

eigerPanel57C ● eigerPanel57H

Hardware-Manual



eigerGraphics – The smart Touchpanel Solutions
eine Produktlinie der S-TEC electronics AG, Zug, Schweiz.

eigerPanel – das vielseitige Bedienpanel und Steuerungsgerät mit farbigem Display mit Touchscreen

- I/O-Anschlüsse digital (Zähler-Eingang, Funktionstasten, Digital- und PWM-Ausgang)
- I/O-Anschlüsse analog (Spannungs- und Analog-Eingänge, Analog-Ausgänge)
- Audio-Ausgang, Stereo
- serielle Schnittstellen RS485, RS232, I²C
- Anschluss für TFT VGA 640x480 LCD
- Compact Flash Card als Datenträger
- Hardware-Uhr batteriegestützt
- Eingangsspannung 9..30VDC
- Leistung: 4.4W typisch
- Bootzeit ab Kaltstart: 4 Sekunden
- Taktfrequenz: 50 MHz

Das eigerPanel gibt es zu verschiedenen Displaygrößen, in erster Linie aber für 5.7“ VGA und 7“ WVGA TFT Displays mit LED Backlight. Ausführliche Infos erhalten Sie auf der Homepage des eigerPanels: www.eigergraphics.com.

Inhaltsverzeichnis

Hardware/Aufbau-Übersicht	2
Pinbelegung.....	3
Spezifikationen	15
Abmessungen.....	17
Unterschiede F57C zu F57H	18
Abmessungen von Rahmen und Gehäuse (optional).....	19
Firm- und Software.....	19
Kontakt / Information	19
Datenblatt Version.....	19
Notizen	21

Hardware/Aufbau-Übersicht

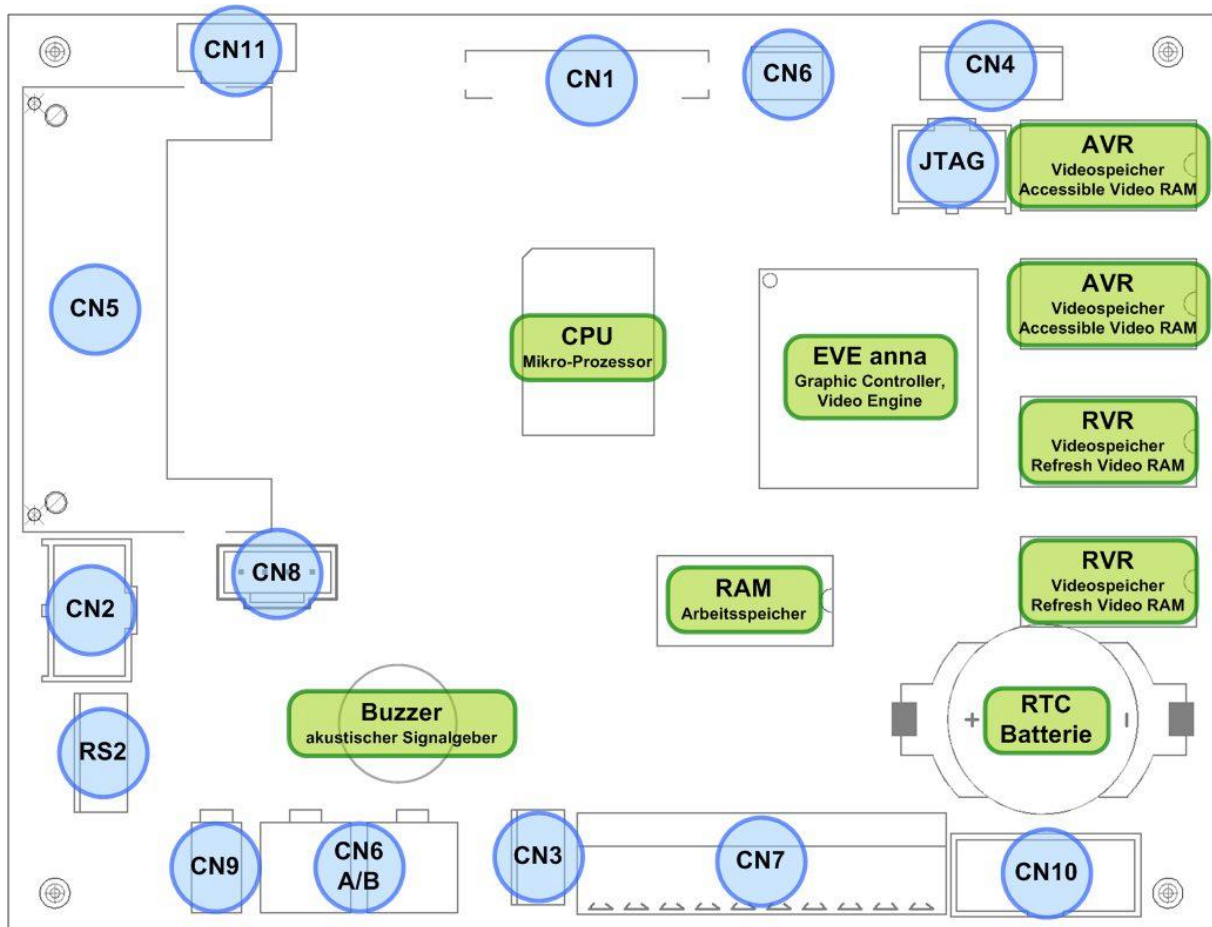


Abbildung 1: Stecker-Übersicht F57C_v0944 (identisch zu F57H_V0944)

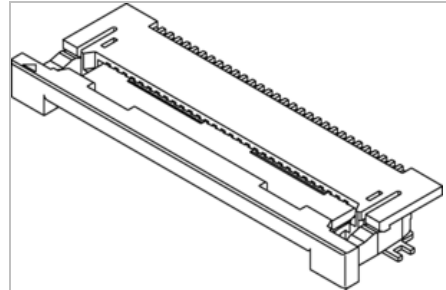
CNx	Bezeichnung	Link	Seite
CN1	Anschluss Display TFT VGA	»»	3
CN2	FOX-COM1 (UART1) Programmier- und Debug-Schnittstelle RS232	»»	4
CN3	Analog Ausgänge Sound DA	»»	5
CN4	Analog Eingänge NTC, Potentiometer	»»	6
CN5	Schnittstelle für Compact Flash Card	»»	7
CN6	Analog Eingang Potentiometer	»»	8
CN6A/B	BUS (serielle Schnittstelle RS485)	»»	8
CN7	Externe Ein- / Ausgänge	»»	9
CN8	BUS (serielle Schnittstelle I ² C)	»»	10
CN9	Power Supply VDC	»»	10
CN10	Digitale Eingänge für 8 Funktionstasten (Keyboard)	»»	11
CN11	Anschluss externer Touchscreen (optional)	»»	12
JTAG	Programmier-Schnittstelle für EVE anna (CPLD)	»»	12
RS2	FOX-COM2 (UART2) serielle Schnittstelle RS232	»»	13
-	Buzzer (akustischer Signalgeber)	»»	13

Pinbelegung

CN1 Anschluss Display TFT VGA mit Backlight und Touch

Steckertyp Molex 54132, 40-Pol (0.5mm, bottom Contact)

