

## **Dokumentation Analog-MF-Modul für Schalttafeleinbau (KTY/LED-Temperaturmodul)**

Das **Analog-MultiFunktionsModul** stellt folgende Funktionen zur Verfügung:

- 16 Analog-Eingangskanäle zur Signalerfassung über
  - KTY-Temperatursensor für Temperaturmessung von -20°C bis 100 °C
    - Raumtemperatur
    - Außentemperatur
    - Heizungstemperatur
    - Wassertemperatur
  - LDR-Widerstände zur Erfassung der Sonneneinstrahlung (Rollosteuerung)
  - Potentiometer für Raumtemperaturvorgabe
- Option: Ansteuerung von 8 extern montierten Leuchtdioden zur
  - Zustandsanzeige für Raumheizung (Ein/Aus)
- Option: Datenspeicherung von Parameterdaten auf EEPROM
  - Parameterdaten stehen auch nach Reset oder Neustart der Steuerung zur Verfügung
  - Datenübertragung über I2C-Bus
- Option: Echtzeituhr mit Akkupufferung
  - Aktuelle Uhrzeit nach Reset und Neustart der Steuerung
  - Datenübertragung über I2C-Bus

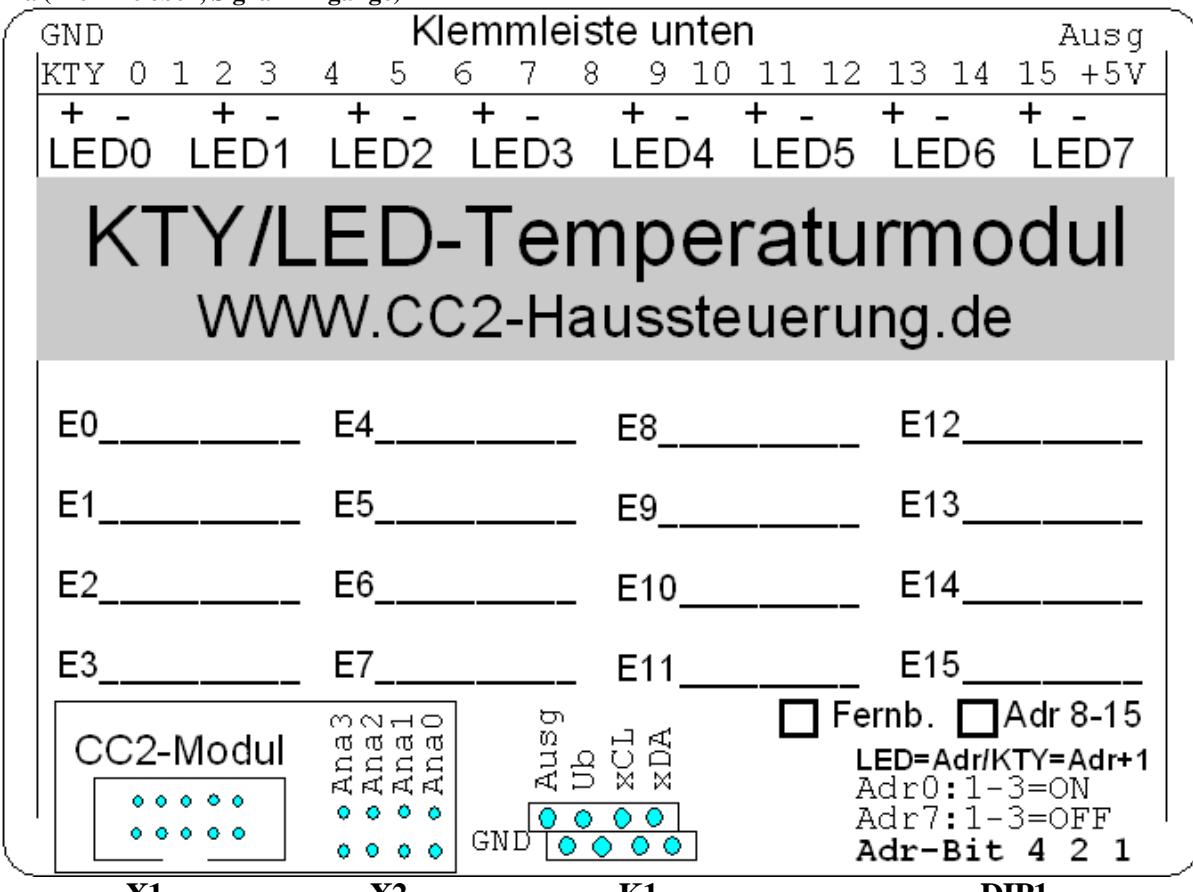
### **Technische Daten:**

Anzahl Analog-Eingänge:	16
AD-Wandler	es wird der AD-Wandler der CC2 benutzt, dieser hat eine Auflösung von 10 Bit
Modul-Versorgungsspannung	wahlweise 9 - 24V
Logikspannung modulintern	5V, werden im Modul aus der Versorgungsspannung erzeugt
I2C-Bus	I2C-Bus mit/ohne Reichweitenverlängerung an X1: SDA und SCL an K1: xDA und xCL ohne Fernbus-Option: xDA entspricht SDA xCL entspricht SCL mit Fernbus-Option: xDA entspricht LDA xCL entspricht LCL
Modulbreite	4 Teilungseinheiten (3 Module pro Zeile in der Elektroverteilung)
Anzahl Module am I2C-Bus (Anzahl der verfügbaren Adressen)	16
