

mkc

MKC Michels & Kleberhoff Computer GmbH

Vohwinkeler Str. 58, D-42329 Wuppertal

Tel.: ++49 (0)202 27317 0 Fax: ++49 (0)202 27317 49

Internet: <http://www.mkc-gmbh.de>



Technisches Handbuch

eNetMaxi

Hinweise:

Die Informationen in diesem Handbuch wurden sorgfältig zusammengestellt und überprüft. Dieses Handbuch wird stetig auf dem aktuellen Zustand gehalten. Jedoch wird von MKC keine Gewähr für fehlerhafte Informationen übernommen.

MKC behält sich das Recht vor, jederzeit ohne weitere Ankündigung technische Änderungen zur Verbesserung der Zuverlässigkeit, der Funktion oder des Designs der Produkte und Überarbeitungen des Handbuchs durchzuführen. Änderungen des Handbuchs zwischen 2 Ausgaben werden im Text nicht markiert.

Das Datum einer Ausgabe bezieht sich auf das Handbuch. Dieses muss nicht mit dem Datum der Änderung der Hardware oder Software übereinstimmen. Bei der Versionsgeschichte wird der Grund für die Handbuch Änderungen genannt.

MKC übernimmt keine Haftung für die Anwendung des hier beschriebenen Produktes. MKC übernimmt weiterhin keine Haftung für Schäden oder Folgeschäden, die durch Verwendung dieses Produktes entstehen. Diese Haftungseinschränkung bezieht sich sowohl auf jeden direkten Abnehmer sowie auf alle seine Kunden und alle Anwender des Produktes.

Es gelten ausschließlich die in diesem Dokument gemachten Zusagen über die Anwendbarkeit des hier beschriebenen Produktes.

Kommentare:

Kommentare oder Korrekturen jedweder Art sind dem Autor jederzeit willkommen. Senden Sie diese bitte an:

**MKC Michels & Kleberhoff Computer GmbH
Vohwinkeler Str. 58
42329 Wuppertal**

oder

info@mkc-gmbh.de

Inhaltsverzeichnis

1 EINLEITUNG.....	7
1.1 Hinweise zu Angaben in diesem Handbuch.....	8
2 MITGELIEFERTE SOFTWARE, HARDWARE UND ZUBEHÖR.....	9
3 TECHNISCHE DATEN.....	9
3.1 Elektrische Eigenschaften.....	10
3.1.1 Leistungsaufnahme.....	10
3.1.2 Signalpegel.....	10
3.2 Temperaturbereich.....	10
4 DEFINITION DER ANSCHLÜSSE.....	11
4.1 Power	11
4.2 Modulstecker (Belegung MKC502).....	11
4.3 Konfigurierbare Pins im Modulstecker.....	13
4.3.1 UART Schnittstellen.....	14
4.3.2 SPI Schnittstellen.....	15
4.3.3 I ² C Schnittstellen.....	16
4.3.4 CAN Schnittstelle.....	17
4.3.5 SD/SDIO Schnittstelle.....	17
4.3.6 PWM Schnittstelle.....	17
4.3.7 GPMC.....	18
4.4 Fixe Pins im Modulstecker.....	19
4.4.1 System-Control Pins.....	19
4.4.2 SD-Karten Schnittstelle.....	19
4.4.3 Batterie-Eingang.....	20
4.4.4 UART0 Schnittstelle.....	20
4.4.5 USB Schnittstellen.....	20
4.4.6 Ethernet Netzwerk.....	21
5 ARM SOC.....	23
5.1 PLL.....	23
5.2 DDR3 SDRAM.....	23
5.3 RTC.....	23
5.4 Serielles SPI FLASH.....	24
5.5 eMMC.....	24
5.6 Watchdog.....	25
5.7 Timer.....	26
6 ANHANG.....	27
6.1 JTAG/DEBUG (optional).....	27
6.2 Mechanische Abmessungen.....	27

Liste der Abbildungen

Abbildung 1: Modulansicht.....	9
Abbildung 2: Lage der Modulkontakte.....	11
Abbildung 3: SPI Interface ohne FIFO.....	15
Abbildung 4: SPI Interface mit Empfangs-FIFO.....	16
Abbildung 5: SPI Interface mit Sende-/Empfangs-FIFO.....	16
Abbildung 6: Ethernet LED2 Statusanzeigen.....	21
Abbildung 7: Ethernet Anschluss Extension-Board.....	22
Abbildung 8: Hardware-Schreibschutz SPI FLASH.....	24
Abbildung 9: JTAG/DEBUG Pads.....	27
Abbildung 10: Bemaßung.....	27

Liste der Tabellen

Tabelle 1: Elektrische Eigenschaften, Leistungsaufnahme.....	10
Tabelle 2: Elektrische Eigenschaften, Signalpegel GPIO.....	10
Tabelle 3: Modulstecker.....	12
Tabelle 4: Übersicht konfigurierbare Pins im Modulstecker.....	13
Tabelle 5: UART0 / UART1 / UART2 Konfiguration Modul-Pin zu CPU-Pin/Block.....	14
Tabelle 6: UART Baudraten und Fehler.....	14
Tabelle 7: SPI Konfiguration Modul-Pin zu CPU-Pin/Block.....	15
Tabelle 8: I2C Konfiguration Modul-Pin zu CPU-Pin/Block.....	16
Tabelle 9: CAN Konfiguration Modul-Pin zu CPU-Pin/Block.....	17
Tabelle 10: SD/SDIO Konfiguration Modul-Pin zu CPU-Pin/Block.....	17
Tabelle 11: PWM Konfiguration Modul-Pin zu CPU-Pin und PWM Generator.....	17
Tabelle 12: GPMC Konfiguration Modul-Pin zu CPU-Pin/Block.....	18
Tabelle 13: System-Control Konfiguration Modul-Pin zu CPU-Pin/Block.....	19
Tabelle 14: SPI Konfiguration Modul-Pin zu CPU-Pin/Block.....	19
Tabelle 15: VBATT Konfiguration Modul-Pin zu CPU-Pin/Block.....	20
Tabelle 16: USB Konfiguration Modul-Pin zu CPU-Pin/Block.....	20
Tabelle 17: Ethernet Konfiguration Modul-Pin zu CPU-Pin/Block.....	21
Tabelle 18: Funktion der Ethernet LED Statusanzeigen.....	21
Tabelle 19: Watchdog Register.....	25
Tabelle 20: Timer Interrupt Intervalle.....	26

1 Einleitung

Mit der Platine eNetMaxi wird das embedded Modul Konzept vom MKC weitergeführt. Der verwendete Texas Instruments Am3352 Cortex A8 ARM 32Bit Mikroprozessor arbeitet mit einer Taktfrequenz von max. 1000MHz. Weiter befinden sich Speicher in Form von FLASH-Speicher und DDR3-SDRAM auf dem eNetMaxi.

Das eNetMaxi ist als steck-/lötbares Modul ausgeführt. Es benötigt zum Betrieb eine Trägerplatine (Extension) welches dem eNetMaxi die Versorgungsspannung von 5V liefert. Alle benötigten Spannungen mit dem zugehörigen Power-Management sowie alle benötigten Taktfrequenzen werden auf dem Modul selbst generiert.

Die Funktionen des Moduls können durch das gewählte PinMux und die eingesetzte Extension variieren. Über die Steckverbinder werden der Extension diverse Schnittstellen-Signale wie Ethernet, GPMC, UART, SPI, usw. zugeführt.

Um das Modul leicht in kundenspezifische Projekte einzubinden, sind Bibliotheken für unterschiedliche CAD-Systeme, wie zum Beispiel Altium Designer, erhältlich.

Bei MKC sind Design Kits (z.B. MKC1502) für das eNetMaxi Modul verfügbar. Auf diesen sind die Schnittstellen des Moduls auf Standard-Stecker geführt. Die Spannungsversorgung kann zukunftsweisend direkt über das Netzwerk (PoE, Power over Ethernet) erfolgen. Ein zusätzliches externes Netzteil ist nicht erforderlich.

Für das Modul können verschiedene Versionen eines Linux System (Kernel, Treiber, etc.) zum Einsatz kommen. Ob eine spezifische Funktionalität genutzt werden kann bzw. unterstützt wird, hängt von der Implementierung des Systems ab.

