

Softwarebeschreibung E/A-Modul "IP65"
Version 1.1

VORLÄUFIG

(preliminary - sorry, there is no English translation of this document yet)

Inhaltsverzeichnis

1.	VERSIONSTAND DIESES DOKUMENTS	3
2.	VORWORT	3
3.	CANOPEN	4
3.1	OBJEKTVERZEICHNIS	4
3.2	PROZEßDATENOBJEKTE (PDO).....	4
3.2.1	<i>Knotennummer-abhängige, automatische Umschaltung der PDO-CAN-Identifizier</i>	6
4.	OPTIONALE FUNKTIONEN	7
4.1	FREQUENZMESSER.....	7
5.	TESTFUNKTIONEN	9
5.1	SYSTEMTEST PER DIAGNOSESCHNITTSTELLE	9

1. Versionstand dieses Dokuments

Versionsnummer	Datum (ISO8601)	Autor	Hinweise, Änderungen
V1.0	2005-01-26	W.Büscher	Ersterstellung (deutsch)
V1.1	2005-07-15	W.Büscher	Beschreibung der 2 neuen Frequenzgänge

2. Vorwort

Dieses Dokument beschreibt die Softwarefunktionen des E/A-Moduls "IP65" von MKT Systemtechnik, inklusive verschiedener kundenspezifischer Sondervarianten, z.B. mit Frequenzzähler (zusätzlich zur normalen Funktion als E/A-Modul).

Zur Kommunikation auf der Feldebene dient der CAN-Bus.

Das Kommunikationsprotokoll ist kompatibel zu CANopen DS301V4.0 .

Das Geräteprofil entspricht weitgehend CANopen DS401, mit einigen herstellerspezifischen Erweiterungen.

Wesentliche Eigenschaften der Firmware :

- CANopen mit speziellen Erweiterungen (vgl. Kapitel 3.2.1)
- CANopen-Protokollstack der Firma SYS TEC electronic GmbH
- drei Empfangs- und drei Sende-PDOs (Prozeßdatenobjekte)
- variables PDO-Mapping, Defaulteinstellung siehe Kapitel 3.2 .
- Synchroner oder asynchroner PDO-Übertragung, periodisch oder ereignisgesteuert
- ein SDO (Servicedatenobjekt) für die Parametrierung per CAN
- Heartbeat, Emergency Messages
- per CAN parametrierbare SYNC-Sende-Funktion

Weitere Informationen :

- Anschlußbelegung, technische Daten, elektrische Spezifikation entnehmen Sie bitte der Hardwarebeschreibung, Dokument Nummer 85265. Erhältlich bei www.mkt-sys.de .
- Eine von CANopen-Konfigurationstools einlesbare elektronische Gerätebeschreibung (EDS-File) finden Sie ebenfalls auf der Webseite von MKT Systemtechnik.
- Details zu den CANopen-Übertragungsprotokollen finden Sie bei CiA (CAN in Automation), <http://www.can-cia.org/canopen/>

3. CANopen

3.1 Objektverzeichnis

Das Objektverzeichnis entspricht weitgehend dem CANopen-Geräteprofil für ein Ein/Ausgabemodul nach DS401. Darin sind verschiedene standardisierte Objekte enthalten, mit denen die digitalen und analogen Ein- und Ausgänge abgefragt bzw. angesteuert werden. Die Einträge im Objektverzeichnis werden üblicherweise durch einen hexadezimal notierten Index und Subindex beschrieben, z.B. 0x6000.01 = Objekt 6000h Subindex 01h = digitale Eingänge 1 bis 8. Der Aufbau des Objektverzeichnisses ist einer EDS-Datei beschrieben, die von einem CANopen-Konfigurationstool (kein MKT-Produkt!) eingelesen werden kann.

