Ingenieurbüro für Echtzeitprogrammierung

Dokumentation IF555-5

Dok-Rev. 1.2 vom 14.11.2019

Hardware-Rev. 2.2 vom 16.04.2019

Inhaltsverzeichnis

1 Allgemeine Hinweise 5
1.1 Handhabung 5
1.2 Installation 5
1.3 Erklärung 5
1.4 Reparaturen 5

2 Technische Daten 6
2.1 Umgebungsbedingungen 6
2.2 Mechanische Abmessungen 6
2.3 Elektrische Daten 6

3 Inbetriebnahme 7
3.1 Einbau 7
3.2 Spannungsversorgung 7
3.3 Steckverbinder 7
3.4 Beschreibung der Taster 7
3.5 Lage der Jumper und Anschlüsse 8
3.6 Beschreibung der Jumper 8
3.6.1 JU9/JU10: RS485 Receiver enable 9
3.6.2 DIP1/DIP2: Abschlußwiederstand CAN-Bus 9
3.6.3 JU1-JU8: Analoge Ausgänge Strom/Spannung umschalten 9
3.6.4 JU11: Watchdog Timeout 9
3.6.5 JU12: Watchdog automatisch starten 9
3.7 Jumper des IEP CORE-5125 Moduls 9

4 Hardwarebeschreibung 10
4.1 Verteilung der 24 Volt 10
4.2 Ethernetschnittstelle 10
4.3 Serielle Schnittstellen 10
4.3.1 Schnittstelle A1 10
4.3.2 Schnittstelle A2 11
4.3.3 Schnittstelle A5 11
4.3.4 Schnittstelle A3 12
4.3.5 Schnittstelle A4 12
4.4 CAN-Bus 12
4.4.1 Belegung CAN-Bus 12
4.5 Micro SD Card 13
4.6 Digitale Ausgänge 13

4.6.1 Zuordnung der Ausgänge DO6..DO14 14
4.6.2 Sonderfunktion des GPT 14
4.7 Digitale Eingänge 15
4.8 Analoge Ausgänge 16
4.9 Analoge Eingänge 16
4.10 Leuchtdioden 17
4.11 Externer Watchdog 17

5 Programmierung 18
5.1 Events 18
5.2 Watchdog 19
5.3 Digitale Ausgänge 19
5.4 Digitale Eingänge 19
5.5 Analoge Ausgänge 20
5.6 Analoge Eingänge 20
5.7 µSD Card-Slot auf Core5125 20
5.7.1 Formatieren der µSD-Card 20
5.8 NandFlash Disk auf Core5125 20

6 PEARL Zugriffsfunktionen für den I/O Zugriff 21
6.1 Spezifikationen der I/O-Funktion im Überblick 21
6.1.1 Digitale Ein/Ausgänge 21
6.1.2 Analoge Ein/Ausgäng 21
6.1.3 Serielle Ein/Ausgänge 21
6.2 Zugriff auf digitale Ein-/Ausgänge 22
6.2.1 Fehlerstatus der Digitale Ausgänge 22
6.3 Pieper 23
6.4 Zugriff auf analoge Ausgänge 23
6.5 Zugriff auf analoge Eingänge 24

7 RS485 Schnittstellen A3/A4 25
7.1 Wahl der Schnittstelle 25
7.2 Setup der RS485 Schnittstelle 25
7.3 Senden auf der RS485 Schnittstelle 26
7.4 Empfangen auf der RS485 Schnittstelle 26
7.4.1 SPI_WAIT_SERIELL 26
7.4.2 SPI_LEN_SERIELL 26
7.4.3 SPI_RX_SERIELL 27

8 RTOS-UH im internen FLASH ablegen 28

9 Autostart von Programmen 28

Revisionsliste:

Rev.	Datum	Na.	Änderung
1.0	29.04.2019	Kr	Erstellung
1.1	06.05.2019	Ko	Ergänzungen
1.2	14.11.2019	Ko	Dig. Ausgänge ggf. externe GND-Verbindung

1 Allgemeine Hinweise

1.1 Handhabung

1. Lesen Sie bitte zuerst sorgfältig diese Dokumentation bevor Sie die Hardware auspacken und einschalten. Sie sparen Zeit und vermeiden Probleme.
2. Beachten Sie bitte die Vorsichtsmaßnahmen bei der Handhabung elektrostatisch gefährdeter Hardware.
3. Wenn die Hardware Batterien enthält, legen Sie sie nicht auf elektrisch leitfähige Unterlagen. Die Batterie könnte kurzgeschlossen werden und Schäden verursachen.
4. Achten Sie bitte darauf, daß der spezifizierte Temperaturbereich nicht verlassen wird.

1.2 Installation

1. Überprüfen Sie, ob alle Jumper entsprechend Ihrer Anwendung gesetzt sind.
2. Schalten Sie die Spannungsversorgung der externen Anschlüsse ab, bevor Sie eine Verbindung herstellen.
3. Wenn Sie sicher sind, daß alle Verbindungen korrekt installiert sind, schalten Sie die Spannungsversorgung ein.

1.3 Erklärung

Wir behalten uns das Recht vor, Änderungen, die einer Verbesserung der Schaltung oder des Produktes dienen, ohne besondere Hinweise vorzunehmen. Trotz sorgfältiger Kontrolle kann für die Richtigkeit der hier gegebenen Daten, Schaltpläne, Programme und Beschreibungen keine Haftung übernommen werden. Die Eignung des Produktes für einen bestimmten Einsatzzweck wird nicht zugesichert.

1.4 Reparaturen

Sollte das Produkt defekt sein, so senden Sie es bitte frei in geeigneter Verpackung mit folgender Beschreibung an uns zurück:

- Fehlerbeschreibung
- Trat der Fehler nur unter bestimmten Bedingungen auf?
- Was war angeschlossen?
- Wie sahen die angeschlossenen Signale aus?
- Garantiereparatur oder nicht?

