

BENUTZERHANDBUCH

**CM509
CM517A**

CONTROLLERMODUL

frenzel + berg electronic GmbH & Co.KG

**Turmgasse 4
89073 Ulm**

**Tel (07 31) 9 70 57-0
Fax (07 31) 9 70 57-39**

**Copyright © 1996, 2000
Stand 04.06**

Copyright

Copyright © 1996 durch frenzel + berg electronic GmbH & Co.KG. Alle Rechte vorbehalten.

Haftungsausschluss

Die frenzel + berg electronic GmbH & Co.KG übernimmt keinerlei Haftung oder Garantie bezüglich des Inhalts dieser Veröffentlichung und übernimmt für fehlerhafte Angaben und deren Folgen weder eine juristische Verantwortung, noch irgend eine Haftung. Ferner verneinen wir jegliche Gewährungsansprüche bezüglich der Eignung des Produkts zu irgend einem speziellen Zweck.

Weiterhin halten wir uns das Recht vor, dieses Handbuch sowie das darin beschriebene Produkt ohne jegliche Informationspflicht an irgendwelche Personen zu ergänzen oder zu ändern.

Liefermöglichkeiten und technische Änderungen bleiben vorbehalten.

Inhaltsverzeichnis

| | |
|---|-----------|
| INHALTSVERZEICHNIS | 2 |
| APPLIKATIONEN UND SOFTWARE-ENTWICKLUNGEN | 3 |
| 1. EINFÜHRUNG | 4 |
| 1.1. Hardwarekomponenten..... | 6 |
| 1.2. Allgemeine Anschlussdefinition für CM-Module | 7 |
| 2. MODULPINNING | 8 |
| 2.1. Anschlussbild..... | 8 |
| 2.2. Anschlussbelegung..... | 9 |
| 3. APPLIKATIONSHARDWARE | 12 |
| 4. SPEICHERAUSBAU | 13 |
| 4.1. Datenspeicher U4 (SRAM)..... | 13 |
| 4.2. Programm-Speicher U5 (Flash-EPROM) | 13 |
| 4.3. Auswahl der Speicherseite | 14 |
| 4.4. Das Paging-Register | 15 |
| 4.5. Speicherbelegung / Betriebsmodi des CM509..... | 16 |
| 4.6. Serielles IIC-Bus-EEPROM U6..... | 18 |
| 5. PERIPHERIE | 18 |
| 5.1. Serielle Schnittstellen..... | 18 |
| 5.1.1. <i>Serielle Schnittstelle SIO0</i> | 18 |
| 5.1.2. <i>Serielle Schnittstelle SIO1</i> | 18 |
| 5.2. CAN-Schnittstelle | 19 |
| 5.3. Echtzeituhr (RTC: Real Time Clock) | 20 |
| 5.4. Supervisory | 21 |
| 5.4.1. <i>Überwachungs-, Resetschaltung und Schreibschutzverriegelung</i> | 21 |
| 5.4.2. <i>Zusätzlicher Watchdog</i> | 22 |
| 5.5. Externe Hardware-Erweiterungen über den Datenbus | 22 |
| 6. DECODER | 23 |
| 6.1. RAM-Decoder U14 | 23 |
| 6.2. Flash-Decoder U13 | 24 |
| 6.2.1. <i>PLD-Listing</i> | 26 |
| 6.3. Peripherie-Decoder U12 | 27 |
| 6.3.1. <i>PLD-Listing</i> | 28 |
| 7. JUMPER | 29 |
| 7.1. Jumperplan | 30 |
| 8. TECHNISCHE DATEN | 31 |
| 8.1. Elektronische und Temperatur Daten | 31 |
| 8.2. Mechanische Abmessungen..... | 32 |
| 9. LAGEPLAN | 33 |
| 10. SCHALTPLAN | 35 |

Applikationen und Software-Entwicklungen

Zur 8051-Familie und C166-Familie erhalten Sie beim Hersteller des Minirechnermoduls CM509:

frenzel + berg electronic GmbH & Co.KG
Turmgasse 4
89073 Ulm
Tel. 0731 / 97057- 0
Fax 0731 / 97057- 39

Nutzen Sie unser Know-How und unsere Erfahrung.

1. Einführung

Das Controller-Modul CM509 ist ein vollständiges 8-Bit-Mikrocontrollermodul für den universellen Einsatz in einem breiten Applikationsspektrum. Es basiert auf dem Single-Chip-Mikrocontroller SAB-C509-L / SAB-C509-16F (mit internem Flash-Speicher) und wurde für den Einsatz im Steuerungs- und Automatisierungsbereich konzipiert. Viele Komponenten, die zur Realisierung solcher Aufgaben benötigt werden, sind bereits auf dem Modul integriert, so dass für die meisten Anwendungen keine Hardware-Entwicklung von CPU- und Peripherie-Einheiten mehr erforderlich ist. Dies spart Zeit und Geld bei der Durchführung ihrer Entwicklungsvorhaben.

Die integrierte Peripherie:

Speicherbausteine, zwei serielle RS232-Pegelumsetzer oder ein RS232- und ein RS485-Pegelumsetzer jumperkonfigurierbar, standalone Full-CAN-Controller, Echtzeituhr, serielles IIC-EEPROM (z.B. für Setup-Datenhaltung), Spannungsüberwachung mit optionaler Pufferung durch eine extern anzubringende Lithiumbatterie, ein Watchdogtimer und optional eine Funkuhr mit extern anzubringender Ferritantenne.

Das Modul gehört zu einer neu konzipierten Modulgeneration bei der Firma frenzel + berg electronic GmbH & Co.KG, die alle mit weitgehend identischem (aufwärtskompatiblen) Pinning ausgeführt sind.

Hierfür hat die Firma frenzel + berg electronic GmbH & Co.KG eine Definition der Anschlussstecker festgelegt, die allen Mikrocontrollermodulen der CM-Reihe zugrunde liegt. Somit können bei entsprechender Berücksichtigung der kompletten Definition und der controllerspezifischen Unterschiede alle CM-Module gegen leistungsstärkere CM-Typen ausgetauscht werden. Diese Anschlussdefinition finden Sie ebenfalls in diesem Handbuch.

Das Modul CM509 kann neben dem Mikrocontroller SAB-C509-L / SAB-C509-16F auch mit dem SAB 80C517A betrieben werden. Diese Mikrocontrollerderivate von Siemens basieren auf dem weitverbreiteten 8051-Standard.

Das Controller-Modul wird wie ein großer IC auf die applikationsspezifische Schaltung aufgesteckt. Mit einer Größe von nur 79 x 64 mm ist der Platzbedarf sehr gering. Durch moderne SMD-Bestückung konnte diese extreme Komplexität erreicht werden. Dennoch wurden für das SRAM, den EPROM / Flash-Speicher und den Peripherie-Decoder (GAL 16V8) aus Flexibilitätsgründen gesockelte Bausteine im DIL-Gehäuse verwendet.

Für die Programmierung des Mikrocontrollers können die Compiler bzw. Assembler der 8051-Familie verwendet werden, sofern sie Registerdefinitionen zulassen.

Zusätzliche Dokumentation

Diesem Manual können alle modulspezifischen Kennwerte und Informationen entnommen werden. Zusätzlich sollten Sie jedoch Unterlagen über die auf dem Controller-Modul verwendeten Bausteine verfügbar haben, sofern die entsprechenden Bausteine auf Ihrem Board bestückt sind und Sie diese auch verwenden wollen.

Dies sind:

Tabelle 1-1: Zusätzliche Dokumentation

| Baustein | Referenz | Hersteller | Verwendung |
|--|----------|--------------------------------------|---|
| SAB-C509-L / SAB-C509-16F User's Manual | U1 | Siemens | Mikrocontroller |
| SAB 80C517A / SAB 83C517A-5 Addendum to User's Manual SAB 80C517/ 80C537 SAB 80C517/ 80C537 U. M. | U1 | Siemens | Mikrocontroller |
| 24C04 | U6 | Catalyst/ST/.. | Seriellles I ² C-Bus- E ² PROM |
| RTC72423 | U10 | Suwa Seiko Epson | Echtzeituhr (RTC: Real Time Clock) |
| U4224B | U15 | TEMIC | Funkuhr (TCR: Time Code Receiver) |
| SAE 81C91 | U9 | Siemens | CAN-Controller |
| 29C010 / B | U5 | Je nach verwendetem Hersteller | Flash-Speicher |

1.1. Hardwarekomponenten

Das Controller-Modul CM509 verfügt über folgende Hardware-Komponenten:

- 8-Bit-Single-Chip-Mikrocontroller SAB-C509-L / SAB-C509-16F (mit internem Flash-Speicher) oder SAB 80C517A (ROM-lose Version)
- Taktfrequenz 16 MHz (Standardmodul), auf Wunsch 3,5 - 16 MHz (509), 3,5 - 18 MHz (517) möglich
- Controller mit maskenprogrammiertem ROM einsetzbar
- Befehlszykluszeit 375ns (SAB C509 bei Taktfrequenz 16 MHz), 666ns (SAB 80C517 bei Taktfrequenz 18 MHz)
- controllerinterner und externer Watchdogtimer für Programmlaufüberwachung
- 2 voll duplex serielle Schnittstellen (RS232) mit programmierbarer Baudrate (eine alternativ als RS485 nutzbar)
- 39 frei verfügbare digitale Ein- und Ausgänge
- 6 frei verfügbare digitale Eingänge
- 8 frei verfügbare digitale / analoge Eingänge
- bis zu 128 kB Flash-Speicher (alternativ 32K, 64K oder 128K EPROM einsetzbar)
- bis zu 128 kB SRAM mit möglicher Batteriepufferung
- Adressdecodierung mit Speicherpaging
- 2 freie Chip-Select-Signale für Hardwareerweiterungen
- Versorgungsspannungsüberwachung und Netzausfallfrüherkennung
- Schreibschutzverriegelung für SRAM
- Echtzeituhr inklusive Kalenderfunktion (Real Time Clock: RTC) mit möglicher Batteriepufferung
- uneingeschränkt lauffähig ohne zusätzliche Hardware
- geringer Platzbedarf durch kleine Abmessungen (79 x 64 mm)
- Speicherbausteine (Flash und SRAM) und Peripheriedecoderbaustein (GAL 16V8) im Dual-In-Line Gehäuse gesockelt
- gesockelter Peripherie-Decoder ermöglicht Konfigurationsänderung
- für die Programmierung der Festspeicher sind keine Adapter auf teure PLCC- oder andere Gehäuseformen erforderlich
- voll unterstützter Monitorbetrieb
- serielles IIC_EEPROM zur Setup-Datenhaltung
- CAN-Interface zur Vernetzung mehrerer Controllermodule („Embedded Networking“)
- optional bestückte Funkuhr (Time Code Receiver: TCR)
- kundenspezifische Bestückung und Software auf Anfrage

