

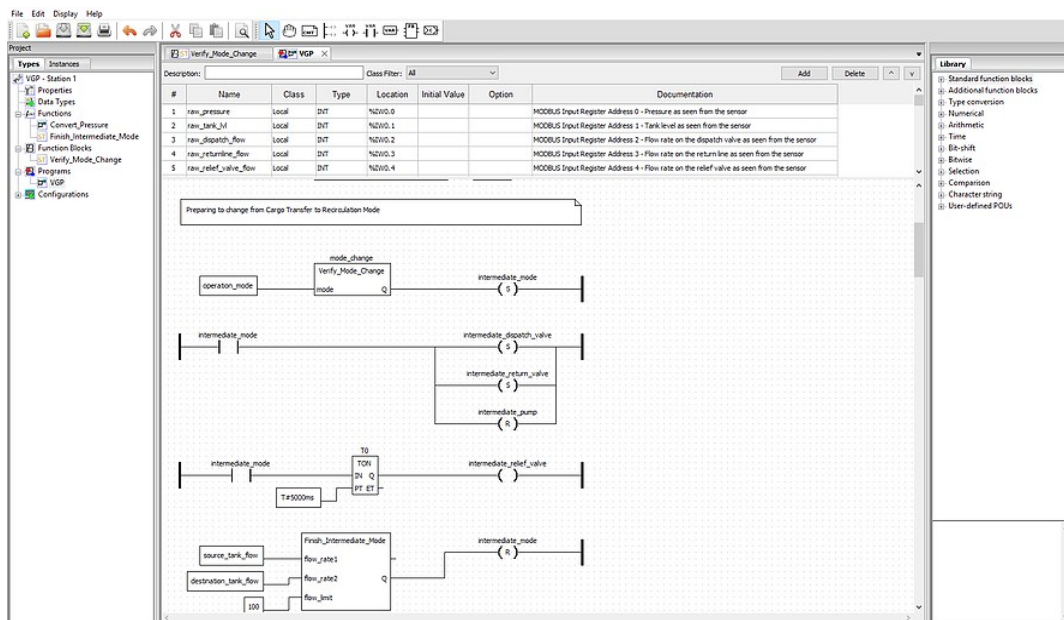


PiXtend

Application-Note: PiXtend mit OpenPLC Project

Application Note PiXtend mit OpenPLC Project

Installation, Einrichtung, Programmierung



(Bildquelle: www.OpenPLCProject.com)

APP-PX-301

Stand 29.10.2016, V1.01

Qube Solutions UG (haftungsbeschränkt)

Arbachtalstr. 6, 72800 Eningen, Germany

<http://www.qube-solutions.de/>

<http://www.pixtend.de>



Versionshistorie

Version	Beschreibung	Bearbeiter
1.00	Dokument erstellt	TG
1.01	Überarbeitung der Formatierung und Rechtschreibung	TG

Inhaltsverzeichnis

1. Einleitung.....	3
1.1 Voraussetzungen.....	4
1.3 Haftungsausschluss.....	4
1.3 Sicherheitshinweise.....	4
2. Installation.....	5
2.1 Linux Runtime.....	6
2.1 Programmierumgebung.....	7
3. Programmierung.....	8
3.1 Hello World Programm.....	9
4. Weitere Informationen.....	13
4.1 Verfügbare PiXtend I/Os.....	13
4.2 Umgang mit Tasks im OpenPLC Editor.....	14
4.3 OpenPLC Runtime automatisch starten.....	15
5. Frequently Asked Questions (FAQ).....	16



1. Einleitung

Wir freuen uns Ihnen eine weitere Möglichkeit präsentieren zu können, mit der sich PiXtend wie eine Industriesteuerung programmieren lässt.

Das **OpenPLC Project** hat sich zum Ziel gesetzt eine komplett **freie und standard-konforme¹ Software-Basis für speicherprogrammierbare Steuerungen (SPS, engl.: PLC)** zu kreieren. Wir finden die Umsetzung sehr gelungen und zum Konzept von PiXtend passend.

