

Cupboard Locking System

Inhalt

Purpose oft the system	2
Hardware	2
Lock.....	2
Raspberry PI B+	3
Relais Card DAYPOWER RB-8/5V, 8-Channel	4
Software	4
Requirements	4
Code (Python).....	4
Bedienung.....	6
Administration RPI.....	7
Putty	7
Kommandos über Konsole	7
Automatisch nach Boot Script ausführen.....	8
LXDE Autostarts	8
Mit RC.LOCAL.....	8
WLAN über Kommandozeile einrichten	9
PSK generieren	9
PSK eintragen	9

Purpose of the system

The cupboard locking system can lock and unlock two doors (refrigerator and shelf) in the configured time. If the system is unlocked, the lamp is green, otherwise it is red.

Hardware

Lock



Digital Schublade Schloss Schubladenschloss Intelligent
Schrankschloss Lock RFID

(312847783764)

Anzahl: 2

✓ Zugestellt am **Mar 25, 2020**

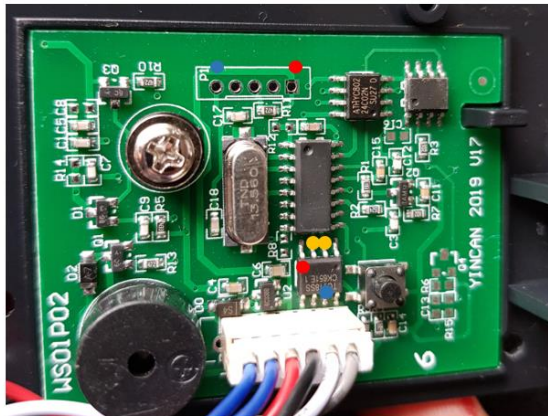
Sendungsnummer: **09445474864585**



Artikelpreis:

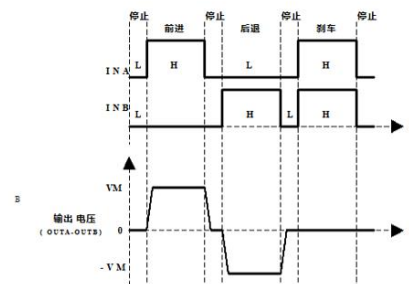
EUR 32,54

Motordriver TC118SS

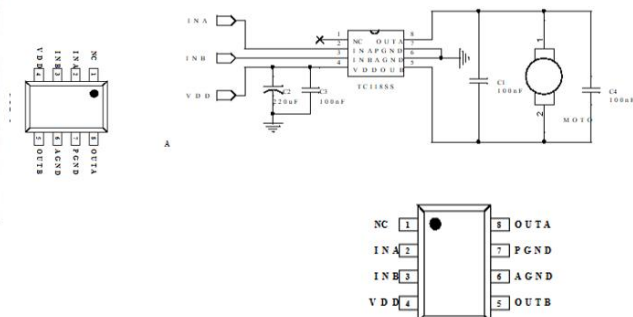


Auf: Brücke VDD zu VINB

Zu: Brücke VDD zu VINA

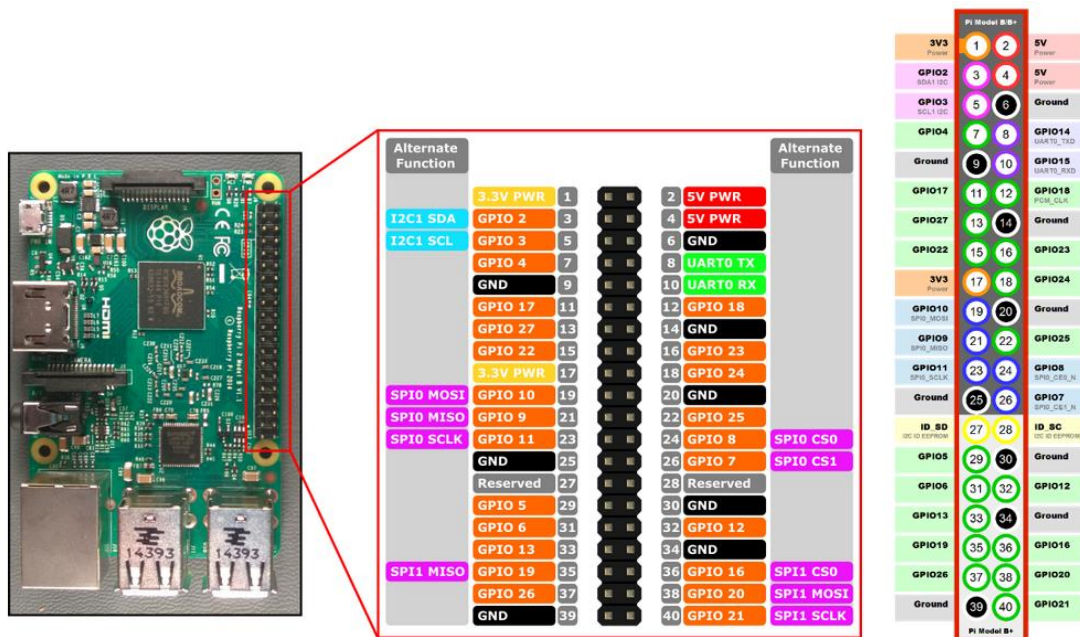


九、应用参考电路图与 PCB 布线指导



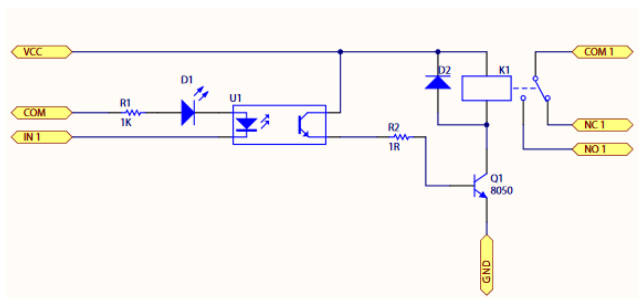
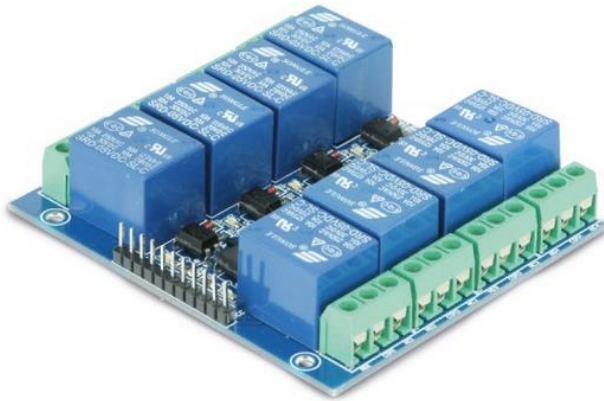
Designation	Wire color
3,3 V	gn
Lock	bu
Unlock	bu/wh
5 V	or/wh
GND	li/wh

Pinning



Funktion	GPIO	Relais	Leitungsfarbe	Kommentar
Licht rot	27	1	gn	
Licht grün	3	2	gn/ws	
Verriegeln Kühlschrank	4	3	bl	
Entriegeln Kühlschrank	17	4	bl/ws	
Verriegeln Regal	18	5	vi	
Entriegeln Regal	22	6	vi /ws	
Ton	23	7	or/ws	
Relais-Platine	5V		or	
Taster Signal	25			
Taster GND	GND			

Relais Card DAYPOWER RB-8/5V, 8-Channel



① Relaisausgänge

NO1...NO8	Normally Open (Schließerkontakt)
COM1...COM8	Common (Mittelkontakt)
NC1...NC8	Normally Closed (Öffnerkontakt)

② Eingangs-Stiftleiste

VCC	Versorgungsspannung für die Platine (+5 V)
GND	Masse für die Versorgungsspannung der Platine (-)
COM	Gemeinsamer Steuereingang (+)
IN1...IN8	Steuereingänge (-)