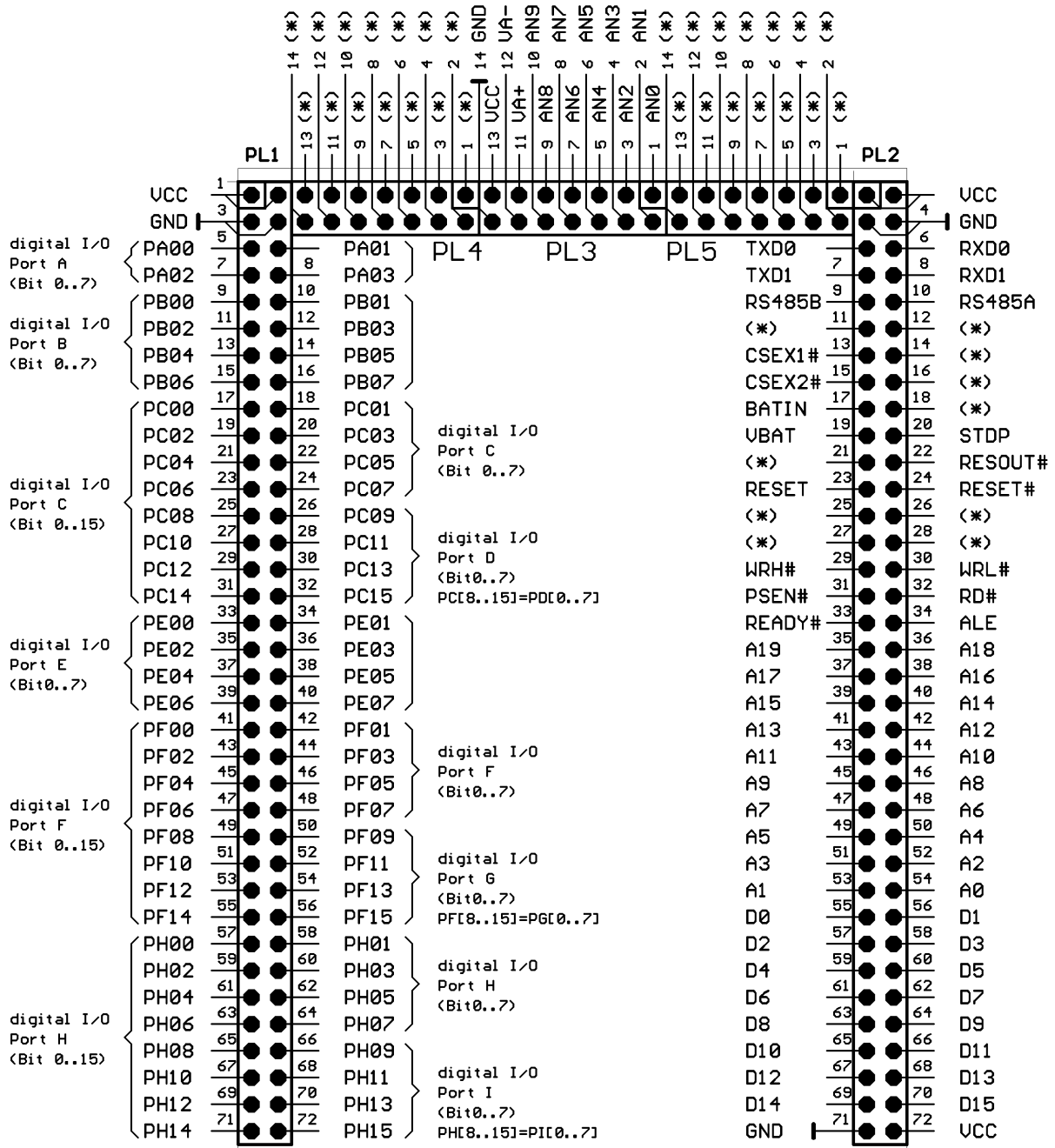
Das Modul hat zwei doppelreihige Steckerleisten mit jeweils 62 Polen, über die es mit der Applikationshardware verbunden werden kann. Über die Steckerleisten sind die Portleitungen, Adress- und Datenbus, sowie die Steuersignale verfügbar.

Die Versorgungsspannung des Moduls wird ebenfalls über diese Steckverbinder zugeführt.

1.2. Allgemeine Anschlussdefinition für CM-Module

Die nachfolgende Anschlussdefinition liegt allen Mikrocontrollermodulen der CM-Reihe zugrunde. Bei Berücksichtigung der kompletten Definition bei der Entwicklung von Applikationen ist eine Aufrüstung mit leistungsstärkeren oder peripherieerweiterten CM-Modulen gewährleistet, sofern die hardwarebedingten Unterschiede der einzelnen Controller bei der Applikationsentwicklung berücksichtigt werden.

**Abbildung 1-1: Allgemeine Anschlussdefinition für CM-Module
(von oben gesehen)**

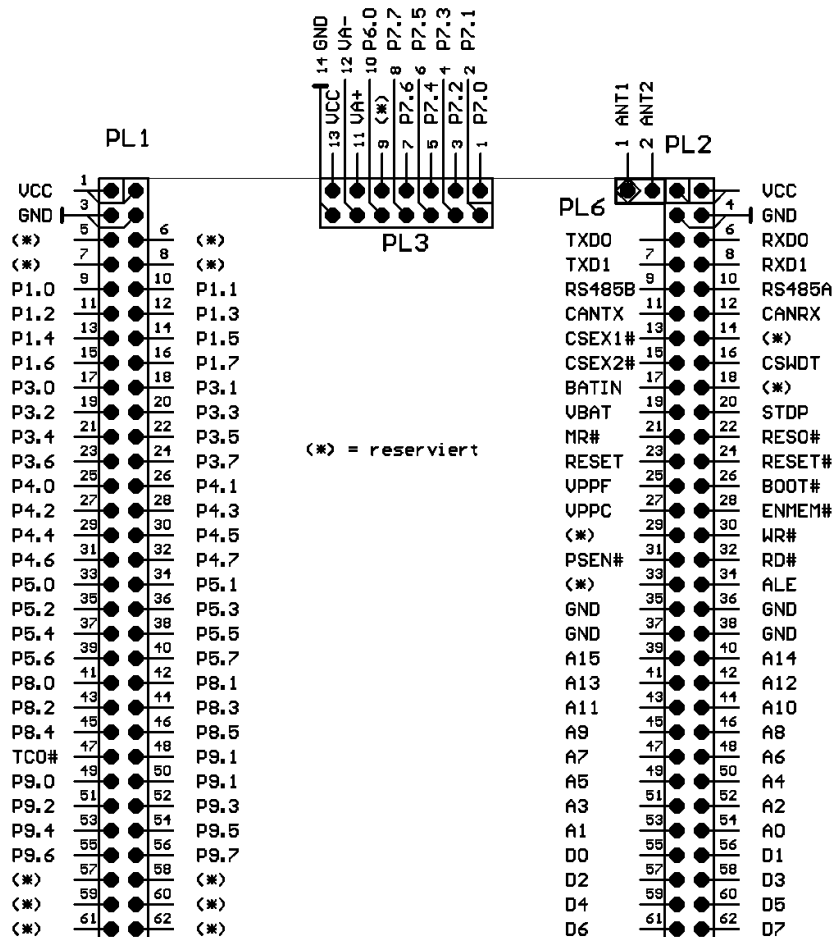


2. Modulpinning

2.1. Anschlussbild

Das Controller-Modul CM509 verfügt über zwei 62pol.doppelreihige Steckerleisten (PL1 u. PL2), an denen die Ports, Adress- und Datenbus, Steuersignale sowie die Spannungsversorgung zugänglich sind. Des weiteren sind über eine 14polige doppelreihige Steckerleiste (PL3) an der Stirnseite des Moduls 8 Analogeingänge, die digitale und analoge Spannungsversorgung zugänglich. Die Antenne der optional erhältlichen Funkuhr kann über die 2pol. Steckerleiste PL6 angeschlossen werden.

Abbildung 2-1: Anschlussbild CM 509 von oben gesehen



2.2 Anschlussbelegung

Alle Controller-Module verfügen über ein vergleichbares Pinning, wodurch die Austauschbarkeit untereinander erreicht wird. Außerdem kann bei gleicher Applikationshardware auf immer leistungsfähigere Module zurückgegriffen werden. Selbstverständlich ist bei unterschiedlichen Controllern keine hundertprozentige Pincompatibilität zu gewährleisten. Über 4 doppelreihige Stiftleisten sind die Signale und Versorgungsspannungen zugänglich.

Tabelle 2-1: Anschlusssteckverbinder:

| Stecker | Signale |
|---------|---|
| PL1 | Portleitungen des Mikrocontrollers, Betriebsspannung |
| PL2 | Adressbus, Datenbus, Steuersignale, serielle Schnittstellen, Betriebs- Programmierspannungen |
| PL3 | Analog- / Digitalwandler, Betriebsspannungen, Referenzspannung |
| PL4 | Anschlüsse für die Antenne der Funkuhr (TCR: Time Code Receiver) |

Nicht herausgeführt wurden folgende Controllerleitungen:

- Port 6.1 (RXD1) und
Port 6.2 (TXD1): Sende- und Empfangsleitung der 2. seriellen Schnittstelle
- Port 6.3 (WRF#): Write-Flash-Signal (Programmieren des Flashes über den
Programmiermodus des SAB C509)
- Port 6.4 (DEN: Driver Output Enable): Sendetreiber kann hochohmig bzw.
treibend geschaltet werden
- Port 6.5 (SDA: serial Data) und
Port 6.6 (SCL: serial Clock): Ansteuerung des seriellen IIC-Bus-EEPROM
- Port 6.7: MON#-Signal (Umschaltung zwischen Monitor- und Run-Mode)

Portleitungen mit eingeschränkter Funktion:

- Port 0: wird als Datenbus verwendet
- Port 2: wird als Adressbus verwendet
- Port 3.6 (WR#) und
Port 3.7 (RD#): Steuersignale für Adress- und Datenbus
- Port 3.2: zusätzlich mit Power-Fail Interrupt belegt
- Port 8.6: #TCO# Datensignal Funkuhr

Tabelle 2-2 : Pinbelegung des CM509

| Stecker | Pin | Signale | Bedeutung | |
|---------|--------|-------------------|--|----|
| PL1 | 1, 2 | VCC | Betriebsspannung +5V | 1) |
| PL1 | 3, 4 | GND | Masse | 1) |
| PL1 | 5..8 | (*) | reserviert | 2) |
| PL1 | 9..16 | P1.0..P1.7 | µC-Port P1.0..P1.7 (digitale Ein-/Ausgänge) | |
| PL1 | 17..22 | P3.0..P3.5 | µC-Port P3.0..P3.5 (digitale Ein-/Ausgänge) | |
| PL1 | 23..24 | (*) | reserviert | 2) |
| PL1 | 25..32 | P4.0..P4.7 | µC-Port P4.0..P4.7 (digitale Ein-/Ausgänge) | |
| PL1 | 33..40 | P5.0..P5.7 | µC-Port P5.0..P5.7 (digitale Ein-/Ausgänge) | |
| PL1 | 41..46 | P8.0..P8.5 | µC-Port P8.0..P8.5 (digitale Eingänge) | |
| PL1 | 47 | TCO# | µC-Port P8.6: Time Code Output der Funkuhr (TCR) | |
| PL1 | 48 | P9.1 | µC-Port P9.1 (digitaler Ein-/Ausgang) | |
| PL1 | 49..56 | P9.0..P9.7 | µC-Port P9.0..P9.7 (digitale Ein-/Ausgänge) | |
| PL1 | 57..62 | (*) | reserviert | 2) |
| PL2 | 1, 2 | VCC | Betriebsspannung +5V | 1) |
| PL2 | 3, 4 | GND | Masse | 1) |
| PL2 | 5 | TXD0 | Sendeleitung RS232 (1. µC-interne ser. Schnittstelle) | |
| PL2 | 6 | RXD0 | Empfangsleitung RS232 (1. µC-interne serielle Schnittstelle) | |
| PL2 | 7 | TXD1 | Sendeleitung RS232 (2. µC-interne ser. Schnittstelle) | |
| PL2 | 8 | RXD1 | Empfangsleitung RS232 (2. µC-interne serielle Schnittstelle) | |
| PL2 | 9,10 | RS485A, RS485B | RS485-Schnittstelle Leitungen A und B | |
| PL2 | 11,12 | CANTX, CANRX | CAN-Bus-Schnittstelle (Anschluss an den CAN-Transceiver) | |
| PL2 | 13 | CSEX1# | Chip Select 1. Hardwareerweiterung | |
| PL2 | 14 | (*) | reserviert | 2) |
| PL2 | 15 | CSEX2# | Chip Select 2. Hardwareerweiterung | |
| PL2 | 16 | CSWDT | Chip Select Watch Dog Timer: kann für externe Hardwareerweiterungen eingesetzt werden | |
| PL2 | 17 | BATIN | Einspeisung RAM und Echtzeituhr Pufferspannung (RTC) | |
| PL2 | 18 | (*) | reserviert | 2) |
| PL2 | 19 | VBAT | Ausgang Batterie Pufferspannung | |
| PL2 | 20 | STDP | Standard Pulse: Takt-Ausgang des Echtzeituhrenbaustein (RTC) | |
| PL2 | 21 | MR# | Manual Reset Input (low-aktiv): für Taster nach GND | |
| PL2 | 22 | RESO# | Reset-Ausgang des µC (low-aktiv) | |
| PL2 | 23 | RESET | Reset-Ausgang (high-aktiv) | |
| PL2 | 24 | RESET# | Reset-Ausgang des Überwachungsbausteins (Supervisory: low-aktiv) | |
| PL2 | 25 | VPPF | Programmierspannung für Flash-Speicher | |
| PL2 | 26 | BOOT# | Aktivierung des Bootstrap-Loaders (low-aktiv) | |
| PL2 | 27 | VPPC | Aktivierung des Sector Protection/Unprotection-Features des SAB-C509-16F(internes Flash) | |
| PL2 | 28 | ENMEM# | Freigabesignal für modul-externe Speicher oder Peripherie-Bausteine | |

