

Relaismatrix 4x4

Technische Dokumentation

V1.1
30.5.2017

Änderungsliste:

17.01.11	V1.0	Erstellung	Schneider
30.5.2017	V1.1	Update auf aktuelle HW	Schneider

Inhaltsverzeichnis

- 1 Funktionsbeschreibung.....2
- 2 Technische Daten:.....3
- 3 Ansteuerung.....4
 - 3.1 mit USB.....4
 - 3.1.1 USB unter Linux.....4
 - 3.1.2 USB unter Windows.....4
 - 3.1.3 Kommunikation mit der Relaismatrix4x4.....4
 - 3.2 mit JTAG-Boundary Scan.....5

1 Funktionsbeschreibung

Die Relaismatrix 4x4 besteht aus 16 Relais die als 4facher 1:4 Multiplexer (4 Kanäle C1-C4 mit jeweils 4 Relais K1-K4) verbunden sind. Es sind jeweils die Schließer von 4 Relais einseitig miteinander verbunden und zusammen mit den offenen Enden der Kontakte auf einen 25pol. D-Sub-Steckverbinder geführt. Durch externe Verbindung der einzelnen Schließer kann auch eine 4x4 Matrix realisiert werden. Angesteuert wird die Relaismatrix über USB oder JTAG-Testbus.