Zu den vorhandenen Standard-Objekten zählen:

- 6000 Digitale Eingänge in 8-Bit-Gruppen
- 6200 Digitale Ausgänge in 8-Bit-Gruppen
- 6401 Analoge Eingänge mit maximal 16 Bit Auflösung
- 6411 Analoge Ausgänge mit maximal 16 Bit Auflösung

Zusätzlich existieren weitere Objekte für die in Kapitel 4 beschriebenen optionalen Funktionen. Dazu zählen:

- 5C01 Frequenzzähler (Kapitel 4.1)

3.2 Prozeßdatenobjekte (PDO)

Die Prozeßdatenobjekte dienen zur schnellen Übertragung von Prozeßdaten per CAN-Bus. Das Gerät verfügt über drei Empfangs- und drei Sende-Prozeßdatenobjekte, deren Inhalt und Übertragungsart individuell konfiguriert werden kann. Sie benötigen dazu ein CANopen-Konfigurationstool, welches per Parameterkanal (SDO, d.h. per CAN) auf das E/A-Modul zugreifen kann. Falls kein CANopen-Konfigurationstool zur Verfügung steht, müssen die folgenden Default-Einstellungen verwendet werden, die sich weitgehend am CANopen-Standard DS401 orientieren :

1400 Kommunikationsparameter für 1. Empfangs-PDO :

CAN-Identifizier: 0x200 + Knotennummer¹
Übertragungsart: 0xFE
Inhibit-Time : 0 (keine)

1401 Kommunikationsparameter für 2. Empfangs-PDO :

CAN-Identifizier: 0x300 + Knotennummer
Übertragungsart: 0xFE
Inhibit-Time : 0 (keine)

1402 Kommunikationsparameter für 3. Empfangs-PDO :

CAN-Identifizier: 0x400 + Knotennummer
Übertragungsart: 0xFE
Inhibit-Time : 0 (keine)

¹ Siehe Hinweis zur automatischen, Knotennummer-abhängigen Umschaltung der PDO-CAN-Identifizier in Kapitel 3.2.1 .

1600 Mapping-Parameter ("Aufbau des Datenfeldes") für 1. Empfangs-PDO :

Erstes Datenbyte : Digitale Ausgänge 1..8 (Objekt 6200.01, 8 bit)
Zweites Datenbyte : Digitale Ausgänge 9..16 sofern vorhanden

1601 Mapping-Parameter für 2. Empfangs-PDO :

1. und 2. Datenbyte : Analog-Ausgang 1 (Objekt 6411.01, 16 bit)
3. und 4. Datenbyte : Analog-Ausgang 2 (Objekt 6411.02, 16 bit)

1602 Mapping-Parameter für 3. Empfangs-PDO :

Per Defaulteinstellung nicht belegt (keine Objekte im PDO gemappt)

1600 Mapping-Parameter ("Aufbau des Datenfeldes") für 1. Empfangs-PDO :

Erstes Datenbyte : Digitale Ausgänge 1..8 (Objekt 6200.01)
Zweites Datenbyte : Digitale Ausgänge 9..16 sofern vorhanden

1800 Kommunikationsparameter für 1. Sende-PDO :

CAN-Identifizier: 0x180 + Knotennummer
Übertragungsart: 0xFE
Inhibit-Time : 100 µs
Event-Timer : 0 (aus, d.h. kein überlagerter periodischer Sendezyklus)

1801 Kommunikationsparameter für 2. Sende-PDO :

CAN-Identifizier: 0x280 + Knotennummer
Übertragungsart: 0xFE
Inhibit-Time : 5000 µs
Event-Timer : 0 (aus, d.h. kein überlagerter periodischer Sendezyklus)

1802 Kommunikationsparameter für 3. Sende-PDO :

CAN-Identifizier: 0x380 + Knotennummer
Übertragungsart: 0xFE
Inhibit-Time : 5000 µs
Event-Timer : 0 (aus, d.h. kein überlagerter periodischer Sendezyklus)

1A00 Mapping-Parameter für 1. Sende-PDO : Digitale Eingänge

Erstes Datenbyte : Digitale Eingänge 1..8 (Objekt 6000.01, 8 bit)
Zweites Datenbyte : Digitale Eingänge 9..16 sofern vorhanden