Besonders für die **Berufsausbildung** und den Einsatz im **technischen Studium** sehen wir aufgrund der **Einfachheit der Programmierung** klare Vorteile im Gegensatz zu ausgesprochenen „Profi-Tools“. Deshalb haben wir gemeinsam mit dem Projektersteller eine Treiber-Unterstützung für PiXtend erarbeitet, welche wir Ihnen hiermit vorstellen möchten.

**DESIGNED TO BE
OPEN SOURCE**

The OpenPLC is the first fully functional standardized open source PLC. We believe that opening the black-box of a PLC will create opportunities for people to study its concepts, create new technologies and share resources.

Abbildung 1: Designed to be Open Source - www.OpenPLCProject.com

In dieser AppNote möchten wir Ihnen Schritt für Schritt aufzeigen wie schnell und einfach Sie das OpenPLC Project auf Ihrem PiXtend-System einrichten und erste Programme erstellen.

Eine Übersicht über die verwendbaren I/Os und Funktionen finden Sie im Kapitel 4. *Weitere Informationen.*

Wir wünschen viel Spaß beim Testen, Programmieren und Experimentieren!

Viele weitere Informationen, Tipps und Tricks finden Sie auch in unserem Support-Forum unter: <http://www.pixtend.de/forum/>

Die jeweils neusten Versionen aller Dokumente und Software-Komponenten finden Sie im Download-Bereich unserer Homepage: <http://www.pixtend.de/downloads/>

¹ Programmiersprachen nach IEC/EN 61131-3 – DE (EN): ST, AWL (IL), KOP (LD), FBS (FBD), AS (SFC)



1.1 Voraussetzungen

Die Treiber-Unterstützung von PiXtend für das *OpenPLC Project* ist gleichermaßen für **PiXtend V1.2 und V1.3** verwendbar. Sie können daher jedes PiXtend-System verwenden.

Auch gibt es keine spezielle Festlegung auf ein Raspberry Pi Modell.

Wir empfehlen jedoch eines der folgenden Modelle: **B, B+, 2 B, 3 B**.

Im Folgenden bauen wir auf einem unveränderten **Raspbian Jessie Image** (Release Datum: 23.09.2016).

Auf den Raspberry Pi können Sie entweder mit direkt angeschlossener Tastatur und Monitor oder per SSH (TeraTerm / putty) von einem PC zugreifen.

1.3 Haftungsausschluss

Weder Qube Solutions UG noch das OpenPLC Project können für etwaige Schäden verantwortlich gemacht werden die unter Umständen durch die Verwendung der zur Verfügung gestellten Software, Hardware, Treiber oder der hier beschriebenen Schritte entstehen können.

1.3 Sicherheitshinweise



PiXtend darf nicht in sicherheitskritischen Systemen eingesetzt werden.

Prüfen Sie vor der Verwendung die Eignung von Raspberry Pi und PiXtend für Ihre Anwendung.



2. Installation

Die Installation der benötigten Software wird, abgesehen von diesem Dokument, auch auf der Homepage des *OpenPLC Project* erläutert:

<http://www.OpenPLCProject.com/getting-started> → PiXtend

Es steht Ihnen frei die Schritte auf der genannten Webseite (in englischer Sprache) oder in diesem Dokument nachzuvollziehen.

Wir starten mit einem originalen **Raspbian Jessie Image** (Release: 23.09.16). Bei diesem Image startet nach dem ersten Booten automatisch die neue PIXEL-Oberfläche. Da wir diese hier nicht benötigen, deaktivieren wir sie.

Dazu greifen wir per SSH auf den Raspberry Pi zu:

```
sudo raspi-config
```

Das Konfigurationsprogramm für den RPi wird ausgeführt. Im Menüpunkt „*3 Boot Options*“ wählen wir „*B2 Console Autologin*“.

Bei dieser Gelegenheit können wir auch gleich den **SPI-Bus aktivieren**. Dies geschieht ebenfalls im Konfigurationsprogramm **raspi-config**:

```
„9 Advanced Options“ → „A6 SPI“ → <Yes>
```

Nach dieser Änderung wird ein Reboot notwendig:

```
sudo reboot
```

Nun können wir in den nächsten Schritten die benötigten Komponenten herunterladen und installieren.



