

Ethernet Modul



Netzwerk-Modul für synchronen und asynchronen Datenaustausch

Eigenschaften:

- | | |
|--|--|
| <ul style="list-style-type: none">• Ethernet 10/100 Controller• TCP/IP Protokoll implementiert• bis zu 4 MByte/s Übertragungsrate• einfaches Handshake-Nutzer-Interface | <ul style="list-style-type: none">• frei einstellbare Netzwerkparameter• Spannungsversorgung 3.3V• Rastermaß: 2.54 mm• Maße: 43 x 31 mm |
|--|--|

Versorgungsspannung:

Zum Betrieb des Moduls ist eine Spannung von 3.3V notwendig. Diese ist an einem der Versorgungspins über die Pinleiste bereitzustellen (siehe Pinbelegungen).

Kommunikation:

Die Verarbeitung der OSI Schichten 1-4 ist komplett in der Hardware des Moduls implementiert. Die Netzwerkkommunikation erfolgt über TCP/IP, wobei das Modul den TCP Server abbildet. Alle relevanten Parameter des Ethernet Controllers sind über ein bereitgestelltes Konfigurationstool frei einstellbar.

Der Datenaustausch über die Pinleisten basiert auf controllerunabhängigen, einfachen Protokollen. Um die Interaktion mit der steuernden Hardware einfach und nutzerfreundlich zu gestalten, sind Kommunikationsprotokolle für den asynchronen bzw. synchronen Modus auf dem Modul implementiert. Die Sende- und Empfangspfade sind dabei unabhängig voneinander. Zum Empfangen bzw. Senden von Daten dienen die Signale Rx_D/Tx_D, Rx_D_Vd/Tx_D_Vd und Rx_D_Ack/Tx_D_Ack. Im synchronen Modus erzeugt das Modul zusätzlich einen 60MHz-Takt, der über den Pin Syn_Clk zur Verfügung gestellt wird. Die expliziten Ablaufprotokolle sind in Anhang A erläutert.

Abhängig vom Netzwerkverkehr kann eine Datenrate von bis zu 4 MByte/s erreicht werden.

Konfiguration der Netzwerkparameter:

Zur Anpassung der Modulparameter steht auf der Homepage des Herstellers ein Java-basiertes Tool zur Verfügung, mit dem alle Netzwerkeinstellungen (MAC, IP, Port, Subnetz, Gateway) sowie der Modus des Nutzerinterfaces konfiguriert werden können.

Ablauf: Nach dem Anlegen der Versorgungsspannung sowie nach Beendigung jeder Kommunikationsverbindung wartet das Modul für ca. 4 Sekunden auf einen Verbindungsversuch durch das Konfigurationstool. Nur in dieser Zeit ist eine Änderung der eingestellten Parameter möglich. Kommt es zu keiner Konfigurationsverbindung werden die zuletzt durch den Nutzer gespeicherten Parameter geladen. Anschließend erwartet das Modul eingehende Datenverbindungen.

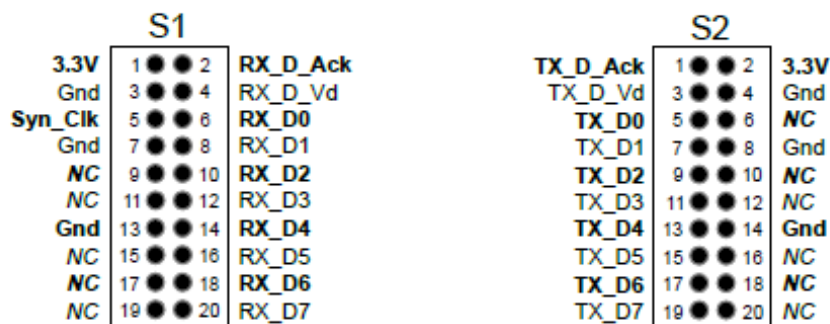
Zur Konfiguration wartet das Modul immer auf der festen IP-Adresse 192.168.0.100. Für eine Parameteränderung muss sichergestellt sein, dass diese Netzwerkadresse für das Tool erreichbar ist. (Hinweis: Diese unveränderliche IP wird nur zur Parametrierung verwendet und ist unabhängig von den durch den Nutzer individuell einstellbaren Parametern.)