Pin 1	VD	
Pin 2	L3V3	Logic +3.3V
Pin 3	HSYNC	Horizontal synchronization
Pin 4	L3V3	Logic +3.3V
Pin 5	L3V3	Logic +3.3V
Pin 6	L3V3	Logic +3.3V
Pin 7	L3V3	Logic +3.3V
Pin 8	VSYNC	Vertical synchronization
Pin 9	DE	Data enable
Pin 10	TP_Y-	Touchpanel Y-direction -
Pin 11	TP_X+	Touchpanel X-direction +
Pin 12	ADJ	Adjust
Pin 13	B5	Blue bit 5
Pin 14	B4	Blue bit 4
Pin 15	B3	Blue bit 3
Pin 16	LGND	Logic Ground
Pin 17	B2	Blue bit 2
Pin 18	B1	Blue bit 1
Pin 19	LGND	Logic Ground
Pin 20	LGND	Logic Ground
Pin 21	G5	Green bit 5
Pin 22	G4	Green bit 4
Pin 23	G3	Green bit 3
Pin 24	LGND	Logic Ground
Pin 25	G2	Green bit 2
Pin 26	G1	Green bit 1
Pin 27	LGND	Logic Ground
Pin 28	LGND	Logic Ground
Pin 29	R5	Red bit 5
Pin 30	R4	Red bit 4
Pin 31	R3	Red bit 3
Pin 32	LGND	Logic Ground
Pin 33	R2	Red bit 2
Pin 34	R1	Red bit 1
Pin 35	LGND	Logic Ground
Pin 36	TP_Y+	Touchpanel Y-direction +
Pin 37	TP_X-	Touchpanel X-direction -
Pin 38	PCLK	Processor Clock
Pin 39	LGND	Logic Ground
Pin 40	LR	Left/Right



passendes Kabel: F4396 (76mm)

auf Anfrage: F4375 (85mm)

eigerScript EVM Command-Class: **Display**

```

Display.Clear( )
Display.ClearColor( VarInt:Color )
Display.CopyLineDown( )
Display.CopyLineRight( )
Display.CopyWindow( )
Display.Direct( )
Display.Prepare( )
Display.RestoreWriteMode( )
Display.Show( )
Display.ShowLineDown( )
Display.ShowLineRight( )
Display.ShowTranslateWindow( )
Display.ShowWindow( )
Display.ShowWindowEffect( VarInt:ShowWindowEffect , VarInt:Speed )

```

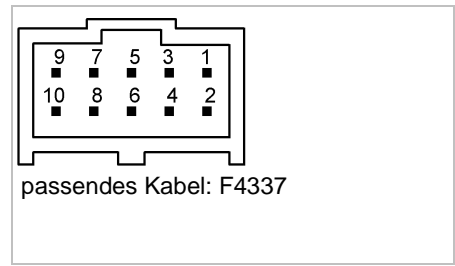
Bei diesen Commands handelt es sich um einen unvollständigen Auszug aus der Command-Class **Display**.

[< Inhaltsverzeichnis](#)

CN2 S-PROG20 Download / Debug (FOX-COM1, UART1)

Steckertyp Molex C-Grid III 10-Pol Wiring Harness: Molex C-Grid III Gehäuse 10-Pol

Pin 1	/HOLD	/Hold Line
Pin 2	LGND	Logic Ground
Pin 3	BYTE	Byte Line
Pin 4	L12V	Logic +12V
Pin 5	CNVSS	Cnvss Line
Pin 6	L3V3	Logic +3.3V
Pin 7	/RESET	/Reset Line
Pin 8	LGND	Logic Ground
Pin 9	TXD_1	S-PROG20 Transmit Line
Pin 10	RXD_1	S-PROG20 Receive Line

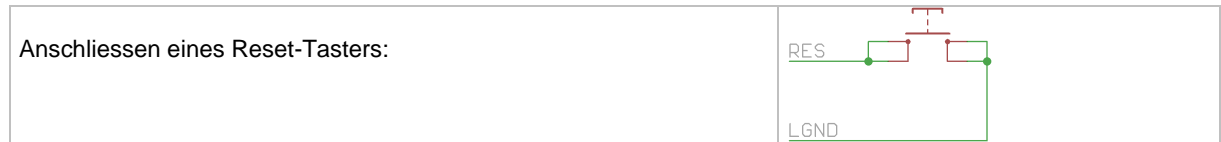


eigerScript eVM Command-Class: **Debug**

```

Debug.Mark( Str )
Debug.DumpBytes( VarInt )
Debug.Print_IntegerHex( Str , VarInt )
Debug.Print_LongHex( Str , VarLong )
Debug.Print_SingleHex( Str , VarSingle )
Debug.Print_Char( VarInt )
Debug.Print_CLRf( )
Debug.Print_String( VarStr )
Debug.Print_StringHeader( VarStr )
    
```

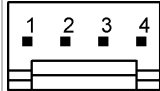
Schaltungsbeispiele



< [Inhaltsverzeichnis](#)

CN3 Analog Ausgänge für Sound DA

Steckertyp	TYCO CST100 4-Pol	Wiring Harness: AMP MTA100 4-Pol
Pin 1	OP93	Analog Out / Sound DA Out
Pin 2	LGND	Logic Ground
Pin 3	OP94	Analog Out / Sound DA Out
Pin 4	L5V	Logic +5V



passendes Kabel: F4388 , F4389

eigerScript	eVM Command-Class: Sound
<code>Sound.SetSamplingRate(VarInt:SamplingRate_11025)</code>	
<code>Sound.SetBitsPerSample(VarInt: BitsPerSample)</code>	
<code>Sound.PlaySamples(VarInt:Sound_MODE_STEREO , VarStr:File , VarLong:StartOffset , VarLong:NoOfSamples , VarInt:Loops)</code>	
<code>Sound.InstallLocal(LabelRelative24:Event)</code>	

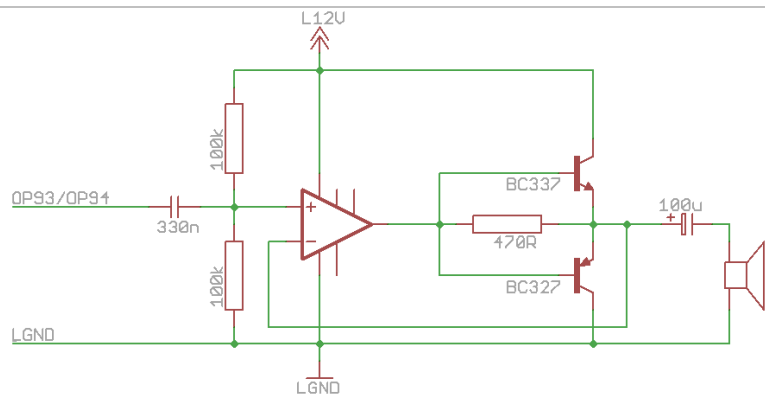
eigerScript	eVM Command-Class: File
<code>File.Read_ToBuffer(VarInt:Handle.I , VarStr:Buffer , VarLong:Offset , VarLong:NumBytes , VarLong:BytesRead)</code>	

eigerScript	eVM Command-Class: Buffer
<code>Buffer.Get_UINT8_I(VarInt:UINT8 , VarStr:Buffer , VarLong:Offset)</code>	
<code>Buffer.Get_IN8_I(VarInt:INT8 , VarStr:Buffer , VarLong:Offset)</code>	
<code>Buffer.Get_INT16(VarInt:INT16 , VarStr:Buffer , VarLong:Offset)</code>	
<code>Buffer.Get_UINT16(VarInt:UINT16 , VarStr:Buffer , VarLong:Offset)</code>	
<code>Buffer.Get_INT32(VarInt:INT32 , VarStr:Buffer , VarLong:Offset)</code>	
<code>Buffer.Put_UINT8_I(VarInt:UINT8 , VarStr:Buffer , VarLong:Offset)</code>	

