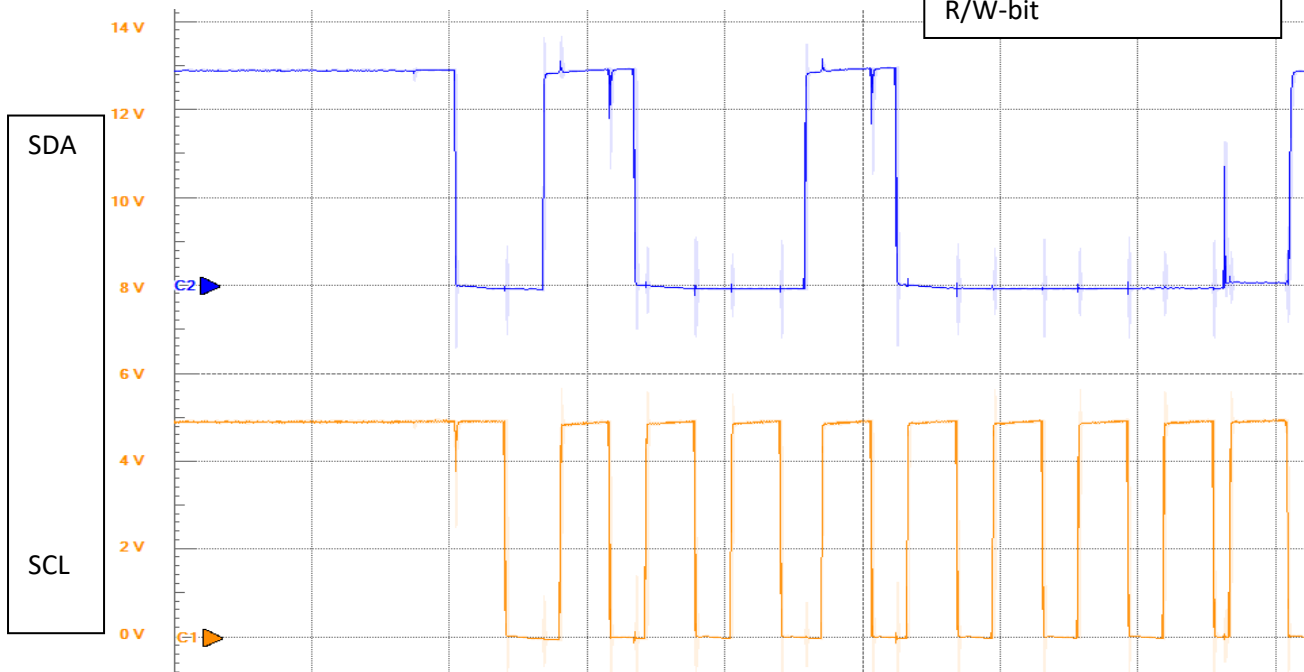


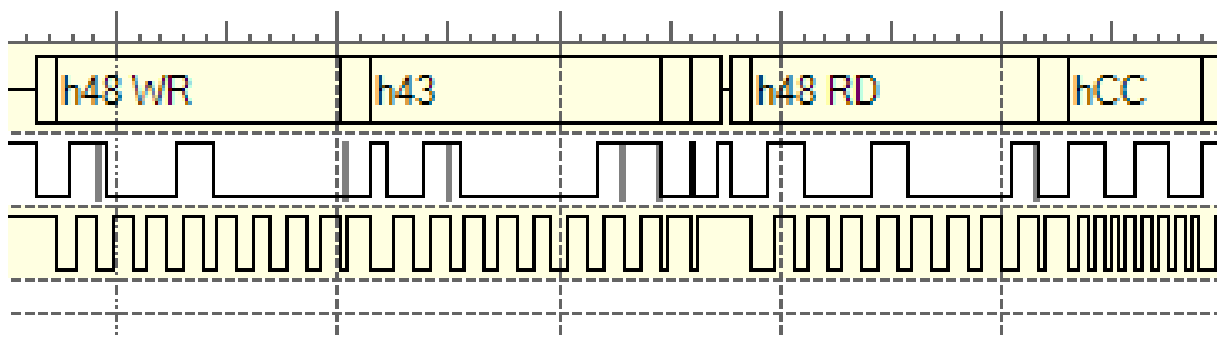
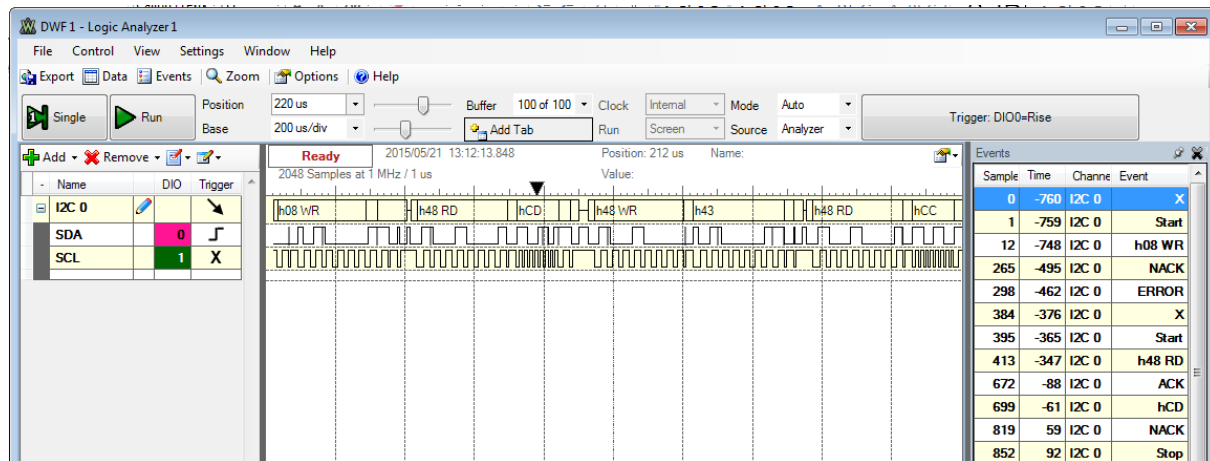
### Messung am IIC-Bus mit dem USB Messadapter „Analog Discovery“

