

# JOY-IT

## I2C Serial 20x4 2004 LCD Modul



## Index

1. Verwendung mit einem Arduino
  - 1.1 Anschließen des Displays
  - 1.2 Installation der Bibliothek
  - 1.3 Beispiel-Code
  
2. Verwendung mit einem Raspberry Pi
  - 2.1 Installation der Software
  - 2.2 Anschließen des Displays
  - 2.3 Installation der Bibliotheken
  - 2.4 Kommunikation mit dem Display
  
3. Support

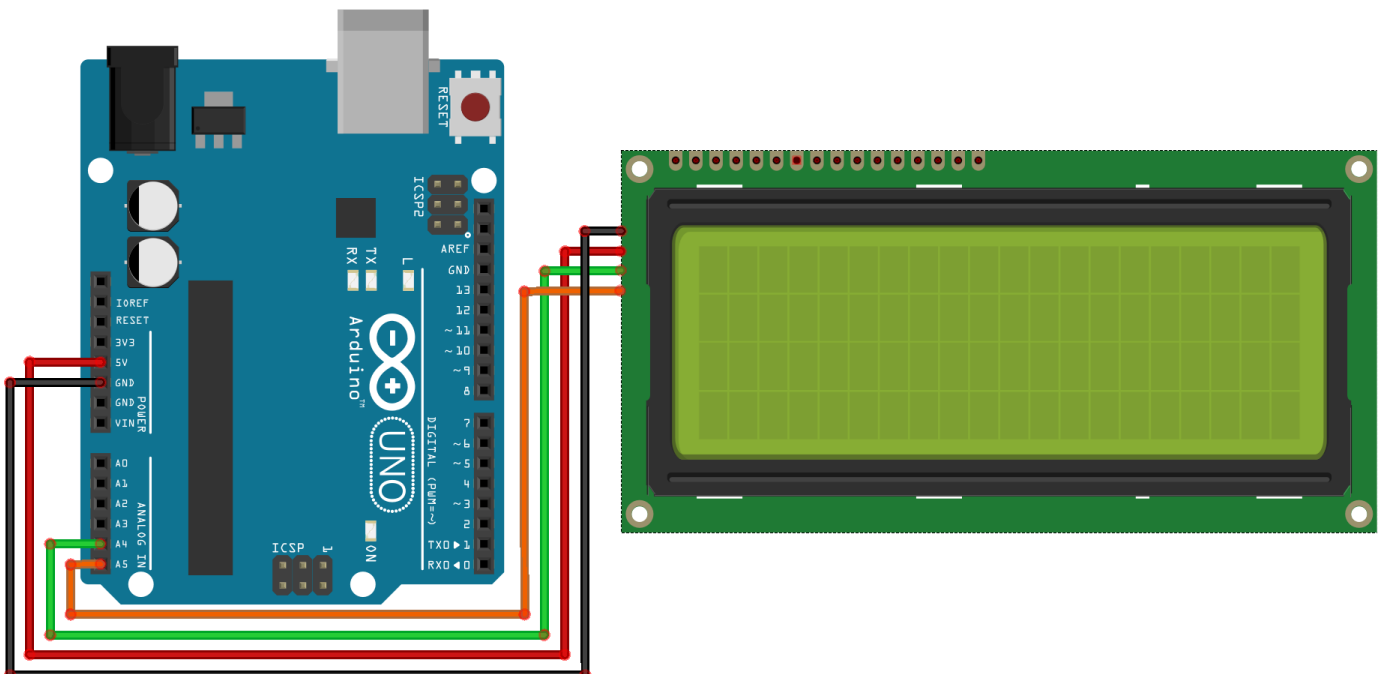
Sehr geehrter Kunde,

vielen Dank, dass Sie sich für unser Produkt entschieden haben.  
 Im Folgenden haben wir aufgelistet, was bei der Inbetriebnahme zu beachten ist.

