

## 84X48 LCD DISPLAY

SBC-LCD84x48

## 1. ALLGEMEINE INFORMATIONEN

Sehr geehrter Kunde,  
vielen Dank, dass Sie sich für unser Produkt entschieden haben. Im Folgenden zeigen wir Ihnen, was bei der Inbetriebnahme und der Verwendung zu beachten ist.

Sollten Sie während der Verwendung unerwartet auf Probleme stoßen, so können Sie uns selbstverständlich gerne kontaktieren.

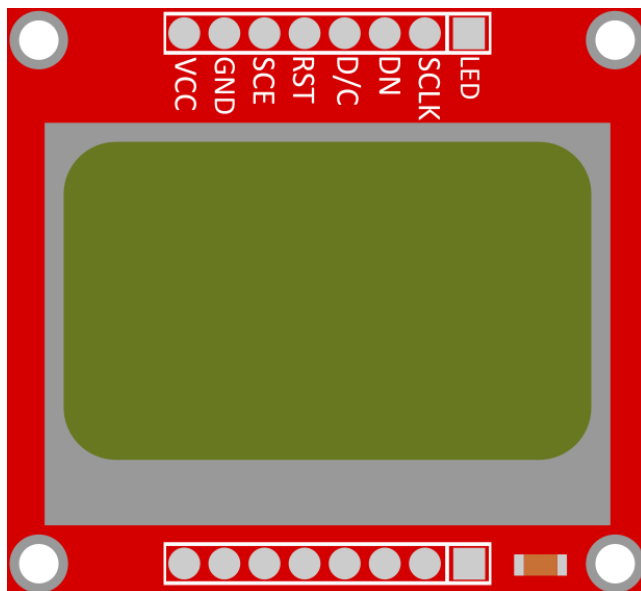
Das Display kann mit vielen gängigen Entwicklerboards wie Raspberry Pi oder Arduino verwendet werden, um beispielsweise Messwerte oder Messwertgrafiken direkt auf dem Mikrocontroller oder Einplatinencomputer anzuzeigen.

## 2. UNTERSCHIEDE ZWISCHEN DEN REVISIONEN

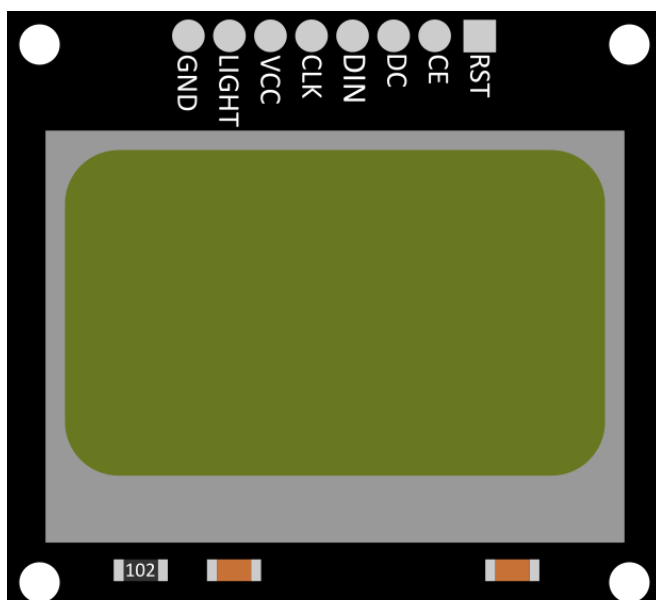
Bei diesem Produkt gibt es eine neue Revision, welche die vorherige Revision ersetzt. Diese Änderungen beinhalten das Pin Out, die LED und den Silkscreen.

Im Folgenden sehen Sie 2 Zeichnungen der verschiedenen Revisionen.

