

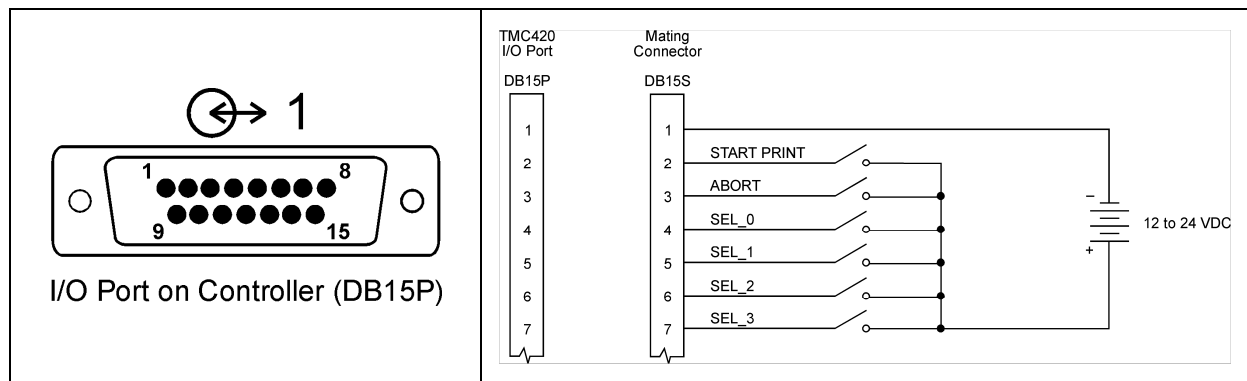
Kommunikation HOST → TMC420 Controller

Sofern möglich, sollte zusätzlich neben der seriellen Kommunikation (RS232 bzw. RS485) ein Signalaustausch mittels digitaler E/A erfolgen (24 VDC). Dieses ermöglicht eine Echtzeitüberwachung des Markerstatus. Über diese E/A Signale können folgende Zustände bzw. Befehle übermittelt werden:

Eingänge am TMC420 Controller:

- START PRINT (Prägestart von SPS)
- ABORT (Abbruch von SPS)
- SEL_0
- SEL_1
- SEL_2
- SEL_3 (Die SELECT Eingänge können verwendet werden, um bis zu 15 ($2^4-1=15$) Prägedateien mittels binärer Kombination extern auszuwählen. Sind alle SELECT Eingänge deaktiviert, so wird keine Datei geladen und die aktuell bzw. zuvor geladene wird verwendet. Der SEL_3 Eingang kann softwareseitig umgeschaltet werden, so dass dieser für die externe Ansteuerung des Online-Befehls verwendet werden kann. Der Markierkopf führt bei Ansteuerung eine Referenzfahrt durch und wird Online gesetzt. Bei Verwendung des SEL_3 Einganges als ONLINE Eingang, verringert sich die Anzahl der extern auswählbaren Prägedateien von 15 auf 7 ($2^3-1=7$).

Die Eingänge reagieren auf eine steigende Flanke und müssen daher nur per Impuls (ca. 500 ms) angesteuert werden.



