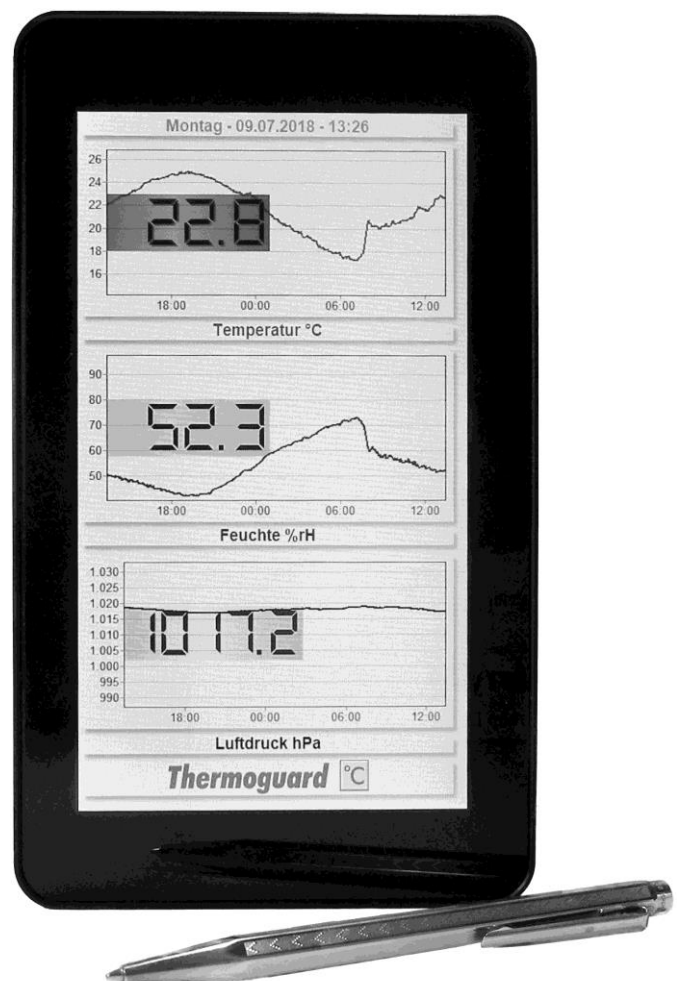


Thermoguard Report Display mit Raspberry Pi



Inhalt

- Einleitung	3
- Das Beispiel	4
- Das wird benötigt	4
- Das Prinzip	4
- Zwischenfrage (und Antwort)	4
- Das ist zu tun Teil 1: Auf dem "Pi"	5
- Vorbereitung	5
- Display um 90° drehen	5
- Samba Freigabe auf dem Pi einrichten	6
- Webseite erstellen	7
- Autostart einrichten	8
- Helligkeit des Display anpassen	8
- "Vertikal-Hilfe"	8
- Code der html-Seite	9
- Das ist zu tun Teil 2: Auf dem <i>Thermoguard Report</i> Rechner	11
- <i>Live!</i> -Fenster erstellen	11
- Die zugehörige *.tgi Datei	12
- Datei für Kopier-Job erstellen	12
- Datei für Kopier-Job als Aufgabe einrichten	13

Einleitung

Diese Dokumentation erläutert die Verwendung eines [Raspberry Pi](#) Moduls als *Thermoguard Report Live!* Anzeigeeinheit mit folgenden Vorteilen:

- Preiswert
- Anzeigeeinheit ohne Windows, keine Microsoft-Lizenzgebühren
- Eigenständige kompakte Einheit
- Es wird nur ein USB-Steckernetzteil benötigt (Betrieb über WLAN)
- Positionierung an jedem beliebigen Ort in WLAN-Reichweite
- Horizontaler oder vertikaler Betrieb
- Darstellung der Anzeige-Elemente mittels html-Code einfach anzupassen
- Schlichtes unauffälliges Design
- Geringer Stromverbrauch
- Keine tieferegehenden UNIX-Kenntnisse erforderlich
- Breite "How To"-Unterstützung im Internet für die beliebten Raspberry-Module



*Die fertige Anwendung (Kugelschreiber als Größenvergleich)
Sechs Thermoguard Report Live! Fenster werden alle 2 Minuten aktualisiert.*

Das Beispiel

In unserem Beispiel zeigt das Display in vertikaler Ausrichtung die Außentemperatur, die Luftfeuchte und den atmosphärischen Luftdruck in jeweils einem Werte- und einem Diagramm-Fenster (Verlauf der letzten 24 Stunden).

