

phyCORE-ADuC812

Hardware-Manual

Ausgabe April 2002

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.

Die Informationen in diesem Handbuch wurden sorgfältig überprüft und können als zutreffend angenommen werden. Dennoch sei ausdrücklich darauf verwiesen, daß die Firma PHYTEC Meßtechnik GmbH weder eine Garantie noch die juristische Verantwortung oder irgendeine Haftung für Folgeschäden übernimmt, die auf den Gebrauch oder den Inhalt dieses Handbuches zurückzuführen sind. Die in diesem Handbuch enthaltenen Angaben können ohne vorherige Ankündigung geändert werden. Die Firma PHYTEC Meßtechnik GmbH geht damit keinerlei Verpflichtungen ein.

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 2002 PHYTEC Meßtechnik GmbH, D-55129 Mainz.

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.

Informieren Sie sich:

	EUROPA	NORD AMERIKA
Adresse:	PHYTEC Technologie Holding AG Robert-Koch-Str. 39 D-55129 Mainz GERMANY	PHYTEC America LLC 255 Ericksen Avenue NE Bainbridge Island, WA 98110 USA
Angebots Hotline:	+49 (800) 0749832 order@phytec.de	+1 (800) 278-9913 info@phytec.com
Technische Hotline:	+49 (6131) 9221-31 support@phytec.de	+1 (800) 278-9913 support@phytec.com
Fax:	+49 (6131) 9221-33	+1 (206) 780-9135
Web Seite:	http://www.phytec.de	http://www.phytec.com

2. Auflage, April 2002

Einleitung	1
1 Kurzübersicht über das phyCORE-ADuC812	3
1.1 Blockschaltbild	6
1.2 Ansicht des phyCORE-ADuC812	7
2 Anschlußbelegung	9
3 Jumper	17
3.1 J2 Spannungsversorgung SRAMs U5/U13	20
3.2 J1, J9, J11 Speichergröße der SRAMs.....	20
3.3 J3, J4, J12 Serielle Schnittstelle.....	21
3.4 J5 Interrupt-Ausgang des CAN Controllers	22
3.5 J6 Interner oder externer Programmspeicher.....	22
3.6 J7 Interrupt-Ausgang der RTC	23
3.7 J8 CAN-Schnittstelle	23
3.8 J13, J14 Konfiguration von SDATA/MOSI und SCLOCK für I ² C-Bus.....	24
4 Speichermodelle	25
4.1 Speichermodell nach Reset.....	26
4.2 Runtime-Modell.....	27
4.3 Von Neumann Speichermodell.....	29
4.4 Programmiermodell	31
4.5 Control Register 1	33
4.6 Adressregister	39
4.7 Maskenregister.....	40
4.8 Steuerregister 1	43
4.9 Input Register 1.....	43
4.10 Output Register 1	44
5 Serielle Schnittstellen	45
5.1 RS-232 Schnittstelle	45
5.2 RS-485 Schnittstelle	45
5.3 CAN-Schnittstelle.....	46
6 Flash Speicher	47
6.1 On-Board Flash Speicher (U4)	47
6.2 On-Chip Flash Speicher.....	49
7 Serielles EEPROM (U11)	50

8	Echtzeituhr RTC-8563 (U12)	51
9	RESET-Controller (U6)	52
10	Remote Supervisor Chip (U7)	53
11	Batteriepufferung	54
12	A/D-Wandler und D/A-Wandler	55
13	Technische Daten	57
14	Hinweise zum Umgang mit dem Modul	59
15	Das phyCORE-ADuC812 auf dem Development Board	
	phyCORE LD 5V	61
	15.1 Konzept des Development Board phyCORE-LD 5V	61
	15.2 Anschlüsse und Jumper des Development Board	
	phyCORE-LD 5V	63
	15.2.1 Anschlüsse.....	63
	15.2.2 Jumper des Development Board phyCORE-LD 5V	64
	15.2.3 Nicht unterstützte Features und unzulässige Jumper-Stellungen.....	66
	15.3 Funktionsgruppen des Development Board	
	phyCORE-LD 5V	67
	15.3.1 Spannungsversorgung an X1	67
	15.3.2 Starten der FlashTools.....	69
	15.3.3 Erste serielle Schnittstelle an Buchse P1A.....	71
	15.3.4 Buchse P1B	72
	15.3.5 CAN-Schnittstelle an Stecker P2A	74
	15.3.6 RS-485 Schnittstelle an Stecker P2B	80
	15.3.7 Visualisierungs-LED D3	82
	15.3.8 Belegungen des phyCORE, des Expansion-Bus und des Patchfeldes im Überblick.....	83
	15.3.9 Batterieanschluß BAT1	90
	15.3.10 Nummernchip	90
	15.3.11 Stiftleiste X4.....	91
Index	93

Bild- und Tabellenverzeichnis

Bild 1:	Blockschaltbild phyCORE-ADuC812.....	6
Bild 2:	Ansicht des phyCORE-ADuC812 (Ansicht von oben)	7
Bild 3:	Ansicht des phyCORE-ADuC812 (Ansicht von unten)	8
Bild 4:	Pinout des phyCORE-ADuC812 (Ansicht Bestückungsseite)	9
Bild 5:	Zählmatrix des phyCORE-Connectors (Ansicht von oben)	11
Bild 6:	Zählweise der Jumper	17
Bild 7:	Lage der Jumper (Ansicht Platinenoberseite).....	17
Bild 8:	Lage der Jumper (Ansicht Platinenunterseite).....	18
Bild 9:	Speichermodell nach Hardware-Reset.....	26
Bild 10:	Runtime-Modell.....	28
Bild 11:	Von Neumann Modell	30
Bild 12:	Programmiermodell	32
Bild 13:	Flash-Programmiermodell	34
Bild 14:	Aufteilung des I/O-Bereichs	36
Bild 15:	Beispiel-Speichermodell.....	42
Bild 16:	Speicherbereiche des Flash.....	47
Bild 17:	Mechanische Abmaße.....	57
Bild 18:	Das Mehrplatinenkonzept mit phyCORE-ADuC812, Development Board und Erweiterungsplatine.....	62
Bild 19:	Lage der Anschlüsse auf dem Development Board phyCORE-LD 5V	63
Bild 20:	Zählweise der Jumper	64
Bild 21:	Lage der Jumper (Ansicht Bestückungsseite).....	65
Bild 22:	Default Jumper-Stellungen des Development Board phyCORE-LD 5V mit phyCORE-ADuC812.....	65
Bild 23:	Anschluß der Versorgungsspannung an X1	68
Bild 24:	Belegung der DB-9 Buchse P1A als RS-232 (Ansicht Vorderseite).....	71
Bild 25:	Belegung der DB-9-Buchse P1B (Ansicht Vorderseite)	72

Bild 26:	Belegung des DB-9 Steckers P2A (CAN-Treiber auf phyCORE-ADuC812)	75
Bild 27:	Belegung des DB-9 Steckers P2A (CAN Treiber auf Development Board)	76
Bild 28:	Belegung des DB-9 Steckers P2A (CAN-Treiber auf Development Board und galvanische Trennung).....	79
Bild 29:	Belegung des DB-9-Steckers P2B (Ansicht Vorderseite, RS-485 Mode)	80
Bild 30:	Zählweise für den Expansion-Bus	84
Bild 31:	Zählweise des Patchfeldes.....	84
Bild 32:	Anschluß des Nummernchips.....	91
Bild 33:	Pinbelegung Nummernchip.....	91
Tabelle 1:	Pinout phyCORE-Connector A/B (ADuC812, ADuC824).....	12
Tabelle 2:	Pinout phyCORE-Connector C/D (ADuC812, ADuC824).....	13
Tabelle 3:	Pinout phyCORE-Connector E/F (ADuC812, ADuC824).....	14
Tabelle 4:	Pinout phyCORE-Connector G/H (ADuC812).....	15
Tabelle 5:	Pinout phyCORE-Connector G/H (ADuC824).....	15
Tabelle 6:	Jumperbelegung	19
Tabelle 7:	J2 Spannungsversorgung SRAM U5/U13	20
Tabelle 8:	J1, J9, J11 Speichergröße der SRAMs	20
Tabelle 9:	J3, J4, J12 Konfiguration Serielle Schnittstelle	21
Tabelle 10:	J5 Interrupt-Ausgang des CAN Controller.....	22
Tabelle 11:	J6 Zugriff auf externen oder internen Programmspeicher	22
Tabelle 12:	J7 Interrupt-Ausgang der RTC.....	23
Tabelle 13:	J8 CAN-Schnittstelle.....	24
Tabelle 14:	J13 und J14 Konfiguration I ² C-Schnittstelle.....	24
Tabelle 15:	Control Register 1 des Adressdekoders	33
Tabelle 16:	Adressregister des Adressdekoders.....	39
Tabelle 17:	Maskenregister des Adressdekoders	40
Tabelle 18:	Beispiel Arbeitsweise Adressdekoder.....	41

Tabelle 19: Steuerregister 1 des Adressdekoders	43
Tabelle 20: Input Register 1 des Adressdekoders	43
Tabelle 21: Output Register 1 des Adressdekoders.....	44
Tabelle 22: Bestückungsmöglichkeiten U11 und Adressierung.....	50
Tabelle 23: Unzulässige Jumper-Stellung auf dem Development Board....	66
Tabelle 24: JP9, JP36 Konfiguration der Versorgungsspannung VCC1 und AVCC	67
Tabelle 25: JP9, JP36 Unzulässige Jumper-Stellungen der Hauptspannungsversorgung	68
Tabelle 26: JP28 Konfiguration des BOOT-Tasters.....	69
Tabelle 27: JP28 Konfiguration einer ständigen FlashTools Startbedingung.....	70
Tabelle 28: JP22, JP23, JP10 Konfiguration von BOOT über RS-232.....	70
Tabelle 29: Unzulässige Jumper-Stellung beim BOOT über RS-232.....	70
Tabelle 30: Jumper-Konfiguration für die erste RS-232 Schnittstelle	71
Tabelle 31: Jumper-Konfiguration der DB-9 Buchse P1B.....	72
Tabelle 32: Unzulässige Jumper-Stellungen bei der Konfiguration von P1B	73
Tabelle 33: Jumper-Konfiguration des CAN-Steckers P2A mit CAN-Treiber auf phyCORE-ADuC812	74
Tabelle 34: Unzulässige Jumper-Stellungen CAN-Stecker P2A (CAN-Treiber auf phyCORE-ADuC812)	75
Tabelle 35: Jumper-Konfiguration des DB-9 Steckers P2A mit CAN Treiber auf Development Board	76
Tabelle 36: Unzulässige Jumper-Stellungen CAN-Stecker P2A (CAN-Treiber auf Development Board)	77
Tabelle 37: Jumper-Konfiguration des DB-9 Steckers P2A mit CAN Treiber auf Development Board und galvanischer Trennung	78
Tabelle 38: Unzulässige Jumper-Stellungen bei galvanisch getrenntem CAN-Bus (CAN-Treiber auf Development Board)	79
Tabelle 39: Jumper-Konfiguration des DB-9 Steckers P1B als RS-485	80

Tabelle 40: Unzulässige Jumper-Stellung P2B als RS-485 Schnittstelle....	81
Tabelle 41: JP17 Konfiguration der Visualisierungs-LED D3	82
Tabelle 42: Daten/Adressbus-Pinzuordnung phyCORE-ADuC812 / Development Board / Erweiterungsplatine	85
Tabelle 43: Steuersignal-Pinzuordnung phyCORE-ADuC812 / Development Board / Erweiterungsplatine	86
Tabelle 44: Schnittstellen-Pinzuordnung phyCORE-ADuC812 / Development Board / Erweiterungsplatine	86
Tabelle 45: Pinzuordnung Input- und Outputport phyCORE-ADuC812 / Development Board / Erweiterungsplatine	87
Tabelle 46: Pinzuordnung Analogleisten phyCORE-ADuC812 / Development Board / Erweiterungsplatine	87
Tabelle 47: Pinzuordnung Analogleisten phyCORE-ADuC-824 / Development Board / Erweiterungsplatine	88
Tabelle 48: Nicht verwendete Pins des phyCORE-ADuC812 beim Development Board und der Erweiterungsplatine	88
Tabelle 49: Pinzuordnung Versorgungsspannung phyCORE-ADuC812 / Development Board / Erweiterungsplatine	89
Tabelle 50: JP19 Konfiguration des Nummernchip	90

Einleitung

Dieses Handbuch beschreibt nur die Schaltung und Funktionen des phyCORE-ADuC812, nicht aber die verschiedenen, bestückbaren Controller selbst. Es wird ergänzt durch das entsprechende Controllerhandbuch z.B. "ADuC812 Data Sheet" sowie die Dokumentation zu gegebenenfalls mitgelieferter Software. Bitte beachten Sie daher auch diese Dokumentationen.

In diesem Handbuch sowie im dazugehörigen Schaltplan werden Low-aktive Signale durch einen Schrägstrich "/" vor dem Signalnamen gekennzeichnet (z.B. "/RD"). Die Darstellung "0" deutet auf eine logische Null oder low-Pegel hin, während "1" für eine logische Eins oder high-Pegel steht.

Anmerkungen zum EMV-Gesetz für das phyCORE-ADuC812



Das phyCORE-ADuC812 (im Folgenden Produkt genannt) ist als Zulieferteil für den Einbau in ein Gerät (Weiterverarbeitung durch Industrie (siehe § 5 Abs. 5 EMVG) bzw. als Evaluierungsboard für den Laborbetrieb (zur Hardware- und Softwareentwicklung) bestimmt.

Achtung!

Das Produkt ist ESD empfindlich und darf nur an ESD geschützten Arbeitsplätzen von geschultem Fachpersonal ausgepackt und gehandhabt bzw. verarbeitet werden. Im Betrieb dürfen ohne weitere Schutzbeschaltung und Prüfung keine Leitungen von mehr als 3 m Länge an die Verbinder angeschlossen werden.

Das Produkt erfüllt die Anforderungen des EMVG (CE-Konformität) nur für den in diesem Handbuch beschriebenen Anwendungsbereich unter Einhaltung der gegebenen Hinweise zur Inbetriebnahme.

Nach dem Einbau in ein Gerät oder bei Änderungen/Erweiterungen an diesem Produkt muß die Konformität nach dem EMV-Gesetz neu festgestellt und bescheinigt werden. Erst danach dürfen solche Geräte in Verkehr gebracht werden.

Auszug aus dem EMVG § 5 Abs. 5

Geräte, die ausschließlich zur Verwendung in eigenen Laboratorien, Werkstätten und Räumen hergestellt, Anlagen, die erst am Betriebsort zusammengesetzt werden, und Netze bedürfen keiner EG-Konformitätserklärung und CE-Kennzeichnung.

Dies gilt auch für Bausätze, die ausschließlich für Funkamateure im Sinne des § 1 Abs. 2 hergestellt und bestimmt sind.

Geräte, die ausschließlich als Zulieferteile oder Ersatzteile zur Weiterverarbeitung durch Industrie, Handwerk oder sonstige auf dem Gebiet der elektromagnetischen Verträglichkeit fachkundige Betriebe hergestellt und bereitgehalten werden, brauchen weder die Schutzanforderungen gemäß § 4 Abs. 1 einzuhalten noch bedürfen sie einer EG-Konformitätserklärung oder CE-Kennzeichnung, vorausgesetzt, es handelt sich dabei nicht um selbständig betreibbare Geräte.

Das phyCORE-ADuC812 ist ein Modul aus der Serie der nano-/micro-/mini-/phyCORE-Module der Firma PHYTEC, die eine Bestückung mit verschiedenen Controllern erlauben, und dadurch eine Vielzahl von Funktionen und Konfigurationen ermöglichen.

PHYTEC unterstützt alle gängigen 8- und 16-bit-Controller auf zwei Arten:

- (1) als Grundlage für Starter Kits, die die Kombination mit benutzer-eigenen Schaltungen auf einem eigens dafür vorgesehenen Wrap-Feld erlauben und
- (2) als universelle, sofort einsetzbare, voll funktionsfähige micro-, mini- und phyCORE-MODULE, die direkt in die benutzereigene Peripherie-Schaltung eingesteckt werden können.

Mit dem Konzept der Microcontroller-Module von PHYTEC ist es Entwicklungsingenieuren möglich, Entwicklungszeiten zu verkürzen, Entwicklungskosten zu reduzieren und die Durchführung eines Projektes von der Idee bis zur Markteinführung wesentlich zu beschleunigen. Für weitere Informationen wenden Sie sich bitte an folgende Adressen:

	EUROPA	NORD AMERIKA
Adresse:	PHYTEC Technologie Holding AG Robert-Koch-Str. 39 D-55129 Mainz GERMANY	PHYTEC America LLC 255 Ericksen Avenue NE Bainbridge Island, WA 98110 USA
Web Seite:	http://www.phytec.de	http://www.phytec.com
e-mail:	info@phytec.de	info@phytec.com
Tel.:	+49 (6131) 9221-0	+1 (800) 278-9913
Fax:	+49 (6131) 9221-33	+1 (206) 780-9135

1 Kurzübersicht über das phyCORE-ADuC812

Das phyCORE-ADuC812 gehört zur Familie von PHYTECs phyCORE-Modulen. Die phyCORE-Modulfamilie stellt die konsequente Weiterentwicklung der bekannten mini-, micro- und nanoMODULE von PHYTEC dar. Genauso wie mini-, micro- and nanoMODULE vereinen die phyCORE Module alle notwendigen Komponenten eines Embedded Microcontroller Systems auf kleinster Fläche. Sie sind für die einfache Erweiterung bzw. Implementierung in unterschiedliche, periphere Schaltungen und kundenspezifische Applikationen designed.

Die phyCORE-Modulfamilie zeichnet sich unter anderem durch eine wesentlich erhöhte Anzahl von Anschlüssen aus. Dabei sind ca. 20 % aller Anschlüsse Ground-Verbindungen. Dies ist ein Resultat unabhängiger Untersuchungen, die gezeigt haben, daß ca. 70 % aller EMV-Problem (elektromagnetische Verträglichkeit) auf unzureichende Masseverbindungen zurückzuführen sind. Dies trifft insbesondere in Umgebungen mit hochfrequenter Störstrahlung zu. Auf Grund der hohen Anzahl von Masseanschlüssen bei der phyCORE-Modulfamilie werden die EMV-Eigenschaften erheblich verbessert, wodurch der Einsatz von phyCORE-Modulen die einfache Einhaltung von EMV-Richtlinien selbst in stark gestörten Umgebungen ermöglicht.

phyCORE-Module sind durch die konsequente Anwendung moderner SMD-Technik, Multilayer-Design, kleinsten Bauteileformen und lasergebohrten Microvias extrem kompakt. Damit können Benutzer von phyCORE Modulen diese 'state of the art' Techniken in ihrer eigenen Applikation implementieren und nutzen.

Das phyCORE-ADuC812 ist ein universelles Microcontrollerboard im Format 55 x 60 mm. Es kann mit den Microcontrollern ADuC812 und ADuC824 von Analog Device bestückt werden. Der ADuC812 verfügt über eine integrierte Referenzspannungsquelle, einen Temperatursensor, zwei 12-bit DA-Wandler und einen 12-bit 8 Kanal AD-Wandler. Der ADuC824 besitzt eine integrierte Referenzspannungsquelle, einen Temperatursensor, einen 24-bit und einen 16-bit AD-Wandler mit insgesamt 5 Kanälen und einen 12-bit DA-Wandler. Das universelle Design ermöglicht die einfache Integration des phyCORE-ADuC812 in eine große Vielzahl von Applikationen. Da alle Controllersignale sowie Ein- und Ausgänge auf Stiftleisten im Rastermaß 2,54 mm am Rande des Boards geführt sind, kann das phyCORE-ADuC812 wie ein großer Chip in eigene Zielhardware gesteckt werden.

Unser Haus sieht sich in diesem Zusammenhang in der Verantwortung, als zuverlässiger Partner, ISO 9001 zertifizierter und termintreuer Lieferant, den Einsatz unserer Microcontroller-Module in Ihrer Produktserie zu ermöglichen. Dies ist eine wichtige Voraussetzung für den Erfolg Ihres Produktes.

Die controllerspezifischen Eigenschaften entnehmen Sie bitte dem Handbuch zum Microcontroller, im Hardware-Manual zum phyCORE-ADuC812 wird auf keinerlei Besonderheiten des Controllers näher eingegangen, da diese für die grundlegende Funktion des phyCORE-ADuC812 ohne Belang sind.