2.1 Linux Runtime

Das *OpenPLC Project* setzt, für den Zugriff auf die Schnittstellen und GPIOs des Raspberry Pi, auf **wiringPi**² (wie auch unsere *pxdev Linux Library*). Beginnen wir mit deren Installation. Es wird vorausgesetzt, dass Sie sich im Home-Verzeichnis des Users *Pi* befinden (`/home/pi`):

```
sudo apt-get install git-core
sudo apt-get update
git clone git://git.drogon.net/wiringPi
cd wiringPi
./build
```

Als nächstes benötigen wir noch einige Debian-Packages, die sich aber sehr komfortabel per `apt-get` installieren lassen:

```
sudo apt-get install build-essential bison flex autoconf automake libtool make nodejs git
```

Nun laden wir die neuste Version von OpenPLC von *github* herunter und kompilieren die Inhalte:

```
cd /home/pi
git clone https://github.com/thiagoralves/OpenPLC_v2.git
cd OpenPLC_v2
./build.sh
```

```
The OpenPLC needs a driver to be able to control physical or virtual hardware.
Please select the driver you would like to use:
1) Blank          5) UniPi          9) Arduino+RaspberryPi
2) Modbus         6) PiXtend         10) Simulink
3) Fischertechnik 7) Arduino
4) RaspberryPi    8) ESP8266
#? 6
```

Gegen Ende des Build-Prozesses werden Sie nach der verwendeten Hardware gefragt. Wir wählen die **Nummer 6 PiXtend** und bestätigen mit *Enter*.

Fast geschafft! Nun starten wir die Runtime mit folgendem Befehl:

```
sudo nodejs server.js
```

² WiringPi – OpenSource GPIO/SPI/I²C/Seriell Bibliothek für den Raspberry Pi – www.wiringpi.com

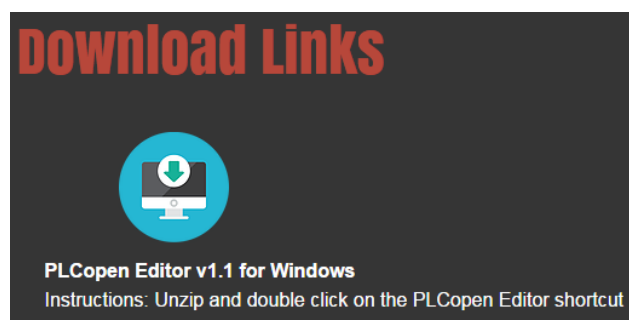


2.1 Programmierungsumgebung

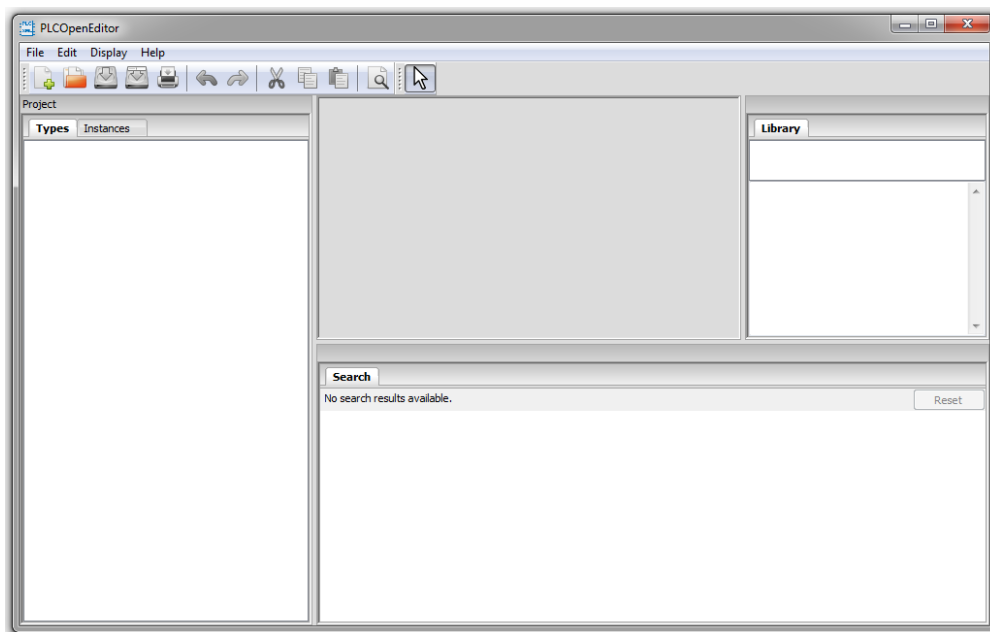
Die Programmierungsumgebung "PLCOpen Editor" ist kostenlos und für Windows, Linux und MacOS verfügbar. Wir verwenden die Windows-Variante in der Version 1.1.