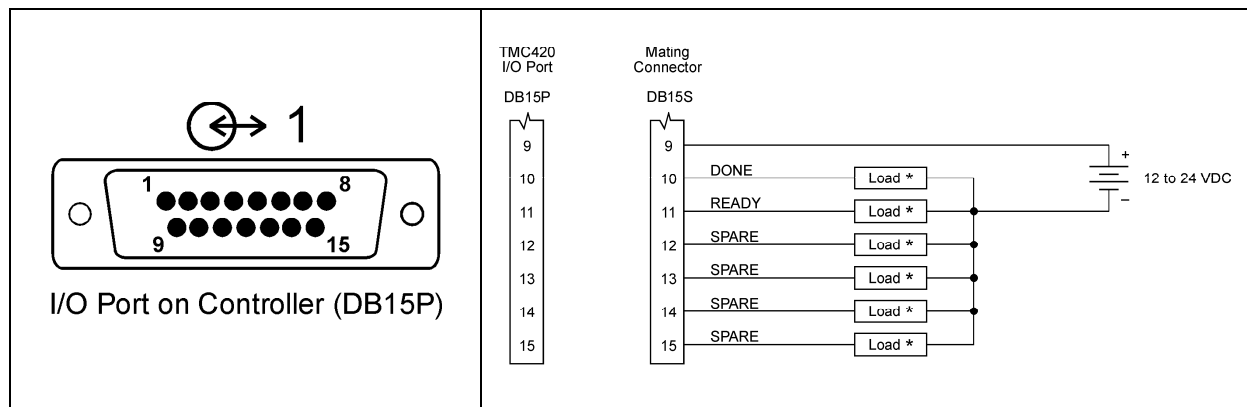
E/A Port am TMC420

Digitale Eingänge am TMC420

Nach dem Auswählen einer Prägedatei benötigt der TMC420 Controller eine entsprechende Zeit, um die Datei in den Speicher zu laden. Während des Ladevorganges wird der READY Ausgang auf "low" gesetzt und anschließend wieder "high" gesetzt. Die dafür benötigte Zeit ist abhängig von der Größe der zu ladenden Prägedatei.

Ausgänge am TMC420 Controller:

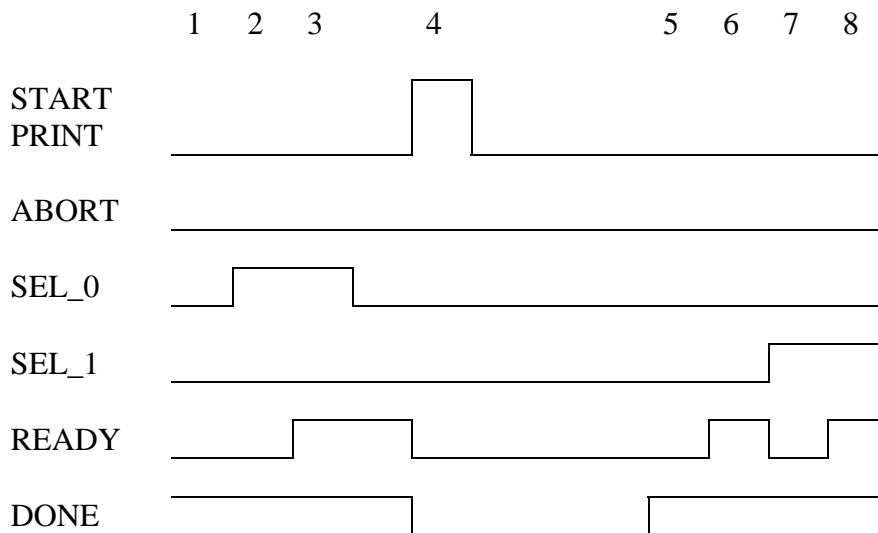
- **READY** (Bereit vom TMC420). Das READY Signal wird unter folgenden Bedingungen ausgegeben: das System ist Online **und** es ist eine Prägedatei mit enthaltenen Daten geladen **und** der Marker prägt nicht bzw. verarbeitet aktuell keine Prägedatei. Ist eine der drei Bedingungen nicht erfüllt, so ist der Ready Ausgang nicht gesetzt. Wird eine Prägedatei entweder über SELECT Eingänge oder mittels serieller Datenkommunikation (einschließlich Profibus), so wird während des Ladevorgangs im TMC420 Controller das READY Signal auf "low" gesetzt. Sobald der Ladevorgang abgeschlossen ist, wird das READY Signal wieder auf "high" gesetzt.
- **DONE** (Fertig vom TMC420). Das DONE Signal ist immer dann gesetzt, wenn der Marker nicht prägt.