Revision 1



Revision 2



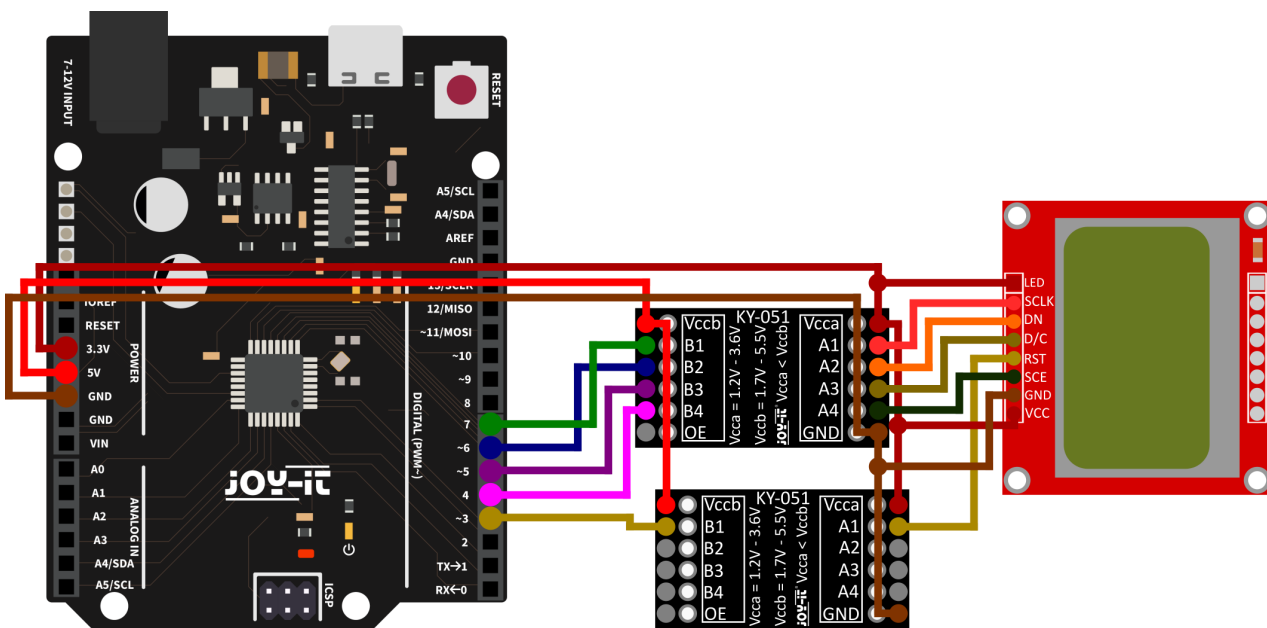
### 3. VERWENDUNG MIT EINEM ARDUINO

#### 3.1 Anschließen des Displays (REV 1)

Schließen Sie das Display, wie im folgenden Bild, bzw. in der folgenden Tabelle, zu sehen ist, an die Pins der "A-Seite" Ihres Voltage Translators an und dann von der "B-Seite" des Voltage Translators an Ihren Arduino. Hier empfehlen wir den [KY-051 Voltage Translator](#) von [Joy-IT](#).



Es ist nötig einen Logiklevel Wandler zu benutzen, da das Display ein 3,3 V Logiklevel hat, während der Arduino ein 5 V Logiklevel hat. Dies nicht zu beachten, kann zu Schäden am Display führen.