Alle Rechte vorbehalten. © IEP GmbH 2000-2019

2 Technische Daten

2.1 Umgebungsbedingungen

Umgebungstemperatur (Betrieb)	0-50° C
Umgebungstemperatur (Lagerung)	-20-85° C
rel. Luftfeuchte	max. 95%, nicht kondensierend
Höhe	-300m bis +3000m

2.2 Mechanische Abmessungen

Kartengröße	257 x 222 mm
Anschlüße	Pfostensteckverbinder

2.3 Elektrische Daten

Versorgungsspannung:	18-30 Volt DC, max. 2A
Prozessor	MPC5125
ext. Speicher	µSD Card
Batteriepufferung:	für Echtzeituhr, über Lithium-Batterie oder Gold-Cap-Kondensator
Analogeingänge	32 differenzielle, Eingangsempfindlichkeit 0..5V oder anpassbar, Auflösung 12 Bit
Analogausgänge	8 Stück, 0..20mA oder 0-10 Volt über DAC (12Bit)
Digitaleingänge:	32 Stück, 24 Volt über Optokoppler, z.t. als Zähler nutzbar (max. 80 KHz)
Digitalausgänge:	32 Stück 24 Volt, max. 0,4 A
Ser. Schnittstellen	3x 5-Draht RS232, 2x RS422/485
CAN	1x CAN galvanisch getrennt, 1x CAN nicht getr.
Ethernet	2x 10/100BaseT auf RJ45-Buchse

3 Inbetriebnahme

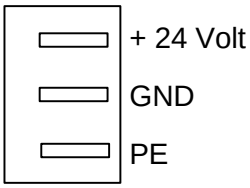
3.1 Einbau

Das IF555-5-Trägerboard ist zum Einbau in Schaltschränke oder ähnliche EMV-dichte Gehäuse bestimmt. Die Verkabelung ist EMV-gerecht von entsprechend ausgebildetem Fachpersonal durchzuführen.

3.2 Spannungsversorgung

Die Spannungsversorgung ist für 24 Volt Gleichspannung ausgelegt. Spannungen von 18-30 Volt sind zulässig. Die Spannungsversorgung erfolgt über den 3 poligen Steckverbinder ST17 mit folgender Belegung:

Aufsicht von oben:



Die Baugruppe ist gegen Verpolung der Speisespannung geschützt. Als Steckverbinder wird der Typ MSTB 2,5/3-ST-5,08 der Firma Phoenix Contact eingesetzt.

3.3 Steckverbinder

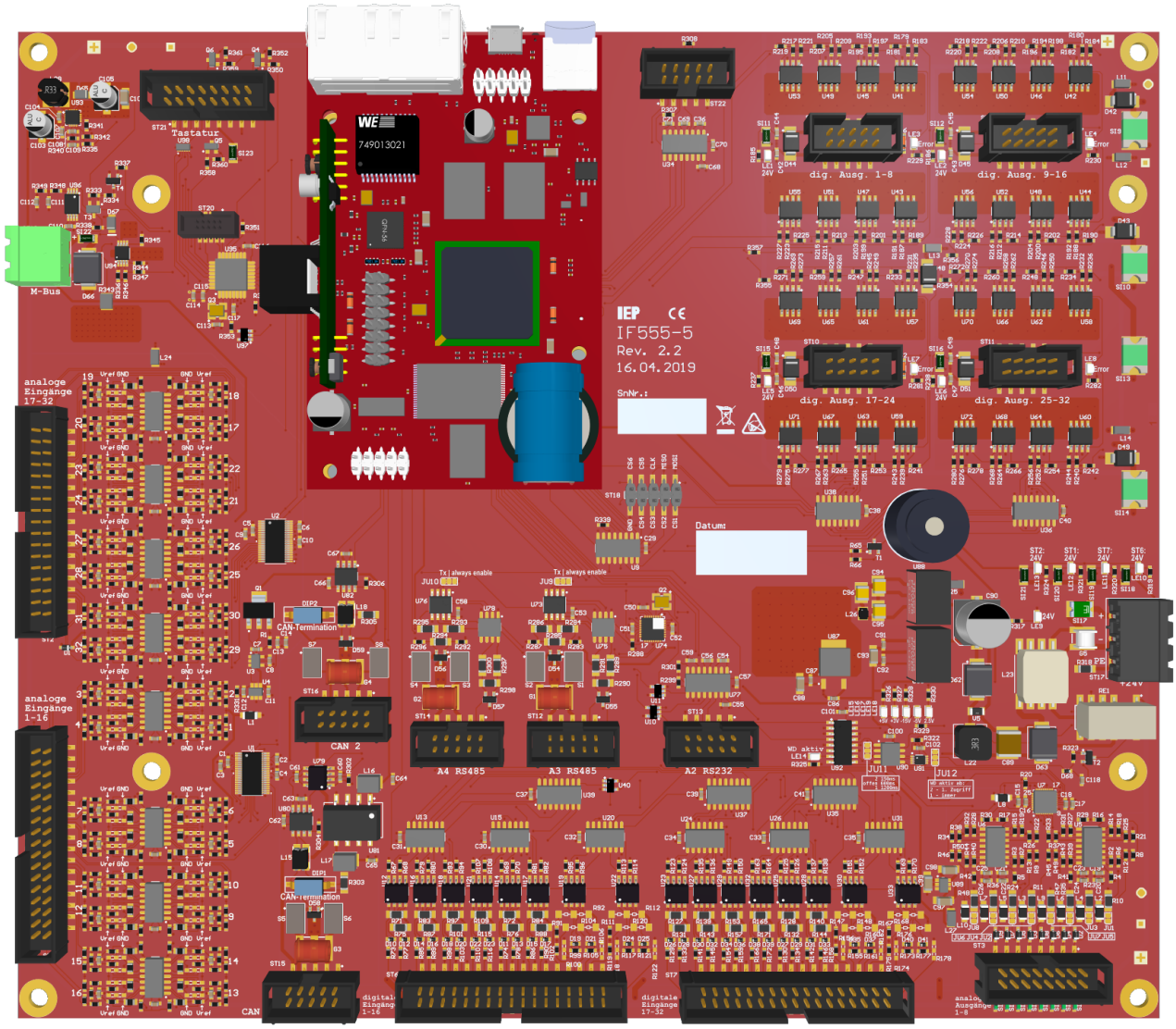
Die Anschlüsse für die I/O‘s werden über 2,5 mm Pfostensteckverbinder geführt:

Anschluß	Verbinder
digitale Eingänge	2x 34 polig (je 16 Eingänge)
digitale Ausgänge	4x 10 polig (je 8 Ausgänge)
analoge Eingänge	2x 34 polig (je 16 Eingänge)
analoge Ausgänge	16 polig (8 Ausgänge)
serielle Schnittstellen	4x 10 polig
CAN-Schnittstellen	2x 10 polig
Ethernet	2xRJ45 Buchse
Micro SD Card	Halter auf MPC5125
USB	Anschluss auf MPC5125

3.4 Beschreibung der Taster

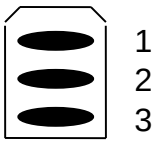
Auf dem CORE-5125 befindet sich unter der µSD-Karte der Reset-Taster.

