um am Ausgang OC\_REF ein PWM Signal mit einer Periode von 600 ms und einem Duty Cycle von 80% zu generieren?

Geben Sie die Werte als Dezimalzahlen an. Die Herleitung muss ersichtlich sein.

# Frage 11 Lösung

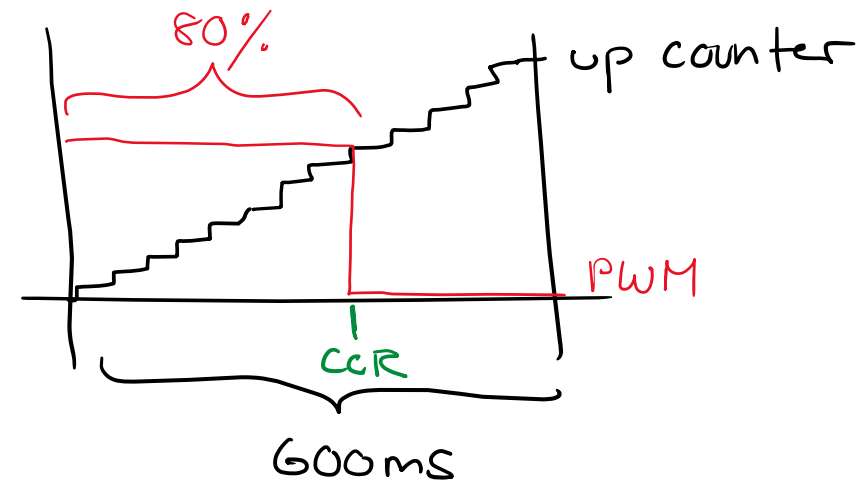
PWM 80% Duty, 600 ms Periode

Periode fcount =  $1s / 100kHz = 0.00001 s$

Ziel Periode = 600 ms = 0.6s

fcount x 60'000 = 0.6s -> ARR = (60'000 - 1)

80% von 60'000 = 48'000 -> CCR = (48'000 - 1)



# Frage 12

## Timer/Counter

Gegeben ist ein universeller 16-Bit Timer mit Capture / Compare - Einheit. Er ist wie folgt konfiguriert:

- Alle Register sind 16 Bit breit.
- Die Quelle liefert ein Signal der Frequenz 35 MHz.
- Der Prescaler ist so eingestellt, dass jeder 70. Tick gezählt wird.
- Der Timer arbeitet als Downcounter.

Für das PWM-Signal gelten die folgenden Einstellungen:

- Das PWM Signal wird low gesetzt, wenn der Counter 0 erreicht.
- Das PWM Signal wird high gesetzt, wenn der Counter den Compare-Wert erreicht.

Es soll nun ein PWM-Signal mit einer Periode von 96 ms erzeugt werden. Der Duty Cycle soll 6/8 betragen.

Bestimmen Sie den Zahlenwert (dezimal), der im Capture-Compare-Register (**CCR**) stehen muss.

Antwort:

# Frage 12 Lösung

## Timer/Counter

Gegeben ist ein universeller 16-Bit Timer mit Capture / Compare - Einheit. Er ist wie folgt konfiguriert:

- Alle Register sind 16 Bit breit.
- Die Quelle liefert ein Signal der Frequenz 35 MHz.
- Der Prescaler ist so eingestellt, dass jeder 70. Tick gezählt wird.
- Der Timer arbeitet als Downcounter.