Software

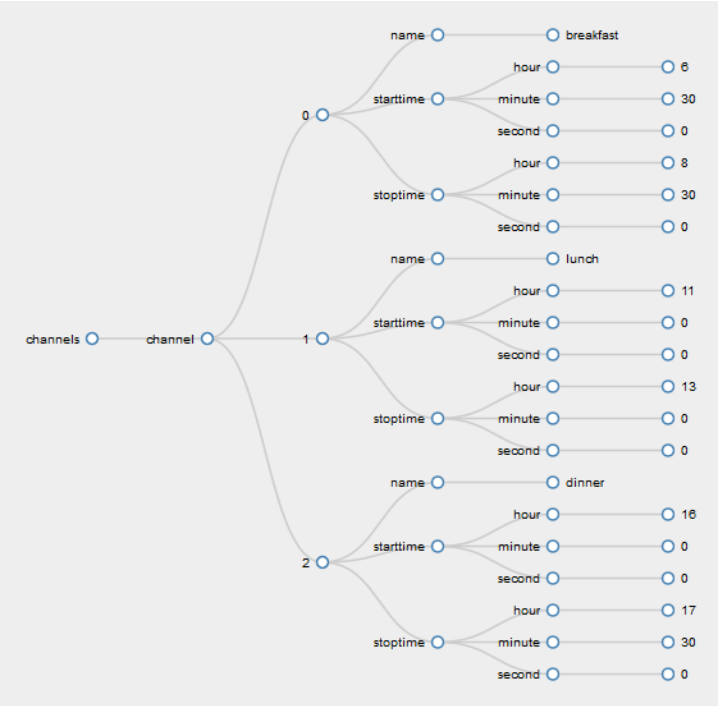
Requirements

1. Die Anlage muss mindestens 4 Timer besitzen.
2. Die Zeiten müssen über ein PC oder Smartphone einstellbar sein.
3. Die Uhr muss über ein PC oder Smartphone einstellbar sein.
4. Wenn die Zeit für Entriegelung erreicht ist, muss ein High-Signal für eine Sekunde eingestellt werden.
5. Wenn die Zeit für Verriegelung erreicht ist, muss ein Low-Signal für eine Sekunde eingestellt werden.

Code (Python)

See source code.

Data structure



```
object ▶ channels ▶ channel ▶ 0 ▶ starttime ▶ hour
  ▢ ▼ object {1}
  ** ▢ ▼ channels {1}
  ** ▢ ▼ channel [3]
  ** ▢ ▼ 0 {3}
  ** ▢   name : breakfast
  ** ▢   ▼ starttime {3}
  ** ▢     hour : 6
  ** ▢     minute : 30
  ** ▢     second : 0
  ** ▢   ▼ stoptime {3}
  ** ▢     hour : 8
  ** ▢     minute : 30
  ** ▢     second : 0
  ** ▢   ▼ 1 {3}
  ** ▢     name : lunch
  ** ▢     ▼ starttime {3}
  ** ▢       hour : 11
  ** ▢       minute : 0
  ** ▢       second : 0
  ** ▢     ▼ stoptime {3}
  ** ▢       hour : 13
  ** ▢       minute : 0
  ** ▢       second : 0
  ** ▢   ▼ 2 {3}
  ** ▢     name : dinner
  ** ▢     ▼ starttime {3}
  ** ▢       hour : 16
  ** ▢       minute : 0
  ** ▢       second : 0
  ** ▢     ▼ stoptime {3}
  ** ▢       hour : 17
  ** ▢       minute : 30
  ** ▢       second : 0
```

Bedienung

Entriegeln	Verriegeln
(6, 30, 0)	(7, 30, 0)
(11, 15, 0)	(12, 00, 30)
(16, 30, 0)	(17, 30, 0)

Administration RPI

Putty

WLAN: IP: 192.168.8.146 Port 22

Netzwerkkabel: IP: 192.168.8.142 Port 22

Login name: *

Password: *

```
pi@pi:~$ python Schrankverriegelungsanlage_v01_AOr.py
```

Beispiel:

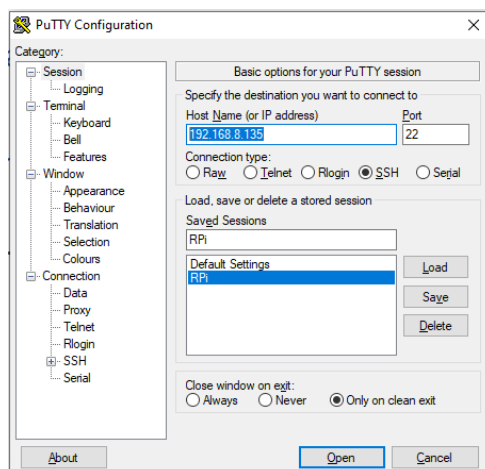
```
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BCM)

GPIO.setup(23, GPIO.OUT)
GPIO.setup(24, GPIO.IN)

for i in range(5):
    GPIO.output(23, GPIO.HIGH)
    time.sleep(0.5)
    GPIO.output(23, GPIO.LOW)
    time.sleep(0.5)

# Endlosschleife
while True:
    if GPIO.input(24) == 0:
        # Ausschalten
        GPIO.output(23, GPIO.LOW)
    else:
        # Einschalten
        GPIO.output(23, GPIO.HIGH)
```



Kommandos über Konsole

Datum und Zeit auslesen: datetime

Automatisch nach Boot Script ausführen

LXDE Autostarts

Raspbian kann wahlweise auf eine Kommandozeile oder direkt in die grafische Oberfläche booten, die Einstellung kann via **raspi-config** verändert werden. Der Standard Desktop bei Raspbian ist LXDE (**L**ightweight **X**11 **D**esktop **E**nvironment). Auch beim Starten bzw. Anmelden eines Benutzers am Desktop können Autostarts ausgeführt werden, beim Desktop hat man hierbei die Wahl das für einen einzelnen Benutzer oder für alle Benutzer auf dem System zu konfigurieren. Die Dateien sind jeweils in den folgenden Pfaden zu finden und können mit nano editiert werden

- Für alle Benutzer: `/etc/xdg/lxsession/LXDE-pi/autostart`
- Für einen bestimmten Benutzer: `~/.config/lxsession/LXDE/autostart`

```
sudo nano /etc/xdg/lxsession/LXDE-pi/autostart
```

Hier ein paar Beispiele für Autostarts: Ein Shell Script, der Browser Midori und ein Python Script

Shell

```
@lxpanel --profile LXDE-pi
```

```
@pcmanfm --desktop --profile LXDE-pi
```

```
@xscreensaver -no-splash
```

```
@python Schrankverriegelungsanlage_v01_AOr.py
```

```
point-rpi
```

Jede Zeile enthält ein Kommando das ausgeführt wird, wenn einem Befehl mit einem @ beginnt, wird dieser Befehl automatisch erneut ausgeführt wenn das Programm crashed.

Mit RC.LOCAL

```
sudo nano /etc/rc.local
```



```
python /home/pi/Schrankverriegelungsanlage_v01_AOr.py &
```

WLAN über Kommandozeile einrichten

PSK generieren

```
wpa_passphrase "*"
```

```
# reading passphrase from stdin
```

```
*
```

```
network={
```

```
    ssid="*"
```

```
    #psk="*"
```

```
    psk=*
```

```
}
```

Davor:

```
network={
```

```
    ssid="*"
```

```
    psk="*"
```

```
    key_mgmt=WPA-PSK
```

```
}
```

PSK eintragen

```
sudo nano /etc/wpa_supplicant/wpa_supplicant.conf
```

Reconfigure the interface with `wpa_cli -i wlan0 reconfigure`.

You can verify whether it has successfully connected using `ifconfig wlan0`. If the `inet addr` field has an address beside it, the Raspberry Pi has connected to the network. If not, check that your password and ESSID are correct.