Für Ihren Anwendungsfall können Sie das Beispiel entsprechend abändern. Natürlich ist z.B. auch die Anzeige nur eines einzigen Digital-Fensters im waagerechten Format möglich. Dann kann das gesamte Display ausgefüllt werden und ist auch aus größerer Entfernung gut ablesbar. Sie können aber auch viel mehr Anzeige-Fenster in der Webseite definieren als auf das (touchfähige!) Display "passen"; der Anwender kann diese dann bei Bedarf per Touch durchscrollen.

Das wird benötigt

- Raspberry Pi Modell 3B+
- Raspberry Pi Shield - LCD-Touch-Display, 7", 800x480 Pixel
- Gehäuse für Raspberry Pi 3 & 7" Touch-Display
- Micro-SD-Karte mit Raspbian Betriebssystem
- USB-Steckernetzteil 5V, >= 1,5A (Anschlusskabel mit Micro-USB-Stecker)

Das Prinzip

Das Funktionsprinzip entspricht dem, welches bereits für unsere Demo-Website im Dokument [TG-Upload](#) beschrieben wurde. Der Unterschied besteht lediglich darin, dass die *Live!*-Grafiken vom *Thermoguard Report*-Rechner nicht auf einen Webserver hochgeladen, sondern einfach in eine Verzeichnis-Freigabe des Raspberry Pi kopiert werden:

- Ein beliebiger Windows-Rechner, auf dem *Thermoguard Report* (im folgenden "TGR") läuft, speichert die gewünschten *Live!*-Fenster bei jeder Änderung automatisch ab.
- Von diesem Rechner kopiert ein Windows-Task ("geplante Aufgabe") alle 2 Minuten die Dateien in die Freigabe auf dem Raspberry Pi
- Auf dem Raspberry Pi läuft ständig eine Webseite im Vollbildmodus. Sie referenziert die Grafiken mit den identisch passenden Dateinamen. Der Code der Webseite enthält eine Anweisung für einen automatischen Refresh. So werden immer die gerade hochgeladenen Grafiken angezeigt.

Zwischenfrage (und Antwort)

F: Es gibt doch eine Windows-Emulation für Raspbian. Warum verwenden wir diese nicht und lassen TGR direkt auf dem Raspbian laufen?

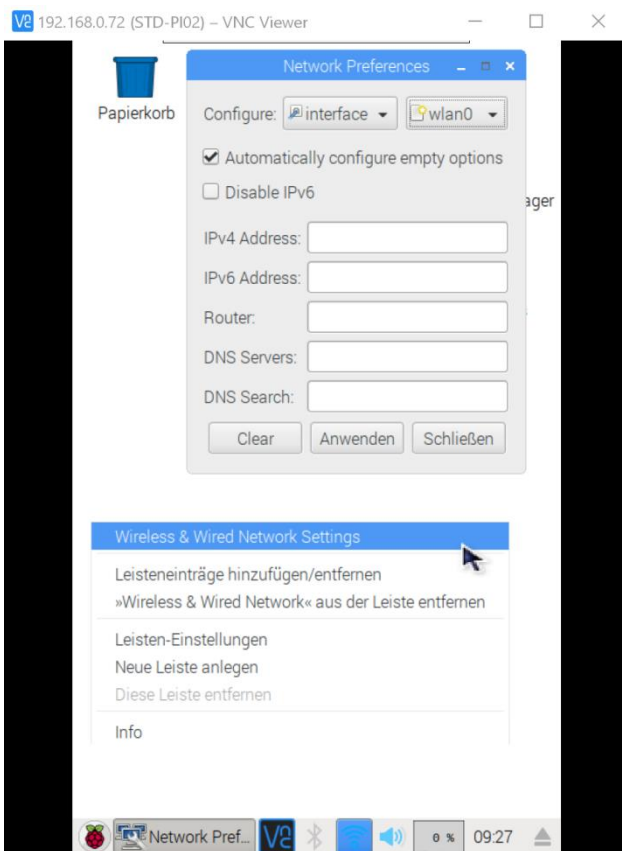
A: Der Softwarestapel Debian/Arm, Exagear X86 Emulation, Debian X86, Wine Windows Emulation lief in unseren Tests mit TGR leider nicht langzeitstabil, daher dokumentieren wir hier nur das o.a. Prinzip.