| Stecker | Pin | Signale | Bedeutung | |
|---------|--------|------------|---|----|
| PL2 | 29 | (*) | reserviert | 2) |
| PL2 | 30 | WR# | Datenspeicher-Schreibsignal des μC (für modul-externe Speicher verwendbar) | |
| PL2 | 31 | PSEN# | Programmspeicher-Lesesignal des μC (für modul-externe Speicher verwendbar) | |
| PL2 | 32 | RD# | Datenspeicher-Lesesignal des μC (für modul-externe Speicher verwendbar) | |
| PL2 | 33 | (*) | reserviert | 2) |
| PL2 | 34 | ALE | Adress Latch Enable des μC | 3) |
| PL2 | 35..38 | GND | Masse | |
| PL2 | 37..54 | A15..A0 | Adressbus A15..A0 | |
| PL2 | 55..62 | D0..D7 | Datenbus D0..D7 (gemultiplexte Signale) | |
| PL3 | 1..8 | P7.0..P7.7 | Analog- oder Digitaleingänge des μC | |
| PL3 | 9 | (*) | reserviert | 2) |
| PL3 | 10 | P6.0 | μC -Port P6.0: (ADST#: AD-Start) Start des μC -internen AD-Wandlers | |
| PL3 | 11 | VA+ | Referenzspannung für A/D-Wandler | |
| PL3 | 12 | VA- | Referenzmasse für A/D-Wandler | |
| PL3 | 13 | VCC | Betriebsspannung +5V | 2) |
| PL3 | 14 | GND | Masse | 2) |
| PL6 | 1 | ANT1 | Antennenanschluss des Funkuhrenbausteins (TCR) | |
| PL6 | 2 | ANT2 | Antennenanschluss des Funkuhrenbausteins (TCR) | |

- 1) Alle Betriebsspannungsanschlüsse (bzw. Masseanschlüsse) des Mikrocontrollermoduls sollten von der Basisträgerplatine gespeist werden, um eine möglichst niederohmige Versorgungsspannungseinspeisung zu erreichen.
- 2) Diese Leitungen können bei anderen Modulen der CM-Serie belegt sein. Wenn in Ihren Applikationen zu einem späteren Zeitpunkt ein leistungsfähigeres Modul eingesetzt werden soll, dürfen diese Anschlüsse nicht belegt werden.
- 3) Diese Leitungen werden von anderen Modulen der CM-Serie teilweise nicht unterstützt. In Applikationen sollten diese Leitungen daher nur verwendet werden, wenn der Einsatz von Modulen mit Controllern anderer Controllerfamilien nicht in Betracht kommt.

3. Applikationshardware

Bei der Realisierung Ihrer Applikationshardware sollten einige wesentliche Punkte beachtet werden:

- Batteriepufferung: Bei der Verwendung eines Akkus zur Pufferung muss eine entsprechende Ladeelektronik auf der Trägerkarte des Moduls realisiert werden, da der Eingang BATIN für Lithiumbatterien vorgesehen ist.
- Flash-PROMs: Sofern Flash-Speicher mit einer Programmierspannung von 12V eingesetzt werden, muss diese extern zugeführt werden. Auf entsprechende Ein- und Ausschaltmöglichkeiten ist zu achten. Gleiches gilt bei der Verwendung eines Mikrocontrollers mit integriertem Flash-PROM.
- Versorgungsspannung: Die Betriebsspannung des Moduls sollte über alle entsprechenden Pins angeschlossen werden, um eine möglichst niederohmige Einspeisung zu erhalten.
- Unbelegte Stecker-Pins: Unbelegte Pins auf den Steckern sollten von Ihrer Applikationshardware nicht beschaltet werden. Diese können von späteren Modulversionen verwendet werden.

4. Speicherausbau

Das Controller-Modul CM509 kann mit Flash-(oder wahlweise mit EPROM) und SRAM-Speichern im Dual-In-Line-Gehäuse bestückt werden.

Das System verfügt in der Maximalkonfiguration über folgenden Speicherausbau:

- Datenspeicher 128 kByte RAM mit optionaler Pufferung mittels extern anzubringender Lithiumbatterie. 40 kByte stehen direkt zur Verfügung. Das gesamte SRAM (128 kByte) wird in 16 8 kByte große Seiten (Pages) eingeteilt und kann über ein entsprechend großes Fenster adressiert werden.
- 128 kByte Flash-EPROM. Der Flash-Speicher kann in der Schaltung programmiert werden. Davon werden die ersten 64 kByte als Programmspeicher verwendet. Die restlichen 64 kByte werden als Datenspeicher adressiert, insbesondere für Konstante z.B. Display-Texte in mehreren Sprachen.

4.1. Datenspeicher U4 (SRAM)

Vom SRAM ist ein 40 kByte großer Adressteil (0000h-9FFFh) als Datenspeicher immer verfügbar (Standard SRAM). In diesem Speicher werden die allgemeinen Programmdateien gehalten. In diesen Bereich kann auch indirekt über das RAM-Fenster (Page 0-4) zugegriffen werden. Im Monitormodus wird in diesem Baustein zusätzlich der komplette Programmcode gespeichert. Da in diesem Modus Programm und Daten physikalisch im gleichen Baustein gespeichert sind, muss eine Adressüberlappung durch entsprechende Compiler- bzw. Linkersteuerbefehle unbedingt vermieden werden.

Weitere 88 kByte sind nur über jeweils 8 kByte-große Einzelseiten (Page 5-15) adressierbar. Bei der Adressierung muss jeweils im Vorab die Nummer der zu adressierenden Seite ausgegeben werden. Erst danach ist ein Zugriff auf die Daten der 8 kByte-großen Page möglich.

4.2. Programm-Speicher U5 (Flash-EPROM)

Das Flash-Prom dient im wesentlichen als Programmspeicher für die Applikation. Für diesen Zweck stehen im Flash-Run-Modus (Normalbetrieb) insgesamt 64 kByte Programmspeicher zur Verfügung. Weitere 64 kByte können in diesem Modus in Form von 8 kByte großen Blöcken in den Datenspeicher des Systems eingeblendet werden. In diesem Speicherbereich können Konstante für das Programm gehalten werden. Ideal ist dieser Speicher für Konstante, die in Abhängigkeit einer bestimmten Systemkonfiguration benötigt werden. Als Anwendungsbeispiel kann hier die Speicherung von LCD-Texten angeführt werden. Bei Applikationen, die in mehreren Ländern eingesetzt werden sollen, kann über die Auswahl der Sprache gleichzeitig die Auswahl der entsprechenden Seitennummer erfolgen. Die einzelnen Texte können vom Applikationsprogramm jetzt ohne Berücksichtigung der Sprache (für alle Sprachen absolut identisch) adressiert werden.

Das gesamte Flash-EPROM (alle 16 Pages) kann in beiden Betriebsmodi (Normalbetrieb → „Flash-Run-Mode“ und Monitorbetrieb) des CM509 über das 8 kByte große Flash-Fenster programmiert und auch gelesen werden.

4.3. Auswahl der Speicherseite

Die Auswahl der Speicherseite erfolgt über das Paging-Register (Baustein U11). Dieses 1-Byte-große Register kann über einen beliebigen Schreibzugriff in den Adressbereich E000h-E0FFh (→ Peripherie-Bereich) beschrieben werden.

Ein Lesezugriff in diesen Adressbereich bewirkt ein Triggern des externen Watchdog-Timers. Der gelesene Wert entspricht keinesfalls dem Wert des Paging-Registers.

Mit den Bits 0..3 (Lower Nibble) wird die Flash-Seite und den Bits 4..7 (Higher Nibble) die RAM-Seite ausgewählt. Die Seitennummer ist dabei binär codiert. Die jeweils selektierte Seite wird immer an der gleichen Adresse des Speichers eingeblendet (RAM-Fenster: A000h-BFFFh; Flash-Fenster: C000h-DFFFh).

Tabelle 4-1: Schreibzugriff in Adressbereich E000h-E0FFh

| Adressbereich | RPAGE | | | | FPAGE | | | |
|---------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| | RPAGE3 | RPAGE2 | RPAGE1 | RPAGE0 | FPAGE3 | FPAGE2 | FPAGE1 | FPAGE0 |
| E000h-E0FFh | | | | | | | | |
| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Wertigkeit | $2^3 = 8$ | $2^2 = 4$ | $2^1 = 2$ | $2^0 = 1$ | $2^3 = 8$ | $2^2 = 4$ | $2^1 = 2$ | $2^0 = 1$ |

RPAGE3..RPAGE0: Auswahl der binär codierten Seitennummer für den SRAM-Speicher ($RPAGE = RPAGE3 * 2^3 + RPAGE2 * 2^2 + RPAGE1 * 2^1 + RPAGE0 * 2^0$). Bei Maximalkonfiguration (128 kByte) stehen insgesamt 16 Seiten mit je 8 kByte zur Verfügung. Die ausgewählte Page wird im RAM-Fenster (Adressbereich: A000h-BFFFh) eingeblendet.

FPAGE3..FPAGE0: Auswahl der binär codierten Seitennummer für den Flash-Speicher ($FPAGE = FPAGE3 * 2^3 + FPAGE2 * 2^2 + FPAGE1 * 2^1 + FPAGE0 * 2^0$). Bei Maximalkonfiguration (128 kByte) stehen insgesamt 16 Seiten mit je 8 kByte zur Verfügung. Die ausgewählte Page wird im Flash-Fenster (Adressbereich: C000h-DFFFh) eingeblendet.

Es muss unbedingt beachtet werden, dass durch Ausgabe neuer Seitennummern immer beide Speicher gemeinsam konfiguriert werden. Soll z. B. die Flash-Page unabhängig von der RAM-Page konfiguriert werden, so muss eine Kopie des Wertes im Paging-Register in einer Variablen gespeichert werden, da das Paging-Register nicht gelesen werden kann. Des Weiteren muss der gewünschte Speicherbereich maskiert werden. Soll beispielsweise für den RAM-Speicher die fünfte Seite aktiviert werden, so würde die dafür gültige Sequenz in der Programmiersprache „C“ folgendermaßen lauten:

```

... // vorhergehende Befehlssequenzen
unsigned char PagingReg; // Kopie des Paging-Registers mit Bytegröße anlegen!
XBYTE[0xE000]=PagingReg=0x... ; // Beim Initialisieren des Paging-Registers immer die
// Kopie mitinitialisieren!
... // zwischenliegende Befehlssequenzen
PagingReg &= 0x0F // lösche Einstellung fuer RAM-Seitenauswahl
// (RAMPAGE=0)
XBYTE[0xE000]=PagingReg|=0x50; // Setze Seite 5 fuer RAM-Page, Flash-Page bleibt
// unverändert, immer Paging-Register und Kopie
// konsistent halten!
... // nachfolgende Befehlssequenzen

```

Innerhalb der Seiteneinstellungssequenz darf die Seiteneinstellung nicht durch Interruptroutinen verändert werden. Entweder muss dies durch Abschaltung dieser Interrupts vermieden werden, oder alle Umschaltungen der Speicherseiten müssen innerhalb der gleichen Softwareprioritätsebene ausgeführt werden.