3.5 Lage der Jumper und Anschlüsse



3.6 Beschreibung der Jumper

Die Löt-Jumper werden folgendermaßen gezählt:



3.6.1 JU9/JU10: RS485 Receiver enable

JU9 bestimmt das Verhalten des Receivers der Schnittstelle A3 (ST12), JU10 das der Schnittstelle A4 (ST14), wenn die Schnittstelle als RS485 bestückt ist.

Der Receiver kann auf immer „enabled“ gejumpert werden, dann werden auch die vom eigenen Transmitter gesendeten Zeichen empfangen. In der anderen Jumperstellung wird nur empfangen, wenn der eigene Transmitter nicht sendet.

Receiver	Jumper
nicht beim Senden enabled	1-2
immer enabled	2-3

3.6.2 DIP1/DIP2: Abschlußwiederstand CAN-Bus

Mit DIP1 wird der Abschlußwiederstand des CAN-Bus Kanals 1, mit DIP2 von Kanal 2 ein oder ausgeschaltet.

Abschlußwiederstand	Jumper
abgeschaltet	Off
eingeschaltet	On

3.6.3 JU1-JU8: Analoge Ausgänge Strom/Spannung umschalten

Die analogen Ausgänge können 0-20mA liefern. Wird der entsprechende Jumper geschlossen, so wird eine Bürde von 536 Ω auf den Ausgang geschaltet, so dass 0-10V ausgegeben werden.

Ausgang	Jumper
0-20 mA	Offen
0-10 V	1-2

3.6.4 JU11: Watchdog Timeout

Die Zeit, bis der der Watchdog anspricht, kann eingestellt werden.

Zeit			Jumper
min.	typ.	max.	
62,5 ms	150 ms	250 ms	2-3
250 ms	600 ms	1000 ms	offen
500 ms	1200 ms	2000 ms	1-2

3.6.5 JU12: Watchdog automatisch starten

Funktion	Jumper
Watchdog startet automatisch beim Einschalten	1-2
Watchdog startet mit dem ersten Zugriff auf den Watchdog	2-3

3.7 Jumper des IEP CORE-5125 Moduls

Die Jumper sind ab Werk richtig gesetzt und müssen nicht verändert werden.

4 Hardwarebeschreibung

Das IF555-5-Trägerboard ist für die Aufnahme des IEP CORE-5125 vorgesehen. Es stellt die Anpassung für die I/O-Signale des MCP5125 zur Verfügung. Die Beschreibung finden Sie in den folgenden Kapiteln.

4.1 Verteilung der 24 Volt

Die eingespeisten 24 Volt werden auf verschiedene Stecker verteilt. Zum Teil sind eigene Sicherungen auf der Platine vorgesehen. Die 24 Volt werden vor dem EMV-Filter abgegriffen. Um die Spannungen auf einen Blick überwachen zu können, sind LED's vorhanden. Die folgende Tabelle gibt einen Überblick:

Signal	Sicherung		Stecker	Anschluß	LED
+24V_1	0,2 A selbstrückstellend	SI18	ST6	DI1 - DI16	LE10
+24V_2	0,2 A selbstrückstellend	SI19	ST7	DI17- DI32	LE11
+24V_31	0,2 A selbstrückstellend	SI9+SI11	ST8	DO9 - DO16	LE1
+24V_32	0,2 A selbstrückstellend	SI10+SI12	ST9	DO1 –DO8	LE2
+24V_41	0,2 A selbstrückstellend	SI13+SI15	ST10	DO17- DO24	LE5
+24V_42	0,2 A selbstrückstellend	SI14+SI16	ST11	DO25- DO32	LE6
+24V_5	0,2 A selbstrückstellend	SI20	ST1	AI1 - AI16	LE12
+24V_6	0,2 A selbstrückstellend	SI21	ST2	AI17- AI32	LE13

Bei den digitalen Ausgängen sind jeweils 8 Ausgänge mit einer zusätzlichen 3A Sicherung versehen. Bitte beachten Sie, dass ein dauerhafter Kurzschluß nicht zulässig ist, da die Sicherungen dann so heiß werden, dass Schäden an der Platine nicht ausgeschlossen werden können!

4.2 Ethernetschnittstelle

Das Core5125 Modul stellt einen Switch mit zwei 10/100BaseT-Schnittstellen (ST7) zur Verfügung. Über die jeweiligen beiden Leuchtdioden im Stecker wird der Zustand der Ethernet-Schnittstelle angezeigt. Die gelbe LED leuchtet bei einer 100 MBit Verbindung. Die grüne LED leuchtet bei einer bestehenden Verbindung, sie blinkt, wenn Daten übertragen werden.

4.3 Serielle Schnittstellen

Es stehen 5 serielle Schnittstellen zur Verfügung, die unterschiedlich bestückt sind.

4.3.1 Schnittstelle A1

Die Schnittstelle A1 ist die Bedienschnittstelle des RTOS-UH. Über diese Schnittstelle werden z.B. die Einschaltmeldung und ggf. Fehlermeldungen ausgegeben. Diese Schnittstelle ist als 5-Draht RS232-Schnittstelle realisiert. Die Startbaudrate beträgt 57600 Baud. Die Schnittstelle ist auf dem CORE5125 auf der Pfostenleiste ST5 (10 polig) verfügbar.

Die Belegung ist so gewählt, dass ein angepreßter 9 poliger SubD-Stecker eine PC-kompatible Belegung hat:

ST5	PIN	PIN	ST5
	1	2	
RxD	3	4	RTS
TxD	5	6	CTS
	7	8	
GND	9	10	

4.3.2 Schnittstelle A2

Die Schnittstelle A2 ist als Modemschnittstelle ausgelegt. Sie ist als 5-Draht RS232-Schnittstelle realisiert. Die Startbaudrate beträgt 57600 Baud.