Das ist zu tun Teil 1: Auf dem "Pi"

Vorbereitung

Das 7" Display und das Raspberry-Modul selbst sind in wenigen Minuten in das passende Gehäuse eingebaut. Im weiteren gehen wir davon aus, dass die Micro-SD-Karte bereits mit einem Raspbian Betriebssystem versehen ist, so dass nach Einstecken der Karte und Anschluss des Steckernetzteils schon ein betriebsfähiges Modul zur Verfügung steht.

Maus und Tastatur werden temporär via USB angeschlossen und der "Raspi" zunächst mit einem Standard-Patchkabel in das Netzwerk gehängt und eine passende IP-Adresse eingerichtet bzw. per DHCP vergeben:



Netzwerkeinstellungen

Hier ist das Display bereits um 90° gedreht und der Zugriff auf den "Pi" erfolgt via WLAN und VNC

Danach lässt sich der kleine Rechner bequem von einem Windows-Rechner mit dem VNC-Client weiter einrichten und das integrierte WLAN aktivieren.

Display um 90° drehen

Fügen Sie der Datei config.txt mit ...

```
sudo nano /boot/config.txt
```

... folgenden Eintrag hinzu:

```
display_rotate=1
```

Samba Freigabe auf dem Pi einrichten

[Information hierzu](#). Die verwendete smb.conf sieht so aus:

```
[global]
workgroup = IHRWORKGROUPNAME
security = user
encrypt passwords = yes

[SambaUpload]
comment = Samba-Upload-Directory
path = /home/shares/upload
read only = no
```

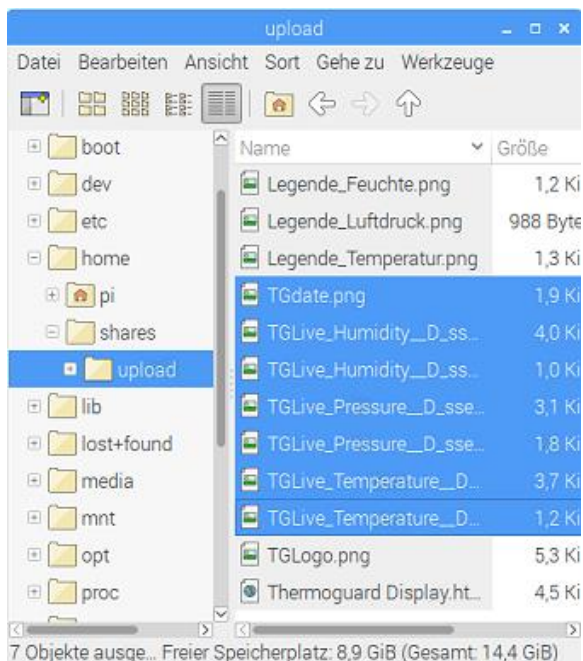
Das Anlegen bzw. Editieren von Konfigurationsdateien geschieht stets an der Kommandozeile eines Terminal-Fensters mit dem Befehl

```
sudo nano [Dateiname]
hier
sudo nano /etc/samba/smb.conf
```

In **diesen Pfad** werden später alle 2 Minuten die Grafiken vom TGR-Rechner hineinkopiert. Hier erstellen wir auch die Webseite, die die Grafiken referenziert.

"SambaUpload" ist der im Netzwerk sichtbare Freigabename.

Testen Sie anschließend den Zugriff auf die Freigabe von einem Windows-Rechner aus.



Das Samba Freigabeverzeichnis im Dateimanager

*Die markierten *.png Dateien werden regelmäßig vom TGR-Rechner aktualisiert, alle anderen Dateien verbleiben unverändert im Verzeichnis.*

Webseite erstellen

Sie können Ihre eigene Datei Thermoguard Display.html bequem am Windows-Rechner erstellen und dann in das o.a. Freigabeverzeichnis kopieren.

Den Code der html-Datei unserer Beispiel-Webseite finden Sie auf der Seite 9f.