4.4. Das Paging-Register

Mit Hilfe des Paging-Registers kann die RAM- und Flash-Page eingestellt werden, auf die sich nachfolgende Zugriffe in das RAM- oder Flash-Fenster beziehen.

Tabelle 4-2: Adressbereiche des RAM- und Flash-Fensters

| Fenster | Adressbereich | Größe |
|---------------|---------------|---------|
| RAM-Fenster | A000h-BFFFh | 8 kByte |
| Flash-Fenster | C000h-DFFFh | 8 kByte |

Auf das 1-byte-große Paging-Register kann nur schreibend zugegriffen werden, wobei jeder Schreibzugriff im Adressbereich **E000h-E0FFh** das Paging-Register beschreibt.

Tabelle 4-3: Adressbereich des Paging-Registers

| Peripherie | Größe | Adressbereich |
|-----------------|--------|---------------|
| Paging-Register | 1 Byte | E000h-E0FFh |

Da das Paging-Register nicht zurückgelesen werden kann, sollte programmtechnisch immer mit einer Kopie gearbeitet werden. (Siehe Beispiel im obigem Kapitel)

Es können jeweils 16 Pages eingestellt werden, wobei jede Page einer Größe von 8 kByte entspricht. Somit wird sowohl im SRAM als auch im Flash mit Hilfe des Paging-Registers der Zugriff auf die vollen 128 kByte ermöglicht.

Das Prinzip des Pagings kann z. B. dafür verwendet werden, die lokalen Daten eines Tasks in einer der Task zugehörigen Page zu halten (RPAGE), während FPAGE dazu verwendet werden kann, das Flash zu programmieren. Programme können nur von den unteren 64 kByte des Flash ausgeführt werden. Der obere 64 kByte-Bereich kann z.B. dazu verwendet werden, sich in den Pages den Text für verschiedene Sprachen zu halten.

Die untersten 5 Seiten des Datenspeichers (RPAGE 0-4) sind identisch mit dem „Standard-RAM-Bereich“.

4.5. Speicherbelegung / Betriebsmodi des CM509

Das CM509 verfügt über 2 Betriebsmodi:

- **„Flash-Run-Mode“:** Normalbetrieb; Applikationsprogramm wird abgearbeitet.
- **Monitor-Mode:** dieser Modus dient zur Entwicklung und zum Testen von Applikationsprogrammen; hierzu steuert ein Monitorprogramm das zu testende Applikationsprogramm; Monitor- und Applikationsprogramm wird abgearbeitet.

Der Betriebsmodus wird mit Hilfe von Portpin P6.7 (Signal MON#) eingestellt .

Tabelle 4-4: Betriebsmodiwahl

| P6.7 = MON# | Betriebsmodus |
|--------------------|----------------------------------|
| 0 | Monitor-Mode |
| 1 | „Flash-Run-Mode“ (Normalbetrieb) |

In Abhängigkeit der beiden Betriebsmodi werden die physikalischen Speicherbausteine in teilweise unterschiedliche logische Adressbereiche des Controllers eingeblendet. Die Einteilung des Speichers ergibt sich nach folgender Tabelle:

Tabelle 4-5: Speicherorganisation

| Betriebsmodus | „Flash-Run-Mode“ | Monitor-Mode |
|--------------------|--|--|
| P6.7 = MON# | 1 | 0 |
| Flash | Code: 0000h - FFFFh | - |
| Flash-Fenster | Data: C000h -DFFFh Schreib- und Lesezugriff auf das gesamte Flash (16 Pages) möglich! | Data: C000h -DFFFh Schreib- und Lesezugriff auf das gesamte Flash (16 Pages) möglich! |
| Flash-Seiten-Größe | 8 kByte | 8 kByte |
| SRAM | Data: 0000h - 9FFFh | Data: 0000h - 9FFFh Code: 0000h - FFFFh |
| SRAM-Fenster | Data: A000h -BFFFh Zugriff auf das gesamte SRAM (16 Pages) möglich! | Data: A000h -BFFFh Zugriff auf das gesamte SRAM (16 Pages) möglich! |
| SRAM-Fenster-Größe | 8 kByte | 8 kByte |

Data: In diesem Bereich können Daten gespeichert werden. Der μ C adressiert diesen Bereich mit den Steuersignalen RD# und WR#.

Code: In diesem Bereich liegt der Programmcode. Die Adressierung erfolgt mit dem Steuersignal PSEN# .

Allgemeine Hinweise zur Speicherbelegung:

- In die mit Flash- und RAM-Fenster gekennzeichneten Adressbereiche wird immer eine 8 kByte große Seite des physikalischen Speicherbausteins eingeblendet. Die Seite muss vorher, wie in Kapitel **** „Auswahl der Speicherseiten“ beschrieben, ausgewählt werden.
- Will der Anwender die Betriebsmodi des Controllers SAB C509 (Operating Modes: - External / Internal Mode, - External / Internal XRAM Mode, - External / Internal Bootstrap Mode, - External / Internal Programming Mode) nutzen, so sind Code- und Datenspeicher entsprechend der Spezifikation im User's Manual SAB C509 vertauscht.
- Der volle Zugriff auf alle Pages im SRAM und Flash (unabhängig vom Betriebsmodus) birgt Vorteile jedoch auch gewisse Gefahren. So sollte z. B. ein Schreibzugriff in die unteren 5 SRAM-Pages sorgfältig durchdacht werden, da dieser Standard-RAM-Bereich auch direkt zugänglich ist und so eventuell ungewollt Daten überschrieben werden.

4.6. Serielles IIC-Bus-EEPROM U6

Mit U6 sind auf dem Controllermodul 512 Byte serielles IIC-Bus-EEPROM untergebracht. Dieser verhältnismäßig kleine Speicher dient zur Speicherung von Setup-Daten oder ähnlichem. Insbesondere ist dieses EEPROM bei Applikationen interessant, bei denen die Menge der resistent zu speichernden Daten so klein ist, dass sich der Einsatz von Flash-Speichern nicht lohnt.

Das EEPROM wird nach der IIC-Bus-Spezifikation angesteuert, wobei der Mikrocontroller P6.6 als Takt (SCL: Serial Clock) und der Port P6.5 als Daten-Ein / Ausgang (SDA: Serial Data) verwendet. Auf Grund der seriellen Datenübertragung können in diesem EEPROM nur zeitunkritische Daten gespeichert werden; es ist für 100.000 Schreibzyklen ausgelegt.

5. Peripherie

Auf dem Modul CM509 ist ein breites Spektrum an Peripherie-Einheiten untergebracht.

5.1. Serielle Schnittstellen

Das Mikrocontrollermodul CM509 bietet zwei serielle Schnittstellen für asynchrone Datenübertragung. Es handelt sich hierbei um die im Mikrocontroller implementierten Schnittstellen, die über Pegelumsetzer dem RS232- bzw. RS485-Standard angepasst sind.

Mit Hilfe des Jumpers J4 kann die 2. serielle Schnittstelle alternativ entweder als RS232- oder RS485-Schnittstelle konfiguriert werden. Als Pegelumsetzer (Transceiver) dienen die Bausteine U7 (MAX232CSE) und U6 (MAX485CSE).

Die RS232- und RS485-Schnittstellensignale sind über den Stecker PL2 zugänglich.

5.1.1. Serielle Schnittstelle SIO0

Die Schnittstelle SIO0 kann im Gegensatz zur SIO1 nur als RS232-Schnittstelle betrieben werden. Die Schnittstellensignale sind an folgenden Anschlüssen am Stecker PL2 des Moduls verfügbar:

Tabelle 5-1: Pinbelegung SIO0

| Stecker | Pin Nr. | Signal | Bedeutung |
|---------|---------|--------|-----------------------|
| PL2 | 5 | TXD0 | Sendeleitung RS232 |
| PL2 | 6 | RXD0 | Empfangsleitung RS232 |

5.1.2. Serielle Schnittstelle SIO1

Die im Controller integrierte Schnittstelle SIO1 kann mit Jumper J4 alternativ als RS232-Schnittstelle (voll duplex) oder RS485-Schnittstelle (halbduplex) konfiguriert werden.

Tabelle 5-2: Hardware-Konfiguration SIO1

| Schnittstellenstandard | Jumper J4 | Wirkung |
|------------------------|-----------|--|
| RS232 | Pos. a | RS232 Empfangsleitung wird mit μ C verbunden |
| RS485 | Pos. b | RS485 Empfangsleitung wird mit μ C verbunden |

Die Sendeleitung des μ C (P6.2: TXD2) ist immer mit beiden Pegelumsetzern U7 und U8 verbunden.

Beim Betrieb als RS485-Schnittstelle muss der serielle Sende-Kanal des RS485-Treibers (U23: MAX485) vom Controller mit Portpin P6.4 freigegeben werden. Der RS485-Empfangskanal ist von der Hardware grundsätzlich freigeschaltet.

Tabelle 5-3: RS485-Sendetreiberfreigabe (Software)

| Port P6.4 | Bedeutung DEN# (DriverEnable#) |
|-----------|--|
| 1 | RS485-Sendetreiber inaktiv (Ausgänge RS485A, RS485B hochohmig) |
| 0 | RS485-Sendetreiber aktiv (Ausgänge RS485A, RS485B werden gemäß RS485-Pegeln getrieben) |

Die Schnittstellensignale sind an folgenden Anschlüssen am Stecker PL2 des Moduls verfügbar:

Tabelle 5-4: Pinbelegung SIO1

| Port P6.4 | Pin Nr. | Signal | Bedeutung |
|-----------|---------|--------|-----------------------|
| 1 | 7 | TXD1 | Sendeleitung RS232 |
| 0 | 8 | RXD1 | Empfangsleitung RS232 |

5.2. CAN-Schnittstelle

Die CAN-Schnittstelle wird mit Hilfe des Standalone Full-CAN-Controllers SAE81C91 (U9) von Siemens realisiert.

Dieser Controller ermöglicht Datenraten bis zu 1 Mbaud. Er entspricht der CAN-Spezifikation V2.0B passive.

Der SAE81C91 besitzt 256 interne Register, die vom Mikrocontroller im externen Datenbereich von **E100h-E1FFh** anzusprechen sind (memory mapped IO). Kernstück des SAE81C91 ist ein Dual-Ported-RAM-Bereich für 16 CAN-Messages (bis zu 8 Datenbytes). Zu jeder Message existieren noch 2 Bytes für den Identifier und den Data Length Code (Anzahl

der Datenbytes). Er bietet einen "quasitransparenten" Zugriff auf den CAN-Bus. Zu sendende Messages werden einfach in diesen dual-ported-RAM-Bereich hineingeschrieben und ein entsprechendes Transmittbit gesetzt. Auf diese Weise ist es möglich die Übertragung mehrerer Messages (bis zu 16) mit einem Kommando einzuleiten.

Die komplette Umsetzung in die CAN-Spezifikation wird vom CAN-Controller übernommen. Für die Messages, die empfangen werden sollen, muss der entsprechende Identifier (ID) und Data Length Code (DLC) geschrieben und ein zur Message gehörendes Receive-Flag gesetzt werden. Der Identifier wirkt auf diese Weise wie ein Filter, welches nur ankommende Messages mit diesem Identifier ins dual-ported-RAM schreibt. Wurde über entsprechende RegisterEinstellungen der allgemeine und individuelle Receiver-Interrupt freigegeben und Jumper J8 gesetzt, so wird über Port P3.3 (INT1#) ein Interrupt ausgelöst (falls controllerseitig freigegeben).

Tabelle 5-5: Jumper J8, Nutzung des CAN-Controller-Interrupt-Ausgang

| Jumper J8 | Wirkung |
|--------------------|---|
| gebrückt (gelötet) | Port P3.3 ist mit dem Interrupt-Ausgang des CAN-Controllers verbunden |
| offen | Port P3.3 kann frei benutzt werden |

Weitere Features des Bausteins sind:

- Basic-CAN-fähig: Über Message 0 können alle Messages auf dem CAN-Bus abgehört werden. Dadurch ist ein Überwachen des Busses möglich.
- Zusätzlich zur CAN-Spezifikation des BIT-Monitorings werden im Baustein die Übertragung der Messages (vom dual-ported-RAM auf den CAN-Bus) byteweise überprüft und fehlerhafte Übertragungen bei Bedarf gemeldet.
- Die Message ID kann während des Betriebs geändert werden.
- Für 8 Messages werden zusätzlich noch Zeitmarken (Time Stamps) gespeichert, um Rückschlüsse auf die Aktualität der Daten zu ermöglichen.