1A01 Mapping-Parameter für 2. Sende-PDO : Analoge Eingänge

1. und 2. Datenbyte : Analog Eingang 1 (Objekt 6401.01, 16 bit)
3. und 4. Datenbyte : Analog Eingang 2 (Objekt 6401.02, 16 bit)

1A02 Mapping-Parameter für 3. Sende-PDO : Frequenzmesser aus Kapitel 4.1

1. und 2. Datenbyte : Frequenzzähler (Objekt 5C01.02, 16 bit)
3. und 4. Datenbyte : Zeitdifferenz (Objekt 5C01.03, 16 bit)
5. und 6. Datenbyte : gemessene Frequenz (Objekt 5C01.04, 16 bit)

3.2.1 Knotennummer-abhängige, automatische Umschaltung der PDO-CAN-Identifizier

Um den Anwendern, die nicht über ein CANopen-Konfigurationstool verfügen, die Inbetriebnahme mehrerer Module in einem CANopen-Netzwerk zu erleichtern, wurde die folgende nicht ganz CANopen-konforme Erweiterung implementiert (die aus unten genauer erläuterten Gründen aber keine Probleme mit CANopen-Konfigurationstools verursachen dürfte).

Beim vorliegenden Gerät werden alle PDO-Kommunikations- und -Mapping-Parameter nichtflüchtig abgespeichert. Dazu zählen auch die CAN-Identifizier ("COB-Identifizier") der Prozessdatenkanäle. In manchen Anwendungen stört die Tatsache, daß die einmal per SDO eingestellten PDO-CAN-Identifizier nicht mehr "automatisch" mit dem Umschalten der Knotennummer per DIP-Schalter geändert werden (dies würde bedeuten, daß der Inhalt des Objektverzeichnisses beim Ändern der Knotennummer des Moduls automatisch wieder auf die Defaultwerte gesetzt würde - eine unakzeptable Funktion).

Im CANopen-Standard DS301 sind die folgenden Default-Identifizier für die Prozessdatenobjekte definiert (als Teil des sogenannten "predefined connection sets") :

- RPDO1: 200h + < Knotennummer des Moduls>
- RPDO2: 300h + < Knotennummer des Moduls>
- RPDO3: 400h + < Knotennummer des Moduls>
- TPDO1: 180h + < Knotennummer des Moduls>
- TPDO2: 280h + < Knotennummer des Moduls>
- TPDO3: 380h + < Knotennummer des Moduls>

Modulintern werden die PDO-CAN-ID's als 32-Bit-Werte in Objekt 140X bzw 180X unter Subindex 1 abgespeichert, und zwar komplett (d.h. inklusive der niederwertigsten Bits, die beim "predefined connection set" der Knotennummer 1..127₁₀ entsprechen). Wenn nach der Parametrierung des Moduls die Knotennummer (per DIP-Schalter) umgeschaltet wird, ändern sich die CAN-Identifizier der PDO-Kanäle daher nicht mehr. Dies ist in einigen Anwendungen erwünscht, in anderen nicht.

Um diese Problematik zu umgehen und beiden Anwendungen gerecht zu werden, wurde der folgende Ansatz gewählt:

- Normalerweise wird beim Schreibzugriff auf den PDO-Kommunikationsparameter, Subindex 1, der "komplette" CAN-Identifizier programmiert, so wie er später verwendet werden soll. Dies ist vollkommen CANopen-konform.
- Alternativ kann beim Schreibzugriff auf den Kommunikationsparameter, Subindex 1, auch "nur" Basis-Identifizier (z.B. 200h, 300h) ohne Knotennummer programmiert werden. Beim Lesezugriff addiert die Firmware dann automatisch die Knotennummer, wenn in Subindex 1 nur der Basis-Identifizier (z.B. 200h, 300h..) programmiert wurde. Beim Lesezugriff erscheint daher immer der tatsächlich verwendete CAN-Identifizier.
- Bei der Auslieferung des Gerätes sind unter Subindex 1 in allen PDO-Kommunikationsparametern tatsächlich nur die "Basis-Identifizier" abgelegt, d.h. ohne Knotennummer. Die Knotennummer wird daher beim Einschalten abhängig vom DIP-Schalter addiert. Dies geht auch aus dem EDS-File hervor: Dort steht z.B. unter Objekt 1400, Subindex 1, der Eintrag 0x200 (nicht 0x201).