Schaltungsbeispiele

Verwenden des Audio-Ausgang durch
Zuschalten eines einfachen Audio-
Verstärkers:

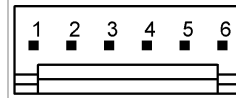
*(Funktionalität des abgebildeten
Verstärkers wird nicht gewährleistet)*



< [Inhaltsverzeichnis](#)

CN4 Analog Eingänge für NTC, Potentiometer

Steckertyp	TYCO CST100 6-Pol	Wiring Harness: AMP MTA100 6-Pol
Pin 1	L3V3	Logic +3.3V
Pin 2	PA2	Analog In
Pin 3	L3V3	Logic +3.3V
Pin 4	PA3	Analog In
Pin 5	LGND	Logic Ground
Pin 6	LGND	Logic Ground



passendes Kabel: F4340

eigerScript	eVM Command-Class: InOut
<code>InOut.Read_ADC(VarInt:Kanal_PA2_PA3 , VarInt:Value_0-1023)</code>	

Schaltungsbeispiele

Einsatz eines Potentiometers unter Verwendung einer der Analog-Eingänge PA2 / PA3:	
Einsatz eines NTC (z.B. als Temperatur-Sensor) unter Verwendung einer der Analog-Eingänge PA2 / PA3:	

Durch Entfernen der Widerstände R59 & R62 (siehe Bild unten) werden die Analog-Eingänge PA2 & PA3 zu Spannungsmesseingänge. Der zulässige Spannungsbereich liegt bei 0..10VDC.

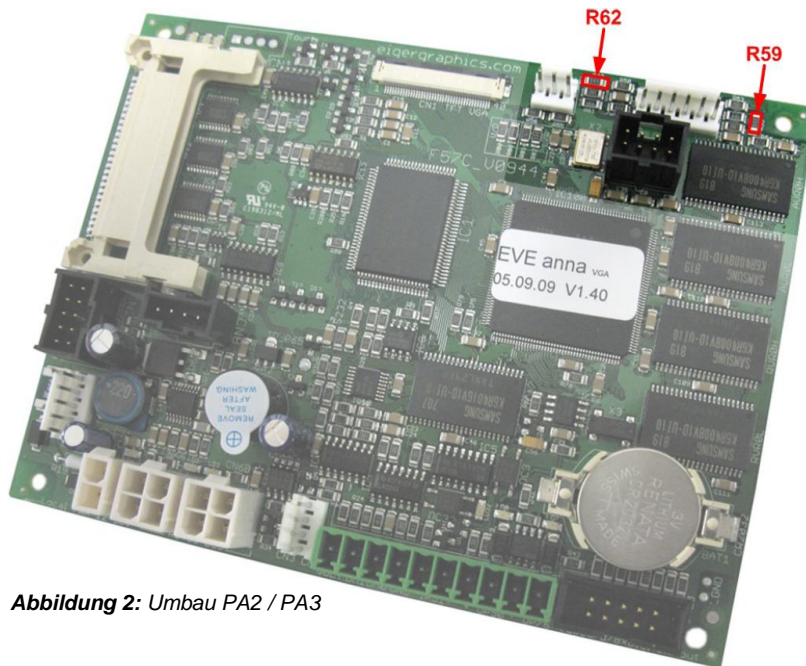


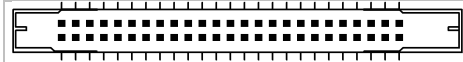
Abbildung 2: Umbau PA2 / PA3

[< Inhaltsverzeichnis](#)

CN5 Compact Flash Card / Massenspeicher

Steckertyp TCO 752DF50NC00001 , CFA Type 1 , 50-polig

Pin 1	LGND	Logic Ground
Pin 2	CD3	
Pin 3	CD4	
Pin 4	CD5	
Pin 5	CD6	
Pin 6	CD7	
Pin 7	C/CE2.2	
Pin 8	LGND	Logic Ground
Pin 9	LGND	Logic Ground
Pin 10	LGND	Logic Ground
Pin 11	LGND	Logic Ground
Pin 12	LGND	Logic Ground
Pin 13	CF+3.3V	Logic +3.3V
Pin 14	LGND	Logic Ground
Pin 15	LGND	Logic Ground
Pin 16	LGND	Logic Ground
Pin 17	LGND	Logic Ground
Pin 18	CA3	
Pin 19	CA2	
Pin 20	CA1	
Pin 21	CD0	
Pin 22	CD1	
Pin 23	CD2	
Pin 24	IOCS16	
Pin 25	/CD2	
Pin 26	/CD1	
Pin 27	CD11	
Pin 28	CD12	
Pin 29	CD13	
Pin 30	CD14	
Pin 31	CD15	
Pin 32	C/CE2.3	
Pin 33	VS1	
Pin 34	C/RD	
Pin 35	C/WR	
Pin 36	CF+3.3V	Logic +3.3V
Pin 37	INTRQ	
Pin 38	CF+3.3V	Logic +3.3V
Pin 39	/CSEL	
Pin 40	VS2	
Pin 41	CFRES	
Pin 42	/IORDY	
Pin 43	INPACK	
Pin 44	CF+3.3V	Logic +3.3V
Pin 45	/DASP	
Pin 46	PDIAG	
Pin 47	CD8	
Pin 48	CD9	
Pin 49	CD10	
Pin 50	LGND	Logic Ground
Pin 51	LGND	Logic Ground
Pin 52	LGND	Logic Ground

eigerScript eVM Command-Class: **File**

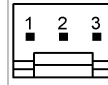
```
File.AppendString( VarStr:FileName , VarStr:String )
File.DeleteFile( VarStr:FileName )
File.Read_CSV( VarStr:FileName , VarStr:Buffer )
File.Read_EGI( VarStr:FileName )
File.Read_TextFile( VarStr:FileName , VarStr:Buffer )
File.ScreenDump( VarStr:FileName )
```

Bei diesen Commands handelt es sich um einen unvollständigen Auszug aus der Command-Class **File**.[< Inhaltsverzeichnis](#)

CN6 Analog Eingang für Potentiometer

Steckertyp	TYCO CST100 3-Pol	Wiring Harness: AMP MTA100 3-Pol
------------	-------------------	----------------------------------

Pin 1	L3V3	Logic +3.3V
Pin 2	PA0	Analog IN
Pin 3	LGND	Logic Ground



passendes Kabel: F4336

eigerScript	eVM Command-Class: InOut
<code>InOut.Read_ADC(VarInt:Kanal_PA0 , VarInt:Value_0-1023)</code>	

Schaltungsbeispiele

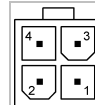
Einsatz eines Potentiometers unter Verwendung des Analog-Eingangs PA0:	
------------------------------------------------------------------------	--