Die Anbindung des CAN-Controllers an den physical Layer muss mit Hilfe eines CAN-Transceivers auf der Applikationsplatine realisiert werden. Die flexible registerprogrammierbare Schnittstelle des SAE 81C91 erlaubt den Anschluss an die verschiedenen Implementierungen des physical Layers.

Die Schnittstellensignale für den CAN-Transceiver sind an folgenden Anschlüssen am Stecker PL2 des Moduls verfügbar:

Tabelle 5-6: Pinbelegung CAN-Bus

| Stecker | Pin Nr | Signal | Bedeutung |
|---------|--------|--------|---------------------|
| PL2 | 11 | CANTX | Sendeleitung CAN |
| PL2 | 12 | CANRX | Empfangsleitung CAN |

5.3. Echtzeituhr (RTC: Real Time Clock)

Auf dem Mikrocontrollermodul CM509 ist mit U10, ein Echtzeituhrenbaustein vom Typ RTC72423 der Firma Seiko Epson, vorgesehen. Er ist am Stecker PL2 zugänglich. Der Baustein unterstützt neben Uhrzeit auch Kalenderfunktionen und eignet sich daher neben Zeitmessungen auch für Logbuchfunktionen.

Tabelle 5-7: Pinbelegung RTC

| Stecker | Pin Nr | Signal | Funktion |
|---------|--------|--------|---|
| PL2 | 20 | STDP | Interrupt-Ausgang des RTC für periodische Ausgangssignale |
| PL2 | 17 | BATIN | Eingang für Batterie zur Pufferung |

Die Echtzeituhr wird grundsätzlich mit der Lithiumbatterie gepuffert, sofern am Modul eine Batteriespannung über Signal **BATIN** (Stecker PL2 Pin17) angelegt wird.

Außer der eigentlichen Uhrenfunktion kann der RTC72423 mit dem Signal **STDP** (Standard Pulse) ein periodisches Signal ausgeben, z. B. für die Erzeugung von periodischen Interrupts (jede Stunde, Minute, Sekunde oder alle 1/64 Sekunden). Dieses Signal liegt am Stecker PL2 Pin20 und kann über die Träger-Platine auf einen Interrupt-Eingang des μC geführt werden. Eine Verbindung on Board ist nicht vorgesehen.

Die 16 Register des Echtzeituhrenbausteins sind standardmäßig im Adressbereich von **E200h-E2FFh** eingeblendet (Memory-Mapped-IO). Sie wiederholen sich in diesem Adressbereich alle 16 Byte. Dies bedeutet, dass z. B. das Einer-Sekunden-Register (Register 0) über die Adressen E200h, E210h, E220h usw. gelesen werden kann.

5.4. Supervisory

Mit Hilfe von U2 (MAX791) wurde auf dem CM509 eine Kombination aus Spannungsüberwachungs- und Resetschaltung, Schreibschutzverriegelung (für das SRAM) und ein zusätzlicher Watchdog implementiert.

5.4.1. Überwachungs-, Resetschaltung und Schreibschutzverriegelung

Auf dem Mikrocontrollermodul CM509 ist mit U2 eine Überwachung der Versorgungsspannung untergebracht. Die Betriebsspannung des Controllermoduls wird permanent überwacht. Bei Unterschreitung der Schwelle von typ. 4.8 V wird vom MAX791 der Port-Pin P3.2 (und damit dem Interrupt-Eingang INT0#) am Mikrocontroller auf Low gezogen. Auf diese Weise kann bereits vor dem endgültigen Ausfall der Betriebsspannung ein Interrupt generiert werden. Damit kann ein Absinken der Spannung rechtzeitig erkannt werden und die Rettung von wichtigen Daten in das mit der Lithiumbatterie gepufferte SRAM oder ins serielle EEPROM veranlasst werden. Der Portpin P3.2 ist somit nicht mehr für Ein- und Ausgabezwecke zur Verfügung.

Wenn die Auswertung der Spannungsüberwachung gewünscht wird, so dass Ausfälle der Versorgungsspannung frühzeitig erkannt werden können, so muss der Interrupt-Eingang auf Low-Potential- oder auf negative Flankentriggerung initialisiert werden, da die Spannungsüberwachung im Normalzustand einen High-Pegel liefert. Zusätzlich sollte der Interrupt-Eingang INT0# auf die höchste Prioritätsebene gelegt und auch freigegeben werden.

Bei weiterem Absinken der Versorgungsspannung aktiviert die Überwachungsschaltung die Schreibschutzverriegelung des pufferbaren SRAM-Bausteins (U4) und bildet ein Reset-Signal (Reset#) für den μ C. Zusätzlich wird die Spannungsversorgung des SRAMs und der Echtzeituhr automatisch auf Lithiumbatterie umgeschaltet, sofern eine entsprechende Batteriespannung am Eingang BATIN (Stecker PL2 Pin 17) anliegt. Die Batteriespannung sollte ca. 3..4 V betragen.

5.4.2. Zusätzlicher Watchdog

Um weitere Systemsicherheit durch Überwachung des Programmlaufs zu erhalten, wurde zusätzlich, zu dem im μ C integrierten Watchdog, auch der im Supervisory-Baustein MAX791 integrierte Watchdog-Timer zugänglich gemacht. Dieser muss jedoch mit Hilfe von Jumper J5 freigegeben werden. Der Watchdogtimer wird durch Brücken des Jumpers J5 aktiviert. Er kann dann nicht mehr softwaremäßig vom μ C ein- oder ausgeschaltet werden. Der WatchDog-Mechanismus wird durch einen offenen Jumper J5 außer Funktion gesetzt. Es wird dann kein Reset-Signal erzeugt.

Sofern Jumper J5 geschlossen ist, beträgt die Watchdog-Timeout-Periode 1,6 sec. Wird der Watchdog nicht innerhalb dieser Zeit neu angestoßen, so wird automatisch ein Reset-Signal (Reset#) für den Mikrocontroller gebildet.

Der Anstoß des Watchdogtimers erfolgt durch einen Lesezugriff auf den Adressbereich (**E000h-E0FFh**) des Watchdog-Timers (CSWDT-Signal: ChipSelectWatchDogTimer) im Bereich der Peripherie (Memory-Mapped-IO). Dem gelesenen Byte ist keine Bedeutung beizumessen, da durch die internen Pull-Up-Widerstände von Port 0 (gemultiplexer Adress-/Datenbus) FFh gelesen wird.

5.5. Externe Hardware-Erweiterungen über den Datenbus

Für externe Hardware-Erweiterung über den Datenbus sind bereits zwei Chip-Select-Signale (CSEX1# u. CSEX2#) vorgesehen. Das Signal CSEX1# umfasst 256 Bytes und ist dem Adressbereich E300h-E3FFh zugeordnet, während CSEX2# 3 kBytes und Adressbereich E400h-EB0FFh zugeordnet ist.

Tabelle 5-8: Pinning u. Adresslagen externer HW-Erweiterungen

| Signal | Pin Nr. | Adressbereich | Größe |
|--------|---------|---------------|----------|
| CSEX1# | 13 | E300h-E3FFh | 256 Byte |
| CSEX2# | 15 | E400h-EBFFh | 3 kByte |

Jeder Schreib- oder Lesezugriff in die entsprechenden Adressbereich aktiviert das jeweilige Chip-Select-Signal.

Funkuhr U15 (TCR: Time Code Receiver)

Die Funkuhr ist optional und wurde wegen Platzmangels auf der Modulunterseite als einzige SMD-Baugruppe auf der Oberseite platziert.

Die notwendige Ferritantenne ist extern anzubringen und kann bei Firma frenzel + berg electronic GmbH & Co.KG bezogen werden. Sie ist an Stecker PL6 anzuschließen .

Tabelle 5-9: Steckerbelegung PL6 (Funkuhrantenne)

| Stecker | Pin Nr | Signal |
|---------|--------|--------|
| PL6 | 1 | ANT1 |
| PL6 | 2 | ANT2 |

Das digitale Ausgangssignal TCO# der Funkuhr ist direkt mit Portpin P8.6 verbunden und kann somit vom Mikrocontroller jederzeit ausgewertet werden. Ist die Funkuhr nicht bestückt, ergeben sich für Portpin P8.6 keine Einschränkungen.

Das deutsche Funkuhrensinal wird in Mainflingen (Station DCF 77) mit einer Trägerfrequenz von 77,5 kHz ausgesendet. Das Signal wird ununterbrochen gesendet und enthält die vollständige Zeit- und Datumsinformation. Der Zeitrahmen der kompletten Information beträgt eine Minute, wobei insgesamt 60 Bits mit je einer Sekunde Dauer ausgestrahlt werden. Mit dem Beginn jeder Sekunde stellt sich ein High-Pegel mit 100 ms (binäre Null) oder 200 ms (binäre Eins) Länge ein. Der Zeitrahmen enthält die BCD-codierte Information der Minuten, Stunden, Kalendertag, Wochentag, Monat und Jahr. Nähere Informationen können sie dem Datenblatt des U4224B entnehmen.

Die Funkuhr kann auch für den Einsatz in Japan und England bestückt werden.

6. Decoder

Das Controllermodul CM509 verwendet als Decoder programmierbare Logikbausteine vom Typ GAL16V8. Hierbei wird für den RAM-, Flash- und Peripheriedecoder jeweils ein GAL-Baustein benutzt.

6.1. RAM-Decoder U14

Der RAM-Decoder-Baustein U14 ist ohne Sockel bestückt und kann nicht geändert werden. Er bildet aus den Eingangssignalen A13, A14, A15, RD#, WR#, PSEN#, MON#, RPAGE0, RPAGE1, RPAGE2 und RPAGE3 die Ausgangssignale RA13, RA14, RA15, RA16, RDOUT# und CSRI#.

Die RPAGE-Signale beinhalten die im Paging-Register eingestellte RAM-Page. Das Signal Mon# wird für die Umschaltung zwischen Monitorbetrieb und Normalbetrieb benötigt und kann softwaremäßig mit Portpin P6.7 bedient werden. Die restlichen Eingangssignale sind die direkten Signale des μ C.

Mit dem Ausgangssignal RDOUT# wird ein generelles Auslesesignal für Speicherbausteine gebildet, das sowohl falls PSEN# oder RD# aktiv ist, das Auslesen initiiert. Das Signal CSRI# (Chip-Select-RAM-In) ist das noch nicht schreibschutzverriegelte RAM-Chip-Select-Signal und führt auf den Supervisory-Baustein, der es dann schreibschutzverriegelt und an den RAM-Baustein U4 weitergibt.

Die Signale RA13-RA16 führen direkt auf den RAM-Baustein und entsprechen den Adressen. Sie enthalten die gemultiplexte Information der Controller-Adressen, bzw. der RPAGE-Signale je nach dem ob in das RAM-Fenster oder außerhalb zugegriffen wird. Für diese Multiplexer-Funktion werden die Eingangssignale PSEN#, RD#, WR# u. A13-A15 als Steuersignale verwendet.