## 1. Verwendung mit einem Arduino

### 1.1 Anschließen des Displays

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



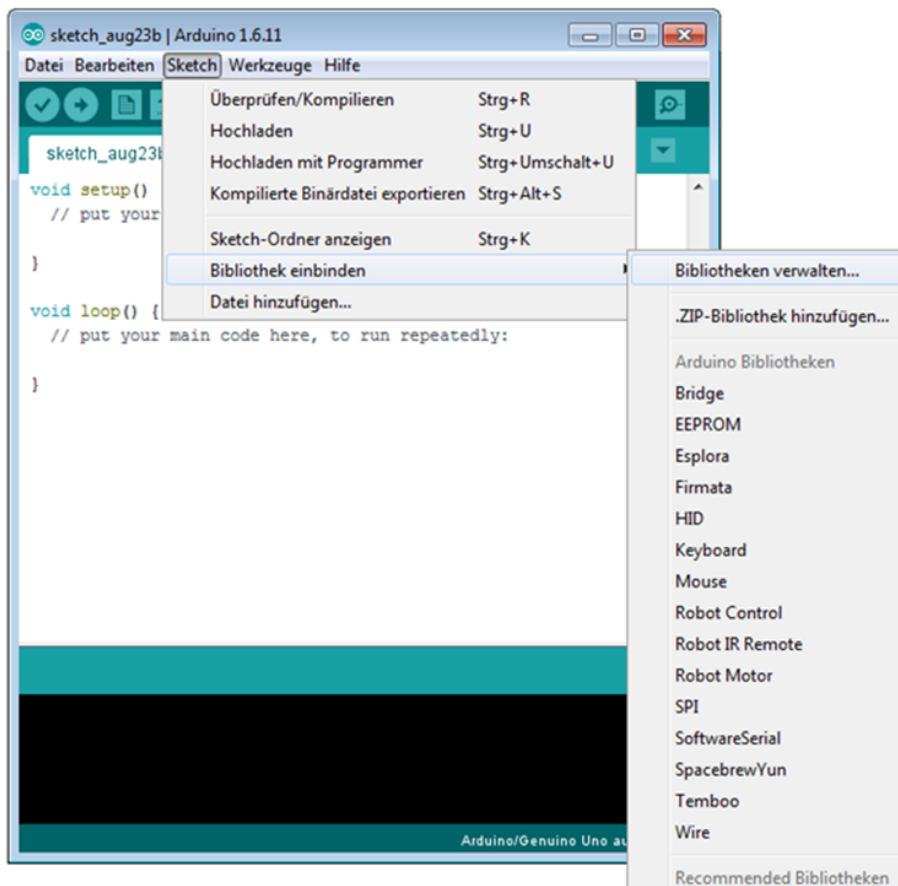
Arduino Uno	I2C 20x4 LCD
GND	GND
+5V	VCC
SDA	SDA
SCL	SCL