E/A Port am TMC420

Digitale Ausgänge am TMC420

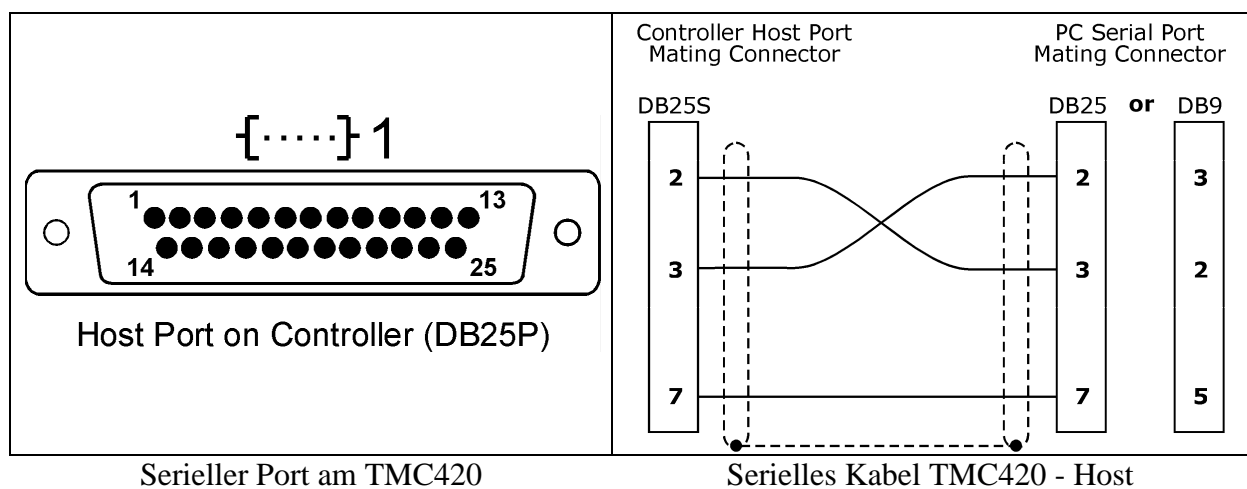
Zeitdiagramm Ein- und Ausgänge (Beispiel):



1. TMC420 wird eingeschaltet und automatisch Online gesetzt (inkl. Referenzfahrt).
2. Die Datei PAT01 wird über den Eingang SEL_0 geladen (Impulsdauer ca. 500 ms).
3. Nachdem die Datei PAT01 komplett im Speicher geladen wurde, wird das READY Signal ausgegeben.
4. Der Start wird von der SPS gesendet. READY und DONE werden auf "low" gesetzt. Der Markiervorgang ist aktiv. Der START PRINT Eingang kann unmittelbar nachdem READY und DONE auf "low" gesetzt wurden zurückgenommen werden (er sollte nicht während des gesamten Markierprozesses auf "high" gesetzt sein).
5. Nachdem der letzte Markierpunkt geprägt wurde, wird der DONE Ausgang auf "high" gesetzt.
6. Nachdem der Marker seine "Parkposition" erreicht hat, wird auch der READY Ausgang auf "high" gesetzt.
7. Die Datei PAT02 wird mittels SEL_1 Eingang geladen (Impulsdauer ca. 500 ms). Hierbei wird der READY Ausgang während des Ladeprozesses auf "low" gesetzt.
8. Der Ladevorgang ist abgeschlossen und der READY Ausgang wird wieder auf "high" gesetzt.

RS232-Schnittstelle:

Die serielle RS232-Schnittstelle wird meistens zur Kommunikation mit externen Datenquellen (z. B. Host-PCs, Terminals oder Barcodescanner) verwendet. Die RS232-Schnittstelle kann mittels Extended oder Programmable Protokoll kommunizieren. Das Extended Protokoll wird grundsätzlich bei Einsatz eines Host verwendet, da dieses Protokoll eine bidirektionale Kommunikation ermöglicht. Nach jedem Erhalt eines Datenstrings sendet der TMC420 eine Antwort an den Host (siehe weiter unten). Das Programmable Protokoll bietet nur eine unidirektionale Kommunikation (Daten werden nur erhalten, es erfolgt keine Rückmeldung des TMC420) und wird hauptsächlich in Verbindung mit Barcodescannern verwendet.



Das System unterstützt XON/XOFF Handshake. Wenn das System nicht bereit ist, serielle Übertragung durchzuführen, so wird XOFF gesendet. Ist der serielle Port verfügbar, so wird XON gesendet. Im umgekehrten Fall unterbricht das System die Datenübertragung bei Erhalt eines XOFF und führt die Übertragung fort, sobald ein XON erhalten wird.