6.1.1. PLD-Listing

*IDENTIFICATION

CM509

RAM-Decoder U14:

CSRI: Chip-Select-RAM-Signal

RDOUT: Read-Signal

RA13-RA16: RAM-Adressen 13-16 -> Paging

*DECLARATIONS

X-VAR = 12

Y-VAR = 6

*X-NAMES

A[13..15], RD, WR, PSEN, MON, RPAGE[0..3], CTOGND;

*Y-NAMES

CSRI, RDOUT, RA[13..16];

*RUN-CONTROL

PROGFORMAT = JEDEC;

LISTING = FUSE-PLOT, PINOUT, EQUATIONS;

TESTVECTORS = GENERATE;

*BOOLEAN-EQUATIONS

/CSRI = /MON & /PSEN

+ /A15 & (/RD + /WR)

+ A15 & /A14 & (/RD + /WR);

!Code-Fetch im Mon. Mode

! (0000-7FFF)

! (8000-BFFF)

RDOUT = RD & PSEN;

RA13 = A13 & /PSEN

+ A13 & /A15 & (/RD + /WR)

! (0000-7FFF)

+ A13 & A15 & /A14 & /A13 & (/RD + /WR)

! (8000-9FFF)

+ RPAGE0 & A15 & /A14 & A13 & (/RD + /WR); ! (A000-BFFF) RAM- Fenster

RA14 = A14 & /PSEN

+ A14 & /A15 & (/RD + /WR)

! (0000-7FFF)

+ A14 & A15 & /A14 & /A13 & (/RD + /WR)

! (8000-9FFF)

+ RPAGE1 & A15 & /A14 & A13 & (/RD + /WR); ! (A000-BFFF) RAM- Fenster

RA15 = A15 & /PSEN

+ A15 & /A14 & /A13 & (/RD + /WR)

! (8000-9FFF)

+ RPAGE2 & A15 & /A14 & A13 & (/RD + /WR); ! (A000-BFFF) RAM- Fenster

/RA16 = /PSEN

+ /A15 & (/RD + /WR)

! (0000-7FFF)

+ A15 & /A14 & /A13 & (/RD + /WR)

! (8000-9FFF)

+ /RPAGE3 & A15 & /A14 & A13 & (/RD + /WR); ! (A000-BFFF) RAM- Fenster

*PAL

TYPE = GAL16V8_C8;

*PINS

A13=1, A14=2, A15=3, PSEN=4, RPAGE0=5, RPAGE1=6, RPAGE2=7, RPAGE3=8, WR=9,

GND=10, MON=11, CTOGND=12, RA14=13, RA16=14, RDOUT=15, CSRI=16, RA13=17,

RA15=18, RD=19, VCC=20;

*END

6.2. Flash-Decoder U13

Der Flash-Decoder-Baustein U13 ist ohne Sockel bestückt und kann nicht geändert werden. Er bildet aus den Eingangssignalen A13, A14, A15, RD#, WRF#, WR#, PSEN#, MON#, FPAGE0, FPAGE1, FPAGE2 und FPAGE3 die Ausgangssignale FA13, FA14, FA15, FA16, WRFLS# und CSFLS#.

Die FPAGE-Signale beinhalten die im Paging-Register eingestellte Flash-Page. Das Signal Mon# wird für die Umschaltung zwischen Monitorbetrieb und Normalbetrieb benötigt und kann vom μ C mit Port P6.7 gesteuert werden. Die restlichen Eingangssignale sind die direkten Signale des μ C.

Mit dem Ausgangssignal WRFLS# wird ein Schreibsignal für den Flash-Speicherbaustein gebildet, welches die Programmierung des Flash's über das Flash-Fenster und direkt (mit Hilfe des WRF#-Signals des μ C) im Monitorbetrieb ermöglicht. Das Signal CSFLS# (Chip-Select-Flash) ist das Chip-Select-Signal des Flash-Bausteins.

Die Signale FA13-FA16 führen direkt auf den Flash-Baustein und entsprechen den Adressen. Sie enthalten die gemultiplexte Information der Controlleradressen, bzw. der FPAGE-Signale, je nachdem ob in das Flash-Fenster oder außerhalb zugegriffen wird. Für diese Multiplexer-Funktion werden die Eingangssignale PSEN#, WRF#, RD#, WR# u. A13-A15 als Steuersignale verwendet.

6.2.1. PLD-Listing

*IDENTIFICATION

CM509

Flash-Decoder U13:

CSFLS: Chip-Select-Flash-Signal

WRFLS: Write-Flash-Signal

FA13-FA16: Flash-Adressen 13-16 -> Paging

*DECLARATIONS

X-VAR = 12

Y-VAR = 6

*X-NAMES

A[13..15], RD, WR, PSEN, WRF, MON, FPAGE[0..3];

*Y-NAMES

CSFLS, WRFLS, FA[13..16];

*RUN-CONTROL

PROGFORMAT = JEDEC;

LISTING = FUSE-PLOT, PINOUT, EQUATIONS;

TESTVECTORS = GENERATE;

*BOOLEAN-EQUATIONS

/CSFLS = MON & (/PSEN + /WRF)
+ A15 & A14 & /A13 & (/RD + /WR);

! MON C (0000-FFFF)

! FPR C (C000-DFFF) Flash-
Fenster

/WRFLS = A15 & A14 & /A13 & /WR
+ /WRF;

! (C000-DFFF) Flash-Fenster

FA13 = A13 & (/PSEN + /WRF)
+ FPAGE0 & A15 & A14 & /A13 & (/RD + /WR);

! (C000-DFFF) Flash-Fenster

FA14 = A14 & (/PSEN + /WRF)
+ FPAGE1 & A15 & A14 & /A13 & (/RD + /WR);

! (C000-DFFF) Flash-Fenster

FA15 = A15 & (/PSEN + /WRF)
+ FPAGE2 & A15 & A14 & /A13 & (/RD + /WR);

! (C000-DFFF) Flash-Fenster

/FA16 = /PSEN + /WRF
+ /FPAGE3 & A15 & A14 & /A13 & (/RD + /WR);

! (C000-DFFF) Flash-Fenster

*PAL

TYPE = GAL16V8_C8;

*PINS

FPAGE1=1, FPAGE2=2, FPAGE3=3, PSEN=4, A15=5, A14=6, A13=7, RD=8, MON=9,

GND=10, WR=11, WRF=12, CSFLS=13, FA13=14, FA14=15, FA15=16, WRFLS=17, FA16=18,

FPAGE0=19, VCC=20;

*END

6.3. Peripherie-Decoder U12

Der Peripherie-Decoder-Baustein U12 ist im DIL-Gehäuse mit Sockel bestückt und kann bei Bedarf vom Entwickler einer Applikation mit Hilfe eines GAL-Programmiergeräts geändert werden.

Er bildet aus den Eingangssignalen A8-A15, RD#, WR# und PSEN# die Ausgangssignale CSWDT, CSCAN#, CSEX1#, CSEX2#, CSRTC#, CLKPGL:

- Das Signal CSWDT (Chip-Select-WatchDog-Timer) ist das Triggersignal für den Watchdog des Supervisory-Bausteins. Es wird bei jedem Lesezugriff in den Adressbereich E000h-E0FFh getoggelt.
- Mit Hilfe des Signals CSCAN# (Chip-Select-CAN) wird der CAN-Controller angesprochen. Dies ist bei jedem Schreib- und Lesezugriff in den Adressbereich zwischen E100h und E1FFh der Fall.
- Die Signale CSEX1# (Chip-Select-Extern1) und CSEX2# (Chip-Select-Extern2) stehen dem Entwickler für externe Hardwareerweiterungen zur freien Verfügung und werden bei jedem Schreib-/Lesezugriff in den Adressbereich E300h-E3FFh (CSEX1#) und E400h-EBFFh (CSEX2#) aktiviert.
- Das Signal CSRTC# (Chip-Select-RealTimeClock) aktiviert den Echtzeituhrenbaustein RTC72423 U10 (16 interne Register) und wird bei jedem Schreib- und Lesezugriff in den Adressbereich E200h-E2FFh betätigt. Die Register wiederholen sich alle 16 Bytes.
- Das Signal CLKPGL (Clock-Paging-Latch) wird benötigt, um die Nibbles RPAGE und FPAGE (Selektion der RAM- und Flash-Page) in das Paging-Register einzuspeichern. Dieses „Register“ (1Byte) kann nur beschrieben werden und wird bei jedem Schreibzugriff in den Adressbereich E000h-E0FFh angesprochen.

6.3.1. PLD-Listing

*IDENTIFICATION

CM509

Peripheriedecoder U12:

CSWDT: Watchdog-Timer

CSCAN: CAN-Controller

CSEX1: Chip-Select-Signal 1 zur freien (externen) Verfuegung

CSEX2: Chip-Select-Signal 2 zur freien (externen) Verfuegung

CSRTC: Real Time Clock (Echtzeituhr)

CLKPGL: Clock Paging-Latch

*DECLARATIONS

X-VAR = 12

Y-VAR = 6

*X-NAMES

A[8..15], RD, WR, PSEN, CTOGND;

*Y-NAMES

CSWDT, CSEX2, CSEX1, CSRTC, CSCAN, CLKPGL;

*RUN-CONTROL

PROGFORMAT = JEDEC;

LISTING = FUSE-PLOT, PINOUT, EQUATIONS;

TESTVECTORS = GENERATE;

*BOOLEAN-EQUATIONS

CLKPGL = A15 & A14 & A13 & /A12 & /A11 & /A10 & /A9 & /A8 & /WR; !(E000-E0FF)
&WR

CSWDT = A15 & A14 & A13 & /A12 & /A11 & /A10 & /A9 & /A8 & /RD; !(E000- E0FF)
&RD

/CSCAN = A15 & A14 & A13 & /A12 & /A11 & /A10
& /A9 & A8 & (/RD + /WR); ! (E100-E1FF)

/CSRTC = A15 & A14 & A13 & /A12 & /A11 & /A10
& A9 & /A8 & (/RD + /WR); ! (E200-E2FF)

/CSEX1 = A15 & A14 & A13 & /A12 & /A11 & /A10
& A9 & A8 & (/RD + /WR); ! (E300-E3FF)

/CSEX2 = A15 & A14 & A13 & /A12 & /A11 & A10 & (/RD + /WR)
+ A15 & A14 & A13 & /A12 & A11 & (/RD + /WR); ! (E400-E7FF)
! (E800-EFFF)

*PAL

TYPE = GAL16V8_C8;

*PINS

A9=1, A11=2, A15=3, A10=4, A14=5, A13=6, A12=7, A8=8,

RD=9, GND=10, PSEN=11, WR=12, CSWDT=13, CSEX2=14,

CSEX1=15, CSRTC=16, CSCAN=17, CLKPGL=18, CTOGND=19,

VCC=20;

