

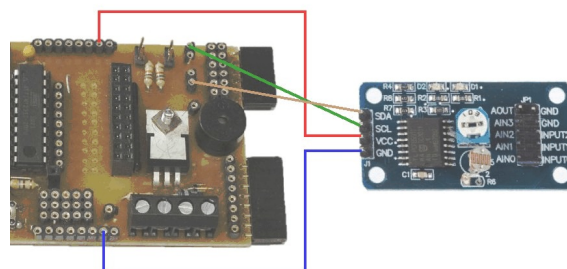
## 1. PCF-8574

Schreiben Sie für die folgenden Aufgaben Programme; Sie können sie anschließend den Kollegen vorstellen und testen.

- 1.1 Alle LEDs sollen gleichzeitig blinken; die An-Phase soll jeweils 1 Sekunde dauern, die Aus-Phase 100 ms. Beachten Sie dabei, dass die LEDs auf dem PCF-8574-Modul **nicht “gegen Masse”, sondern “gegen Vcc (5V)” geschaltet** sind.
- 1.2 Die LEDs sollen ein Lauflicht anzeigen; dies soll endlos wiederholt werden. Denken Sie sich etwas Hübsches aus!
- 1.3 Auf dem Terminal soll die **Schreibadresse** aller angeschlossenen I2C-Module angegeben werden.  
*Anleitung:* Adressieren Sie mit einer Schleife alle möglichen **geraden** Adressen; mit Hilfe der Systemvariable **err** können Sie überprüfen, ob die Adresse zu einem angeschlossenen I2C-Slave gehört.  
*Hinweis:* Solange wir nur einen einzigen Slave angeschlossen haben, erhalten wir auf dem Terminal natürlich auch nur eine einzige Adresse. Es gibt allerdings auch Module, auf denen mehrere verschiedene I2C-Slaves untergebracht sind, z.B. das Temperatursensor-Modul aus Kapitel 3.

## 2. PCF-8591

- 2.1 Schließen Sie das PCF-8591-Modul an:



Benutzen Sie das Programm aus 1.3, um die Schreibadresse des Moduls zu ermitteln.

Die Schreibadresse lautet: \_\_\_\_\_

Die Leseadresse lautet also: \_\_\_\_\_

- 2.2 Auf dem Modul befindet sich eine LED, welche (über einen Widerstand) an den Digital-Analog-Wandler des PCF-8591 angeschlossen ist. Auf dem Modul ist sie mit D2 beschriftet.  
Die Spannung am Ausgang des D/A-Wandlers soll in 25 Schritten von 0 bis  $U_{\max} \approx 5\text{ V}$  innerhalb von 2,5 Sekunden ansteigen und dann wieder auf 0 zurückspringen; dieser Vorgang soll endlos ablaufen.

Hinweis: Zur Vereinfachung können Sie hier und im Folgenden den Rohwert 250 für die Spannung 5 V benutzen; dann ist die Ausgangsspannung zwar in Wirklichkeit geringfügig kleiner als  $U_{\max}$ , aber dafür ist das Programmieren deutlich einfacher!

Spannungssignale, bei denen die Spannung immer wieder gleichmäßig ansteigt und schlagartig auf Null zurückspringt, bezeichnet man als **Rampensignal**.

- 2.3 Vielleicht erkennen Sie bei der Beobachtung der LED noch feine Helligkeitssprünge. Wenn Sie mögen, können Sie dann die Schrittweite bei 2.2 verringern, bis diese Sprünge nicht zu erkennen sind.
- 2.4 Ermitteln Sie nun, bei welcher Spannung des D/A-Wandlers die LED anfängt zu leuchten. Ändern Sie dazu das Programm aus 2.2 oder 2.3 so ab, dass sie es mit dem Taster Ta0 anhalten können. Es empfiehlt sich, nun die Spannung etwas langsamer ansteigen zu lassen.

Halten Sie das Rampensignal mit dem Taster an, wenn die LED anfängt zu leuchten; messen Sie anschließend die Spannung am D/A-Wandler.

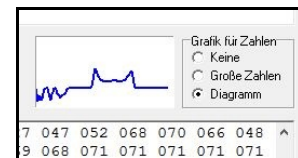
*Hinweis:* Zur Messung der Spannung halten Sie das rote Kabel an den Anschluss Aout (auf dem Modul) und das schwarze Kabel an GND (auf der Attiny-Platine).

- 2.5 Nun sollen mit dem A/D-Wandler des PCF-8591 Spannungen am Poti, am LDR und am NTC gemessen werden.
- 2.5.1 Jede Sekunde soll eine einzige Messung der Spannung am Poti (Kontrollbyte: 0) vorgenommen werden. Dabei soll jedes Mal das gesamte Protokoll von der Initialisierung bis zum Stopp-Signal durchlaufen werden. Der erhaltene Rohwert soll direkt, d. h. ohne Umrechnung in einen echten Spannungswert, an das Terminal gesendet werden.