1.1 Hinweise zu Angaben in diesem Handbuch

Zahlenangaben

Hexadezimale Zahlen werden in diesem Handbuch durch ein vorangestelltes Dollarzeichen „\$“ gekennzeichnet. Andere geläufige Schreibweisen für Hexadezimale Zahlen sind z.B. durch den Präfix „0x“ oder den Suffix „h“ in der Literatur angegeben. Sie werden hier nur der Vollständigkeit halber erwähnt.

Um die Lesbarkeit von langen hexadezimalen Zahlen zu verbessern, werden diese von rechts durch einen Punkt in 4er Gruppen unterteilt. Eine mathematische Bedeutung liegt diesem Punkt nicht zugrunde.

Logikpegel

Alle Logikpegel werden in diesem Handbuch mit „HIGH“ und „LOW“ bezeichnet. Signale die, als activ-low beschrieben werden, sind durch den Präfix „/“ gekennzeichnet.

Hardware Konfiguration

Die Lage aller Jumper und Lötbrücken der Beschreibung der Platine zu entnehmen. Die Position 1 eines Jumpers oder einer Lötbrücke ist durch eine zusätzliche Markierung hervorgehoben. Jumper bzw. Steckverbinder sind grundsätzlich mit „J“ oder mit „X“ gekennzeichnet. Alle Lötbrücken sind mit „JB“ bezeichnet. Bei der Beschreibung der einzelnen Konfigurationsmöglichkeiten geben die grau hinterlegten Felder den Auslieferungszustand der Karte wieder.

Lieferversionen

Die unten angegebenen Lieferversionen sind zur Zeit verfügbar. Damit ist nicht zugesagt, dass alle diese Versionen weiterhin lieferbar bleiben. MKC behält sich das Recht vor, die Produktion dieser Hardware oder Software aus technischen Gründen ohne vorherige Ankündigung einzustellen.

Vorläufige Angaben

In dieser Handbuchversion sind mehrere Kapitel noch vorläufig, diese Stellen sind mit dem Textzusatz '*TDB: ...*' an den entsprechenden Stellen gekennzeichnet.

2 Mitgelieferte Software, Hardware und Zubehör

Im Auslieferungszustand sind der Bootloader und die Produktionsdaten im SPI FLASH des Moduls gespeichert. Linux Firmware und Board Support Packages stellen wir auf Anfrage gerne zur Verfügung.

Kundenspezifische Änderungen (OEM) an der Homepage, Firmware, weitere Schnittstellen und Anpassungen sind prinzipiell möglich. Auch können Erweiterungen, um das Gerät als eigenständigen Controller einsetzen zu können, jederzeit implementiert werden.

3 Technische Daten

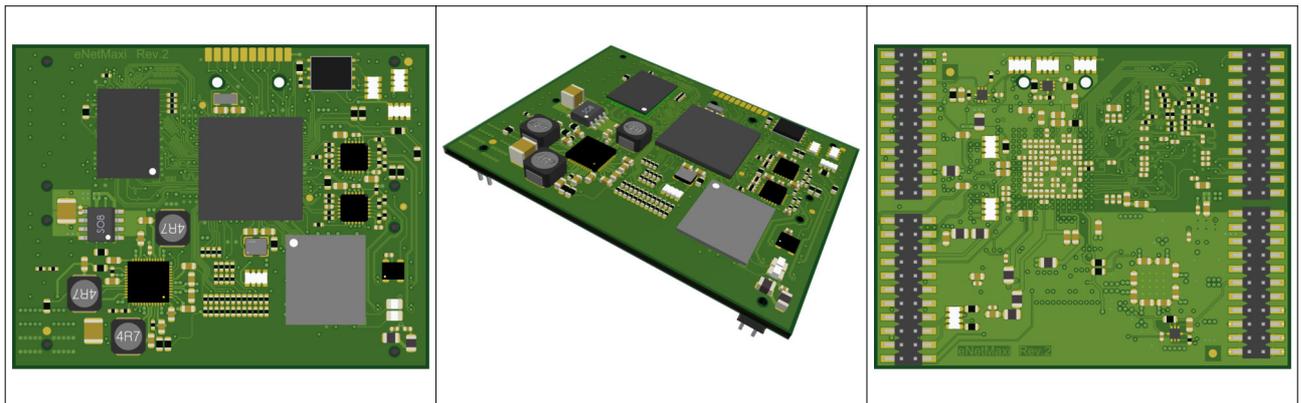


Abbildung 1: Modulansicht

- Texas Instruments Am3352B Cortex A8 ARM 32Bit CPU mit max. 1000MHz Takt
 - 32KB L1 Cache, 256KB L2 Cache
 - 512MB DDR3-SDRAM (800 MHz Datenrate, 1024MB möglich)
 - 4GB eMMC FLASH
 - 16 MB SPI FLASH (MLO, Bootloader und Produktionsdaten)
 - RTC
- SD Interface
- 2x 10/100BaseT Ethernet Schnittstellen
- 2x USB 2.0 Schnittstellen
- 3x UART oder 2x UART und 2x CAN Schnittstellen
- 1x SPI Schnittstelle
- 1x I²C Schnittstelle
- 1x Timer
- Reset-in, Reset-out
- optionales 16 Bit Daten-Interface (GPMC)
- optionales 2. SD/SDIO Interface
- JTAG Debug Interface
- Platinengröße: 47x60cm

3.1 Elektrische Eigenschaften

3.1.1 Leistungsaufnahme

		Min	Typ	Max	Unit
V_{CC}	Supply Voltage	4,75	5	5,25	V
I_{CC}	Power Supply Current ¹⁾			0,7	A

Tabelle 1: Elektrische Eigenschaften, Leistungsaufnahme

¹⁾ für das Modul mit aktivem Netzwerk, ohne Beschaltung aller IOs

3.1.2 Signalpegel

Sämtliche Signale werden unmittelbar von dem Mikroprozessor an die Stecker geführt. Der Prozessor wird modul-intern von einem 3,3V (V_{DD}) DC/DC Wandler versorgt. Die elektrischen Eigenschaften eines Pins differieren je nach programmierter Funktion. Für GPIO gelten die Daten folgender Tabelle. Für Einzelheiten zu den elektrischen Spezifikationen siehe [AM335x Sitara™ Processors datasheet](#).

Parameter	Parameter Name	Min	Nom	Max	Unit
V_{IH}	GPIO high-level input voltage	2	-	V_{DD}	V
V_{IL}	GPIO low-level input voltage	0	-	0,8	V
I_L	GPIO input leakage current	-	-	18	μ A
V_{HYS}	GPIO Input Hysteresis	0,27	-	0,44	V
V_{OH}	GPIO High-level output voltage ($I_{OH} = 6mA$)	$V_{DD} - 0,45$	-	-	V
V_{OL}	GPIO Low-level output voltage ($I_{OL} = 6mA$)	-	-	0,45	V

Tabelle 2: Elektrische Eigenschaften, Signalpegel GPIO

3.2 Temperaturbereich

- Arbeitstemperatur: 0 bis 50° Celsius Umgebungstemperatur
- Lagertemperatur: -40 bis 80° Celsius Umgebungstemperatur
- relative Feuchte: 0 ... 90%, nicht kondensierend

4 Definition der Anschlüsse

Bei dem embedded Modul eNetMaxi können die meisten der GPIO (General Purpose In/Out) des Am3352B Mikroprozessor mit unterschiedlichen Funktionen belegt werden. Je nach Einsatz des Moduls werden die zugehörigen Pins vom Betriebssystem (Bootloader, Linux) für die entsprechenden Funktionalitäten konfiguriert.

Texas Instruments stellt hierfür auf deren Internetseiten ein '[PINMUXTOOL](#)' zur Verfügung. Dieses generiert eine Konfigurationsdatei, die dem Betriebssystem zur Verfügung gestellt wird. Nähere Informationen hierzu können Sie unseren Software Handbüchern entnehmen

Um die Funktionalität des embedded Modul Konzepts zu gewährleisten hat MKC bestimmte Funktionalitäten festgelegt. Diese dürfen nicht anders konfiguriert werden und sind im Modulstecker **fett** dargestellt. Bei den frei konfigurierbaren Signalen wird die Belegung für unsere MKC1502 Trägerkarte dargestellt.

Die Modulstecker bestehen aus 4x22 Pins im Rastermaß 2 mm.

4.1 Power

Das Modul benötigt eine Versorgungsspannung von 5V DC.