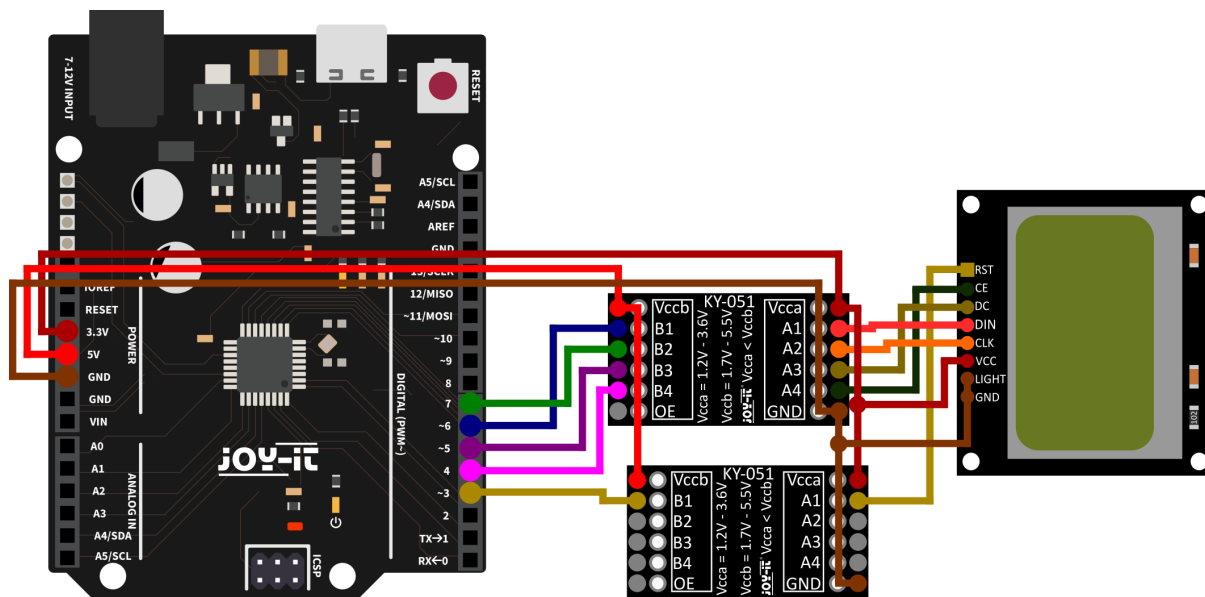
LCD	Arduino	KY-051 (1)	KY-051 (2)
VCC	3V3	Vcca	Vcca
-	5V	Vccb	Vccb
GND	GND	GND	GND
SCE	-	A4	-
-	D4	B4	-
RST	-	-	A1
-	D3	-	B1
D/C	-	A3	-
-	D5	B3	-
DN	-	A2	-
-	D6	B2	-
SCLK	-	A1	-
-	D7	B1	-

### 3.2 Anschließen des Displays (REV 2)

Schließen Sie das Display, wie im folgenden Bild, bzw. in der folgenden Tabelle, zu sehen ist, an die Pins der "A-Seite" Ihres Voltage Translators an und dann von der "B-Seite" des Voltage Translators an Ihren Arduino. Hier empfehlen wir den [KY-051 Voltage Translator](#) von [Joy-IT](#).



Es ist nötig einen Logiklevel Wandler zu benutzen, da das Display ein 3,3 V Logiklevel hat, während der Arduino ein 5 V Logiklevel hat. Dies nicht zu beachten, kann zu Schäden am Display führen.



LCD	Arduino	KY-051 (1)	KY-051 (2)
GND	GND	GND	GND
LIGHT	GND	-	-
VCC	3V3	Vcca	Vcca
-	5V	Vccb	Vccb
CLK	-	A2	-
-	D7	B2	-
DIN	-	A1	-
-	D6	B1	-
DC	-	A3	-
-	D5	B3	-
CE	-	A4	-
-	D4	B4	-
RST	-	-	A1
-	D3	-	B1