4. Optionale Funktionen

4.1 Frequenzmesser

Dieser Frequenzmesser (~ "Frequenzähler") ist eine Sonderfunktion, die nicht im Geräteprofil DS401 enthalten ist. Aus diesem Grund finden Sie das entsprechende Objekt im *herstellerspezifischen Teil* des Objektverzeichnisses. Zum Zeitpunkt der Erstellung dieses Dokumentes waren die folgenden Indizes bzw Subindizes für den Frequenzmesser implementiert:

- 5C01.00 : enthält den höchsten Subindex, hier '5' .
- 5C01.01 : Torzeit in Millisekunden, 16 Bit Integerzahl ohne Vorzeichen. Schreib- und Lesbar. Kann nicht ein PDO (Prozeßdatenobjekt) gemappt werden, Zugriff daher nur über SDO. Die maximale Torzeit beträgt 327 ms. Default-Einstellung nach dem Ausliefern 10 ms.
- 5C01.02 : Während der letzten Torzeit gezählte Impulse als 16-Bit-Integer-Wert. Kann zur schnellen Übertragung in ein Prozeßdatenobjekt eingeblendet werden.
- 5C01.03 : Gemessene Zeit zwischen erstem und letzten Impuls innerhalb der Torzeit als 16-Bit-Integer-Wert. Die Einheit hängt von der Torzeit ab, siehe Unten.
- 5C01.04 : Aus 5C01.02 und 5C01.03 berechnete Frequenz als 16-Bit-Integer-Wert, Einheit Hz.
- 5C01.05 : Aus 5C01.02 und 5C01.03 berechnete Frequenz als 32-Bit-Integer-Wert, Einheit 0.1 Hz.

Meßprinzip

Der Frequenzmesser zählt während der sogenannten Torzeit (z.B. 10 oder 100 ms) die Impulse am entsprechenden Eingang, und misst die Zeit zwischen erstem und letzten Impuls. Dieser Vorgang wird ständig wiederholt, bei einer Torzeit von 100 ms finden daher ca. 10 Messungen pro Sekunde statt.

Aus der Anzahl der Impulse innerhalb der Torzeit und der präzise gemessenen Zeit zwischen erstem und letzten Impuls (innerhalb einer Torzeit) kann die Firmware auch bei relativ kleinen Torzeiten eine höhere Auflösung erzielen als durch eine reine Impulzzählung.

(Aus diesem Grund ist die Bezeichnung "Frequenzähler" nicht ganz korrekt, es handelt sich um eine Kombination von Zähler mit Torzeit und zusätzlichem Zeitmesser - daher "Frequenzmesser".)

Die Messfrequenz (in Objekt 5C01.04 und 05) wird nach dem folgenden Prinzip berechnet:

Messfrequenz := < Anzahl Impulse 5C01.02 > / < gemessene Zeit 5C01.03 >

Die physikalische Einheit von Objekt 5C01.03 hängt von der parametrisierten Torzeit ab, um trotz der hardwarebedingten Wortbreite eine möglichst hohe Auflösung zu erzielen. Es gilt :

- Torzeit bis zu 10 Millisekunden: Einheit von 5C01.03 = Mikrosekunden (1 μ s)
- Torzeit 11 bis 100 Millisekunden: Einheit von 5C01.03 = 0.01 Millisekunden (10 μ s)
- Torzeit 101 bis 327 Millisekunden: Einheit von 5C01.03 = 0.1 Millisekunden (100 μ s)

Bei der Parametrierung der Torzeit (5C01.01) muß ein Kompromiß zwischen Genauigkeit und Aktualisierungsrate gefunden werden. Eine höhere Torzeit bringt ein genaueres Ergebnis, aber auch -prinzipbedingt- eine langsamere Aktualisierung. Es macht daher wenig Sinn, bei einer Torzeit von z.B. 200 Millisekunden das Ergebnis alle 10 Millisekunden per PDO zu übertragen.