Alle auf dem Modul zusätzlich benötigten Spannungen (1.1V, 1.2V, 1.5V, 1.8V, 3.3V) werden intern (PMIC TPS65910 von Texas Instruments) erzeugt.

4.2 Modulstecker (Belegung MKC502)

Alle Signale haben ein 3,3V Interface. Die angegebenen Pullup Widerstände (Spalte PU) sind an den internen 3,3V und nicht an VIN [Modulstecker XD] angeschlossen.

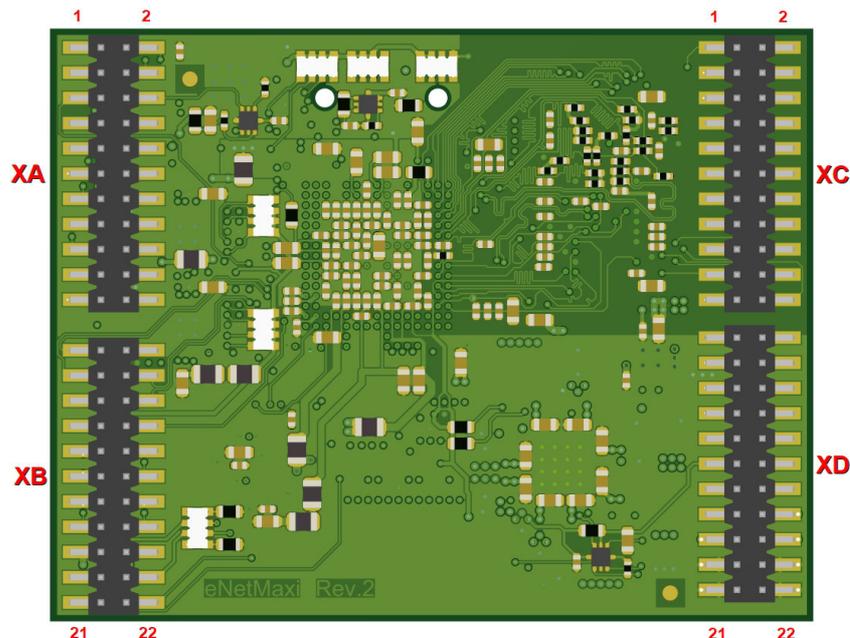


Abbildung 2: Lage der Modulkontakte

			XA				
Funktion	PU	Name	Pfosten	Name	PU	Funktion	
Transmit		UART0.TX	1 2	GPIO1.8	4K7	programmierbarer GPIO	
Receive	4K7	UART0.RX	3 4	/SPI1.CS0		SPI Master /CS0	
Transmit		UART2.TX	5 6	/UART2.CTS	4K7	Clear to send	
Receive	4K7	UART2.RX	7 8	/UART2.RTS		Ready to send	
Transmit		UART1.TX	9 10	/UART1.CTS	4K7	Clear to send	
Receive	4K7	UART1.RX	11 12	/UART1.RTS		Ready to send	
SPI-Master CLK	4K7	SPI1.CLK	13 14	GPIO3.17	4K7	programmierbarer GPIO	
SPI-Master MOSI	4K7	SPI1.D1	15 16	SPI1.D0	4K7	SPI-Master MISO	
SD-Karte	10K	MMC0.COMD	17 18	MMC0.CLK	10K	SD-Karte	
SD-Karte	10K	MMC0.DAT0	19 20	MMC0.DAT1	10K	SD-Karte	
SD-Karte	10K	MMC0.DAT2	21 22	MMC0.DAT3	10K	SD-Karte	

			XB				
Funktion	PU	Name	Pfosten	Name	PU	Funktion	
USB Master D-		USB0.DM	1 2	USB1.DRVVBUS		USB M. voltage enable	
USB Master D+		USB0.DP	3 4	USB1.DM		USB Slave D-	
USB M. voltage detect		USB0.VBUS	5 6	USB1.DP		USB Slave D+	
USB M. voltage enable		USB0.DRVVBUS	7 8	USB1.VBUS		USB Slave voltage detect	
Ethernet Receive		NET1.RX_N	9 10	NET2.RX_N		Ethernet Receive	
		NET1.RX_P	11 12	NET2.RX_P			
Ethernet Transmit		NET1.TX_N	13 14	NET2.TX_N		Ethernet Transmit	
		NET1.TX_P	15 16	NET2.TX_P			
Mittenspannung Übertrager		NET1.VDD	17 18	NET2.VDD		Mittenspannung Übertrager	
Link Status		NET1.LED1	19 20	NET2.LED1		Link Status	
USB ID Pin		USB0.ID	21 22	USB1.ID		USB ID Pin	

			XC				
Funktion	PU	Name	Pfosten	Name	PU	Funktion	
SPI Master /CS1		/SPI1.CS1	1 2	GPIO0.17		programmierbarer GPIO	
programmierbarer GPIO		TIMER7	3 4	GPIO2.19		programmierbarer GPIO	
Card Detect Eingang	10K	MMC0.CD	5 6	GPIO2.18		programmierbarer GPIO	
Interrupt Eingang		/INT0	7 8	MMC1.CLK		SDIO-BUS Clock	
programmierbarer GPIO	4K7	GPIO2.2	9 10	GPIO1.29	4K7	programmierbarer GPIO	
programmierbarer GPIO	4K7	GPIO2.3	11 12	GPIO2.4	4K7	programmierbarer GPIO	
programmierbarer GPIO		GPIO2.5	13 14	MMC1.COMD		SDIO-BUS Command	
SDIO-BUS Daten		MMC1.DAT0	15 16	GPIO0.22		programmierbarer GPIO	
SDIO-BUS Daten		MMC1.DAT1	17 18	GPIO0.23		programmierbarer GPIO	
SDIO-BUS Daten		MMC1.DAT2	19 20	GPIO0.26		programmierbarer GPIO	
SDIO-BUS Daten		MMC1.DAT3	21 22	GPIO0.27		programmierbarer GPIO	

			XD				
Funktion	PU	Name	Pfosten	Name	PU	Funktion	
programmierbarer GPIO		GPIO1.4	1 2	GPIO1.12		programmierbarer GPIO	
programmierbarer GPIO		GPIO1.5	3 4	GPIO1.13		programmierbarer GPIO	
programmierbarer GPIO		GPIO1.6	5 6	GPIO1.14		programmierbarer GPIO	
programmierbarer GPIO		GPIO1.7	7 8	GPIO1.15		programmierbarer GPIO	
I2C Master Clock	1K5	I2C0.SCL	9 10	I2C0.SDA	1K5	I2C Master Daten	
Hard Reset (POR)	10K	/RST_IN	11 12	/RST_OUT		programmierbarer Reset Out	
Backup-Spannung, z.B. Goldcap		BATT	13 14	/CONFIG_IN		General Input für Config-Mode	
+5V		VIN	15 16	VIN		+5V	
+5V		VIN	17 18	VIN		+5V	
Masse		GND	19 20	GND		Masse	
Masse		GND	21 22	GND		Masse	

Tabelle 3: Modulstecker

4.3 Konfigurierbare Pins im Modulstecker

Die folgende Tabelle zeigt die konfigurierbaren Pins der Modulkontakte. Nicht konfiguriert haben diese Pins die in der Spalte Default angegebenen Funktionen. MKC unterstützt die Konfiguration auf die hier angegebenen Funktionen.

Es sind wesentlich mehr Funktionen/Kombinationen möglich.

