

# **miniMODUL-537 / 509**

## **Hardware-Manual**

**Ausgabe September 1996**

Im Buch verwendete Bezeichnungen für Erzeugnisse, die zugleich ein eingetragenes Warenzeichen darstellen, wurden nicht besonders gekennzeichnet. Das Fehlen der ® Markierung ist demzufolge nicht gleichbedeutend mit der Tatsache, daß die Bezeichnung als freier Warename gilt. Ebenso wenig kann anhand der verwendeten Bezeichnung auf eventuell vorliegende Patente oder einen Gebrauchsmusterschutz geschlossen werden.

Ferner sei ausdrücklich darauf verwiesen, daß PHYTEC Meßtechnik GmbH weder eine Garantie noch die juristische Verantwortung oder irgendeine Haftung für Folgeschäden übernimmt, die auf falschen Gebrauch oder falschen Einsatz der Hard- bzw. Software zurückzuführen sind. Ebenso können ohne vorherige Ankündigung Layout oder Design der Hardware geändert werden. PHYTEC Meßtechnik GmbH geht damit keinerlei Verpflichtungen ein.

© Copyright 1996 PHYTEC Meßtechnik GmbH. Alle Rechte vorbehalten. Kein Teil dieses Buches darf in irgendeiner Form ohne schriftliche Genehmigung der Firma PHYTEC Meßtechnik GmbH unter Einsatz entsprechender Systeme reproduziert, verarbeitet, vervielfältigt oder verbreitet werden.

PHYTEC Meßtechnik GmbH  
Robert-Koch-Straße 39  
D-55129 Mainz

2. Auflage September 1996

---

Inhaltsverzeichnis .....	1
Bild- und Tabellenverzeichnis.....	2
1. Kurzübersicht über das miniMODUL-537/509 .....	3
2. Anschlußbelegung .....	5
3. Jumper .....	11
3.1 Serielle Schnittstelle .....	13
3.2 Speicherauswahl .....	15
3.3 Spezielle Features .....	16
4. Speichermodelle .....	19
4.1 Controlregister 1:.....	21
4.2 Controlregister 2:.....	26
4.3 Adreßregister:.....	27
4.4 Maskenregister: .....	28
5. Flash-Memory .....	31
6. Die Batteriepufferung .....	32
7. Technische Daten .....	33
8. Hinweise zum Umgang mit dem Modul .....	35
9. Flash-Tools .....	37
9.1 Starten der Flash-Tools .....	38
9.2 Flash-Programmierung.....	40
9.3 RAM-Download .....	40
Revisionswechsel des miniMODUL-537/509.....	43

## **Bild- und Tabellenverzeichnis**

Bild 1:	Lage der Pins.....	6
Bild 2a:	Zählweise der Jumper .....	11
Bild 2b:	Lage der Jumper (Platinenoberseite) .....	11
Bild 2c:	Lage der Jumper (Platinenunterseite) .....	12
Bild 3:	Default-Speichermodell nach Hardware-Reset .....	20
Bild 4:	Flash-Programmiermodell miniMODUL-537/509.....	22
Bild 5:	Aufteilung des I/O-Bereichs .....	23
Bild 6:	Beispiel-Speichermodell .....	30
Bild 7:	Mechanische Abmaße .....	33
Tabelle 1:	Pinout mit Erläuterung.....	7
Tabelle 2:	Revisionswechsel.....	43

## **1. Kurzübersicht über das miniMODUL-537/509**

Das miniMODUL-537/509 ist ein universelles Microcontrollerboard im Scheckkartenformat. Es basiert auf dem Microcontroller 80C537 oder alternativ auf dem C509 von SIEMENS.

Der C509 Controller kann mit maximal 16Mhz getaktet werden und erreicht damit die Rechenleistung eines mit 32Mhz getakteten 8032 (375ns Zykluszeit).

Die controllerspezifischen Eigenschaften entnehmen Sie bitte dem Handbuch zum jeweiligen Microcontroller, im Hardware-Manual zum miniMODUL-537/509 wird auf keinerlei Besonderheiten des jeweiligen Microcontrollers näher eingegangen, da diese für die grundlegende Funktion des miniMODUL-537/509 ohne Belang sind.

## Das miniMODUL-537/509 bietet folgende Features:

- Rechner im Scheckkartenformat 55 x 85 mm durch Einsatz moderner SMD-Technik
- Verbesserte Störsicherheit durch Multilayer-Technik
- Aufsetzbar auf die Anwendungsschaltung wie ein großer Chip
- SIEMENS Controller 80C537 (auch 80C517...517A) im PLCC84 Gehäuse oder C509 im QFP100 Gehäuse, befehlskompatibel zur Familie der 8051-Prozessoren von INTEL
- Einzige Versorgungsspannung 5V, typ. <110mA
- Bis zu 512 kByte Flash On-Board (PLCC)
- On-Board Flash-Programmierung
- Keine separate Programmierspannung durch Verwendung von 5V-Flash-Bausteinen
- Bis zu 160 kByte RAM On-Board (SMD)
- Wahlweise auch mit 32 kByte EEPROM (SMD)
- Alle Ports sowie Daten- und Adreßleitungen am Platinenrand über Stiftleisten verfügbar
- Flexible, per Software konfigurierbare Adreßdecodierung durch komplexen Logikbaustein
- Banklatches für Flash und RAM im Adreßdekoeder integriert
- Wahlweise Schnittstellentreiber für zwei RS-232- oder eine RS-232- und eine RS-485-Schnittstelle
- Wahlweise zwei Echtzeituhren RTC8583 oder RTC72423
- Echtzeituhren und RAMs mit externer Batterie pufferbar
- 3 freie Chip-Select-Signale für einfachen Anschluß externer Peripherie

## 2. Anschlußbelegung

Es sei ausdrücklich darauf hingewiesen, daß bei allen Modulan-  
schlüssen unbedingt die Maximalspannungen und -ströme nicht  
überschritten werden dürfen. Die Grenzwerte hierfür können Sie  
dem jeweiligen Controller-Handbuch entnehmen. Da eventuell  
auftretende Störungen stark vom Einsatzgebiet bzw. Anwendungsfall  
abhängen, obliegt es der Verantwortung des Anwenders, in  
entsprechend kritischer Umgebung geeignete Schutzmaßnahmen zu  
treffen.

Wie in *Bild 1* dargestellt, werden alle relevanten Signale an drei Sei-  
ten des Moduls an den Platinenrand geführt, dort ist eine Bestückung  
mit Stiftleisten im Rastermaß 2,54 mm möglich. Die folgende  
*Tabelle 1* (S. 7ff.) gibt Ihnen eine Übersicht über die Belegung aller  
Anschlüsse.

### **Achtung:**

**Das miniMODUL-537 wurde im Rahmen einer Umstellung  
auf Flash-Technologie überarbeitet. Hierbei wurde auf  
bestmögliche Kompatibilität geachtet, allerdings sind  
einige Unterschiede zum Vorgängermodul unausweichlich.  
Bitte beachten Sie in diesem Zusammenhang die Hinweise  
im Anhang A.**