*END

7. Jumper

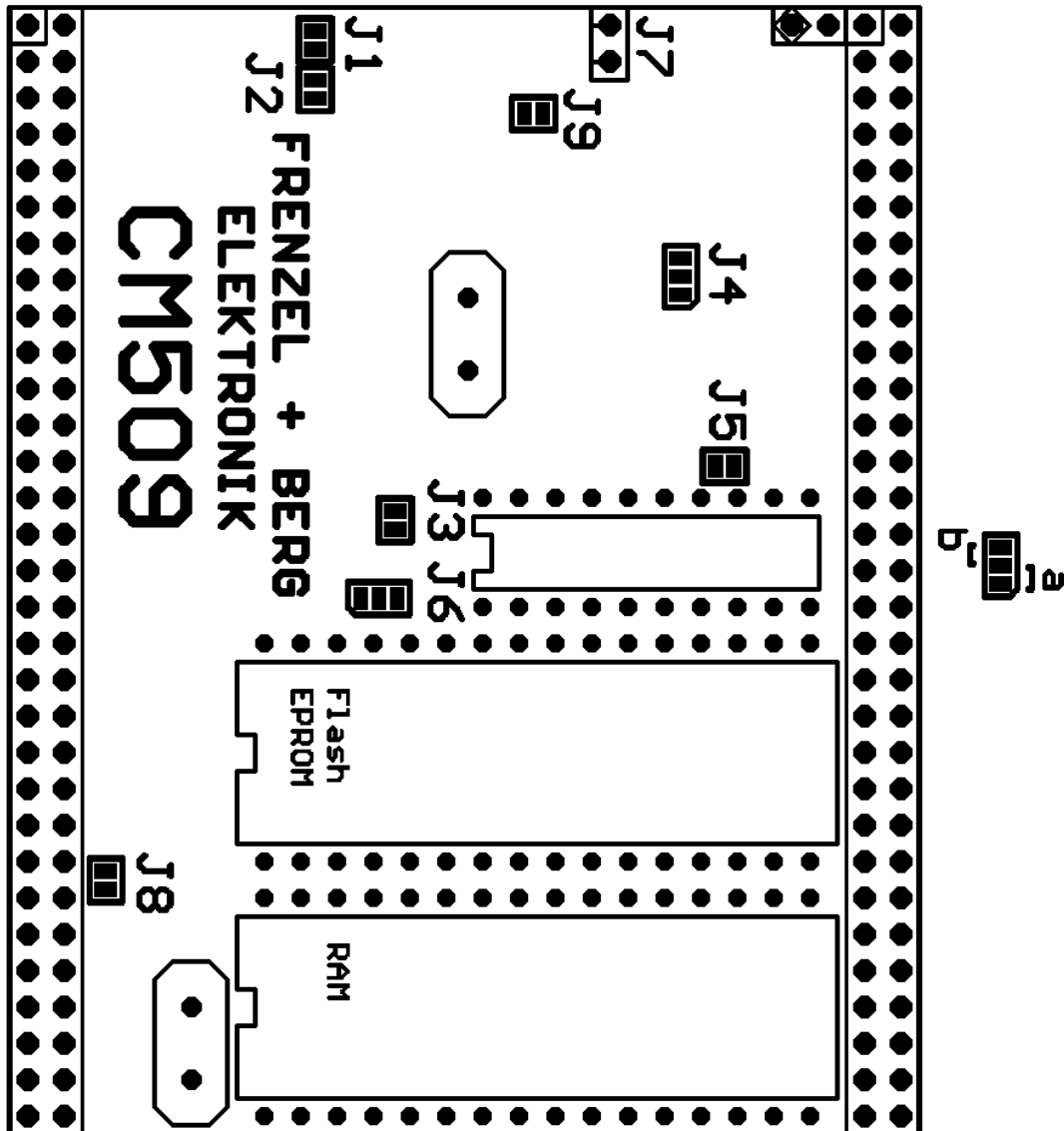
Auf dem Controllermodul CM509 wurden verschiedene Jumper zur Festlegung der Konfiguration bzw. Anpassung der Modulhardware an die jeweiligen Anforderungen vorgesehen. Jumper J7 (Initiierung des Boot-Strap-Mechanismus beim Power-Up oder Reset) ist als zweipolige Stiftleiste zum Aufstecken einer Codierbrücke realisiert. Alle anderen Jumper wurden in Form von Lötbrücken auf der Platinenoberseite ausgeführt.

Tabelle 7-1: Jumperbelegung

| Jumper | Position | Wirkung |
|--------|----------|---|
| J1 | | Referenz für A/D-Wandler |
| | gebrückt | Referenzspannung des A/D-Wandler wird von der Betriebsspannung des Systems abgegriffen |
| | offen | Referenzspannung muss extern über Stecker PL3 Pin 11 zugeführt werden |
| J2 | | Referenz-Masse für A/D-Wandler |
| | gebrückt | Referenz-Masse des A/D-Wandler wird von der Masse des Systems abgegriffen |
| | offen | Referenzmasse muss extern über Stecker PL3 Pin 12 zugeführt werden |
| J3 | | External/Internal ROM-Select |
| | gebrückt | Konfiguration für μC 's mit internem ROM |
| | offen | Konfiguration für μC 's mit externem Programmspeicher |
| J4 | | Konfiguration Serielle Schnittstelle |
| | a | RS232-Schnittstelle |
| | b | RS485-Schnittstelle |
| J5 | | Freigabe des Watchdog-Timers |
| | gebrückt | Watchdog-Timer freigegeben |
| | offen | Watchdog-Timer gesperrt |
| J6 | | EPROM-Typ-Auswahl für U5 |
| | b | Auswahl für EPROM-Typ 27C256: EPROM Pin 1 = VCC |
| | a | Auswahl aller restlichen EPROM's bzw. Flash's |
| J7 | | Bootstrap-Loader |
| | gesetzt | Bootstrap-Loader wird beim Reset aktiviert |
| | offen | Bootstrap-Loader wird beim Reset nicht aktiviert. |
| J8 | | Freigabe des CAN-Interrupts |
| | gebrückt | CAN-Interrupt-Ausgang ist auf Port P3.3 des μC (INT1#) geführt |
| | offen | CAN-Interrupt gesperrt. Port P3.3 des μC (INT1#) kann frei verwendet werden. |
| J9 | | μC Port P3.0 Konfiguration |
| | gebrückt | Port P3.0 ist als Empfangsleitung für serielle RS232-Übertragung konfiguriert. |
| | offen | Port P3.0 steht für Ein-/Ausgabezwecke zur Verfügung |

7.1. Jumperplan

Abbildung 7-1: Jumperplan



8. Technische Daten

8.1. Elektronische und Temperatur Daten

Es gelten grundsätzlich die Betriebsdaten und die Timingspezifikation der eingesetzten Bauelemente, sofern keine andere Spezifikation erfolgt.

Tabelle 8-1: Elektronische Daten

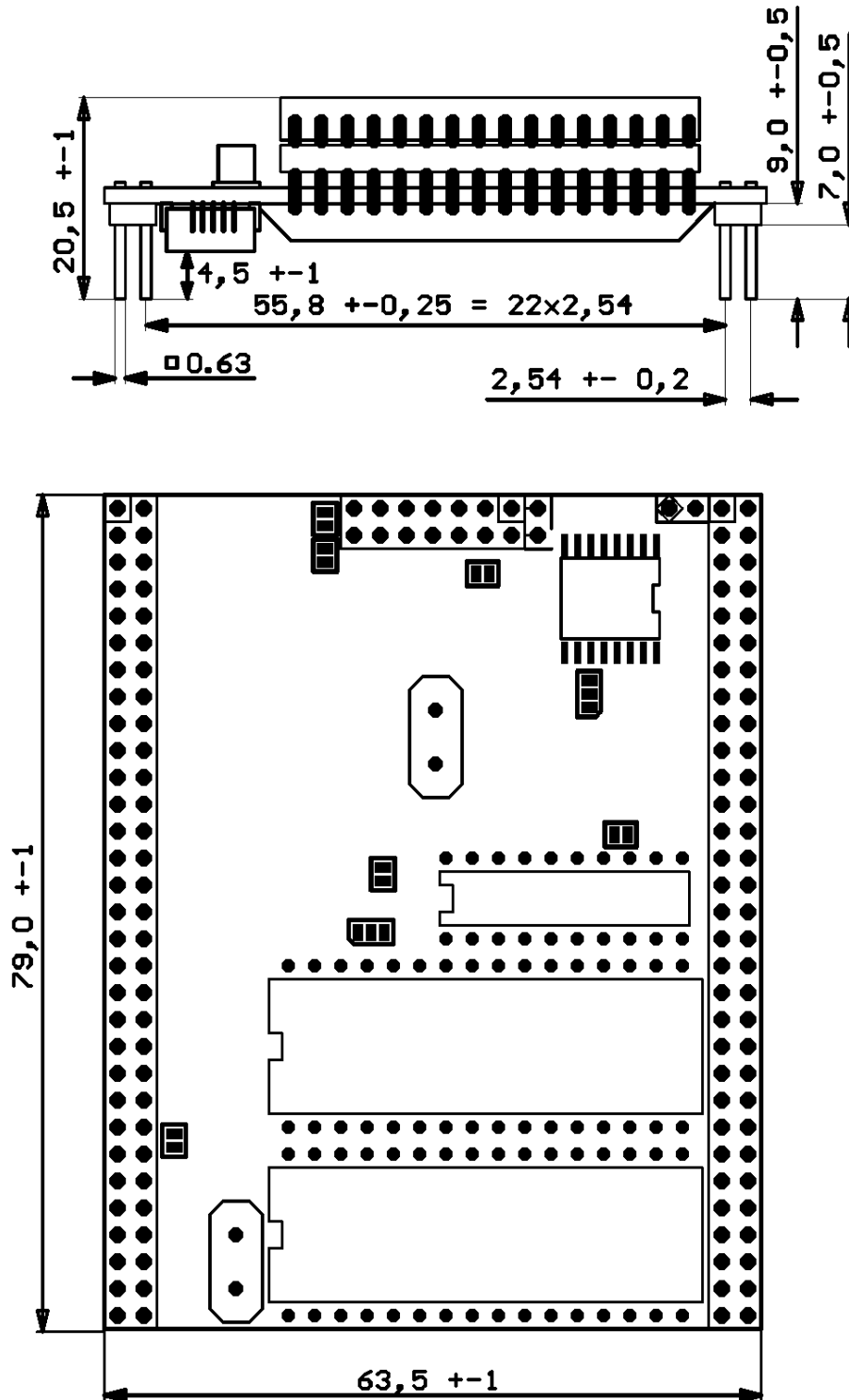
| Parameter | Symbol | min. | typ. | max. | Einheit |
|--|--------|---------|------|----------|---------|
| Betriebsspannung | VCC | 4.8 | 5 | 5.2 | V |
| Betriebsstrom | ICC | | 190 | | mA |
| Digitaler Eingangsspegel High (Ports) | VIH | 0.7*VCC | | VCC+0.5 | V |
| Digitaler Eingangsspegel Low (Ports) | VIL | -0.5 | | 0.3* VCC | V |
| Digitaler Ausgangsspegel High (Ports) | VOH | 0.9*VCC | | | V |
| Digitaler Ausgangsspegel Low (Ports) | VOL | | | 0.45 | V |
| Analoge Referenzspannung (AD-Wandler) | VAREF | 4 | | VCC+0.1 | V |
| Analoge Referenzmasse (AD-Wandler) | VAGND | VSS-0.1 | | VSS+0.2 | V |
| Analoge Eingangsspannungen | VAIN | VAGND | | VAREF | V |
| Ansprechschwelle für Reset-Signal (Power Fail) | VPF | 4.5 | 4.65 | 4.75 | V |
| Spannungsdifferenz (LowLine-Signal bevor Reset-Signal) | | 150 | | | mV |
| Externer Watchdog Timeout | tWD | 1 | 1.6 | 2.25 | s |
| U2-Time-Delay (Zeitverzögerung durch Schreibschutzverriegelung) | tPU2 | | | 10 | ns |
| Decoder-Time-Delay | tPD | | | 15 | ns |

Tabelle 8-2: Temperatur

| Parameter | Symbol | min. | typ. | max. | Einheit |
|---------------------------|--------|------|------|------|---------|
| Lagertemperaturbereich | Ts | -40 | | 85 | °C |
| Betriebstemperaturbereich | Tb | 0 | | 55 | °C |

8.2. Mechanische Abmessungen

Abbildung 8-1: Mechanische Abmessungen



9. Lageplan

Abbildung 9-1: Lageplan des Controllermoduls CM509 von oben gesehen.