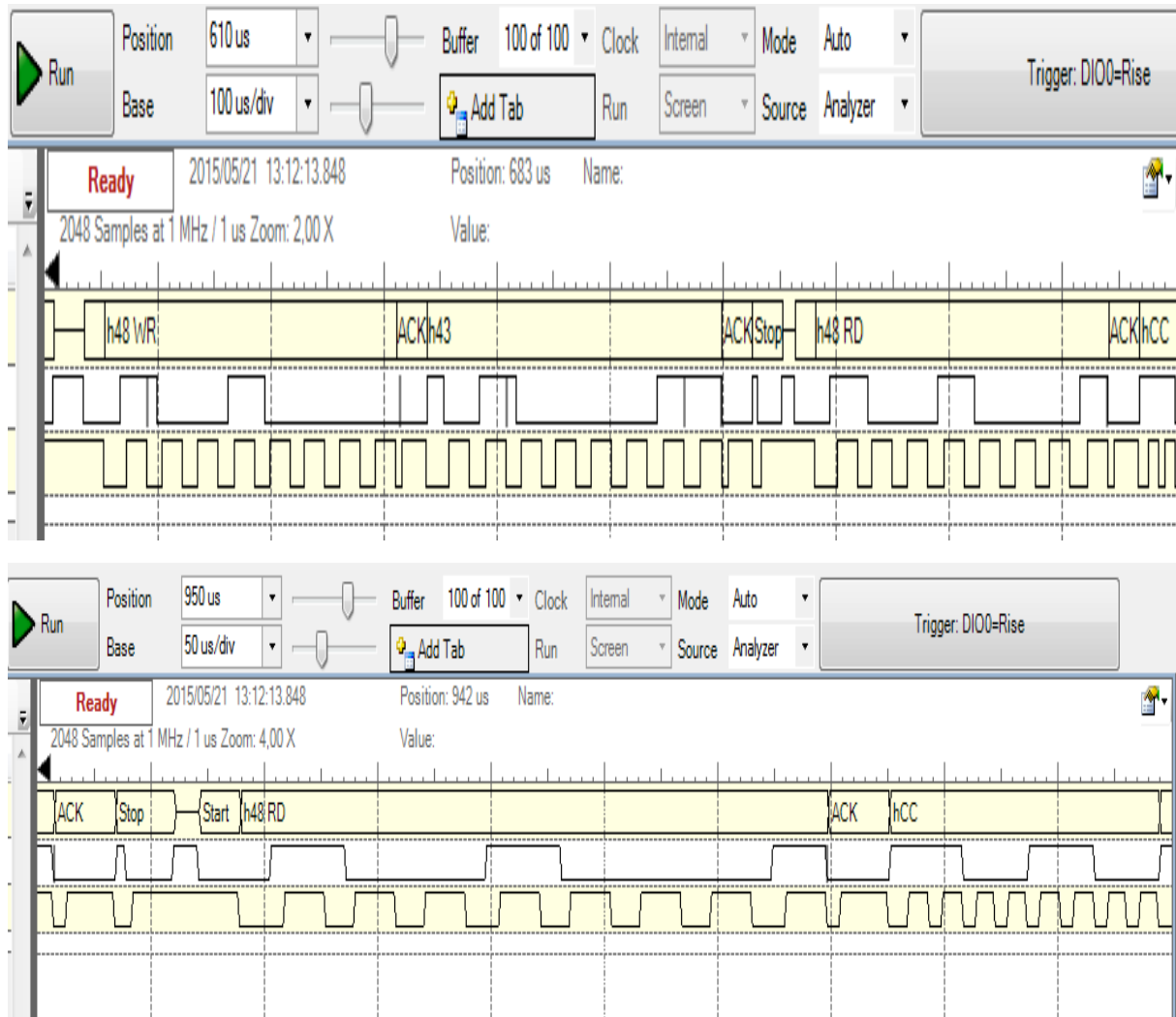
E3GS Do 21.Mai 2015 Messprotokoll  
 Kommunikation uC mit IC AD/DAWandler PCF8591  
 Oszilloskopmessung Zeitachse: 100µsec/Div

