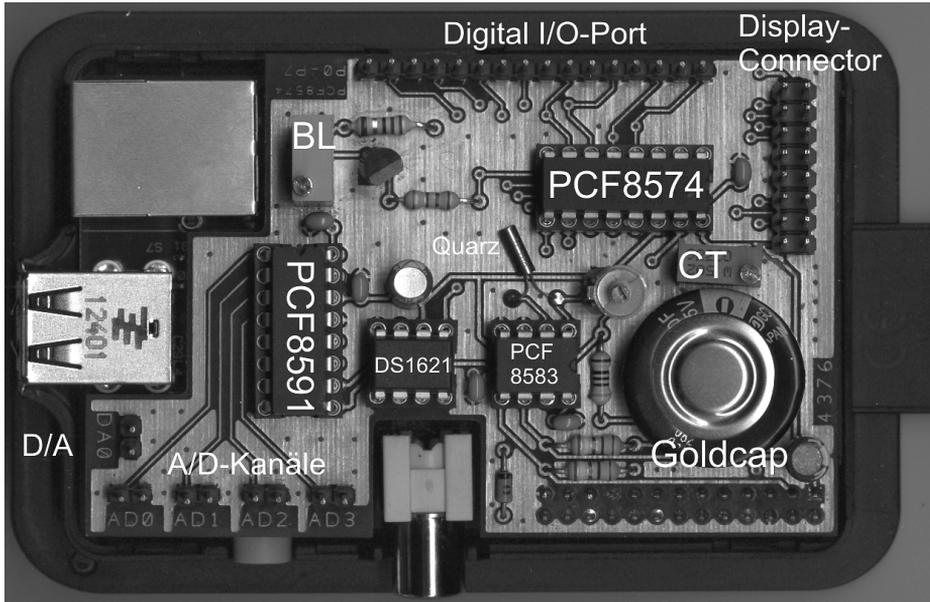


Für die Inbetriebnahme ist es zunächst wichtig, dass alle bestückten Chips korrekt erkannt werden, wie es in der Abbildung 7–11 dargestellt ist. Je nach gewünschter Applikation ist die Software unterschiedlich zu gestalten. Dabei empfiehlt es sich zunächst, möglichst viele Ausgabezeilen (mit printf) vorzusehen, um die korrekte Abarbeitung der einzelnen Schritte kontrollieren zu können.



**Abb. 7–14** Die Erweiterungsplatine in der voll bestückten Version auf dem Raspberry-Pi-Board

### 7.3.1 Temperaturmessung

Als erstes Beispiel für die Erweiterungsplatine wird im Folgenden ein C-Programm (`ds1621_demo.c`) gezeigt, das mit dem Temperatursensor DS1621 unter Verwendung der zwei erwähnten I<sup>2</sup>C-Bibliotheken kommuniziert. Es enthält die Funktion `int read_temp(void)` zum Auslesen der Temperatur des DS1621, die die Temperatur als Integer-Wert zurückgibt.

Die gemessene Temperatur entspricht allerdings nicht der Umgebungstemperatur, was daran liegt, dass die Messplatine von der CPU des Raspberry Pi von unten her aufgeheizt wird und somit nährungsweise die Temperatur der Raspberry-Pi-CPU gemessen wird und nicht die eigentliche Umgebungstemperatur, was nach einiger Betriebszeit zu einer Messtemperatur von 35 °C und mehr führt.

```
# Temperatursensor DS1621 an I2C-Bus (ds1621_demo.c)
#include <stdio.h>           // Standard-Bibliotheken
#include <stdint.h>
#include <stdlib.h>
#include <linux/i2c-dev.h>   // i2c-Kerneltreiber
#include <fcntl.h>          // Dateizugriffssteuerung

// I2C-Adresse
#define ds1602_adr 0x48;

int8_t read_temp(void){

    const int i2c_address = ds1602_adr; // Chipadresse
    int device_ds1621; // Platzhalter fuer das geoeffnete Geraet
    int8_t temperatur; // Temperaturvariable

    const char *i2cDevice = "/dev/i2c-1";
    // Bei Raspberry Pi Rev. 1 : "/dev/i2c-0"

    // I2C-Interface oeffnen
    if((device_ds1621 = open(i2cDevice, O_RDWR)) < 0 ) {
        printf(" Das I2C-Interface konnte nicht geoeffnet
        werden! \n");
        exit(1);
    }

    // I2C-Bus-Adresse zuweisen
    if(ioctl(device_ds1621,I2C_SLAVE,i2c_address) < 0 ){
        printf(" Die Zuweisung der I2C-Bus-Adresse ist
        fehlgeschlagen! \n");

        exit(1);
    }

    i2c_smbus_write_byte_data(device_ds1621, 0xAC, 0x00);
    //DS1621 Konfiguration

    i2c_smbus_write_byte(device_ds1621, 0xEE);
    // Messung starten

    temperatur = i2c_smbus_read_byte_data(device_ds1621, 0xAA);
    // Temperatur lesen

    i2c_smbus_write_byte(device_ds1621, 0x22);
    // Messung beenden

    close(device_ds1621); // I2C-Interface schliessen

    return temperatur;
}
```

**ds1621\_demo.h:**

```
#ifndef DS1621_DEMO_H
#define DS1621_DEMO_H

int read_temp(void); // Funktion zum Auslesen der Temperatur

#endif
```

Das folgende Listing zeigt unter Verwendung von *ds1621\_demo.h* die Temperaturerfassung mit der Ausgabe auf dem Monitor und auf dem LCD. Hierfür wird die Bibliothek *i2c\_lcd\_drv\_2.h* eingebunden, die zahlreiche LCDs (HD44780-kompatibel) unterstützt. Nach Betätigung der Eingabetaste wird die Messung wiederholt und mit STRG+C beendet bzw. abgebrochen.

```
# Temperaturmessung (temp_test.c)

#include <stdio.h>           // Standard-Bibliotheken
#include <stdint.h>
#include <stdlib.h>
#include "i2c_lcd_drv_2.h" // LCD-Treiber (HD44780-kompatibel)
#include "ds1621_demo.h"   // DS1621-Funktionen

/*Hauptprogramm*/
int main(void)
{
    char str[10];           // String-Zwischenspeicher
    int8_t temp;           // Variable fuer Temperatur

    /*Endlosschleife*/
    while(1){
        /* Einlesen und Ausgabe der Temperatur vom DS1621*/
        temp = read_temp(); // Auslesen der Temperatur
        printf("Temperatur auslesen? - [Enter] druecken ...");
        getchar();
        printf("Die Temperatur betraegt: %d °C\n\n", temp);
        sprintf(str, "%d", temp); // Umwandlung in String

        init_i2c();
        // Initialisierung des PCF8574 zur Ansteuerung des LCDs
        lcd_init(); // Initialisierung des LCDs
        lcd_print("Temperatur: "); // Ausgabe auf LCD
        lcd_print(str);
        lcd_print(" C");
        i2c_close(); // Schliessen der PCF8574-i2c-Verbindung
    }
    return 0;
}
```