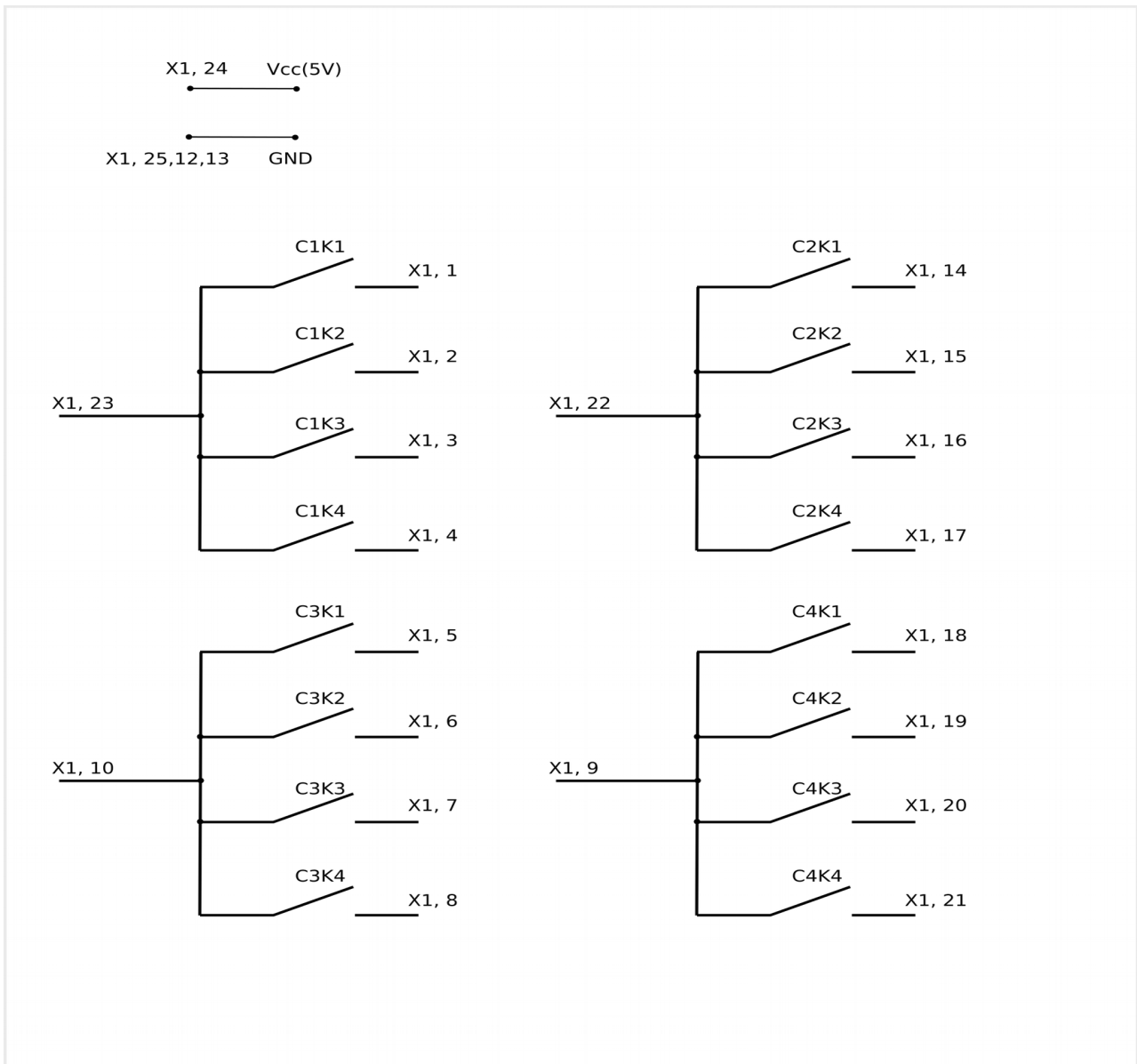
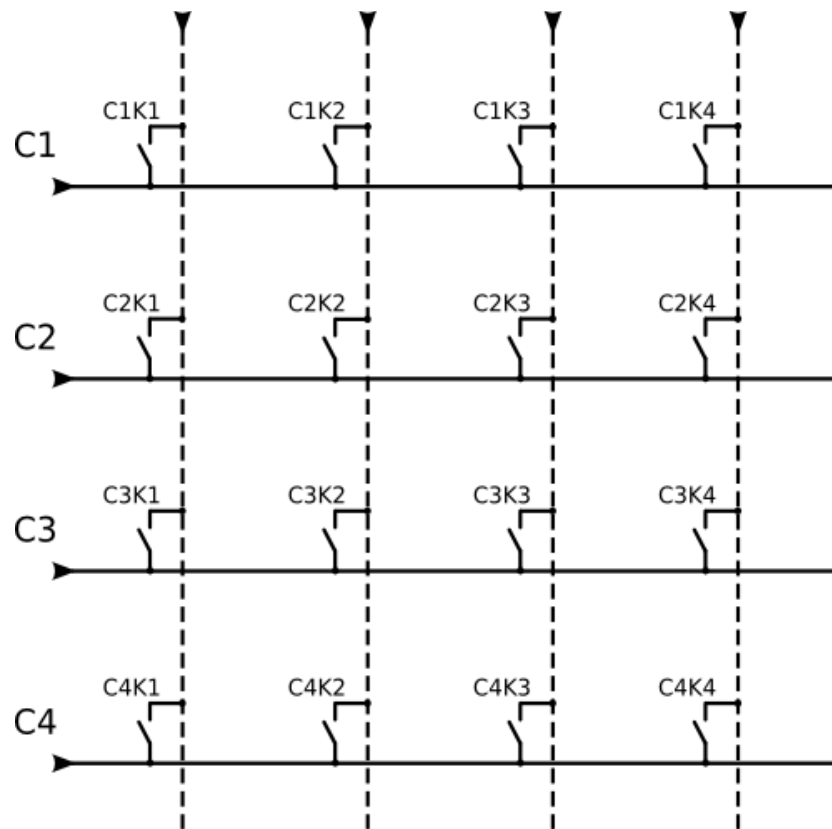


Abbildung 1: Kontaktplan Relaismatrix 4x4



gestrichelte Linien sind externe Verbindungen die nicht auf der Relaismatrix 4x4 vorhanden sind.