Die Belegung ist so gewählt, dass ein angepreßter 9 poliger SubD-Stecker eine PC-kompatible Belegung hat:

ST13	PIN	PIN	ST13
	1	2	
RxD	3	4	RTS
TxD	5	6	CTS
	7	8	
GND	9	10	

4.3.3 Schnittstelle A5

Die Schnittstelle A5 ist als Modemschnittstelle ausgelegt. Auf dieser Schnittstelle stellt RTOS-UH keinen USER zur Verfügung. Sie ist als 5-Draht RS232-Schnittstelle realisiert. Die Startbaudrate beträgt 57600 Baud.

Die Belegung ist so gewählt, dass ein angepreßter 9 poliger SubD-Stecker eine PC-kompatible Belegung hat:

ST22	PIN	PIN	ST22
	1	2	
RxD	3	4	RTS
TxD	5	6	CTS
	7	8	
GND	9	10	

4.3.4 Schnittstelle A3

Die Schnittstelle A4 kann als RS422- (U78+U76) oder RS485-Schnittstelle (U76) bestückt sein. Dementsprechend können die Abschlußwiederstände R297-R300 bestückt werden.

Ist die Schnittstelle als RS485 bestückt, sind die Signale R*x* am Anschluß ST14 nicht belegt.

ST12	PIN	PIN	ST12
	1	2	
	3	4	Rx -
Tx +	5	6	Tx -
Rx +	7	8	
GND	9	10	

4.3.5 Schnittstelle A4

Die Schnittstelle A4 kann als RS422- (U78+U76) oder RS485-Schnittstelle (U76) bestückt sein. Dementsprechend können die Abschlußwiederstände R297-R300 bestückt werden.

Ist die Schnittstelle als RS485 bestückt, sind die Signale R*x* am Anschluß ST14 nicht belegt.

ST14	PIN	PIN	ST14
	1	2	
	3	4	Rx -
Tx +	5	6	Tx -
Rx +	7	8	
GND	9	10	

4.4 CAN-Bus

Es stehen die 2 CAN-Busschnittstellen des MPC5125 zur Verfügung. Die Schnittstelle CAN 1 (ST15) ist galvanisch vom Rest des Boards getrennt. Mit DIP1 kann der Abschlußwiederstand eingeschaltet werden.

! Achtung CAN1 ist aus RTOS Sicht der CAN-Kanal 3 für die Programmierung.

Die Schnittstelle CAN 2 (ST16) ist direkt mit dem CAN-Tranceiver auf dem Prozessor-Board verbunden. Überspannungen können zur **Zerstörung** des Prozessors-Boards führen! Mit DIP2 kann der Abschlußwiederstand eingeschaltet werden.

! Achtung CAN2 ist aus RTOS Sicht der CAN-Kanal 4 für die Programmierung.

4.4.1 Belegung CAN-Bus

Beide Schnittstellen sind identisch belegt. Wird eine 9 polige SubD-Buchse angepreßt, so ergibt sich eine CANOpen® konforme Belegung:

ST15/16	PIN	PIN	ST15/16
	1	2	GND
CAN Low	3	4	CAN High
GND	5	6	
	7	8	
PE	9	10	

4.5 Micro SD Card

Die µSD-Card kann im PC formatiert werden, es darf aber kein FAT32-Dateisystem mit ClusterSize > 16KB verwendet werden. Unter dem Device `/H0` ist die SD-Card im RTOS-UH zu finden.

4.6 Digitale Ausgänge

Es stehen 32 digitale Ausgänge in 4 Gruppen á 8 Ausgängen zur Verfügung. Die High-Side-Schalter können bei 24 Volt bis zu 400 mA liefern. Alle 32 Ausgänge können max. 12 A Summenstrom liefern, die selbstrückstellenden Sicherung `SI9/10/15/16` (je 3A) übernehmen die Absicherung. Die Ausgänge sind zum Schalten induktiver Lasten geeignet und dauerkurzschlußfest. Die Treiberbausteine sind gegen Übertemperatur gesichert. Im Fehlerfall wird der betroffene Ausgang zurückgesetzt, d.h. der Ausgang ist gesperrt. Dieser Zustand wird über eine rote Error-LED je Gruppe angezeigt. Der Fehlerzustand wird bei Beendigung des Fehlers automatisch zurückgesetzt. Es werden weitere Fehlerzustände erkannt, z.B. keine Last angeschlossen usw. Nähere Informationen entnehmen Sie bitte dem Datenblatt des `BSP742`.

Die 4 Stecker sind gleich belegt:

ST8	ST9	ST10	ST11	PIN	PIN	ST11	ST10	ST9	ST8
DO1	DO9	DO17	DO25	1	2	DO26	DO18	DO10	DO2
DO3	DO11	DO19	DO27	3	4	DO28	DO20	DO12	DO4
DO5	DO13	DO21	DO29	5	6	DO30	DO22	DO14	DO6
DO7	DO15	DO23	DO31	7	8	DO32	DO24	DO16	DO8
+24V_31	+24V_32	+24V_42	+24V_41	9	10	GND	GND	GND	GND

Die Ausgänge `DO1-DO7` sind an die GeneralPurposeTimer GPT angeschlossen und können Sonderfunktionen ausführen.

Bitte beachten Sie, dass über die GND-Anschlüsse von ST8-11 gesamt max. 300 mA fließen dürfen. Werden also größere Lasten geschaltet, muß eine externe Verbindung von den Lasten zum Netzteil für den GND vorhanden sein!

4.6.1 Zuordnung der Ausgänge DO6..DO14

Ausgang	GPT	Kanal	Basisadresse
DO6	2	0	$F0005000
DO7	2	1	$F0005010
DO8	2	2	$F0005020
DO9	1	2	$F0000B20
DO10	1	3	$F0000B30
DO11	1	4	$F0000B40
DO12	1	5	$F0000B50
DO13	1	6	$F0000B60
DO14	1	7	$F0000B70

4.6.2 Sonderfunktion des GPT

Sonderfunktionen des GPT können über die mode-Register eingestellt werden. Dazu muss das PEARL-Bit(30) gesetzt sein (GPIO-Mode). Im GPIO-Mode verhalten sich die Ausgänge wie normale Ausgänge, d.h. sie werden mit der PEARL-Funktion `SPI_RW_DIO` mitgesetzt. Sind die Ausgänge auf eine andere Funktionalität programmiert, werden sie von `SPI_RW_DIO` nicht beeinflußt.

```
TYPE  gpt_type STRUCT[ mode BIT(32), count BITT(32), pwm BIT(32), state BIT(32) ] ;
```