Pfosten	Am3352 Default	Am3352 Pin	GPIO	MKC1502	Alternative
XA2	/UART0.CTS	E18	GPIO1.8	GPIO1.8	
XA4	/UART0.RTS	E17	GPIO1.9	/SPI1.CS0	
XA5	MII1_RXCLK	L18	GPIO3.10	UART2.TX	
XA6	LCD_DATA8	U1	GPIO2.14	/UART2.CTS	
XA7	MII1_TXCLK	K18	GPIO3.9	UART2.RX	
XA8	LCD_DATA9	U2	GPIO2.15	/UART2.RTS	
XA9	UART1.TX	D15	GPIO0.15	UART1.TX	CAN1.RX
XA10	/UART1.CTS	D18	GPIO0.12	/UART1.CTS	CAN0.TX
XA11	UART1.RX	D16	GPIO0.14	UART1.RX	CAN1.TX
XA12	/UART1.RTS	D17	GPIO0.13	/UART1.RTS	CAN0.RX
XA13	MCASP0_ACLKX	A13	GPIO3.14	SPI1.CLK	
XA14	MCASP0_AHCLKR	C12	GPIO3.17	GPIO3.17	
XA15	MCASP0_AXR0	D12	GPIO3.16	SPI1.D1	
XA16	MCASP0_FSX	B13	GPIO3.15	SPI1.D0	
XC1	XDMA_EVENT_INTR0	A15	GPIO0.19	/SPI1.CS	
XC2	MII1_TXD2	K15	GPIO0.17	GPIO0.17	
XC3	XDMA_EVENT_INTR1	D14	GPIO0.20	TIMER7	
XC4	MII1_RXD2	L16	GPIO2.19	GPIO2.19	
XC5	/SPI0_CS1	C15	GPIO0.6	MMC0.CD	
XC6	MII1_RXD3	L17	GPIO2.18	GPIO2.18	
XC7	ECAP0_IN_PWM0_OUT	C18	GPIO0.7	/INT0 ¹⁾	
XC8	/GPMC_CS1	U9	GPIO1.30	MMC1.CLK	GPMC.CLK
XC9	/GPMC_ADV	R7	GPIO2.2	GPIO2.2	/GPMC.ADV
XC10	/GPMC_CS0	V6	GPIO1.29	GPIO1.29	/GPMC.CS
XC11	/GPMC_OE	T7	GPIO2.3	GPIO2.3	/GPMC.OE
XC12	/GPMC_WE	U6	GPIO2.4	GPIO2.4	/GPMC.WE
XC13	/GPMC_BE0	T6	GPIO2.5	GPIO2.5	/GPMC.BE0
XC14	/GPMC_CS2	V9	GPIO1.31	MMC1.CMD	/GPMC.BE1
XC15	GPMC_AD0	U7	GPIO1.0	MMC1.DAT0	GPMC.AD0
XC16	GPMC_AD8	U10	GPIO0.22	GPIO0.22	GPMC.AD8
XC17	GPMC_AD1	V7	GPIO1.1	MMC1.DAT1	GPMC.AD1
XC18	GPMC_AD9	T10	GPIO0.23	GPIO0.23	GPMC.AD9
XC19	GPMC_AD2	R8	GPIO1.2	MMC1.DAT2	GPMC.AD2
XC20	GPMC_AD10	T11	GPIO0.26	GPIO0.26	GPMC.AD10
XC21	GPMC_AD3	T8	GPIO1.3	MMC1.DAT3	GPMC.AD3
XC22	GPMC_AD11	U12	GPIO0.27	GPIO0.27	GPMC.AD11
XD1	GPMC_AD4	U8	GPIO1.4	GPIO1.4	GPMC.AD4
XD2	GPMC_AD12	T12	GPIO1.12	GPIO1.12	GPMC.AD12
XD3	GPMC_AD5	V8	GPIO1.5	GPIO1.5	GPMC.AD5
XD4	GPMC_AD13	R12	GPIO1.13	GPIO1.13	GPMC.AD13
XD5	GPMC_AD6	R9	GPIO1.6	GPIO1.6	GPMC.AD6
XD6	GPMC_AD14	V13	GPIO1.14	GPIO1.14	GPMC.AD14
XD7	GPMC_AD7	T9	GPIO1.7	GPIO1.7	GPMC.AD7
XD8	GPMC_AD15	U13	GPIO1.15	GPIO1.15	GPMC.AD15
XD9	I2C0_SCL	C16	GPIO3.6	I2C0.SCL	
XD10	I2C0_SDA	C17	GPIO3.5	I2C0.DAT	

Tabelle 4: Übersicht konfigurierbare Pins im Modulstecker

4.3.1 UART Schnittstellen

Der Am3352 Mikroprozessor verfügt über 6 unabhängige UART Funktionsblöcke, von denen das eNetMaxi maximal drei (UART0, UART1 und UART2) nutzt. Die Features der UART Schnittstellen in Stichworten:

- 16C750 kompatibel
- jeweils 64 Byte Receive und Transmit FIFO
- Baudraten von 300 bps bis 3.6864 Mbps
- Auto-baud von 1200 bps bis 115.2 Kbps
- Software/Hardware Flusskontrolle
- Programmierbare Xon/Xoff Character
- Programmierbares Auto-RTS und Auto CTS
- 5, 6, 7, or 8-bit Character
- Even, odd, mark, space, no Parity
- 1, 1.5, oder 2 Stopp Bits
- Line Break Erzeugung und Erkennung
- Interrupt fähig

Schnittstelle	Signalname	Stecker-Pin	CPU Pin (CPU-Block)
UART0 ¹⁾	RX	XA3	E15 (UART0)
UART0 ¹⁾	TX	XA1	E16 (UART0)

Schnittstelle	Signalname	Stecker-Pin	CPU Pin (CPU-Block)
UART1	CTS	XA10	D18 (UART1)
UART1	RTS	XA12	D17 (UART1)
UART1	RX	XA11	D16 (UART1)
UART1	TX	XA9	D15 (UART1)

Schnittstelle	Signalname	Stecker-Pin	CPU Pin (CPU-Block)
UART2	CTS	XA6	U1 (UART2)
UART2	RTS	XA8	U2 (UART2)
UART2	RX	XA7	K18 (UART2)
UART2	TX	XA5	L18 (UART2)

Tabelle 5: UART0 / UART1 / UART2 Konfiguration Modul-Pin zu CPU-Pin/Block

¹⁾ Die UART0 Schnittstelle ist prinzipiell umprogrammierbar, sollte aber nicht geändert werden, da diese bei MKC als Systemkonsole genutzt wird. Die UART Schnittstellen arbeiten mit einem Takt von 48 MHz. Damit sind die folgenden Baudraten möglich:

Baudrate	Fehler in %
300, 600, 1200, 2400, 4800	0
9600, 14400, 19200, 28800, 38400, 57600, 115200, 230400, 460800, 921600, 1843200	0.16
3000000	0
3686400	0.16

Tabelle 6: UART Baudraten und Fehler

4.3.2 SPI Schnittstellen

Der Am3352 Mikroprozessor verfügt über 2 unabhängige SPI Funktionsblöcke. SPI0 wird auf dem eNetMaxi für ein serielles SPI FLASH benutzt (Kapitel Serielles SPI FLASH). SPI1 ist auf den Modulstecker (Belegung MKC502) geführt.

Die Features der SPI Schnittstellen in Stichworten:

- maximale Übertragungsrate von 48 MHz
- gepufferte receive/transmit Register (1 Wort)
- programmierbares 64 Byte FIFO
- DMA und Interrupt fähig
- Full duplex / Half duplex
- Multi-channel master oder single channel slave Interface
- SPI Wortlänge 4 bis 32 Bit
- Clock Phase und Polarität einstellbar

Schnittstelle	Signalname	Stecker-Pin	CPU Pin (CPU-Block)
SPI1	MOSI	XA15	D12 (SPI1)
SPI1	MISO	XA16	B13 (SPI1)
SPI1	SCLK	XA13	A13 (SPI1)
SPI1	CS0	XA4	E17 (SPI1)
SPI1	CS1	XC1	A15 (SPI1)

Tabelle 7: SPI Konfiguration Modul-Pin zu CPU-Pin/Block

Die Schnittstelle lässt sich mit oder ohne FIFO betreiben, wobei das FIFO aufgeteilt werden kann zwischen Receive und Transmit Kanal. Die folgenden Prinzipschaltbilder sind dem „Technical Reference Manual“ von Texas Instruments entnommen.