Konfigurationstool:

Das Programm zur Parametrierung ist Java-basiert und wird als ausführbare jar-Datei bereitgestellt.

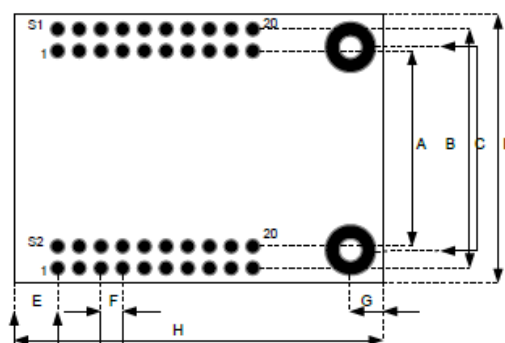
- Über den „Start Connection“-Button wird ein Verbindungsaufbau zum Modul initiiert. Ist dieser erfolgreich, werden die aktuell gespeicherten Nutzerparameter ausgelesen und in den Feldern der GUI dargestellt.
- In der Konsole auf der rechten Seite werden aktuelle Informationen ausgegeben.
- In den Textfeldern auf der linken Seite können die Einstellungen angepasst werden.
- Über den „Set new Values“-Button werden die Parameter aus den Textfeldern an das Modul übertragen. Ist der Vorgang erfolgreich, wird dieses in der Konsole auf der rechten Seite der GUI bestätigt.
- Über den „Disconnect“-Button wird die Verbindung zum Modul abgebaut und das Modul neu gestartet.

Pinbelegungen:



Abmessungen:

| Symbol | Maß (mm) |
|--------|----------|
| A | 22.86 |
| B | 23.72 |
| C | 27.94 |
| D | 31.00 |
| E | 5.08 |
| F | 2.54 |
| G | 3.63 |
| H | 42.93 |



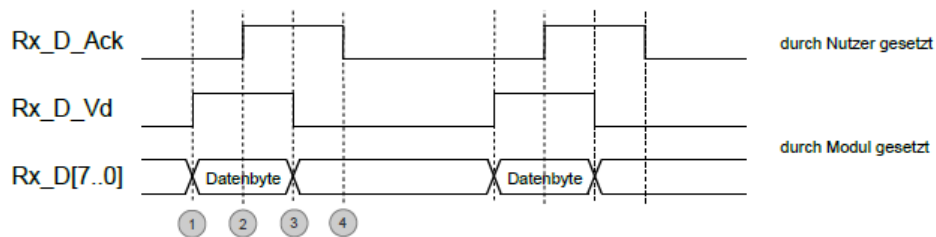
Anhang A: Zeitverläufe des Nutzer-Interfaces

Asynchroner Modus

Der asynchrone Datenaustausch wird über ein Handshake-Verfahren realisiert. Somit kann auf die Einhaltung zeitlicher Rahmenbedingungen für das Setzen der Kommunikationssignale verzichtet werden. Entscheidend ist lediglich die korrekte Abfolge der Signale, die Zeitdauer zwischen den aufeinander folgenden Wertzuweisungen ist irrelevant.

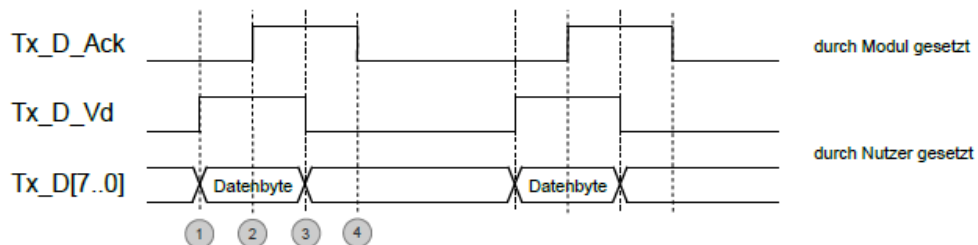
- Empfangen

- 1) Liegt ein neues Datenbyte vor, wird dieses durch das Modul auf die Rx_D Pins gelegt und Rx_D_Vd auf High gezogen.
- 2) Hat der Nutzer das Datenbyte verarbeitet, zieht er Rx_D_Ack auf High.
- 3) Das Modul erkennt das gesetzte Acknowledge-Signal und zieht Rx_D_Vd zurück auf Low.
- 4) Wenn das Valid-Signal auf Low liegt muss auch der Nutzer Rx_D_Ack zurück auf Low ziehen, bevor ein neues Datenbyte gelesen werden kann.