Laden Sie sich die neuste Version bitte hier herunter:

<http://www.OpenPLC Project.com/plcopen-editor>



Nach dem Entpacken des ZIP-Archivs können Sie den Editor direkt über die beiliegenden Verknüpfung öffnen. Die Programmierungsumgebung startet:



Damit haben wir nun alle Software-Komponenten vorbereitet. Im nächsten Kapitel geht es ans Testen und Programmieren!



3. Programmierung

Im ersten Schritt bedienen wir uns eines Testprogramms, welches auf der Homepage des OpenPLC Project herunterladen werden kann:

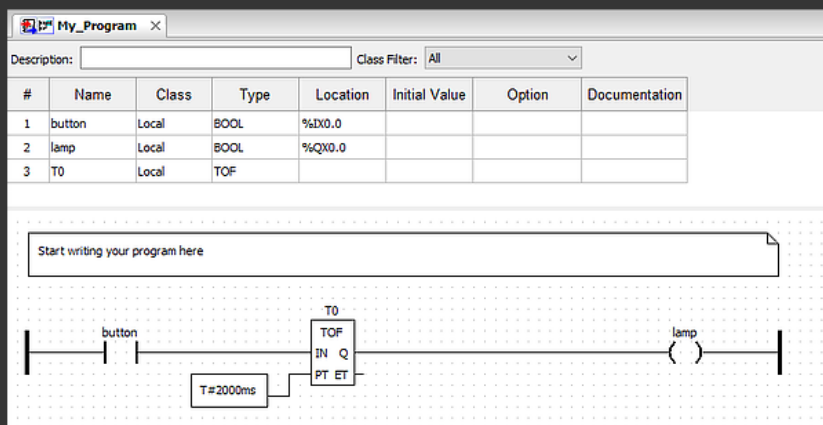
<http://www.OpenPLC Project.com/getting-started>

→ PiXtend

→ ganz nach unten scrollen und das „Hello World“ Programm herunterladen

Running the first project

This is a simple hello world project for the OpenPLC. The button is attached to %IX0.0 and the lamp is attached to %QX0.0. When the button is pressed and released, the lamp remains on for 2 seconds and then turns off.



Download the project below and open it on the PLCopen Editor. Click on **File -> Generate Program** in order to compile the diagram into a ST file. Then upload the ST file to the OpenPLC using the web interface.



Hello World

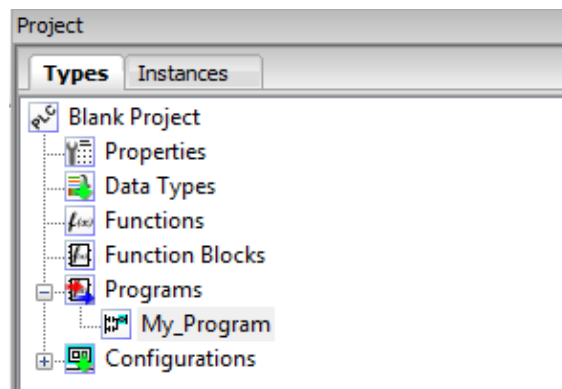
Die Programm- bzw. Projektdateien sind beim *OpenPLC Project* als XML-Datei organisiert.