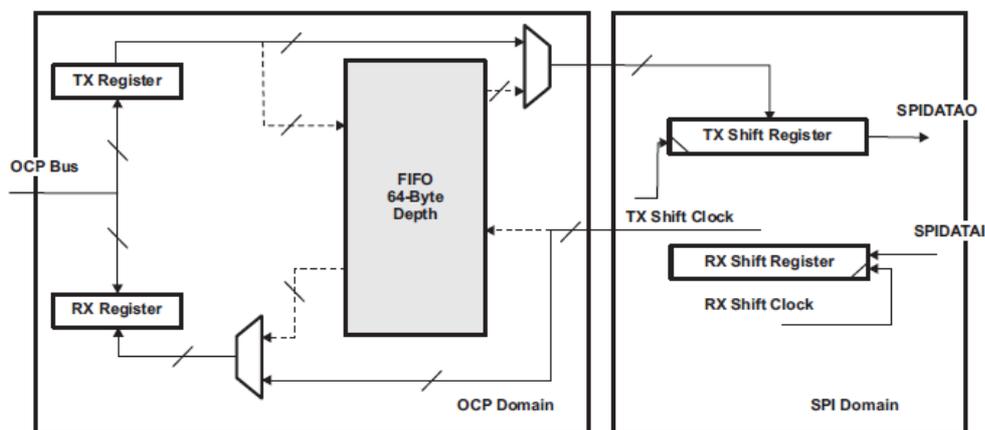


Abbildung 3: SPI Interface ohne FIFO

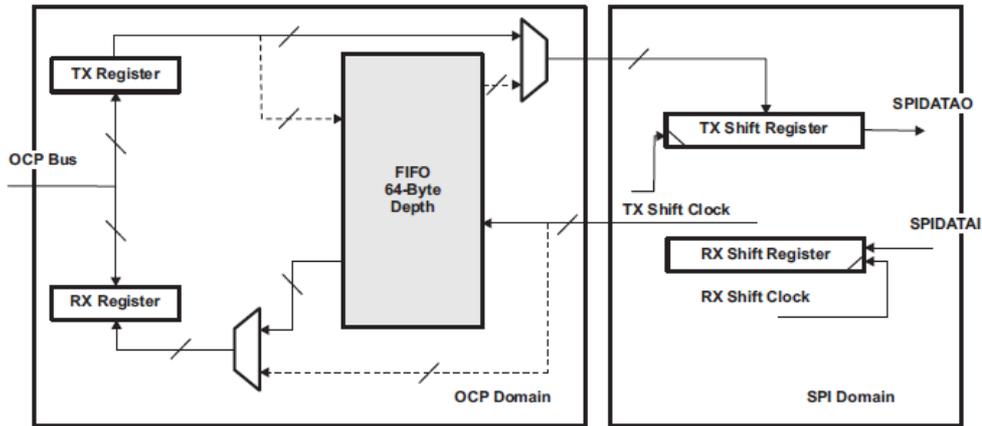


Abbildung 4: SPI Interface mit Empfangs-FIFO

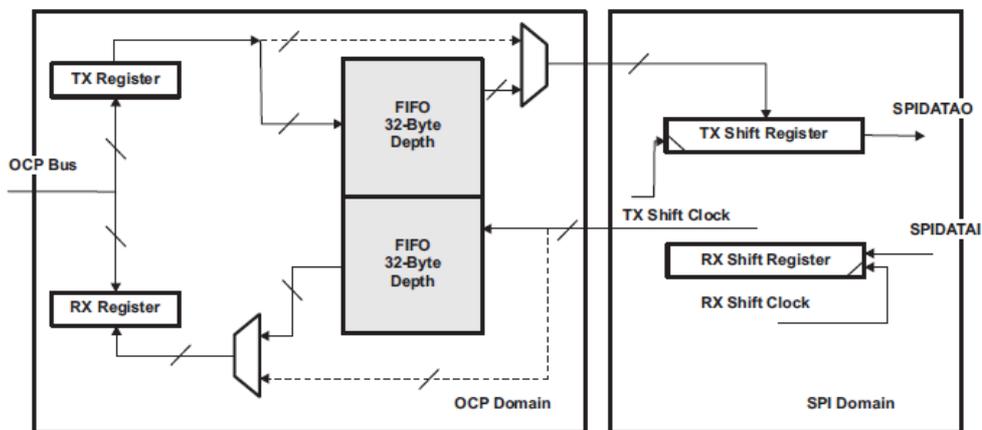


Abbildung 5: SPI Interface mit Sende-/Empfangs-FIFO

4.3.3 I²C Schnittstellen

Der Am3352 Mikroprozessor verfügt über 3 unabhängige fähige I²C Funktionsblöcke. Auf dem eNetMaxi wird nur das I2C0 Interface genutzt. Das I2C Interface ist auf den Modulstecker geführt.

Die Features der I²C Schnittstellen in Stichworten:

- Philips I²C Version 2.1 Spezifikation
- unterstützt Standard Mode (bis 100Kb/s) und Fast Mode (bis 400Kb/s)
- 7 und 10 Bit Adressierung
- Multimaster fähig
- 32 Bit FIFO für read oder write
- DMA und Interrupt fähig

Schnittstelle	Signalname	Stecker-Pin	CPU Pin (CPU-Block)
I2C0	SCL	XD9	C16 (I2C0)
I2C0	SDA	XD10	C17 (I2C0)

Tabelle 8: I2C Konfiguration Modul-Pin zu CPU-Pin/Block

4.3.4 CAN Schnittstelle

Der Am3352 Mikroprozessor verfügt über 2 unabhängige CAN (D_CAN) Funktionsblöcke. Das eNetMaxi nutzt diese Schnittstellen nicht. Dieser Funktionsblock kann alternativ zur Schnittstelle UART1 auf dem Modulstecker genutzt werden. Die Features der CAN Schnittstellen in Stichworten:

- CAN Protokoll Version 2.0 Part A, B (ISO 11898-1)
- Bitrate bis 1MBit/s
- 16, 32, 64 oder 128 Message Objekte
- Programmierbares FIFO für Message Objekte
- Automatischer bus-on mittels 32-Bit Timer
- Message RAM Parity Check
- Interrupt und DMA Support

Schnittstelle	Signalname	Stecker-Pin	CPU Pin (CPU-Block)
CAN0	TX	XA10	D18 (CAN0)
CAN0	RX	XA12	D17 (CAN0)
CAN1	TX	XA11	D16 (CAN1)
CAN1	RX	XA9	D15 (CAN1)

Tabelle 9: CAN Konfiguration Modul-Pin zu CPU-Pin/Block

4.3.5 SD/SDIO Schnittstelle

Der Am3352 Mikroprozessor verfügt über 3 unabhängige MMC (Multi Media Card) Funktionsblöcke. Das eNetMaxi nutzt optional die Schnittstelle MMC1 am Modulstecker (Belegung MKC502).

Schnittstelle	Signalname	Stecker-Pin	CPU Pin (CPU-Block)
MMC1	CMD	XC14	V9 (MMC1)
MMC1	CLK	XC8	U9 (MMC1)
MMC1	DAT0	XC15	U7 (MMC1)
MMC1	DAT1	XC17	V7 (MMC1)
MMC1	DAT2	XC19	R8 (MMC1)
MMC1	DAT3	XC21	T8 (MMC1)

Tabelle 10: SD/SDIO Konfiguration Modul-Pin zu CPU-Pin/Block

4.3.6 PWM Schnittstelle

Beim eNetMaxi ist der Ausgang von Timer7 (siehe Kapitel Timer auf Seite 26) auf Ausgänge des Mikroprozessors gemultiplext (Modulstecker (Belegung MKC502)). Damit lässt sich dieser Ausgang z.B. als Taktgenerator für Frequenzen bis 24 MHz nutzen. Da das Tastverhältnis der Ausgangsfrequenz frei einstellbar ist, bietet sich auch eine Verwendung als PWM Generator an.

Schnittstelle	Signalname	Stecker-Pin	CPU Pin (CPU-Block)
TIMER7	TIMER7	XC3	D14 (TIMER7)

Tabelle 11: PWM Konfiguration Modul-Pin zu CPU-Pin und PWM Generator

4.3.7 GPMC

Alternativ verfügt das eNetMaxi über ein paralleles 16 Bit gemultiplextes Adress/Daten Interface:

- Synchroner oder asynchroner Betrieb
- 4-8-16 Word Burst Transfers im synchron mode
- 100 MHz max. Speichertakt
- 3,3V Interface

Schnittstelle	Signalname	Stecker-Pin	CPU Pin (CPU-Block)
GPMC	CLK	XC8	U9 (GPMC)
GPMC	CS	XC10	V6 (GPMC)
GPMC	ADV	XC9	R7 (GPMC)
GPMC	OE	XC11	T7 (GPMC)
GPMC	WE	XC12	U6 (GPMC)
GPMC	BE0	XC13	T6 (GPMC)
GPMC	BE1	XC14	V9 (GPMC)
GPMC	AD0	XC15	U7 (GPMC)
GPMC	AD1	XC17	V7 (GPMC)
GPMC	AD2	XC19	R8 (GPMC)
GPMC	AD3	XC21	T8 (GPMC)
GPMC	AD4	XD1	U8 (GPMC)
GPMC	AD5	XD3	V8 (GPMC)
GPMC	AD6	XD5	R9 (GPMC)
GPMC	AD7	XD7	T9 (GPMC)
GPMC	AD8	XC16	U10 (GPMC)
GPMC	AD9	XC18	T10 (GPMC)
GPMC	AD10	XC20	T11 (GPMC)
GPMC	AD11	XC22	U12 (GPMC)
GPMC	AD12	XD2	T12 (GPMC)
GPMC	AD13	XD4	R12 (GPMC)
GPMC	AD14	XD6	V13 (GPMC)
GPMC	AD15	XD8	U13 (GPMC)

Tabelle 12: GPMC Konfiguration Modul-Pin zu CPU-Pin/Block

4.4 Fixe Pins im Modulstecker

4.4.1 System-Control Pins

Das /RST_IN Signal wird mit dem Signal /RSTPWRON des PMIC (DC/DC-Wandler TPS65910) verknüpft auf den /PWRONRST der CPU geführt.