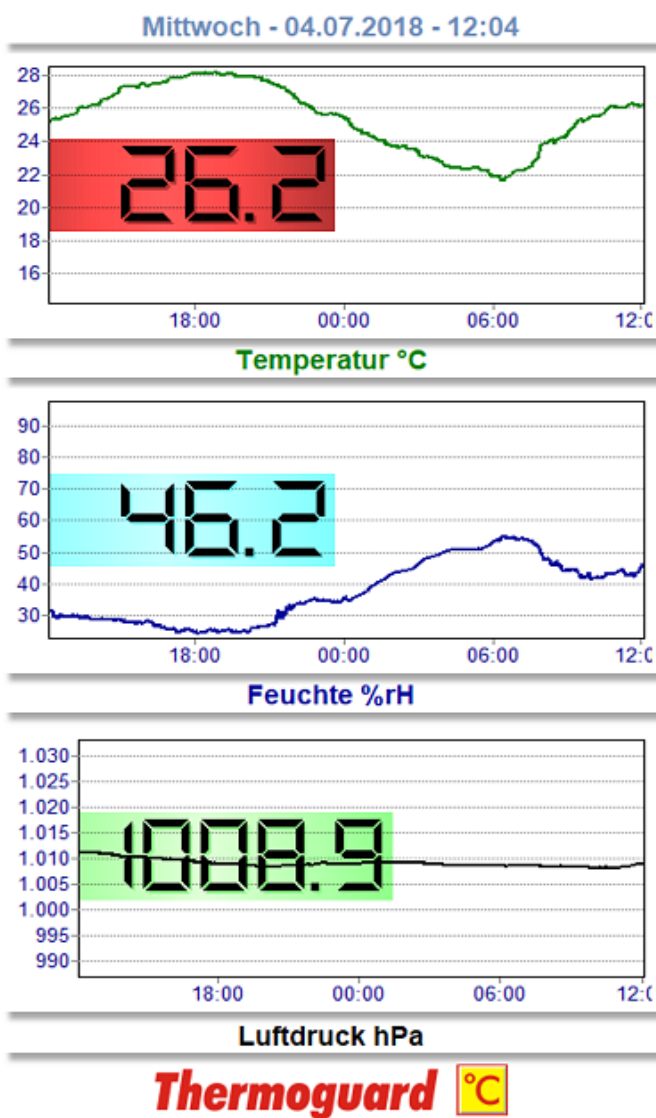
Wir verwenden darin die Überblend-Fähigkeiten ("background-blend-mode") von CSS, um die Werte-Fenster transparent über die Diagramm-Fenster zu legen.

"background-blend-mode" wird derzeit nur vom Chromium-Browser und dem aktuellen Firefox unterstützt, nicht von anderen Browsern (z.B. Microsoft Internet-Explorer). Da wir ohnehin den Chromium-Browser verwenden, ist dieser Umstand für unseren "internen Einsatzzweck" nicht relevant.

Bis die Grafiken genau wie gewünscht auf das Display passen ist ein wenig "Trial and Error" erforderlich.

Beachten Sie die Refresh-Angabe in der <head>-Sektion des html-Codes. Hier ist angegeben, dass alle 10 Sekunden die Webseite neu geladen werden soll:

```
<meta http-equiv="refresh" content="10; URL=file:///home/shares/upload/Thermoguard%20Display.html">
```



Die Webseite als direkter Screenshot

Die Schatten werden durch die Anweisung box-shadow... erzeugt

Autostart einrichten

Damit auch nach dem Aus- und Wiedereinschalten des Raspberry Pi unsere Anzeige automatisch im Vollbildmodus erscheint ("Kioskmodus"), fügen wir mit ...

```
sudo nano /home/pi/.config/lxsession/LXDE-pi/autostart
```

... der Datei autostart die folgenden Zeilen hinzu:

```
@xset s off
@xset -dpms
@xset s noblank
@chromium-browser --noerrdialogs --kiosk --incognito /home/shares/upload/Thermoguard%20Display.html
#~~~ Hinweis: Kioskmodus beenden mit STRG+F4
```

Helligkeit des Display anpassen

Ändern Sie den Wert in der Datei brightness mit ...

```
sudo nano /sys/class/backlight/rpi_backlight/brightness
```

... von 0 (aus) bis 255 (maximal)

"Vertikal-Hilfe"



Doppelseitig klebendes Montageband und ein Stück Karton verhelfen unserem Modul zu einem aufrechten Stand.

