



## **DOKUMENTATION SCHWABENPLAN® MC-PROTOKOLL**

### **1. EINLEITUNG**

Das Schwabenplan® MC-Protokoll wurde entwickelt um jede Hauptplatine aus unserem Lieferprogramm mit einer Software zu verbinden. Die Programmiersprache für die Anwendung muss nicht zwingend beachtet werden. Wir verwenden üblicherweise Visual Basic (Version 2010 und höher). Es können aber auch andere Entwicklungsumgebungen zum Einsatz kommen.

Die Hauptplatine wird über eine RS-232 bzw. über eine RS-485 Schnittstelle angebunden. Die Konfiguration Schnittstelle ist frei definierbar. Unsere Programmbeispiele basieren auf folgenden Einstellungen:

**Baud-Rate:** 19200 / 115200 Baud

**Datenbits:** 8

**Stopbits:** 1

**Parität:** Keine

#### **1.1 INTEGRATION**

Für die Integration dieses Übertragungsprotokolls stellen wir Ihnen sehr gerne unser Demo-Projekt „MC Agent“ zur Verfügung. Der MC Agent wurde mit Visual Basic 2010 programmiert. Für den Microcontroller haben wir Beispiele in der Programmiersprache C für den PIC18F452, den PIC18F46K80 sowie den PIC18F97J60. Sprechen Sie uns im Bedarfsfall einfach an.

### **2. ALLGEMEINES**

Das Protokoll ist grundsätzlich in drei Kategorien aufgeteilt. Es gibt:

✓ **Status-Befehle**

Über die Status-Befehle kann die Peripherie der gesamten Hard- und Firmware abgefragt werden.

✓ **PORT-Befehle**

Mit diesen Befehlen kann jeder beliebige PORT-Zustand gelesen und gesteuert werden.

✓ **I<sup>2</sup>C-Bus Befehle**

Über die I<sup>2</sup>C-Befehle können eine Vielzahl von I<sup>2</sup>C-Bausteinen direkt abgefragt und gesteuert werden.



### 3. BEFEHLSAUFBAU

Die Befehle sind sehr einfach gehalten und bestehen immer aus wenigen Bytes. Es gibt in diesem Protokoll Lese- und Schreibbefehle. Für die Lesebefehle werden nur drei Bytes benötigt. Daher ist der Inhalt des Bytes [VALUE] nicht relevant. Es wird von unserer Seite empfohlen das Byte [VALUE] mit dem Steuerbefehl [CMD] zu beschreiben.

Als erstes muss dem Microcontroller mitgeteilt werden, dass ein Datensatz übertragen werden soll. Dieser Wakeup-Befehl wird mit dem ASCII-Zeichen „S“ -> 0x53 durchgeführt. Danach muss der Sender 100 ms warten. In dieser Zeit stellt sich der Empfänger auf ein Datentelegramm ein. Danach wird der zu sendende Befehlssatz an den Microcontroller gesendet. Zum Abschluss der Übermittlung müssen die Zeichen „\0“ gesendet werden. Über diese Zeichen wird der Microcontroller über das Ende der Datenübertragung informiert.

#### Beispiele:

Befehlsaufbau: [STX][CMD][VALUE][ETX][\0]

#### Beispiel Nr. 1 - Auslesen des Microcontroller-Typ (STATUS-Befehl)

[0x53] -> 100ms warten -> [0x02][0x20][0x20][0x03][\0]

#### Beispiel Nr. 2 - Auslesen des Analogwertes AN0 (PORT-Befehl)

[0x53] -> 100ms warten -> [0x02][0xA0][0xA0][0x03][\0]

#### Beispiel Nr. 3 - PORTB mit dem Wert 0xFF steuern (PORT-Befehl)

[0x53] -> 100ms warten -> [0x02][0xB0][0xFF][0x03][\0]

#### Beispiel Nr. 4 - PORTB Auslesen (PORT-Befehl)

[0x53] -> 100ms warten -> [0x02][0xB1][0xB1][0x03][\0]

#### Beispiel Nr. 4 – Temperaturbaustein mit Adresse Nr. 1 Auslesen (I<sup>2</sup>C-Befehl)

[0x53] -> 100ms warten -> [0x02][0x41][0x41][0x03][\0]

#### Beispiel Nr. 5 – Uhrzeit lesen (I<sup>2</sup>C-Befehl)

[0x53] -> 100ms warten -> [0x02][0x31][0x31][0x03][\0]

#### Beispiel Nr. 6 – Minuten schreiben - hier im Beispiel 00 Minuten (I<sup>2</sup>C-Befehl)

[0x53] -> 100ms warten -> [0x02][0x51][0x00][0x03][\0]



### 3.1 Rückgabewerte

Die Rückgabewerte der einzelnen Befehle sind Strings bzw. 4 stellige Integer Werte. Der Microcontroller sendet zum Abschluss seiner Datenübertragung immer den ASCII-Code „\n\r“. -> z.B. printf("\n\r");

### 3.2 Protokoll Prüfung

Der Microcontroller prüft als erstes jeden empfangenen Befehl auf seine Plausibilität ab. Er kontrolliert hierbei die Summe der eingegangenen Bytes und die Positionierung der Steuerzeichen [STX] und [ETX] ab. Ist dies nicht der Fall, so antwortet der Microcontroller er mit dem String "**ERROR**".