Das /RST_OUT Signal wird vom Ausgang eines SN74AUP2G08 UND-Gatters getrieben (totem-pole) und ist eine Verknüpfung von /WARMRST der CPU und dem GPIO3.4. Hierzu muss der Pin MII1_RX_DV der CPU entsprechend umprogrammiert werden. Daraus folgt auch, dass dieser Pin nicht für andere Funktionen genutzt werden kann.

/CONFIG_IN liegt auf dem GPIO des PMIC und kann von der CPU ausgelesen werden. Er wird in der Regel dazu verwendet, eine Konfiguration zu initiieren.

Schnittstelle	Signalname	Stecker-Pin	CPU Pin (CPU-Block)
RESET	/RST_IN	XD11	B15 (PRCM)
RESET	/RST_OUT	XD12	J17 (GPIO3.4)
CONFIG Ethernet	/CONFIG_IN	XD14	PMIC Pin39

Tabelle 13: System-Control Konfiguration Modul-Pin zu CPU-Pin/Block

4.4.2 SD-Karten Schnittstelle

Der Am3352 Mikroprozessor verfügt über 3 unabhängige MMC (MultiMediaCard) Funktionsblöcke. Das eNetMaxi nutzt die Schnittstelle MMC0 für ein 4Bit SD-Karten Interface auf dem Modulstecker (Belegung MKC502).

Die maximale Übertragungsrate beträgt 24 MByte/s. Damit können SDHC Class 10 Speicherkarten betrieben werden.

Schnittstelle	Signalname	Stecker-Pin	CPU Pin (CPU-Block)
MMC0	CMD	XA17	G18 (MMC0)
MMC0	CLK	XA18	G17 (MMC0)
MMC0	DAT0	XA19	G16 (MMC0)
MMC0	DAT1	XA20	G15 (MMC0)
MMC0	DAT2	XA21	F18 (MMC0)
MMC0	DAT3	XA22	F17 (MMC0)

Tabelle 14: SPI Konfiguration Modul-Pin zu CPU-Pin/Block

In erster Linie dient die SD-Karte als wechselbarer Massenspeicher. Daneben steht das SD Interface an erster Stelle der Boot-Code Suchliste des ROM-Codes. Wird eine entsprechend beschriebene Karte eingesetzt, wird das eNetMaxi von diesem Medium gebootet. Damit bietet die SD-Karte eine elegante Möglichkeit, die Firmware des eNetMaxi zu aktualisieren.

Weitere Stichworte:

- Support für SDA 3.0 Part A2 Programmiermodell
- SD command/response sets definiert in der SD Physical Layer specification v2.00
- SD Host Controller Standard Specification sets definiert in der SD card specification Part A2 v2.00
- 3,3V Interface
- 10K Pullups auf allen 7 Signalen (CLK, CMD, DATA[0..3],CD)

4.4.3 Batterie-Eingang

Für diesen Modul-Pin kann auf der Trägerkarte eine Kapazität zum Erhalt der RTC Daten vorgesehen werden. Der zugehörige Pin VBACKUP des PMIC (DC/DC-Wandler TPS65910) beinhaltet eine Ladeschaltung für Supercaps bis 2F. Soll kein Supercap verwendet werden, ist eine 'normaler' Kondensator zwischen 1 μ F und 40 μ F vorzusehen. Ein Supercap muss über einen Serienwiderstand angebunden werden. Für Einzelheiten wird an dieser Stelle auf das [Datenblatt](#) verwiesen.

Schnittstelle	Signalname	Stecker-Pin	
Power	BATT	XD13	PMIC Pin 27

Tabelle 15: VBATT Konfiguration Modul-Pin zu CPU-Pin/Block

Soll eine Batterie oder andere Spannungsquelle verwendet werden, ist eine Schutzdiode vorzusehen.

4.4.4 UART0 Schnittstelle

Der ROM-Code der CPU und der mitgelieferte Boot-Loader verwenden diese Schnittstelle (nur RX/TX). Auch verwenden Betriebssysteme sie in der Regel als System-Konsole. Deshalb sollte diese Default-Konfiguration nicht geändert werden. Features und Pin-Belegung siehe Kapitel UART Schnittstellen auf Seite 14.

4.4.5 USB Schnittstellen

Die zwei im Am3352 Mikroprozessor vorhandenen USB Subsystemen werden auf dem eNetMaxi im OTG-Mode verwendet. Die Features der USB Subsysteme in Stichworten:

- Mentor USB 2.0 OTG core
- USB 2.0 Peripherie Schnittstelle mit HS (480 Mb/s) und FS (12 Mb/s) Geschwindigkeit
- USB 2.0 Host oder OTG Schnittstelle mit HS (480 Mb/s), FS (12 Mb/s), und LS (1.5 Mb/s) Geschwindigkeit
- unterstützt alle Transfer-Modi (control, bulk, interrupt, and isochronous)
- unterstützt ISO mode mit hoher Bandbreite
- unterstützt 16 Empfangs- und 16 Sendende-Endpunkte
- 32K Endpoint FIFO
- dedizierter DMA Controller mit 15 Empfangs- und 15 Sendende-Kanälen

Schnittstelle	Signalname	Stecker-Pin	CPU Pin (CPU-Block)
USB0	VBUS	XB5	P15 (USB0)
USB0	ID	XB21	P16 (USB0)
USB0	DM	XB1	N18 (USB0)
USB0	DP	XB3	N17 (USB0)
USB0	DRVVBUS	XB7	F16 (USB0)
USB1	VBUS	XB8	T18 (USB1)
USB1	ID	XB22	P17 (USB1)
USB1	DM	XB4	R18 (USB1)
USB1	DP	XB6	R17 (USB1)
USB1	DRVVBUS	XB2	F15 (USB1)

Tabelle 16: USB Konfiguration Modul-Pin zu CPU-Pin/Block

Die Signale USBx_CE des Prozessors werden nicht verwendet.

4.4.6 Ethernet Netzwerk

Das eNetMaxi verfügt über zwei unabhängige Netzwerk Schnittstellen. Die MAC Controller befinden sich im Am3352 Mikroprozessor, die zugehörigen PHYs sind die Typen LAN8720A von SMSC. Hierbei handelt es sich um 10/100 Fast Ethernet Transceiver mit Auto-MDIX und Autonegotiation. Die Übertrager und RJ45 Stecker sind nicht auf dem Modul und müssen auf dem Extension-Modul realisiert werden.

Schnittstelle	Signalname	Stecker-Pin	CPU-Block
NET1	RX_N	XB9	RMII1
NET1	RX_P	XB11	RMII1
NET1	TX_N	XB13	RMII1
NET1	TX_P	XB15	RMII1
NET1	VDD	XB17	
NET1	LED1	XB19	
NET2	RX_N	XB10	RMII2
NET2	RX_P	XB12	RMII2
NET2	TX_N	XB14	RMII2
NET2	TX_P	XB16	RMII2
NET2	VDD	XB18	
NET2	LED1	XB20	

Tabelle 17: Ethernet Konfiguration Modul-Pin zu CPU-Pin/Block

Als Interface zwischen MAC und PHY wird das RMII Interface des Mikroprozessors genutzt.

Die PHYs besitzen jeweils 2 Ausgänge für LED Status Anzeigen.