Zeichnen sie ein:  
 Startbedingung,  
 Adressbyte ausgewertet  
 R/W-bit



### Messprotokolle (Screenshots) vom Logicanalyzer:





### Fragen:

1. Im  $\mu$ Controllerprogramm (siehe unten), wird als Adresse schreib eine hex 90 angegeben, der Analyser meldet eine hex 48 WR. Wie ist das zu erklären?
2. Im  $\mu$ Controllerprogramm (siehe unten), wird als Adresse lies eine hex 91 angegeben, der Analyser meldet eine hex 48 RD. Wie ist das zu erklären?
3. Mit welcher Frequenz arbeitet SCL bei der Ausgabe einer Adresse?
4. Mit welcher Frequenz arbeitet SCL beim Einlesen eines Wertes vom Slave?
5. Wie kommt es zu diesen zwei Geschwindigkeiten? Erklärungshilfe: C-Programm des Controllers betrachten!
6. Versuchen sie den C-Quellcode und die Messung zu verstehen.
  - Welcher Kanal am AD Wandler wird benutzt? ->Steuerwortregister im Datenblatt PCF8591.
  - Welcher Messwert wird vom AD Wandler erfasst? ->Messprotokoll
  - Welcher Spannung entspricht das?  $U_{ref} = 2,56$  Volt

**Messprotokoll mit der Software vom Logicanalyser erstellt:**

Sample	Time	Channel	Event
0	-760	I2C 0	X
1	-759	I2C 0	Start
12	-748	I2C 0	h08 WR
265	-495	I2C 0	NACK
298	-462	I2C 0	ERROR
384	-376	I2C 0	X
395	-365	I2C 0	Start
413	-347	I2C 0	h48 RD
672	-88	I2C 0	ACK
699	-61	I2C 0	hCD
819	59	I2C 0	NACK
852	92	I2C 0	Stop
878	118	I2C 0	X
905	145	I2C 0	Start
923	163	I2C 0	h48 WR
1180	420	I2C 0	ACK
1207	447	I2C 0	h43
<b>1468</b>	<b>708</b>	<b>I2C 0</b>	<b>ACK</b>
<b>1495</b>	<b>735</b>	<b>I2C 0</b>	<b>Stop</b>
<b>1521</b>	<b>761</b>	<b>I2C 0</b>	<b>X</b>
<b>1532</b>	<b>772</b>	<b>I2C 0</b>	<b>Start</b>
<b>1550</b>	<b>790</b>	<b>I2C 0</b>	<b>h48 RD</b>
<b>1809</b>	<b>1049</b>	<b>I2C 0</b>	<b>ACK</b>
<b>1836</b>	<b>1076</b>	<b>I2C 0</b>	<b>hCC</b>
<b>1955</b>	<b>1195</b>	<b>I2C 0</b>	<b>NACK</b>
1988	1228	I2C 0	Stop
2014	1254	I2C 0	X

**Controllerprogramm, auszugsweise. Dieses Programm erzeugt mit Hilfe eines Treibers die IIC-Signale, die oben dargestellt gemessen wurden.**

```

24
25 unsigned char   adr_lies   = 0x91; // 10010001->100
26 unsigned char   adr_schreib = 0x90; // 10010000 ->100
27 unsigned char   contr_ain   = 0x40; // 01000000 ->Kon
28 unsigned char   kanal = 3;      //Auswahl Kanal 3
29 unsigned char   analogwert;
30 // *****
31 main()
32 {
33     P1 = 0;
34     while(1)
35     {
36         analogwert = ana_ein(kanal); // Wert vom AD_Wandl
37         P1 = analogwert;             // Am Port1 wird Erge.
38
39     }
40 }
41 unsigned char ana_ein(unsigned char adkanal)
42 {
43     unsigned char a_wert;
44     start_i2c(); // Startbedi.
45     byte_out(adr_schreib); // Adresse d
46     byte_out( (adkanal&0x03) | contr_ain); // Contr
47     stop_i2c(); // Stoppbedi.
48     start_i2c(); // Startbedi.
49     byte_out (adr_lies); // Adresse d
50     a_wert = byte_in (); // Wert a
51     nak_out(); //kein Ackno
52     stop_i2c ();
53     return(a_wert);|
54 }
55

```

Fragen: Hinweis: Die Funktionen start\_i2c() stop\_i2c() byte\_out(xx) xx byte\_in() nak\_out() sind in einer Treiberdatei hinterlegt.

1. Was macht das Programm? Erklärung Zeile für Zeile, dann die Gesamtfunktion beschreiben.
2. Erklären sie die Zeile 46 indem sie die einzelnen bits beschreiben