



www.qfix.de

qfix ASCII-Protokoll

Referenz

Inhaltsverzeichnis

1	Einleitung	3
1.1	Referenz	3
2	qfix ASCII Protokoll	4
2.1	Verbindung und Zeichensatz	4
2.2	Startup Meldung	5
2.3	Befehls-Syntax	6
2.4	Befehls-Referenz	8
2.5	Liste der Variablen und Konstanten	11
3	Beispiele	12
3.1	LEDs an und ausschalten	12
3.2	Motoren ansteuern	12
3.3	Power-Ausgänge an und ausschalten	13
3.4	Button abfragen	13
3.5	Digitaleingang abfragen	13
3.6	Analogeingang abfragen	13

Kapitel 1

Einleitung

Das qfix ASCII-Protokoll ist ein Verbindungsprotokoll zwischen PC-Programmen und Microcontrollern für Robotik-Anwendungen. Es wird typischerweise benutzt, um ein Controllerboard mit Motoren oder anderen Aktuatoren zu steuern und Sensordaten von analogen oder digitalen Eingängen abzufragen.

Das Protokoll ist bewusst sehr einfach gehalten, um einen schnellen Einstieg und einen möglichst universellen Einsatz zu ermöglichen. Es handelt sich daher auch um ein reines ASCII-Protokoll, d.h. sämtliche Daten werden in lesbarer Form verschickt. Hierdurch wird eine einfache Daten-Ein- und -Ausgabe über ein ASCII-Terminal ermöglicht. Desweiteren kann das Controllerboard leicht vom PC aus in einer beliebigen Programmiersprache angesprochen werden.

Das qfix ASCII-Protokoll wird beispielsweise von der PC-Software *qfixControl* verwendet. Für die qfix Boards MiniBoard und SoccerBoard, sowie den USB-Stick existieren fertige Client-Programme. Diese lassen sich per *qfixControl* auf die Controllerboards laden.

1.1 Referenz

Dieses Dokument beschreibt das qfix ASCII-Protokoll Version 1.

Die Referenz bezieht sich auf die *qfixControl* CD Version 1.0.

Kapitel 2

qfix ASCII Protokoll

2.1 Verbindung und Zeichensatz

- Die Einstellungen für die serielle Schnittstelle (bzw. virtuelle serielle Schnittstelle über USB) sind wie folgt:
38400, 8N1 (38400 Baud, 8 Bit, kein Parity Bit, 1 Stopp Bit)
- Es werden nur lesbare ASCII Zeichen benutzt.
- Jede Zeile beinhaltet genau einen Befehl.
- Zur Markierung des Zeilen- bzw. Befehls-Endes wird (in beide Richtungen) das Zeichen `\n` (Dezimal 10, Hex 0A) benutzt.
- Gross-/Kleinschrift wird unterschieden.

2.2 Startup Meldung

Beim Starten des Boards kann dieses einmalig eine Beschreibung zum PC senden. Voraussetzung ist natürlich, dass das entsprechende PC-Programm die Beschreibung verarbeiten oder zumindest ignorieren kann. Jede Zeile wird wie üblich mit `\n` beendet. Die Beschreibung besteht aus zwei Gruppen:

Die erste Gruppe beschreibt die verwendete Hard- und Software sowie die vorhandenen I/Os des Boards (Siehe hierzu die `V`-Befehle weiter unten):

```
// <Name und Version des Programmes> //
V hw_name= <Board Name>
V sw_name= <Software Name>
V sw_version= <Software Version>
V num_buttons= <Anzahl Buttons>
V num_leds= <Anzahl LEDs>
V num_analog= <Anzahl Analogeingänge>
V num_digital= <Anzahl Digitaleingänge>
V num_motors= <Anzahl Motoranschlüsse>
V num_power= <Anzahl Poweranschlüsse>
V analog_auto=off/on
```

Es folgt eine zweite Gruppe, die den aktuellen Zustand der I/Os zum Startzeitpunkt beschreibt:

```
B0 0/1      (Zustand Button 0)
B1 0/1      (Zustand Button 1)
...
D0 0/1      (Zustand Digital 0)
...
M0 0        (Zustand Motor 0)
...
P0 0/1      (Zustand Power 0)
...
```

Die Punkte sollen andeuten, dass hier beliebig viele Einträge folgen können.