Das phyCORE-AduC812 bietet folgende Features:

- Microcontroller-Board im Format 55 x 60 mm durch Einsatz moderner SMD-Technik
- bestückt mit Analog Devices ADuC812 Microcontroller
- 8 kByte on-chip Flash für Code und 640 Byte on-chip EEPROM-Flash für Daten
- integrierte Referenzspannung und Temperatursensor
- zwei 12-bit D/A-Wandler und ein 12-bit 8 Kanal A/D-Wandler verfügbar mit Microcontroller ADuC812
- ein 12-bit D/A-Wandler, ein 24-bit und ein 16-bit A/D-Wandler mit 5 Kanälen verfügbar bei Bestückung mit optionalem ADuC824
- verbesserte Störsicherheit durch Multilayer-Technik sowie verringerte Störabstrahlung durch verbesserte Ground-Anbindung
- alle digitalen Ports sowie Daten- und Adressleitungen am Platinenrand über Stiftleisten verfügbar
- analoge Eingänge bzw. Ausgänge sind über separate Stiftleisten verfügbar
- aufsetzbar auf die Anwendungsschaltung wie ein großer Chip
- 128 kByte bis 1 MByte SRAM on-board (SMD)¹
- 128 kByte bis 512 kByte externer Flash on-board (SMD)¹
- on-board Flash-Programmierung mit FlashTools
- keine separate Programmierspannung durch Verwendung von 5 V Flash-Bausteinen
- flexible, per Software konfigurierbare Adressdekodierung durch komplexen Logikbaustein
- Banklatches für den Flash im Adressdekoer integriert
- über zusätzlichen Pointer im Microcontroller linearer Zugriff auf 16 MByte Data
- wahlweise RS-232 oder RS-485-Schnittstelle
- optionale CAN-Schnittstelle mit SJA1000 und CAN-Treiber 82C251
- I²C-Real-Time Clock
- optional 2 bis 8 kByte I²C-EEPROM
- Reset-Logik und Batterieüberwachung

¹: North America: Support Hotline: + 1-800-278-9913 • <http://www.phytec.com>
Europe: Support Hotline: 0 800-0-749-832 • <http://www.phytec.de>

- Remote - Supervisory Circuit²
- freie Chip-Select Signale für einfachen Anschluß externer Peripherie
- I/O-Porterweiterung mit 8 TTL-Eingängen und 8 TTL-Ausgängen
- einzige Versorgungsspannung 5 V, typ. <120 mA
- für den einfachen Anschluß von Emulatoren von der Firma Accutron Limited ist das Signal /EA von außen zugänglich

1.1 Blockschaltbild

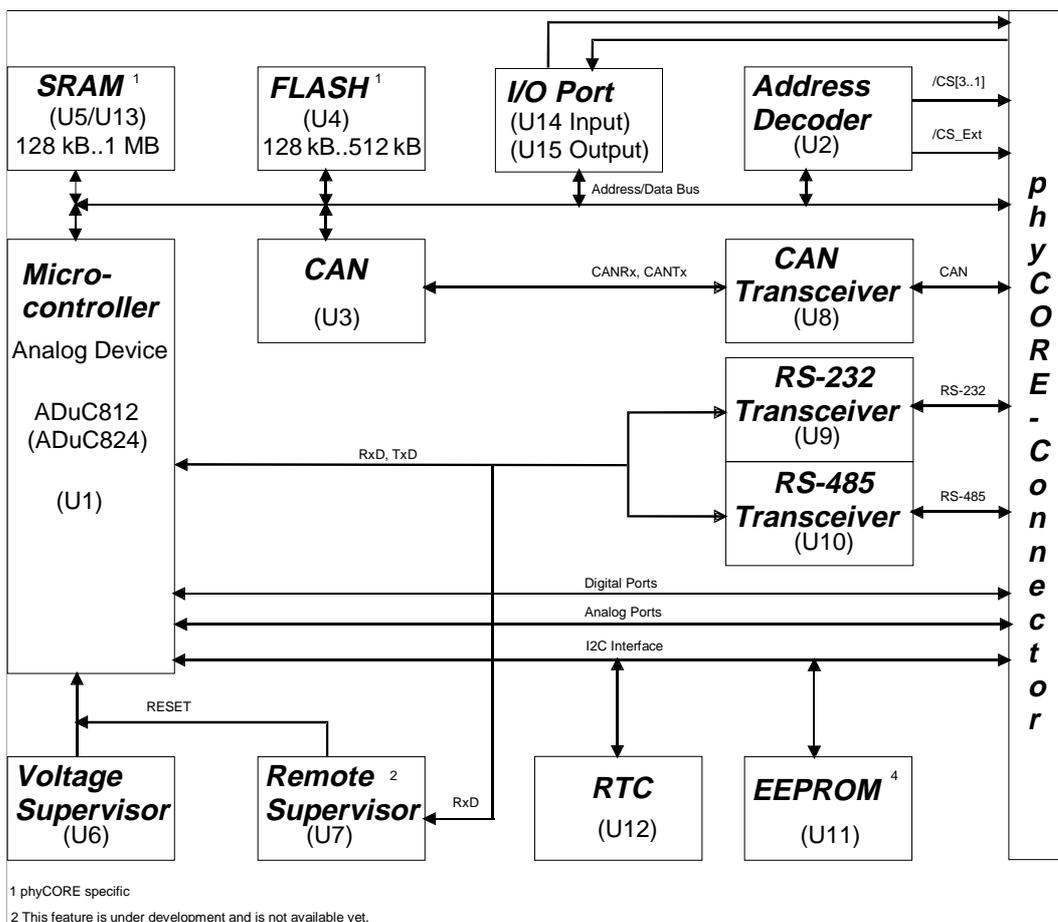


Bild 1: Blockschaltbild phyCORE-ADuC812

²: dieses Feature befindet sich noch in der Entwicklung und steht noch nicht zur Verfügung

1.2 Ansicht des phyCORE-ADuC812

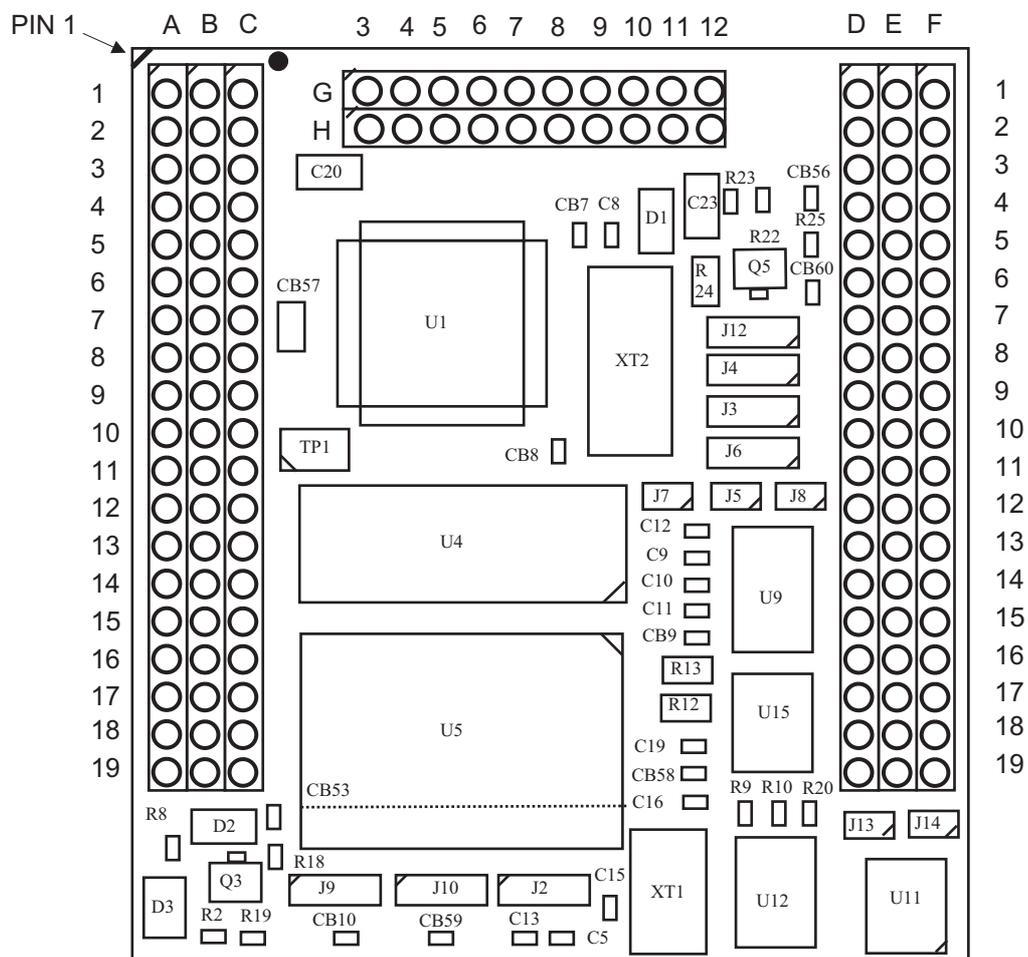


Bild 2: Ansicht des phyCORE-ADuC812 (Ansicht von oben)

2 Anschlußbelegung

Es sei ausdrücklich darauf hingewiesen, daß bei allen Modulanschlüssen 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 4* dargestellt, werden alle relevanten Signale an drei Seiten des Moduls auf Stiftleisten im Rastermaß 2,54 mm (im folgenden phyCORE-Connector genannt) an den Platinenrand geführt. Damit kann das phyCORE-ADuC812 wie ein großer Chip in verschiedene Applikationen integriert werden.

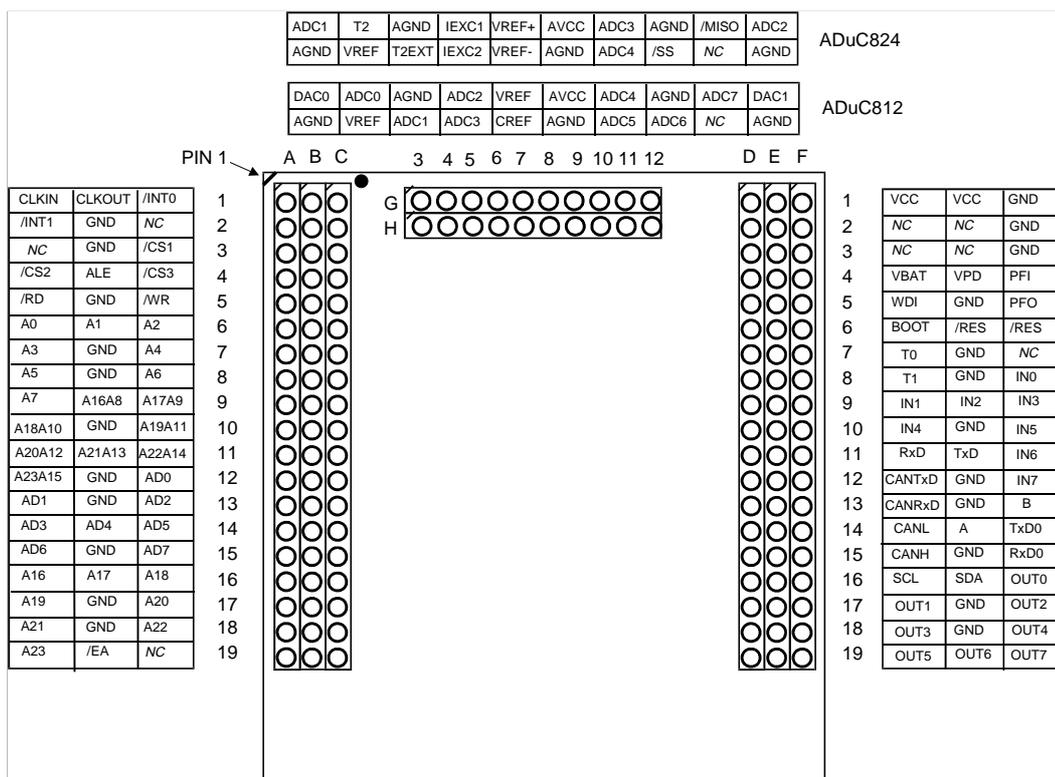


Bild 4: Pinout des phyCORE-ADuC812 (Ansicht Bestückungsseite)

Im Rahmen der phyCORE-Spezifikation wurde eine neue Zählweise für die Pins des phyCORE-Connectors eingeführt. Diese Zählweise erleichtert das Auffinden der Position von gesuchten Pins und beugt Mißverständnissen und Fehlern bei der Übertragung der Pinbelegung von den phyCORE-Modulen auf die entsprechenden Sockel der Anwendungsschaltung oder dem Development Board phyCORE-LD 5V vor.

Die Nummerierung des phyCORE-Connectors in dieser neuen Zählweise ergibt sich aus einer zweidimensionalen Zählmatrix bei der den Spalten Buchstaben und den Zeilen Zahlen zugeordnet sind. Dabei befindet sich Pin 1A immer in der linken, oberen Ecke der Zählmatrix. Die Ziffern werden von dieser Ecke nach unten aufsteigend gezählt, während die Buchstaben nach rechts aufsteigen (*siehe Bild 5*).

Die Zählmatrix wird gedanklich von oben auf das phyCORE-ADuC812 (Draufsicht; Stiftleiste des phyCORE-Connector zeigt nach unten) oder auf den entsprechenden Sockel des Development Board phyCORE-LD 5V / der Anwendungsschaltung gelegt. Dabei wird die linke obere Ecke der Zählmatrix (Pin 1A) mit der durch ein weißes Dreieck markierten Ecke des phyCORE-ADuC812 zur Deckung gebracht.

Selbst wenn ein phyCORE-Modul nur Kontakte auf der Unterseite führt, wird die Nummerierung immer gleichermaßen durch das "Auflegen" der Zählmatrix auf die Oberseite bzw. Draufsicht definiert.

Der Vorteil dieser Vorgehensweise besteht darin, daß jeder Pin des phyCORE-Moduls die gleiche Pin-Nummer führt wie der korrespondierende Pin des entsprechenden Sockels auf dem Development Board phyCORE-LD 5V. Die Verwechslungsgefahr ist damit ausgesprochen gering.

Da die Pins über die oben beschriebene Zählmatrix exakt definiert sind, werden die Steckverbinder des phyCORE-Connectors jeweils nur unter einem einzelnen Bezeichner (beispielsweise X1) zusammengefaßt. Sie bilden somit immer eine logische Einheit, ungeachtet der Tatsache, daß es sich um mehrere physikalische Steckverbinder handeln kann. Die mit einem weißen Dreieck versehene Ecke im Bestückungsdruck gibt die Position der Zeile 1 an und erleichtert damit die Orientierung.

Das folgende Bild (*siehe Bild 5*) dient der Verdeutlichung der Zählmatrix. Es zeigt ein auf dem Development Board phyCORE-LD 5V steckendes phyCORE-ADuC812. Der grau schattierte Bereich zeigt die nicht vom phyCORE-Modul genutzten Steckerleisten. Das Development Board phyCORE-LD 5V kann phyCORE-Module mit einer maximalen Anzahl von 32 Pins je Steckverbinderreihe A, B, C, D, E und F aufnehmen.

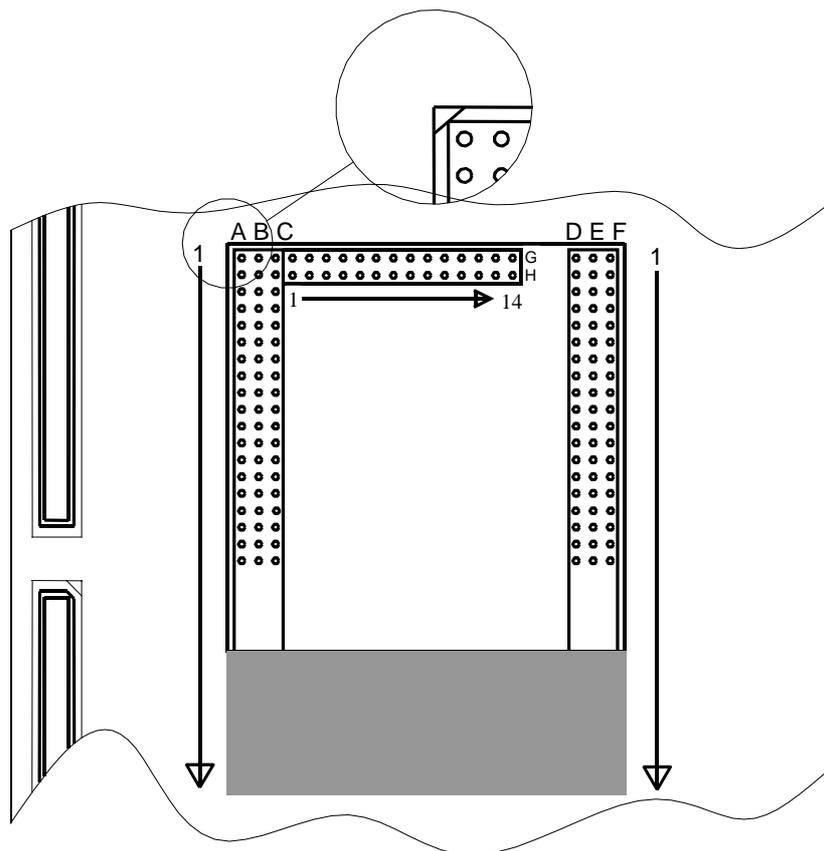


Bild 5: Zählmatrix des phyCORE-Connectors (Ansicht von oben)

Viele der am Platinenrand verfügbaren Portpins des Controllers sind mit alternativen Funktionen versehen, die in der Regel durch die Software entsprechend aktiviert werden müssen.

Die folgenden Tabellen geben eine Übersicht über die Belegung des phyCORE-Connectors, sowie Hinweise auf mögliche Alternativfunktionen einiger Portpins. *Bitte ziehen Sie im Zweifelsfall das Data Sheet des auf dem phyCORE-ADuC812 bestückten Controllers zu Rate.*

Pin Nr.	Bezeichnung	E/A	Beschreibung
Leiste X1A			
1A	ClkIn	E	optionaler externer Taktgenerator direkt mit XTAL1 des μ C verbunden
2A	P3.3 (/INT1)	E/A	Port Pin μ C
3A	NC		nicht verwendet
4A	/CS2	A	vordekodiertes Chip-Select Signal #2
5A	/RD	A	/RD-Signal
6A, 7A, 8A, 9A, 10A, 11A, 12A, 16A, 17A, A18A, 19A	A0, A3, A5, A7, A18A10, A20A12, A23A15, A16, A19, A21, A23	A	Adressleitungen des μ C bzw. des Adresslatches
13A, 14A, 15A	AD1, AD3, AD6	A	Adress-/Datenleitungen des μ C
Leiste X1B			
1B	ClkOut	-	direkt mit XTAL2 des μ C verbunden
2B, 3B, 5B, 7B, 8B, 10B, 12B, 13B, 15B, 17B, 18B	GND	-	Schaltungsmasse 0 V
4B	ALE	A	Adresslatch-Enable-Ausgang des μ C
6B, 9B, 11B, 16B	A1, A16A8, A21A13, A17	A	Adressleitungen des μ C bzw. des Adresslatches
14B	AD4	E/A	Adress-/Datenleitung des μ C
19B	/EA	E	/EA Pin des μ C

Tabelle 1: Pinout phyCORE-Connector A/B (ADuC812, ADuC824)

Pin Nr.	Bezeichnung	E/A	Beschreibung
Leiste X1C			
1C	P3.2 (/INT0)	E/A	Port Pins μ C
2C	NC		nicht verwendet
3C, 4C	/CS1, /CS3	A	vordekodiertes Chip-Select Signal #1, #3
5C	P3.6 (/WR)	E/A	/WR-Signal des μ C
6C, 7C, 8C, 9C, 10C, 11C, 16C, 17C, 18C	A2, A4, A6, A17A9, A19A11, A22A14, A18, A20, A22	A	Adressleitungen des μ C bzw. des Adresslatches
12C, 13C, 14C, 15C	AD0, AD2, AD5, AD7	E/A	Adress-/Datenleitung des μ C
19C	NC		nicht verwendet
Leiste X1D			
1D	VCC	-	Versorgungsspannung +5 V=
2D, 3D	NC	-	nicht verwendet
4D	VBAT	E	Anschluss für ext. Batterie (+)
5D	WDI	E	WDI-Eingang des RESET-Controllers
6D	BOOT	E	Boot=1 nach RESET → Starten des Boot-Vorganges
7D, 8D	P3.4 (T0), P3.5 (T1)	E/A	Port Pins μ C
9D,10D	IN1, IN4	E	Eingänge Digital Input Port
11D	RxD (P3.0)	E	RxD Eingang des μ C
12D	CANTxD	A	CANTxD-Ausgang des SJA1000
13D	CANRxD	E	CANRxD-Eingang des SJA1000
14D	CANL	E/A	CANL-Signal des CAN-Treibers
15D	CANH	E/A	CANH-Signal des CAN-Treibers
16D	SCL	A	Clock Ausgang I2C-Bus
17D, 18D, 19D	OUT1, OUT3, OUT5	A	Ausgänge Digital Output Port

Tabelle 2: Pinout phyCORE-Connector C/D (ADuC812, ADuC824)

Pin Nr.	Bezeichnung	E/A	Beschreibung
Leiste X1E			
1E	VCC	-	Versorgungsspannung +5 V=
2E, 3E	NC	-	nicht verwendet
4E	VPD	A	Ausgangsspannung für externe Pufferung (+)
5E, 7E, 8E, 10E, 12E, 13E, 15E, 17E,18E	GND	-	Schaltungsmasse 0 V
6E	/RES	A	Resetausgang des Modules direkt mit Reseteingang verbunden
9E	IN2	E	Eingang Digital Input Port
11E	TxD (P3.1)	E/A	TxD Ausgang des μ C (Portpin)
14E	A	E/A	differentielle A-Leitung des RS-485-Treibers
16E	SDA	E/A	Daten I ² C-Bus
19E	OUT6	A	Ausgang Digital Output Port
Leiste X1F			
1F, 2F, 3F	GND	-	Schaltungsmasse 0 V
4F	PFI	E	Power-Fail-Input des Reset-IC
5F	/PF0	A	Power-Fail-Output des Reset-IC
6F	/RES	E	Reseteingang des Modules
7F	NC	-	nicht verwendet
8F, 9F, 10F, 11F, 12F	IN0, IN3, IN5 IN6, IN7	E	Eingänge Digital Input Port
13F	B	E/A	differentielle B-Leitung des RS-485-Treibers
14F	TxD0	A	TxD-Ausgang des RS-232-Treibers
15F	RxD0	E	RxD-Eingang des RS-232-Treibers
16F, 17F, 18F, 19F	OUT0, OUT2 OUT4, OUT7	A	Ausgänge Digital Output Port

Tabelle 3: Pinout phyCORE-Connector E/F (ADuC812, ADuC824)

Pin Nr.	Bezeichnung	E/A	Beschreibung für ADuC812
Leiste X1G			
3G, 12G	DAC0, DAC1	A	Ausgänge D/A-Wandler
4G, 6G, 9G, 11G	ADC0, ADC2, ADC4, ADC7	E	Eingänge A/D-Wandler
5G,10G	AGND	-	analoge Schaltungsmasse 0 V
7G	VREF	-	Eingang Referenzspannung
8G	AVCC	-	analoge Versorgungsspannung +5 VDC
Leiste X1H			
4H	VREF	-	Ausgang Referenzspannung
5H, 6H, 9H, 10H	ADC1, ADC3, ADC5, ADC6	E	Eingänge A/D-Wandler
7H	CREF		Kondensator für Referenzspannung
3H, 8H, 12H	AGND	-	analoge Schaltungsmasse 0 V
11H	NC	-	nicht verwendet

Tabelle 4: Pinout phyCORE-Connector G/H (ADuC812)

Pin Nr.	Bezeichnung	E/A	Beschreibung für ADuC824
Leiste X1G			
4G	T2	E/A	Timer T2
6G	IEXC1	E	Eingang Kondensator C1
3G, 12G, 9G	ADC1, ADC2, ADC3	E	Eingänge A/D-Wandler
5G, 10G	AGND	-	analoge Schaltungsmasse 0 V
7G	VREF+	E	Eingang positive Referenzspannung
8G	AVCC	-	analoge Versorgungsspannung +5 VDC
11G	/MISO	E/A	/MISO für SPI Interface des Controllers
Leiste X1H			
4H	VREF+	-	Ausgang positive Referenzspannung
5H	T2EXT	E/A	T2EXT, Kontrolleingang für Timer 2
6H	IEXC2	E	Eingang Kondensator C2
7H	VREF-	E	Eingang negative Referenzspannung
9H	ADC4	E	Eingang A/D-Wandler
3H, 8H, 12H	AGND	-	analoge Schaltungsmasse 0 V
11H	NC	-	nicht verwendet

Tabelle 5: Pinout phyCORE-Connector G/H (ADuC824)

3 Jumper

Das phyCORE-ADuC812 besitzt zur Konfiguration 14 Lötjumper, die teilweise bereits bei der Auslieferung vorverbunden sind. *Bild 6* verdeutlicht die verwendete Zählweise bei den Jumpern, *Bild 7* und *Bild 8* die Lage der Jumper auf dem Board.