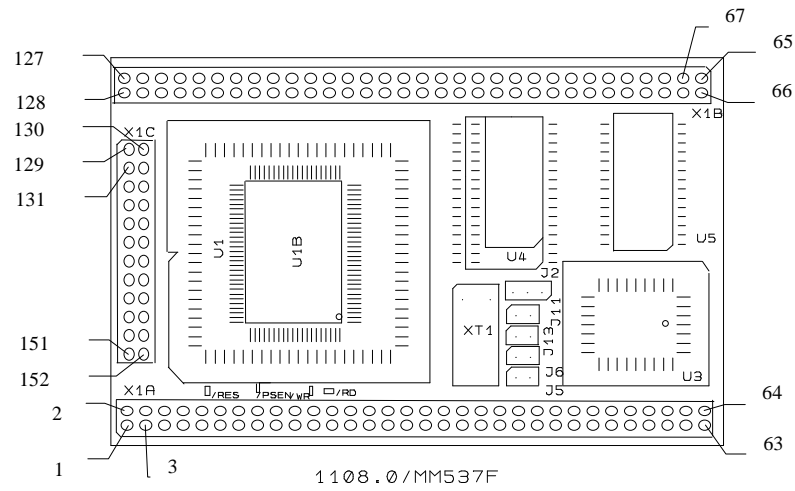


Bild 1 : Lage der Pins

PIN Nr.	Bezeichnung	Beschreibung
1,2	NC	nicht verwendet
3...10	P4.0...P4.7	Port 4
11	MDIS	Memory-Disable-Eingang für U4 und U5
12	/PSEP	trennbares /Program-Store-Enable-Signal des Controllers <sup>1</sup>
13	/WRP	trennbares /WR-Signal des Controllers <sup>1</sup>
14	/RDP	trennbares /RD-Signal des Controllers <sup>1</sup>
15...20	P3.0...P3.7	Port 3
21	/WR	trennbares /WR-Signal des Moduls <sup>1</sup>
22	/RD	trennbares /RD-Signal des Moduls <sup>1</sup>
23...30	P1.7...P1.0	Port 1
31	/PSEN	trennbares /Program-Store-Enable-Signal des Moduls <sup>1</sup>
32	VPD	Spannungsausgang für externe Pufferung
33	RES	Reset-Ausgang des Moduls
34	/RES	trennbarer /Reset-Ein-/Ausgang des Moduls <sup>1</sup> (Open-Kollektor)
35	/CS1	vordekodiertes Chip-Select-Signal #1
36	/CS2	vordekodiertes Chip-Select-Signal #2
37	/CS3	vordekodiertes Chip-Select-Signal #3
38	/PFO	/Power-Fail-Ausgang
39	PFI	Power-Fail-Eingang
40	/HPD	opt. /HWPD-Eingang bei 80C517A und C509
41,43,51, 52	TI1...TI4	Transmitter Eingänge 1-4 des RS-232-Treibers (TI1 und TI2 über J5 bzw. J8 mit seriellen Schnittstellen des Controllers verbunden)
42	VBAT	Eingang für Anschluß externe Pufferbatterie
44,54,50	RO1...RO3	Receiver Ausgänge 1-3 des RS-232-Treibers (RO1 und RO2 über J6 bzw. J7 mit seriellen Schnittstellen des Controllers verbunden)
45,55,49	RI1...RI3	Receiver Eingänge 1-3 des RS-232-Treibers
47,46,48, 56	TO1...TO4	Transmitter Ausgänge 1-4 des RS-232-Treibers
53	RSDIS	RS-232-Treiber-Disable Eingang
57	ALE	Adresslatch-Enable-Ausgang
58	/RESP	trennbares Reset-Signal des Controllers <sup>1</sup>

<sup>1</sup>: Für die Verwendung von Emulatoren können die Controllersignale /XXP von den im restlichen Modul verwendeten Signalen /XX getrennt und demzufolge von außen eingespeist werden. Hiervon betroffen sind die Signale /PSEN, /RD, /WR und /RES.

59	/CSRTC	Chip-Select-Signal der RTC72423 (über Jumper J13 mit /CS1 verbunden)
60	/IRTC	Interrupt-Ausgang der beiden RTC's
61	/RESI	/Reset-Eingang des Moduls
62	WDP	Watchdog Eingang des Moduls
63,64	NC	nicht verwendet
65,66	VCC	Versorgungsspannung +5V=
67...74	D7...D0	Datenbus (Port 0 des Controllers)
75...82	A7...A0	Adreßbus (Low-Byte)
83...90	A9,A8,A11, A10,A13,A12, A15,A14	Adreßbus (High-Byte)
91	PRGEN	Prog-Enable-Eingang (nur C509 Controller)
92	NC	nicht verwendet
93	DE	Daten-Enable-Eingang des RS-485-Treibers (mit Pin 13 von U11 beschaltet)
94	D	Daten-Eingang des RS-485-Transmitters (über Jumper J8 mit serieller Schnittstelle des Controllers verbunden)
95	B	differentielle B-Leitung des RS-485-Treibers
96	R	Daten-Ausgang des RS-485-Receiver (über Jumper J7 mit serieller Schnittstelle des Controllers verbunden)
97	A	differentielle A-Leitung des RS-485-Treibers
98	/R	invertierter Daten-Ausgang des RS-485- Receiver (über Jumper J11 mit P3.2 des Controllers verbunden)
99	/DE	invertierender Data-Enable-Eingang des RS- 485-Transmitters (über J9 mit P5.1 des Controllers verbunden)
100	/RE	/Receive-Enable-Eingang des RS-485- Receiver (über Jumper J10 mit GND verbunden)
101..108	P9.7...P9.0	Port 9 (nur C509 Controller)
109..116	P5.7...P5.0	Port 5
117	/RO	/Reset-Ausgang des Controllers
118	/PE	Watchdog-Timer / Power-Saving-Modus des Controllers (über Jumper J3 mit GND verbunden)
119..126	P6.0...P6.7	Port 6

<sup>1</sup>: Für die Verwendung von Emulatoren können die Controllersignale /XXP von den im restlichen Modul verwendeten Signalen /XX getrennt und demzufolge von außen eingespeist werden. Hiervon betroffen sind die Signale /PSEN, /RD, /WR und /RES.

---

127,128	GND	Schaltungsmasse 0V
129,133, 137,141, 143,145, 147,149	AGND	Bezugsmasse Analogeingänge 0V
131,135, 139	AN12...AN14	Analogeingänge AN12...AN14 (nur C509 Controller)
136,134, 132,130, 152,150, 148,146, 144,142, 140,138	AN11...AN0	Analogeingänge AN11...AN0
151	AREF	Referenzspannung Analogeingänge +5V=

*Tabelle 1: Pinout mit Erläuterung*

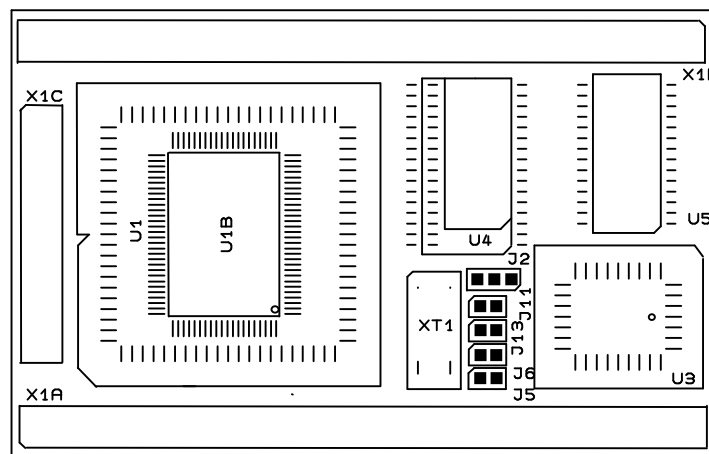


### 3. Jumper

Das miniMODUL-537/509 besitzt zur Konfiguration 14 Lötjumper, die teilweise bereits bei der Auslieferung vorverbunden sind. Das *Bild 2a* verdeutlicht die verwendete Zählweise bei den Jumpern, die *Bilder 2b und 2c* die Lage der Jumper auf der jeweiligen Platinenseite.



*Bild 2a: Zählweise der Jumper*



*Bild 2b: Lage der Jumper (Ansicht Platinenoberseite)*

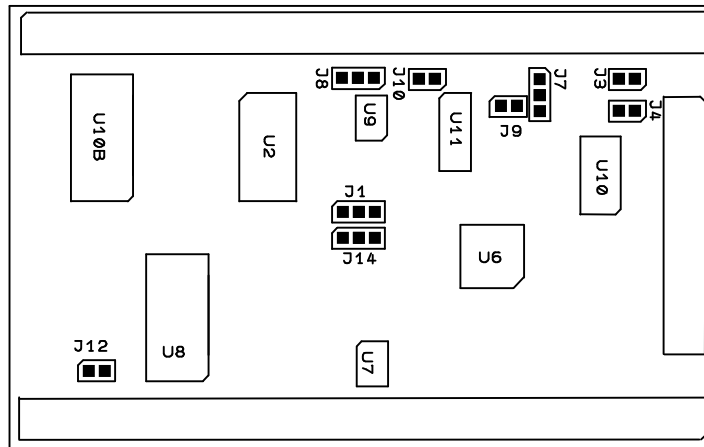


Bild 2c: Lage der Jumper (Ansicht Platinenunterseite)

Die Jumper können in drei Gruppen unterteilt werden:

1. serielle Schnittstellen J5, J6, J7, J8, J9, J10, J11 und J12
2. Speicherauswahl (U5) J2
3. spezielle Features J1, J3, J4, J13 und J14

### 3.1 Serielle Schnittstellen

Über die Jumper J5 und J6 lassen sich die beiden Pins der ersten seriellen Schnittstelle (Serial0) des Controllers mit dem RS-232-Transceiver verbinden. Zudem sind die Controllerpins immer mit TTL-Pegel direkt an den Modulpins 15 und 16 (i.e. P3.0 und P3.1) verfügbar. Im Auslieferungszustand ist die erste serielle Schnittstelle des Controllers mit dem RS-232-Transceiver verbunden, die entsprechenden RS-232-Signale liegen an den Modulpins 45 und 47 an.