## 1.2 Installation der Bibliothek

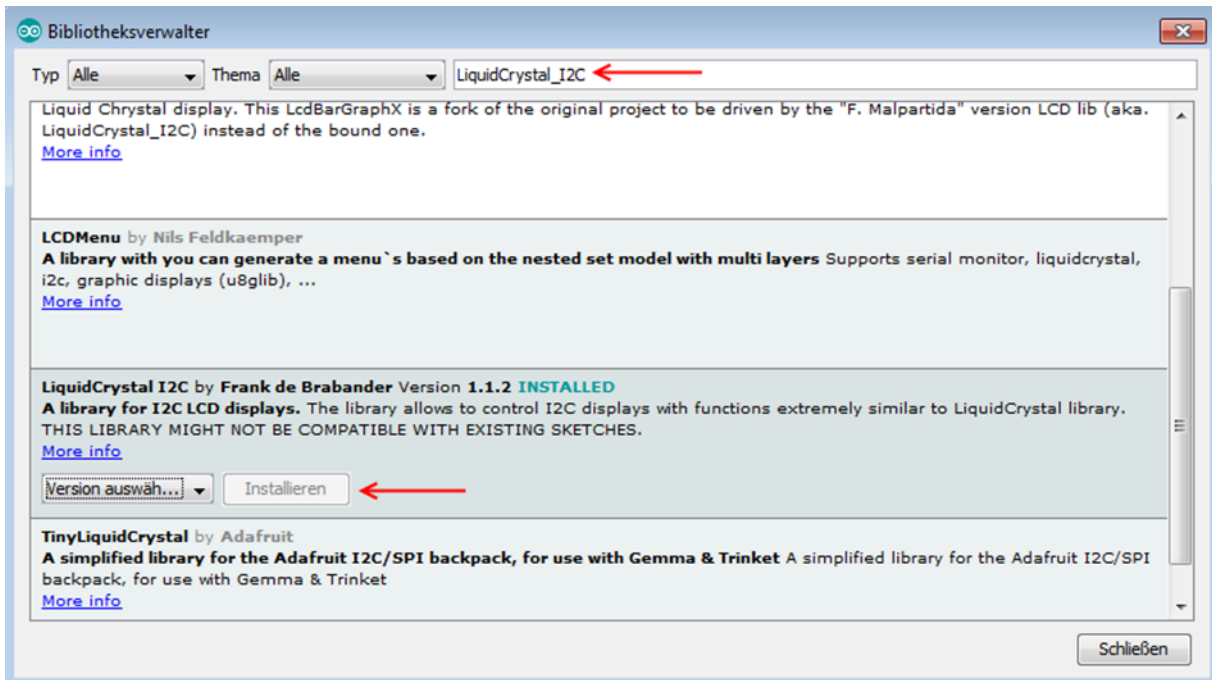
Bevor Sie das Code-Beispiel auf Ihren Arduino übertragen können, müssen Sie zunächst die zusätzliche Bibliothek **LiquidCrystal\_I2C** installieren.

Diese Bibliothek ermöglicht Ihnen eine möglichst einfache und schnelle Verwendung des Displays.

Öffnen Sie hierzu zunächst die Bibliotheksverwaltung in Ihrer Arduino Software.



Suchen Sie nun in dem sich öffnenden Bibliotheksverwalter nach der **LiquidCrystal\_I2C** Bibliothek und installieren Sie diese.



Sobald die Bibliothek installiert ist, ist die Installation des Displays abgeschlossen. Sie können nun im nächsten Schritt unseren Beispielcode übertragen, um das Display zu testen.

### 1.3 Beispiel-Code

Im Nachfolgenden können Sie ein Codebeispiel zur Verwendung des Displays entnehmen. Kopieren Sie dies vollständig auf Ihren Arduino.

Da es unterschiedliche Hardware-Adressierungen der Displays gibt, muss auch im Code die richtige Adressierung verwendet werden.

Sollte Ihr Display beim ersten Versuch nicht die gewünschte Aktion durchführen, so tauschen Sie bitte die Adressierung im dafür vorgesehenen Bereich aus.

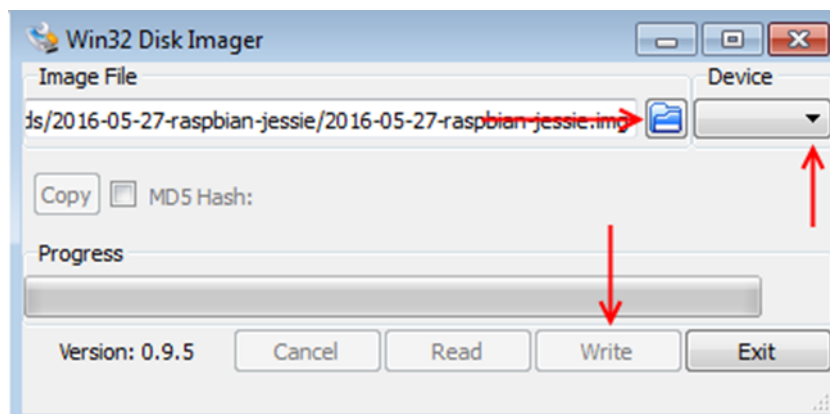
```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
//-----Hardware Adressierung-----
//Bei falscher Funktion bitte obere Zeile auskommentieren,
//und untere Zeile freigeben
LiquidCrystal_I2C lcd(0x27,20,4);
//LiquidCrystal_I2C lcd(0x3F,20,4);
void setup()
{
  lcd.init();
}
void loop()
{
  lcd.backlight();
  //Nachricht ausgeben
  lcd.setCursor(0,0);
  lcd.print(" joy-IT");
  lcd.setCursor(0,1);
  lcd.print(" ");
  lcd.setCursor(0,2);
  lcd.print(" I2C Serial");
  lcd.setCursor(0,3);
  lcd.print(" LCD");
}
```