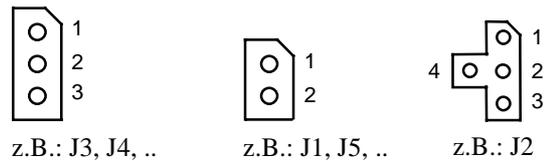


Bild 6: *Zählweise der Jumper*

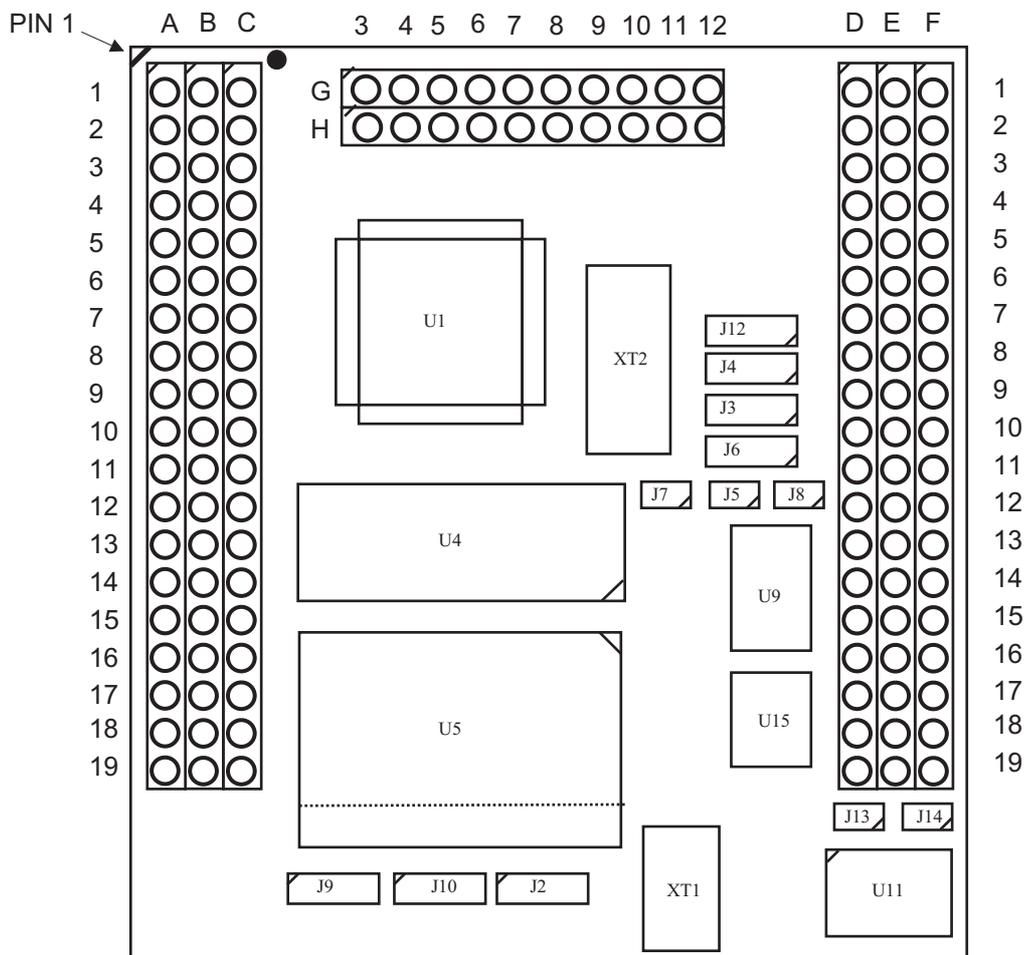


Bild 7: *Lage der Jumper (Ansicht Platinenoberseite)*

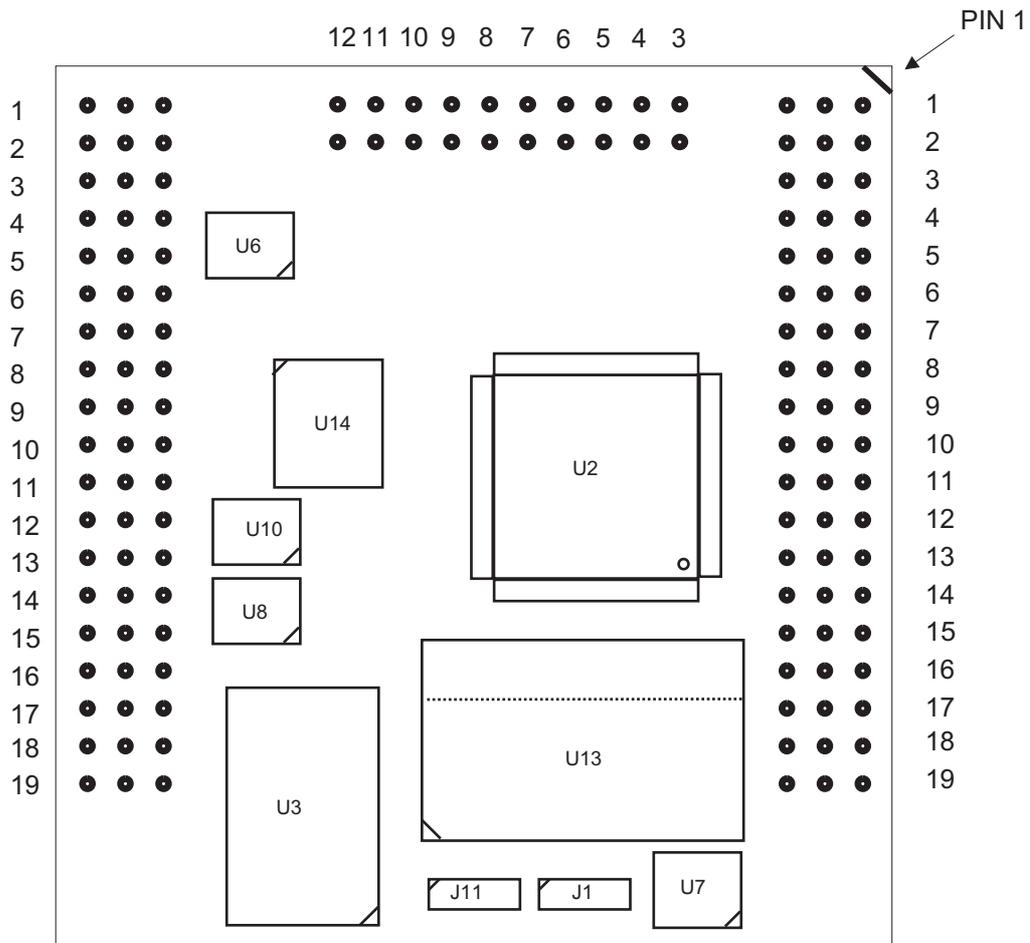


Bild 8: Lage der Jumper (Ansicht Platinenunterseite)

Die Jumper (J = Lötjumper) haben folgende Funktionen:

	Default-Einstellung	Alternative Einstellung
J1	(1+2) 128 kB auf SRAM U5 (RAM1)	(2+3) 512 kB auf SRAM U5 (RAM1)
J2	(1+2) VPD an SRAMs	(2+3) VCC an SRAMs
J3	(1+2) RxD mit RS-232 Treiber verbunden	(2+3) RxD mit RS-485 Treiber verbunden
J4	(1+2) TxD mit RS-232 Treiber verbunden	(2+3) TxD mit RS-485 Treiber verbunden
J5	(geschl.) Interrupt CAN an /INT0	(offen) Interrupt CAN nicht angeschlossen
J6	(1+2) mit 1 k Ω ; externer Programmspeicher	(2+3) mit 1 k Ω ; interner Programmspeicher
J7	(geschl.) Interrupt RTC an /INT1	(offen) Interrupt RTC nicht angeschlossen
J8	(geschl.) on-board CAN-Treiber mit CANRxD verbunden	(offen) on-board CAN-Treiber nicht mit CANRxD verbunden
J9	(1+2) 128 kB SRAM auf U5 (RAM1)	(2+3) 512 kB SRAM auf U5 (RAM1)
J10	(1+2) externer Speicher enabled	(2+3) externer Speicher disabled
J11	(1+2) 128 kB SRAM auf U13 (RAM2)	(2+3) 512 kB SRAM auf U13 (RAM2)
J12	(1+2) RS-485 Enable mit Signal OUT0	(2+3) RS-485 Enable mit Signal T1 verbunden
J13	(geschl.) PSDA Leitung vom I ² C-Bus mit Pin 27 des Controllers verbunden	(offen) PSDA Leitung vom I ² C-Bus nicht mit Controller verbunden
J14	(geschl.) PSCL Leitung vom I ² C-Bus mit Pin 26 des Controllers verbunden	(offen) PSCL Leitung vom I ² C-Bus nicht mit Controller verbunden

Tabelle 6: Jumperbelegung

3.1 J2 Spannungsversorgung SRAMs U5/U13

Mit dem Jumper J2 wird die Versorgungsspannung der SRAMs eingestellt. Wenn die SRAMs über die Batterie (VPD) gepuffert werden sollen, ist J2 auf Position 1+2 zu schliessen. Ist J2 auf Position 2+3 geschlossen, werden die SRAM Bausteine mit VCC verbunden.

Spannungsversorgung SRAM	J2
SRAMs an VPD, Spannungsversorgung mit Batteriepufferung	1 + 2*
SRAMs an VCC, Spannungsversorgung ohne Batteriepufferung	2 + 3

*= Default-Einstellung

Tabelle 7: J2 Spannungsversorgungs SRAM U5/U13

3.2 J1, J9, J11 Speichergröße der SRAMs

Die Jumper J1, J9, J11 werden bei der Produktion des Moduls fest eingestellt. Mit diesen Jumpers werden die Speicherbereiche und Speichergrößen der beiden SRAM Bausteine U5 und U13 definiert. J1 und J11 stellen die Speichergröße für U5 und U13 ein. Der SRAM auf U13 wird vom Adressdecoder ab Adresse 0x20000 (J9 auf Position 1+2 geschlossen) oder ab Adresse 0x80000 (J9 auf Position 2+3 geschlossen) in den Speicherbereich des Controllers einblendet.

Es ergeben sich folgende Konfigurationen:

SRAM Speichergröße	J1	J11	J9
U5 = 128 kB	1 + 2*		1 + 2*
U5 = 512 kB	2 + 3		2 + 3
U13 = 128 kB		1 + 2*	
U13 = 512 kB		2 + 3	

*= Default-Einstellung, Standard phyCORE-ADuC812

Tabelle 8: J1, J9, J11 Speichergröße der SRAMs

3.3 J3, J4, J12 Serielle Schnittstelle

Mit Hilfe der Jumper J3 und J4 kann eingestellt werden, ob die interne serielle Schnittstelle des Controllers mit dem RS-232-Treiber (U9) oder mit dem RS-485-Treiber (U10) verbunden werden soll. Wenn die Jumper auf Position 1+2 geschlossen sind (default), wird die serielle Schnittstelle als RS-232 Interface verwendet. Sind die Jumper J3 und J4 auf der Position 2+3 geschlossen, dann wird der RS-485 Treiber verwendet.

Mit Jumper J12 wird das Signal zur Steuerung des RS-485 Enable Signals konfiguriert. Dabei kann zwischen dem Controller-Signal T1 oder dem Signal OUT0 vom I/O-Port gewählt werden.

Die Verwendung der on-board FlashTools Firmware zum einfachen Download von Anwendungsprogrammen in den Flash des phyCORE-ADuC812 ist nur mit einer RS-232 Schnittstelle möglich. Die beiden Jumper J4 und J5 müssen dazu auf Position 1+2 geschlossen sein.

Konfiguration serielle Schnittstelle	J3/J4	J12
RS-232 Interface	1 + 2*	
RS-485 Interface, OUT0 als RS-485 Transceiver Enable Signal	2 + 3	1 + 2*
RS-485 Interface, T1 als RS-485 Transceiver Enable Signal	2 + 3	2 + 3

*= Default-Einstellung

Tabelle 9: J3, J4, J12 Konfiguration Serielle Schnittstelle

3.4 J5 Interrupt-Ausgang des CAN Controllers

Mit Jumper J5 kann ausgewählt werden, ob der Interrupt-Ausgang des CAN Controllers SJA1000 mit dem Interrupt-Signal /INT0 (Port 3.2) des Controllers verbunden ist. Um /INT0 (Port 3.2) als Standard I/O-Pin verwenden zu können, muß Jumper J5 geöffnet werden.

Es ergeben sich folgende Konfigurationen:

Interrupt-Ausgang CAN Controller	J5
Interrupt-Ausgang ist mit /INT0 (Port 3.2) verbunden	geschlossen*
/INT0 (Port 3.2) frei verfügbar	offen

*= Default-Einstellung

Tabelle 10: J5 Interrupt-Ausgang des CAN Controller

3.5 J6 Interner oder externer Programmspeicher

Der Jumper J6 ist im Auslieferungszustand des phyCORE-Moduls mit einen 1 kOhm Widerstand auf Position 1+2 bestückt. Dadurch wird nach einem Hardware-Reset das im externen Programmspeicher abgelegte Programm abgearbeitet. Um eine Programmabarbeitung aus dem internen Code-Speicher zu ermöglichen, muß der 1 kOhm Widerstand auf die Position 2+3 bestückt werden. Dieser Widerstand darf nicht durch einen 0R (Null Ohm) oder eine Lötbrücke ersetzt werden, weil sonst die Funktion eines Accutron-Emulators nicht sichergestellt ist.

Es ergeben sich folgende Konfigurationen:

Code-Zugriff	J6 (1 kΩ)
Externer Programmspeicher	1 + 2*
Interner Programmspeicher	2 + 3

*= Default-Einstellung

Tabelle 11: J6 Zugriff auf externen oder internen Programmspeicher

3.6 J7 Interrupt-Ausgang der RTC

Mit Jumper J7 kann ausgewählt werden, ob der Interrupt-Ausgang der RTC mit /INT1 (Port 3.3) verbunden ist. Durch Öffnen des Jumpers wird das Signal /INT1 (Port 3.3) an Pin X1C1 des phyCORE-Connectors frei verfügbar.

Es ergeben sich folgende Konfigurationen:

Interrupt-Ausgang der RTC	J7
Interrupt-Ausgang ist mit /INT1 (Port 3.3) verbunden	geschlossen*
/INT1 (Port 3.3) frei verfügbar	offen

*= Default-Einstellung

Tabelle 12: J7 Interrupt-Ausgang der RTC

3.7 J8 CAN-Schnittstelle

Das phyCORE-ADuC812 kann optional mit einem CAN-Controller SJA1000 auf U3 bestückt werden. Die Signale CANTx und CANRx des CAN-Controller sind an den Pins X1D12 und X1D13 des phyCORE-Connectors verfügbar. Bei Bestückung mit dem optionalen CAN-Controller ist zusätzlich ein CAN-Transceiver von Typ Philips PCA82C251 auf dem Modul vorhanden. Die Signale CANH und CANL von diesem Treiberbaustein sind an den Pins X1D14 und X1D15 des phyCORE-Connectors verfügbar und können direkt mit einem CAN-Zweidrahtbus verbunden werden.

Mit Jumper J8 kann festgelegt werden, ob die CANRx Leitung mit dem on-board CAN-Treiber verbunden wird oder ein externer CAN-Treiber angeschlossen werden soll. Für den letzteren Fall ist der Jumper J8 zu öffnen.

Es ergeben sich folgende Konfigurationen:

CAN-Transceiver	J8
on-board CAN Transceiver	geschlossen*
externer CAN Transceiver an CANRx und CANTx	offen

*= Default-Einstellung

Tabelle 13: J8 CAN-Schnittstelle

3.8 J13, J14 Konfiguration von SDATA/MOSI und SCLOCK für I²C-Bus

Auf dem phyCORE-ADuC812 sind zwei IC's mit I²C-Schnittstelle (Real-Time Clock auf U12 und EEPROM auf U11) vorhanden. Diese sind über die Jumper J13 und J14 mit den Portpins SDATA/MOSI und SCLOCK verbunden. Um die alternativen SPI-Signale dieser Pins zum Anschluss eines externen SPI-Interfaces verwenden zu können, sind die Jumper zu öffnen. Dabei ist zu beachten, daß in diesem Fall die auf dem phyCORE-Modul vorhandene RTC und das EEPROM nicht mehr benutzt werden können.

Es ergeben sich folgende Konfigurationen:

Konfiguration I²C	J13	J14
SDATA/MOSI als I ² C-SDA	geschlossen*	
SDATA/MOSI extern als MOSI	offen	
SCLOCK als I ² C-SCLOCK		geschlossen*
SCLOCK extern als SPI-SCLOCK		offen

*= Default-Einstellung

Tabelle 14: J13 und J14 Konfiguration I²C-Schnittstelle

4 Speichermodelle

Das phyCORE-ADuC812 verfügt über einen konfigurierbaren Adressdeko-der, 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-, einem Masken sowie einen Steuerregister innerhalb des Dekoders. Alle genannten Register sind im XDATA-Bereich des Controllers über Lese- bzw. Schreibzugriffe zu erreichen. 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 auf einen definierten Wert initialisiert, wodurch die Einstellung des bereits erwähnten Default-Speichermodells gewährleistet wird.

Achtung!

Falls Sie die FlashTools - eine Firmware zur komfortablen on-board Flash-Programmierung - verwenden, so ist zu beachten, daß beim Start Ihrer Anwendersoftware bereits die Adresse FA16 (*siehe Kapitel 3.5*) gesetzt wurde.

4.1 Speichermodell nach Reset

Nach einem Hardware-Reset entscheidet die on-board FlashTools Firmware, ob das Runtime-Modell zur Ausführung der Anwenderapplikation oder das Programmiermodell zur Ausführung der FlashTools eingestellt wird. Deshalb werden nach Reset die untersten 64 kByte des Flashs in den Programmspeicherbereich des Microcontrollers eingeblendet (FA18..FA16 = 000b).

Folgendes Bild zeigt das Speichermodell nach Reset:

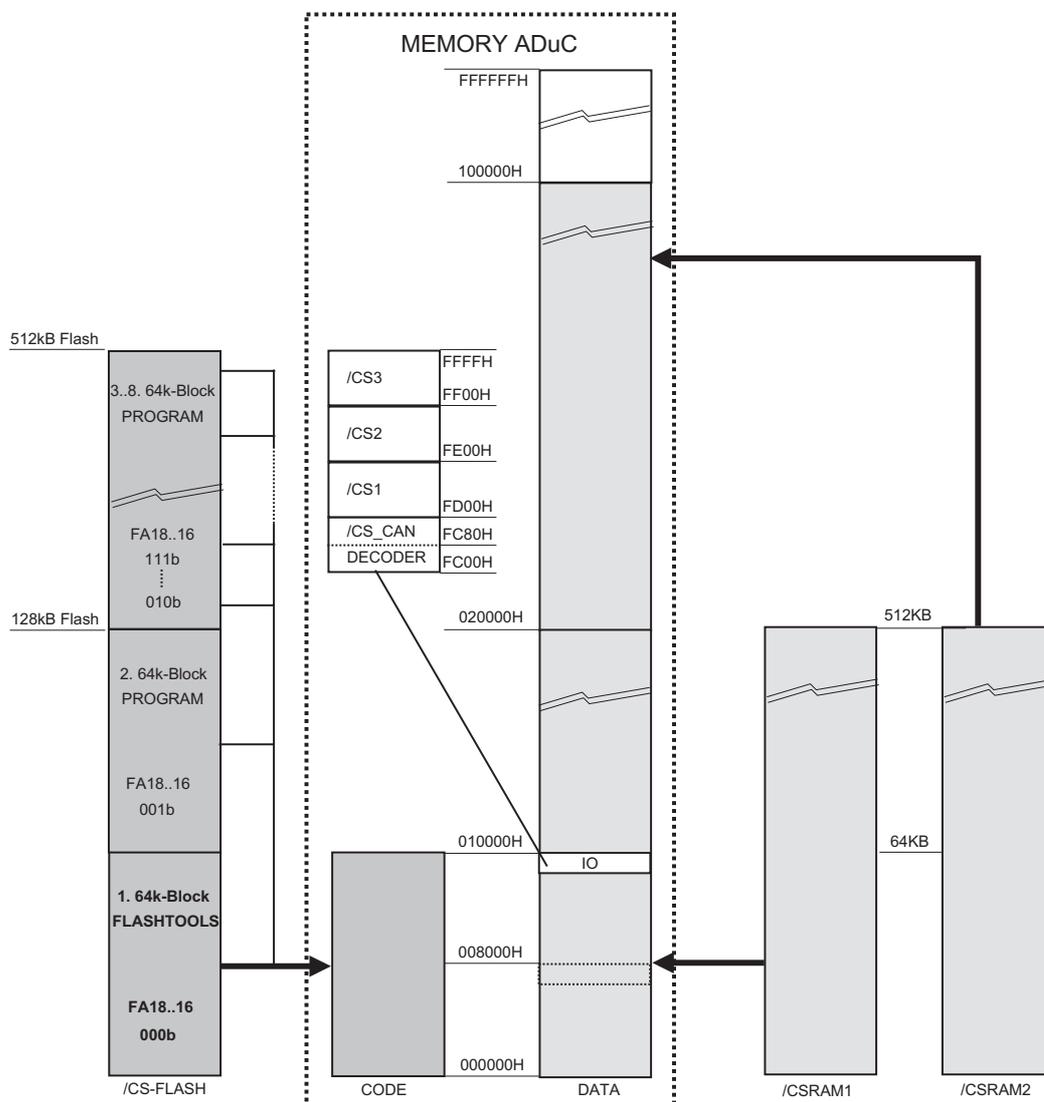


Bild 9: Speichermodell nach Hardware-Reset

Die FlashTools Firmware überprüft nun, ob die Bootbedingung aktiv ist (z.B. Boot-Taster auf dem phyCORE Development Board-LD 5V gedrückt). Wenn dies der Fall ist, wird das Programmiermodell konfiguriert. Ist die Bootbedingung nicht aktiv, wird das Runtime-Modell eingestellt.