2.3 Befehls-Syntax

2.3.1 I/O Kommandos

Die Kommandos, mit denen sich die I/Os setzen und abfragen lassen entsprechen folgendem Aufbau:

<C>	<i>	<value>	\n
-----	-----	---------	----

Die Einträge besitzen hierbei folgende Bedeutung:

- <C> = Kommando (1 Byte)
- <i> = Index (1 Byte)
- Leerzeichen (1 Byte)
- <value> = Wert (Wert in Ascii-Zeichen)
- \n (1 Byte)

Bei einigen Befehlen (z.B. Abfragebefehle) entfällt die Übergabe eines Wertes. Das Kommando reduziert sich dann auf die folgende Form:

<C>	<i>	\n
-----	-----	----

Die möglichen Kommandos sind in Kapitel 2.4 aufgelistet.

2.3.2 Kommandos zur Variablenabfrage

Ein Board kann interne Variablen und Konstanten verwalten.

Der Wert einer Variablen oder Konstanten lässt sich folgendermaßen auslesen:

V	<key>	\n
---	-------	----

- V = Kommando V (1 Byte)
- Leerzeichen (1 Byte)
- <key> = Name der Variable oder Konstante (Länge beliebig)
- \n (1 Byte)

Mit folgendem Kommando lässt sich einer Variablen ein neuer Wert zuordnen:

V	<key>	=	<value>	\n
---	-------	---	---------	----

- V = Kommando V (1 Byte)
- Leerzeichen (1 Byte)
- <key> = Name der Variable oder Konstante (Länge beliebig)
- = = Zuweisungsoperator (1 Byte)
- <value> = Neuer Wert (Länge beliebig)

- \n (1 Byte)

Das Kommando V wird immer vom PC initiiert. Ausnahme: Startup Meldung (siehe 2.2).

Standardisierte Variablen- und Konstanten-Namen sind in Kapitel 2.5 aufgelistet.

2.3.3 Fehlermeldungen

Kann ein Kommando, das vom PC zum Board geschickt wird, nicht ausgewertet werden, sendet das Board den String `ERROR:` gefolgt von dem empfangenen Kommando und \n zurück.

2.4 Befehls-Referenz

Die folgenden Kommandos werden meist durch den PC initiiert und erhalten darauf eine Antwort vom Board. Um die Antwort des Boards dem gesendeten Befehl zuordnen zu können, enthält sie nicht nur die eigentlichen Antwortdaten, sondern wiederholt den entsprechenden Befehl (s.u.).

Alternativ kann das Board auch selbständig Nachrichten senden, wenn sich beispielsweise der Zustand eines Eingangs ändert.

2.4.1 ? - Info

Das Info Kommando `?` wird zur Abfrage der Board-Beschreibung und des Zustandes benutzt. Es liefert als Antwort die komplette Startup-Meldung wie in 2.2 beschrieben.

PC → Board:

?	\n
---	----

Ausnahme! Es wird lediglich ein Fragezeichen gefolgt von `\n` gesendet.

Board → PC: (Siehe 2.2)

Das Board gibt als Antwort die Beschreibung des Boards und der aktuellen Zustände zurück.

2.4.2 A - Analog

Das Kommando `A` wird zur Abfrage der Analogeingänge benutzt.

PC → Board:

A	<i>	\n
---	-----	----

Der Index `i` entspricht dem Index des Analogeingangs und kann die Werte 0 bis `num_analog-1` annehmen.

Board → PC:

A	<i>	<value>	\n
---	-----	---------	----

Das Board liefert als Ergebnis den Wert des Analog-Eingangs mit dem Index `i`. Der Wert `<value>` kann hierbei Wertezwischen 0 und 255 annehmen.

Hinweis: Falls die Variable `analog_auto` den Wert `on` besitzt, werden andauern die Werte sämtlicher Analogeingänge automatisch vom Board gesendet.

2.4.3 B - Button

PC → Board:

B	<i>	\n
---	-----	----

Der Index `i` entspricht dem Index des Buttons und kann die Werte 0 bis `num_buttons-1` annehmen.

Board → PC:

B	<i>	0 / 1	\n
---	-----	-------	----

Das Board liefert als Ergebnis den Zustand des Buttons mit dem Index `i`. Der zurückgelieferte Wert kann 0 oder 1 sein.

2.4.4 D - Digital

Das Kommando `D` wird zur Abfrage der Digitaleingänge benutzt.

PC → Board:

D	<i>	\n
---	-----	----

Der Index `i` entspricht dem Index des Digitaleingangs und kann die Werte 0 bis `num_digital-1` annehmen.

Board → PC:

D	<i>	0 / 1	\n
---	-----	-------	----

Das Board liefert als Ergebnis den Zustand des Digitaleingangs mit dem Index `i`. Der zurückgelieferte Wert kann 0 oder 1 sein.

2.4.5 L - LED

Das Kommando `L` wird zur Ansteuerung der LEDs benutzt.