### 3.3 Unbekannte Befehle

Der Microcontroller wertet als nächstes das Steuerkommando und das Datenbyte aus. Ist dem Microcontroller ein Befehl unbekannt, so antwortet er mit dem String "**NO COMMAND**".

## 4.1 STATUS BEFEHLE

Inzwischen haben die meisten Microcontroller einen großen Speicher. Daher werden in Ihnen zum Teil einige Zusatzinformationen wie Microcontroller-Typ, Hardware-Version usw. hinterlegt.

Befehl	Bezeichnung	Datenbyte	Rückgabewert
0x20	Microcontroller-Typ lesen	-	String
0x21	Hardware-Version lesen	-	String
0x22	Taktfrequenz	-	String
0x23	Software-Version lesen	-	String
0x24	Hex-File lesen	-	String
0x25	Erstelldatum lesen	-	String
0x26	Bootloader lesen	-	String
0x27	Debug-Mode lesen	-	String
0x28	Protokoll Version lesen	-	String
0x29	Interface on Board auslesen	-	String

**4.2 PORT BEFEHLE**

Befehl	Bezeichnung	Datenbyte	Rückgabewert
0xA0	A/D-Wert Kanal Nr. 0	-	Integer
0xA1	A/D-Wert Kanal Nr. 1	-	Integer
0xA2	A/D-Wert Kanal Nr. 2	-	Integer
0xA3	A/D-Wert Kanal Nr. 3	-	Integer
0xA4	A/D-Wert Kanal Nr. 4	-	Integer
0xA5	A/D-Wert Kanal Nr. 5	-	Integer
0xA6	A/D-Wert Kanal Nr. 6	-	Integer
0xA7	A/D-Wert Kanal Nr. 7	-	Integer
0xA8	A/D-Wert Kanal Nr. 8	-	Integer
0xA9	A/D-Wert Kanal Nr. 9	-	Integer
0xAA	A/D-Wert Kanal Nr. 10	-	Integer
0xAB	A/D-Wert Kanal Nr. 11	-	Integer
0xAC	A/D-Wert Kanal Nr. 12	-	Integer
0xAD	A/D-Wert Kanal Nr. 13	-	Integer
0xAE	A/D-Wert Kanal Nr. 14	-	Integer
0xAF	A/D-Wert Kanal Nr. 15	-	Integer
0xF8	Portzustand PORTA schreiben	Char	Integer
0xF9	Portzustand PORTA lesen	-	Integer
0xB0	Portzustand PORTB schreiben	Char	Integer
0xB1	Portzustand PORTB lesen	-	Integer
0xC0	Portzustand PORTC schreiben	Char	Integer
0xC1	Portzustand PORTC lesen	-	Integer
0xD0	Portzustand PORTD schreiben	Char	Integer
0xD1	Portzustand PORTD lesen	-	Integer
0xE0	Portzustand PORTE schreiben	Char	Integer
0xE1	Portzustand PORTE lesen	-	Integer
0xF0	Portzustand PORTF schreiben	Char	Integer
0xF1	Portzustand PORTF lesen	-	Integer
0xF2	Portzustand PORTG schreiben	Char	Integer
0xF3	Portzustand PORTG lesen	-	Integer
0xF4	Portzustand PORTH schreiben	Char	Integer
0xF5	Portzustand PORTH lesen	-	Integer
0xF6	Portzustand PORTJ schreiben	Char	Integer
0xF7	Portzustand PORTJ lesen	-	Integer



### 4.3 I<sup>2</sup>C-Bus BEFEHLE

Befehl	Bezeichnung	Datenbyte	Rückgabewert
0x30	Datum lesen	-	String
0x31	Uhrzeit lesen	-	String
0x32	Stunden lesen	-	Integer
0x33	Minuten lesen	-	Integer
0x34	Sekunden lesen	-	Integer
0x35	Tag lesen	-	Integer
0x36	Monat lesen	-	Integer
0x37	Jahr lesen	-	Integer
0x38	Wochentag lesen	-	String
0x39	Typ RTC-Baustein lesen	-	String
0x40	Temperatur lesen Adr. 0 lesen	-	Integer
0x41	Temperatur lesen Adr. 1 lesen	-	Integer
0x42	Temperatur lesen Adr. 2 lesen	-	Integer
0x43	Temperatur lesen Adr. 3 lesen	-	Integer
0x44	Temperatur lesen Adr. 4 lesen	-	Integer
0x45	Temperatur lesen Adr. 5 lesen	-	Integer
0x46	Temperatur lesen Adr. 6 lesen	-	Integer
0x47	Temperatur lesen Adr. 7 lesen	-	Integer
0x48	Typ Temperatur-Baustein lesen	-	String
0x50	Stunden schreiben	Char	Integer
0x51	Minuten schreiben	Char	Integer
0x52	Sekunden schreiben	Char	Integer
0x53	Tag schreiben	Char	Integer
0x54	Monat schreiben	Char	Integer
0x55	Jahr schreiben	Char	Integer
0x56	Wochentag schreiben	Char	Integer