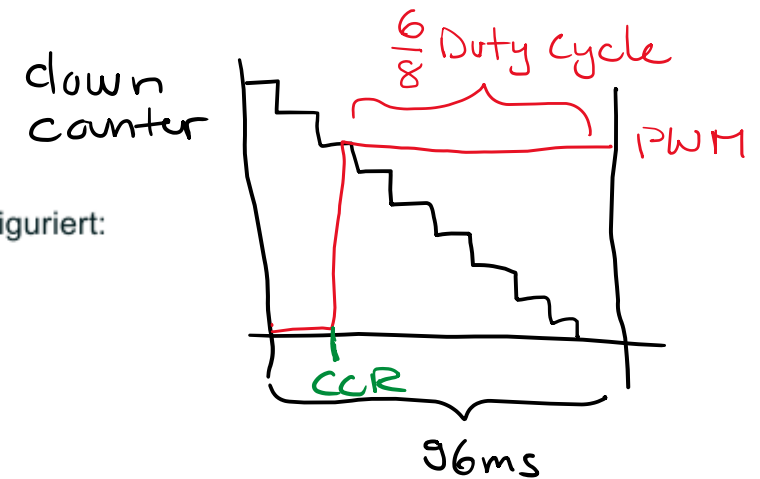
Für das PWM-Signal gelten die folgenden Einstellungen:

- Das PWM Signal wird low gesetzt, wenn der Counter 0 erreicht.
- Das PWM Signal wird high gesetzt, wenn der Counter den Compare-Wert erreicht.

Es soll nun ein PWM-Signal mit einer Periode von 96 ms erzeugt werden. Der Duty Cycle soll  $\frac{6}{8}$  betragen.

Bestimmen Sie den Zahlenwert (dezimal), der im Capture-Compare-Register (**CCR**) stehen muss.

Antwort: 36'000 -1



35MHz  
→ Prescaler  $\frac{1}{70}$  → 500kHz  
Timer

Tick Times  $1/500\text{kHz}$   
 $= 0,000002\text{s}$

Wenn Timer also auf 96ms  
eingestellt ist muss

ARR = (48'000 - 1) sein

weil  $0,000002 \times 48'000 = 96\text{ms}$

$\frac{6}{8}$  von ARR → CCR = (36'000 - 1)



# Frage 13

## ADC Offset (ADC Folien)

Der ADC1 des STM32F429xx-Mikrocontrollers wird mit den folgenden Eigenschaften verwendet:

- Vref des ADC ist 3 V
- der Offsetfehler des ADC ist +2 LSB
- der ADC verwendet 8-Bit

Wie lautet die absolute Adresse des Registers, in dem die Wandlungsergebnisse gelesen werden können (in Hex) (3 P)?

Welche Spannung entspricht dem Offsetfehler (Ergebnis in Millivolt auf 1 Dezimalstelle. Die Einheit nicht schreiben) (5 P)

mv

# Frage 13 Lösung

## ADC Offset (ADC Folien)

Der **ADC1** des STM32F429xx-Mikrocontrollers wird mit den folgenden Eigenschaften verwendet:

- Vref des ADC ist **3 V**
- der Offsetfehler des ADC ist +2 LSB
- der ADC verwendet 8-Bit

$$3V \rightarrow 1 \text{ LSB} = \frac{V_{REF}}{2^N} [V] = \frac{3V}{2^8} = 0,01171875$$

$$\text{offset error} = 2 \cdot 0,01171875 = 23,4 \text{ mV}$$

Wie lautet die absolute Adresse des Registers, in dem die Wandlungsergebnisse gelesen werden können (in Hex) (3 P)?

0x4001'204C

Welche Spannung entspricht dem Offsetfehler (Ergebnis in Millivolt auf 1 Dezimalstelle. Die Einheit nicht schreiben) (5 P)

23.4

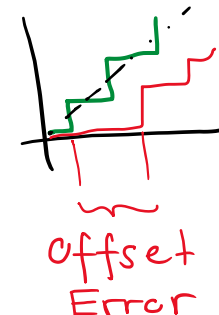
mv

①

0x4001 2000 - 0x4001 23FF	ADC1 - ADC2 - ADC3
---------------------------	--------------------

Section 13.13.18: ADC register map on page 430

②



### 13.13.14 ADC regular data register (ADC\_DR)

Address offset: 0x4C

Reset value: 0x0000 0000

# Frage 14

## Memory

Wie gross ist der Speicherbereich in **kBytes**, der mit **22** Adressleitungen maximal angesprochen werden kann, wenn jede Adresse ein individuelles Byte identifiziert?