4.2 Runtime-Modell

Beim Runtime-Modell wird die zweite 64 kByte Bank des Flashs in den Programmspeicherbereich des Microcontrollers eingeblendet und das Anwendungsprogramm ab Adresse 0000H (Flash-Adresse 10000H) gestartet.

In der Standard-Liefervariante ist das phyCORE-ADuC812 mit einem 128 kByte Flash bestückt. In den untersten 64 kByte befindet sich immer die FlashTools Firmware, der obere 64 kByte Flash-Sektor steht dem Anwender zur Ablage der Applikationssoftware zur Verfügung. Bei einer optionalen Bestückung des phyCORE-ADuC812 mit einem 512 kByte Flash-Baustein können sieben Blöcke à 64 kByte zur Ablage von Anwendungsprogrammen benutzt werden. Die Umschaltung zwischen den einzelnen Flash-Sektoren erfolgt über die Register FA16 bis FA18, *siehe Kapitel 4.5, "Control Register 1"*. Es ist zu beachten, daß beim Umschalten der 64 kByte Blöcke immer der gesamte Programmspeicher des Microcontrollers getauscht wird. Weiterhin darf das Registerbit FA15 nicht geändert werden.

Der Microcontroller ADuC812 bzw. ADuC824 kann bis zu 16 MByte Datenspeicher adressieren. Auf dem phyCORE-Modul werden in der Standardkonfiguration 128 kByte SRAM bestückt, optional sind 2* 512 kByte SRAM möglich. Der Speicherbereich für Daten läßt sich vom Microcontroller immer linear ansprechen. Ist der Baustein auf Position SRAM1 128 kByte groß, wird der auf SRAM2 bestückte Speicher ab Adresse 20000H aktiv. Ist der Baustein auf Position SRAM1 512 kByte groß, wird der Speicher auf SRAM2 ab Adresse 80000H aktiv.

Eine Ausnahme bildet der I/O-Bereich der default in den Adressbereich 00FC00H bis 00FFFFH eingeblendet wird. Mit Hilfe des Bits I/O-SW ist es möglich, den I/O-Bereich auf die Adressen 007C00H bis 007FFFH zu legen (siehe I/O-SW im „Control Register 1“).

Nachfolgendes Bild zeigt das Runtime-Modell:

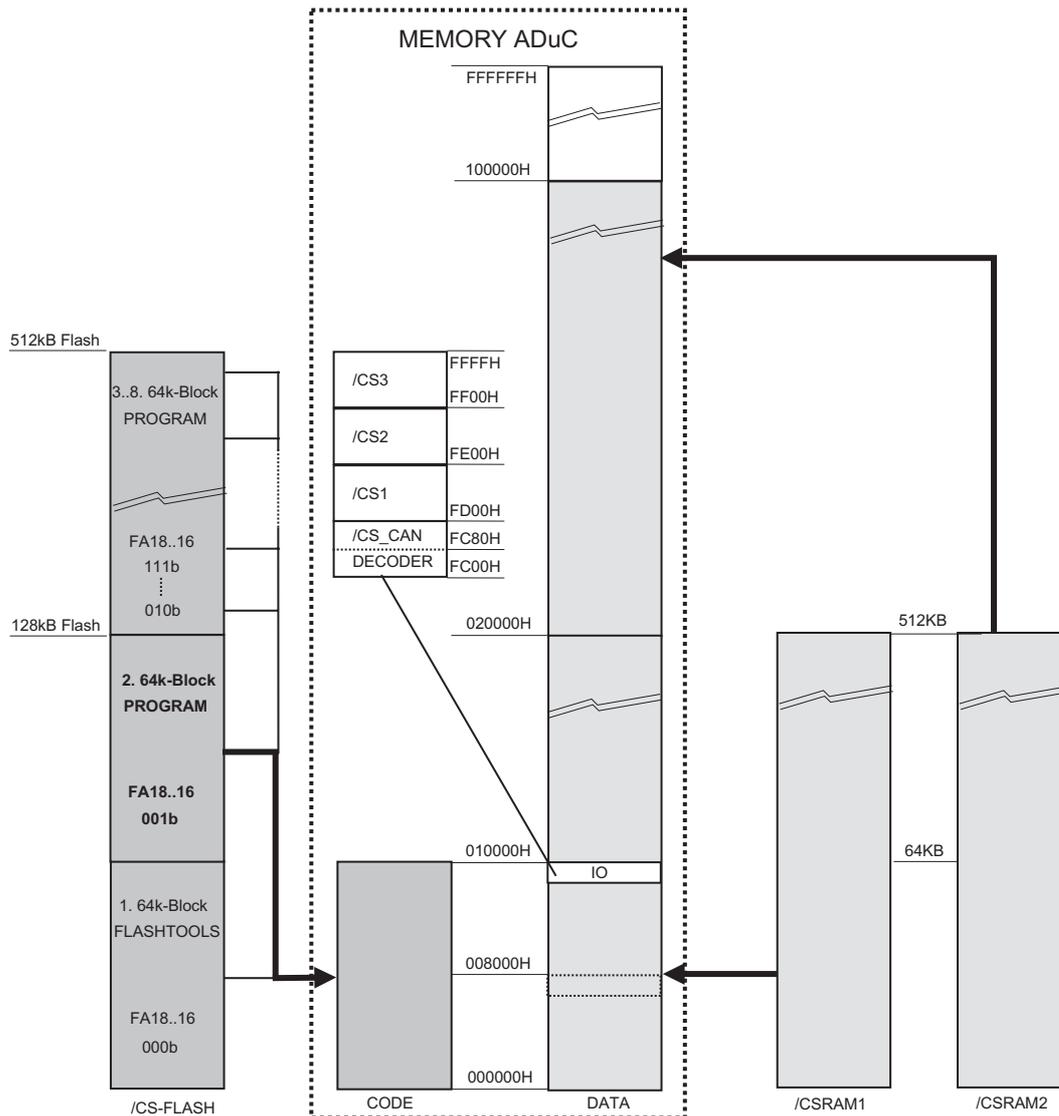


Bild 10: Runtime-Modell

4.3 Von Neumann Speichermodell

Der Adressdecoder des phyCORE-ADuC812 unterstützt ein von Neumann Speichermodell. Im von Neumann Speichermodell wird sowohl im CODE- als auch im DATA-Memory des Microcontrollers RAM eingeblendet. Dadurch ist es möglich, Maschinenbefehle im CODE-Bereich durch einen DATA-Schreibzugriff während der Programmausführung zu ändern. Das findet z.B. im Falle eines Monitorprogramms Verwendung, um einen Breakpoint für die Ausführung der Applikation zu setzen. Da sowohl CODE- als auch DATA-Memory im RAM abgebildet werden, sind beim Linken der Applikation getrennte Speicherbereiche zu verwenden.

Das phyCORE-ADuC812 unterstützt 64 kByte von Neumann Speicher für Code und Daten. Datenzugriffe über diese 64 kByte hinaus sind in diesem Speichermodell möglich. Das von Neumann Speichermodell wird durch das Setzen von Bit VN_EN im Control Register 1 aktiviert. Im von Neumann Speichermodell ist es möglich, einzelne Speicherbereiche innerhalb des 64 kByte Bereiches als Havard Architektur zu definieren. Dies wird im Kapitel Maskenregister in diesem Handbuch näher erläutert.

Im folgenden Bild ist das von Neumann Speichermodell dargestellt:

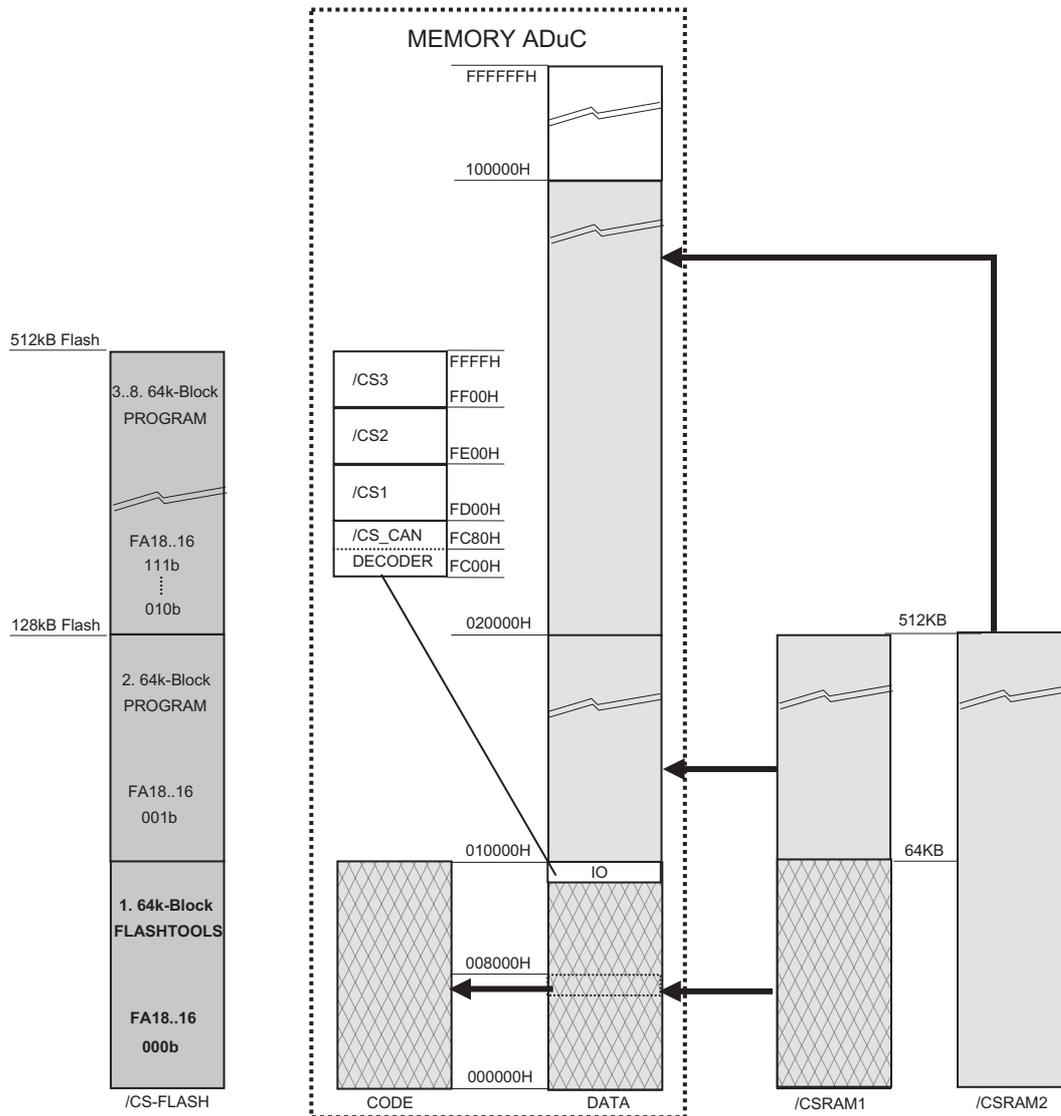


Bild 11: Von Neumann Modell

4.4 Programmiermodell

Dieses Modell wird innerhalb der FlashTools¹ zur Flash-Programmierung verwendet und ist aufgrund der vorhandenen Restriktionen nicht bzw. nur bedingt innerhalb Ihrer Applikation zu verwenden.

Die FlashTools Firmware kopiert sich nach einem Hardware-Reset und bei gleichzeitig aktiver Bootbedingung in die unteren 32 kByte des Speicherbereiches für Daten. Danach wird das Programmiermodell eingestellt und die FlashTools Firmware läuft nun aus dem SRAM heraus weiter. Es ist jetzt möglich, mit den Adressen FA15 bis FA18 jeden 32 kByte Flashblock in den Code- und Daten-Speicherbereich von 8000H bis FFFFH einzublenden und zu programmieren. Es ist zu beachten, daß sich in den untersten 64 kByte des Flashs die FlashTools Firmware befindet. Bei Verwendung von eigenen Programmerroutinen darf diese Firmware nicht gelöscht werden.

¹: Eine Firmware zur komfortablen on-board Flash-Programmierung; beim Erwerb des Moduls incl. Flash Speicher ist diese Software bereits in das Flash einprogrammiert.

Im folgenden Bild ist das Programmiermodell dargestellt:

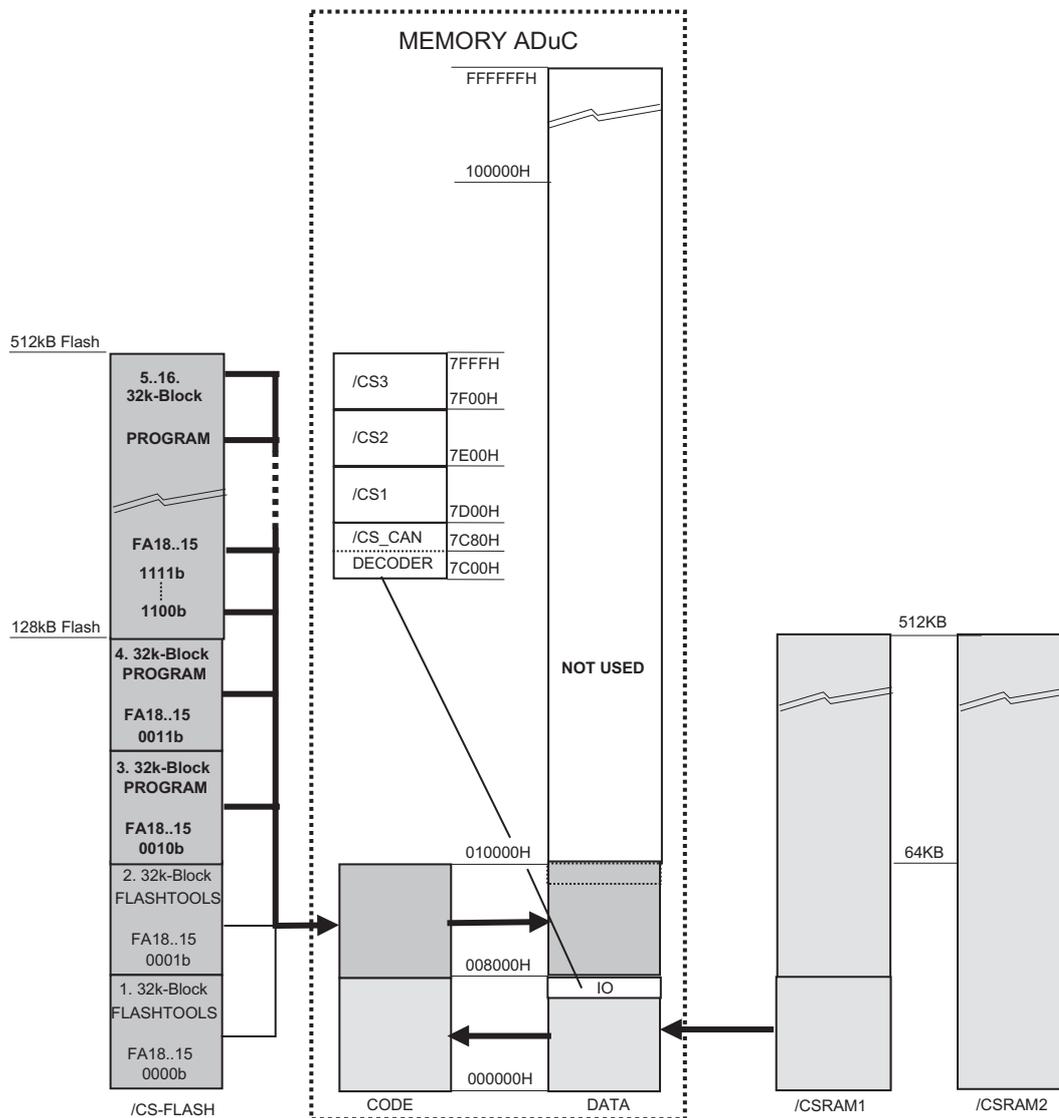


Bild 12: Programmiermodell

In den folgenden Abschnitten sind die Register des Adressdekoders zur Anpassung des Speichermodells erläutert. Die Register sind per default im I/O-Bereich auf den Adressen 00FC00H bis 00FC06H erreichbar. Wenn der I/O-Bereich auf 007C00H bis 007FFFH geschaltet wurde, ist ein Zugriff auf den Adressen 007C00H bis 007C06H möglich.

4.5 Control Register 1

Control Register 1 (Adresse 7C00H / FC00H)							
Bit 7							Bit 0
PRG-EN	IO-SW	Res.	VN-EN	FA18	FA17	FA16 ¹	FA15
Anfangswerte:							
Reset Value:				0000 0000 b			
Run Time Modell:				0000 0010 b			

Tabelle 15: Control Register 1 des Adressdekoders

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

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

PRG-EN: Dient dem Aktivieren des gesonderten Flash-Programmiermodells (PRG-EN = 1). Dieses Modell wird innerhalb der FlashTools 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 kByte RAM im Adressbereich von 0000H-7FFFH sowie auf 32 kByte Flash im Bereich von 8000H-FFFFH. Zugriffe über 64 kByte XDATA sind im Programmiermodell nicht erlaubt. 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.

¹: Bei Einsatz der FlashTools - 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.

Nur im Programmiermodell wird die Adressleitung A15 des Flashs ebenfalls vom Control Register 1 (Bit 0, FA15) gesteuert, im Runtime-Modell (PRG-EN = 0) wird die Adressleitung A15 des Controllers unmittelbar an das Flash durchgeschleift. Das Bits IO-SW bleibt auch im Programmiermodell relevant, das Bit VN-EN hingegen nicht.

Folgendes Bild verdeutlicht das Programmiermodell (I/O-Bereich nicht dargestellt):

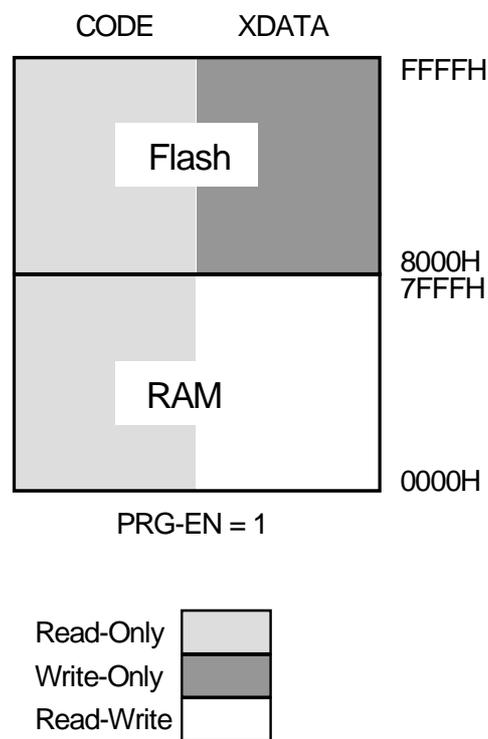


Bild 13: Flash-Programmiermodell

IO-SW: Mittels dieses Bits kann der I/O-Bereich des Moduls wahlweise in die oberen oder die unteren 32 kByte 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 2 Blöcken à 128 Bytes und 3 Blöcken à 256 Bytes. In den drei 256 Byte Blöcke stellt der Adressdekorer jeweils ein vordekodiertes Chip-Select Signal zur Verfügung (/CS1.../CS3), 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. Das Chip-Select Signal /CSCAN steuert den optionalen on-board CAN-Controller an. Der /CSREG Block ist reserviert für Zugriffe auf die internen Register. 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:

/CS3	FFFFH / 7FFFH*
/CS2	FF00H / 7F00H*
/CS1	FE00H / 7E00H*
/CSCAN	FD00H / 7D00H*
/CSREG	FC80H / 7C80H*
	FC00H / 7C00H* (* IO_SW = 1)

Bild 14: Aufteilung des I/O-Bereichs

Hierbei sind /CS1 bis /CS3 die frei verfügbaren Chip-Select-Signale, /CSCAN steuert den CAN-Controller auf dem Modul an. 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 FlashTools zur on-board-Programmierung des Flashes zu gewährleisten. Die internen Register belegen momentan lediglich die Adressen 7C00H-7C04H bzw. FC00H-FC04H, der Rest des /CS-REG-Blockes bleibt ungenutzt und ist für künftige Erweiterungen reserviert.

VN-EN: Mit diesem Bit werden im Adressraum des Controllers optionale von Neumann¹ Speicherbereiche freigeschaltet. Nach einem Reset ist per default eine Harvard²-Architektur vorhanden. Von Neumann Speicherbereiche sind insbesondere dann sinnvoll, wenn zur Laufzeit Programmcode nachgeladen und anschließend ausgeführt werden soll (z.B. Monitor-Anwendung). Die Lage dieser optionalen von Neumann Speicherbereiche wird über das Adress- sowie das Maskenregister definiert (*siehe Kapitel 4.7, "Maskenregister"*).

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, andernfalls ist es ohne Bedeutung und wird ignoriert.

-
- ¹: 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.
- ²: 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.
-

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

Im Runtime-Modell (PRG-EN = 0) können Bänke à 64 kByte 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 kByte Flash zugegriffen werden kann, dient es als Adressleitung A15 am Flash-Baustein. Im Runtime-Modell (PRG-EN = 0) mit 64 kByte 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 kByte aus.