PC → Board:

L	<i>	0 / 1	\n
---	-----	-------	----

Der Index `i` entspricht dem Index der LED und kann die Werte 0 bis `num_leds-1` annehmen.

Der übergebene Wert kann 0 oder 1 entsprechend AUS oder AN sein.

Board → PC:

L	<i>	0 / 1	\n
---	-----	-------	----

Das Board antwortet mit der selben Sequenz.

2.4.6 M - Motor

Das Kommando `M` wird zur Ansteuerung der Motoren benutzt.

PC → Board:

M	<i>	<speed>	\n
---	-----	---------	----

Der Index `i` entspricht dem Index des Motors und kann die Werte 0 bis `num_motors-1` annehmen.

Der übergebene Wert für `<speed>` muss im Bereich -255 bis 255 liegen.

Board → PC:

M	<i>	<speed>	\n
---	-----	---------	----

Das Board antwortet mit der selben Sequenz.

2.4.7 P - Power

Das Kommando `P` wird zur Ansteuerung der Power-Ausgänge benutzt.

PC → Board:

P	<i>	0 / 1	\n
---	-----	-------	----

Der Index `i` entspricht dem Index des Power-Ausgangs und kann die Werte 0 bis `num_power-1` annehmen.

Der übergebene Wert kann 0 oder 1 entsprechend AUS oder AN sein.

Board → PC:

P	<i>	0 / 1	\n
---	-----	-------	----

Das Board antwortet mit der selben Sequenz.

2.4.8 V - Value

Siehe Kapitel 2.3.2.

2.5 Liste der Variablen und Konstanten

Die folgenden Variablen und Konstanten sind in den qfix Boards definiert.

Konstanten

Folgende Konstanten sind fest vordefiniert. Die Werte können auch nicht verändert werden. Sie werden beim Start des Programms ausgegeben (siehe 2.2) und können jederzeit per `V`-Kommando abgefragt werden.

hw_name - Name des Controllerboards.

sw_name - Name der Software.

sw_version - Software-Version.

num_analog - Anzahl Analogeingänge.

num_buttons - Anzahl Buttons.

num_digital - Anzahl Digitaleingänge.

num_leds - Anzahl LEDs.

num_motors - Anzahl Motorausgänge.

num_power - Anzahl Power-Ausgänge.

Variablen

Folgende Variablen sind bereits vordefiniert. Sie werden beim Start des Programms ausgegeben (siehe 2.2) und können jederzeit per `V`-Kommando abgefragt und die Werte verändert werden.

analog_auto - Automatisches Senden der Analogwerte (`on` oder `off`)

Kapitel 3

Beispiele

Die folgenden Beispiele zeigen die Kommunikation zwischen PC und Controllerboard. Die erste Zeile enthält jeweils den Befehl vom PC zum Board, die zweite Zeile die Antwort des Boards. Jede Zeile endet mit \n.

3.1 LEDs an und ausschalten

Anschalten der LED 0:

```
L0 1
```

```
L0 1
```

Ausschalten der LED 0:

```
L0 0
```

```
L0 0
```

3.2 Motoren ansteuern

Anschalten des Motors 0 auf volle Leistung:

```
M0 255
```

```
M0 255
```

Anschalten des Motors 0 auf volle Leistung rückwärts:

```
M0 -255
```

```
M0 -255
```

Abschalten des Motors 0:

```
M0 0
```

```
M0 0
```

Abschalten aller Motoren:

```
M-
```

```
M-
```

3.3 Power-Ausgänge an und ausschalten

Anschalten des Power-Ausgangs 0:

```
P0 1
```

```
P0 1
```

Ausschalten des Power-Ausgangs 0:

```
P0 0
```

```
P0 0
```

3.4 Button abfragen

Abfragen des Buttons 0 (hier: gedrückt):

```
B0
```

```
B0 1
```

Abfragen des Buttons 1 (hier: nicht gedrückt):

```
B1
```

```
B1 0
```

3.5 Digitaleingang abfragen

Abfragen des Digitaleingangs 0 (hier: high):

```
D0
```

```
D0 1
```

Abfragen des Digitaleingangs 1 (hier: low):

```
D1
```

```
D1 0
```

3.6 Analogeingang abfragen

Abfragen des Analogeingangs 0 (hier: 178):

```
A0
```

```
A0 178
```

Anschalten der automatischen Analogwert-Abfrage:

```
V analog_auto=on
```

```
V analog_auto=on
```

```
A0 178
```

```
A1 23
```

A2 233

•
•
•



qfix robotics GmbH
Erich-Rittinghaus-Straße 2/2
89250 Senden

Tel.: 07307-800321
Fax.: 07307-800322
Mail: info@qfix.de
Web: www.qfix.de