Einsatz eines NTC (z.B. als Temperatur-Sensor) unter Verwendung des Analog-Eingangs PA0:	
------------------------------------------------------------------------------------------	--

< [Inhaltsverzeichnis](#)**CN6A / CN6B BUS, serielle Schnittstelle RS485**

Steckertyp	Molex MiniFit jr. 5566 4-Pol	Wiring Harness: Molex MiniFit jr. 5566 Gehäuse 4-Pol
------------	------------------------------	------------------------------------------------------

Pin 1	L12V	Logic +12V
Pin 2	RXTX-	RS485 Line minus
Pin 3	LGND	Logic Ground
Pin 4	RXTX+	RS485 Line plus



passendes Kabel: F4368

eigerScript	eVM Command-Class: DataServer
<code>DataServer.ReceiveToFile(VarInt:NodeAdress , VarStr:FileName)</code>	
<code>DataServer.ReceiveToString(VarInt:NodeAdress , VarStr:ZielString)</code>	
<code>DataServer.Rx_ReadInteger(VarInt:NodeAdress , VarInt:Register , VarInt:Value)</code>	
<code>DataServer.Rx_ReadLong(VarInt:NodeAdress , VarInt:Register , VarLong:Value)</code>	
<code>DataServer.Rx_String(VarInt:NodeAdress , VarInt:Register , VarLong:Value)</code>	
<code>DataServer.SendFromFile(VarInt:NodeAdress , VarStr:FileName)</code>	
<code>DataServer.SendFromString(VarInt:NodeAdress , VarStr:QuellString)</code>	
<code>DataServer.Tx_Command(VarInt:NodeAdress , VarInt:Register , VarInt:Command)</code>	
<code>DataServer.Tx_String(VarInt:NodeAdress , VarInt:Register , VarStr:String)</code>	
<code>DataServer.Tx_WriteInteger(VarInt:NodeAdress , VarInt:Register , VarInt:Value)</code>	
<code>DataServer.Tx_WriteLong(VarInt:NodeAdress , VarInt:Register , VarLong:Value)</code>	

eigerScript	eVM Command-Class: Debug
<code>Debug.Mark(Str)</code>	
<code>Debug.DumpBytes(VarInt)</code>	
<code>Debug.Print_IntegerHex(Str , VarInt)</code>	
<code>Debug.Print_LongHex(Str , VarLong)</code>	
<code>Debug.Print_SingleHex(Str , VarSingle)</code>	
<code>Debug.Print_Char(VarInt)</code>	
<code>Debug.Print_CLRf()</code>	
<code>Debug.Print_String(VarStr)</code>	
<code>Debug.Print_StringHeader(VarStr)</code>	

< [Inhaltsverzeichnis](#)

CN7 Externe Ein- / Ausgänge

Steckertyp	Phoenix MCS 3.81mm 10-Pol	Wiring Harness: PTR Buchsenleiste 3.81mm
Pin 1	OP76	Open collector (PWM) Pulsweite 0..1000
Pin 2	UVL	Supply plus
Pin 3	OP90	Open collector
Pin 4	UVL	Supply plus
Pin 5	IP91+	Input plus / AC
Pin 6	IP91-	Input minus / AC
Pin 7	LGND	Logic Ground
Pin 8	DA0	0 .. 255 → 0 .. 10VDC
Pin 9	DA1	0 .. 255 → 0 .. 10VDC
Pin 10	VI	0 .. 50V → 0 .. 1023


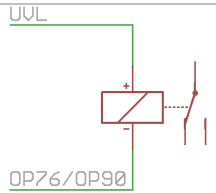
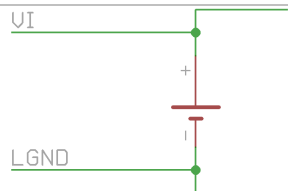
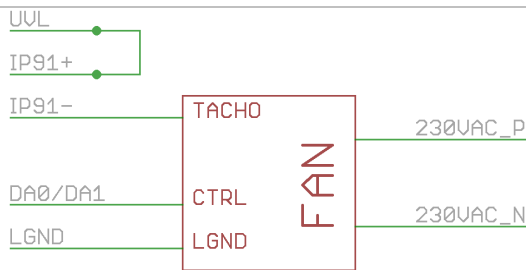
10	9	8	7	6	5	4	3	2	1
○	○	○	○	○	○	○	○	○	□

passendes Kabel: F4405

```

eigerScript eVM Command-Class: InOut
InOut.Read_ADC( VarInt:Kanal , VarInt:ADC-Value0-1023 )
InOut.DigitalOutputDriver( VarInt.KanalOutput_OP90 , VarInt:Output_Function )
InOut.PWM_Out( VarInt:Output_OP72_OP76 , VarInt:Value_0-1000 ) 1)
InOut.DA_Out_8( VarInt:Output_OP93_OP94 , VarInt:Value_0-255 )
InOut.Read_Input( VarInt:Input_IP91 , VarInt:Value )
InOut.UpCounter_Get( VarInt:Input_IP91 , VarLong:Counter )
InOut.UpCounter_Reset( VarInt:Input_IP91 )
InOut.DownCounter_Get( VarInt:Input_IP91 , VarLong:Counter )
InOut.DownCounter_Load( VarInt:Input_IP91 , VarLong:Counter )
InOut.DownCounter_Zero_InstallLocal( labelRelative24:Event )
    
```

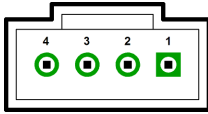
Schaltungsbeispiele

Verwendung des PWM-Ausgang OP76 am Beispiel einer LED-Beleuchtungseinheit:	
Ansteuerung eines Relais unter Verwendung des Digital-Ausgang OP90: (Zum Ansteuern eines Relais sollte nicht der PWM-Ausgang genutzt werden.)	
Messen einer Spannung mit dem Volt-Eingang: (z.B. Spannungsüberwachung an einer Quelle)	
Ansteuern und zurücklesen der Umdrehungszahl eines AC-Ventilators unter Verwendung der Analog-Ausgänge DA0 / DA1 und des Zähler-Eingangs IP91:	

[< Inhaltsverzeichnis](#)

1) Alternativ kann das Port-Register (el.DA_P76) direkt beschrieben werden ; Bereich: 0 .. 3000.


CN8 Anschluss I²C-Bus (Ext.)