4.6 Adressregister

Das Adressregister (Adresse 7C02H / FC02H) dient zusammen mit dem Maskenregister (s.u.) der Definition von von Neumann¹- und Harvard²-Speicherbereichen im Adressraum des Controllers. Durch Setzen des Bits VN-EN im Control Register 1 werden die Einstellungen freigeschaltet und in die Adressdekodierung einbezogen (siehe Control Register 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 das RAM 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 variabler Breite. 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	Res. ³	Res.	Res.	Res.
Reset Value:				0000 0000 b			

Tabelle 16: Adressregister des Adressdekoders

-
- 1: 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.
 - 2: 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.
 - 3: Reservierte Bits dürfen nicht verändert werden, der Reset-Inhalt 0 muß erhalten bleiben
-

Das Adressregister dient der Aufnahme des geschilderten Adressmusters. Jedes Bit des Musters wird mit der entsprechenden Adressleitung des Controllers verglichen (HA15 mit A15, ..., HA12 mit A12), was bedingt durch die zur Verfügung stehenden Adressen A15..A12 eine Granularität bei der Konfiguration von Harvard-Bereichen von minimal 4 kByte bewirkt. Blöcke kleiner 4 kByte lassen sich demzufolge nicht einstellen.

4.7 Maskenregister

Das Maskenregister (Adresse 7C03H / FC03H) dient der Maskierung einzelner Bitstellen im Adressregister. 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	Res. ¹	Res	Res	Res.
Reset Value:				0000 0000 b			

Tabelle 17: Maskenregister des Adressdekoders

¹: 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 (dargestellt sind A15 – A8):

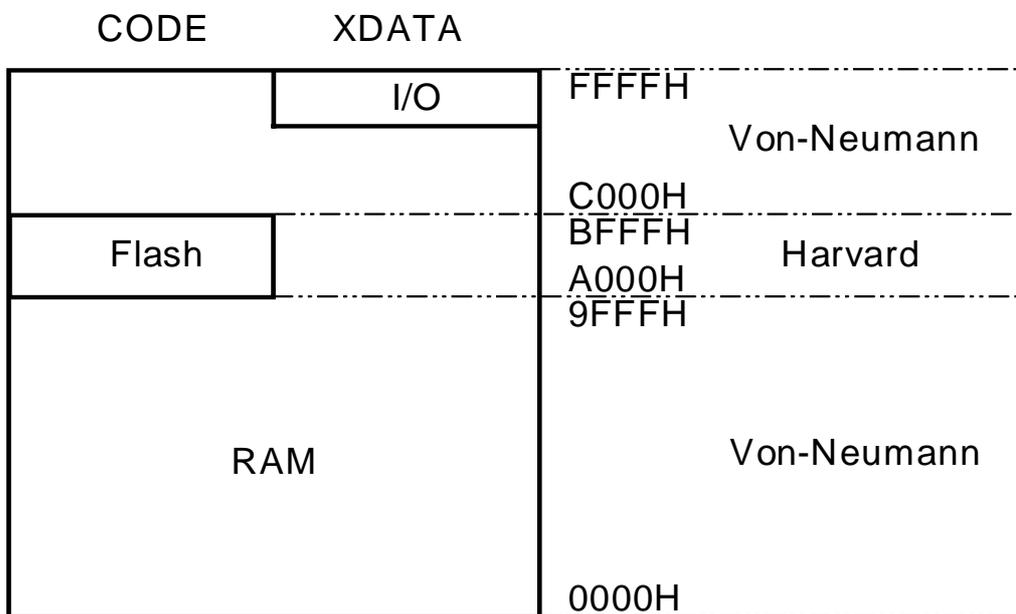
Adress-Reg.	Masken-Reg.	Bemerkung (nur für VN-EN = 1)
1XXX0000 b	01110000 b	Harvard 8000H-FFFFH, von Neumann 0000H-7FFFH
0XXX0000 b	01110000 b	Harvard 0000H-7FFFH, von Neumann 8000H-FFFFH
11110000 b	00000000 b	Harvard F000H-FFFFH, von Neumann 0000H-EFFFH
01X00000 b	00100000 b	Harvard 4000H-4FFFH 6000H-6FFFH, von Neumann 0000H-3FFFH, 5000H-5FFFH 7000H-FFFFH
10000000 b	00000000 b	Harvard 8000H-8FFFH, von Neumann 0000H-7FFFH 9000H-FFFFH
101X0000 b	00010000 b	Harvard A000H-BFFFH, von Neumann 0000H-9FFFH C000H-FFFFH

Tabelle 18: Beispiel Arbeitsweise Adressdekoder

Reservierte Bits ohne Funktion für die Adressdekodierung, siehe 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 = 1
 IO-SW = 0
 RAM-SW = 0
 Addr.-Reg. = 101X0000b
 Mask.-Reg. = 00010000b

Bild 15: Beispiel-Speichermodell

Nach einem Hardware-Reset ist der gesamte Speicher als Harvard strukturiert. Erst durch Setzen des VN-EN-Bit im Control Register 1 kann die Konfiguration über das Adress- und Maskenregister vorgenommen werden!

4.8 Steuerregister 1

Steuerregister 1 (Adresse 7C04H / FC04H)							
Bit 7							Bit 0
OE1	N/A	N/A	N/A	N/A	N/A	N/A	BOOT
Anfangswerte:							
Reset Value:				1000 000x b			
Run Time Modell:				1000 000x b			

Tabelle 19: Steuerregister 1 des Adressdekoders

BOOT: Auf Bit 0 des Steuerregisters 1 kann der Status des BOOT-Eingangs des Moduls zurückgelesen werden.

OE1: Das phyCORE-ADuC812 ist mit einem TTL-Ausgangslatch (74AHC573 auf U15) bestückt. Mit dem Bit OE1 werden die Ausgänge des Latch aktiv geschaltet.
(OE1 = 0 Latch aktiv, OE1 = 1 Latch inaktiv).

4.9 Input Register 1

Input Register 1 (Adresse 7C05H / FC05H)							
Bit 7							Bit 0
IN7	IN6	IN5	IN4	IN3	IN2	IN1	IN0
Anfangswerte:							
Reset Value:				xxxx xxxx b			

Tabelle 20: Input Register 1 des Adressdekoders

IN0...7: Das phyCORE-ADuC812 ist mit einem TTL-Treiber (74AHC245 auf U14) bestückt. Über das Input Register 1 können die Eingangssignale IN0...7 ausgelesen werden. Es ist zu beachten, daß der TTL-Treiber je Eingang einen Pull-up Widerstand von 100 kOhm gegen VCC besitzt. Dies hat zur Folge, daß bei Nichtbeschalten des Einganges das entsprechende Bit im Input Register 1 den Wert "1" zurückliefert.

5 Serielle Schnittstellen

5.1 RS-232 Schnittstelle

Auf dem phyCORE-ADuC812 befindet sich ein RS-232-Treiber (U9) für die Pegelanpassung der Leitungen P3.0/RxD0 und P3.1/TxD0. Die Schnittstelle kann für die Verbindung zu einer COM-Schnittstelle (z.B. PC) verwendet werden. Hierzu ist die RxD0-Leitung (X1F15) mit der TxD-Leitung der COM-Schnittstelle und die TxD0-Leitung (X1F14) mit der RxD-Leitung der COM-Schnittstelle zu verbinden. Weiterhin ist die Schaltungsmasse des phyCORE-ADuC812 an der Schaltungsmasse der COM-Schnittstelle anzuschließen.

Durch die on-chip UART des Microcontrollers werden keine Handshake-Leitungen unterstützt. Diese können bei Bedarf durch Portpins des Microcontrollers nachgebildet werden. Für die Unterstützung eines RS-232-Pegels für diese Handshake-Leitungen ist außerhalb des Moduls ein RS-232-Treiber vorzusehen.

5.2 RS-485 Schnittstelle

Alternativ zur Pegelanpassung nach RS-232 für die Leitungen P3.0/RxD0 und P3.1/TxD0 ist auf dem phyCORE-ADuC812 eine Pegelanpassung nach RS-485 vorgesehen. Mit Hilfe der Jumper J3 und J4 kann die Auswahl zwischen RS-232 und RS-485 vorgenommen werden (*siehe Kapitel 3.3*).

Der RS-485-Treiber (U10) unterstützt bis zu 32 Knoten in einer Busstruktur. Die Datenübertragung erfolgt mit einem differentiellen Pegel nach RS-485.

Achtung!

Um die RS-485 Schnittstelle zu nutzen, muß Jumper J12 bestückt werden. Je nach Konfiguration muß der RS-485 Treiber über T1 oder OUT0 eingeschaltet werden (*siehe Kapitel 3.3*).

5.3 CAN-Schnittstelle

Das phyCORE-ADuC812 kann optional mit einem CAN-Controller vom Typ Philips SJA1000 bestückt werden. Der CAN-Controller ist in einen 128 Byte großen Speicherbereich ansprechbar (*siehe Kapitel 4, „Speichermodule“ für nähere Informationen*). Über den Jumper J5 wird der CAN-Interrupt mit dem /INT0 vom Controller ADuC812 verbunden. Wird der CAN Controller im Polling Modus betrieben, kann der Jumper J5 geöffnet werden. Der Interrupteingang /INT0 des Controllers steht dann für externe Aufgaben zur Verfügung.

Für die Pegelanpassung der CAN-Sende/Empfangsleitungen (CANTx/CANRx) ist auf dem phyCORE-ADuC812 ein CAN-Bustreiber (U8, 82C251) vorgesehen. Wenn der on-board CAN-Treiber verwendet werden soll, muß J8 auf dem phyCORE-ADuC812 geschlossen sein (default).

Der interne CAN-Bustreiber unterstützt bis zu 110 Knoten an einem CAN-Bus. Die Datenübertragung erfolgt mit einem differentiellen Pegel auf CANH und CANL. Eine Masseverbindung zwischen den Knoten am CAN-Bus ist nicht zwingend erforderlich, wird jedoch für eine Verbesserung der EMV-Eigenschaften empfohlen. Für die korrekte Funktion der Datenübertragung auf dem CAN-Bus sind 2 Abschlußwiderstände, je ein 120 Ohm-Widerstand am Ende des CAN-Bus¹, vorzusehen.

Bei größeren Busausdehnungen wird weiterhin eine externe galvanische Entkopplung zwischen CAN-Bustreiber und dem phyCORE-ADuC812 empfohlen. In diesem Fall ist die Leitungen CANRx mit Hilfe von Jumper J8 vom on-board Bustreiber zu trennen. Für die galvanisch getrennte Verbindung zum externen Bustreibers sollten schnelle Optokoppler verwendet werden. Hierfür eignen sich die Typen TLP113 der Fa. Toshiba oder HCPL06xx der Fa. Hewlett Packard. Für die korrekte Busanschaltung sind die Empfehlungen DS102 vom CiA¹ zu berücksichtigen.

¹: CiA CAN in Automation - Internationale Nutzer und Herstellervereinigung, gegründet im März 1992. CiA bietet technische, Produkt- und Marketinginformationen zum Thema Controller Area Network an, mit dem Ziel, den Bekanntheitsgrad dieser Technologie zu erhöhen, und zukünftige Weiterentwicklungen des CAN Protokolls zu unterstützen.

6 Flash Speicher

Durch den Einsatz von Flash Speichern als nichtflüchtiger Codespeicher können Sie die Vorteile der modernen Flash-Technik nutzen.

6.1 On-Board Flash Speicher (U4)

Als on-board Flash-Baustein für das phyCORE-ADuC812 steht entweder ein 29F010 mit zwei Bänken à 64 kByte oder ein 29F040 mit acht Bänken à 64 kByte zur Verfügung.

Die Verwendung von Flash Speicher erlaubt die Realisierung einer on-board Programmierung des Moduls. Die Flash Speicher sind mit 5 VDC programmierbar, wodurch keine besondere Programmierspannung benötigt wird. In Bank 0 des Flash ist bereits ein Software-Werkzeug (FlashTools) integriert, welches diese on-board Reprogrammierung des Flashs ermöglicht. Dadurch ist die maximale Größe des nutzbaren Speichers 64 kByte (29F010) bzw. 448 kByte (29F040) (siehe Bild 16).

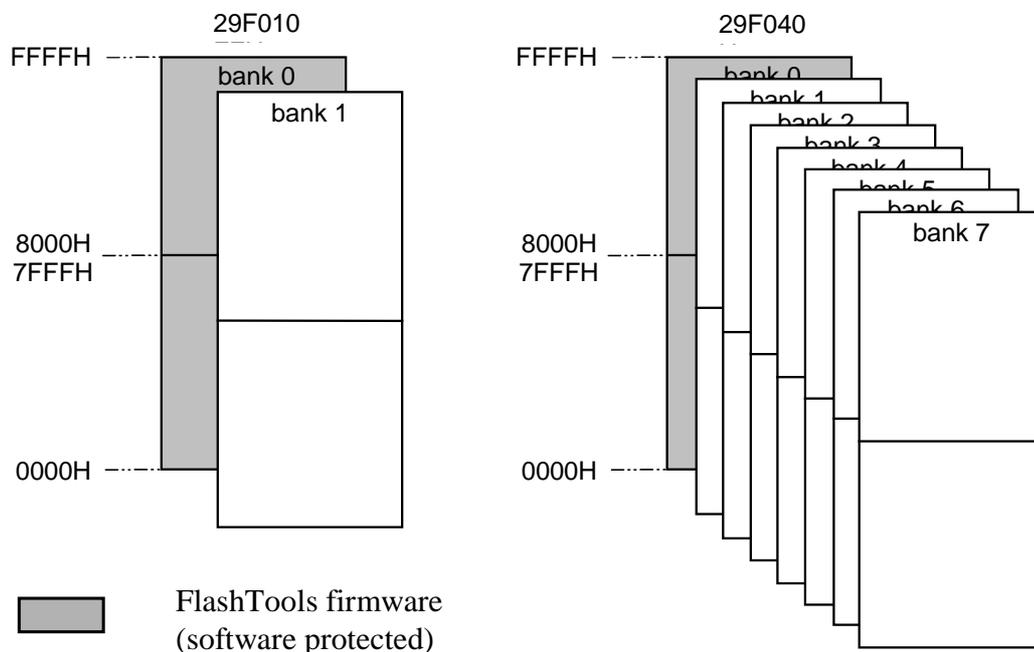


Bild 16: Speicherbereiche des Flash

Sollte die FlashTools Firmware ohne gleichwertigen Ersatz aus dem Flash gelöscht werden, so ist eine Reprogrammierung nicht mehr möglich!

Beachten Sie bitte, daß sich die Software selbst gegen ein absichtliches oder versehentliches Löschen bzw. Überprogrammieren schützt. 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 nichtflüchtigen 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 Flash-Programmierung 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. 100.000 Lösch-/Programmierzyklen auf.

6.2 On-Chip Flash Speicher

Der 8 kByte on-chip Flash Speicher der ADuC-812/824 Controllers ist ISP- (In-System Programming) programmierbar. ISP bedeutet, der Microcontroller kann innerhalb der Schaltung neu programmiert werden. Ein Entfernen aus der Schaltung für die Programmierung mit einem zusätzlichen Programmiergerät ist nicht erforderlich. Der Microcontroller enthält on-chip Standardfunktionen zum Löschen und Programmieren des Flashs. Ein zusätzlicher Loader für die ISP-Anwendung ist nicht erforderlich.

Zur einfachen Programmierung bietet Analog Devices auf ihrer Homepage unter <http://www.analog.com> das Tool "WSD" zum kostenlosen Download an. Die Tool-Software WSD versucht nach dem Start, die Verbindung zum on-chip Bootloader aufzunehmen. Der on-chip Bootloader wird analog zu den FlashTools gestartet. Die Programmierung des on-chip Flash setzt allerdings Jumper J6 = 2+3 (mit 1 kOhm Widerstand) voraus (*siehe Kapitel 3.5*).

Achtung!

Der on-chip Bootloader kann nur gestartet werden, wenn der Jumper J6 auf Position 2+3 mit einem 1 kOhm Widerstand bestückt ist.

Der on-chip Flash Speicher ist mit 5 VDC programmierbar, wodurch keine besondere Programmierspannung benötigt wird.

Nach Stand der Technik zur Drucklegung dieses Manuals sind im Minimum 10.000 Lösch/Programmierzyklen für jedes Byte möglich.

7 Serielles EEPROM (U11)

Für das Sichern von Einstellungen und Betriebsdaten kann auf dem Modul ein EEPROM auf U11 bestückt werden. Dieses kann über den on-board I²C-Bus angesprochen werden. Der Bus wird über die Controller-Pins SCLOCK (SCL) und SDATA/MOSI (SDA) realisiert. Durch Öffnen der beiden Lötjumper J13 und J14 kann die RTC und das EEPROM von den Controller-Pins getrennt werden, welche somit als freie Pins für eine externe SPI-Schnittstelle zur Verfügung stehen.

Optional wird U11 mit einem EEPROM bestückt, dessen Anzahl von Schreib-/Löschzyklen den meisten Anforderungen genügt (ca. 10⁶ Schreib-/Löschzyklen). Sollten jedoch in einer Applikation viele Daten sehr häufig und schnell gespeichert werden, so kann U11 auch mit anderen Speicherbausteinen bestückt werden. So können z.B. I²C-FRAMs (ca. 10¹⁰ Schreib-/Lesezyklen) eingesetzt werden, welche als ferroelektrische Speicher ebenfalls ihre Daten bei Stromausfall behalten.

Adressierung:

Die Adresspins A0 (IC-Pin 1) und A1 (2) liegen an GND. Pin A2 (3) ist mit VCC verbunden. Damit ergibt sich folgende Adressierung für die Speicher:

Speicher	Größe	Typ	Adresse
EEPROM	4 kByte	Catalyst 24WC32	1010100 *
EEPROM	8 kByte	Catalyst 24WC64	1010100
FRAM	512 Byte	Ramtron FM24C04	101010x
FRAM	8 kByte	Ramtron FM24C64	1010100

*= *Vorzugstyp*

Tabelle 22: Bestückungsmöglichkeiten U11 und Adressierung

8 Echtzeituhr RTC-8563 (U12)

Für Echtzeitanwendungen ist das phyCORE-ADuC812 mit einer Echtzeituhr des Typs RTC-8563 ausgestattet. Diese Echtzeituhr bietet die folgenden Funktionen:

- Serielle Kommunikation über I²C Bus
- Stromaufnahme
 - Bus aktiv: max. 50 mA
 - Bus inaktiv, CLKOUT = 32 kHz: max. 1,7 µA
 - Bus inaktiv, CLKOUT = 0 kHz: max. 0,75 µA
- Uhrfunktion mit 4-Jahres Kalender
- Jahrhundertbit
- Universeller Timer mit Alarm- und Überlaufanzeige
- 24-Stunden Format
- Automatische Word-Adressen Inkrementierung
- Programmierbare Alarm-, Timer- und Interruptfunktion

Wird das phyCORE-ADuC812 über eine Batterie gepuffert, läuft die Echtzeituhr auch nach Ausfall der Spannungsversorgung des Boards.

Die Programmierung der Echtzeituhr findet über den I²C-Bus (Adresse 1010001), der mit Hilfe der Ports SCLOCK (SCL) und SDATA/MOSI (SDA) realisiert ist, statt. Zusätzlich verfügt die Echtzeituhr über einen Interruptausgang, der über Jumper J7, auf den Anschluss /INT1 (Port P3.3) des Controllers führt. Ereignisse für einen Interrupt können ein Uhrenalarm, ein Timeralarm, ein Timerüberlauf bzw. ein Alarm des Ereigniszählers sein. Alle Interrupts müssen durch Software gelöscht werden. Durch die Interruptfunktion ist die Echtzeituhr für die unterschiedlichsten Applikationen verwendbar. *Weitere Informationen über die Register der Echtzeituhr finden Sie im beigelegten Datenblatt.*

Achtung!

Nach Anlegen der Versorgungsspannung, bzw. nach einem Reset generiert die Echtzeituhr **keinen** Interrupt, da sie hierfür erst initialisiert werden muß.

9 RESET-Controller (U6)

Der RESET-Controller U6 dient dem definierten Auslösen eines RESET-Signales bei Unterschreitung der Spannung an VCC ($< 4,65$ V). Damit wird ein sauberes Starten des Microcontrollers sichergestellt. Weiterhin kann er bei Spannungsausfall eine Backup-Batterie an einzelne IC's durchschalten (als VPD-Spannung). *Die grundlegenden Eigenschaften entnehmen Sie bitte dem beigefügten Datenblatt.*

Die Pins des RESET-Controllers wurden alle nach außen an die phyCORE-Leiste geführt. VPD liegt am OUT-Pin des ICs. Es wird im Normalfall von VCC (über eine Diode) gespeist. VBAT liegt zusätzlich über einem Spannungsteiler R22/R23 an PFI. Wenn $VBAT = 3,3$ V, dann liegen hier $1,65$ V an. Unterschreitet diese Spannung an PFI $1,25$ V wird das Signal /PFO ausgelöst. WDI und /PFO liegen an den Pins X1D5 und X1F5 des phyCORE-Connectors an.

10 Remote Supervisor Chip (U7)

U7 ist vorgesehen für einen Remote Supervisor Chip. Mit diesem IC kann über ein serielles Interface (RS-232 oder RS-485) ein Bootvorgang ausgelöst werden. Damit wird die Bedingung für den Start der PHYTEC FlashTools geschaffen. Dies ermöglicht, ohne die Betätigung eines BOOT-Jumpers oder -Tasters, ferngesteuert ein Update der Software im Flash einzuleiten.

Da dieser IC noch in Entwicklung ist, kann das Feature erst bei zukünftigen Modulen angeboten werden.

11 Batteriepufferung

Die zur Batteriepufferung nötige Batterie ist für die Grundfunktion des phyCORE-ADuC812 nicht zwingend erforderlich. Allerdings bietet sich die Batteriepufferung als eine günstige und einfache Möglichkeit des nichtflüchtigen Abspeicherns von Daten an und sie ist notwendig für den Datenerhalt der Real-Time Clock.

Der VBAT-Eingang am Modulpin X1D4 ist für den Anschluß einer externen Batterie vorgesehen. Der Minuspol der Batterie ist mit der Schaltungsmasse GND des phyCORE-ADuC812 zu verbinden. Wir empfehlen, nach dem Stand der Technik zur Drucklegung dieses Manuals, Lithium-Batterien, da diese hohe Kapazitäten bei sehr geringer Selbstentladung aufweisen. Der RAM-Baustein wird bei fehlender Versorgungsspannung VCC von einer eventuell vorhandenen Batterie über VBAT gespeist.