Minimale Eingangsfrequenz

Bitte beachten Sie, daß durch die Torzeit auch die minimal messbare Eingangsfrequenz festgelegt wird. Bei 10 ms sollte die Frequenz beispielsweise über 200(!) Hz liegen, damit innerhalb eines Torzeit-Intervalls mindestens zwei(!) Impulse gezählt werden können. Bei 100ms liegt die minimal meßbare Frequenz bei 20 Hz. Liegt die Eingangsfrequenz in vier aufeinanderfolgenden Messperioden unter diesem Wert, wird als Messwert die Frequenz "Null" geliefert !

Maximale Eingangsfrequenz

Die *spezifizierte* Maximalfrequenz liegt bei 30 kHz, bei einem Test unter "Laborbedingungen" arbeitete der Frequenzmesser noch bei einigen hundert kHz. Das 16-Bit-Ergebnis in Hertz (5C01.04) funktioniert prinzipbedingt nur bis 65535 Hz, bei höheren Frequenzen wird der Meßwert auf diese Frequenz begrenzt (es erfolgt kein Überlauf nach Null oder ähnliches). Frequenzen über 65 kHz können nur per Objekt 5C01.05 gelesen werden. Die einwandfreie Funktion oberhalb der spezifizierten Maximalfrequenz kann allerdings nicht garantiert werden !

Eingang des Frequenzmessers

Bei der Variante "SAHM" liegt der Eingang des Frequenzmessers an Klemme X9, Pin 2, Signal "DIGITAL_OUT 9 / FREQ_IN" (siehe Hardwarebeschreibung). Per Jumper muss diese Klemme als EINGANG (d.h. "FREQ_IN") konfiguriert werden.

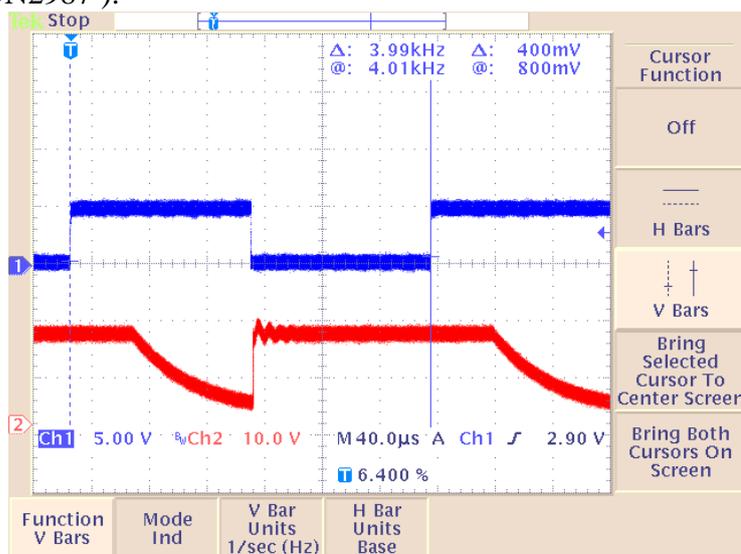
Vorsicht: Wenn Klemme X9, Pin 2 versehentlich als "Ausgang" konfiguriert wurde, besteht die Gefahr einen eventuell angeschlossenen Sensor zu zerstören !

4.2 Frequenzgänge

Optional können die Ausgänge DIGITAL_OUT 9 und DIGITAL_OUT 10 auch als Frequenzgänge verwendet werden. Die Ansteuerung kann per PDO oder SDO erfolgen (CANopen-Objekt 5C11, im manufacturer specific profile area):

- Index 0x5C11, Subindex 0: Anzahl der Frequenzgänge (2 seit Juli 2005)
 - Index 0x5C11, Subindex 1: Ausgangsfrequenz Kanal 1 (DO9) in Hertz, 0=Generator aus
 - Index 0x5C11, Subindex 2: Ausgangsfrequenz Kanal 2 (DO10) in Hertz, 0=Generator aus
- Die minimale Ausgangsfrequenz beträgt für beide Kanäle 12 Hz, die maximale Ausgangsfrequenz 4 kHz (bei 1 Hz Auflösung); Frequenzen bis 20 kHz sind bei verringerter Auflösung möglich (Auflösung ca. 17 Hz bei 20 kHz Ausgangsfrequenz).