Kanalwahl Multiplexer	Ausgabe an PCF8574A / PCF8574AP Ausgabewerte: 0 ...15
Moduladresse	über DIP-Schalter wählbar: 0...7    bei Bestückung mit PCF8574A, 8...15   bei Bestückung mit PCF8574AP

## Modulanschluß

**K2b** (Klemme unten, GND-Signal)

**K2a** (Klemme oben, Signal-Eingänge)



### Anschluß an CC2-Basismodul

Verbindung erfolgt über Steckverbindung X1 (Flachbundleitung mit 10-poligem Pfostenstecker).

Weitere Leitungsverbindungen zur CC2 sind nicht erforderlich.

An X2 wird über einen Jumper festgelegt, mit welchem Analog-Eingang der CC2 das Ausgangssignal des Multiplexers gemessen werden kann.

### Anschluß an CC2-Station

Der Anschluß an die CC2-Station erfolgt über die Klemmleiste K1. Da die Station keine Fernbusoption besitzt, ist diese auch am Analog-MF-Modul nicht erforderlich. Die Leitungslänge beim Anschluss ohne Fernbusoption sollte 2-3 Meter nicht überschreiten.

Klemmenbezeichnung an K1 Analog-MF-Modul		Klemmenbezeichnung / Kl.-Nr. Station
xDA	nach	I <sup>2</sup> C SDA / Klemme 8
xCL	nach	I <sup>2</sup> C SCL / Klemme 9
Ub	nach	+12V out / Klemme 16
Ausgang	nach	wahlweise AD0 ... AD6 / Klemme 28 ... 34 z.B. Klemme 7 / 15 / 17 / 35 / 38 / 40 / 54
GND (K1 Reihe unten)	nach	

## Fühleranschluss

Die Sensoren wie z.B. KTY-Temperaturfühler, Lichtsensoren usw. werden an die Klemmleiste K2 an die Klemmen **a** und **b** angeschlossen.

Der ohmsche Widerstand des angeschlossenen Sensors sollte sich zwischen 1200 und 3400 Ohm bewegen.  
Das entspricht beim Fühler vom Typ KTY81-210 ungefähr einem Temperaturbereich von –25 bis 100 °C.