Erstellen Sie das zugehörige Programm und laden Sie es auf den Mikrocontroller. Aktivieren Sie anschließend das Terminal und starten Sie den Empfang. Verstellen Sie dann mit einem Schraubendreher die Stellung des Potentiometers und beobachten Sie die empfangenen Rohwerte; sie sollten alle Zahlen zwischen 0 und 255 erhalten können.

- 2.5.2 Ändern Sie Ihr Programm aus 2.5.1 folgendermaßen ab: Initialisierung, Adressierung usw. sollen diesmal nur **ein einziges Mal** vorgenommen werden. Dann sollen 200 Messungen im Zeitabstand von jeweils 50 ms erfolgen. Achten Sie darauf, dass die **letzte Messung mit ACK** vorgenommen werden sollte.
- 2.5.3 Ändern Sie Ihr Programm aus 2.5.2 folgendermaßen ab: Jetzt soll nicht die Spannung am Poti, sondern die am LDR gemessen werden. Ermitteln Sie zuvor selbst mit Hilfe der Folien aus dem Handout das passende Kontrollbyte.
- 2.5.4 Testen Sie zunächst Ihr Programm mit einer Taschenlampe aus. Die empfangenen Rohwerte sollten sich mit der Lichtintensität ändern: Je heller das Licht, desto kleiner sollten die Rohwerte sein.

Wenn Sie mögen, können Sie sich die empfangenen Werte auch in Form eines kleinen Diagramms anzeigen lassen: Aktivieren Sie dazu im Terminal die Option “Grafik für Zahlen - Diagramm”.



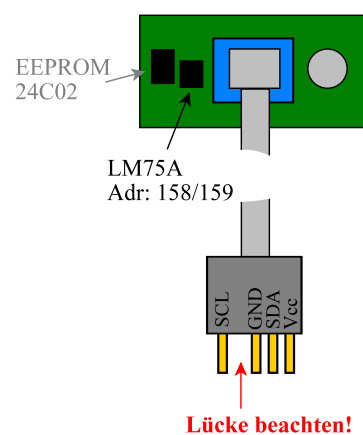
- 2.6 **Fakultativ:** Jetzt wird es anspruchsvoller! Programmieren Sie einen automatischen 3-Stufen-Dimmer: Je nach Helligkeit soll die LED (D2) auf dem Modul heller oder dunkler werden.
- 2.7 **Fakultativ:** Mit einer kleinen Änderung im Programm von 2.5.3 können Sie nun auch den NTC austesten. Wie verhalten sich die Messwerte, wenn der NTC mit dem Finger (oder einem Fön®) erwärmt wird?

### 3. LM75

Das Kombi-Modul aus der rechten Abbildung besitzt zwei I2C-Slaves: den Temperatursensor LM75A und das EEPROM 24C02. Wir werden hier nur mit dem Temperatursensor arbeiten.

Im Gegensatz zu dem PCF8591 kommen wir beim LM75A ohne ein Kontrollbyte aus. Am einfachsten geht man folgendermaßen vor:

Nach der Initialisierung sendet man direkt die **Leseadresse 159**. Dann liest man genau zwei Bytes; dabei muss das erste mit ACK, das zweite mit NACK quittiert werden. Anschließend sendet man ein Stopp-Signal. Für die nächsten Messungen verfährt man genauso.



Das erste Byte gibt die Temperatur an, genauer gesagt: die entsprechenden Ziffern vor dem Komma. Im zweiten Byte stehen Informationen zum Vorzeichen und zu Nachkommastellen; diese Informationen wollen wir hier der Einfachheit halber aber nicht benutzen. Im Manual des Herstellers findet man dazu genauere Informationen.

 **Wichtig** 

1. Auch wenn wir nur das erste Byte benutzen, muss das 2. Byte trotzdem gelesen werden; ansonsten kann der Baustein nicht korrekt arbeiten.
  2. Die **Konversionszeit** des LM75A kann bis zu 300 ms betragen. Deswegen sollte zwischen zwei aufeinander folgenden Messungen mindestens eine Zeit von 300 ms verstreichen.
- 3.1 Schließen Sie das Kombi-Modul wie in der Abbildung dargestellt an die Attiny-Platine an. Da auf dem Modul keine Pull-Up-Widerstände vorhanden sind, **müssen hier unbedingt die Pull-Up-Widerstände auf der Attiny-Platine mit Jumpfern aktiviert werden!**

Kontrollieren Sie mit dem Programm aus 1.3 nach, dass auf dem Modul tatsächlich zwei I2C-Slaves vorhanden sind.

- 3.2 Schreiben Sie ein Programm, welches die Temperatur (in °C) jede Sekunde auf dem LCD anzeigt.