Extended Datenprotokoll:

Der Aufbau des Datenstrings beim Senden an den TMC420 Controller hat folgenden grundsätzlichen Aufbau:

<SOH>Typ<STX>DATEN<ETX>[BCC]<CR>

mit

- <SOH> = Start of Header (= 01_{hex})
- Typ = Messagetyp (bestimmt den Inhalt der Nachricht, siehe weiter unten)
- <STX> = Start of Text (= 02_{hex})
- DATEN = Daten (abhängig vom Messagetyp)
- <ETX> = End of Text (= 03_{hex})
- [BCC] = Optionaler Block Check Code (Prüfsumme), generiert aus der Addition von Messagetyp und Daten
- <CR> = Carriage Return (= 0D_{hex})

Die Antwort vom TMC420 Controller hat grundsätzlich folgendes Format:

<SOH>Typ<ACK><STX>DATEN<ETX>[BCC]<CR> bzw.
<SOH>Typ<NAK><STX>DATEN<ETX>[BCC]<CR>

mit

<SOH>	=	Start of Header (= 01 _{hex})
Typ	=	Message Typ (bestimmt den Inhalt der Nachricht, siehe weiter unten)
<ACK>	=	Acknowledge (= 06 _{hex}) bei erfolgreicher Kommunikation
<NAK>	=	Negative Acknowledge (= 15 _{hex}) bei nicht erfolgreicher Kommunikation
<STX>	=	Start of Text (= 02 _{hex})
DATEN	=	Daten (abhängig vom Message Typ)
<ETX>	=	End of Text (= 03 _{hex})
[BCC]	=	Optional Block Check Code (Prüfsumme), generiert aus der Addition von Message Typ und Daten
<CR>	=	Carriage Return (= 0D _{hex})

Message Typen:

1. Laden einer Datei

Es wird hierzu der Message Typ 'P' (Zeichen 'P' = 50_{hex}) verwendet. Mit Hilfe dieses Message Typs wird eine Datei, welche im TMC420 gespeichert ist, zum Markieren geladen. Der Sendestring an den TMC420 hat folgenden Aufbau:

<SOH>P<STX>DATEINAME<ETX>[BCC]<CR>

mit

P	=	Message Typ 'P' (= 50 _{hex})
DATEINAME	=	Name der zu ladenden Prägedatei (muss bereits im TMC420 Controller vorhanden und gespeichert sein)

Die Antwort vom TMC420 Controller lautet:

<SOH>P<ACK><STX><ETX>[BCC]<CR>

Achtung: Der Ladevorgang der Prägedatei benötigt je nach Größe der Datei eine gewisse Zeit. Mit Erhalt der Antwort wird lediglich sichergestellt, dass der TMC420 die Daten einwandfrei erhalten hat (bei <ACK> in der Antwort), die Datei jedoch noch nicht zum Verarbeiten in den Speicher geladen wurde. Hierzu muss der digitale READY Ausgang (siehe oben) ausgewertet

werden. Dieser wird beim Anstoß des Ladevorganges auf "low" gesetzt. Nach dem kompletten Laden in den Speicher wird der READY Ausgang wieder auf "high" gesetzt. Wird der READY Ausgang nicht ausgewertet, so muss hier eine Wartezeit programmiert werden. Diese Wartezeit sollte zwischen 500 ms (für eine kleine Datei) und 5 s (für eine große Datei mit vielen Textfeldern) betragen.