Es ergeben sich folgende Konfigurationen für die erste serielle Schnittstelle:

Signalqualität erste serielle Schnittstelle	J5	J6
RS-232 (Modulpins 45 und 47)	geschlossen	geschlossen
TTL (Modulpins 15 und 16)	offen	offen

Über die Jumper J7 und J8 lassen sich die beiden Pins der zweiten seriellen Schnittstelle (Serial1) des Controllers wahlweise mit dem RS-232- oder dem RS-485-Transceiver verbinden. Auch diese Controllerpins sind zudem immer mit TTL-Pegel direkt an den Modulpins 120 und 121 (i.e. P6.1 und P6.2) verfügbar.

Es ergeben sich folgende Konfigurationen für die zweite serielle Schnittstelle:

Signalqualität zweite serielle Schnittstelle	J7	J8
RS-232 (Modulpins 46 und 55)	2+3	2+3
TTL (Modulpins 120 und 121)	offen	offen
RS-485 (Modulpins 95 und 97)	1+2	1+2

Durch Schließen des Jumpers J9 wird der invertierende Data-Enable-Eingang des RS-485-Transceivers mit dem Pin P5.1 des Controllers verbunden. Auf diese Weise kann eine softwaregesteuerte Freigabe des RS-485-Transmitters erfolgen, was beispielsweise für den Betrieb von  $\mu$ NET erforderlich ist.

RS-485-Transmitter	J9	P5.1	RES
Sendefreigabe	geschlossen	Low	Low
Sendesperre	geschlossen	High	don't care
	geschlossen	don't care	High
	offen	don't care	don't care

Durch Schließen des Jumpers J10 wird die Empfangsbereitschaft des RS-485-Transceivers hergestellt.

RS-485-Receiver	J10
Empfangsfreigabe	geschlossen
Empfangssperre	offen

Durch Schließen des Jumpers J11 wird der invertierte Daten-Ausgang des RS-485-Empfängers mit dem Pin P3.2 des Controllers verbunden. Da dieser Pin bitadressierbar ist, kann auf diese Weise mittels bitadressierenden Befehlen auf den Zustand des Daten-

Ausgangs reagiert werden. Dies ist beispielsweise ebenfalls für den Betrieb von  $\mu$ NET erforderlich.

P3.2 des Controllers	J11
verbunden mit invertiertem Daten-Ausgang	geschlossen
nicht verbunden	offen

Für einen eventuellen Einsatz des PHYTEC eigenen RS-485-Netzwerks namens  $\mu$ NET auf dem miniMODUL-537/509 sind unbedingt die Jumper J9, J10 und J11 zu schließen sowie die Konfiguration von  $\mu$ NET entsprechend anzupassen.

Durch Öffnen des Jumpers J12 kann der RS-232-Transceiver deaktiviert werden. In diesem Falle ist eine Steuerung der Aktivität des Transceivers über den Eingang RSDIS des Moduls (Modulpin 53) möglich. Im Auslieferungszustand ist der Jumper J12 geschlossen und somit der RS-232-Transceiver aktiviert.

RS-232-Transceiver	J12	RSDIS
aktiviert	geschlossen	unbeschaltet
	offen	Low
deaktiviert	offen	unbeschaltet
	offen	High

### 3.2 Speicherauswahl

Mit dem Jumper J2 kann die Versorgungsquelle für den Speicherbaustein U5 in Abhängigkeit des Bausteintyps ausgewählt werden. Bei bestücktem EEPROM auf U5 ist die Versorgung mit VCC zwingend erforderlich (J2 = 1+2), um ein vorzeitige Entladung einer eventuell angeschlossenen Pufferbatterie zu vermeiden. Ein RAM hingegen sollte über VPD (J2 = 2+3) versorgt werden, damit im Falle einer Batteriepufferung der Dateninhalt auch bei abgeschaltetem VCC gewährleistet ist.

Bausteintyp auf U5	J2
EEPROM	1+2
RAM	2+3

### 3.3 Spezielle Features

Über die Jumper J1, J3, J4, J13 und J14 werden spezielle Features zur Verfügung gestellt.

- **Ausführung aus internem oder externem Programmspeicher**  
Spezielle Features:

Der Jumper J1 ist bei der Auslieferung zwischen den Pads 1+2 verbunden. Dadurch wird nach einem Hardware-Reset das im externen Programmspeicher abgelegte Programm abgearbeitet. Um bei entsprechenden Controllern eine Abarbeitung eines internen Programmspeichers zu ermöglichen, muß am Jumper J1 eine Verbindung zwischen den Pads 2+3 vorgenommen werden.

Code-Zugriff	J1
externer Programmspeicher	1+2
interner Programmspeicher	2+3

**- Power-Saving-Modes / Watchdog-Timer**

Durch Öffnen des Jumpers J3 kann die Aktivierung der Power-Saving-Modes des Controllers gesperrt werden, gleichzeitig wird der Watchdog-Timer automatisch nach Reset gestartet. Im Auslieferungszustand ist der Watchdog-Timer zunächst inaktiv. Sie können diesen aber per Software aktivieren oder die Power-Saving-Modes des Controllers benutzen.

Power-Saving-Modes	Watchdog-Timer	J3
freigegeben	gesperrt	geschlossen
gesperrt	freigegeben	offen

**- Oszillator-Watchdog**

Der Oszillator-Watchdog ist im Auslieferungszustand aktiviert und ermöglicht so einen schnellen Power-On-Reset und einen sicheren Betrieb des Controllers

Oszillator-Watchdog	J4
inaktiv	geschlossen
aktiv	offen

**- Chip-Enable der Echtzeituhr RTC72423**

Bei geschlossenem Jumper J13 wird die RTC72423 mit dem vordekodierten Chip-Select-Signal /CS1 des Adressdecoders verbunden. Bei geöffnetem Jumper J13 kann über das Signal /CSRTC des Moduls (Modulpin 59) ein beliebiges Chip-Select-Signal angeschlossen werden.

Aus Gründen der Kompatibilität zu älteren Modulen ist dieser Jumper im Auslieferungszustand offen.

Chip_Enable RTC-72423	J13
/CS1 vom Adressdecoder	geschlossen
externe /CSRTCModulpin 59	offen

**- Internes Programmiermodell bei C509**

Der Jumper J14 ist nur relevant bei Verwendung eines C509-Controllers. Bei Verbindung zwischen den Pads 2+3 wird ein optionales Programmiermodell des C509 freigegeben, welches per Software aktiviert werden kann. Hierbei werden die XDATA- und CODE-Bereiche ausgetauscht, so daß man aus einer im RAM ablaufenden Applikation ein Flash-Memory programmieren kann. Hierzu existiert ein spezielles /WRF-Signal als Schreibsignal für das Flash. Da PHYTEC diesen Modus allerdings nicht zur Programmierung der On-Board Flash-Memories verwendet, ist das Board auch nicht für dessen Einsatz vorgesehen, so daß der Jumper 14 auf 1+2 geschlossen werden sollte.

Programmier-Modus C509	J14
gesperrt	1+2
freigegeben	2+3

## 4. Speichermodelle

Das miniMODUL-537/509 verfügt über einen konfigurierbaren Adreßdekodeur, der Anpassungen des Speichermodells per Software zuläßt. Nach einem Hardware-Reset ist ein Default-Speichermodell vorgegeben, welches bereits für eine Vielzahl von Applikationen geeignet ist, jedoch bei Bedarf zu Beginn der jeweiligen Applikation verändert bzw. angepaßt werden kann.

Die Einstellung bzw. Konfiguration des Speichermodells vollzieht sich anhand von zwei Control-, einem Adreß- sowie einem Maskenregister innerhalb des Dekoders. Alle genannten Register sind als Write-Only-Register mit Zugriff im XDATA-Bereich des Controllers ausgeführt. Es existieren zwei verschiedene Adreßbereiche für den Zugriff auf die Register, die durch das Bit IO-SW im Controlregister 1 ausgewählt werden können (siehe Beschreibung des Bits IO-SW). Aufgrund mangelnder Lese-Zugriffe sollte unbedingt eine Kopie aller Registerinhalte in der Applikation gepflegt werden. Reservierte Bits dürfen durch das Schreiben der Register nicht verändert werden, der Inhalt sollte unbedingt auf 0 verbleiben. Alle Register werden durch einen Hardware-Reset gelöscht, wodurch die Einstellung des bereits erwähnten Default-Speichermodells gewährleistet wird.