**Hinweis:** Das LCD benutzt **nicht** die Anschlüsse B.7 (SCL) und B.5 (SDA)!

**Für die folgenden Anwendungs-Aufgaben gibt es weniger Hilfestellung. Suchen Sie sich eine Aufgabe aus...**

### 3.3 Hitzefrei-Alarm

Der Attiny soll einen optischen Alarm (Dauerlicht oder Blinklicht) auslösen, wenn die Raumtemperatur 25 °C überschreitet. Der Alarm soll erst wieder ausgehen, wenn die Temperatur entsprechend abgesunken ist.

**Nur für Freaks:** Neben dem optischen soll auch ein akustischer Alarm ausgelöst werden. Dieser Alarm soll allerdings nur 5 Sekunden dauern!

**Zur Erinnerung:** Durch Ein- und Ausschalten wird im Beeper eine Membran bewegt. Wird die Membran sehr rasch hin und her bewegt, so hört man einen Ton. Der Ton ist um so höher, je schneller die Membran sich bewegt.

### 3.4 Temperaturwerte loggen und mit Tabellenkalkulationsprogramm darstellen

Im Terminal sollen die Temperaturen im Sekundenrhythmus angezeigt werden, während der LM75 mit einer Infrarotlampe oder einem Fön® erwärmt wird. Die Messreihe soll beendet werden, wenn die Temperatur (nahezu) gleich bleibt (Im Terminal die Schaltfläche “Empfang stoppen” betätigen).

Zur Auswertung der Daten mit EXCEL gehen Sie dann bitte folgendermaßen vor:

*Terminal:*

- Schaltfläche “Anzeige speichern” betätigen
- Spaltenanzahl: 1 eingeben
- OK
- Dateinamen eingeben
- OK

*Tabellenkalkulation:*

- Datei mit EXCEL öffnen, ggf. Dateischutz aufheben
- Datei in einem neueren Format speichern
- Neue Datei öffnen
- Messdaten markieren
- Diagramm erstellen mit “Einfügen - Diagramme - Linie”

## 4. Interrupts

- 4.1 Eine LED bei PortB.0 soll fortwährend blinken. Wenn der Taster Ta0 betätigt wird, dann soll die Blinkzeit um 10 ms erhöht werden; wenn der Taster Ta1 betätigt wird, dann soll die Blinkzeit um 10 ms verringert werden. Achten Sie bei der Programmierung darauf, dass die Blinkzeit den Wert 10 nicht unterschreitet und den Wert 250 nicht überschreitet.

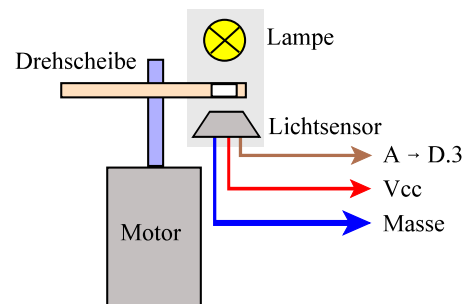
*Hinweis:* Durch den Taster Ta1 wird das Interruptereignis “INT1” ausgelöst. Es wird genauso behandelt wie das Ereignis “INT0” (also hierbei einfach 0 durch 1 ersetzen).

- 4.2 **Fakultativ:** Wie 4.1, jedoch soll das Blinken einer einzigen LED durch ein Lauflicht ersetzt werden.
- 4.3 Durch Betätigen des Tasters Ta0 soll eine LED bei PortB.0 “getoggelt” (d. h. umgeschaltet) werden.
- 4.4 Wenn Sie in der Aufg. 4.3 den Taster Ta0 durch Ta1 ersetzen, kann es geschehen, dass bei der Betätigung des Tasters Ta1 die LED **nicht** ihren Zustand wechselt. Testen Sie

dies aus. Geben Sie eine Erklärung dafür!

#### 4.5 Drehzahlmesser

Die Drehzahl eines Motors (genauer: die Anzahl der Umdrehungen eines Motors pro Minute) kann man mit Hilfe einer auf die Motorachse gesteckten Drehscheibe bestimmen. Die Drehscheibe besitzt in der Nähe des Randes ein Loch. Dieses Loch lässt bei jeder Umdrehung für einen kurzen Augenblick das Licht einer Lampe auf den darunter befindlichen Sensor fallen (Lichtschranke). Fällt Licht auf den Sensor, liegt an seinem Ausgang A ein 0-Signal an, sonst ein 1-Signal.



- 4.5.1 Schreiben Sie ein Programm, welches die Drehzahl des Motors (pro Minute) auf einem LCD ausgibt. Benutzen Sie eine Messzeit von 6 Sekunden.

*Hinweis:* Wenn Sie INT1 benutzen, ist das Programm erstaunlich einfach!

- 4.5.2 Testen Sie Ihr Programm aus: Schließen Sie dazu die 3 Anschlüsse der Lichtschranke an Ihre Attiny-Platine an und starten Sie den Motor. Warten Sie kurz, bis der Motor seine Endgeschwindigkeit erreicht hat. Starten Sie dann Ihr Programm, indem Sie den Reset-Taster betätigen...