4.7 Digitale Eingänge

Die digitalen Eingänge DI1-DI12 und DI17-DI28 sind einseitig auf Masse bezogen. Die Eingänge DI13-DI16 und DI29-DI32 können sowohl einseitig auf Masse bezogen werden, als auch als potentialgetrennte Eingänge genutzt werden. Beide Steckverbinder sind identisch belegt:

ST6	ST7	PIN	PIN	ST7	ST6
DI1	DI17	1	2	GND	GND
DI2	DI18	3	4	GND	GND
DI3	DI19	5	6	GND	GND
DI4	DI20	7	8	GND	GND
DI5	DI21	9	10	GND	GND
DI6	DI22	11	12	GND	GND
DI7	DI23	13	14	GND	GND
DI8	DI24	15	16	GND	GND
DI9	DI25	17	18	GND	GND
DI10	DI26	19	20	GND	GND
DI11	DI27	21	22	GND	GND
DI12	DI28	23	24	GND	GND
+DI13	+DI29	25	26	-DI29	-DI13
+DI14	+DI30	27	28	-DI30	-DI14
+DI15	+DI31	29	30	-DI31	-DI15
+DI16	+DI32	31	32	-DI32	-DI16
+24V_1	+24V_2	33	34	+24V_2	+24V_1

Werden die Eingänge DI13-DI16 und DI29-DI32 potentialfrei genutzt, kehrt sich die Polarität der Eingänge um!

4.8 Analoge Ausgänge

Die analogen Ausgänge sind für 0..20mA ausgelegt. Ausgang Nr. 8 kann auch negative Spannungen ausgeben. Je nach Bestückung sind auch andere Bereiche möglich. Der Rückkanal der analogen Ausgänge ist jeweils über eine selbstrückstellende 20 mA Sicherung geführt, die bei zu großen Unterschieden im Ground-Potential auslöst. Spannungsausgänge von 0-10 Volt stehen durch Schließen einer Bürde von 536 Ω zur Verfügung.

Jumper	ST3	PIN	PIN	ST3	Sicherung
JU1	AO1	1	2	GND	SI1
JU3	AO2	3	4	GND	SI2
JU5	AO3	5	6	GND	SI3
JU6	AO4	7	8	GND	SI4
JU2	AO5	9	10	GND	SI5
JU4	AO6	11	12	GND	SI6
JU6	AO7	13	14	GND	SI7
JU8	AO8	15	16	GND	SI8

4.9 Analoge Eingänge

Die analogen Eingänge sind als differentielle Eingänge ausgeführt. Der Eingangsspannungsbereich kann über eine unterschiedliche Bestückung in weiten Grenzen variiert werden.

ST2	ST1	PIN	PIN	ST1	ST2
AI17+	AI1+	1	2	AI1-	AI17-
AI18+	AI2+	3	4	AI2-	AI18-
AI19+	AI3+	5	6	AI3-	AI19-
AI20+	AI4+	7	8	AI4-	AI20-
AI21+	AI5+	9	10	AI5-	AI21-
AI22+	AI6+	11	12	AI6-	AI22-
AI23+	AI7+	13	14	AI7-	AI23-
AI24+	AI8+	15	16	AI8-	AI24-
AI25+	AI9+	17	18	AI9-	AI25-
AI26+	AI10+	19	20	AI10-	AI26-
AI27+	AI11+	21	22	AI11-	AI27-
AI28+	AI12+	23	24	AI12-	AI28-
AI29+	AI13+	25	26	AI13-	AI29-
AI30+	AI14+	27	28	AI14-	AI30-
AI31+	AI15+	29	30	AI15-	AI31-
AI32+	AI16+	31	32	AI16-	AI32-
+24V_6	+24V_5	33	34	+24V_5	+24V_6

4.10 Leuchtdioden

Zur Kontrolle der unterschiedlichen Spannungen sind auf der Platine 11 grüne Leuchtdioden vorhanden:

Spannung	LED
+24V_1	LE10
+24V_2	LE11
+24V_31	LE1
+24V_32	LE2
+24V_41	LE5
+24V_42	LE6
+24V_5	LE12
+24V_6	LE13
+5 Volt	LE15
+3,3 Volt	LE16
+15 Volt	LE17
-5 Volt	LE19
+2.5 V Referenz	LE18

4.11 Externer Watchdog

Es steht zusätzlich zum internen Watchdog des MPC5125 noch ein externer Watchdog zur Verfügung. Dieser Watchdog wird je nach Jumperstellung aktiviert und muß dann regelmäßig getriggert werden (s. cap. 5.2). Spricht der Watchdog an, so unterbricht ein Relais die 24 Volt Versorgungsspannung des Boards.

5 Programmierung

Die Programmierung der auf dem IEP CORE-5125 befindlichen Bausteine wird vom Betriebssystem durchgeführt. Die Programmierung aller externen Komponenten wird im Folgenden beschrieben. Absolut notwendig zum Verständnis ist die detailierte Kenntnis des MPC5125 User‘s Manual.

Es ist unter https://www.nxp.com/docs/en/reference-manual/MPC5125RM.pdf zu finden.

Adreßbelegung

Chip-Select	Anschluß	Größe	Adresse	Programmierung
LPC_CS2	Watchdog	128 KB	$F7000000-$F701FFFF	0 WS
Interner MPC5125 Bereich		4 MB	$F0000000-$F1FFFFFF	
SRAM des MPC5125		128K	$F0200000-F$F021FFF	

5.1 Events

Folgende Events sind belegt:

Event	Funktion
EV 00000001	GPT1[0]
EV 00000002	GPT1[1]
EV 00000004	GPT1[2]
EV 00000008	GPT1[3]
EV 00000010	GPT1[4]
EV 00000020	GPT1[5]
EV 00000040	GPT1[6]
EV 00000080	GPT1[7]
EV 00000100	GPT2[0]
EV 00000200	GPT2[1]
EV 00000400	GPT2[2]
EV 00000800	GPT2[3]
EV 00001000	GPT2[4]
EV 00002000	GPT2[5]
EV 00004000	GPT2[6]
EV 00008000	GPT2[7]

Damit der entsprechende Event ausgelöst wird, muss der Interrupt physikalisch freigegeben werden.