Steckertyp		Molex C-Grid III 4-Pol (einreihig)	Wiring Harness: Molex SL Gehäuse 4-Pol
Pin 1	SDA	Serial data line	 <p>passendes Kabel: F4404</p>
Pin 2	L5V	Logic +5V	
Pin 3	LGND	Logic Ground	
Pin 4	SCL	Serial clock line	

eigerScript	eVM Command-Class: I2C
<code>I2C.Init()</code>	
<code>I2C.Start()</code>	
<code>I2C.RepeatedStart()</code>	
<code>I2C.Stop()</code>	
<code>I2C.WriteConfigByte(VarInt:SlaveAddress , VarInt:I2C_DATADIRECTION)</code>	
<code>I2C.WriteByte(VarInt:Value)</code>	
<code>I2C.ReadStreamByte(VarInt:Value)</code>	
<code>I2C.ReadLastByte(VarInt:Value)</code>	
<code>I2C.SetMode(VarInt.I2C_MODE)</code>	

[< Inhaltsverzeichnis](#)

CN9 Power Supply VDC

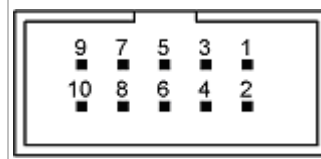
Steckertyp		Molex MiniFit jr. 5566 2-Pol	Wiring Harness: Molex MiniFit jr. 5566 Gehäuse 2-Pol
Pin 1	LVDC	Logic +8 .. 30VDC	 <p>appropriate power supply: F4325 (12V) oder F4394 (24V)</p>
Pin 2	LGND	Logic Ground	

[< Inhaltsverzeichnis](#)

CN10 Digital Eingänge bis zu 8 Funktionstasten (Keyboard)

Steckertyp: Box Header 10-Pol Wiring Harness: Flachbandkabelstecker 10pol

Pin 1	L3V3	Logic +3.3V
Pin 2	SW_A	Switch A
Pin 3	SW_B	Switch B
Pin 4	SW_C	Switch C
Pin 5	SW_D	Switch D
Pin 6	SW_E	Switch E
Pin 7	SW_F	Switch F
Pin 8	SW_G	Switch G
Pin 9	SW_H	Switch H
Pin 10	LGND	Logic Ground



passendes Kabel: F4406
(to Keyboard K2050)

```
eigerScript eVM Command-Class: HotKey
HotKey.DeInstallKey( VarInt:Key )
HotKey.DisableLocalKeys( )
HotKey.EnableLocalKeys( )
HotKey.InputFlush( )
HotKey.InputSelect( )
HotKey.Install( b , procName )
HotKey.InstallLocalKey( VarInt:Key , labelRelative24:Event , VarInt:Tag )
HotKey.TableDisable( )
HotKey.TableEnable( )
HotKey.TableInit( )
```

Schaltungsbeispiele

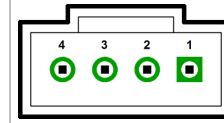
<p>Verwenden eines Tasters mit einem Keyboard-Eingang: (SW_X steht als Platzhalter für SW_A .. SWH)</p>	
<p>Verwenden eines Schalters mit einem Keyboard-Eingang (SW_X steht als Platzhalter für SW_A .. SWH)</p>	

[< Inhaltsverzeichnis](#)

CN11 Anschluss Touchscreen (Ext.)**(wird nicht bestückt)**

Steckertyp	AVX 8370 4-Pol	Wiring Harness: Molex SL Gehäuse 4-Pol
------------	----------------	----------------------------------------

Pin 1	TP_X-	Touchpanel X-direction -
Pin 2	TP_Y+	Touchpanel Y-direction +
Pin 3	TP_X+	Touchpanel X-direction +
Pin 4	TP_Y-	Touchpanel Y-direction -



eigerScript	eVM Command-Class: HotSpot
-------------	-----------------------------------

```
HotSpot.Install( labelRelative24:Event , labelRelative24:Event ,
                labelRelative24:Event , labelRelative24:Event )
```

```
HotSpot.DeInstall( )
```

```
HotSpot.DeInstallGroup( )
```

```
HotSpot.GetNextGroupNumber( )
```

```
HotSpot.GetCurrGroupNumber( )
```

```
HotSpot.TableDestroy( )
```

```
HotSpot.TableDisable( )
```

```
HotSpot.TableEnable( )
```

```
HotSpot.DisableRegion( )
```

```
HotSpot.EnableRegion( )
```

```
HotSpot.Disable_BaseGroup( )
```

```
HotSpot.Enable_BaseGroup( )
```

```
HotSpot.Disable_By_ID( VarInt:HotSpot_ID )
```

```
HotSpot.Enable_By_ID( VarInt:HotSpot_ID )
```

```
HotSpot.DisableGroup( VarInt:Group )
```

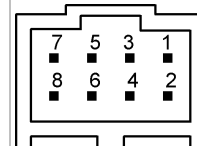
```
HotSpot.EnableGroup( VarInt:Group )
```

```
HotSpot.GetInfo_By_ID( VarInt:HotSpot_ID )
```

[< Inhaltsverzeichnis](#)
JTAG Programmier-Schnittstelle für EVE anna (CPLD)

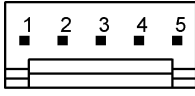
Steckertyp	Molex C-Grid III 8-Pol	Wiring Harness: Molex C-Grid III Gehäuse 8-Pol
------------	------------------------	------------------------------------------------

Pin 1	L3V3	Logic +3.3V
Pin 2	TDO	Test Data Output
Pin 3	TDI	Test Data Input
Pin 4	ISPEN	VSS
Pin 5	TRST	Test Reset
Pin 6	TMS	Test Mode Select
Pin 7	LGND	Logic Ground
Pin 8	TCK	Test Clock


[< Inhaltsverzeichnis](#)

RS2 FOX-COM2 (UART2), serielle Schnittstelle RS232

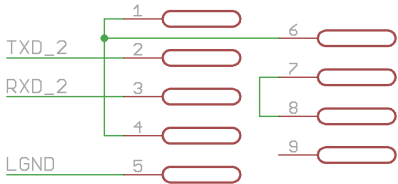
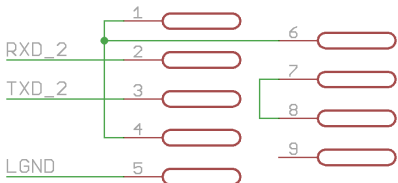
Steckertyp	TYCO CST100 5-Pol	Wiring Harness: AMP MTA100 5-Pol
Pin 1	L3V3	Logic +3.3V
Pin 2	TXD_2	RS232 Transmit Line
Pin 3	LGND	Logic Ground
Pin 4	RX_2	RS232 Receive Line
Pin 5	L12V	Logic +12V



passendes Kabel: F4259 , F4339

eigerScript	eVM Command-Class: Serial
<code>Serial.Rx_Get_EventChar(VarInt:COM2 , VarInt:Char)</code>	
<code>Serial.Rx_InBuf_Clear(VarInt:COM2)</code>	
<code>Serial.Rx_InBuf_LastChar(VarInt:COM2 , VarInt:LastChar)</code>	
<code>Serial.Rx_InBuf_Length(VarInt:COM2 , VarInt:Length)</code>	
<code>Serial.Rx_InBuf_to_String_Append(VarInt:COM2 , VarStr:InputString)</code>	
<code>Serial.Rx_Monitor_ClearLocal(VarInt:COM2)</code>	
<code>Serial.Rx_Monitor_DisableLocal(VarInt:COM2)</code>	
<code>Serial.Rx_Monitor_EnableLocal(VarInt:COM2)</code>	
<code>Serial.Rx_Monitor_ScanContinue(VarInt:COM2)</code>	
<code>Serial.Rx_MonitorSlot_Disable(VarInt:COM2 , VarInt:Slot)</code>	
<code>Serial.Rx_MonitorSlot_Enable(VarInt:COM2 , VarInt:Slot)</code>	
<code>Serial.Rx_MonitorSlot_InstallLocal(VarInt:COM2 , VarInt:Slot , labelRelative24:Event , VarInt:CharLO , VarInt:CharHI)</code>	
<code>Serial.SetBaudrate(VarInt:COM2 , VarInt:Baud_9600)</code>	
<code>Serial.Tx_Char(VarInt:COM2 , VarInt:Char)</code>	
<code>Serial.Tx_CRLF(VarInt:COM2)</code>	
<code>Serial.Tx_NUL(VarInt:COM2)</code>	
<code>Serial.Tx_String(VarInt:COM2 , VarStr:String)</code>	