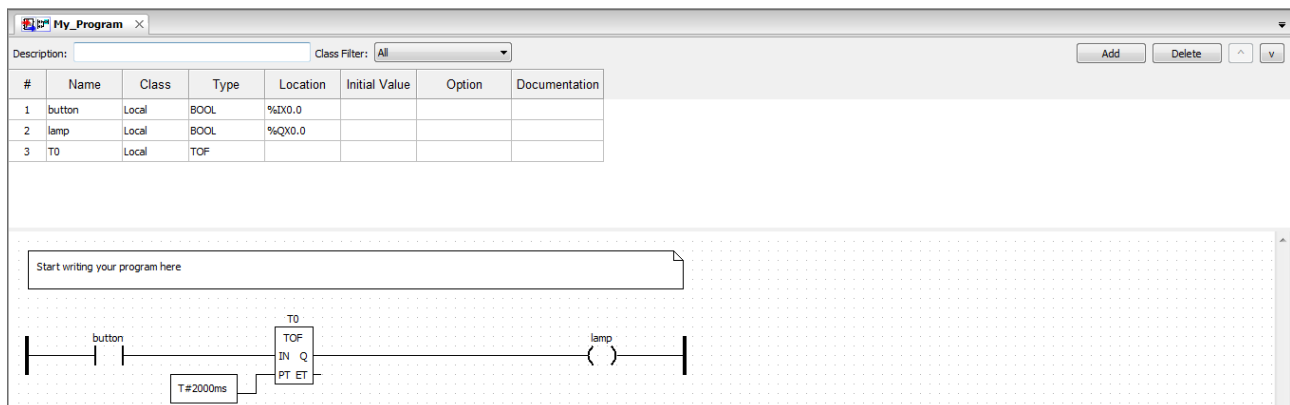
3.1 Hello World Programm

Starten Sie den PLCOpenEditor und wählen unter „File“ → „Open“ das zuvor heruntergeladene Testprogramm „Hello_World.xml“ aus.

Im Projektbaum erscheint unter „Programs“ das Beispielprogramm „MyProgram“:



Durch Doppelklick auf „My_Program“ öffnet sich der Editor.



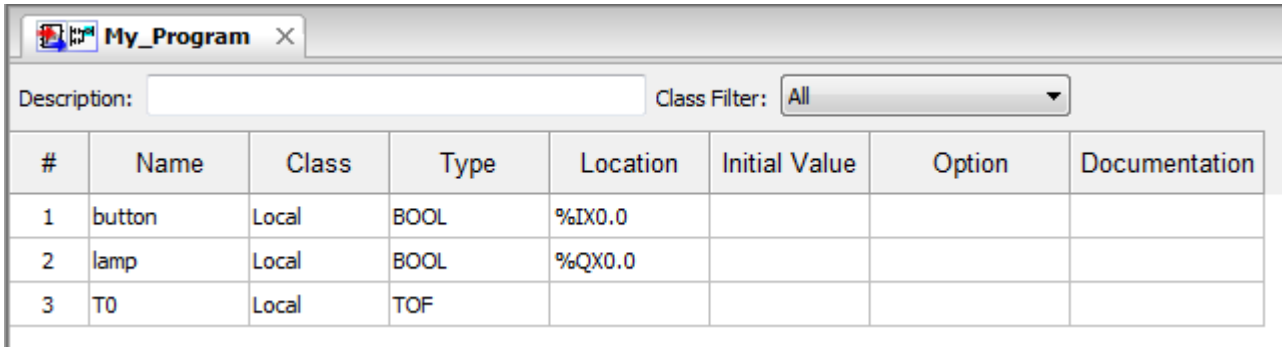
Das kleine Programm ist in Ladder Diagramm (deutsch: Kontaktplan - KOP) programmiert.