- Senden

- 1) Soll ein neues Datenbyte gesendet werden, wird dieses durch den Nutzer auf die Tx_D Pins gelegt und Tx_D_Vd auf High gezogen.
- 2) Hat das Modul das Datenbyte verarbeitet, zieht es Tx_D_Ack auf High.
- 3) Nach der Detektion des gesetzten Acknowledge-Signals zieht der Nutzer Tx_D_Vd zurück auf Low.
- 4) Das Modul zieht Tx_D_Ack ebenfalls zurück auf Low, im Anschluss kann ein neues Datenbyte gesendet werden.



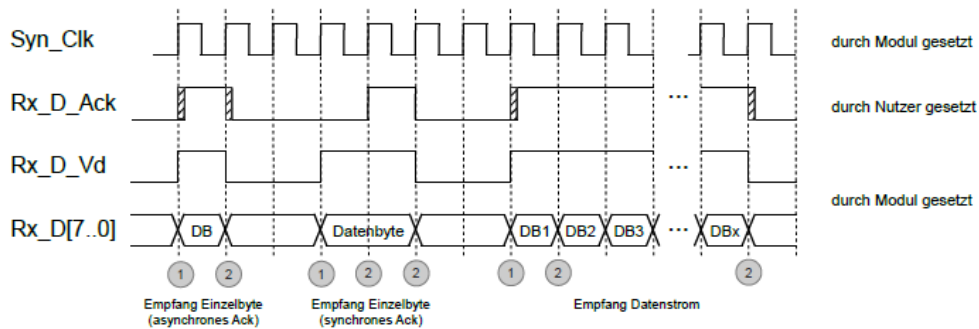
Synchroner Modus

Im synchronen Modus erfolgt die Verarbeitung der Daten auf Basis eines 60MHz-Taktes, welcher vom Modul erzeugt und auf dem Syn_Clk Pin zur Verfügung gestellt wird.

- Empfangen

- 1) Liegt ein neues Datenbyte vor, wird dieses durch das Modul auf die Rx_D Pins gelegt und mit der positiven Taktflanke Rx_D_Vd auf High gezogen.
- 2) Mit der nächsten Flanke wird das durch den Nutzer erzeugte Rx_D_Ack Signal vom Modul ausgewertet. Liegt Rx_D_Ack auf Low, bleibt das ausgegebene Datenbyte und Rx_D_Vd gesetzt. Liegt Rx_D_Ack auf High, wird, wenn vorhanden, das nächste empfangene Datenbyte ausgegeben und das Rx_D_Vd auf High belassen. Sind keine weiteren Daten vorhanden, wird Rx_D_Vd wieder auf Low gezogen.

Der höchste Datendurchsatz wird erreicht, wenn das Signal Rx_D_Ack durch den Nutzer asynchron erzeugt wird (z.B. als direktes Feedback des Rx_D_Vd). Eine synchrone Erzeugung mit den Taktflanken ist ebenso möglich, hat allerdings den Nachteil einer verzögerten Signaldetektion.



- Senden

- 1) Soll ein Datenbyte gesendet werden, wird es vom Nutzer auf die Tx_D Pins gelegt und mit dem Takt das Tx_D_Vd auf High gezogen.
- 2) Zur nächsten positiven Taktflanke muss das vom Modul asynchron erzeugte Tx_D_Ack ausgewertet werden. Liegt Tx_D_Ack auf Low, muss das zu sendende Byte sowie das Valid Signal weiterhin gehalten werden, da die Verarbeitung durch das Modul noch nicht abgeschlossen wurde. Liegt Tx_D_Ack auf High, wurden die Daten erfolgreich verarbeitet. Soll ein weiteres Datenbyte gesendet werden, kann dieses direkt gesetzt und das Tx_D_Vd auf High belassen werden. Sind keine weiteren Daten zu senden, wird das Tx_D_Vd wieder auf Low gezogen.