Schaltungsbeispiele

<p>Verwendung des FOX-COM2 (UART2) zur Verbindung des FOX mit dem Computer:</p> <p>(F4339)</p>	<p>Sub-D9 female</p> 
<p>Verwendung des FOX-COM2 (UART2) zur Verbindung des FOX (als DTE) mit einem anderen Gerät:</p> <p>(F4259)</p>	<p>Sub-D9 male</p> 

[< Inhaltsverzeichnis](#)**Buzzer**

eigerScript	eVM Command-Class: Buzzer
<code>Buzzer.Off()</code>	
<code>Buzzer.On()</code>	
<code>Buzzer.Timer(VarInt:Time)</code>	
<code>Buzzer.Alarm(VarInt:ON_Time , VarInt:OFF_Time , VarInt:Pulse , VarInt:Pause , VarInt:Loops)</code>	
<code>Buzzer.Click()</code>	
<code>Buzzer.Touch(VarInt:Time)</code>	

Alternativ kann der Buzzer gleich angesteuert werden wie der digitale Ausgang **OP90** (siehe Seite 9). Der Portname für den Buzzer lautet **OP92**.

Displaybild spiegeln

Ist eine «verdrehte» Montage des eigerPanels unabdingbar, kann die Darstellung des Displaybildes mittels einiger Manipulationen an den Widerständen **R80**, **R81**, **R82** und **R86** auf dem eigerPanel-Mainboard, korrigiert werden. Diese Korrektur erfolgt unabhängig vom laufenden Betriebssystem oder der auszuführenden Software.

ACHTUNG! Der Touchscreen wird mit dieser Manipulation nicht mitgespiegelt!

Die Standard-Werte für die Widerstände sind:

U/D	R80 = 00hm	R86 = 22kOhm
L/R	R81 = 22kOhm	R82 = 00hm

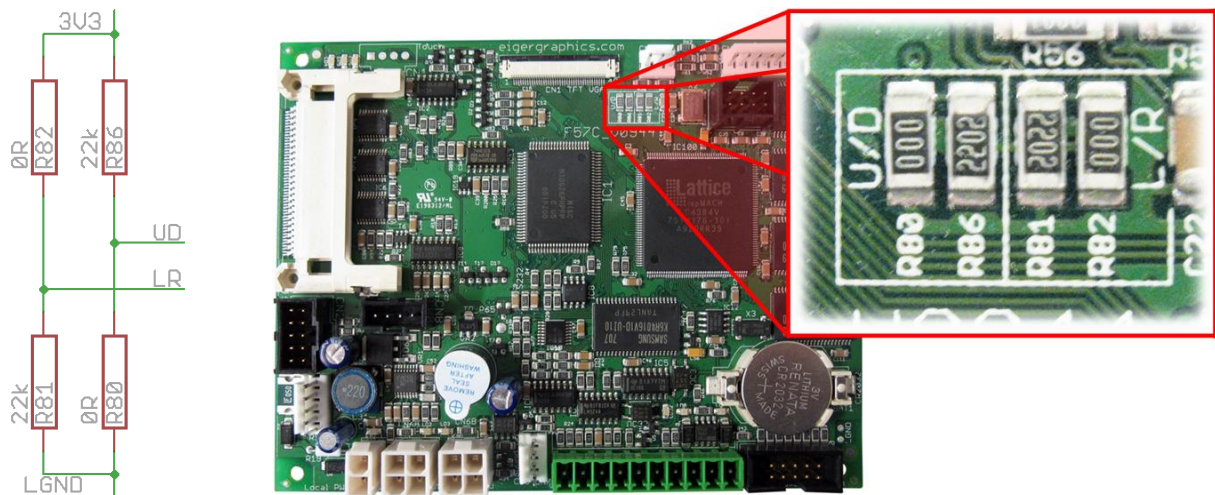


Abbildung 3: Displaybild-Einstellung

Durch Vertauschen der Widerstände **R80** & **R86**, wird das Bild an der X-Achse gespiegelt.



Abbildung 4: Displaybild an der X-Achse spiegeln (UD = Up/Down)

Durch Vertauschen der Widerstände **R81** & **R82** wird das Bild an der Y-Achse gespiegelt.

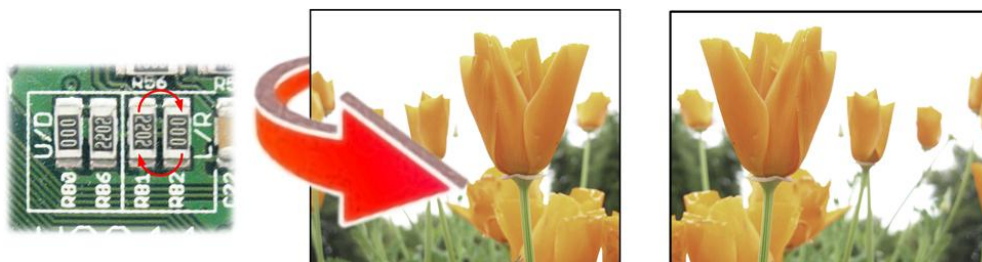


Abbildung 5: Displaybild an der Y-Achse spiegeln (LR = Left/Right)

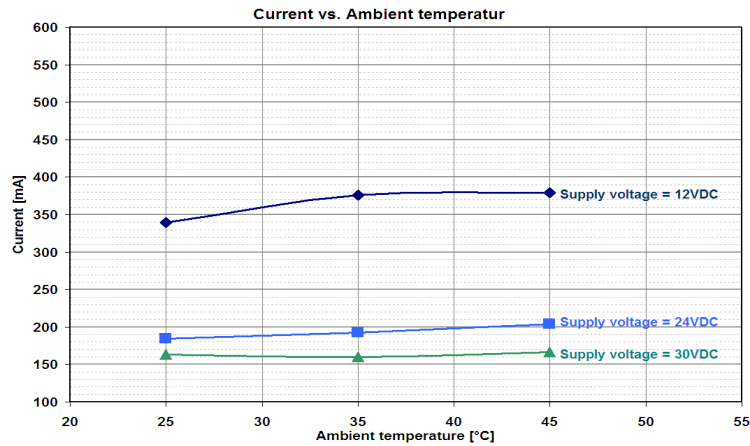
Spezifikationen

Energie

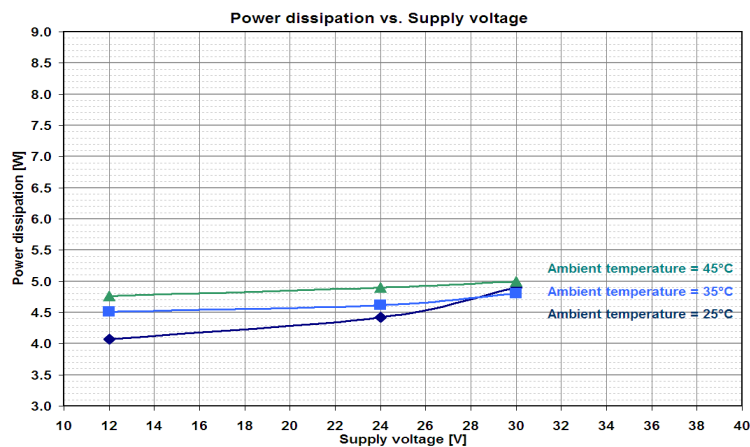
		25°C	35°C	45°C	25°C	35°C	45°C		
		mit Display			ohne Display				
Stromverbrauch	Speisung: 12VDC	339.9	376.0	397.0	124.6	139.8	139.3	mADC	
	Speisung: 24VDC	184.2	192.3	204.0	79.9	83.4	82.1		
	Speisung: 30VDC	163.4	159.9	166.8	73.3	72.9	72.3		
Verlustleistung	Speisung: 12VDC	4.07	4.51	4.76	1.50	1.68	1.65	W	
	Speisung: 24VDC	4.42	4.62	4.90	1.92	2.00	1.97		
	Speisung: 30VDC	4.90	4.80	5.01	2.20	2.19	2.17		

