

1. Einführung .....	3
2. Funktionsumfang .....	3
2.1 Hauptmerkmale des Jog Wheel Communicator Frameworks .....	3
2.2 Plugins.....	5
2.3 Module .....	5
3 Einrichtung von Jog Wheel Communicator.....	6
3.1 Installation von Jog Wheel Communicator .....	6
3.1.1 Installation für EDING CNC (Als Beispiel für API-Verwendung ).....	6
3.2 Lizenz und Demo-Modus.....	7
3.3 Ordnerstruktur.....	7
3.4 Start von Jog Wheel Communicator.....	7
3.5 Elemente des JWC Hauptfensters .....	8
3.6 Verwaltung und Erstellung der Handräder .....	9
3.7 Verwaltung und Erstellung der gesteuerten Applikationen .....	10
3.8 Handrad-Parameter Konfiguration.....	11
3.8.1 Tastenkombinationen anlegen.....	13
3.8.2 Simulation - Tastendefinitionen .....	14
3.8.3 Simulations-Einstellungen.....	14
3.8.4 Native Modus Einstellungen .....	15
3.8.5 Funktionen und Pools erstellen.....	15
3.9 Andere Einstellungen .....	15
3.9.1 LCD Bildschirm-Zusatzinfos .....	15
3.9.2 Native Einstellungen .....	16
3.9.3 Weitere Einstellungen .....	18
3.9.4 Weitere Einstellungen 2 .....	21
4. Funktionen .....	22
4.1 Funktion erstellen.....	22
4.1.1 Tastenkombinationen - Funktion .....	22
4.1.2 Bildschirm wechseln - Funktion.....	23
4.1.3 Plugin / Modul - Funktion .....	24
4.1.4 MDI Befehl - Funktion .....	26
4.1.5 Zustand / Modus / Variable setzen - Funktion .....	27
4.1.6 Befehl mit Bedingung erstellen - Funktion.....	31
4.1.7 Funktions-Pool erstellen.....	32
4.2 Verwaltung der Funktionen .....	33
4.2.1 Funktion löschen .....	34
4.2.2 Eigenschaften der Funktion bearbeiten.....	34
4.2.3 Zuordnungen verwalten .....	35
4.2.4 Funktions-Pool verwalten.....	35
5. Module.....	36
5.1 Installation, Aktivierung und Konfiguration eines Moduls .....	36
5.2 Verwendung .....	36
6. Anwendungsbeispiele .....	37
6.1 Verwenden einer Funktion in mehreren Profilen.....	37
6.2 Umschalten von Bildschirmen auf dem Handrad.....	38
6.3 Anlegen einer Tastenkombination .....	40
6.4 Realisierung einer Tinkerforge-basierten DRO Anzeige für 4 Achsen.....	42
Hardware .....	42
Software- Konfiguration.....	44
6.5 Handrad-Schrittweite Modus rotierend umschalten mit nur einer Taste .....	47
6.6 Realisierung eines Tinkerforge-basierten Handrades mit Drehencoder und 4 Tasten .....	57
Beispiele in späteren Versionen geplant .....	59
7 Profilbeschreibung für Eding CNC und LCD Touch Pendant .....	60
7.1 Startbildschirm .....	60
7.2 Achsenbewegungsbildschirm .....	60
7.3 Referenz-Bildschirm .....	61
7.4 2/4 Takt Modus und Override-Bildschirm .....	62
7.5 Tools-Bildschirm.....	63
7.6 Programm-Run Bildschirm .....	65
7.7 Wizards-Bildschirm.....	66

# **Jog Wheel Communicator Version 1.0**

Bedienungsanleitung V1.0

2015, Copyright (c) Dimitrij Saponov

Kopieren und Weiterverbreiten dieser Anleitung oder Teilen davon ist ohne schriftliche Erlaubnis des Autors untersagt.