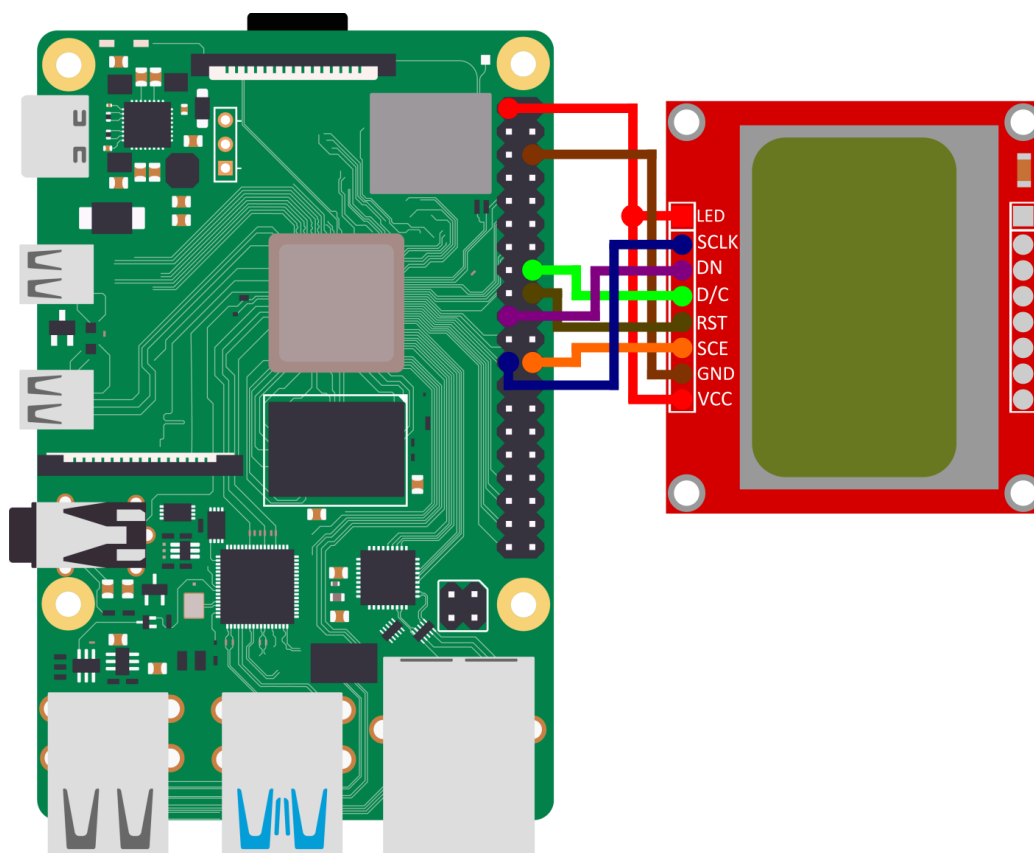
### 3.3 Codebeispiel Arduino

Für dieses Beispiel wird die Bibliothek [Adafruit-PCD8544-Nokia-5110-LCD-library](#) verwendet, welche von [Adafruit](#) unter der [BSD Lizenz](#) veröffentlicht wurde. Nachdem Sie diese Bibliothek in Ihrer Arduino IDE installiert haben, finden Sie ein Codebeispiel unter **Datei → Beispiele → Adafruit PCD8544 Nokia 5110 LCD Library → pcdtest**. Diesen Code können Sie mittels Hochladen auf Ihren Arduino laden. Achten Sie auch darauf, dass unter **Werkzeuge → Board** und **Werkzeuge → Port** die richtigen Einstellungen gemacht sind.