## 2. Verwendung mit einem Raspberry Pi

### 2.1 Installation der Software

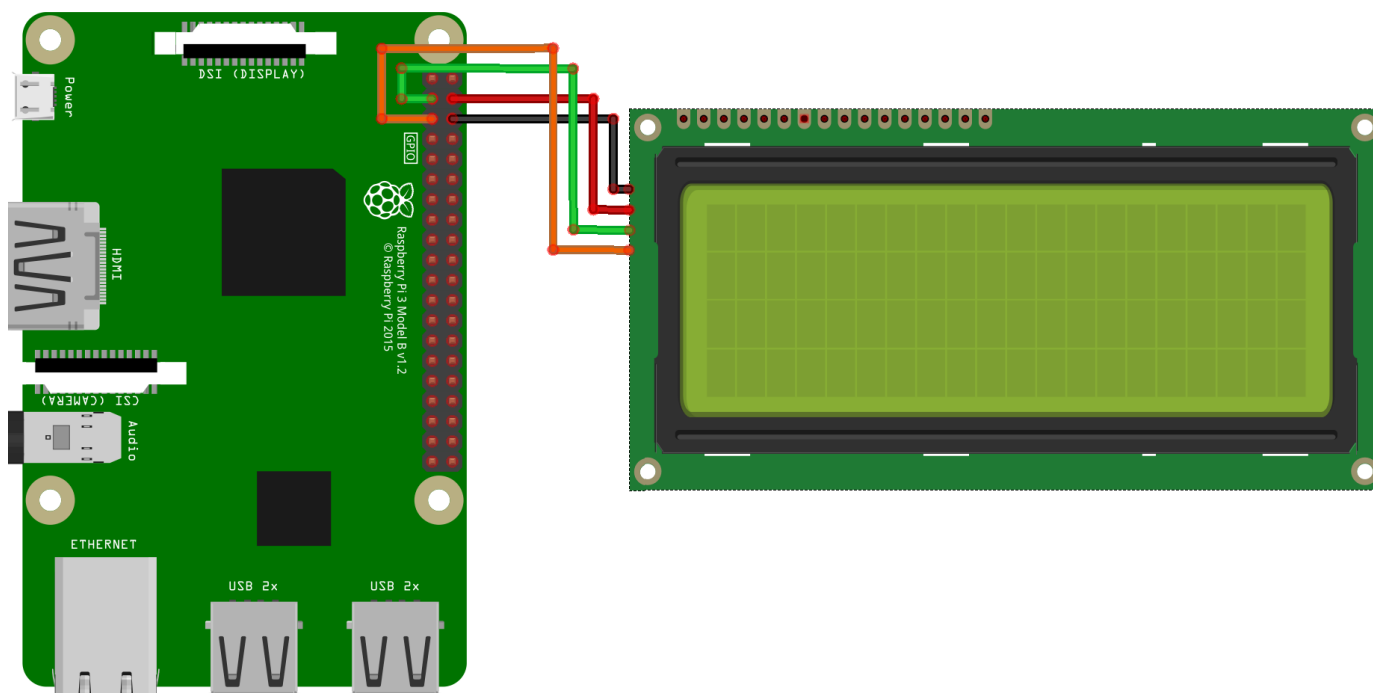
Sollten Sie bereits ein aktuelles Raspbian-System auf Ihrem Raspberry verwenden, so können Sie diesen Schritt überspringen und sofort mit Schritt 2.2 fortfahren.