## Einstellung der Moduladressen (DIP-Schalter)

Dip-Schalter			Moduladressen	
	2 <sup>2</sup>	2 <sup>1</sup>	2 <sup>0</sup>	
off	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	PCF8574A: Adresse 0 PCF8574AP: Adresse 8
on	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
off	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	PCF8574A: Adresse 7 PCF8574AP: Adresse 15
on	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

## Funktionsweise der Analogeinheit

Die prinzipielle Funktionsweise des Temperaturmoduls ist folgendermaßen:

Über den I2C-Bus wird an den im Modul eingebauten Portbaustein PCF8574(A)P ein Wert zwischen 0 und 15 gesendet. Dabei ist bei der Adressierung die am DIP-Schalter eingestellte Adresse zu beachten. Dies kann der Adressbereich von 0-7 beim PCF8574P, und der Adressbereich 8-15 beim PCF8574AP sein.

Der an die eingestellte Adresse übertragene Wert stellt den gewünschten Multiplexerkanal des Analogschalters ein. Der gewünschte Fühler wird aktiviert. Das Fühlersignal wird auf den eingebauten Operationsverstärker geschaltet, verstärkt und geglättet. Dieses aufbereitete Signal liegt jetzt am Analog-Eingang der CC2 oder der CC2-Station an und kann gelesen (digitalisiert) werden.

## Software der Analogeinheit

Für den Betrieb der Analogeinheit wird das Modul **anaein16.c2** benötigt.

Mit der folgendem Funktionsaufruf wird die Grundinitialisierung ausgeführt:

```
Init ( int i2C_AdresseMultiplexer )
```

Als Parameter wird die Adresse des PCF8574-Bausteins übergeben, mit dem die Adressierung der Analog-Kanäle (Multiplexerkanal) erfolgt.

Danach wird folgende Funktion, die den kontinuierliche Messvorgang in einem separaten Thread startet, aufgerufen:

```
StartMessung ( )
```

Der Mess-Thread hat folgenden Namen:

```
thread Messen_AnaEin16_Werte
```

Hier werden nun maximal 4 Analogmodule mit insgesamt max. 64 angeschlossenen Sensoren kontinuierlich abgefragt. Die Anzahl der möglichen Analogmodule wird durch die Anzahl der freien Analog-Eingänge der CC2 begrenzt.

Jedem Analog-Eingang kann nun ein Fühlertyp zugewiesen werden, damit der Analogwert für den Fühlertyp korrekt berechnet werden kann. Dies erfolgt in der Funktion

```
function InitAnalogKanalTyp ()
```

Ein möglicher Fühleroffset wird korrigiert, indem in folgender Funktion der Offset angegeben wird:

```
function InitFuehlerOffset ()
```

In folgenden Datenfeldern liegen nun die Rohdaten der gemessenen Werte:

```
int PA_AnaEing [ANZ_ANALOGPORTS]; // Prozessabbild der Analogkanäle
```

Nach der Berechnung der Fühlerdaten unter Berücksichtigung von Fühlertyp und Offset werden die Ergebnisse in folgendem Datenfeld abgelegt:

```
int PA_AnaBerechnet [ANZ_ANALOGPORTS]; // Fühlertyp abhängig
                                         // berechnete Werte der Analogkanäle
```

Temperaturen liegen hier in 10tel-Grad vor. So kann man ohne Gleitpunktoperationen eine Auflösung von 0,1 Grad erreichen.

Mit diesen berechneten Werten kann man nun im Programm arbeiten.

## Option EEPROM

Das optionale EEPROM ist ein nicht flüchtiger Speicher und kann für die Speicherung von Parameterdaten verwendet werden. Auf der im Gehäuse optional eingebauten Zusatzplatine kann mit Hilfe eines DIP-Schalters die Adresse des EEPROMs eingestellt werden. Die voreingestellte Adresse ist 0. Für eine andere Adresse muss das Gehäuse geöffnet werden. Dies ist in der Regel aber nicht erforderlich.