## 4. VERWENDUNG MIT EINEM RASPBERRY PI

### 4.1 Anschließen des Displays (REV 1)

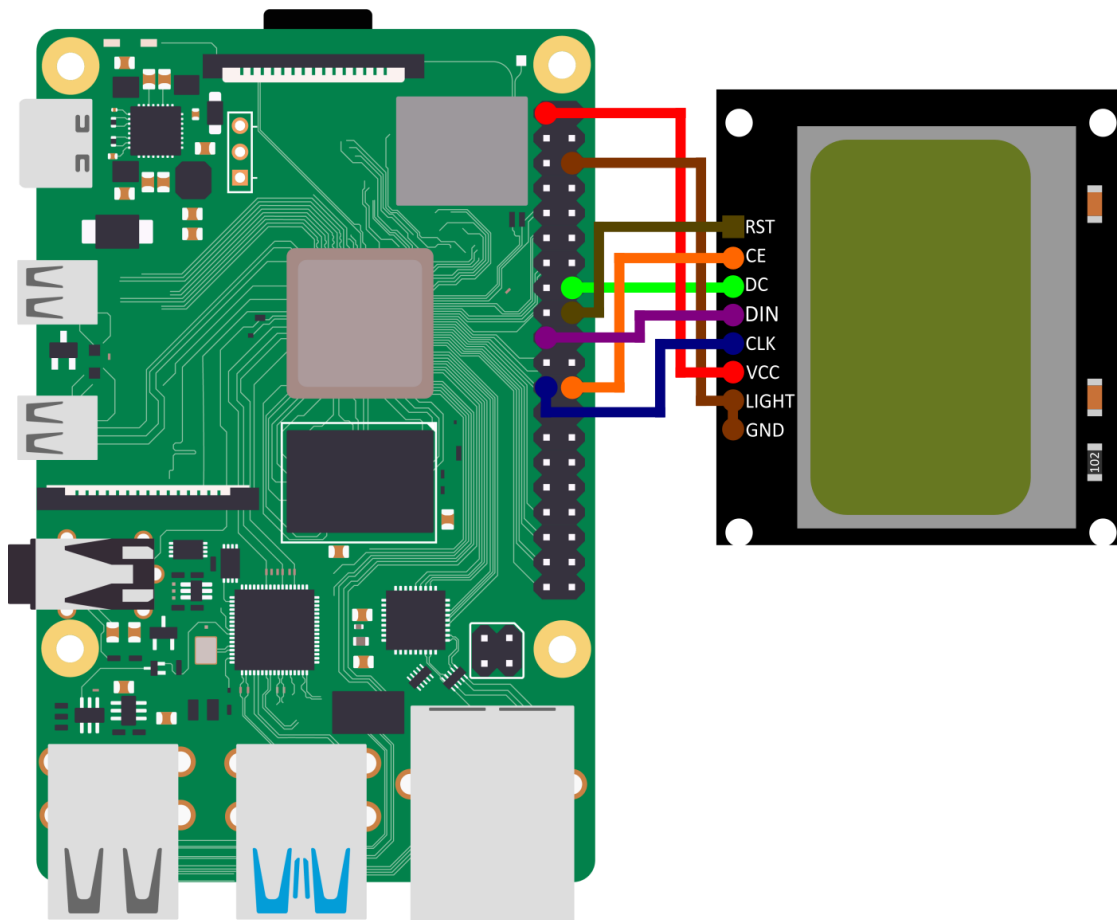
Schließen Sie das Display, wie im folgenden Bild, bzw. in der folgenden Tabelle, zu sehen ist, an die Pins Ihres Raspberry Pis an.



Raspberry Pi	84 x 48 LCD
Pin 1 (3.3 V)	8 - LED
Pin 23 (BCM 11 / SCLK)	7 - SCLK
Pin 19 (BCM 10 / MOSI)	6 - DN < MOSI >
Pin 16 (BCM 23)	5 - D/C
Pin 18 (BCM 24)	4 - RST
Pin 24 (BCM 8)	3 - SCE
Pin 6 (GND)	2 - GND
Pin 1 (3.3 V)	1 - VCC

## 4.2 Anschließen des Displays (REV 2)

Schließen Sie das Display, wie im folgenden Bild, bzw. in der folgenden Tabelle, zu sehen ist, an die Pins Ihres Raspberry Pis an.



Raspberry Pi	84 x 48 LCD
Pin 18 (BCM 24)	8 - RST
Pin 24 (BCM 8 / CE0)	7 - CE
Pin 16 (BCM 23)	6 - DC
Pin 19 (BCM 10 / MOSI)	5 - DIN
Pin 23 (BCM 11 / SCLK)	4 - CLK
Pin 1 (3.3 V)	3 - VCC
Pin 6 (GND)	2 - LIGHT
Pin 6 (GND)	1 - GND



Diese Anleitung wurde unter Raspberry Pi OS Bookworm für den Raspberry Pi 4 und 5 geschrieben. Es wurde keine Überprüfung mit anderen/neueren Betriebssystemen oder Hardware durchgeführt.