Abbildung 2: Schaltung als Matrix mit zusätzlichen Verbindungen

2 Technische Daten:

Parameter	Wert	Einheit
max. schaltbare Last AC	125V / 0,3	V / A
max. schaltbare Last DC	30V / 1A	V / A
max. Schaltspannung AC / DC	125 / 60	V / V
max. Schaltstrom	1	A
max. Übergangswiderstand pro Relaiskontakt	100	mΩ
min. elektrische Lebensdauer der Relaiskontakte	100.000	Schaltspiele
mechanische Lebensdauer der Relaiskontakte	50.000.000	Schaltspiele
Schaltzeit (nach Ende des Steuerbefehls)	< 10ms	ms
Abmessungen	80x100	mm

3 Ansteuerung

Die Relaismatrix4x4 kann über USB oder über den JTAG-Testbus angesteuert werden. Dadurch ist sie sehr flexibel in verschiedenen Umgebungen einsetzbar.

3.1 mit USB

Für die USB-Ansteuerung ist auf der Relaismatrix4x4 ein IC von FTDI verbaut. Der Hersteller stellt Treiber für Windows und Linux zur Verfügung.

3.1.1 USB unter Linux

Seit Kernel 2.4 ist ein passender Treiber integriert. Dieser erstellt beim Einstecken der Relaismatrix4x4 einen neuen Device „ttyUSBn“ (z.B. ttyUSB0 als 1. Device) im Verzeichnis /dev. Dieser Device lässt sich wie ein normaler serieller Port ansprechen.

3.1.2 USB unter Windows

Für Windows wird ein Treiber von FTDI benötigt. Dieser erstellt beim Einstecken der Relaismatrix4x4 einen virtuellen COM-Port. Die Nr. des COM-Ports lässt sich in der Systemsteuerung->Hardware Manager->Anschlüsse einstellen. Wenn der Port nicht zwihschendurch anderweitig belegt wird, bekommt die Relaismatrix4x4 immer den selben COM-Port zugewiesen.

3.1.3 Kommunikation mit der Relaismatrix4x4

Parameter für den COM-Port bzw ttyUSBn:

38400Bit/s , 8 Datenbits , No Parity, 1 Stopbit
keine Flusskontrolle

Um ein Relais zu schalten wird eine Zeichenkette in der Form „CnKnCn1Kn2\r“ über die serielle Schnittstelle an die Relaismatrix4x4 gesendet („\r“ bedeutet dabei Carriage Return CR,ASCII-Code hex: 0x0D). Dies ist mit jedem Terminal(Hyperterminal, minicom) oder anderweitig möglich.

Beispiel: „C1K1C2K3\r“ schaltet das erste Relais K1 im ersten Kanal C1 und das 3. Relais K3 im zweiten Kanal C2

Zum Abschalten aller Relais bzw. zum Rücksetzen der Relaismatrix4x4 wird einfach ein „\r“ gesendet.

Weitere Befehle:

Befehl	Beispielantwort	Beschreibung
*IDN?	'masla Relaismatrix 4x4 2 Channels, SW-Ver. 1.11, SNr: 10'	Identifizierung der Baugruppe mit Anzahl der erkannten Kanäle(C), Firmwareversion & Seriennummer
RELAIS:OPEN	-	Öffnet einzelne Relais z.B.: 'RELAIS:OPEN C1K1C8K2'
RELAIS:CLOSE	-	Schließt einzelne Relais z.B.: 'RELAIS:CLOSE C1K1C8K2'
RELAIS:STRING?	'C1K1C8K2\n'	Relaisstring der aktuell geschlossenen Relais

3.2 mit JTAG-Boundary Scan

Über den Steckverbinder SV1 kann der ATmega16 in den Boundary Scan-Modus versetzt werden. Mit einem geeigneten JTAG-Tool kann die SPI-Schnittstelle des ATmega nachgebildet werden um so die beiden Relaisreiber anzusteuern.