5.2 Watchdog

Der interne Watchdog des MPC5125 wird vom Betriebssystem initialisiert und von der Task `Watch` auf sehr hoher Priorität (-9) im 0.5s Zyklus getriggert. Unterbleibt die Triggerung des Watchdogs für mehr als 1 s, wird ein Reset des IF555-5 ausgelöst.

Die Task `Watch` kann problemlos terminiert und mit einer anderen Wunschpriorität wieder aktiviert werden. Auch eine Neutriggerung des Watchdogs direkt über eine Anwendung ist möglich.

Der externe Watchdog des IF555-5 muß anwendungsspezifisch konfiguriert (s. Jumper, cap. 3.6.4) und von der Anwendung regelmäßig getriggert werden. Zum Starten und Triggern des Watchdogs ist ein beliebiger schreibender Byte-Zugriff in den Adressbereich des Watchdogs (s. Adressbelegung, CS2, `$F7000000-$F701FFFF`) erforderlich.

5.3 Digitale Ausgänge

Die Fehlermeldungen der 4 Ausgangsgruppen sind an folgende Ports angeschlossen:

Port [Bit]	Ausgangsgruppe
GPIO1[9]	DO1 -DO8
GPIO1[19]	DO9 -DO16
GPIO1[20]	DO17-DO24
GPIO1[6]	DO25-DO32

Ein Low-Pegel zeigt einen Fehler an.

5.4 Digitale Eingänge

Die digitalen Eingänge sind folgendermassen angeschlossen:

Eingang	Port [Kanal/Bit]
DI1-DI8	SPI
DI9	GPT2[4]
DI10-DI24	SPI
DI25	GPT2[3]
DI26	GPT2[5]
DI27	GPT2[6]
DI28-DI29	SPI
DI30	GPT2[7]
DI31	GPT1[0]
DI32	GPT1[1]

Fließt Strom durch den Eingang (Schalter geschlossen), liegt am Eingang High-Pegel.

5.5 Analoge Ausgänge

Die 8 analogen Ausgänge werden über einen DAC mit 12 Bit Auflösung realisiert. Der DAC ist über das SPI an den Prozessor angeschlossen.

WandlerCodierung ist folgend: $0000 [0V] … $0FFF [10V] mit 5µs Settling Time

5.6 Analoge Eingänge

Die analogen Eingänge sind an HighSpeed AD-Wandler mit 12 Bit Auflösung angeschlossen. Die Wandlungszeit beträgt ca 1 µs je Kanal.

Die Wandlercodierung ist folgend: $0000 [0V] …$0800[Vref]…$0FFF[2*Vref]

Die effektive Codierung ist von der Eingangsbestückung des jeweiligen AD-Kanals abhängig.

z.B.:

Eingang -2,5V…+2,5V ergibt bei 0V = $0800 bei -2,5V = $0000

Eingang 0V…10V ergibt bei 5V = 4096/2 = 2048 = $0800

5.7 µSD Card-Slot auf Core5125

Es können handelsübliche µSD-Cards genutzt werden. Die maximal unterstützte Größe beträgt 128 GByte. Die µSD-Card ist unter RTOS als Device ***/H0/*** verfügbar.

5.7.1 Formatieren der µSD-Card

Die µSD-Card läßt sich mit folgendem Kommando formatieren:

```
FORM D /H0/FULL
```

Bitte beachten Sie, dass das Formatieren mehrere Minuten dauern kann und die Schnittstelle solange blockiert ist.

5.8 NandFlash Disk auf Core5125

Im Nandflash kann eine ca. 100MB grosse FlashDisk ***/FD/*** eingerichtet werden. Bitte beachten Sie, dass diese Flashdisk nur eine begrenzte Anzahl Schreib/Löschzyklen unterstützt und daher nicht für das Speichern von sich schnell ändernden Daten genutzt werden sollte! Für kleinere Datenmengen können auch die EEPROMs auf dem CORE5125 genutzt werden.

Die FlashDisk läßt sich mit folgendem Kommando formatieren:

```
FORM D /FD/FULL
```

6 PEARL Zugriffsfunktionen für den I/O Zugriff

6.1 Spezifikationen der I/O-Funktion im Überblick

In den folgenden Kapiteln werden die einzelnen Funktionen beschrieben.

6.1.1 Digitale Ein/Ausgänge

```
SPC SPI_RW_DIO     ENTRY( value BIT(32) ) RETURNS( BIT(32) ) GLOBAL ;
SPC SPI_READ_DIO   ENTRY                  RETURNS( BIT(32) ) GLOBAL ;
SPC SPI_GET_DO_ERR ENTRY                  RETURNS( BIT(16) ) GLOBAL ;
SPC SPI_SET_HUPE   ENTRY( FIXED(15) )                        GLOBAL ;
```

6.1.2 Analoge Ein/Ausgäng

```
DCL MAX_DA_CHANNEL INV FIXED INIT( 8) ;
DCL MAX_AD_CHANNEL INV FIXED INIT(16) ;
TYPE dac_type STRUCT[ c( MAX_DA_CHANNEL ) BIT(16)] ;
TYPE adc_type STRUCT[ c( MAX_AD_CHANNEL ) BIT(16)] ;

SPC SPI_WRITE_DAC ENTRY( val REF dac_type ) GLOBAL ;
SPC SPI_READ_ADC1 ENTRY( val REF adc_type ) GLOBAL ;
SPC SPI_READ_ADC2 ENTRY( val REF adc_type ) GLOBAL ;
```

6.1.3 Serielle Ein/Ausgänge

```
SPC SPI_TX_SERIELL ENTRY( chan FIXED, len FIXED, REF CHAR(1) ) GLOBAL ;
SPC SPI_RX_SERIELL ENTRY( chan FIXED,
                          maxlen FIXED,
                          REF CHAR(1) ) RETURNS( FIXED ) GLOBAL ;
SPC SPI_WAIT_SERIELL ENTRY( chan FIXED,
                            timeout DURATION ) RETURNS( FIXED ) GLOBAL ;
SPC SPI_LEN_SERIELL ENTRY( chan FIXED ) RETURNS( FIXED ) GLOBAL ;
SPC SPI_SETUP_SERIELL ENTRY( chan FIXED,
                             baud FIXED(31),
                             conf REF CHAR(3) ) RETURNS( FIXED ) GLOBAL ;
```

6.2 Zugriff auf digitale Ein-/Ausgänge

Die Funktion SPI_RW_DIO wird genutzt, um die digitalen Ausgänge zu setzen und gleichzeitig die digitalen Eingänge einzulesen.

```
SPC SPI_RW_DIO ENTRY( out BIT(32) ) RETURNS( BIT(32) ) GLOBAL ;
```

out enthält den neu zu setzenden Ausgangszustand und die Funktion gibt als Rückgabe den aktuellen Istzustand der digitalen Eingänge zurück.