## 4.3 Codebeispiel Raspberry Pi

### Installation der Bibliotheken

Um Ihnen eine möglichst schnelle und einfache Verwendung des Displays zu ermöglichen, greifen wir auf die Bibliothek [Adafruit CircuitPython PCD8544](#) von [Adafruit](#) zurück, welche unter der [MIT-Lizenz](#) veröffentlicht wurde. Öffnen Sie die Konsole und führen Sie die folgenden Befehle aus:

```
sudo apt-get install python3-pip
sudo apt-get install fonts-dejavu
sudo apt-get install python3-pil
```

Um die SPI-Schnittstelle zu aktivieren, müssen Sie zunächst in die raspi-config gelangen. Geben Sie dazu folgenden Befehl ein:

```
sudo raspi-config
```

Wählen Sie dann die Option "**3 Interface Options**" und gehen Sie zu "**I4 SPI**", um die SPI-Schnittstelle zu aktivieren. Bestätigen Sie Ihre Auswahl durch Drücken von "**OK**".

Geben Sie Anschließend den folgenden Befehl ins Terminal ein:

```
sudo nano /boot/firmware/config.txt
```

Fügen Sie nun folgende Zeile an das Ende der Datei an. Anschließend speichern Sie die Datei mit **STRG+O** und schließen Sie mit **STRG+X** und **Enter**:

```
dtoverlay=spi0-0cs
```

Starten Sie jetzt Ihren Raspberry Pi neu mit dem folgendem Befehl:

```
sudo reboot
```

Nun müssen Sie virtuelle Umgebung für dieses Projekt erstellen, geben Sie dazu folgende Befehle ein:

```
mkdir dein_projekt
cd dein_projekt
python -m venv --system-site-packages env
source env/bin/activate
```

Installieren Sie nun die verwendete Bibliothek mit dem folgendem Befehl:

```
pip3 install adafruit-circuitpython-pcd8544
```

## Verwendung der Beispielcodes

Nach Abschluss der Installation, können Sie die Beispiel Datei erstellen, geben Sie dazu den folgenden Befehl ein:

```
nano Displaytest.py
```

Nun müssen Sie noch den folgenden Code in die Datei kopieren. Natürlich können Sie dieses Beispiel nach Ihren Wünschen und Vorstellungen modifizieren. Sie können dieses Codebeispiel auch [hier](#) herunterladen.

```
import time
import board
import busio
import digitalio

import adafruit_pcd8544

# SPI-Bus und Steuerpins initialisieren
spi = busio.SPI(board.SCK, MOSI=board.MOSI)
dc = digitalio.DigitalInOut(board.D23)
ce = digitalio.DigitalInOut(board.D8)
reset = digitalio.DigitalInOut(board.D24)

display = adafruit_pcd8544.PCD8544(spi, dc, ce, reset)

display.bias = 4
display.contrast = 60

# LED-Hintergrundbeleuchtung einschalten
# über einen extra GPIO Pin
backlight = digitalio.DigitalInOut(board.D26)
backlight.switch_to_output()
backlight.value = False

print("Pixel test")

# Löschen Sie die Anzeige. Rufen Sie show immer nach der
# Änderung von Pixeln auf
# um die Aktualisierung der Anzeige sichtbar zu machen!
display.fill(0)
display.show()

# Setzt ein Pixel an den Ursprung 0,0.
display.pixel(0, 0, 1)
# Setzt ein Pixel in die Mitte des Displays.
display.pixel(display.width // 2, display.height // 2, 1)
# Setzen eines Pixels in der gegenüberliegenden Eckposition.
display.pixel(display.width - 1, display.height - 1, 1)
display.show()
time.sleep(2)
```



```

print("Linien test")
# Wir zeichnen von Ecke zu Ecke, wobei wir hier ebenfalls
# alle Koordinatenpaare definieren
corners = (
    (0, 0),
    (0, display.height - 1),
    (display.width - 1, 0),
    (display.width - 1, display.height - 1),
)
display.fill(0)
for corner_from in corners:
    for corner_to in corners:
        display.line(corner_from[0], corner_from[1], corner_to[0], corner_to[1], 1)
display.show()
time.sleep(2)

print("Rechtecks test")
display.fill(0)
w_delta = display.width / 10
h_delta = display.height / 10
for i in range(11):
    display.rect(0, 0, int(w_delta * i), int(h_delta * i), 1)
display.show()
time.sleep(2)

print("Text test")
display.fill(0)
display.text("Presenting an ", 0, 0, 1)
display.text(" Examplecode ", 0, 8, 1)
display.text("for the SBC- ", 0, 16, 1)
display.text("LCD84x48 based", 0, 24, 1)
display.text("on the pcd8455", 0, 32, 1)
display.text("Nokia 5110 LCD", 0, 40, 1)
display.show()

while True:
    display.invert = True
    time.sleep(0.5)
    display.invert = False
    time.sleep(0.5)