**Falls Sie die Flash-Tools - eine Firmware zur komfortablen On-Board Flash-Programmierung - verwenden, so ist zu beachten, daß beim Start Ihrer Anwendersoftware bereits die Adresse FA16 (s. Controlregister 1) gesetzt wurde. Dieser Sachverhalt ist bei der Anlage der Softwarekopie der Registerinhalte unbedingt zu berücksichtigen. Im Kapitel 9. *Flash-Tools* wird dieser Sachverhalt näher erläutert.**

Folgendes Bild zeigt das Default-Speichermodell:

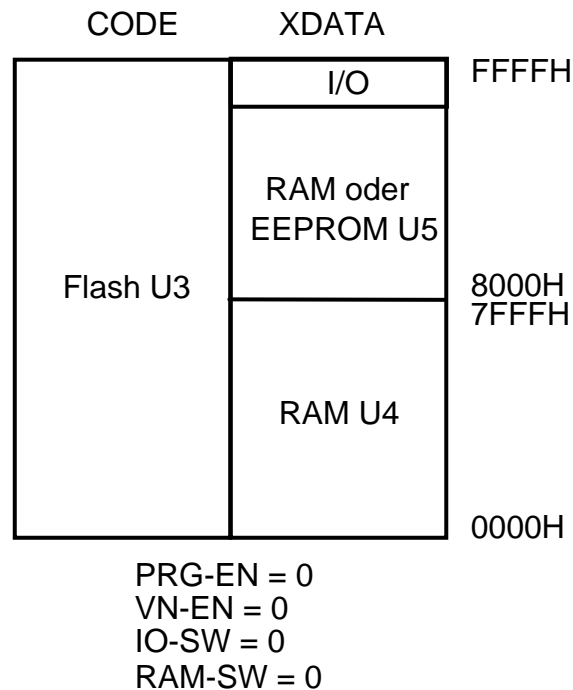


Bild 3: Default-Speichermodell nach Hardware-Reset

Hierbei gilt zu beachten, daß jedem der beiden Speicherbausteine U4 und U5 jeweils ein getrennter, 32 KB großer Speicherbereich im XDATA-Adreßraum des Controllers zukommt. Im Falle einer Bestückung von U4 mit einem 128 KB RAM-Baustein kann dieser mittels Bank-Latching in Blöcken à 32 KB angesprochen bzw. umgeschaltet werden. Falls einer der Bausteine U4 und U5 nicht bestückt ist, besteht im entsprechenden Speicherbereich kein Zugriff auf Speicher. Der jeweils aktuelle I/O-Bereich wird im XDATA-Adreßbereich eingeblendet, in ihm besteht kein Zugriff auf einen eventuell vorhandenen Speicherbaustein.

In den folgenden Abschnitten sind die Register des Adreßdekoders zur Anpassung des Speichermodells erläutert:

#### 4.1 Controlregister 1:

Controlregister 1 (Adresse 7C00H / FC00H)							
Bit 7							Bit 0
PRG-EN	IO-SW	RAM-SW	VN-EN	FA18	FA17	FA16 <sup>1</sup>	FA15

Bit im Programmiermodell nicht relevant (s. PRG-EN)

Bit nur im Programmiermodell relevant (s. PRG-EN)

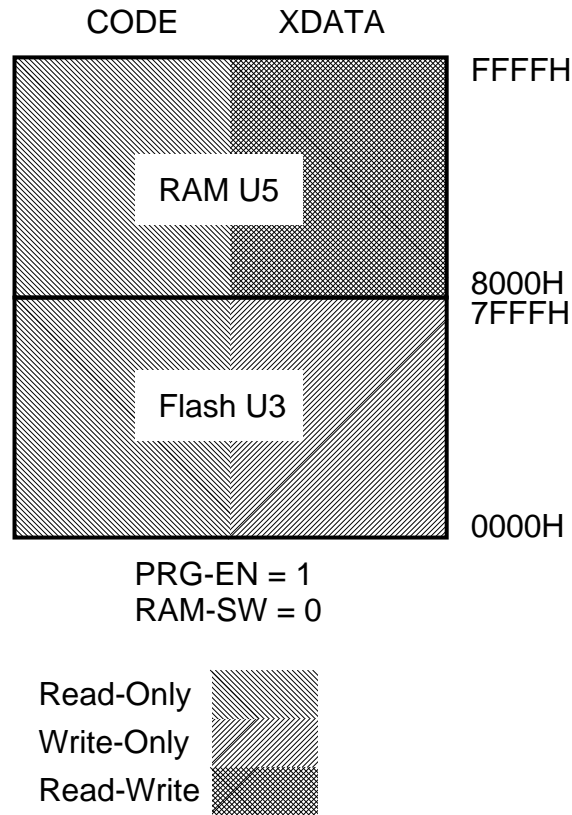
**PRG-EN:** Dient dem Aktivieren des gesonderten Flash-Programmiermodells (PRG-EN = 1). Dieses Modell wird innerhalb der Flash-Tools<sup>2</sup> zur Flash-Programmierung verwendet und ist aufgrund der vorhandenen Restriktionen nicht bzw. nur bedingt innerhalb Ihrer Applikation zu verwenden.

In diesem Modell besteht Zugriff auf 32 KB Flash im Adreßbereich von 0000H-7FFFH sowie auf 32 KB RAM im Bereich von 8000H-FFFFH. Das Flash ist im XDATA-Bereich lediglich zu schreiben, es kann ausschließlich im CODE-Bereich gelesen werden. Das RAM kann im XDATA-Bereich sowohl gelesen als auch geschrieben werden, das Lesen im CODE-Bereich ist ebenfalls möglich. Nur im Programmiermodell wird die Adreßleitung A15 des Flash ebenfalls dem Controlregister 1 (Bit 0, FA15) entnommen, im Runtime-Modell (PRG-EN = 0) wird die Adreßleitung A15 des Controllers unmittelbar an das

<sup>1</sup>: Bei Einsatz der Flash-Tools - einer Firmware zur komfortablen On-Board Flash-Programmierung - ist dieses Bit beim Start Ihrer Anwendung bereits gesetzt. Dies muß bei der Anlage der Softwarekopie Berücksichtigung finden.

<sup>2</sup>: Eine Firmware zur komfortablen On-Board Flash-Programmierung; beim Erwerb des Moduls incl. Flash-Memory ist diese Software bereits in das Flash einprogrammiert.

Flash durchgeschleift. Die Bits IO-SW und RAM-SW bleiben auch im Programmiermodell relevant, das Bit VN-EN hingegen nicht. *Bild 4* verdeutlicht das Programmiermodell des miniMODUL-537/509 (I/O-Bereich nicht dargestellt):



*Bild 4: Flash-Programmiermodell des miniMODUL-537/509*

**IO-SW:** Mittels dieses Bits kann der I/O-Bereich des Moduls wahlweise in die oberen oder die unteren 32 KB des Adreßraums gelegt werden. Nach einem Hardware-Reset (IO-SW = 0) liegt der I/O-Bereich von FC00H bis FFFFH, nach Setzen des IO-SW-Bits liegt er im Bereich von 7C00H-7FFFH.

Dieser I/O-Bereich besteht generell aus 4 Blöcken à 256 Bytes. In drei dieser Blöcke stellt der Adreß-

dekoder jeweils ein vordekodiertes Chip-Select-Signal zur Verfügung, das den Hardware-Aufwand zum Anschluß eigener Peripherie an das Modul reduziert. Diese Chip-Select-Signale werden bei XDATA-Zugriffen (Read-Write Zugriffe) im entsprechenden Adreßbereich aktiviert. Der vierte Block ist reserviert für Zugriffe auf die dekodern internen Register (Write-Only Zugriffe). Dieser Block steht Ihnen als Anwender daher für den Anschluß externer Peripherie nicht zur Verfügung.