# 1. Einführung

Handrad gehört zu den sinnvollsten Erweiterungen für eine CNC Maschine, es erlaubt schnelles positionieren und erleichtert die Einrichtung des Werkstückes erheblich. Durch die grosse Anzahl der Steuerungsprogrammen und Steuerungen existiert allerdings eine Reihe von Einschränkungen bei der Auswahl und Benutzung der Handräder:

- Handräder können nicht an alle Steuerungsprogramme oder Steuerungen angeschlossen werden. Vor allem Budget-Lösungen und Light-Versionen bieten oft die Möglichkeit nicht;
- Die verfügbaren Handräder sind proprietär für viele Systeme und können nicht an anderen Systemen verwendet werden;
- Die Handräder sind meistens sehr teuer, bieten dafür meistens aber nur grundlegende Funktionen;
- Funktionsumfang der meisten Handräder ist stark eingeschränkt;
- Konfigurierbarkeit beschränkt sich meist nur auf freie Belegung der Tasten (und das ist bereits eine fortschrittliche Funktion)

Aus diesen Einschränkungen ist der Wunsch nach einer universellen Lösung entstanden die es ermöglichen soll bei möglichst vielen Steuerungsprogrammen die Bedienung zu erleichtern mit Hilfe preiswerter und universell einsetzbaren Komponenten.

Aber es geht nicht nur um Handräder. In einem CNC-Umfeld ist es oft notwendig andere Maschinen und Geräte zu steuern, Sensoren auszulesen oder Automatisierungskomponenten einzubinden. Auch möchte man Anzeigen (DROs) anschliessen können oder komplexe Abläufe erstellen die z.B. mit einfachem Tastendruck abrufbar sind.

Diese und viele weitere Funktionen bietet das Jog Wheel Communicator (JWC) Framework das im Folgenden beschrieben wird.

## 2. Funktionsumfang

Jog Wheel Communicator Framework ist aus einem Hauptprogramm, Open Source Hardware, Software Modulen und Plugins bestehendes Framework das auf Basis von Microsoft .NET 4.0 Technologie aufgebaut ist und (in der erster Version) auf Windows XP, Windows 7 und Windows 8 Betriebssystemen einsetzbar ist. Die Handräder oder andere Eingabe/Ausgabe Komponenten werden mittels USB an den Rechner angeschlossen wobei auch die Möglichkeit besteht diese mit WLAN (also kabellos) anzubinden.

### 2.1 Hauptmerkmale des Jog Wheel Communicator Frameworks

JWC ist ein konfigurierbares Programm das die Anbindung und Benutzung der unterstützten CNC-Handräder an CNC-Steuerungsprogrammen (in Rahmen deren Möglichkeiten, siehe unten) ermöglicht. Das Programm hat folgende Eigenschaften:

- Freie Demo-Version: testen Sie das JWC Programm mit fast vollständigen Funktionsumfang (Begrenzung der Demo Version: nur X und Z Achsen können gefahren werden) damit Sie vor dem Kauf sicher sein können das JWC wie gewünscht funktioniert.
- Unterstützte Handräder:
  1. Das Handradmodell V8 von der Firma „cnc-techmics.de“ aufgebaut auf Basis des Touchscreen-Handrades (LCD Touch Pendant) dass zum ersten Mal im Forum <http://www.cncecke.de/forum/> (Suchen nach „Handrad Studie“) beschrieben worden ist. Dieses Handrad eignet sich besonders gut da es einen berührungssensitiven Bildschirm und somit sehr viele Konfigurations- und Anzeigemöglichkeiten besitzt. Die JWC Software wurde mit dem Handrad getestet, für andere Modelle des Handrades können weitere Anpassungen der Parameter / Aktualisierung der Firmware notwendig sein.

**Achtung !** Das mitgelieferte Profil funktioniert nur mit dem mitgelieferten (kostenlosen) Screenset für das LCD Touch Pendant Handrad! Falls ein eigenes Screenset verwendet wird, muss man das mitgelieferte Profil anpassen!

2. Handräder die auf Basis von handelsüblichen Rechtecksignalbasierten Drehencoder mit A B Ausgang (TTL 3.3 V oder 5V) sowie Tasten aufgebaut worden sind. Diese Handräder können über Tinkerforge-Bausteine wie Master Brick + IO16 Modul (<http://www.tinkerforge.com/>) angeschlossen werden (diese Module müssen separat erworben werden).
- Bei der Auslieferung wird folgende CNC – Steuersoftware auf API-Level über Plugin Unterstützt :
    1. Eding CNC.  
Das Plugin beinhaltet Funktionen die das Steuern der Bewegungen der Achsen mit einem Handrad (Track und Velocity Modus) über die vom Hersteller von Eding CNC angebotene API ermöglicht, sowie folgende Funktionen:
      - MDI Befehle ausführen;
      - Not-Stopp auslösen und zurücksetzen;
      - Programm starten, pausieren und stoppen;
      - Auslesen der Koordinaten der Achsen X,Y,Z,A
      - Interne Variablen setzen und auslesen
      - Auslesen der Werte für HeartBeat , Feed, Drehzahl und Override sowie Referenzschalter
      - Setzen der Werte für Override von Feed und Drehzahl
      - Auslesen der Werte für Job-Progress