```

Nachdem Sie den Code in Ihre Datei kopiert haben, können Sie diese durch Drücken von "**STRG+O**" speichern und durch "**STRG+X**" schließen. Um sicherzustellen, dass das Codebeispiel einwandfrei funktioniert, müssen Sie abschließend die Schriftart-Datei herunterladen und entpacken. Dazu führen Sie die folgenden beiden Befehle aus. **Um sicherzustellen, dass alles funktioniert, muss die entpackte Schriftart-Datei im selben Ordner wie der auszuführende Code platziert werden.**

```

wget https://joy-it.net/files/files/Produkte/SBC-LCD84x48/font5x8.zip
unzip font5x8.zip

```

Der folgende Befehl führt dann die Datei mit Ihrem Code aus und beenden können Sie den Code mit **STRG+C**.

```

python3 Displaytest.py

```

## 5. SONSTIGE INFORMATIONEN

Unsere Informations- und Rücknahmepflichten nach dem Elektroggesetz (ElektroG)

**Symbol auf Elektro- und Elektronikgeräten:**



Diese durchgestrichene Mülltonne bedeutet, dass Elektro- und Elektronikgeräte **nicht** in den Hausmüll gehören. Sie müssen die Altgeräte an einer Erfassungsstelle abgeben. Vor der Abgabe haben Sie Altbatterien und Altakkumulatoren, die nicht vom Altgerät umschlossen sind, von diesem zu trennen.

### **Rückgabemöglichkeiten:**

Als Endnutzer können Sie beim Kauf eines neuen Gerätes, Ihr Altgerät (das im Wesentlichen die gleiche Funktion wie das bei uns erworbene neue erfüllt) kostenlos zur Entsorgung abgeben. Kleingeräte bei denen keine äußeren Abmessungen größer als 25 cm sind können unabhängig vom Kauf eines Neugerätes in Haushaltsüblichen Mengen abgeben werden.

### **Möglichkeit Rückgabe an unserem Firmenstandort während der Öffnungszeiten:**

SIMAC GmbH, Pascalstr. 8, D-47506 Neukirchen-Vluyn

### **Möglichkeit Rückgabe in Ihrer Nähe:**

Wir senden Ihnen eine Paketmarke zu mit der Sie das Gerät kostenlos an uns zurücksenden können. Hierzu wenden Sie sich bitte per E-Mail an [Service@joy-it.net](mailto:Service@joy-it.net) oder per Telefon an uns.

### **Informationen zur Verpackung:**

Verpacken Sie Ihr Altgerät bitte transportsicher, sollten Sie kein geeignetes Verpackungsmaterial haben oder kein eigenes nutzen möchten kontaktieren Sie uns, wir lassen Ihnen dann eine geeignete Verpackung zukommen.

## 6. SUPPORT

Wir sind auch nach dem Kauf für Sie da. Sollten noch Fragen offen bleiben oder Probleme auftauchen stehen wir Ihnen auch per E-Mail, Telefon und Ticket-Supportsystem zur Seite.

E-Mail: [service@joy-it.net](mailto:service@joy-it.net)

Ticket-System: <http://support.joy-it.net>

Telefon: +49 (0)2845 9360 - 50 (Mo - Do: 09:00 - 17:00 Uhr,  
Fr: 09:00 - 14:30 Uhr)

Für weitere Informationen besuchen Sie unsere Website:

[www.joy-it.net](http://www.joy-it.net)