Die Aufteilung des I/O-Bereichs ist folgendem Bild zu entnehmen:

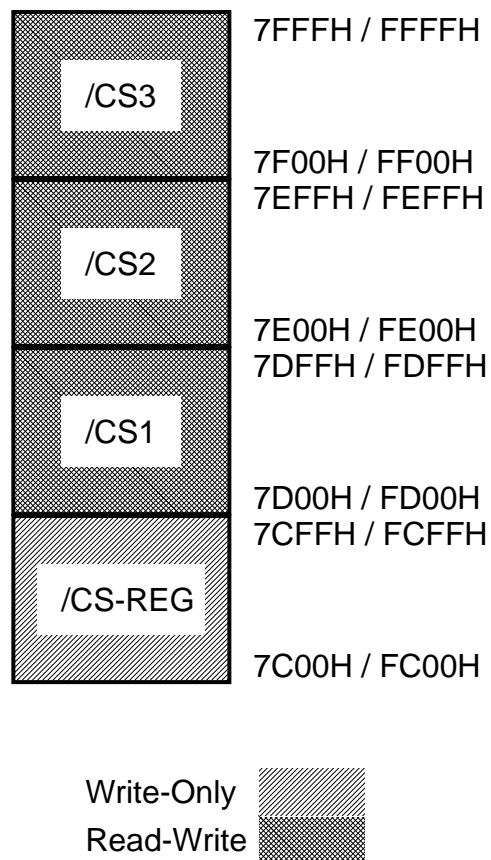


Bild 5: Aufteilung des I/O-Bereichs

Hierbei sind /CS1 bis /CS3 die frei verfügbaren Chip-Select-Signale, das Signal /CS-REG ist lediglich ein dekoderinternes Signal, welches für den Zugriff auf die internen Register benötigt wird. Dieses Signal steht Ihnen als Kunde nicht zur Verfügung, der Anschluß jeglicher Peripherie im Gültigkeitsbereich von /CS-REG sollte unter allen Umständen unterbleiben, um eine korrekte Funktion der Flash-Tools<sup>3</sup> zur On-Board-Programmierung des Flash zu gewährleisten. Die internen Register belegen momentan lediglich die Adressen 7C00H-7C03H bzw. FC00H-FC03H, der Rest des /CS-REG-Blockes bleibt ungenutzt und ist für künftige Erweiterungen reserviert.

**RAM-SW:** Mittels dieses Bits können die 32 KB Speicherbereiche der Speicherbausteine U4 und U5 ausgetauscht werden. Nach einem Hardware-Reset (RAM-SW = 0) ist das RAM U4 im Bereich von 0000H bis 7FFFH und das RAM / EEPROM U5 im Bereich von 8000H bis FFFFH adressierbar, nach Setzen des Bits RAM-SW belegt das RAM U4 den Bereich von 8000H-FFFFH und das RAM / EEPROM U5 den Bereich von 0000H-7FFFH. Im jeweils eingestellten I/O-Bereich existiert kein Zugriff auf die Speicherbausteine.

**VN-EN:** Mit diesem Bit werden im Adreßraum des Controllers optionale Von-Neumann<sup>4</sup>-Speicherbereiche freigeschaltet. Nach einem Reset ist per Default eine Harvard<sup>5</sup>-Architektur vorhanden. Von-Neumann-Speicherbereiche sind insbesondere dann sinnvoll, wenn zur Laufzeit Programmcode nachgeladen und anschließend ausgeführt werden soll (z.B. Monitor-

---

<sup>3</sup>: Software-Werkzeug zur On-Board Flash-Programmierung, ist bereits bei Auslieferung in das Flash vorprogrammiert.

<sup>4</sup>: Speicherbereich, in dem die Trennung zwischen CODE- und XDATA-Zugriffen aufgehoben ist; beide Zugriffsarten zielen auf den physikalisch gleichen Speicherbaustein, in der Regel ein RAM.

<sup>5</sup>: Speicherbereich, in dem CODE- und XDATA-Zugriffe auf physikalisch verschiedene Speicherbausteine abzielen; in der Regel wird für CODE-Zugriffe ein ROM oder Flash, für XDATA-Zugriffe ein RAM eingesetzt.

Anwendung). Die Lage dieser optionalen Von-Neumann-Speicherbereiche wird über das Adreß- sowie das Maskenregister definiert (s.u.).

Nach einem Hardware-Reset ( $VN-EN = 0$ ) sind die Einstellungen im Adreß- und Maskenregister nicht freigeschaltet, d.h., es werden keine Von-Neumann-Bereiche zur Verfügung gestellt. Nach dem Setzen des Bits ( $VN-EN = 1$ ) werden die Einstellungen im Adreß- sowie im Maskenregister freigeschaltet und in die Zugriffssteuerung einbezogen. Dieses Bit ist nur im Runtime-Modell ( $PRG-EN = 0$ ) relevant, im Programmier-Modell ( $PRG=1$ ) ist es ohne Bedeutung und wird ignoriert.

FA[18..15]: Das Modul verfügt über die Option, einen 512 KB großen Flash-Baustein aufzunehmen. Da der Adreßraum des Controllers auf 64 KB beschränkt ist, kann der Rest des Flashs lediglich per Bankumschaltung erreicht werden.

Im Runtime-Modell ( $PRG-EN = 0$ ) kann das Flash in Bänken à 64 KB umgeschaltet werden, indem die hohen Adreßleitungen A[18..16] für das Flash per Software vorgegeben werden. Zu diesem Zwecke stellt der Adreßdekoder mit den Registerbits FA[18..16] bereits Latches zur Verfügung, in welche die gewünschten hohen Adressen eingeschrieben werden müssen.

Besondere Beachtung gilt dem Bit FA15, welches lediglich im Programmier-Modell ( $PRG-EN = 1$ ) relevant wird. Da in diesem Modell auf lediglich 32 KB Flash zugegriffen werden kann, dient es als Adreßleitung A15 am Flash-Baustein. Im Runtime-Modell ( $PRG-EN = 0$ ) mit 64 KB Flash-Bereich wird hingegen die Adreßleitung A15 des Controllers direkt an das Flash durchgeschleift.

Die Funktion der Bits FA[18..16] ist bestückungsabhängig und wirkt sich in der geschilderten Art und Weise nur bei Flash-Bausteinen mit einer Größe von 512 KB aus.

## 4.2 Controlregister 2:

Controlregister 2 (Adresse 7C01H / FC01H)							
Bit 7							Bit 0
N/A <sup>6</sup>	N/A	N/A	N/A	N/A	N/A	RA16	RA15

RA[16..15]: Das Modul verfügt über die Option, einen 128 KB großen RAM-Baustein auf Position U4 aufzunehmen. Da der Adreßraum des Bausteins U4 im XDATA-Adreßbereich des Controllers auf 32 KB beschränkt ist, kann der Rest des RAMs lediglich per Bankumschaltung erreicht werden.

Es können 4 Bänke à 32 KB umgeschaltet werden, indem die hohen Adreßleitungen A[16..15] für das RAM per Software vorgegeben wird. Zu diesem Zweck stellt der Adreßdekoder mit den Registerbits RA[16..15] bereits Latches zur Verfügung, in welche die gewünschte hohen Adressen eingeschrieben werden müssen.

Die Funktion dieser Bits ist bestückungsabhängig und wirkt sich in der geschilderten Art und Weise nur bei RAM-Bausteinen auf U4 mit einer Größe von 128 KB aus.

<sup>6</sup>: N/A: Not Accessible, nicht verfügbar

### 4.3 Adreßregister:

Das Adreßregister (Adresse 7C02H / FC02H) dient zusammen mit dem Maskenregister (s.u.) der Definition von Von-Neumann<sup>7</sup>- und Harvard<sup>8</sup>-Speicherbereichen im Adreßraum des Controllers. Durch Setzen des Bits VN-EN im Controlregister 1 werden die Einstellungen freigeschaltet und in die Adreßdekodierung einbezogen (s. Controlregister 1).

Mit beiden Registern wird die Lage von einem bzw. mehreren Harvard-Bereichen konfiguriert, die verbleibenden Bereiche des Adreßraums werden zu Von-Neumann-Bereichen, in denen die RAMs sowohl bei XDATA- als auch bei CODE-Zugriffen angesprochen wird.

Der verwendete Mechanismus zur Unterscheidung der Bereiche beruht auf einem Vergleich der aktuellen Adressen mit einem vordefinierten Adreßmuster in maskierbaren Bitstellen. Wird eine Übereinstimmung in den relevanten Bitstellen der Adresse erkannt, erfolgen die Zugriffe gemäß einer Harvard-Architektur, andernfalls gemäß einer Von-Neumann-Architektur.

Adreßregister (Adresse 7C02H / FC02H)							
Bit 7							Bit 0
HA15	HA14	HA13	HA12	HA11	HA10	<i>Res.</i> <sup>9</sup>	<i>Res.</i>

Das Adreßregister dient der Aufnahme des geschilderten Adreßmusters. Jedes Bit des Musters wird mit der entsprechenden Adreßleitung des Controllers verglichen (HA15 mit A15, ..., HA10 mit A10), was bedingt durch die zur Verfügung stehenden Adressen A15..A10 eine Granularität bei der Konfiguration von Harvard-Be-

<sup>7</sup>: Speicherbereich, in dem die Trennung zwischen CODE- und XDATA-Zugriffen aufgehoben ist; beide Zugriffsarten zielen auf den physikalisch gleichen Speicherbaustein, in der Regel ein RAM.