Ein „button“ betätigt den Eingang eines TOF-Bausteins. Diese Baustein verursacht eine Ausschaltverzögerung. In diesem Fall von 2000 ms (2 s).

Wenn der „button“ betätigt wird, aktiviert sich die „lamp“ am Ausgang des TOF-Bausteins sofort und wird nach dem loslassen des „button“ noch weitere 2 Sekunden gehalten.



Betrachten wir die Variablen-Liste:



#	Name	Class	Type	Location	Initial Value	Option	Documentation
1	button	Local	BOOL	%IX0.0			
2	lamp	Local	BOOL	%QX0.0			
3	T0	Local	TOF				

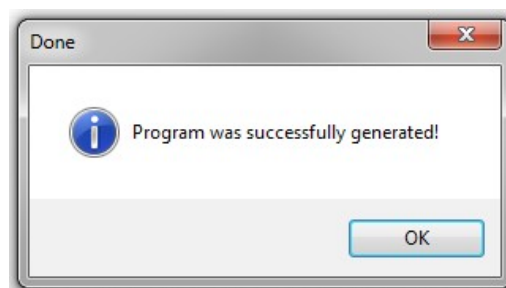
In der Spalte „Name“ ist der Variablenname festgelegt. Der Type ist hier für „button“ und „lamp“ ein BOOL, also je 1 Bit. Die Bits werden über die Spalte „Location“ mit einer Hardware-Adresse verbunden.

Der „button“ liegt hier also auf %IX0.0, was bei PiXtend den digitalen Eingang 0 (DI0) darstellt. Die „lamp“ liegt auf %QX0.0 – dem digitalen Ausgang 0 (DO0) auf PiXtend. Mehr zu den Hardware-Adressen finden Sie im Kapitel 4. *Weitere Informationen*.

Natürlich können auch Variablen festgelegt werden, die nicht mit einem Hardware-Ein-/Ausgang verbunden sind.

Nun ist es Zeit das Programm auf die PiXtend-Steuerung zu übertragen.

Unter „File“ → „Generate Program“ lässt sich das Programm in eine Datei umwandeln, die wir auf die Steuerung (Linux Runtime) laden können. Wird kein Fehler bei der Programmerstellung gefunden, so meldet der Editor den Erfolg:



Um das soeben erstellte Programm auf die Steuerung zu übertragen, öffnen wir an unserem Windows-Rechner einen Browser.



Die Linux Runtime von OpenPLC Project öffnet automatisch einen Webserver, auf den wir nun zugreifen können. Es wird davon ausgegangen, dass Sie die IP-Adresse des Raspberry Pi kennen.

In unserem Fall ist das die 192.168.1.27. Der Webserver läuft auf dem Port 8080:



Ersetzen Sie die IP-Adresse durch die Adresse Ihres Raspberry Pi. Es öffnet sich die übersichtliche Webseite des OpenPLC-Servers:

OpenPLC Server

Current PLC Status: **Running**

Run

Stop

View PLC logs

Change PLC Program

Datei auswählen

Keine ausgewählt

Upload Program

Change Modbus Master Configuration

Changing this only have effect if OpenPLC is using the Modbus Master Driver

Datei auswählen

Keine ausgewählt

Upload Configuration

Über „*Datei auswählen*“ selektieren wir nun das zuvor erstellte Program-File mit der Endung **.st** und laden dieses mit Hilfe des Buttons „*Upload Program*“ auf die Steuerung.

Die Ansicht wechselt und das Programm wird auf dem Raspberry Pi von der OpenPLC-Runtime kompiliert. Kommt es zu keinem Fehler, so startet das Programm im Anschluss automatisch.



Der Anzeige im Webinterface sollte nun folgendermaßen aussehen:

OpenPLC Server

Program compiled without errors!

```
compiling new program...
POUS.c
POUS.h
LOCATED_VARIABLES.h
VARIABLES.csv
Config0.c
Config0.h
Res0.c

Program compiled successfully
moving files...
compiling OpenPLC...
Generating object files...
Generating glueVars.cpp
varName: __IX0_0 varType: BOOL
varName: __QX0_0 varType: BOOL
Compiling main program

compiled without errors
```

Um die Funktion zu testen, legen Sie den DO0 von PiXtend auf High-Pegel (je nach Jumperstellung – auf 5 V oder 24 V). Der Ausgang DO0 wird aktiviert. Wird anschließend ein Low-Pegel an DI0 angelegt, verbleibt DO0 noch 2 Sekunden aktiv (Ausschaltverzögerung).