Funktionen des Plugins sind bereits in dem entsprechenden Profil für LCD Touch Pendant Handrad eingebunden und können zusammen mit dem verwendet werden. Für andere Handräder oder Steuerprogramme müssen vom Benutzer selbstständig Profile angelegt werden.

- Simulation der Drehencoder-Rotation durch Tastenkombinationen - ermöglicht Benutzung des Handrades in Steuerungsprogrammen die kein Handrad-Anschluss unterstützen (soweit Steuerungsprogramm das Fahren der Achsen mit der Tastatur ermöglicht).
  - Modularer Aufbau. Steuerungsspezifische oder anwendungsspezifische Funktionalität ist aus dem Hauptprogramm ausgelagert was Anbindung der neuen Steuerungssoftware oder Anpassungen an neue Softwareversion erleichtert. Das Hauptprogramm muss dafür nicht geändert werden. Verwendung von Tinkerforge Open Source Hardware Komponenten (<http://www.tinkerforge.com/>) ermöglicht einfachen Anschluss von Anzeige- und Eingabegeräten über Module.
  - Offene Schnittstelle für Anbindung der Plugins und Module: eigene Plugins und Module können entsprechend der Schnittstelle entwickelt werden.
  - Test-Anwendung für die Entwicklung der neuen Funktionen und Module sowie Schnittstelle und Templates für Plugins / Module sind verfügbar (nach Anfrage).
  - Konfigurierbarkeit von Handrädern und gesteuerten Programmen über GUI
  - Möglichkeit komplexe Abläufe aus einer Reihe von verfügbaren Funktionen wie: Selbstdefinierte Tastenkombinationen;
    - MDI Befehle;
    - Plugin/Modul-Funktionen;
    - Verwendung der internen Variablen (500 Variablen)
    - Befehle mit Bedingung (IF – THEN - ELSE)
    - Setzen von Bildelementen des LCD Touch Handrades (LED's, RO's, Progressbalken)
    - Ändern des aktuellen Bildschirms des LCD Touch Handrades
- zu erstellen und mit einer Taste des Handrades (oder durch eine andere Möglichkeit, z.b. automatische Zeitintervall-Ausführung) auszulösen (nur wenn die entsprechende Funktionalität von dem Steuerungsprogramm angeboten wird). Die Funktionen können zusammen kombiniert werden und diese Kombinationen wiederum in anderen Funktionen verwendet werden (Funktions-Pools).
- Bequeme Zuordnung der Funktionen / Funktions-Pools über die Benutzeroberfläche
  - Möglichkeit selbsterstellte Funktionen des Programms im Intervall ausführen zu lassen (z.b. jede Sekunde)

- Möglichkeit Variablen zu initialisieren und Werte den Bildschirm-Elementen des LCD Touch Handrades zuzuordnen
- Möglichkeit der Verwaltung von mehreren Handrädern
- Möglichkeiten der Veränderung/Anpassung der Parameter für das Verhalten der Bewegung der Achsen beim Drehen des Handrades wie:
  - Umkehren der Bewegungsrichtung;
  - Definieren der Geschwindigkeiten und Abstufungen;
  - Anpassung des Ansprechverhaltens;
- Override-Funktionen können angelegt werden (falls das Steuerungsprogramm es über API erlaubt)
- 2/4 Takt Modus (LCD Touch Handrad): Bewegung der Achsen mit durch das Drehen des Handrades ausgewählter Geschwindigkeit beim Halten einer Taste (2-Takt Modus) oder durch Umschaltung ( 4-Takt Modus)
- Möglichkeit der Verwaltung von mehreren Steuerungsprogrammen
- Die gesamte Konfiguration wird in Form von XML-Dateien abgespeichert – die bei Bedarf auch manuell bearbeitet und gesichert werden können.