Code der html-Seite

```
<!DOCTYPE html>
<html>

<head>
<title>Thermoguard Display</title>
<meta http-equiv="refresh" content="10; URL=file:///home/shares/upload/Thermoguard%20Display.html">
<style type="text/css" media="screen">

#Datum {
width: 460px;
height: 19px;
margin: 0px -5px;
background-image: url("TGdate.png");
background-repeat: no-repeat;
background-position: center 0px;
box-shadow: 5px 5px 5px 0px #aaaaaa;
}

#Temperatur {
width: 460px;
height: 200px;
margin: 7px -5px;
background-image:
url("TGLive_Temperature__D_sseldorf__crt_w.png"),
url("TGLive_Temperature__D_sseldorf__lcd_d.png");
background-size: 460px 200px, 200px auto;
background-repeat: no-repeat, no-repeat;
background-position: 0px 0px, 33px 60px;
box-shadow: 5px 5px 5px 0 #aaaaaa;
background-blend-mode: multiply;
}

#LegendeTemperatur {
width: 460px;
height: 19px;
margin: 8px -5px;
background-image: url("Legende_Temperatur.png");
background-repeat: no-repeat;
background-position: center 0px;
box-shadow: 5px 5px 5px 0px #aaaaaa;
}

#Feuchte {
width: 460px;
height: 200px;
margin: 0px -5px;
background-image:
url("TGLive_Humidity__D_sseldorf__crt_w.png"),
url("TGLive_Humidity__D_sseldorf__lcd_d.png");
background-size: 460px 200px, 200px auto;
background-repeat: no-repeat, no-repeat;
background-position: 0px 0px, 33px 60px;
box-shadow: 5px 5px 5px 0 #aaaaaa;
background-blend-mode: multiply;
}

#LegendeFeuchte {
width: 460px;
height: 19px;
margin: 8px -5px;
background-image: url("Legende_Feuchte.png");
background-repeat: no-repeat;
background-position: center 0px;
box-shadow: 5px 5px 5px 0px #aaaaaa;
}
}
```

```
#Luftdruck {
  width: 460px;
  height: 200px;
  margin: 10px -5px;
  background-image:
    url("TGLive_Pressure__D_sseldorf__crt_w.png"),
    url("TGLive_Pressure__D_sseldorf__lcd_d.png");
  background-size: 460px 200px, 220px auto;
  background-repeat: no-repeat, no-repeat;
  background-position: 0px 0px, 54px 60px;
  box-shadow: 5px 5px 5px 0 #aaaaaa;
  background-blend-mode: multiply;
}
#LegendeLuftdruck {
  width: 460px;
  height: 19px;
  margin: 8px -5px;
  background-image: url("Legende_Luftdruck.png");
  background-repeat: no-repeat;
  background-position: center 0px;
  box-shadow: 5px 5px 5px 0px #aaaaaa;
}

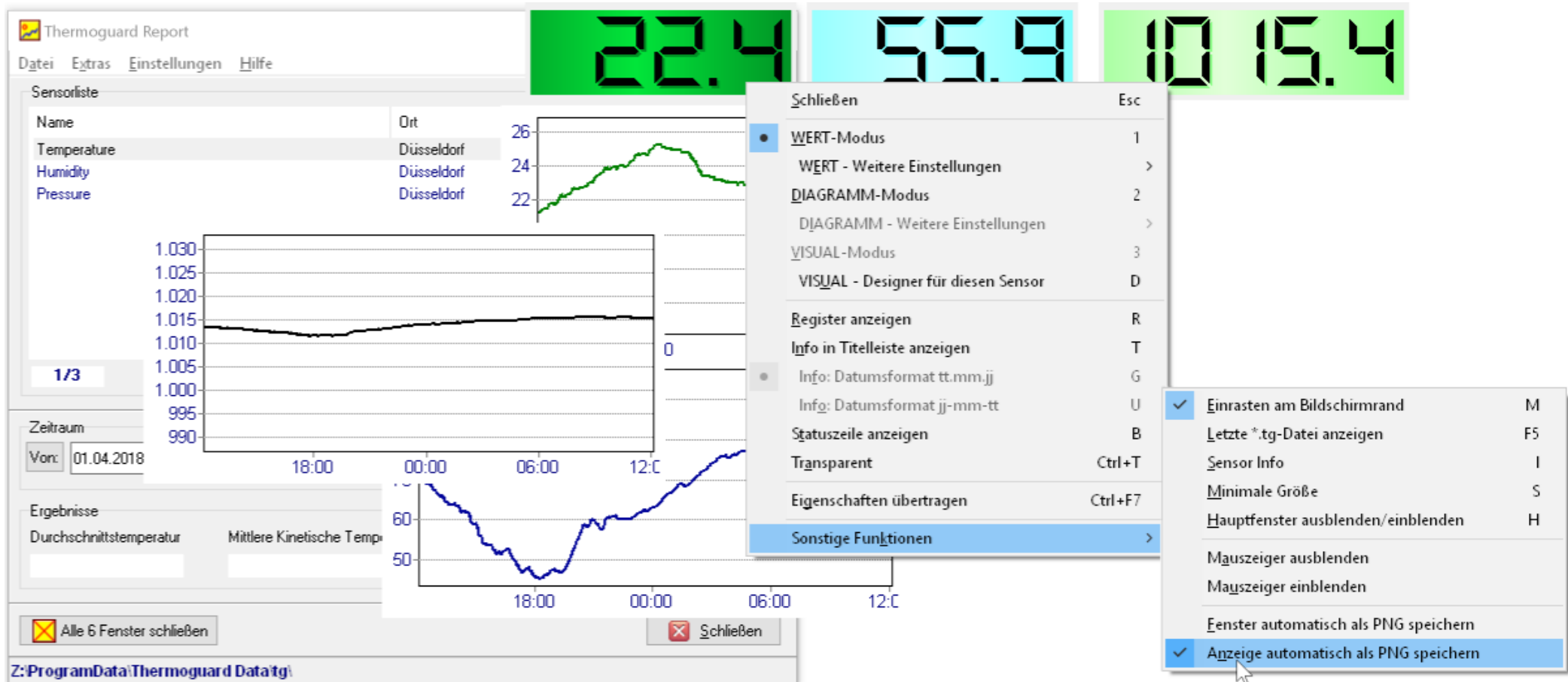
#Logo {
  width: 460px;
  height: 40px;
  margin: 0px -5px;
  background-image: url("TGLogo.png");
  background-repeat: no-repeat;
  background-position: center 0px;
  box-shadow: 5px 5px 5px 0 #aaaaaa;
}
</style>
</head>

<body>
<div id="Datum"> </div>
<div id="Temperatur"> </div>
<div id="LegendeTemperatur"> </div>
<div id="Feuchte"> </div>
<div id="LegendeFeuchte"> </div>
<div id="Luftdruck"> </div>
<div id="LegendeLuftdruck"> </div>
<div id="Logo"> </div>
</body>
</html>
```