<sup>8</sup>: Speicherbereich, in dem CODE- und XDATA-Zugriffe auf physikalisch verschiedene Speicherbausteine abzielen; in der Regel wird für CODE-Zugriffe ein ROM oder Flash, für XDATA-Zugriffe ein RAM eingesetzt.

<sup>9</sup>: Reservierte Bits dürfen nicht verändert werden, der Reset-Inhalt 0 muß erhalten bleiben

reichen von min. 1 KB bewirkt. Blöcke kleiner 1 KB lassen sich demzufolge nicht einstellen.

#### 4.4 Maskenregister:

Das Maskenregister (Adresse 7C03H / FC03H) dient der Maskierung einzelner Bitstellen des Adreßregisters (s. o.) für den geschilderten Adreßvergleich. Nach einem Hardware-Reset sind alle Bits des Adreßregisters relevant, durch Setzen einzelner Bits im Maskenregister werden die entsprechenden Bitstellen des Adreßregisters nicht mehr in einen Adreßvergleich einbezogen.

Maskenregister (Adresse 7C03H / FC03H)							
Bit 7							Bit 0
MA15	MA14	MA13	MA12	MA11	MA10	Res. <sup>10</sup>	Res.

---

<sup>10</sup>: Reservierte Bits dürfen nicht verändert werden, der Reset-Inhalt 0 muß erhalten bleiben

---

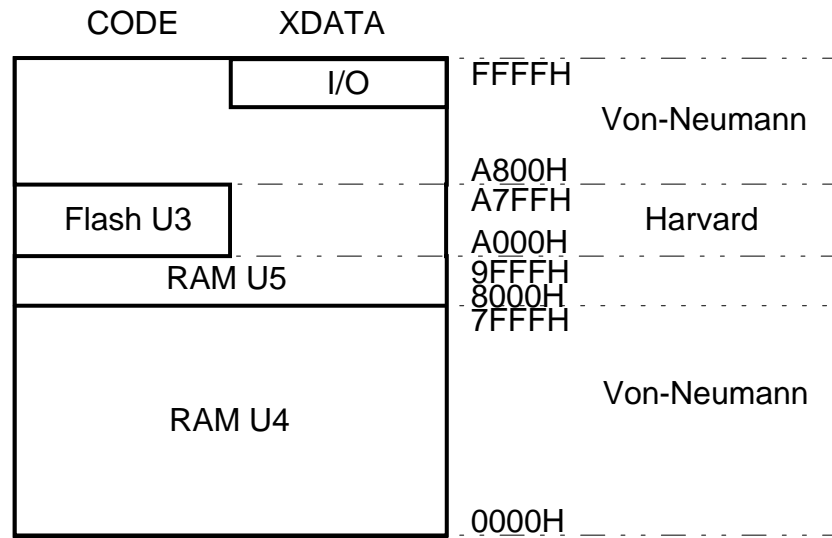
Folgende Beispiele für die Werte des Adreß- sowie des Maskenregisters verdeutlichen die Funktionsweise:

Adr.-Reg.	Mask.-Reg.	Bemerkung (nur für VN-EN = 1)
1XXXXX00 b	01111100 b	Harvard von 8000H-FFFFH, Von-Neumann von 0000H-7FFFH
0XXXXX00 b	01111100 b	Harvard von 0000H-7FFFH, Von-Neumann von 8000H-FFFFH
111111 00 b	00000000 b	Harvard von FC00H-FFFFH, Von-Neumann von 0000H-FBFFFH
010X00 00 b	00010000 b	Harvard von 4000H-43FFFH und von 5000H-53FFFH, Von-Neumann von 0000H-3FFFH, von 4400H-4FFFH und von 5400H-FFFFH
100000 00 b	00000000 b	Harvard von 8000H-83FFFH, Von-Neumann von 0000H-7FFFH und von 8400H-FFFFH
10100X 00 b	00000100 b	Harvard von A000H-A7FFFH, Von-Neumann von 0000H-9FFFH und von A800H-FFFFH

Reservierte Bits ohne Funktion für die Adreßdekodierung,  
s. Registerbeschreibungen

X=don't care (aufgrund gesetzter Bits im Maskenregister)

Das letzte Beispiel der Tabelle soll anhand des folgenden Bildes nochmals verdeutlicht werden:



PRG-EN = 0  
 VN-EN = 0  
 IO-SW = 0  
 RAM-SW = 0  
 Adr.-Reg. = 10100X00b  
 Mask.-Reg. = 00000100b

Bild 6: Beispiel-Speichermodell

## 5. Flash-Memory

Das miniMODUL-537/509 ist zur Bestückung mit 5V-programmierbaren Flash-Memories vorgesehen. Somit wird keine besondere Programmierspannung benötigt. Sofern Sie das Modul mit Flash-Memory bei uns erwerben, ist im Flash bereits ein Software-Werkzeug (sog. Flash-Tools, s. *Kapitel 9. Flash-Tools*) integriert, welches eine On-Board Reprogrammierung des Flashs ermöglicht.

**Sollte diese Software ohne gleichwertigen Ersatz aus dem Flash gelöscht werden, so ist eine On-Board Reprogrammierung nicht mehr möglich !**

Allerdings schützt sich die Software selbst gegen ein absichtliches oder versehentliches Löschen bzw. Überprogrammieren. Da der bei den verwendeten Flashtypen gebotene Hardware-Schutzmechanismus allerdings nicht verwendet wird, beschränkt sich der Schutz auf Software-Maßnahmen. Sie sollten beim eventuellen Einsatz eigener Programmieralgorithmen oder Werkzeuge unbedingt dafür Sorge tragen, daß ein Programmierwerkzeug im Flash verbleibt.

Der Einsatz des Flash-Bausteins als einziger Code-Speicher des Moduls bewirkt, daß das Flash nicht oder nur sehr bedingt zur nicht-flüchtigen Ablage von Daten geeignet ist. Dies ist durch die interne Architektur der Flash-Bausteine verursacht, da während des Flash-internen Programmierprozesses ein Lesen von Daten aus dem Baustein unmöglich ist. Demzufolge muß für eine Flashprogrammierung die Programmausführung aus dem Flash heraus verlagert werden (z.B. in Von-Neumann-RAM), was in der Regel einem einschneidenden Eingriff in den "normalen" Programmablauf gleichkommt.

Nach Stand der Technik zur Drucklegung dieses Manuals weisen die Flash-Bausteine eine Lebenserwartung von min. 100000 Lösch-/Programmierzyklen auf.

## 6. Die Batteriepufferung

Die zur Batteriepufferung nötige Batterie ist für die Grundfunktion des miniMODUL-537/509 nicht zwingend erforderlich. Allerdings bietet sich die Batteriepufferung als eine günstige und einfache Möglichkeit des nichtflüchtigen Abspeicherns von Daten an.

Der VBAT-Eingang am Modulpin 42 ist für den Anschluß einer externen Batterie vorgesehen. Wir empfehlen nach dem Stand der Technik zur Drucklegung dieses Manuals Lithium-Batterien, da diese hohe Kapazitäten bei sehr geringer Selbstentladung aufweisen. Die bestückten RAM-Bausteine werden bei fehlender Versorgungsspannung VCC von einer eventuell vorhandenen Batterie über VBAT gespeist.