Herzlichen Glückwunsch!

Sie haben Ihr erstes SPS-Programm mit PiXtend und der Software vom *OpenPLC Project* in Betrieb genommen.

Einige weitere Informationen für die Weiterarbeit und für komplexere Programme finden Sie im folgenden Kapitel.



4. Weitere Informationen

In diesem Kapitel möchten wir Ihnen die wichtigsten Informationen mitgeben, damit Sie eigene Programme anlegen und die anderen Funktionen von PiXtend und dem OpenPLC Project verwenden können.

4.1 Verfügbare PiXtend I/Os

In der aktuellen Version (Stand: Oktober 2016) der Treiber-Unterstützung für PiXtend sind noch nicht alle I/Os und Funktionen verfügbar. Daher möchten wir Ihnen hier eine Übersicht über die unterstützten I/Os und deren Adressen geben:

PiXtend Bezeichnung	Typ	Beschreibung	Wertebereich	Datentyp	Adresse
DI0 - DI7	Input	Digitaler Eingang	0..1	BOOL	%IX0.0 - %IX0.7
AI0 - A1	Input	Analoger Spanneingang	0..1023	WORD	%IW0 - %IW1
AI2 - AI3	Input	Analoger Stromeingang	0..1023	WORD	%IW2 - %IW3
DO0 - DO5	Output	Digitaler Ausgang	0..1	BOOL	%QX0.0 - %QX0.5
PWM0 - PWM1	Output	(Modellbau)Servo Ausgang	0..255	WORD	%QW2 - %QW3
Relay0 - Relay3	Output	Relais Ausgang	0..1	BOOL	%QW0.6 - %QW0.7 %QW1.0 - %QW1.1
AO0 - AO1	Output	Analoger Ausgang	0..65435	WORD	%QW0 - %QW1

Die beiden PWM-Ausgänge können bisher nur im Servo-Modus betrieben werden.

Die "Rohwerte" der analogen Eingänge lassen sich wie folgt in Spannungen und Ströme umrechnen:

Spannungen:

$\text{Rohwert} * 10 / 1024 = \text{Spannung an Kanal X [V]}$

Ströme:

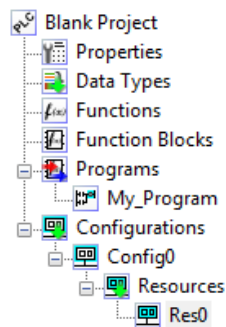
$\text{Rohwert} * 0.02419411599 = \text{Stromfluss bei Kanal X [mA]}$

Die analogen Ausgänge haben bei Wert „0“ die Ausgangsspannung 0 V, bei Wert „65535“ die Ausgangsspannung von 10 V. Sollten die gemessenen Werte abweichen, so korrigieren Sie diese per Potentiometer (T0 / T1).



4.2 Umgang mit Tasks im OpenPLC Editor

Wie Sie im Beispielprogramm „Hello_World“ sehen können, gibt es im Projektbaum noch spezielle Konfigurationen für das Projekt bzw. die Steuerung:



Im Element „Res0“ wird die Zykluszeit der SPS festgelegt. Im Beispielprogramm sieht der Inhalt von „Res0“ folgendermaßen aus:

Config0-Res0							
Class Filter: All							
#	Name	Class	Type	Location	Initial Value	Option	Documentation

Tasks:				
Name	Triggering	Single	Interval	Priority
TaskMain	Cyclic		T#50ms	0
Instances:				
Name	Type	Task		
Inst0	My_Program	TaskMain		



Der TaskMain wurde auf „Cyclic“ (deutsch: zyklisch) und auf ein Intervall von 50 ms eingestellt. Dies ist eine typische Konfiguration für eine Industriesteuerung.