Die Stromaufnahme hängt von den verwendeten Bausteinen bzw. dem Speicherausbau ab (*siehe Kapitel 13, "Technische Daten"*).

Hinweis:

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.

12 A/D-Wandler und D/A-Wandler

Auf dem phyCORE-ADuC812 können die Controller ADuC-812 und ADuC-824 (ggf ADuC816) bestückt werden. Die besondere Eigenschaft dieser Microcontroller sind die integrierten A/D-Wandler und D/A-Wandler. Alle analogen Signale werden auf den phyCORE-Leisten G und H des Moduls herausgeführt. Es ist zu beachten, daß der Analogteil des Controllers ausschließlich über diese Leisten mit der Spannungsversorgung verbunden wird. Die analoge Masse (AGND = 0V) sowie die analoge Versorgungsspannung (AVCC = 5V) auf dem Modul sind nicht mit der digitalen Masse (GND = 0V) und der digitalen Versorgungsspannung (VCC = 5V) verbunden.

Achtung!

Wenn keine externe analoge Stromversorgung (AVCC = 5V, AGND = 0V) angeschlossen ist, dann kann dies zur Zerstörung des phyCORE-Moduls führen. Es ist außerdem zu beachten, daß es zwischen GND und AGND keinen Potentialunterschied geben sollte (*siehe Data Sheet zum jeweiligen Controller*).

Die Anschlüsse für analoge Signale am Controller sind direkt mit den entsprechenden Pins der phyCORE-Leiste G und H verbunden. Es befinden sich keine Operationsverstärker o.ä. auf dem phyCORE-Modul. Bitte beachten Sie deshalb die Designvorgaben für die analogen Eingänge bzw. Ausgänge vom Controllerhersteller Analog Devices und ergänzen Sie die entsprechenden Eingangs- bzw. Ausgangsschaltungen in Ihrer Applikation gemäß der speziellen Anforderungen.

Es wird empfohlen, die analogen Eingänge in Ihrer Applikationsschaltung neu zu kalibrieren. Von Analog Devices wird dafür ein Softwaretool bereitgestellt (*siehe Analog Devices "TechNotes uC005"*). Bei den analogen Ausgängen ist es von Vorteil, wenn Sie in Ihrer Applikationsschaltung eine Schaltung für den Nullpunktgleich usw. vorsehen. Damit können Sie auch den D/A-Wandler nach Ihren Anforderungen kalibrieren.

weitere Daten:

- Modulgröße: 55 mm x 60 mm
- Gewicht: ca 25 g bei Maximalausbau mit gesockeltem Controller
- Lagertemperatur: -40°C bis +90°C
- Betriebstemperatur: 0°C bis +70°C, erweitert -40°C bis +85°C
- Luftfeuchtigkeit: max. 95 % r.F. nicht kondensierend
- Betriebsspannung: 5 V \pm 5 %
- VBAT: 3 V \pm 10 %
- Stromaufnahme: max. 220 mA, typ. 110 mA bei 11.0592 MHz Oszillatorfrequenz und 128 kByte RAM bei 20°C
- Stromaufnahme bei Batteriepufferung: max. 100 μ A, typisch 1 μ A pro RAM-Baustein und 1 μ A Real-Time Clock bei 20°C
- Verzögerungszeit beim Zugriff auf externe Peripherie (Adresse \rightarrow /CS1-3): 10 ns

Diese Daten beziehen sich auf die Standardkonfiguration des phyCORE-ADuC812 bei Drucklegung.

Beachten Sie bitte, daß die Lagertemperatur bei der Verwendung der Batteriepufferung für die RAMs nur 0°C bis +70°C beträgt.

14 Hinweise zum Umgang mit dem Modul

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.

15 Das phyCORE-ADuC812 auf dem Development Board phyCORE LD 5V

Um die schnelle und problemlose Inbetriebnahme unserer Microcontroller-Module unter üblichen Labor-Bedingungen zu vereinfachen, bieten wir als Ergänzung ein passendes Development Board an. Dieses stellt standardisierte Funktionen und Anschlüsse für die Spannungsversorgung und die einfache Kommunikation mit dem Modul zur Verfügung.

15.1 Konzept des Development Board phyCORE-LD 5V

Das Development Board phyCORE-LD 5V ist Kernstück einer flexiblen Inbetriebnahme- und Entwicklungsplattform, die einfach und rasch an die Erfordernisse spezieller Applikationen oder neuer Microcontroller-Module angepaßt werden kann. Als Ergänzung bieten wir zum Aufbau dieser Plattformen Erweiterungsplatinen mit unterschiedlichen Funktionen an.

Dieser "Mehr-Platinen-Ansatz" ist in *Bild 18* dargestellt:

- Das eigentliche **Development Board** (1) stellt, als das Kernstück, die absolut rudimentären Funktionen und Anschlüsse zur Verfügung. Dazu zählen die Anschlüsse für ein **externes Netzteil** (2) sowie für die **seriellen Schnittstellen** (3) des Microcontroller-Moduls (je nach Modul bis zu zwei RS-232 und bis zu zwei RS-485 oder CAN-Schnittstellen).
- Auf dem Development Board werden alle Signale des aufgesteckten Moduls über eine starre 1:1 Zuordnung auf zwei Platinen-Steckverbinder geführt. Durch diese starre Zuordnung hängt die konkrete Belegung des **Expansion-Bus** (4) ausschließlich von der Belegung des eingesetzten Microcontroller-Moduls ab.

- Da die mechanische Ausführung des Erweiterungsbusses bei allen entsprechenden Development Boards unseres Hauses standardisiert ist, können wir eine Auswahl von allgemein passenden **Erweiterungsplatinen** (5) anbieten. Diese unterstützen, in unterschiedlichem Umfang, **zusätzliche I/O-Funktionen** (6) zur gezielten Demonstration von bestimmten Controller-Features des eingesetzten **Microcontroller-Moduls** (9).
- Ein **Patchfeld** (7) auf jeder Erweiterungsplatine erlaubt die flexible Verbindung zwischen den jeweils angebotenen Funktionsgruppen und den entsprechenden Signalen des verwendeten Microcontroller-Moduls auf dem Development Board. Hierzu sind wiederum alle Signale des Erweiterungsbusses in einer starren 1:1 Zuordnung auf Stiftleisten geführt. Durch Aufstecken der beige-fügigen **Patchkabel** (8) auf die korrespondierenden Pins der Stiftleisten werden die erforderlichen Verbindungen hergestellt.

Folgende Skizze verdeutlicht die Zusammenhänge:

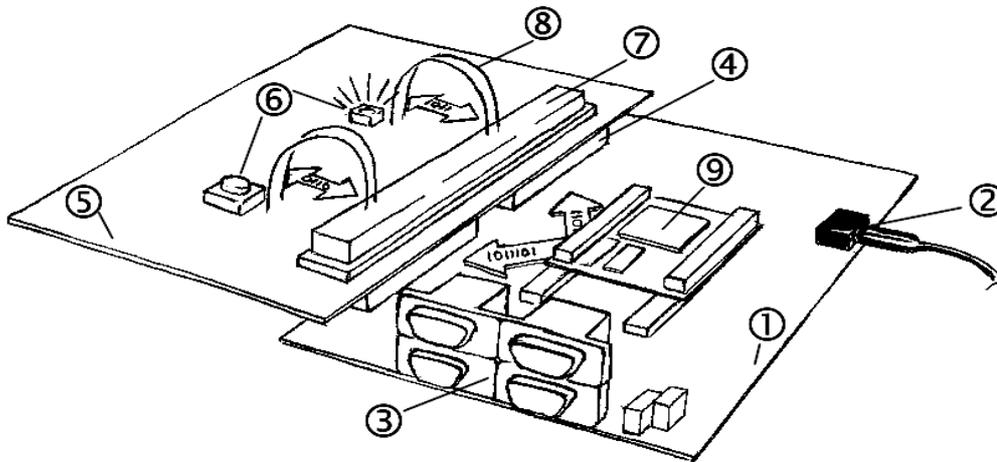


Bild 18: Das Mehrplatinenkonzept mit phyCORE-ADuC812, Development Board und Erweiterungsplatine

Die folgenden Kapitel enthalten spezifische Informationen, die für den Betrieb des phyCORE-ADuC812 auf dem Development Board phyCORE-LD 5V relevant sind. Für eine allgemeine Beschreibung des Development Board phyCORE-LD 5V greifen Sie bitte auf das entsprechende Hardware-Manual zurück.

15.2 Anschlüsse und Jumper des Development Board phyCORE-LD 5V

15.2.1 Anschlüsse

Wie in *Bild 19* dargestellt, stehen folgende Anschlüsse zur Verfügung:

- X1- Kleinspannungsbuchse zum Anschluß der Versorgungsspannung
- X2- Expansion-Connector zum Anschluß einer Erweiterungsplatine
- P1- DB-9 Buchsen zum Anschluß serieller Schnittstellen nach RS-232 Standard
- P2- DB-9 Stecker zum Anschluß von Feldbussen nach CAN und RS-485 Standard
- X4- Spannungsabgriff für die Versorgung externer Baugruppen
- X5- GND-Anschluß für Meßzwecke
- X6- phyCORE-Connector zur Aufnahme des phyCORE-Moduls
- BAT1- Anschluß für eine optionale Pufferbatterie

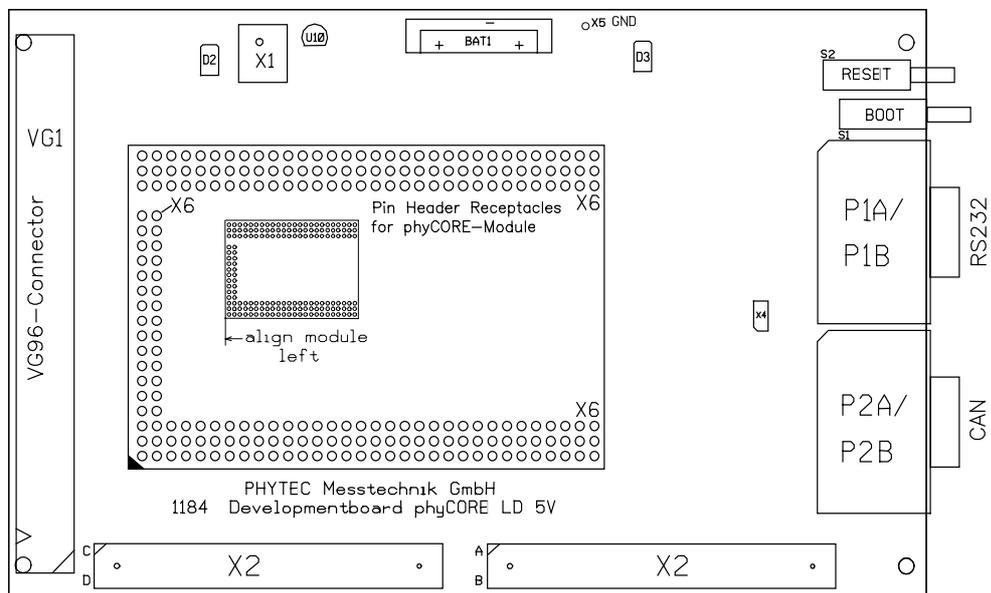


Bild 19: Lage der Anschlüsse auf dem Development Board phyCORE-LD 5V

Es sei ausdrücklich darauf hingewiesen, daß bei allen Modulanschlüssen unbedingt die Maximalspannungen und -ströme nicht überschritten werden dürfen. Die Grenzwerte hierfür können Sie dem jeweiligen Microcontroller-Handbuch und den entsprechenden Datenblättern der eingesetzten Schaltkreise auf dem Development Board 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.

15.2.2 Jumper des Development Board phyCORE-LD 5V

Mit Hilfe von Jumpern werden die peripheren Komponenten des Development Board phyCORE-LD 5V mit den Signalanschlüssen des phyCORE-ADuC812 verbunden.

Ohne Jumperbelegung sind alle Signale des Moduls von den DB-9 Verbindern und den CAN-Treibern getrennt. Der RESET-Eingang des phyCORE-C591 ist direkt mit dem RESET-Taster (S2) verbunden. *Bild 20* verdeutlicht die verwendete Zählweise bei den Jumpern, *Bild 21* die Lage der Jumper auf dem Development Board.

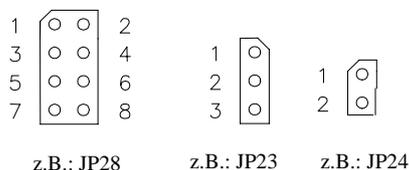


Bild 20: Zählweise der Jumper

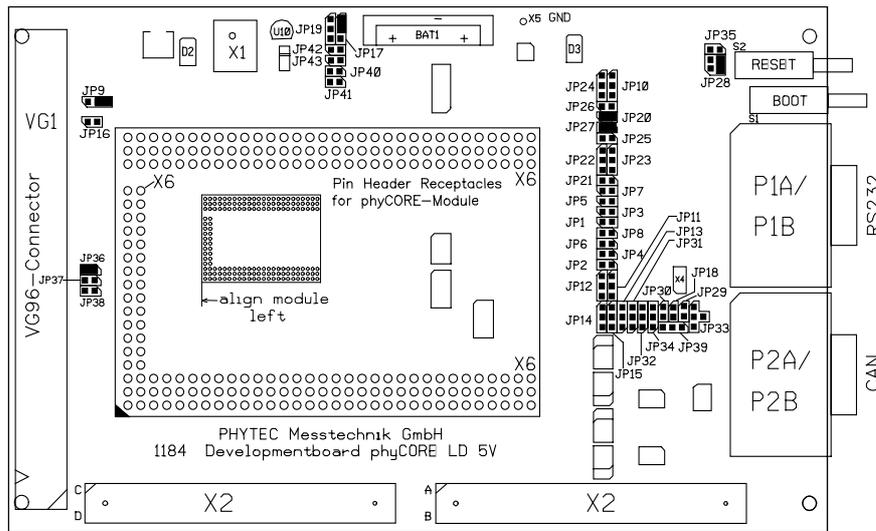


Bild 21: Lage der Jumper (Ansicht Bestückungsseite)

Bild 22 zeigt die Default-Einstellung für die Belegungen der Jumper auf dem Development Board phyCORE-LD 5V.

Die Default-Einstellungen konfigurieren das Development Board phyCORE-LD 5V nur für Funktionen des Standard phyCORE-ADuC812 (Standard = ADuC812 Controller; mit z.B. Nutzung der ersten RS-232 Schnittstelle, der LED D3 und des BOOT-Tasters auf dem Development Board phyCORE-LD 5V). Weiterführende Funktionen können gemäß der Tabellen ab Kapitel 15.3 konfiguriert werden.

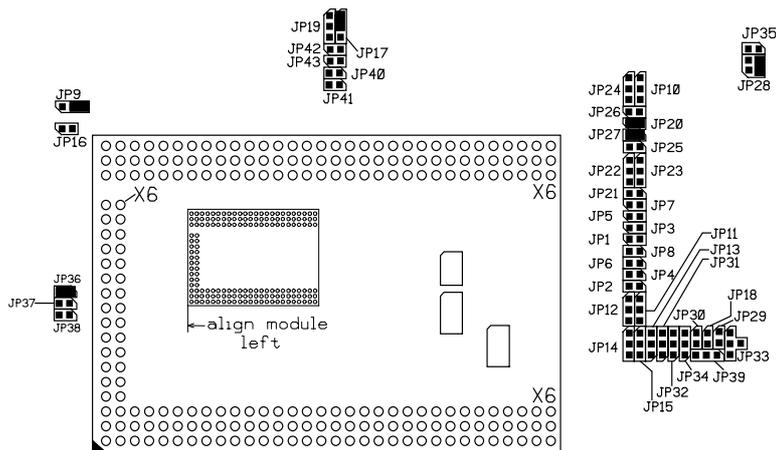


Bild 22: Default Jumper-Stellungen des Development Board phyCORE-LD 5V mit phyCORE-ADuC812

15.2.3 Nicht unterstützte Features und unzulässige Jumper-Stellungen

Die folgende Tabelle weist Jumper-Stellungen aus, die aufgrund von speziellen Produkteigenschaften des phyCORE-ADuC812 auf dem Development Board phyCORE-LD 5V unzulässig sind.

Spannungsversorgung:

Das Development Board phyCORE-LD 5V unterstützt zwei Hauptspannungsversorgungen für den Betrieb verschiedener phyCORE-Module. Beim Einsatz des phyCORE-ADuC812 wird nur eine Hauptspannungsversorgung VCC1 mit 5 V benötigt. Die Anschlüsse für eine zweite Spannungsversorgung VCC2 sind am phyCORE-ADuC812 nicht definiert bzw. kontaktiert und dürfen deshalb nicht benutzt werden.

Jumper	Stellung	Wirkung
JP16	geschlossen	VCC2 an phyCORE-ADuC812

Tabelle 23: Unzulässige Jumper-Stellung auf dem Development Board

15.3 Funktionsgruppen des Development Board phyCORE-LD 5V

Dieser Abschnitt beschreibt im Detail die vom phyCORE-ADuC812 unterstützen Funktionsgruppen des Development Board phyCORE-LD 5V sowie entsprechende Jumper-Stellung. In Abhängigkeit von bestimmten Bestückungsoptionen des eingesetzten phyCORE-ADuC812 können Konfigurationen eingestellt werden, die unter Umständen verschieden von dem in *Kapitel 15.2.3* aufgezeigten Default Jumper-Stellungen sind. Durch die Änderung der Default-Einstellungen auf eine abweichende Konfiguration können alternative oder zusätzliche Funktionen des Development Board phyCORE-LD 5V aktiviert werden.

15.3.1 Spannungsversorgung an X1

Achtung!

Verwenden Sie keine Labornetzteile! Die Einschaltspitzen könnten das eingesetzte Modul zerstören!

Vermeiden Sie außerdem, bei anliegender Spannung das Modul bzw. die Jumperbelegung zu wechseln!

Zulässiger Spannungsbereich : +5 VDC geregelt.

Die erforderliche Strombelastbarkeit ist von den benutzten Optionen des phyCORE-ADuC812 sowie von den verwendeten Zusatzplatinen abhängig. Wir empfehlen die Verwendung von Netzteilen mit mindestens 500 mA Strombelastbarkeit.

Jumper	Stellung	Wirkung
JP9	2 + 3	5 V als Hauptversorgungsspannung des phyCORE-ADuC812
JP36	geschlossen	5 V als analoge Versorgungsspannung AVCC des phyCORE-ADuC812

Tabelle 24: JP9, JP36 Konfiguration der Versorgungsspannung VCC1 und AVCC

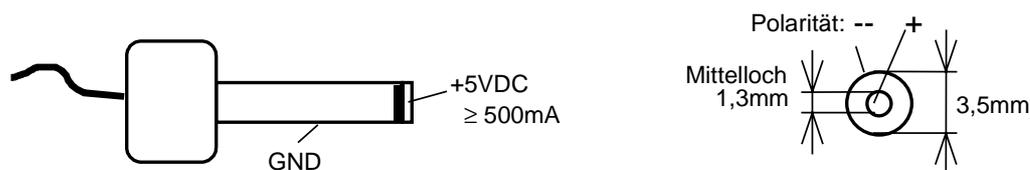


Bild 23: Anschluß der Versorgungsspannung an X1

Achtung!

Für diese Funktion sind die folgenden Jumper-Stellungen unzulässig:

Jumper	Stellung	Wirkung
JP9	1 + 2	3,3 V als Hauptversorgungsspannung des phyCORE-ADuC812
	offen	phyCORE-ADuC812 wird nicht mit einer Hauptversorgungsspannung versorgt
JP36	offen	phyCORE-ADuC812 wird nicht mit einer analogen Versorgungsspannung versorgt

Tabelle 25: JP9, JP36 Unzulässige Jumper-Stellungen der Hauptspannungsversorgung

Bei Konfiguration von Jumper JP9 auf Position 1+2 wird eine Hauptversorgungsspannung von 3,3 V eingestellt, was zur Zerstörung des phyCORE-ADuC812 führen kann. Bei geöffnetem Jumper JP9 liegt keine Versorgungsspannung am phyCORE-ADuC812 an. Diese Jumper-Stellung ist deshalb ebenfalls unzulässig.

Bei geöffnetem Jumper JP36 liegt keine analoge Versorgungsspannung am phyCORE-ADuC812 an. Diese Jumper-Stellung ist deshalb unzulässig.

15.3.2 Starten der FlashTools

Das phyCORE-ADuC812 verfügt über einen externen Flash Speicher, der die FlashTools Firmware enthält. Mit Hilfe dieser Firmware und der auf einem PC installierten korrespondierenden Software ist die on-board Programmierung des Flash Speichers mit Anwenderprogrammen über eine RS-232 Verbindung möglich.

Achtung!

Die FlashTools lassen sich nur starten, wenn der Jumper J6 des phyCORE-ADuC812 in der Stellung **1+2** ist! (siehe Kapitel 3.5)

Zum Starten FlashTools auf dem phyCORE-ADuC812 muss am BOOT-Pin (X1D6) des phyCORE-Moduls zum Zeitpunkt des Wechsels des RESET Signals vom aktiven in den inaktiven Zustand ein High-Pegel anliegen.

Unter Verwendung der auf dem Development Board phyCORE-LD 5V befindlichen Funktionseinheiten kann dies auf drei verschiedenen Wegen erreicht werden:

1. Mit Jumper JP28 kann der BOOT-Taster (S1) mit VCC verbunden werden. Damit wird bei Drücken des BOOT-Tasters während des Reset-Vorgangs oder dem Einschalten der Versorgungsspannung die FlashTools Firmware gestartet.

Jumper	Stellung	Wirkung
JP28	3 + 4	BOOT-Taster (in Verbindung mit RESET oder Einschalten der Versorgungsspannung) startet FlashTools Firmware auf dem phyCORE-ADuC812

Tabelle 26: JP28 Konfiguration des BOOT-Tasters

2. Der BOOT-Eingang des phyCORE-ADuC812 kann auch fest mit VCC verbunden werden. Dies erspart das zusätzliche Drücken des BOOT-Tasters während des Auslösens von RESET oder dem Einschalten der Versorgungsspannung.