Das ist zu tun Teil 2: Auf dem *Thermoguard Report* Rechner

Live!-Fensterset erstellen

Verwenden Sie die aktuelle Version von TGR. Auf dem *TGR*-Rechner wird ein passendes *Live!*-Fensterset erstellt. Dies wurde bereits im Dokument [TG-Upload](#) beschrieben. Unser Beispiel besteht aus drei Werte- und drei Diagramm-Fenstern:



The screenshot shows the Thermoguard Report software interface. At the top, there are three digital readouts: 22.4 (green background), 55.9 (cyan background), and 10 15.4 (green background). Below these, a menu is open, listing various display options for the Live! window set. The menu items are:

- Schließen (Esc)
- WERT-Modus (1)
- WERT - Weitere Einstellungen (>)
- DIAGRAMM-Modus (2)
- DIAGRAMM - Weitere Einstellungen (>)
- VISUAL-Modus (3)
- VISUAL - Designer für diesen Sensor (D)
- Register anzeigen (R)
- Info in Titelleiste anzeigen (T)
- Info: Datumsformat tt.mm.jj (G)
- Info: Datumsformat jj-mm-tt (U)
- Statuszeile anzeigen (B)
- Transparent (Ctrl+T)
- Eigenschaften übertragen (Ctrl+F7)
- Sonstige Funktionen (>)