Beispiel:

```
DCL dig_eingang BIT(32) ;
DCL dig_ausgang BIT(32) ;
..
dig_eingang = SPI_RW_DIO( dig_ausgang ) ;
..
```

Die Funktion benötigt eine Laufzeit von ca 30 µs.

Zum alleinigen Lesen der Eingänge kann die folgende Funktion genutzt werden.

```
SPC SPI_READ_DIO ENTRY  RETURNS( BIT(32) ) GLOBAL ;
```

Beispiel:

```
dig_eingang = SPI_READ_DIO() ;
```

Anmerkung:

Da die Eingänge DI9 / DI31 / DI32 mittels der GPT Funktionalität umgeschaltet werden können, werden die Eingänge nur beim Mode = GPIO richtig abgebildet! Es wird immer der aktuelle PIN-Status dargestellt, der bei anderen Modi abweichen kann.

Die Ausgänge D01..D06 sind ebenfalls über den GPT angeschlossen. Nur wenn der Mode = GPIO gesetzt ist, werden sie von der Funktion SPI_RW_DIO berücksichtigt.

6.2.1 Fehlerstatus der Digitale Ausgänge

Um den aktuellen Fehlerstatus der Digitalen Ausgangsgruppen zu lesen, wird die Funktion

```
SPC SPI_GET_DO_ERR ENTRY  RETURNS( BIT(16) ) GLOBAL ;
```

genutzt. Die Bitzuordnung der Ausgangsgruppen ist folgend:

.BIT(1) - Gruppe DO1…DO8
.BIT(2) - Gruppe DO9…D16
.BIT(3) - Gruppe D17…D24
.BIT(4) - Gruppe D25…D32

Ein .BIT(x) EQ ‘1’B zeigt die fehlerfreie Funktion aller Ausgänge der jeweiligen Gruppe an.

Beispiel:

```
…
   DCL err_state_do BIT(16) ;
…
   err_state_do = SPI_GET_DO_ERR() ;
```

6.3 Pieper

Um den Pieper auf dem IF555-5 zu schalten, steht die Funktion

```
SPC SPI_SET_HUPE ENTRY( onoff FIXED(15) ) GLOBAL ;
```

zur Verfügung.

Der Parameter onoff ist wie folgt belegt:

```
onoff > 0 ;   /* Pieper ist AN */
onoff = 0 ;   /* Pieper ist AUS */
```

6.4 Zugriff auf analoge Ausgänge

Die analogen Ausgänge (8 Kanäle) werden mittels der Funktion `SPI_WRITE_DAC` gesetzt. Der gültige Wertbereich umfaßt 12 Bit und hat die folgende Kodierung

0V / 0mA entspricht $0000 und 10V / 20mA entspricht $0FFF

DA-Ausgangsspannung = Wert / 4096 * 10 Volt

Typenvereinbarung in PEARL

```
DCL MAX_DA_CHANNEL INV FIXED INIT(8) ;
TYPE  dac_type STRUCT[ c( MAX_DA_CHANNEL ) BIT(16)] ;

SPC SPI_WRITE_DAC ENTRY( val REF dac_type ) GLOBAL ;
```

Beispiel:

```
   DCL Dac dac_type ;
..
   FOR i TO 8 REPEAT ;
      Dac.c( i ) = 0 ;
   END ;

   FOR i TO 4095 REPEAT ;
       Dac.c(1) = TOBIT(  I )  ;                  /* Sägezahn vorwärts */
       Dac.c(2) = TOBIT( 4095 - I )  ;            /* Sägezahn rückwärts */
       Dac.c(3) = TOBIT( 2048 + ( 4095 // 2 ) ) ; /* Dreieck */
       Dac.c(4) = TOBIT( ( i REM 2 ) * 4095 ) ;   /* Rechteck */
      SPI_WRITE_DAC( Dac ) ;
   END ;
```

6.5 Zugriff auf analoge Eingänge

Die analogen Eingänge (32 Kanäle) werden mittels der Funktion `SPI_READ_ADC`*x* eingelesen. Der gültige Wertbereich umfaßt 12 Bit. Die Umrechnung der AD-Wandlerwerte ist von der jeweiligen Bestückung der Eingänge abhängig.

Beispiele:

Spannungeingang mit -2,5 Volt .. +2,5 Volt Eingangsbereich.

Wert = ((AD-Wert – 2048) / 2048). * 2.5 ;

Spannungeingang mit 0 Volt .. 10 Volt Eingangsbereich.

Wert = (AD-Wert / 4095.) * 10. ;

Typenvereinbarung in PEARL

```
DCL MAX_AD_CHANNEL INV FIXED INIT(16) ;
TYPE  adc_type STRUCT[ c( MAX_AD_CHANNEL ) BIT(16)] ;

SPC SPI_READ_ADC1 ENTRY( val REF adc_type )  GLOBAL ;
SPC SPI_READ_ADC2 ENTRY( val REF adc_type )  GLOBAL ;
```

Mit der Funktion `SPI_READ_ADC1` werden die AD-Kanäle 1..16 eingelesen, mit `SPI_READ_ADC2` die AD-Kanäle 17..32.

7 RS485 Schnittstellen A3/A4

Für die seriellen Schnittstellen steht kein IO-Warteschlangentreiber zur Verfügung, sondern eine Sammlung von PEARL-Funktionen:

```
SPC SPI_SETUP_SERIELL ENTRY( chan FIXED,
                             baud FIXED(31),
                             conf REF CHAR(3) ) RETURNS( FIXED ) GLOBAL ;

SPC SPI_WAIT_SERIELL ENTRY( chan FIXED,
                           timeout DURATION ) RETURNS( FIXED ) GLOBAL ;
SPC SPI_LEN_SERIELL ENTRY( chan FIXED ) RETURNS( FIXED ) GLOBAL ;
SPC SPI_RX_SERIELL ENTRY( chan FIXED,
                         maxlen FIXED,
                         REF CHAR(1) ) RETURNS( FIXED ) GLOBAL ;

SPC SPI_TX_SERIELL ENTRY( chan FIXED, len FIXED, REF CHAR(1) ) GLOBAL ;
```

7.1 Wahl der Schnittstelle

Der auf der Platine mit **A3** bezeichnete Anschluss wird mit ***chan = 0*** angewählt, der mit **A4** bezeichnete Anschluss mit ***chan = 1***.