Leider sind die im E/A-Modul verwendeten Standard-Ausgangstreiber (High-Side-Treiber) recht langsam, falls eine zu hochohmige Last angeschlossen wird ! Hier z.B. ein Rechtecksignal mit 4 kHz (blau: TTL-Signal an der CPU; rot: verschliffenes Signal an DIGITAL_OUT 9, Ausgangstreiber: UDN2987).



Abhilfe: Externen Lastwiderstand von 1...2 kOhm, 1 Watt, vom Ausgang nach Masse schalten.

5. Testfunktionen

5.1 Systemtest per Diagnoseschnittstelle

Zum Aktualisieren der Firmware und für Testzwecke verfügt das E/A-Modul über eine asynchrone serielle Schnittstelle ("RS232 mit TTL-Pegeln"). Zur Verbindung mit dem PC dient ein Testadapter mit RS232-Pegelwandler und DIP-Schalter für die Umschaltung zwischen "Programmiermodus" und "Testmodus".

Hinweis: Verwenden Sie diese Schnittstelle -wenn überhaupt- nur mit dem geeigneten Testadapter, und wirklich **nur zu Testzwecken !**

Im Testmodus kann mit einem geeigneten Terminalprogramm (z.B. Hyperterminal[®]) eine Verbindung ohne CAN-Bus zwischen PC und E/A-Modul hergestellt werden. Die notwendigen Einstellungen für die serielle Schnittstelle sind:

- 9600 bits pro Sekunde
- 8 Datenbits
- keine Parität
- 1 Stopbit
- kein Protokoll

Zur Steuerung des E/A-Moduls im Testmodus dienen einzelne ASCII-Buchstaben, die z.B. per Hyperterminal gesendet werden können. Das E/A-Modul reagiert dann mit entsprechenden Anzeigen als Klartext. Verfügbare Kommandos (als einzelner Kleinbuchstabe):

- e Enter debug/test mode
Aktiviert den Debug- bzw. Testmodus. Dabei wird die Firmware-Variante, die Firmware-Version und das Compilationsdatum angezeigt. Die meisten der folgenden Testkommandos sind erst nach dieser Aktivierung verfügbar.
- h Help
Zeigt eine Übersicht aller verfügbaren Kommandos an
- a Analog input values
Zeigt den aktuellen Zustand der analogen Eingänge an
- d DIP-switch-test / jumpers used for configuration.
Diese Funktion kann verwendet werden, um zu prüfen ob alle Jumper bzw DIP-Schalter passend gesetzt sind (z.B. für die CAN-Baudrate, Knotennummer, Konfiguration der analogen Ein/Ausgänge, Umschaltung Frequenzeingang oder digitaler Ausgang, ...)
- f Frequency inputs
Zeigt die wichtigsten Variablen des Frequenzmessers an. Zum Beenden der Endlosschleife eine beliebige Taste drücken.
- i digital inputs
Zeigt den aktuellen Zustand der digitalen Eingänge an.
- k Kill object dictionary in non-volatile memory (EEPROM).
Damit kann das Modul "radikal" in den Ursprungszustand versetzt werden. Alle Einträge im nichtflüchtigen Speicher des Objektverzeichnisses werden gelöscht. Empfehlenswert nach Firmware-Update, falls "neues" und "altes" OD nicht zueinander passen.
- o digital Outputs
Erlaubt einen schnellen Test aller digitalen Ausgänge auch ohne CAN-Bus
- r Ramp
Erzeugt eine Rampenfunktion an den analogen Ausgängen, und ein durchlaufendes Testmuster an den digitalen Ausgängen. Verwendbar als Schnelltest aller Ausgänge.

- q Quit
Beendet den Test/Debug-Modus, die normale CANopen-Kommunikation wird wieder aufgenommen.