Installieren Sie auf Ihre SD-Karte mit Hilfe des „[Win32 Disk Imager](#)“-Programms das aktuelle Raspbian Image, welches Sie unter dem folgenden [Link](#) zum Download finden.



## 2.2 Anschließen des Displays

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	I2C 20x4 LCD
Pin 6 (Ground)	GND
Pin 4 (5V)	VCC
Pin 3 (BCM 2 / SDA)	SDA
Pin 5 (BCM 3 / SCL)	SCL



## 2.3 Installation der Bibliotheken

Sobald Sie die Installation abgeschlossen und das System gestartet haben, öffnen Sie die Terminal-Konsole und führen Sie folgende Kommandos aus:

```
sudo apt-get install python-pip python-dev build-essential
sudo pip install RPi.GPIO
```

```
sudo apt-get install python-imaging
```

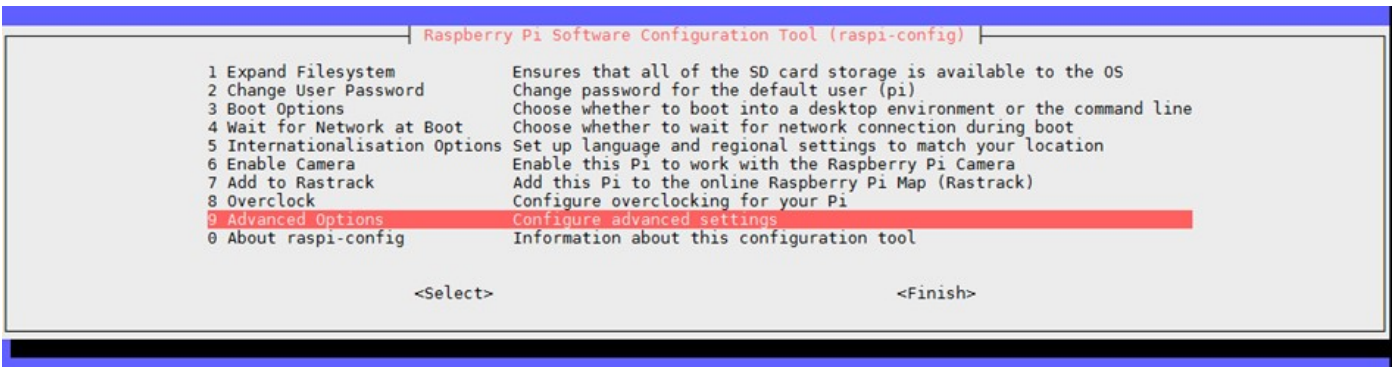
```
sudo apt-get install python-smbus i2c-tools
```

Sollte die I2C-Funktion auf Ihrem Raspberry Pi noch nicht aktiviert sein, so müssen Sie dies nun in den Einstellungen nachholen.

Öffnen Sie hierzu die Einstellungen mit folgendem Befehl:

```
sudo raspi-config
```

In dem sich nun öffnenden Fenster wählen Sie die Option **Advanced Options**.



```
Raspberry Pi Software Configuration Tool (raspi-config)
1 Expand Filesystem          Ensures that all of the SD card storage is available to the OS
2 Change User Password       Change password for the default user (pi)
3 Boot Options               Choose whether to boot into a desktop environment or the command line
4 Wait for Network at Boot   Choose whether to wait for network connection during boot
5 Internationalisation Options Set up language and regional settings to match your location
6 Enable Camera              Enable this Pi to work with the Raspberry Pi Camera
7 Add to Rastrack            Add this Pi to the online Raspberry Pi Map (Rastrack)
8 Overclock                  Configure overclocking for your Pi
9 Advanced Options           Configure advanced settings
0 About raspi-config         Information about this configuration tool

<Select>                                <Finish>
```