7.2 Setup der RS485 Schnittstelle

Beim Start von RTOS-UH wird eine Interruptbetreungstask #tSER für die seriellen Schnittstellen generiert, diese setzt beide Schnittstellen mit den folgenden Werten auf:

19200 Baud, 8 Datenbits, 1 Stopbit, keine Parität.

Je Schnittstellen wird einen Interruptpuffer von 128 Byte Größe verwaltet, in dem die empfangenen Bytes abgelegt werden. Wird der Puffer nicht ausgelesen, werden neu eintreffende Daten verworfen.

Falls eine andere Einstellung der Parameter gewünscht ist, so können mit der Funktion

```
Err = SPI_SETUP_SERIELL( chan, baud, conf ) ;
```

die Werte geändert warden. Zulässige Werte für die einzelnen Parameter sind:

chan = { 0 | 1 }

baud = { 300 | 600 | 1200 | 2400 | 4800 | 9600 | 19200 }

conf = { 81N | 82N | 81E | 81O | 71N | 71E | 71O }

7.3 Senden auf der RS485 Schnittstelle

Mit der Funktion SPI_TX_SERIELL können Daten auf die jeweilige Schnittstellen ausgegeben werden. Die Daten werden in den FIFO des seriellen Bausteines geschrieben, daher kehrt die Funktion schon in das Anwenderprogramm zurück, obwohl noch nicht alle Daten gesendet wurden. Der FIFO ist 64 Bytes gross. Sollen mehr Daten verschickt werden, wartet die Funktion, bis wieder genug Platz im FIFO vorhanden ist.

```
SPC SPI_TX_SERIELL ENTRY( chan FIXED, len FIXED, c REF CHAR(1) ) GLOBAL ;
```

Beispiel:

```
   DCL puffer CHAR(20) INIT('Dies ist ein Text OK') ;
   ..
   SPI_TX_SERIELL( 0, 20, puffer.CHAR(1) ) ;  /* sendet die Daten auf A3 */
```

7.4 Empfangen auf der RS485 Schnittstelle

Für das Empfangen von Daten stehen für die unterschiedlichen Betriebsarten folgende Funktionen zur Verfügung:

```
SPC SPI_WAIT_SERIELL ENTRY( chan FIXED,
                           timeout DURATION ) RETURNS( FIXED ) GLOBAL ;
SPC SPI_LEN_SERIELL ENTRY( chan FIXED ) RETURNS( FIXED ) GLOBAL ;
SPC SPI_RX_SERIELL ENTRY( chan FIXED,
                         maxlen FIXED,
                         c REF CHAR(1) ) RETURNS( FIXED )
```

7.4.1 SPI_WAIT_SERIELL

Diese Funktion dient zum Warten auf min. ein Zeichen mit einem angebbaren Timeout. Die aufrufende Task wird solange in den Zustand SUSP/TICO gesetzt, bis entweder mindestens ein Zeichen auf der RS485 erkannt wurde, oder das Timeout abgelaufen ist.

Achtung: die Zeitangabe für das Timeout wird als DURATION mit interpretiert.

Der Rückgabewert der Funktion kann folgende Werte annehmen.

- -1 ungültiger Aufruf
- -7 Timeout abgelaufen / kein Zeichen erkannt
- 1 ein Zeichen ist erkannt worden

Beispiel:

```
MODE=CLOCK50
…
/* warte auf A4 maximal 2 SEC auf Daten */
State = SPI_WAIT_SERIELL( 1, 2 SEC ) ;
```

7.4.2 SPI_LEN_SERIELL

Diese Funktion liefert die aktuell im Interruptbuffer stehende Anzahl von Zeichen zurück. Es werden keine Zeichen aus dem Buffer entnommen.

```
Anzahl = SPI_LEN_SERIELL( 1 ) ; /* Lese Anzahl Daten von A4 */
```

7.4.3 SPI_RX_SERIELL

Diese Funktion entnimmt, falls vorhanden, bis zu der mit dem Parameter ***maxlen*** angegebenen Anzahl Zeichen aus dem Interruptbuffer und füllt sie in den mit ***c*** übergebenen Buffer ein. Falls im Interruptbuffer weniger Zeichen vorhanden sind, werden entsprechend weniger Zeichen übergeben.

Der Rückgabewert der Funktion entspricht der Anzahl der übergebenen Zeichen.

```
SPC SPI_RX_SERIELL ENTRY( chan FIXED,
                          maxlen FIXED,
                          c REF CHAR(1) ) RETURNS( FIXED )
```

Beispiel

```
 DCL Buff CHAR(128) ;
 DCL Anz FIXED ;
..
  Anz = SPI_RX_SERIELL( 1, 128, Buff.CHAR(1) ) ;
..
```

Falls 25 Zeichen im Interruptbuffer sind, werden 25 Zeichen gelesen und entsprechend Anz = 25 gesetzt.

Bei leerem Interruptbuffer wird Anz = 0 gesetzt.

8 RTOS-UH im internen FLASH ablegen

Ist schon ein RTOS-UH auf dem MPC5125 am laufen, kann der neue RTOS-UH S-Record auf die Adresse 800000 geladen und dann mit START_RTOS angesprungen werden. Danach kann das System mit den folgenden Befehlen upgedated werden.

NAND B legt den Boot-Bereich im NAND ab

NAND S legt das RTOS-UH System im NAND ab

NAND F legt eine FailsafeKopie des Booters und Systemes im NAND ab.

9 Autostart von Programmen

Das RTOS-UH versucht beim Start eine Datei mit dem Namen ***AUTO.EX*** auf den Devices ***/H0/*** und ***/FD/*** zu finden. Die Reihenfolge ist erst /H0/, dann wird auf /FD/ gesucht. Die erste gefundene Datei wird mit der EX-Shell interpretiert und ensprechend die enthaltenen Kommandos ausgeführt.

Falls auf ***/H0/*** eine Datei vorhanden ist, wird eine weitere Datei auf der ***/FD/*** ignoriert.