The 'Sonstige Funktionen' sub-menu is also visible, containing the following options:

- Einrasten am Bildschirmrand (M)
- Letzte *.tg-Datei anzeigen (F5)
- Sensor Info (I)
- Minimale Größe (S)
- Hauptfenster ausblenden/einblenden (H)
- Mauszeiger ausblenden
- Mauszeiger einblenden
- Fenster automatisch als PNG speichern
- Anzeige automatisch als PNG speichern (checked)

The background interface shows a sensor list with columns for Name, Ort, and a numerical value. The sensors listed are Temperature (Düsseldorf, 26), Humidity (Düsseldorf, 24), and Pressure (Düsseldorf, 22). There are also two line graphs: one showing a temperature trend over time (18:00 to 12:00) and another showing a pressure trend over time (18:00 to 12:00). The bottom of the interface shows a status bar with the path 'Z:\ProgramData\Thermoguard Data\tg\'. There are also buttons for 'Alle 6 Fenster schließen' and 'Schließen'.

Die zugehörige *.tjl Datei

Temperature,0,401,208,75,1,0,1,1,0,1,1,1,1,0,1,1,0,283,145,283,145,0,1,32768,0	(Werte-Fenster)
Humidity,0,617,208,75,2,0,1,1,0,1,1,1,1,0,1,1,0,283,145,283,145,0,1,32768,0	(Werte-Fenster)
Pressure,0,841,238,75,3,0,1,1,0,1,1,1,1,0,1,1,0,283,145,283,145,0,1,32768,0	(Werte-Fenster)
Temperature,79,382,283,145,1,1,1,1,0,1,0,0,1,1,0,1,0,400,200,283,145,0,2,32768,0	(Diagramm-Fenster)
Humidity,272,291,283,145,1,1,1,1,0,1,0,0,1,1,0,1,0,400,200,283,145,0,2,10159876,0	(Diagramm-Fenster)
Pressure,169,108,283,145,1,1,1,1,0,1,0,0,1,1,0,1,0,400,200,283,145,0,2,0,0	(Diagramm-Fenster)

Die relevanten Breiten- und Höhenangaben für die Fenster sind hervorgehoben. Im Gegensatz zu den Diagramm-Fenstern sind die Werte für die Breite und Höhe der Werte-Fenster nicht 1:1 in den html-Code übertragbar, da hier nur die innere Anzeige abgespeichert wird (die Angaben in der *.tjl Datei beziehen sich auf das gesamte Fenster).