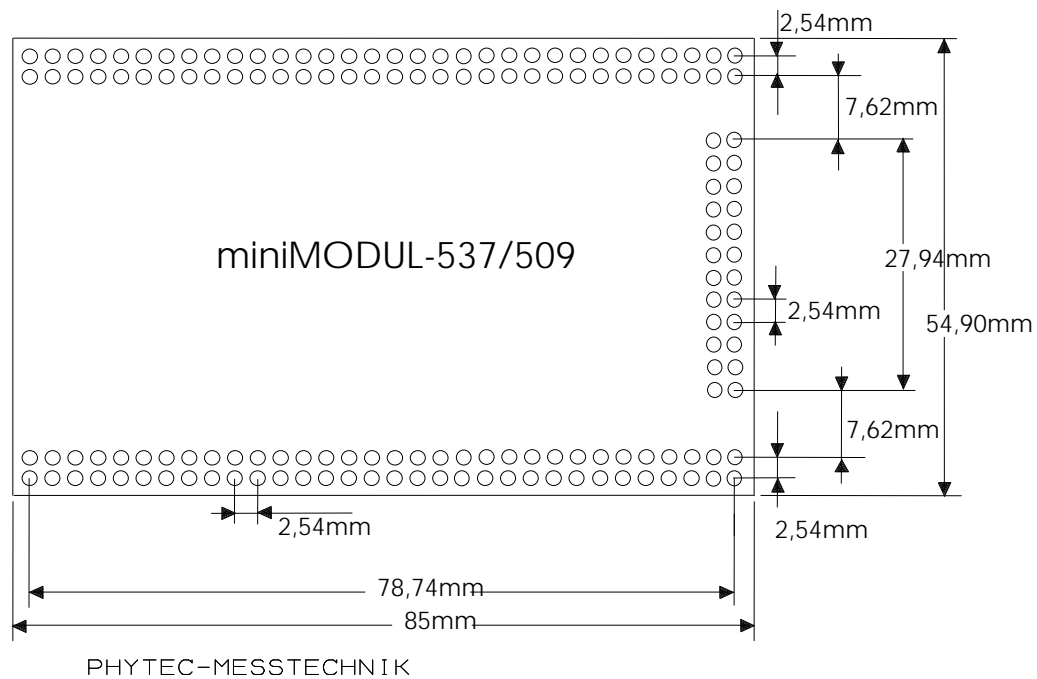
**Besondere Beachtung sollte hierbei die Bestückung eines EEPROMs an Position U5 des Moduls finden. Dieser Baustein ist aufgrund seiner Stromaufnahme sowie dem mit der EEPROM-Technologie verbundenen Datenerhalt nicht für eine Versorgung durch eine Batterie prädestiniert. In diesem Fall muß daher der Jumper J2 (s. Kap. "3.2 Speicherauswahl") entsprechend konfiguriert werden, da andernfalls eine vorzeitige Entladung einer angeschlossenen Batterie erfolgt.**

Die Stromaufnahme hängt von den verwendeten Bausteinen bzw. dem Speicherausbau ab. Sie beträgt bei den verwendeten Bausteinen pro RAM-Baustein typisch 1  $\mu$ A.

Aus Gründen der Betriebssicherheit möchten wir jedoch darauf hinweisen, daß trotz Batteriepufferung eine Veränderung der Dateninhalte im RAM infolge äußerer Störeinflüsse nicht absolut ausgeschlossen werden kann.

## 7. Technische Daten

Das miniMODUL-537/509 ist in seinen mechanischen Abmessungen in *Bild 7* dargestellt. Die Höhe des Moduls beträgt ohne Stiftleisten ca. 10 mm. Hierbei tragen die Bauteile jeweils ca. 3 mm auf der Platinenunterseite sowie ca. 5,5 mm auf der Oberseite auf. Die Platine selbst ist ca. 1,5 mm stark und besteht aus sechs Lagen.



*Bild 7: Mechanische Abmaße*

Weitere Daten:

- Modulgröße 54,90mm x 85mm  $\pm$ 0,01mm
- Gewicht ca. 44g bei Maximalausbau mit 160 KB RAM, gesockeltem Flash und gesockeltem Controller; Gewicht ca. 32g bei Standardausbau mit 32 KB RAM, ohne Sockel
- Lagertemperaturbereich -40°C bis +90°C
- Betriebstemperaturbereich Standard 0°C bis +70°C, erweitert -40° bis +85°C
- Luftfeuchtebereich max. 95% r.F. nicht kondensierend
- Betriebsspannungen 5V  $\pm$ 5%, VBAT 3V  $\pm$ 20%
- Stromaufnahme max. 140 mA, typ. 100 mA bei miniMODUL-537 mit 12 MHz Oszillatorfrequenz und 128 KB RAM bei 20°C
- Stromaufnahme bei Batteriepufferung max. 10  $\mu$ A pro RAM-Baustein, typisch 1  $\mu$ A pro RAM-Baustein bei 20°C

## **8. Hinweise zum Umgang mit dem Modul**

Beim Wechsel eines gesockelten Controllers ist zu beachten, daß der Sockel nicht durch unsachgemäße Werkzeuge beschädigt wird. Bitte verwenden Sie unbedingt ein passendes PLCC-Ausziehwerkzeug.

Von einem Wechsel des Quarzes oder Oszillators ist aufgrund der hohen Packungsdichte des Moduls generell abzuraten. Sollte dies wider Erwarten vonnöten sein, so ist zu beachten, daß beim Auslöten die Leiterplatte sowie umliegende Bauteile oder Sockel nicht beschädigt werden. Die Löt pads können sich bei Überhitzung von der Platine ablösen, wodurch das Modul unbrauchbar wird. Erhitzen Sie vorsichtig paarweise die benachbarten Anschlüsse, nach einigen Wechseln können Sie das Bauteil mit der Lötspitze abheben. Alternativ kann ein entsprechendes Heißluft-Werkzeug zur Erhitzung der Lötstellen verwendet werden.



## 9. Flash-Tools

Durch den Einsatz eines Flash-Memory als nichtflüchtiger Code-Speicher können Sie die Vorteile dieser modernen Technik nutzen. Hierzu zählt unter anderem die Möglichkeit der On-Board Programmierung des Flash-Memory. Zu diesem Zweck erhalten Sie bei Erwerb des miniMODUL-537/509 incl. Flash-Memory die sogenannten Flash-Tools in Form eines bereits vorprogrammierten Flash und einer entsprechenden PC-Software. Diese Werkzeuge ermöglichen während der Entwicklungsphase einen Download Ihrer Applikation in das RAM des Moduls oder alternativ bzw. nach Abschluß der Entwicklung eine Programmierung des Flash-Memory.

**Die Flash-Tools stellen durch entsprechende Software-Schutzmaßnahmen sicher, daß im Rahmen einer Flash-Programmierung die Flash-Tools selbst nicht überschrieben werden können.**

Als Flash-Baustein steht Ihnen nach momentanem Stand der Technik entweder ein 29F010 mit einer Bank à 64KB oder ein 29F040 mit 7 Bänken à 64 KB für Ihre Applikation zur Verfügung.

Die folgenden Ausführungen haben nur bei Verwendung der Flash-Tools Gültigkeit, sie sind beim Einsatz anderer Programmiermechanismen hinfällig.

Prinzipiell wird nach einem Reset des Moduls die Flash-Tools Firmware gestartet, welche entweder in einen Programmiermodus verfällt oder Ihre Applikation startet. Die Flash-Tools belegen immer die erste 64 KB Bank (Bank 0, FA[18..15] = 0000b) des verwendeten Flashes. Die verbleibenden Bänke stehen Ihnen für Ihre Applikation zur Verfügung.

Ihre Applikation wird immer in der zweiten 64 KB Bank (Bank 1, FA[18..15] = 0010b) gestartet, was bei der Anfertigung einer Software-Kopie der Registerinhalte des Adreßdekoders zu beachten ist. Von dieser Bank aus haben Sie die Möglichkeit, das Speichermodell

Ihren Vorstellungen entsprechend anzupassen sowie per Bank-Switching eventuell weitere vorhandene Flash-Bänke anzusprechen. Auf der im Lieferumfang enthaltenen Tool-Diskette finden Sie Programmierbeispiele und vorgefertigte Hexfiles zur Umschaltung in andere Flash-Bänke.

**Verwenden Sie innerhalb Ihrer Applikation keinesfalls die Flash-Bank 0, um durch Erhalt der Flash-Tools die Möglichkeit der On-Board Reprogrammierung zu gewährleisten.**

Neben dem Programmieren des Flashs besteht auch die Möglichkeit eines RAM-Downloads Ihrer Applikation zu Testzwecken, z.B. während der Entwicklungsphase. Dies bedeutet, daß Sie ein Hexfile in das RAM des Moduls übertragen können (RAM-Ausbau beachten), so daß die Daten bei eingeschalteter Versorgungsspannung zunächst erhalten bleiben. Wenn Sie in der Flash-Bank 1 zuvor eine geeignete Applikation plaziert haben, die nach einem Reset ein Von-Neumann Speichermodell einstellt (Voraussetzung zur Code-Ausführung aus dem RAM) und an eine geeignete Stelle im RAM springt, so können Sie Ihre Applikation zunächst aus dem RAM heraus ausführen.

Diese Vorgehensweise erspart Ihnen unter Umständen bis zur Fertigstellung Ihrer Applikation mehrere Lösch-/Programmierzyklen des Flash-Bausteins. Auf der im Lieferumfang enthaltenen Tool-Diskette finden Sie Programmbeispiele und vorgefertigte Hexfiles für die geschilderte Einstellung des Modells incl. Start einer Applikation ab einer zuvor hinterlegten Adresse aus dem RAM.