Die Standard-EEPROMgröße ist 8 KB.

Beispiel: Ansteuerung EEPROM zur Speicherung einer Versionsnummer

```
const EE_ADR_EEPROM=0;      // Adressbits: A0 - A2 = GND, Grundadresse 160 wird
                           // bei neuem EEPROM-Modul intern verarbeitet

//-----
// im Modul eeprom.c2 setzen:
// const Pagewrite = 32; // Puffergröße des EEPROMs in Byte; siehe EEPROM-Datenblatt

function SpeichernEE_Version (int speicheroffset, byte versionsnummer)
//-----
// Diese Funktion zeigt wie in das EEPROM ein Byte geschrieben wird
// Parameter:
//         speicheroffset= Adresse an die die Versionsnummer geschrieben wird
//         versionsnummer= zu speichernde Daten
//-----
{
    // capture mylib.I2C_Sperren; // Vermeidung von Kollisionen auf dem Bus

    eeprom.writebyte(EE_ADR_EEPROM, speicheroffset, versionsnummer);           // byte 0
```

```
release;

}

function LadenEE_Version (int speicheroffset ) returns byte
//-----
// Diese Funktion zeigt wie in das EEPROM ein Byte geschrieben wird
// Parameter:
//           speicheroffset= Adresse von der die Versionsnummer gelesen wird
//           versionsnummer= zu lesende Daten
//-----
{
    byte versionsnummer;

    // capture mylib.I2C_Sperren; // Vermeidung von Kollisionen auf dem Bus

    versionsnummer = eeprom.readbyte(EE_ADR_EEPROM, speicheroffset);           // byte 0

    // release;

    return versionsnummer;
}
```

## Option Echtzeituhr

Die eingebaute Echtzeituhr mit Datumsfunktion ist in der Lage auch einen Stromausfall über längere Zeit zu überbrücken, ohne dass die eingestellte Zeit verloren geht.

Ist an der CC2-Steuerung eine DCF77-Uhr angeschlossen, wird die Echtzeituhr synchronisiert, wenn eine gültige Funk-Uhrzeit erkannt wurde.

### Initialisierung und Adressierung der Echtzeituhr

Für die Adressierung der Echtzeituhr stehen am IC 2 mögliche I2C-Adressen zur Verfügung. Auf der Platine ist die Adresse 1 fest verdrahtet. Es ist darauf zu achten, dass die EEPROM-Adresse nicht mit der Echtzeituhr-Adresse kollidiert.

```
const PCF8583_UHR_ADR = 1; // Wertebereich: 0,1
const PCF_WATCHDOG_TIME = 0; // =0: Watchdog von PCF-Uhr nicht aktivieren
```

```
// (byte A0, byte setCC2clock, byte autosync, byte watchdog, byte enableHostKey)
pcf8583.init(PCF8583_UHR_ADR, 1, 0,PCF_WATCHDOG_TIME, 1));
```

Dieser Funktionsaufruf synchronisiert die CC2-Zeit mit der Zeit der Echtzeituhr. Darum sollte beim Start der CC2 diese Funktion vor der Abarbeitung zeitabhängiger Funktionen stehen.

### Synchronisierung der Echtzeituhr

Mit folgender Funktion bekommt die Echtzeituhr die Systemzeit der CC2:

```
pcf8583.syncpcf();
```

Die CC2 muss hierbei eine gültige Zeit haben. Die gültige Zeit kommt entweder von der DCF-Uhr, andernfalls muss sie manuell eingegeben werden.



## Analog-MF-Modul V2.0

---

Anfragen an:

Michael Gierschner  
Kuhlkamp 22

31275 Lehrte

Tel: 05132 / 83 60 28  
Fax: 05132 / 58 44 23  
<mailto:Kontakt@M-Gierschner.de>

Lehrte, 04.07.2005