		MIN	TYP	MAX	
Versorgungsspannung		9	12 / 24	30	VDC
Temperaturbereich	Betrieb	-20	-	+70	°C
	Lagerung	-30	-	+80	

Strombedarf in Funktion mit der Umgebungstemperatur. Messungen wurden durchgeführt bei drei verschiedenen Versorgungsspannungen. Mit steigender Umgebungstemperatur steigt der Stromverbrauch des eigerPanels.



Verlustleistung in Funktion mit der Versorgungsspannung. Messungen wurden bei drei verschiedenen Umgebungstemperaturen durchgeführt. Der Leistungsbedarf des eigerPanels nimmt mit steigender Versorgungsspannung zu.



< [Inhaltsverzeichnis](#)

Speicher

Typ	Compact Flash™ (CFC) / RAM
nutzbarer Speicher für Programmierdaten	durch Speicherkapazität des CFC gegeben
nutzbarer Arbeitsspeicher	280kB (fix)

Schnittstellen

BUS	seriell, RS485 seriell, I ² C	Speisung über BUS möglich
UART1	seriell, RS232	für Debugging / Download
UART2	seriell, RS232	frei für Applikation
Eingänge		
analog	1x Potentiometer	Spannungsbereich 0 .. 3.3VDC
	2x NTC / Potentiometer	Spannungsbereich 0 .. 3.3VDC
	1x Spannungs-Messeingang	Spannungsbereich 0 .. 50VDC
digital	8x Funktionstasten	Spannungsbereich 0V / 3.3VDC z.B. Keyboard oder Sensoren etc.
	1x Zähler-Eingang bis 3kHz	Spannungsbereich 5 .. 30VDC 3 .. 20VAC
Ausgänge		
analog	2x programmierbare Spannungsausgänge	Spannungsbereich 0 .. 10VDC
digital (PWM)	1x digitaler Leistungsausgang bis 1A (OP76)	0 .. 1000 → Duty Cycle 0 .. 100% Spannungsbereich 9 .. 30VDC (entspricht der Speis-Spannung)
digital	1x digitaler Leistungsausgang bis 1A (OP90)	Spannungsbereich 9 .. 30VDC (entspricht der Speis-Spannung)

Weitere Eigenschaften

Grafische Leistung	Der FOX57 bringt innerhalb einer Sekunde bis 160 Labels (Buttons, Titel, Schriftzüge, etc.) auf den Bildschirm. Pro Sekunde kann der FOX bis zu 70 mal die View wechseln.
Videospeicher	zwei Videospeicher (Accessible Video Ram AVR und Refresh Video Ram RVR)
Uhr/Datum	Hardware-Uhr, batteriegestützt
Batterie RTC Clock	Li-Batterie CR2032, 3V, 235mAh, 10 Jahre Gangreserve
akustischer Signalgeber	Buzzer (OP92)
Bootzeit ab Kaltstart	4 Sekunden

LCD Anschlüsse

LCD Typ	TFT LCD, VGA 640 x 480
Spannungsversorgung für LCD	Keine separate Speisung für Display notwendig, geschieht über den LCD Anschluss (CN1).
Hintergrundbeleuchtung Ansteuerung	5VDC
Touchscreen	4 Wire, resistive

[< Inhaltsverzeichnis](#)

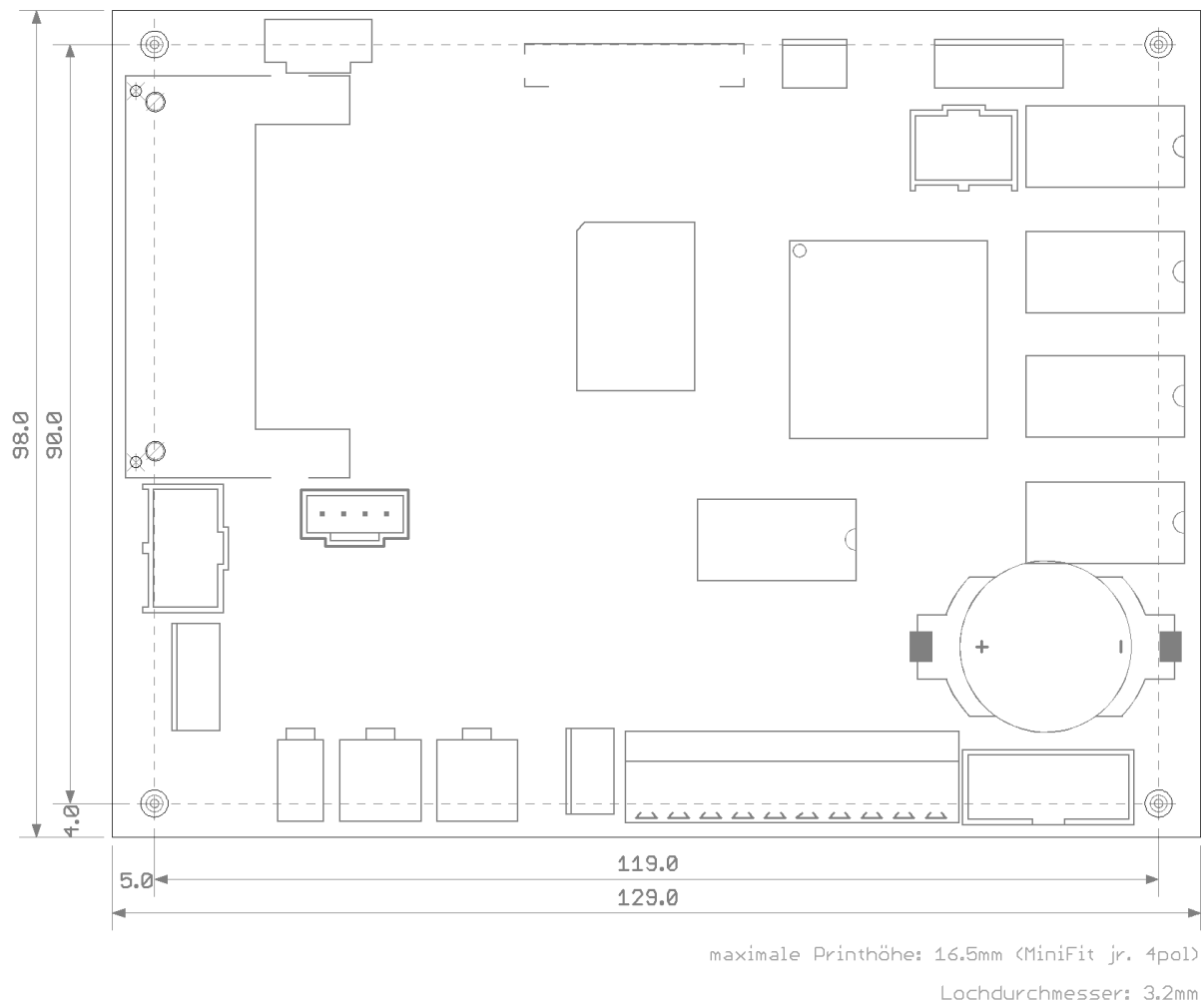
Abmessungen

Abbildung 6: Grundriss und Abmessungen der Rechnerplatine F57C_V0944

[< Inhaltsverzeichnis](#)

Die beiden Modelle F57C und F57H sind grösstenteils geometrisch kompatibel zu einander. Die Unterschiede sind auf der folgenden Seite beschrieben.