	An	Aus	Blinkt
LED1	LINK Status Verbindung etabliert	keine Verbindung	RX/TX aktiv
LED2	100BaseT	10BaseT	---

Tabelle 18: Funktion der Ethernet LED Statusanzeigen

LED1 ist auf dem Modulstecker (Belegung MKC502) verfügbar, LED2 befindet sich auf dem eNetMaxi.

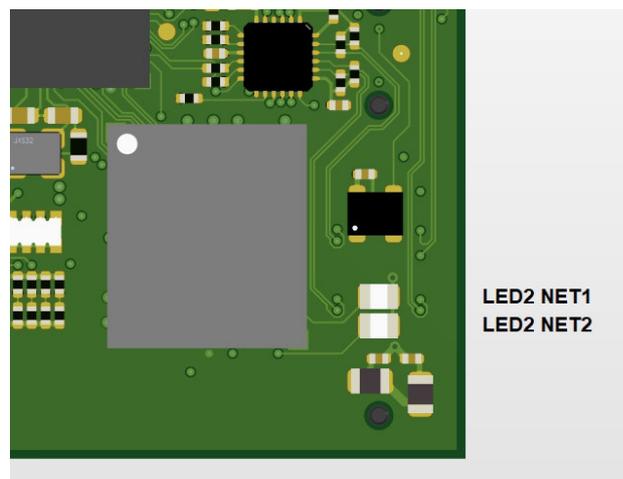


Abbildung 6: Ethernet LED2 Statusanzeigen

Die LED Signale sind 'activ-high' inklusive 330 Ohm Vorwiderstand. Somit sind die LEDs ohne Widerstand gegen Masse zu verbinden.

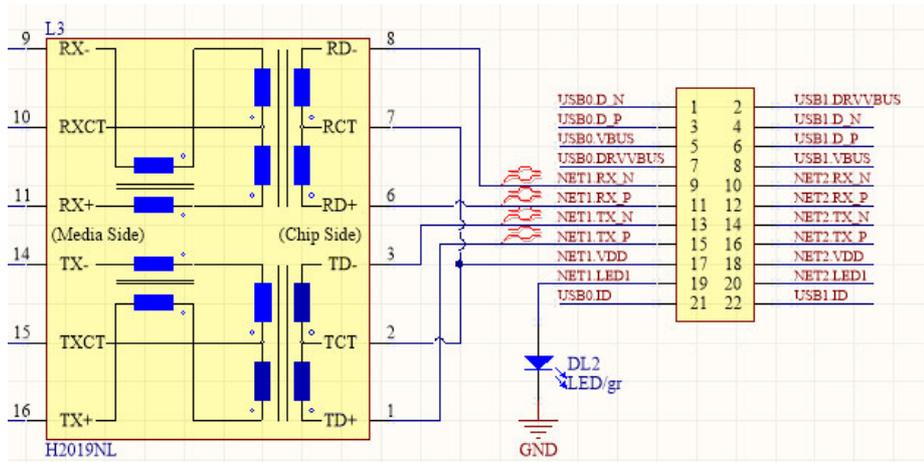


Abbildung 7: Ethernet Anschluss Extension-Board

5 ARM SoC

Auf dem Modul befindet sich der Texas Instruments Am3352 Cortex A8 ARM 32Bit Mikroprozessor im ZCZ BGA-Gehäuse mit 324 Balls.

Neben verschiedenen Speicher-Schnittstellen sind diverse I/O Funktionen auf dem Chip integriert. In den folgenden Kapiteln sind die in dem eNetMaxi verwendeten Funktionen beschrieben. Diese befinden sich entweder auf dem eNetMaxi Modul selbst (lokale Funktionen), oder das für diese Funktion benötigte Interface ist auf die Modulstecker auf der Unterseite des Moduls geführt (externe Funktionen). Diese externen Funktionen sind bereits in Kapitel Definition der Anschlüsse ab Seite 11 beschrieben worden.

Die Möglichkeiten der einzelnen Funktionen sind sehr umfangreich. Eine vollständige Wiedergabe würde den Rahmen dieses Dokumentes sprengen, deshalb befinden sich hier nur eNetMaxi spezifische Detailbeschreibungen. Eine komplette Dokumentation des Am3352 Mikroprozessors ist auf der Herstellerseite von Texas Instruments zu finden.

5.1 PLL

Im Mikroprozessor sind mehrere PLL-Module integriert, die aus dem externen 24 MHz Quarz sämtliche benötigten Frequenzen generieren.

5.2 DDR3 SDRAM

Das DDR3-SDRAM Interface der CPU ist 16 Bit breit mit einer Datenrate von 800MHz und einem Adressraum von 1GB. Eingesetzt wird ein Baustein vom Typ MT41J256M16 von Micron im 96-Ball BGA Gehäuse. Pin-kompatible Typen von anderen Herstellern sind verfügbar.

Die Kapazität beträgt damit 512MB. Eine Kapazität von 1024MB ist verfügbar.

5.3 RTC

Ein Funktionsmodul des Am3352 Mikroprozessors ist eine Standard Real-Time Clock. Sie verfügt auf dem eNetMaxi über einen eigenen 32,768 KHz Quarzoszillator. Es besteht die Möglichkeit über Modulstecker einen SuperCap bzw. GoldCap oder Batterie als Backup-Spannung anzuschließen (Pin BATT). Der auf dem eNetMaxi befindliche Power-Controller beinhaltet den Switch und eine Ladeschaltung für eine solche Kapazität.

Die RTC kann zu bestimmten programmierbaren Zeiten oder auch periodisch einen Interrupt auslösen. Zusätzlich kann ein Interrupt programmiert werden, der ausgelöst wird, wenn ein Register der RTC geändert wird. So kann überprüft werden, ob unberechtigt die Zeit geändert wurde.

5.4 Serielles SPI FLASH

Der Am3352 Mikroprozessor verfügt über 2 unabhängige SPI Funktionsblöcke. SPI0 wird auf dem eNetMaxi für ein serielles FLASH benutzt.

Der ROM-Code der CPU versucht den Bootloader zu laden und auszuführen. Dieser Bootloader befindet sich im Normalbetrieb des eNetMaxi in einem seriellen SPI FLASH am CS0 der SPI0 Schnittstelle der CPU.

Die maximale SPI-Clock Frequenz der CPU beträgt 48 MHz. Eingesetzt wird ein Micron M25QL128 mit 16 MB Kapazität. Das SPI FLASH kann Page-, Subsektor-, Sektorweise oder komplett (Bulk) gelöscht werden.

Die Sektoren können getrennt schreibgeschützt werden. Der Baustein verfügt über Software-Schreibschutz-Mechanismen, mit denen der Schreibschutz für Sektoren getrennt per Software ein/ausgeschaltet werden kann. Der Software Schreibschutz ist nach Reset oder Power-up aufgehoben.

Außerdem gibt es einen Sektor Hardware Schreibschutz, der einmal per Software aktiviert, nicht mehr deaktiviert werden kann. Dazu muss der Pin '/W' des FLASH auf Masse, d.h. logisch '0', liegen. Auf dem eNetMaxi befindet sich an dieser Stelle ein 0 Ohm Widerstand (R52). Ist der Hardware-Schreibschutz einmal aktiviert, kann er nur noch deaktiviert werden, wenn dieser Widerstand entfernt wird.

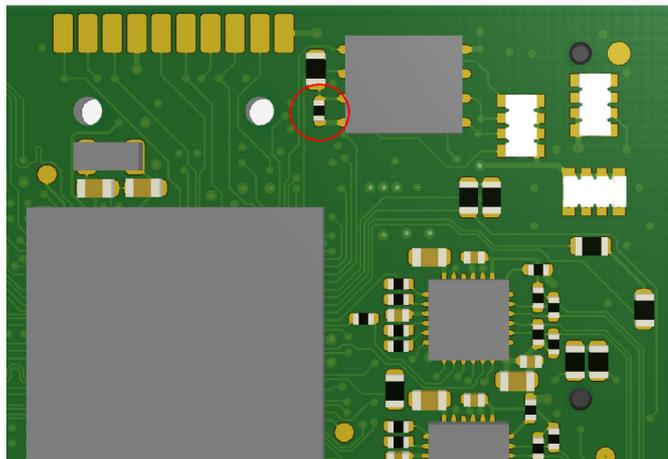


Abbildung 8: Hardware-Schreibschutz SPI FLASH

Eine genaue Beschreibung der Schreibschutz Mechanismen entnehmen sie bitte dem Datenblatt des jeweiligen SPI-FLASH.