Antwort:

# Frage 14 Lösung

## Memory

Wie gross ist der Speicherbereich in **kBytes**, der mit **22** Adressleitungen maximal angesprochen werden kann, wenn jede Adresse ein individuelles Byte identifiziert?

Antwort:

$$2^{22} = 4'194'304 / 1024 = 4096 \text{ kBytes}$$

# Frage 15

## Memory

In einem Flash Baustein wird ein Byte mit den unten angegebenen Operationen modifiziert. Was ist der resultierende Wert?

Ausgangswert	Flash Operationen	Resultierender Wert
0xC3	1. <b>kein Erase</b> 2. Program 0xF3	<input type="text"/>
0xC3	1. Erase 2. Program 0xF3	<input type="text"/>

# Frage 15 Lösung

## Memory

In einem Flash Baustein wird ein Byte mit den unten angegebenen Operationen modifiziert. Was ist der resultierende Wert?

Ausgangswert	Flash Operationen	Resultierender Wert
0xC3	1. <b>kein Erase</b> 2. Program 0xF3	0xC3
0xC3	1. Erase 2. Program 0xF3	0xF3

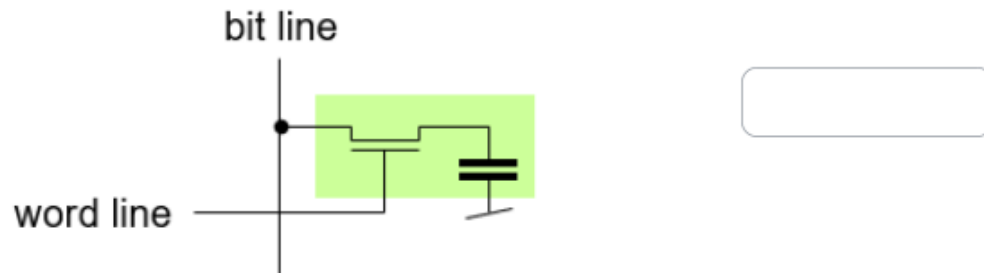
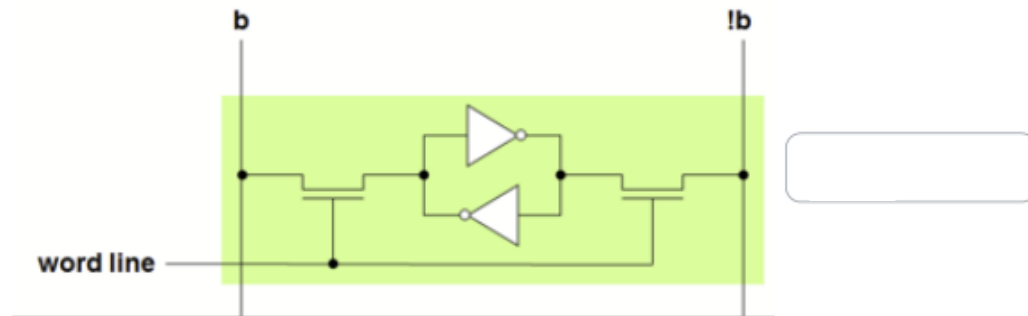
```
0xC3    11000011
0xF3    11110011
-----
0xC3    11000011
```

```
0xC3    11000011
Erase   00000000
0xF3    11110011
-----
0xF3    11110011
```

# Frage 16

## Memory

Weisen Sie die Speicherzellen der entsprechenden Speichertechnologie zu.



# Frage 16 Lösung

## Memory

Weisen Sie die Speicherzellen der entsprechenden Speichertechnologie zu.