Unterschiede F57C zu F57H

Grundsätzlich unterscheiden sich die beiden Modell F57C und F57H nur in der Ausrichtung des CFC-Steckers. Funktionell und technisch sind die beiden Modelle identisch.

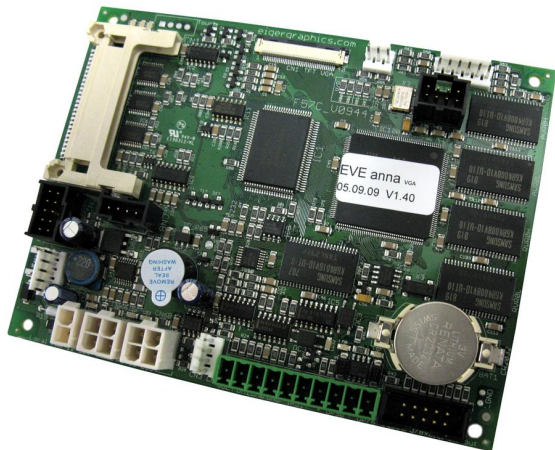


Abbildung 7: Rechnerplatine F57C_V0944

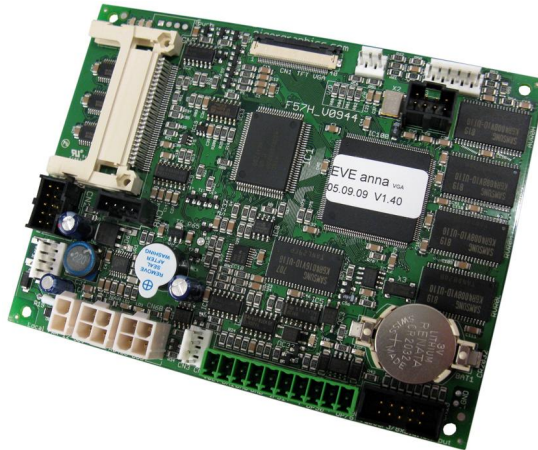


Abbildung 8: Rechnerplatine F57H_V0944



Abbildung 9: F57C (links) und F57H (rechts)

Während beim F57H-Modell die Compact Flash Card über den Rand der Rechnerplatine hinaus ragt, kann das F57C-Modell problemlos in einem geschlossenes Gehäuse montiert werden, da die Compact Flash Card über das Innere der Rechnerplatine ragt.

Zur Auslieferung werden zudem zwei verschiedene Blechgehäuse verwendet. Eines mit CFC-Aussparung für das Modell F57H und eines ohne CFC-Aussparung für das Modell F57C.

Die Variante mit Aussparung (F57H) eignet sich besonders für Entwickler, da die CFC dabei oft gewechselt werden muss.

Die kompaktere Variante F57C ist eher für eine fixe Montage z.B. in einer Maschine gedacht.

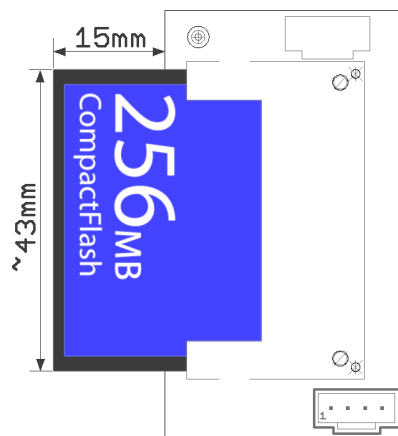


Abbildung 10: Abmessungen eingesteckte CFC (nur F57H)

Abmessungen von Rahmen und Gehäuse (optional)

Die Pläne mit den genauen Abmessungen finden Sie als PDF-Datei im Internet unter folgenden Links:

- Gehäuse-Öffnung:
http://www.eigergraphics.com/Download/docs/Abmessungen_eP57_Oeffnung_fuer_Gehaeuse.pdf
- Gehäuse:
http://www.eigergraphics.com/Download/docs/Abmessungen_eP57_Gehaeuse.pdf
- STEP-Modell (CAD) eigerPanel57C mit Gehäuse:
http://www.eigergraphics.com/Download/eP_CAD/EP57C-STEP.zip
- STEP-Modell (CAD) eigerPanel57H mit Gehäuse:
http://www.eigergraphics.com/Download/eP_CAD/EP57C-STEP.zip
- Display (je nach Display-Typ sind kleinere Abweichungen möglich):
http://www.eigergraphics.com/touchscreens.htm#DI_Display57

Firm- und Software

Graphic Controller, eigerVideo Engine	EVE anna V1.40
Firmware	F57C_V0938_EVE_ANNA_091125_landscape.mot
Firmware Update	http://www.eigergraphics.com/download.htm
Software, Application Notes für die Programmierung der Anschlüsse	http://www.eigergraphics.com/download.htm

Kontakt / Information

Telefon	+ 41 41 754 50 10
Telefax	+ 41 41 754 50 19
Web-Link	http://www.eigergraphics.com/
Suchen Stichworte/Hilfe	http://www.eigergraphics.com/keywords.htm
Email	eiger@s-tec.ch

Datenblatt Version

Version	11. Dezember 2009
Update	30. Juni 2011
Autor	Samuel Nussbaumer, S-TEC electronics AG

Änderungshistory

24. November 2010	SN	- Netzname von CN2.6 korrigiert (S-PROG20-Schnittstelle) zuvor: LGND «Logic Ground» neu: L3V3 «Logic +3.3V»	Seite 4
26. Januar 2011	SN	- Versorgungsspannungsbereich wurde angepasst - Temperatur-Bereich(e) für Betrieb und Lagerung wurde angepasst - Eigenschaften digitaler Zählereingangs wurden angepasst - Eigenschaften analoge Spannungsausgänge wurden angepasst - Eintrag «maximale Erwärmung» wurde entfernt	Seite 15 Seite 15 Seite 16 Seite 16
28. Januar 2011	SN	- Taktfrequenz des Mikrocontrollers angegeben	Seite 2



FOX embedded computers 

the canny swiss solution

© Copyright 2010 S-TEC electronics AG, Inc. Alle Rechte vorbehalten



S-TEC electronics AG

Gewerbestrasse 6 • CH-6314 Unterägeri • Telefon (+41) 41 / 754 50 10 • Telefax (+41) 41 / 754 50 19
info@s-tec.ch • www.s-tec.ch