5.5 eMMC

Ein eMMC Baustein bildet den Massenspeicher des eNetMaxi Moduls. Bei eMMC Bauteilen handelt es sich um Multichip Module mit einem oder mehreren NAND Bausteinen und einem integrierten Controller. Der integrierte Controller entlastet die CPU von den aufwändigen Berechnungen wie Blocking/Deblocking und wear-leveling.

Der Daten- und Kommando-Transfer erfolgt über ein 8 Bit breites serielles Interface. Sowohl die Hardware (Formfaktor und Pinout) als auch die Firmware (Kommandos und Registersatz) sind nach JEDEC/MMCA standardisiert.

Zur Zeit werden 4GB Typen von Sandisk oder Toshiba eingesetzt. Der Speicher ist am MMC2 Port des Am3352 Mikroprozessors angeschlossen und wird mit der maximalen Datenrate von 48 MHz betrieben.

5.6 Watchdog

Der Watchdog des Am3352 Mikroprozessors ist mit einem freilaufenden 32 Bit Aufwärtszähler realisiert. Der Eingangstakt dieses Zählers ist 32,768 KHz mit einem programmierbaren Vorteiler von 1 bis 128. Damit lassen sich Zeiten von ca. 61us bis etwa 194 Tage einstellen. Der Takt kann nicht deaktiviert werden.

Nach Ablauf des Zählers wird ein WARMRESET ausgeführt. Der Unterschied zum Power-on Reset besteht im Wesentlichen darin, dass die Strappings nicht eingelesen werden und die PLLs und die Clock-Divider nicht rückgesetzt werden. Test und Emulationslogik wird ebenfalls nicht zurückgesetzt. Es wird wie beim Kaltstart der ROM-Code ausgeführt.

Um die Funktionsweise des Watchdog zu verdeutlichen, sind in der folgenden Tabelle einige – nicht alle – der Watchdog Register dargestellt.

Register	Funktion
WDT_WCRR	Zähler Register, read-only
WDT_WCLR	Prescaler Einstellungen: Kann nur bei gestopptem Zähler beschrieben werden
WDT_WLDR	Load Register: Startwert des Zählers, kann nur bei gestopptem Zähler beschrieben werden. Wird nach Zähler Überlauf und Trigger in den Zähler (WDT_WCRR) kopiert.
WDT_WDLY	Delay Register: Zählerstand bei dem ein Interrupt generiert werden kann. Kann nur bei gestopptem Zähler beschrieben werden.
WDT_WTRG	Watchdog Trigger: Zum Triggern des Zählers muss in dieses Register ein vom letzten Beschreiben differierender Wert geschrieben werden. Kopiert WDT_WLDR nach WDT_WCRR und setzt den Prescaler zurück.
WDT_WSPR	Start-Stopp Register: Zum Start bzw. Stopp des Zählers muss eine unterschiedliche Sequenz von zwei 32 Bit Worten in dieses Register geschrieben werden.

Tabelle 19: Watchdog Register

Das WDT_WDLY Register dient dazu, kurz vor Ablauf des Watchdogs z.B. einen Alarm auszulösen.
Beschreiben des WDT_WLDR Registers lädt nicht den Zähler!

Nach einem Reset ist der Watchdog aktiv, läuft aber nicht. Er muss durch Schreiben der entsprechenden Sequenz in das WDT_WSPR Register gestartet werden.

5.7 Timer

Der Am3352 Mikroprozessor verfügt über 8 Timer (bei Texas DMTimer genannt):

- 32 Bit freilaufende Aufwärtszähler
- Compare und Capture Modes
- Auto-reload Mode
- Start-Stop Mode
- Programmierbarer Takteingang
- Programmierbarer Interrupt bei compare, capture oder overflow
- Registerzugriff jederzeit möglich

Timer0 und Timer1 nehmen eine Sonderstellung ein. Sie werden über die 'Wakeup-Domain' versorgt, alle anderen Timer über die 'Peripheral-Domain'. Die 'Wakeup-Domain' kann im Gegensatz zur 'Peripheral-Domain' nicht per Software deaktiviert werden.

Timer0 wird nur mit einem internen 32 KHz RC-Oszillator getaktet und ist für Wakeup Zwecke bestimmt.

Timer1 kann von verschiedenen Quellen getaktet werden, unter Anderem vom 32,768 KHz Oszillator der RTC, und ist in erster Linie als 1ms Systemtakt für Betriebssysteme vorgesehen (interner Name: 'DMTimer 1ms').

Die übrigen Timer können entweder vom 24 MHz Systemtakt oder einem PLL generierten 32,768 KHz Takt betrieben werden. Der Takt geht über einen Vorteiler von 1 bis 256 in den 32 Bit Zähler.

Wird eine Timer verwendet um zyklische Interrupts zu generieren, sind folgende Interrupt-Intervalle möglich, wobei in der Tabelle nur die minimalen und maximalen Werte für den Vorteiler aufgeführt sind:

Eingangstakt	Vorteiler	Auflösung	Interrupt-Intervall
32,768 KHz	1	30,5us	30,5us bis 36,4 Std.
32,768 KHz	256	7,8ms	7,8ms bis 388 Tage
24 MHz	1	41,7ns	41,7ns bis 179s
24 MHz	256	10,7us	10,7us bis 12,7 Std.

Tabelle 20: Timer Interrupt Intervalle

Die Compare Ausgänge von Timer4 bis Timer7 können auf Peripherie-Pins des Am3352 Mikroprozessors gelegt werden. MKC macht z.B. mit der MKC1502 als Trägerkarte mit Timer7 Gebrauch von dieser Möglichkeit.

6 Anhang

6.1 JTAG/DEBUG (optional)

Der JTAG Port existiert nur in Form von „Pads“, d.h. zum Debuggen mittels JTAG Interface muss ein Stecker auf diese fixiert werden. Ein Adapter ist auf Anfrage verfügbar.

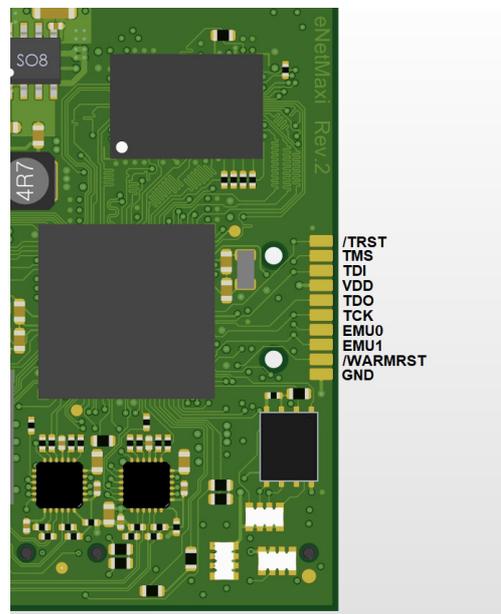


Abbildung 9: JTAG/DEBUG Pads

6.2 Mechanische Abmessungen

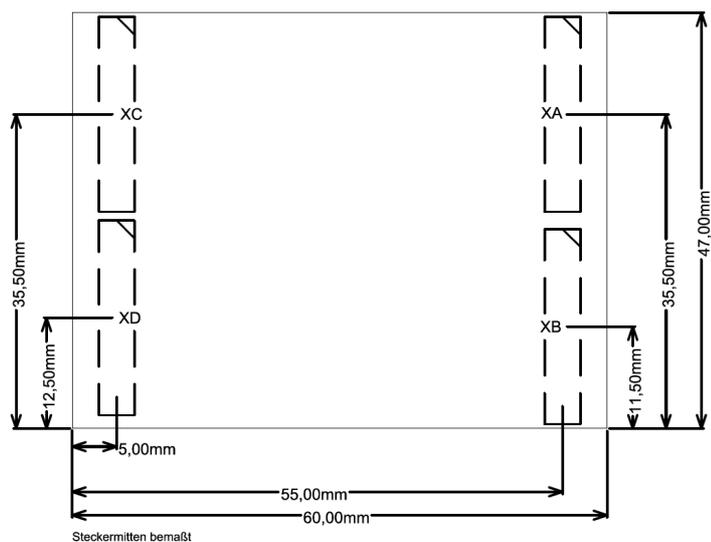


Abbildung 10: Bemaßung