## 2.2 Plugins

Plugins sind Bibliotheken (dll's) die Spezifische Funktionalitäten für Steuerprogramme zur Verfügung stellen. Viele der Steuerprogramme bieten den Entwicklern Schnittstellen (API) zur Entwicklung der Erweiterungen welche aus den Plugins angesprochen werden können.

Alle Plugins implementieren eine universelle Plugin-Schnittstelle und tauschen definierte Datentypen mit dem Hauptprogram aus. Dadurch ist Jog Wheel Communicator für die Zukunft gerüstet da für eine neue Steuerungssoftwareversion (wenn z.b. API vom Hersteller geändert wurde) nur Austausch des Plugins notwendig ist (und somit Kompatibilität zu allen Versionen, auch veralteten, gegeben ist.). Plugins können natürlich auch selbstständig geändert oder erstellt werden. Zum Testen existiert eine Testapplikation. Einer gesteuerten Applikation können mehrere Plugins zugeordnet werden, ein Plugin kann aber auch Funktionen aus mehreren Bereichen beinhalten ( z.b. Daten auslesen, Befehle weitergeben, Daten abspeichern).

Momentan existieren folgende Plugins:

- EDING CNC Plugin

## 2.3 Module

Module sind eine Art der Plugins und sind in erster Linie zur Anbindung externer Hardware gedacht. Module sind im Gegensatz zu Plugins Steuerprogramm-Unabhängig und können in allen Applikationen und mit allen Handrädern genutzt werden. Ein Beispiel dafür ist DRO-Modul zur Anzeige der Achsen-Koordinaten.

Module verwenden die gleiche Plugin-Schnittstelle und können auch mit der Testapplikation getestet werden. Ein Modul bedient dabei nur eine Komponente oder Hardware-Modul.

Module können durch das JWC Programm selber konfiguriert werden. Eine Eigenschaft der Module erlaubt es die Module (unter anderen Namen) zu kopieren und somit mehrere Module gleicher Art zur selben Zeit zu verwenden.

Module (im Gegensatz zu den Plugins) können aktiviert oder deaktiviert werden. Das erlaubt z.b. einer Taste Funktionen unterschiedlicher Module zuzuordnen die in Abhängigkeit vom Modul-Aktivierungsstatus gestartet werden.

Momentan existieren folgende Module:

- Universelles DRO Anzeigemodul für X,Y,Z,A Koordinaten, 8 Stellen pro Koordinate (4 Vor- und 4 Nachkommastellen)

## 3 Einrichtung von Jog Wheel Communicator

Da es unbekannt ist mit welchem Handrad und Steuerungssoftware der Benutzer ausgestattet ist muss Jog Wheel Communicator einmal für die verwendete Software/Hardware eingerichtet werden. Mitgeliefert wird eine Konfiguration für aktuelles „LCD Touch Pendant“ V8 (cnc-technics.de) Handrad ausgestattet mit Drehencoder mit Rasterung und 100 Schritten pro Umdrehung.

Des Weiteren muss das verwendete Steuerungsprogramm eingerichtet werden. Mitgeliefert wird das Plugin und Konfiguration zur Verwendung mit EDING CNC, getestet mit der Version V4.02.27

Diese Konfigurationen dienen als Beispiel und können geändert oder eigene Konfigurationen auf Basis deren erstellt werden.

Im Folgenden wird der Prozess der Konfiguration, Erstellung der Funktionen und deren Zuordnung ausführlich beschrieben. Um den Konfigurationsprozess besser zu verstehen schauen Sie sich das Kapitel 6 „Anwendungsbeispiele“ an. Das mitgelieferte Profil für Eding CNC und LCD Touch Pendant ist im Kapitel 7 beschrieben.