Jumper	Stellung	Wirkung
JP28	2 + 4	BOOT-Eingang permanent mit VCC verbunden, FlashTools werden bei RESET oder Einschalten der Versorgungsspannung= immer gestartet

Tabelle 27: JP28 Konfiguration einer ständigen FlashTools Startbedingung

Achtung!

Bei dieser Konfiguration ist keine Ausführung eines normalen RESET und damit der Start Ihrer Applikation möglich. Es werden immer die FlashTools gestartet.

3. Die FlashTools können auch durch externe Signale, die an der DB-9 Buchse P1A anliegen, gestartet werden. Dazu muß der Zustandswechsel des RESET-Signals über Pin 7 gesteuert werden, während an Pin 4 ein statischer High-Pegel für das BOOT-Signal anliegt.

Jumper	Stellung	Wirkung
JP22	1 + 2	Pin 7 der DB-9 Buchse P1A als RESET-Signal für das phyCORE-ADuC812
JP23	1 + 2	Pin 4 der DB-9 Buchse P1A als BOOT-Signal für das phyCORE-ADuC812
JP10	2 + 3	High-aktives BOOT-Signal mit BOOT-Eingang des phyCORE-ADuC812 verbunden

Tabelle 28: JP22, JP23, JP10 Konfiguration von BOOT über RS-232

Achtung!

Für diese Funktion ist die folgende Jumper-Stellung unzulässig:

Jumper	Stellung	Wirkung
JP10	1 + 2	Jumper-Einstellung erzeugt Low-Pegel am BOOT-Eingang des phyCORE-ADuC812

Tabelle 29: Unzulässige Jumper-Stellung beim BOOT über RS-232

15.3.3 Erste serielle Schnittstelle an Buchse P1A

Der Anschluß P1A ist die untere Buchse der Doppelbuchse P1. P1A ist über Jumper mit der RS-232 Schnittstelle des phyCORE-ADuC812 verbunden. In Verbindung mit einem Host-System kann das phyCORE-ADuC812 über die Buchse P1A in den FlashTools-Modus gebracht werden (siehe Kapitel 15.3.2).

Jumper	Stellung	Wirkung
JP20	geschlossen ¹	Pin 2 mit TxD0 der RS-232 Schnittstelle des phyCORE-ADuC812 verbunden
	offen	Pin 2 der DB-9 Buchse P1A nicht belegt
JP21	offen	Pin 9 der DB-9 Buchse P1A nicht belegt
JP22	offen	Pin 7 der DB-9 Buchse P1A nicht belegt
	1 + 2	RESET-Eingang des Moduls kann durch Signalpegel auf der RTS Leitung vom Host gesteuert werden
JP23	offen	Pin 4 der DB-9 Buchse P1A nicht belegt
	1 + 2	BOOT-Eingang des Moduls kann durch Signalpegel auf der DTR Leitung vom Host gesteuert werden
JP24	offen	Pin 6 der DB-9 Buchse P1A nicht belegt
JP25	offen	Pin 8 der DB-9 Buchse P1A nicht belegt
JP26	offen	Pin 1 der DB-9 Buchse P1A nicht belegt
JP27	geschlossen ¹	Pin 3 mit RxD0 der RS-232 Schnittstelle des phyCORE-ADuC812 verbunden
	offen	Pin 3 der DB-9 Buchse P1A nicht belegt

¹ = erforderlich zur Kommunikation mit FlashTools

Tabelle 30: Jumper-Konfiguration für die erste RS-232 Schnittstelle

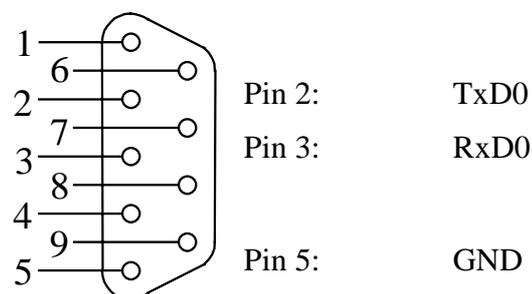


Bild 24: Belegung der DB-9 Buchse P1A als RS-232 (Ansicht Vorderseite)

15.3.4 Buchse P1B

Der Anschluß P1B ist die obere Buchse der Doppelbuchse P1. Das phyCORE-ADuC812 unterstützt keine zweite RS-232 Schnittstelle, die Buchse P1B bleibt deshalb ungenutzt.

Jumper	Stellung	Wirkung
JP1	offen	Pin 2 der DB-9 Buchse P1B nicht belegt
JP2	offen	Pin 9 der DB-9 Buchse P1B nicht belegt
JP3	offen	Pin 7 der DB-9 Buchse P1B nicht belegt
JP4	offen	Pin 4 der DB-9 Buchse P1B nicht belegt
JP5	offen	Pin 6 der DB-9 Buchse P1B nicht belegt
JP6	offen	Pin 8 der DB-9 Buchse P1B nicht belegt
JP7	offen	Pin 1 der DB-9 Buchse P1B nicht belegt
JP8	offen	Pin 3 der DB-9 Buchse P1B nicht belegt
JP40	offen	Pin 2 der DB-9 Buchse P1B nicht belegt
JP41	offen	Pin 3 der DB-9 Buchse P1B nicht belegt

Tabelle 31: Jumper-Konfiguration der DB-9 Buchse P1B

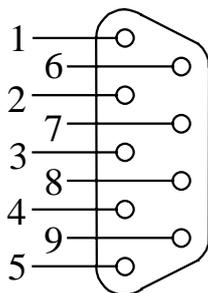


Bild 25: Belegung der DB-9-Buchse P1B (Ansicht Vorderseite)

Achtung!

Beim Betrieb des Development Board phyCORE-LD 5V mit einem phyCORE-ADuC812 sind die folgenden Jumper-Stellungen unzulässig:

Jumper	Stellung	Wirkung
JP1	geschlossen	Pin 2 der DB-9 Buchse P1B ist mit B (RS-485) des phyCORE-ADuC812 verbunden
JP3	geschlossen	Pin 7 der DB-9 Buchse P1B ist mit SCL des phyCORE-ADuC812 verbunden
JP4	geschlossen	Pin 4 der DB-9 Buchse P1B ist mit OUT1 des phyCORE-ADuC812 verbunden
JP5	geschlossen	Pin 6 der DB-9 Buchse P1B ist mit OUT6 des phyCORE-ADuC812 verbunden
JP6	geschlossen	Pin 8 der DB-9 Buchse P1B ist mit SDA des phyCORE-ADuC812 verbunden
JP7	geschlossen	Pin 1 der DB-9 Buchse P1B ist mit OUT5 des phyCORE-ADuC812 verbunden
JP8	geschlossen	Pin 3 der DB-9 Buchse P1B ist mit A (RS-485) des phyCORE-ADuC812 verbunden
JP40	geschlossen	Pin 2 der DB-9 Buchse P1B ist mit Pin IN6 des phyCORE-ADuC812 verbunden
JP41	geschlossen	Pin 3 der DB-9 Buchse P1B ist mit Pin IN7 des phyCORE-ADuC812 verbunden

Tabelle 32: Unzulässige Jumper-Stellungen bei der Konfiguration von P1B

Wenn versehentlich ein RS-232 Kabel an P1B angeschlossen ist, dann kann der Spannungspegel auf den RS-232 Leitungen zur Zerstörung des phyCORE-ADuC812 führen.

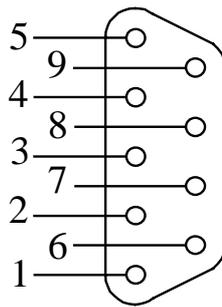
15.3.5 CAN-Schnittstelle an Stecker P2A

Der Anschluß P2A ist der untere Stecker des Doppelsteckers P2. Zu diesem Stecker werden die Signale der CAN-Schnittstelle des phyCORE-ADuC812 geführt. Aufgrund von verschiedenen Konfigurationen bezüglich der CAN-Treiber und deren Spannungsversorgung ergeben sich die drei nachfolgend dargestellten Möglichkeiten:

1. Der CAN-Treiber auf dem phyCORE-ADuC812 wird verwendet und dessen Signale direkt an den Stecker P2A geführt:

Jumper	Stellung	Wirkung
JP31	2 + 3	Pin 2 des DB-9 Steckers P2A ist mit CAN-L vom on-board Treiber des phyCORE-ADuC812 verbunden
JP32	2 + 3	Pin 7 des DB-9 Steckers P2A ist mit CAN-H vom on-board Treiber des phyCORE-ADuC812 verbunden
JP11	offen	Eingang am Optokoppler U4 auf dem Development Board phyCORE-LD 5V offen
JP12	offen	Ausgang am Optokoppler U5 auf dem Development Board phyCORE-LD 5V offen
JP13	offen	CAN-Treiber und Optokoppler auf dem Development Board phyCORE-LD 5V spannungsfrei
JP18	offen	CAN-Treiber und Optokoppler auf dem Development Board phyCORE-LD 5V potentialfrei
JP29	offen	kein Spannungsabgriff über CAN-Bus
JP42	offen	Eingang am Optokoppler U4 auf dem Development Board phyCORE-LD 5V offen
JP43	offen	Ausgang am Optokoppler U5 auf dem Development Board phyCORE-LD 5V offen

Tabelle 33: Jumper-Konfiguration des CAN-Steckers P2A mit CAN-Treiber auf phyCORE-ADuC812



- Pin 3: GND (Schaltungsmasse des Dev. Boards)
- Pin 7: CAN-H0 (keine galvanische Entkopplung)
- Pin 2: CAN-L0 (keine galvanische Entkopplung)
- Pin 6: GND (Schaltungsmasse des Dev. Boards)

Bild 26: Belegung des DB-9 Steckers P2A (CAN-Treiber auf phyCORE-ADuC812)

Achtung!

In diesem Anwendungsfall sind die folgenden Jumper-Stellungen unzulässig:

Jumper	Stellung	Wirkung
JP31	1 + 2	Pin 2 des DB-9 Steckers P2A ist mit CAN-L0 vom CAN-Treiber des Development Board verbunden
JP32	1 + 2	Pin 7 des DB-9 Steckers P2A ist mit CAN-H0 vom CAN-Treiber des Development Board verbunden
JP11	1 + 2	Eingang am Optokoppler U4 auf dem Development Board mit CAN-L0 des phyCORE Moduls verbunden
JP11	2 + 3	CANTxD des phyCORE-ADuC812 ist über Optokoppler U4 mit CAN-Treiber U2 verbunden
JP12	1 + 2	Ausgang am Optokoppler U5 auf dem Development Board mit T0 des phyCORE-ADuC812 verbunden
JP12	2 + 3	CANRxD vom phyCORE-ADuC812 ist über Optokoppler U5 mit CAN-Treiber U2 verbunden
JP13	1 + 2	CAN Treiber und Optokoppler auf dem Development Board phyCORE-LD 5V werden von einer externen Spannung über einen separaten Regler versorgt
JP13	2 + 3	CAN Treiber und Optokoppler auf dem Development Board phyCORE-LD 5V werden lokal versorgt
JP18	geschlossen	CAN Treiber und Optokoppler auf dem Development Board mit lokalen GND Potential verbunden
JP29	geschlossen	Spannungszuführung für separaten Regler über Pin 9 des DB-9 Steckers P2A
JP42	geschlossen	Eingang am Optokoppler U4 auf dem Dev. Board mit T1 (P3.5) des phyCORE-ADuC812 verbunden
JP43	geschlossen	Ausgang am Optokoppler U5 auf dem Dev. Board mit T0 (P3.4) des phyCORE-ADuC812 verbunden

Tabelle 34: Unzulässige Jumper-Stellungen CAN-Stecker P2A (CAN-Treiber auf phyCORE-ADuC812)

2. Der CAN-Treiber des phyCORE-ADuC812 ist deaktiviert und der CAN-Treiber U2 des Development Board phyCORE-LD 5V wird **ohne galvanische Trennung** verwendet.

Jumper	Stellung	Wirkung
JP31	1 + 2	Pin 2 des DB-9 Steckers P2A ist mit CAN-L des Treibers U2 auf dem Development Board verbunden
JP32	1 + 2	Pin 7 des DB-9 Steckers P2A ist mit CAN-H des Treibers U2 auf dem Development Board verbunden
JP11	2 + 3	CANTxD vom phyCORE-ADuC812 ist über Optokoppler U4 mit CAN-Treiber U2 verbunden
JP12	2 + 3	CANRxD vom phyCORE-ADuC812 ist über Optokoppler U5 mit CAN-Treiber U2 verbunden
JP13	2 + 3	CAN-Treiber und Optokoppler auf dem Development Board phyCORE-LD 5V werden lokal versorgt
JP18	geschlossen	CAN-Treiber und Optokoppler auf dem Development Board mit lokalen GND Potential verbunden
JP29	offen	kein Spannungsabgriff über CAN-Bus
JP42	offen	Eingang am Optokoppler U4 auf dem Development Board nicht mit T1 (P3.5) des phyCORE-ADuC812 verbunden
JP43	offen	Ausgang am Optokoppler U5 auf dem Development Board nicht mit T0 (P3.4) des phyCORE-ADuC812 verbunden

Tabelle 35: Jumper-Konfiguration des DB-9 Steckers P2A mit CAN Treiber auf Development Board

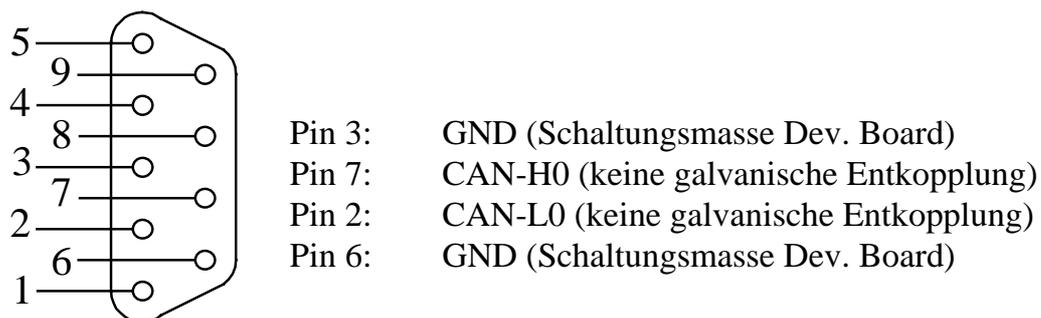


Bild 27: Belegung des DB-9 Steckers P2A (CAN Treiber auf Development Board)

Achtung!

In diesem Anwendungsfall sind die folgenden Jumper-Stellungen unzulässig:

Jumper	Stellung	Wirkung
JP31	2 + 3	Pin 2 des DB-9 Steckers P2A ist mit CANL vom on-board Treiber des phyCORE-ADuC812 verbunden
JP32	2 + 3	Pin 7 des DB-9 Steckers P2A ist mit CANH vom on-board Treiber des phyCORE-ADuC812 verbunden
JP11	1 + 2	Eingang am Optokoppler U4 auf dem Development Board mit CANL des phyCORE Moduls verbunden
JP11	offen	CANTxD vom phyCORE-ADuC812 ist nicht über Optokoppler U4 mit CAN-Treiber U2 verbunden
JP12	1 + 2	Ausgang am Optokoppler U5 auf dem Development Board mit CANH des phyCORE Moduls verbunden
JP12	offen	CANrxD vom phyCORE-ADuC812 ist nicht über Optokoppler U5 mit CAN-Treiber U2 verbunden
JP13	1 + 2	CAN Treiber und Optokoppler auf dem Development Board phyCORE-LD 5V werden von einer externen Spannung über einen separaten Regler versorgt
JP29	geschlossen	Spannungszuführung für separaten Regler über Pin 9 des DB-9 Steckers P2A

Tabelle 36: Unzulässige Jumper-Stellungen CAN-Stecker P2A (CAN-Treiber auf Development Board)

3. Der CAN-Treiber des phyCORE-ADuC812 ist deaktiviert und der CAN-Treiber U2 des Development Board phyCORE-LD 5V wird **mit galvanischer Trennung** verwendet. In diesem Fall ist die externe Zuführung einer CAN-Versorgungsspannung im Bereich von 7 – 13 V, 14 – 20 V oder 21 – 27 V notwendig. Bitte beachten Sie, dass Sie die externe Spannung nur über einen der beiden Stecker P2A **oder** P2B zu führen.

Jumper	Stellung	Wirkung
JP31	1 + 2	Pin 2 des DB-9 Steckers P2A ist mit CAN-L0 des Treibers U2 auf dem Development Board verbunden
JP32	1 + 2	Pin 7 des DB-9 Steckers P2A ist mit CAN-H0 des Treibers U2 auf dem Development Board verbunden
JP11	2 + 3	CANTxD vom phyCORE-ADuC812 ist über Optokoppler U4 mit CAN-Treiber U2 verbunden
JP12	2 + 3	CANRxD vom phyCORE-ADuC812 ist über Optokoppler U5 mit CAN-Treiber U2 verbunden
JP13	1 + 2	CAN-Treiber und Optokoppler auf dem Development Board phyCORE-LD 5V werden von einer externen Spannung über einen separaten Regler versorgt
JP18	offen	CAN-Treiber und Optokoppler sind vom GND Potential des Development Board getrennt
JP29	geschlossen	Spannungszuführung für separaten Regler über Pin 9 des DB-9 Steckers P2A
JP39	1 + 2	externe CAN-Versorgung mit 7 – 13 V
	2 + 3	externe CAN-Versorgung mit 14 – 20 V
	offen	externe CAN-Versorgung mit 21 – 27 V
JP42	offen	Eingang am Optokoppler U4 auf dem Development Board nicht mit T1 (P3.5) des phyCORE-ADuC812 verbunden
JP43	offen	Ausgang am Optokoppler U5 auf dem Development Board nicht mit T0 (P3.4) des phyCORE-ADuC812 verbunden

Tabelle 37: Jumper-Konfiguration des DB-9 Steckers P2A mit CAN Treiber auf Development Board und galvanischer Trennung

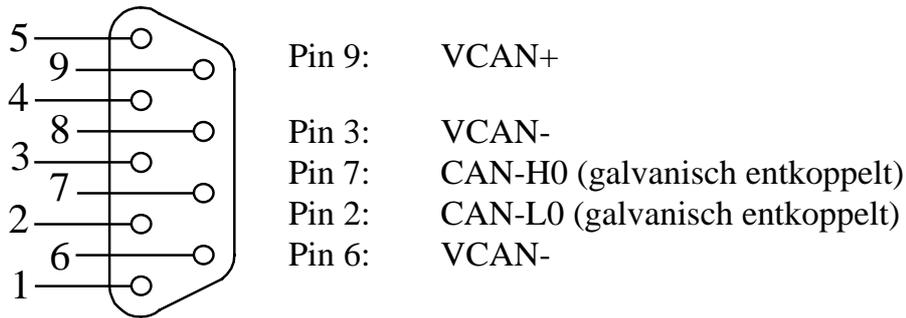


Bild 28: Belegung des DB-9 Steckers P2A (CAN-Treiber auf Development Board und galvanische Trennung)

Achtung!

In diesem Anwendungsfall sind die folgenden Jumper-Stellungen unzulässig:

Jumper	Stellung	Wirkung
JP31	2 + 3	Pin 2 des DB-9 Steckers P2A ist mit CANL vom on-board Treiber des phyCORE-ADuC812 verbunden
JP32	2 + 3	Pin 7 des DB-9 Steckers P2A ist mit CANH vom on-board Treiber des phyCORE-ADuC812 verbunden
JP11	1 + 2	Eingang am Optokoppler U4 auf dem Development Board mit CAN-L0 des phyCORE Moduls verbunden
JP11	offen	CANTxD vom phyCORE-ADuC812 ist nicht über Optokoppler U4 mit CAN-Treiber U2 verbunden
JP12	1 + 2	Ausgang am Optokoppler U5 auf dem Development Board mit CANH des phyCORE Moduls verbunden
JP12	offen	CANRxD vom phyCORE-ADuC812 ist nicht über Optokoppler U5 mit CAN-Treiber U2 verbunden
JP13	2 + 3	CAN Treiber und Optokoppler auf dem Development Board phyCORE-LD 5V werden lokal versorgt
JP18	geschlossen	CAN Treiber und Optokoppler auf dem Development Board mit lokalen GND Potential verbunden

Tabelle 38: Unzulässige Jumper-Stellungen bei galvanisch getrenntem CAN-Bus (CAN-Treiber auf Development Board)

15.3.6 RS-485 Schnittstelle an Stecker P2B

Der Anschluß P2B ist der obere Stecker des Doppelsteckers P2. Zu diesem Stecker werden die Signale der RS-485 Schnittstelle des phyCORE-ADuC812 geführt. Die RS-485 Schnittstelle ist eine alternative Funktion der seriellen Schnittstelle des ADuC812 Controllers. In der Default-Einstellung des phyCORE-ADuC812 ist diese Schnittstelle als RS-232 konfiguriert. Zur Aktivierung einer RS-485 Schnittstelle sind veränderte Jumper-Einstellungen auf dem phyCORE-ADuC812 notwendig (*siehe auch Kapitel 3.3*).

Jumper	Stellung	Wirkung
JP33	1 + 2	Pin 2 des DB-9 Steckers P2B ist mit dem RS-485 A Signal des phyCORE-ADuC812 verbunden
JP34	offen	Keine Verbindung von Pin 7 des DB-9 Steckers P2B zu Signalen auf dem Development Board
JP14	offen	Keine Verbindung zwischen CAN-Optokoppler U6 und Signalen des phyCORE-Moduls
JP15	offen	Keine Verbindung zwischen CAN-Optokoppler U7 und Signalen des phyCORE-Moduls
JP13	offen	CAN Treiber und Optokoppler auf dem Development Board spannungsfrei
JP18	geschlossen	Pins 3 und 6 des DB-9 Steckers P2B sind mit dem GND Potential des Development Board verbunden
JP29	offen	Versorgung über Pin 9 des DB-9 Steckers P2A oder P2B deaktiviert
JP30	geschlossen	Pin 8 des DB-9 Steckers P2B ist mit dem RS-485 B Signal des phyCORE-ADuC812 verbunden

Tabelle 39: Jumper-Konfiguration des DB-9 Steckers P1B als RS-485

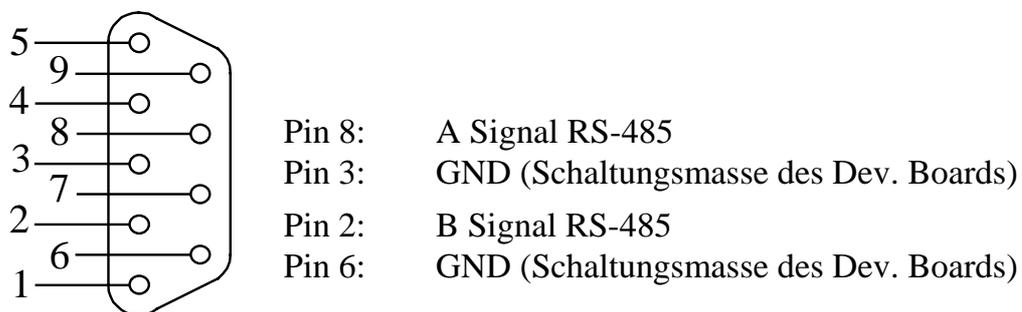


Bild 29: Belegung des DB-9-Steckers P2B (Ansicht Vorderseite, RS-485 Mode)

Achtung!