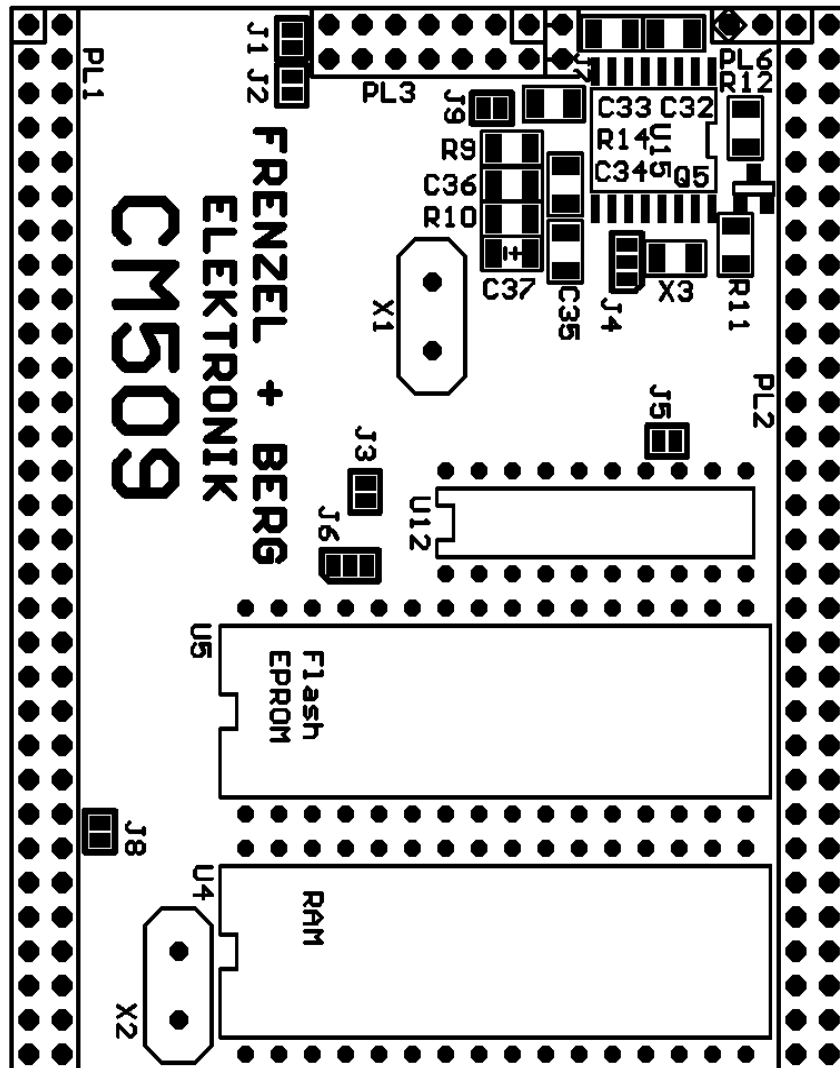
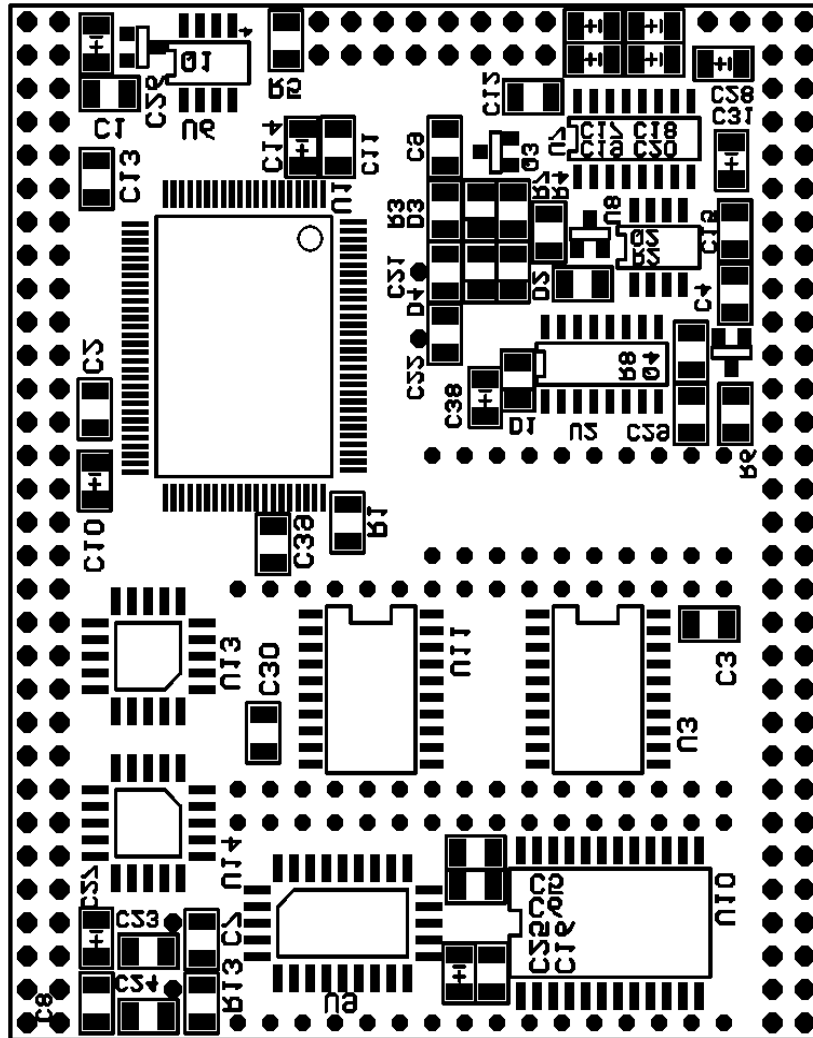
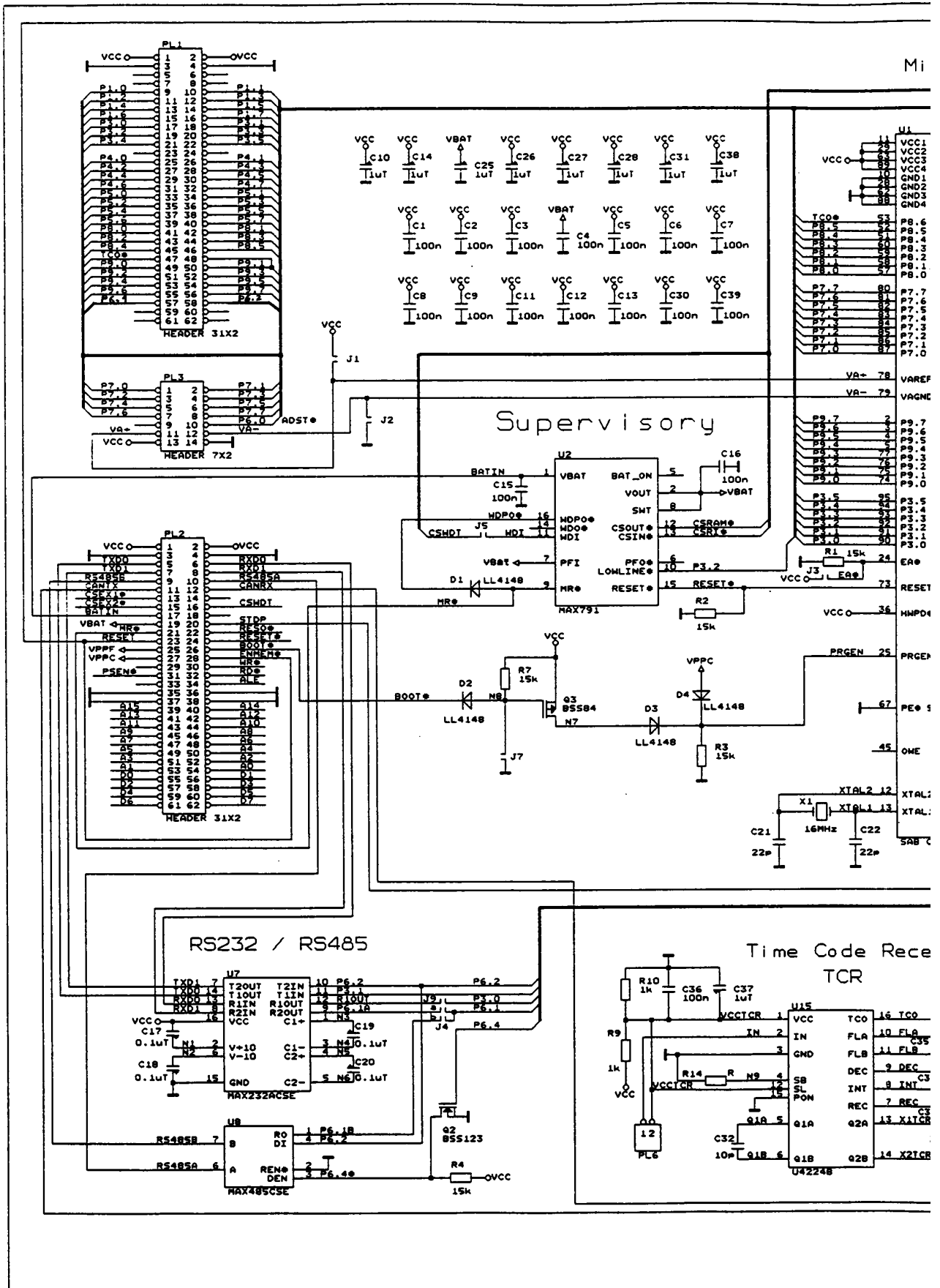
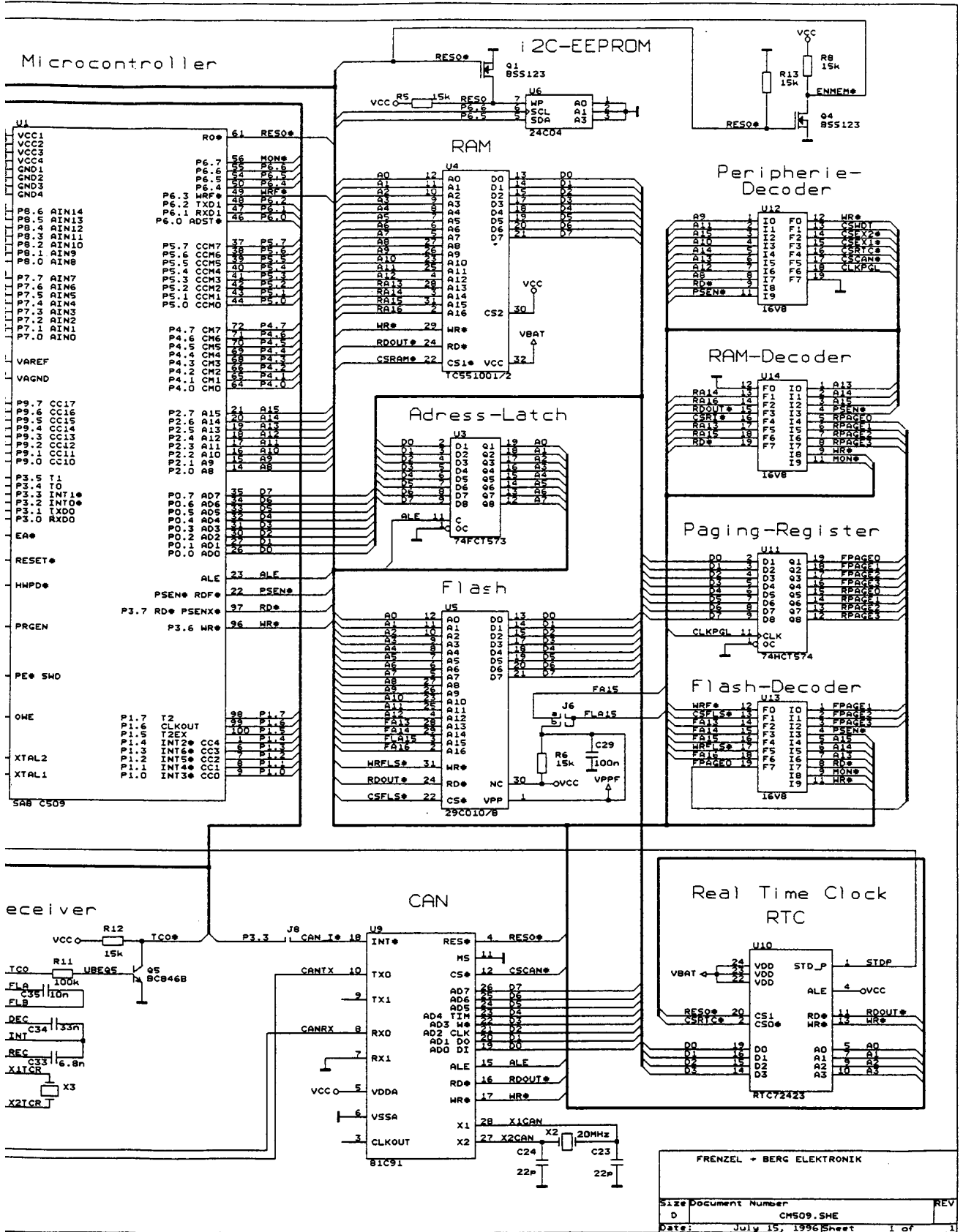


Abbildung 9-2: Lageplan des Controllermoduls CM509 von unten gesehen



10. Schaltplan





FRENZEL + BERG ELEKTRONIK
 Size Document Number: CH509.SHE
 Date: July 15, 1996 Sheet 1 of 1