Datei für Kopier-Job erstellen

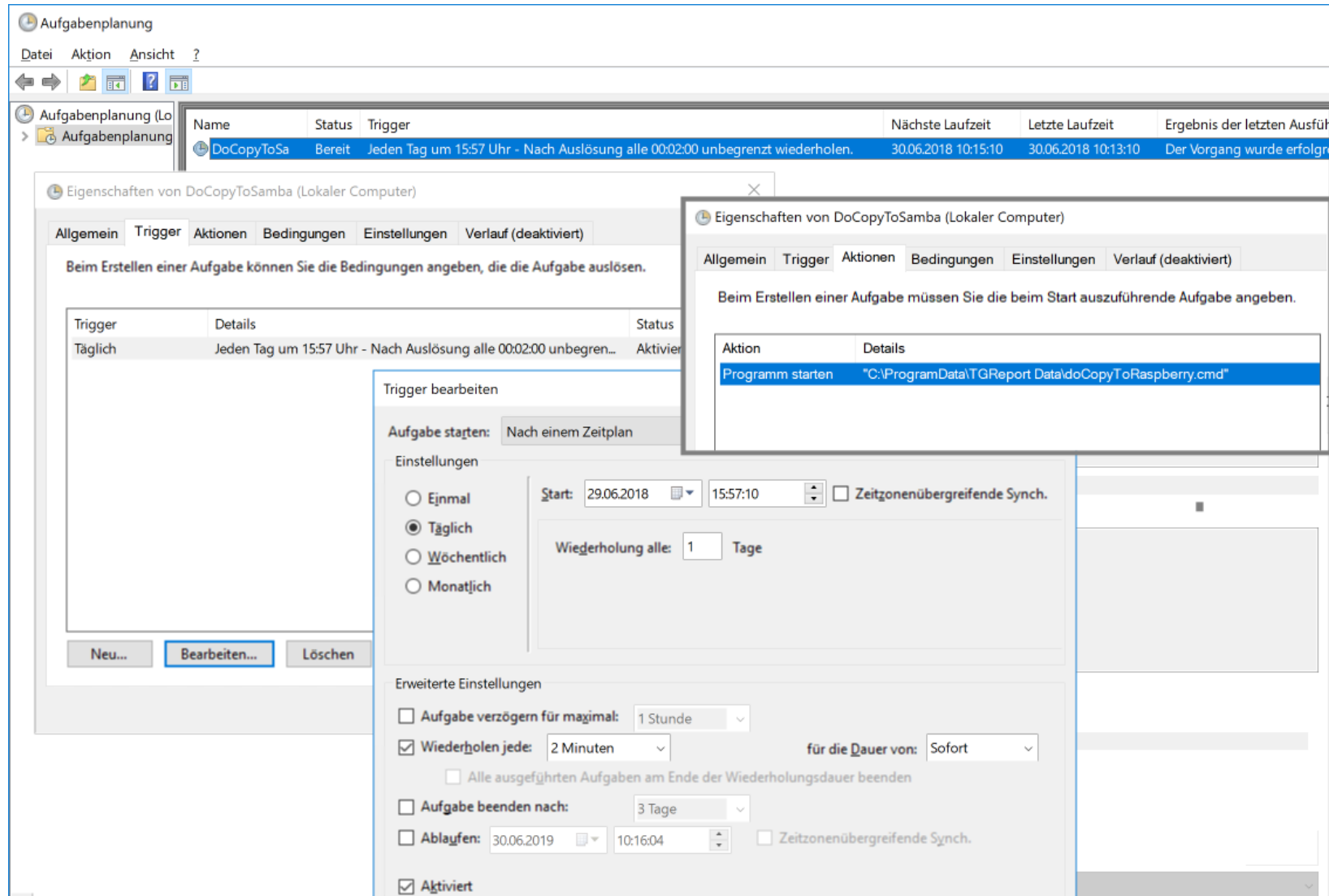
Die Datei doCopyToRaspberry.cmd kopiert die benötigten 6 Dateien auf die Samba Freigabe. Wir speichern sie im TGR-Datenverzeichnis; so sieht sie aus:

```
@echo off
copy "C:\ProgramData\TGReport Data\*Temperature*.png" \\RASPI\SambaUpload
copy "C:\ProgramData\TGReport Data\*Humidity*.png" \\RASPI\SambaUpload
copy "C:\ProgramData\TGReport Data\*Pressure*.png" \\RASPI\SambaUpload
exit
```

RASPI steht hier stellvertretend für den Rechnernamen des Raspberry Pi.

Datei für Kopier-Job als Aufgabe einrichten

Mit der Windows Aufgabenplanung wird nun die Aufgabe "DoCopyToSamba" erstellt; damit ist das Projekt abgeschlossen:



The screenshot displays the Windows Task Scheduler interface. At the top, a table lists the task 'DoCopyToSa' with the following details:

Name	Status	Trigger	Nächste Laufzeit	Letzte Laufzeit	Ergebnis der letzten Ausführung
DoCopyToSa	Bereit	Jeden Tag um 15:57 Uhr - Nach Auslösung alle 00:02:00 unbegrenzt wiederholen.	30.06.2018 10:15:10	30.06.2018 10:13:10	Der Vorgang wurde erfolgreich

Below the table, two property windows are open for the task 'DoCopyToSamba (Lokaler Computer)'. The 'Trigger bearbeiten' window shows the task is set to start 'Nach einem Zeitplan' with the following settings:

- Einstellungen:** Täglich
- Start:** 29.06.2018 15:57:10
- Wiederholung alle:** 1 Tage
- Zeitzoneübergreifende Synchronisierung

The 'Eigenschaften' window shows the action configuration:

- Aktion:** Programm starten
- Details:** "C:\ProgramData\TGRReport Data\doCopyToRaspberry.cmd"

At the bottom of the 'Eigenschaften' window, the 'Erweiterte Einstellungen' section is visible, including options for task delay, repetition (set to 2 minutes), and task completion (set to 3 days).