Hier wählen und aktivieren Sie die Option **I2C**.

```
Raspberry Pi Software Configuration Tool (raspi-config)
A1 Overscan      You may need to configure overscan if black bars are present on display
A2 Hostname      Set the visible name for this Pi on a network
A3 Memory Split  Change the amount of memory made available to the GPU
A4 SSH           Enable/Disable remote command line access to your Pi using SSH
A5 SPI           Enable/Disable automatic loading of SPI kernel module (needed for e.g. PiFace)
A6 I2C           Enable/Disable automatic loading of I2C kernel module
A7 Serial        Enable/Disable shell and kernel messages on the serial connection
A8 Audio         Force audio out through HDMI or 3.5mm jack
A9 1-Wire        Enable/Disable one-wire interface
AA GPIO Server   Enable/Disable remote access to GPIO pins

<Select>                <Back>
```

Als nächstes müssen nun die passenden Einträge in der Modul-Datei hinzugefügt werden.  
Hierfür öffnen Sie die Modul-Datei:

```
sudo nano /etc/modules
```

```
i2c-bcm2708
i2c-dev
```

An das Ende dieser Datei fügen Sie die folgenden beiden Zeilen an:

Speichern Sie nun die Datei mit der Tastenkombination **STRG + O**.  
Sie können danach den Editor mit der Kombination **STRG + X** verlassen.

```
sudo reboot
```

Starten Sie Ihren Raspberry Pi nun mit dem folgenden Befehl neu:

Nun können wir den LCD Treiber herunterladen.

Um Ihnen einen möglichst schnellen und einfachen Einstieg ermöglichen zu können, greifen wir hierbei auf die Treiber **Raspberry Pi LCD 4x20 I<sup>2</sup>C python driver** von **CaptainStouf** zurück.

Diese wurde unter der **GNU General Public License v2.0** freigegeben und ist [hier](#) abrufbar.

Als erstes erstellen wir einen neuen Ordner für das Treiberpaket und laden dieses anschließend herunter:

```
mkdir I2C-LCD && cd I2C-LCD
wget http://tutorials-raspberrypi.de/wp-content/uploads/scripts/
hd44780_i2c.zip
unzip hd44780_i2c.zip
```

Nun beginnen wir mit einer ersten Kommunikation zum Display.

Bitte beachten Sie, dass das Display hierfür bereits angeschlossen sein muss.

```
sudo i2cdetect -y 1
```

Die Ausgabe sollte nun in etwa folgendes anzeigen:

```

    0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  27  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
```

Die eben gestartete Prüfung hat ein I2C-Signal an der Adresse **27** festgestellt.

Dies ist die Standard-Adresse des Displays.

In manchen Fällen kann hier auch eine andere Speicheradresse gefunden werden.

In diesem Fall müssen Sie die Speicheradresse in den Treibereinstellungen auf Ihre Adresse aktualisieren.

Öffnen Sie dazu die Treiberdatei mit folgendem Befehl:

```
sudo nano lcddriver.py
```

An der Stelle **ADDRESS = 0x27** müssen Sie nun die **27** auf die Speicheradresse anpassen, den die I2C-Prüfung bei Ihnen angezeigt hat.

Danach können Sie den Editor mit der gewohnten Kombination verlassen.

## 2.4 Kommunikation mit dem Display

Die Installation des Displays ist nun abgeschlossen.

```
sudo nano LCD.py
```

Als letztes können Sie nun eine neue Datei erstellen, um das Display zu testen und in Betrieb zu nehmen.

Geben Sie nun das folgende Code-Beispiel ein.

Der Text innerhalb des Befehls **lcd.lcd\_display\_string** gibt den Text an, der auf dem Display angezeigt wird.

Die Zahl dahinter gibt die Zeile an, in der die Ausgabe geschrieben wird.

Den Text können Sie selbstverständlich nach Ihren Wünschen modifizieren.

```
import lcddriver
from time import *

lcd = lcddriver.lcd()
lcd.lcd_clear()

lcd.lcd_display_string("    joy-IT", 1)
lcd.lcd_display_string("", 2)
lcd.lcd_display_string("    I2C Serial", 3)
lcd.lcd_display_string("    LCD", 4)
```

Den Editor können Sie danach mit der gewohnten Kombination verlassen.

```
sudo python LCD.py
```

### 3. 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 98469 – 66 (11- 18 Uhr)

Für weitere Informationen besuchen Sie unsere Website:

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