In diesem Anwendungsfall sind die folgenden Jumper-Stellungen unzulässig:

Jumper	Stellung	Wirkung
JP33	2 + 3	Pin 2 des DB-9 Steckers P2B ist mit CAN-L1 Signal des U3 vom Development Board verbunden
	2 + 4	Pin 2 des DB-9 Steckers P2B ist mit CANTxD des phyCORE-ADuC812 verbunden
JP34	1 + 2	Pin 7 des DB-9 Steckers P2B ist mit CAN-H1 Signal des U3 vom Development Board verbunden
	2 + 3	Pin 7 des DB-9 Steckers P2B ist mit CANRxD des phyCORE-ADuC812 verbunden
JP14	1 + 2	CAN-Optokoppler U6 ist mit CAN-L des phyCORE-ADuC812 verbunden
	2 + 3	CAN-Optokoppler U6 ist mit CANTxD des phyCORE-ADuC812 verbunden
JP15	1 + 2	CAN-Optokoppler U7 ist mit CAN-H des phyCORE-ADuC812 verbunden
	2 + 3	CAN-Optokoppler U7 ist mit CANRxD des phyCORE-ADuC812 verbunden
JP13	1 + 2	CAN-Treiber und Optokoppler auf dem Development Board phyCORE-LD 5V werden von einer externen Spannung über einen separaten Regler versorgt
JP13	2 + 3	CAN-Treiber und Optokoppler auf dem Development Board phyCORE-LD 5V werden lokal versorgt
JP18	offen	Pins 3 und 6 des DB-9 Steckers P2B sind nicht mit dem GND Potential des Development Board verbunden
JP29	geschlossen	Spannungszuführung für separaten Regler über Pin 9 des DB-9 Steckers P2A oder P2B

Tabelle 40: Unzulässige Jumper-Stellung P2B als RS-485 Schnittstelle

15.3.7 Visualisierungs-LED D3

Das Development Board phyCORE-LD 5V ist mit einer LED D3 ausgestattet, die als einfaches Anzeige-Element verwendet werden kann. Diese LED kann mit dem Portpin auf GPIO0 (JP17 = 1+2) oder über ein Latch U14 am Datenbus (JP17 = 2+3) angesteuert werden. Beim Einsatz des phyCORE-ADuC812 wird als Voreinstellung das Portpin P3.4 (GPIO0) für die Ansteuerung der LED verwendet. Die Steuerung der LED D3 kann auch mit dem Datenbit D0 auf Adresse FDA0h erfolgen. Beim Anliegen eines Low-Pegels am Latch U14 leuchtet die LED D3, bei einem High-Pegel am Latch U14 bleibt die LED D3 erloschen.

Jumper	Stellung	Wirkung
JP17	1 + 2	LED D3 ist mit Portpin P3.4 (GPIO0) des ADuC812 verbunden
JP17	2 + 3	LED D3 wird über Latch U14 auf dem Development Board durch das Datenbit D0 vom ADuC812 gesteuert

Tabelle 41: JP17 Konfiguration der Visualisierungs-LED D3

15.3.8 Belegungen des phyCORE, des Expansion-Bus und des Patchfeldes im Überblick

Wie bereits in *Kapitel 15.1* erläutert, werden alle Signale des phyCORE-ADuC812 mittels einer starren 1:1 Zuordnung auf den Expansion-Connector X2 geführt. Dieser wird wiederum anhand einer weiteren, ebenfalls starren 1:1 Zuordnung mit dem Patchfeld einer optional angeschlossenen Erweiterungsplatine verbunden.

Bitte beachten Sie hierbei, daß je nach Ausführung und Größe der Erweiterungsplatine unter Umständen nur ein Teil des kompletten Patchfeldes realisiert wird. Dieser stellt eine Untermenge der hier aufgeführten Signale dar. Die verwendeten Nummerierungen behalten dabei trotzdem ihre Gültigkeit.

Analog zur Zählweise des phyCORE-Connectors wird auch beim Expansion-Connector und dem Patchfeld eine zweidimensionale Zählmatrix verwendet. Deren Ausrichtung weicht allerdings aus Gründen des Layouts von der des phyCORE-Connectors ab.

Bitte entnehmen Sie den beiden folgenden Bildern die Ausrichtung der Zählmatrix für den Expansion-Connector sowie des Patchfeldes:

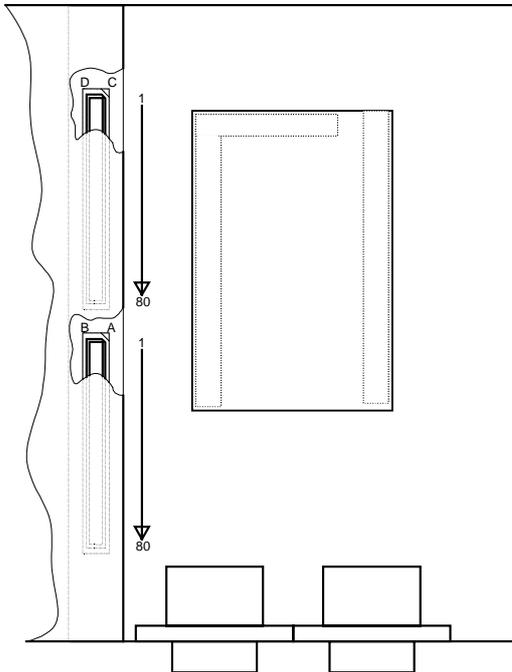


Bild 30: Zählweise für den Expansion-Bus

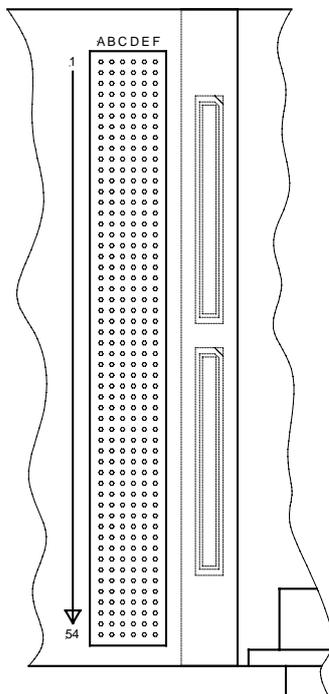


Bild 31: Zählweise des Patchfeldes

Für das phyCORE-ADuC812 auf einem Development Board phyCORE-LD 5V und einer angeschlossenen Erweiterungsplatine ergeben sich folgende Pinbelegungen:

Signal	phyCORE-ADuC812	Expansion-Bus	Patchfeld
P0.0/ AD0	12C	18B	33F
P0.1/ AD1	13A	19A	34A
P0.2/ AD2	13C	20A	34E
P0.3/ AD3	14A	20B	34B
P0.4/ AD4	14B	21A	34D
P0.5/ AD5	14C	21B	34F
P0.6/ AD6	15A	22B	35A
P0.7/ AD7	15C	23A	35E
A0	6A	8B	30B
A1	6B	9A	30D
A2	6C	10A	30F
A3	7A	10B	31A
A4	7C	11A	31E
A5	8A	11B	31B
A6	8C	12B	31F
A7	9A	13A	32A
P2.0/A16A8	9B	13B	32C
P2.1/A17A9	9C	14A	32E
P2.2/ A18A10	10A	15A	32B
P2.3/ A19A11	10C	15B	32F
P2.4/ A20A12	11A	16A	33A
P2.5/ A21A13	11B	16B	33C
P2.6/ A22A14	11C	17B	33E
P2.7/ A23A15	12A	18A	33B
A16	16A	23B	35B
A17	16B	24A	35D
A18	16C	25A	35F
A19	17A	25B	36A
A20	17C	26A	36E
A21	18A	26B	36B
A22	18C	27B	36F
A23	19A	28A	37A

*Tabelle 42: Daten/Adressbus-Pinzuordnung phyCORE-ADuC812 /
Development Board / Erweiterungsplatine*

Signal	phyCORE-ADuC812	Expansion-Bus	Patchfeld
ClkIn	1A	1A	28A
ClkOut	1B	1B	28C
P3.2 / INT0	1C	2B	28E
P3.3/ /INT1	2A	3A	28B
P3.4/ T0	2C	3B	28F
P3.5/ T1	3A	4A	29A
/CS1	3C	5A	29E
/CS2	4A	5B	29B
/CS3	4C	6B	29F
ALE	4B	6A	29D
/RD	5A	7B	30A
/WR	5C	8A	30E
/EA	19B	28B	37C

Tabelle 43: Steuersignal-Pinzuordnung phyCORE-ADuC812 /
Development Board / Erweiterungsplatine

Signal	phyCORE-ADuC812	Expansion-Bus	Patchfeld
BOOT	6D	9C	3B
/RESET	6E	10C	3D
/RESIN	6F	10D	3F
/RESOUT	7F	11C	4E
T0 (P3.4)	7D	11D	4A
T1 (P3.5)	8D	12D	4B
RxD	11D	16D	6A
TxD	11E	17D	6C
RSTxD	14F	22D	7F
RSRxD	15F	23D	8E
CANRxD	13D	20C	7A
CANTxD	12D	18D	6B
CANL	14D	21C	7B
CANH	15D	23C	8A
A	14E	21D	7D
B	13F	20D	7E
SCL	16D	24C	8B
SDA	16E	25C	8D

Tabelle 44: Schnittstellen-Pinzuordnung phyCORE-ADuC812 /
Development Board / Erweiterungsplatine

Signal	phyCORE-ADuC812	Expansion-Bus	Patchfeld
IN0	8F	13C	4F
IN1	9D	13D	5A
IN2	9E	14C	5C
IN3	9F	15C	5E
IN4	10D	15D	5B
IN5	10F	16C	5F
IN6	11F	18C	6E
IN7	12F	19C	6F
OUT0	16F	25D	8F
OUT1	17D	26C	9A
OUT2	17F	26D	9E
OUT3	18D	27D	9B
OUT4	18F	28C	9F
OUT5	19D	28D	10A
OUT6	19E	29C	10C
OUT7	19F	30C	10E

Tabelle 45: Pinzuordnung Input- und Outputport phyCORE-ADuC812 / Development Board / Erweiterungsplatine

Signal	phyCORE-ADuC812	Expansion-Bus	Patchfeld
VREF	4H, 7G	52D, 55D	17F, 18D
CREF	7H	55D	18F
AVCC	8G	56C	19A
AGND	5G, 10G, 3H, 8H, 12H	mit GND verbunden	mit GND verbunden
DAC0	3G	51C	17B
DAC1	12G	60D	20B
ADC0	4G	51D	17D
ADC1	5H	53C	18A
ADC2	6G	53D	18E
ADC3	6H	54C	18B
ADC4	9G	56D	19E
ADC5	9H	57D	19B
ADC6	10H	58C	19F
ADC7	11G	58D	20A

Tabelle 46: Pinzuordnung Analogleisten phyCORE-ADuC812 / Development Board / Erweiterungsplatine

Signal	phyCORE-ADuC824	Expansion-Bus	Patchfeld
VREF	4H, 7G	52D, 55D	17F, 18D
VREF-	7H	55D	18F
AVCC	8G	56C	19A
AGND	5G, 10G, 3H, 8H, 12H	mit GND verbunden	mit GND verbunden
ADC1	3G	51C	17B
DAC	12G	60D	20B
T2	4G	51D	17D
T2EXT	5H	53C	18A
IEXC1	6G	53D	18E
IEXC2	6H	54C	18B
ADC3	9G	56D	19E
ADC4	9H	57D	19B
/SS	10H	58C	19F
/MISO	11G	58D	20A

Tabelle 47: Pinzuordnung Analogleisten phyCORE-AduC-824 /
Development Board / Erweiterungsplatine

Signal	phyCORE-ADuC812	Expansion-Bus	Patchfeld
NC	2D, 3D, 2E, 3E, 19C, 11H	59A, 4C, 5C, 59C, 4D, 5D,	2A, 1B, 2C, 20C; 1D, 20C,
Pin des Development Board, die nicht vom phyCORE- ADuC812 genutzt werden	20A bis 32A 20B bis 32B 20C bis 32C 20D bis 32C 20E bis 32E 20F bis 32F		

Tabelle 48: Nicht verwendete Pins des phyCORE-ADuC812 beim Development
Board und der Erweiterungsplatine

Signal	phyCORE-ADuC812	Expansion-Bus	Patchfeld
PFI	4F	7D	2F
PFO	5F	8C	3E
VCC	1D, 2D	1C, 2C, 1D, 2D	1A, 1C
VPD	4E	6D	2D
VBAT	4D	6C	2B
GND	2b, 3B, 5B, 7B, 8B, 10B, 12B, 13B, 15B, 17B, 18B, 5E, 7E, 8E, 10E, 12E, 13E, 15E, 17E, 18E, 1F, 2F, 3F	2A, 7A, 12A, 17A, 22A, 27A, 32A, 37A, 42A 47A, 52A, 57A, 62A, 67A, 72A, 77A, 4B, 9B, 14B, 19B, 24B, 29B, 34B, 39B, 44B, 49B, 54B, 59B, 64B, 69B, 74B, 79B, 3C, 7C, 12C, 17C, 22C, 27C, 32C, 37C, 42C 47C, 52C, 57C, 62C, 67C, 72C, 77C, 3D, 9D, 14D, 19D, 24D, 29D, 34D, 39D, 44D, 49D, 54D, 59D, 64D, 69D, 74D, 79D	3C, 4C, 7C, 8C, 9C, 12C, 13C, 14C, 17C, 18C, 19C, 22C, 23C, 24C, 27C, 29C, 30C, 31C, 34C, 35C, 36C, 39C, 40C, 41C, 44C, 45C, 46C, 49C, 50C, 51C, 54C, 4D, 5D, 6D, 9D, 10D; 11D, 14D, 15D, 16D, 19D, 20D, 21D, 24D, 25D, 26D, 28D, 31D, 32D 33D, 36D, 37D, 38D, 41D, 42D, 43D, 46D, 47D, 48D, 51D, 52D, 53D

Tabelle 49: Pinzuordnung Versorgungsspannung phyCORE-ADuC812 / Development Board / Erweiterungsplatine

15.3.9 Batterieanschluß BAT1

Der Anschluß BAT1 dient zur Bestückung einer Batterie, die die flüchtigen Speicher (SRAM) und die RTC auf dem phyCORE-ADuC812 während der Zeit, in der keine Versorgungsspannung anliegt, puffert. Die Umschaltung von der normalen Spannungsversorgung auf Batteriepufferung wird vom Reset Controller des phyCORE-ADuC812 automatisch durchgeführt. Die für diesen Anwendungsfall erforderliche Batterie (*siehe auch Kapitel 11*) kann bei der Firma PHYTEC Meßtechnik GmbH bezogen werden.

15.3.10 Nummernchip

Für verschiedene Software-Applikationen im Netzwerkbereich wird für die Definition einer Knotenadresse oder als Kopierschutz ein Nummernchip eingesetzt. Der Nummernchip DS2401 kann je nach Bauform auf Position U10 oder U9 aufgelötet werden.

Der Nummernchip auf dem Development Board phyCORE-LD 5V kann über das Portpin an P1.1 (JP19 1+2) oder mit den Datenbus über das Latch U14 und den Treiber U15 (JP19 = 2+3) angesprochen werden. Beim phyCORE-ADuC812 ist die Ansteuerung sowohl über den Port P3.5 als auch über den Datenbus möglich.

In der Default-Einstellung erfolgt der Zugriff auf den Nummernchip über Port P3.5. Wenn der Datenbus genutzt werden soll, ist der Nummernchip auf Adresse FDA0h mit dem Datenbit D1 über das Latch U14 und den Treiber U15 anzusprechen.

Jumper	Stellung	Wirkung
JP19	1 + 2	Nummernchip ist mit Portpin P3.5 des ADuC812 verbunden
JP19	2 + 3	Nummernchip ist mit Datenleitung D1 über U14 / U15 verbunden

Tabelle 50: JP19 Konfiguration des Nummernchip

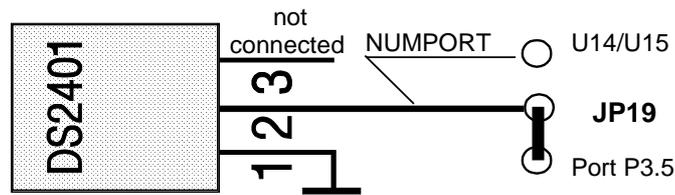


Bild 32: Anschluß des Nummernchips

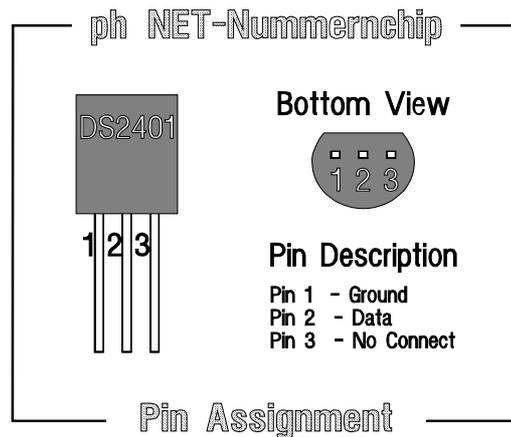


Bild 33: Pinbelegung Nummernchip

15.3.11 Stiflleiste X4

An der Stiflleiste X4 ist die Spannungsversorgung des externen Steckernetzteil abgreifbar. Der Anschluß X4 ist zur Versorgung eines extern angeschlossenen Modems mit einer Spannung 5 V von ausgelegt. An Pin 1 des Steckers ist die Spannung 5 V = verfügbar, am Pin 2 ist das Bezugspotential GND der Development Board phyCORE-LD 5V angeschlossen. Die maximale Strombelastbarkeit ist abhängig vom verwendeten Netzteil. Es wird empfohlen, nur Geräte mit weniger als 250 mA Stromaufnahme einzusetzen.

Index

/	
/INT0	26
/INT1	28
A	
A/D-Wandler	61
Abmessungen	64
Adressdecoder	23, 30
Adressregister	44
Anschlußbelegung	12
Anschlüsse	12
Anschlüsse und Jumper des Development Board	69
B	
Batterieanschluß BAT1	97
Batteriepufferung	60
Buchse P1A	77
Buchse P1B	78
C	
CAN Controller Interrupt	26
CAN-Bustreiber	51
CAN-Controller	28
CANH	28
CANL	28
CANRx	28
CAN-Schnittstelle	28, 51, 80
CAN-Transceiver	28
CANTx	28
Control Register 1	38
D	
D/A-Wandler	61
Default-Speichermodell	30
E	
Echtzeituhr	56
EMV	1
Erste serielle Schnittstelle	77
ESD	1
Expansion-Bus	90
F	
FA[18..15]	43
Features	6
Flash Speicher	52
Funktionsgruppen des Development Board phyCORE-LD 5V	73
H	
Hinweise zum Umgang	66
I	
I ² C-Bus	29, 55
I ² C-Schnittstelle	29
Input Register 1	48
IO-SW	40
J	
J1	23
J11	23
J12	25
J13	29
J14	29
J2	23
J3	25
J4	25
J5	26
J6	26
J7	28
J8	28
J9	23
JP17	89
JP19	97
Jumper	20

Jumperbelegung	22, 70	RS-485 Schnittstelle	50, 87
K		RS-485-Treiber	25, 50
Konzept des		RTC Interrupt-Ausgang	28
Development Board	67	S	
Kurzübersicht	4	SCLOCK	29
L		SDATA	29
LED D3	89	Serielle Schnittstelle	25
M		Seriellles	55
Maskenregister	45	SJA1000	26, 28
N		Spannungsversorgung	73
Nummernchip	97	Spannungsversorgung SRAMs ..	23
O		Speichermodelle	30
Output Register 1	49	SPI-Interface	29
P		SRAM Speichergröße	23
Patchfeld	90	Starten der FlashTools	75
PCA82C251	28	Stecker P2A	80
phyCORE-Connector	15	Stecker P2B	87
Pin-Belegungen	90	Steuerregister 1	48
Pinout	15	Stiftleiste X4	98
Port 3.2	26	T	
Port 3.3	28	Technische Daten	64
PRG-EN	38	U	
Programmspeicher, extern	26	U10	25, 50
Programmspeicher, intern	26	U11	55
R		U12	29, 56
Register des Adressdekoders	37	U13	23
Remote Supervisor Chip	59	U3	28
RESET-Controller	58	U5	23
RESET-Taster	70	U6	58
RS-232 Interface	25	U7	59
RS-232 Schnittstelle	50	U8	51
RS-232-Treiber	25, 50	U9	25, 50
		V	
		VN-EN	42

Dokument: phyCORE-ADuC812
Dokumentnummer: L-461d_2, April 2002

Wie würden Sie dieses Handbuch verbessern?

Haben Sie in diesem Handbuch Fehler entdeckt?

Seite

Eingesandt von:

Kundennummer: _____

Name: _____

Firma: _____

Adresse: _____

Einsenden an:

PHYTEC Technologie Holding AG
Postfach 100403
D-55135 Mainz, Germany
Fax : +49 (6131) 9221-33

Published by

PHYTEC

© PHYTEC Meßtechnik GmbH 2002

Ordering No. L-461d_2
Printed in Germany