2. Prägedaten in variables Feld senden

Hierzu wird der Messagetyt 'V' (Zeichen 'V' = 56_{hex}) verwendet. Mit Hilfe dieses Typs werden die gesendeten Daten an die Position des variablen Platzhalters geschrieben. Die Daten werden hier nur im Platzhalter aktualisiert, die Prägedatei selbst bleibt unberührt. Der Sendestring an den TMC420 hat folgenden Aufbau:

```
<SOH>V<STX>NNDATEN<ETX>[BCC]<CR>
```

mit

V	=	Messagetyt 'V' (= 56 _{hex})
NN	=	Zweistellige Feld- bzw. Zeilennummer (z. B. 01 oder 02)
DATEN	=	Prägedaten für Zeile FF

Die Antwort vom TMC420 Controller lautet:

```
<SOH>V<ACK><STX><ETX>[BCC]<CR>
```

Anmerkung: Es sollte hier eine geringe Wartezeit von ca. 200-300 ms eingesetzt werden, bevor weitere Kommunikationen mit dem TMC420 erfolgen.

3. Prägedaten in Querybuffer senden

Hierzu wird der Messagetyt 'Q' (Zeichen 'Q' = 51_{hex}) verwendet. Mit Hilfe dieses Typs werden die gesendeten Daten an die Position des Querybuffers 1 bis 3 geschrieben. Die Daten werden hier nur im Platzhalter aktualisiert, die Prägedatei selbst bleibt unberührt. Die Platzhalter für den Querybuffer können in verschiedenen Prägedateien verwendet werden und somit können die im Buffer enthaltenen Daten global verwendet werden. Der Sendestring an den TMC420 hat folgenden Aufbau:

```
<SOH>Q<STX>NNDATEN<ETX>[BCC]<CR>
```

mit

Q	=	Messagetyt 'Q' (= 51 _{hex})
NN	=	Zweistellige Nummer des Querybuffers (01 bis 03)
DATEN	=	Prägedaten für Zeile Buffer NN

Die Antwort vom TMC420 Controller lautet:

<SOH>Q<ACK><STX><ETX>[BCC]<CR>

4. Fehlerstatus des TMC420 abfragen

Hierzu wird der Messagetyp 'S' (Zeichen 'S' = 53_{hex}) verwendet. Mit Hilfe dieses Typs kann der Fehlerstatus des TMC420 Controllers abgefragt werden. Der Sendestring an den TMC420 hat folgenden Aufbau:

<SOH>S<STX><ETX>[BCC]<CR>

mit

S = Messagetyp 'S' (= 53_{hex})

Anmerkung: Bei diesem Messagetyp werden keine Daten gesendet.

Die Antwort vom TMC420 Controller lautet:

<SOH>S<ACK><STX>STATUSDATEN<ETX>[BCC]<CR>

Folgende STATUSDATEN sind möglich:

STATUSDATEN vom TMC420	FEHLERTYP Angezeigte Fehlermeldung(en)
0000	(kein Fehler)
0001	ONLINE_ERROR Timeout... Cannot Find Home Position! <i>[Zeitüberschreitung... Referenzposition kann nicht gefunden werden!]</i>
0002	PATTERN_LOAD_ERROR Error Loading Pattern (Host)! <i>[Fehler beim Laden der Datei (Host)]</i> Pattern Not Found (Host) : <pattern_name> <i>[Datei nicht gefunden (Host) : <Dateiname>]</i>
0004	DISALLOWED_NO_PATTERN No Pattern is Loaded! <i>[Keine Datei geladen!]</i> No Pattern is Loaded for Print! <i>[Keine Datei zum Prägen geladen!]</i> No Pattern Loaded. Pre-position What? <i>[Keine Datei geladen. Was vorpositionieren?]</i>
0008	DISALLOWED_OFFLINE Disallowed! Marker is Offline! <i>[Nicht erlaubt! Marker ist offline!]</i> Denied! Marker is Offline! <i>[Abgelehnt! Marker ist offline!]</i> Cannot Park on Load! Marker is Offline! <i>[Kann nicht Parken beim Laden! Marker ist offline!]</i>
0010	PATTERN_FIELD_ERROR Variable Text Not Found! <i>[Variablen Text nicht gefunden!]</i> Text / ArcText / Matrix Field Not Found! <i>[Text- / Bogentext- / Matrix-Feld nicht gefunden!]</i>
0020	MARKER_ABORTED_ERROR Marker Has Been Aborted! <i>[Marker wurde abgebrochen!]</i>