Sie können das Intervall bei PiXtend auf minimal 25 ms einstellen. Eine kleinere Einstellung als 25 ms kann zu Problemen führen und wird daher nicht empfohlen!

Etwas weiter unten sehen Sie den Abschnitt „Instances“ hier wird dem „TaskMain“ unser Programm „My_Program“ zugewiesen.

Sie können bei Bedarf auch mehrere Tasks mit unterschiedlichen Einstellungen verwenden. Für erste Gehversuche empfehlen wir Ihnen aber die genannten Einstellungen zu machen.

4.3 OpenPLC Runtime automatisch starten

Nach einem Reboot / Power-Up des Raspberry Pi startet die OpenPLC Runtime nicht automatisch. Wie bereits bei der Installation beschrieben kann die Runtime mit folgendem Befehl gestartet werden:

```
sudo nodejs server.js
```

Wir können diesen Befehl aber auch automatisch beim Hochfahren des Linux-Systems ausführen lassen. Wir sind ja schließlich Automatisierer!

Die Änderung ist in der Linux-Console schnell erledigt:

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

Es öffnet sich die Datei „rc.local“ mit etwas Inhalt. Wir tragen vor der Zeile „exit 0“ zwei neue Zeilen ein:

```
cd /home/pi/OpenPLC_v2/  
sudo nodejs server.js &
```

Das „&“ ist kein Tippfehler. Damit startet die Runtime als Prozess im Hintergrund.

Abspeichern und mit STRL+X verlassen → Rebooten



5. Frequently Asked Questions (FAQ)

Ich kann mein Programm abspeichern, hochladen und ausführen – jedoch passiert nichts – was kann ich tun?

In der Regel wurde vergessen SPI zu aktivieren. Bitte werfen Sie einen Blick ins Installations-Kapitel. Ansonsten fehlt auch mal gerne der *SPI_EN* Jumper auf PiXtend. Dieser muss gesteckt sein, damit eine Kommunikation zwischen Raspberry Pi und PiXtend zustande kommen kann.

Sollte es sich um Ihre erste Inbetriebnahme von PiXtend handeln, so würden wir Sie gerne im ersten Schritt auf unsere vorbereiteten SD-Images (Linux Tools / CODESYS) verweisen. Diese sind bereits komplett vorbereitet und getestet. Hiermit können Sie ausschließen, dass es nicht noch ein Hardware-Problem gibt. Die häufigsten Hardware-Probleme sind:

- Chips falsch herum eingesteckt oder an falscher Stelle eingesteckt
→ Prüfen
- Bauteile bei der Bestückung vertauscht (beim Bausatz)
→ noch einmal etwas Zeit nehmen und die Bauteile überprüfen

Ich würde gerne auch die PiXtend-GPIOs, RS232, CAN, RS485 usw. verwenden. Ist das irgendwie möglich?

Ja! Allerdings haben wir für die erste Treiber-Unterstützung noch nicht alle Funktionen umgesetzt. Entweder Sie geben uns noch etwas Zeit für die Umsetzung oder Sie greifen selbst zum (Programmier-)Werkzeug. Sowohl PiXtend als auch das OpenPLC Project sind komplett Open Source. Sie können also alles einsehen und nach Ihren Wünschen ändern, weiterentwickeln...

Die Gerätebeschreibung für PiXtend finden Sie auf Ihrem RPi hier:

`/home/pi/OpenPLC_v2/core/hardware_layers/pixtend.cpp`

Der Code basiert auf der C-Library (pxdev) von Qube Solutions, genauer um den „Automatic Mode“. Weitere Infos hierzu finden Sie in der AppNote zur C-Library auf unserer Homepage im Download-Bereich: www.pixtend.de/downloads

Gerne möchten wir Sie für den Informationsaustausch in die Foren von Qube Solutions und den *OpenPLC Project* einladen:

<http://www.pixtend.de/forum/>

<http://openplc.discussion.community/>