## 9.1 Starten der Flash-Tools

Um das miniMODUL-537/509 in den Programmiermodus zu versetzen, muß die Datenleitung D0 während des Resets durch einen Pull-Up-Widerstand mit max. 10 k $\Omega$  auf einen High-Pegel gezogen werden. Wir empfehlen einen 4,7 k $\Omega$ -Widerstand, wobei die Größenangabe lediglich als Richtwert dienen kann, da je nach externer

Datenbus-Beschaltung des Moduls abweichende Werte zu verwenden sind.

Verbinden Sie das Modul mit einer seriellen Schnittstelle Ihres PC und stellen Sie die Verbindung des Pull-Up-Widerstandes mit dem Anschluß D0 (Anschluß 74) des Moduls her. Legen Sie anschließend die Versorgungsspannung an oder lösen Sie einen Reset aus. Starten Sie nun das im Lieferumfang enthaltene Programm FLASHT.EXE. Die Aufrufzeile gestaltet sich zu:

*flasht [Nummer der Schnittstelle] [BR(Baudrate)]*

Die Flash-Tools arbeiten mit einer automatischen Baudratenanpassung. Hierbei wird die beim Aufruf von „FLASHT.EXE“ spezifizierte Baudrate auf dem Modul automatisch eingestellt, sofern die Features des verwendeten Controllers eine entsprechende Baudrate zulassen. Entsprechende Hinweise finden Sie auf der beiliegenden Tool-Diskette.

**Für den korrekten Start der Flash-Tools mit automatischer Baudratenerkennung müssen immer erst die Flash-Tools durch einen Reset oder Power-Up gestartet werden. Erst danach darf das Terminal-Programm FLASHT.EXE auf dem PC ausgeführt werden.**

**Da die verwendbaren Prozessoren unterschiedlich flexible Features zur Generierung der Baudrate besitzen, ist es möglich, daß die von Ihnen gewählte Baudrate nicht eingestellt werden kann. Sollte keine korrekte Übertragung zustande kommen, wiederholen Sie bitte den Vorgang und verwenden Sie eine niedrigere Baudrate beim Aufruf von FLASHT.EXE.**

Nach dem geschilderten Aufruf erscheint das Startmenü der Flash-Tools, das Ihnen die Auswahl zwischen der Flash-Programmierung und einem RAM-Download bietet.

## 9.2 Flash-Programmierung

Wenn Sie im Startmenü den Menüpunkt (1) gewählt haben, gelangen Sie in das Menü zur Flash-Programmierung. Dieses gestattet es Ihnen, Informationen über das Flash anzuzeigen, den Benutzerbereich des Flashs ganz oder teilweise zu löschen sowie das Flash zu programmieren. Sämtliche Menüpunkte sind selbsterklärend und beziehen sich immer nur auf die aktuell ausgewählte Flash-Bank.

Für eigene Programme stehen Ihnen bei Flash-Bausteinen vom Typ 29F040 bis zu 7 Bänke à 64 KB zur Verfügung. Durch einen entsprechenden Menüpunkt können Sie eine der Bänke für die folgende Programmierung auswählen. Falls ein Flash von Typ 29F010 bestückt ist, entfällt dieser Menüpunkt, da lediglich die Bank 1 zur freien Verfügung steht und per Default ausgewählt ist. Zur Programmierung sind ausschließlich Intel-Hexfiles zu verwenden.

Nachdem Sie das Flash programmiert haben, entfernen Sie den geschilderten Pull-Up-Widerstand an D0, um Ihr Programm durch einen normalen Reset zu starten.

**Die Flash-Tools stellen sicher, daß Sie diese beim Programmiervorgang weder löschen noch durch eigene Programme überschreiben können. Dadurch bleibt die Möglichkeit der On-Board Reprogrammierung erhalten.**

## 9.3 RAM-Download

Während der Entwicklungsphase kann der RAM-Download-Mechanismus zur Vermeidung von unnötigen Lösch-/Programmierzyklen des Flash eingesetzt werden. Hierzu steht Ihnen im Hauptmenü der Flash-Tools der Menüpunkt (2) zur Verfügung.

Nach der Anwahl dieses Punktes erscheint das Menü für den RAM-Download, welches Ihnen neben dem eigentlichen Transfer die Angabe einer Startadresse erlaubt. Diese Startadresse wird von der auf

der Tool-Diskette mitgelieferten Software zur Programmausführung aus dem RAM verwendet, um den Einsprung in Ihre Applikation zu finden. Die Adresse wird hierzu an einer definierten Stelle im RAM hinterlegt.

Die Flash-Tools versuchen, während des RAM-Downloads diese Startadresse automatisch zu ermitteln. Daher wird nach einem RAM-Download die niedrigste Adresse aus dem Hexfile als Startadresse angenommen. Bei Bedarf ist die Adresse manuell zu korrigieren.

Sämtliche Menüpunkte sind wiederum selbsterklärend. Durch den RAM-Download haben Sie die Möglichkeit, Ihr Programm vor einer Flash-Programmierung im RAM zu testen.

Beachten Sie bitte nochmals, daß das miniMODUL-537F/509 nach einem Reset immer das Programm in der Flash-Bank 1 ausführt. Um eine Applikation aus dem RAM zu starten, steht Ihnen daher eine entsprechende Software auf der Tool-Diskette zur Verfügung, die in die Flash-Bank 1 einzuprogrammieren ist. Diese Software führt die Einstellung eines Von-Neumann-Bereichs durch und startet Ihre Applikation durch den Sprung zu der von Ihnen definierten Startadresse im RAM. Diese Vorbereitungen gestatten den Start Ihrer Software durch einen normalen Reset.



## Revisionswechsel des miniMODUL-537/509

PHYTEC hat im Zuge einer Umstellung auf Flash-Technologie eine Revision des miniMODUL-537 durchgeführt. Hierbei wurde auf bestmögliche Kompatibilität geachtet, allerdings sind Unterschiede unausweichlich. Folgende Aufstellung dient Ihnen als Übersicht über die für einen Ersatz wesentlichen Unterschiede.

	miniMODUL-537 alt (MM-101)	miniMODUL-537/509 neu (MM-103)
Pin 1+2	VCC	frei (VCC wird zur Verbesserung der EMV-Eigenschaften lediglich über Pins 65+66 zugeführt)
Pin 12	/OFF	/PSEP
Pin 13	BRES	/WRP
Pin 14	RES2	/RDP
Pin 35	/WDO	/CS1
Pin 36	/WRO	/CS2
Pin 37	/PWR	/CS3
Pin 40	/Low-Line	/HPD
Pin 58	ROM_A15	/RESP
Pin 60	STDP	/IRTC
Pin 63+ 64	GND	frei (GND wird zur Verbesserung der EMV-Eigenschaften lediglich über Pins 127+128)
Pin 91	B0	PRGEN
Pin 92	B1	frei
Pin 101	/XCERAM2	Port9.7 vom C509
Pin 102	XRAM2	Port9.6 vom C509

Pin 103	/XCERAM1	Port9.5 vom C509
Pin 104	XRAM1	Port9.4 vom C509
Pin 105	/XOEROM	Port9.3 vom C509
Pin 106	XROM	Port9.2 vom C509
Pin 107	/XCEROM	Port9.1 vom C509
Pin 108	/	Port9.0 vom C509
U3	geeignet für OTPs/EPROMs mit 32Kx8/64Kx8 im PLCC-/LCC- Gehäuse	geeignet für Flash-Memories 29F010/29F040 mit 128Kx8/512Kx8 oder OTPs mit 128Kx8 im PLCC-Gehäuse

*Tabelle 2: Revisionswechsel*

---

**Dokument: miniMODUL-537/509**  
**Dokumentnummer: L-244-02, September 1996**

---

**Wie würden Sie dieses Handbuch verbessern?**

---

---

---

---

---

---

---

---

**Haben Sie in diesem Handbuch Fehler entdeckt?** Seite

---

---

---

---

**Eingesandt von**

Name: \_\_\_\_\_

Firma: \_\_\_\_\_

Adresse: \_\_\_\_\_

---

Kundennummer: \_\_\_\_\_

Manual gekauft am: \_\_\_\_\_

Rechnungsnummer: \_\_\_\_\_

Einsenden an: **PHYTEC Meßtechnik GmbH**  
Postfach 42 11 51  
D-55069 Mainz

---

Published by

**PHYTEC**

---

© PHYTEC Meßtechnik GmbH 1996

Ordering No. L-244-02  
Printed in Germany