0040	(not used) [(nicht verwendet)]
0080	PIX_OUT_OF_RANGE_ERROR Pattern Pixel Range Error! [Datei Bereichsfehler!] Field <number> – Range Error! [Feld <Nummer> - Bereichsfehler!]
0100	RAM_ERROR Ram Error... Default Parameters Loaded! [Ram Fehler... Standardparameter geladen!] Ram Error... Existing Patterns Erased! [Ram Fehler... Vorhandene Dateien gelöscht!]
0200	SN_RANGE_ERROR Error! Serial Number Range Warning! [Fehler! Warnung Seriennummernbereich!]
0400	(not used) [(nicht verwendet)]

5. Löschen des Fehlerstatus

Wird nach dem Senden der Fehlerstatusnachricht (Typ 'S') eine Antwort vom TMC420 erhalten, welche Statusdaten enthält, die verschieden von '0000' sind, so muss anschließend der Fehlerstatus mit Hilfe des Messagetyp 'C' (= 43_{hex}) gelöscht werden. Der Sendestring an den TMC420 hat folgenden Aufbau:

<SOH>C<STX>STATUSDATEN<ETX>[BCC]<CR>

mit

C = Messagetyp 'C' (= 43_{hex})
STATUSDATEN = Daten aus vorheriger Antwort des TMC420 Controllers vom Typ 'S'

Die Antwort vom TMC420 Controller lautet:

<SOH>C<ACK><STX><ETX>[BCC]<CR>

Informationen zu weiteren Messagetypen finden Sie im TMC420 Installation/Maintenance Manual.

Ablaufbeispiel zum Datentransfer von Daten für zwei Prägefelder:

1. Markiersystem wird eingeschaltet (automatisches Online Setzen ist aktiviert). DONE Ausgang ist "high", READY Ausgang ist "low", da noch keine Datei geladen ist.
2. Der Eingang SEL_0 wird über die SPS auf "high" gesetzt. Signaldauer ca. 500 ms. Der READY Ausgang bleibt bzw. wird auf "low" gesetzt.
3. Der Ladevorgang im TMC420 ist abgeschlossen und der READY Ausgang wird auf "high" gesetzt.
4. Über die serielle Schnittstelle werden die Prägedaten für die erste Zeile gesendet:
<SOH>V<STX>01ABCDEFG<ETX>[BCC]<CR>
5. Der TMC420 antwortet über die Schnittstelle mit:
<SOH>V<ACK><STX><ETX>[BCC]<CR> bei korrekt erhaltenen Daten, ansonsten mit:
<SOH>V<NAK><STX><ETX>[BCC]<CR>
6. Eine kurze Wartezeit (Refresh) von 200-500 ms wird für die Aktualisierung der Daten im Speicher des TMC420 benötigt.
7. Der Host sendet die Daten für die zweite Zeile über die serielle Schnittstelle an den TMC420:
<SOH>V<STX>02UVWXYZ<ETX>[BCC]<CR>
8. Der TMC420 antwortet über die Schnittstelle mit:
<SOH>V<ACK><STX><ETX>[BCC]<CR> bei korrekt erhaltenen Daten, ansonsten mit:
<SOH>V<NAK><STX><ETX>[BCC]<CR>
9. Eine kurze Wartezeit (Refresh) von 200-500 ms wird für die Aktualisierung der Daten im Speicher des TMC420 benötigt.
10. Die SPS sendet über digitalen Ausgang ein "high" Signal an den START PRINT Eingang des TMC420, um den Prägevorgang zu starten.
11. Der READY und der DONE Ausgang des TMC420 werden auf "low" gesetzt.
Gleichzeitig setzt die SPS das Signal am START PRINT Eingang zurück auf "low".
12. Der Marker prägt den vorgegebenen Text.
13. Nach Prägen des letzten Prägepunktes wird der DONE Ausgang wieder "high" gesetzt.
14. Nach Erreichen der Parkposition wird der READY Ausgang des TMC420 wieder auf "high" gesetzt.
15. Nach Entnahme und Beladen eines Werkteils weiter bei 4. (falls sich die Prägedatei nicht ändert, sonst weiter bei 2.)