### 3.1 Installation von Jog Wheel Communicator

Die folgenden Voraussetzungen müssen erfüllt werden Um Jog Wheel Communicator ausführen zu können:

- Microsoft .NET Framework 4.0 muss auf dem Rechner installiert sein. Bitte zuerst installieren.
- Wenn Tinkerforge Bausteine und Module verwendet werden, müssen „Brick deamon“ und „Brick viewer“ installiert sein, Download der jeweils letzter Version von der Webseite [„http://www.tinkerforge.com/de/doc/Downloads.html“](http://www.tinkerforge.com/de/doc/Downloads.html)
- Die für den Betrieb notwendige Bibliothek „Tinkerforge.dll“ befindet sich in dem Packet „C# Beispiele und Bindings“ und kann ebenfalls auf der oben angegebenen Seite heruntergeladen werden. Während der Installation wird Setup nach dem Verzeichnis gefragt wo diese Dll zu finden ist.
- Die Konfigurationsdatei des LCD Touch Pendant Handrades „LCD\_config.xml“ muss aus dem Handrad in das JWC Verzeichnis kopiert werden falls eine andere als die mitgelieferte Version der Screensets verwendet wird. Mitausgelieferte Version der Oberfläche für Handrad (im Verzeichnis „JWC\_ScreenSet“) kann auch verwendet werden, in diesem Fall ist die existierende „LCD\_config.xml“ Dateien beizubehalten und die Oberfläche auf dem Handrad zu installieren.
- Falls Eding CNC angesteuert werden soll wird bei der Installation nach dem Eding CNC Installationsverzeichnis gefragt.

Alle anderen Komponenten werden mit Jog Wheel Communicator in Form einer Setup-Datei ausgeliefert.

Setup erzeugt die für das Programm notwendige Verzeichnisstruktur und kopiert notwendige Dateien.

Alle Konfigurationen sind als XML-Dateien gespeichert so dass leicht Sicherungskopien der Konfiguration erstellt werden können. Das wird auch dringend empfohlen denn aufgrund der möglichen Komplexität ungeübt die Konfiguration leicht beschädigt werden kann.

**Achtung !** Erneute Installation in dasselbe Verzeichnis überschreibt alle Dateien – Daher wird dringend geraten eine Sicherung zu machen falls bereits gemachte Einstellungen nicht verloren gehen sollen.

#### 3.1.1 Installation für EDING CNC (Als Beispiel für API-Verwendung )

Da EDING CNC Plugin auf die EDING CNC- spezifische Bibliothek zugreift müssen die folgenden Dateien in das Verzeichnis „**JogWheelCommunicator**“ aus dem EDING CNC- Installationsverzeichnis kopieren:

- „cncapi.dll“
- „cnc.ini“

Dieser Schritt wird bei der Installation automatisch durchgeführt, aber wenn man die Dateien nicht bei der Installation kopiert oder eine neue Eding CNC Version installiert wird – muss dieser Schritt manuell erfolgen.

Dadurch können die Plugins die Verbindung zu dem EDING CNC Server aufbauen. Falls dies nicht gemacht wird oder eine andere Version des Programms EDING CNC installiert ist, wird möglicherweise die Kommunikation mit dem EDING CNC Server nicht funktionieren.

Wenn nur Tastatur-Simulationsmodus verwendet wird - braucht man diese Dateien nicht, in dem Fall muss aber dann der EDING CNC Plugin aus dem „Plugins“ - Verzeichnis gelöscht werden, sonst kommt es zum Fehler beim Start des Programms (das ist nur notwendig wenn EDING CNC Plugin zuvor installiert worden ist).

## 3.2 Lizenz und Demo-Modus

Jog Wheel Communicator ist keine freie Software, zum Betrieb ist eine gültige Lizenz notwendig. Diese wird nach dem Kauf personalisiert erstellt und dem Käufer zugeschickt.

Ohne gültige Lizenz startet das Programm in einem Demo-Modus. Die Beschränkungen des Demo-Modus sind wie folgt beschrieben:

- nur Achsen X und Z lassen sich verfahren
- nach fünf Minuten wird jegliche Achsen-Bewegung gestoppt und eine Erinnerungsmeldung angezeigt. Nach Bestätigung der Meldung kann für weitere fünf Minuten getestet werden.
- Alle Module sind deaktiviert

Mit der Demo-Version kann aber trotzdem festgestellt werden ob die Software den Anforderungen des Benutzers entspricht und so wie gewünscht funktioniert. Testphase ist daher sinnvoll und wird ausdrücklich empfohlen.

## 3.3 Ordnerstruktur

Nachfolgend wird die Ordnerstruktur des Programms beschrieben.

„**JogWheelCommunicator**“ - das ist das Hauptverzeichnis des Programms. Hier sind die ausführbaren Programme, die notwendigen Bibliotheken und Konfigurationsdateien abgelegt. Hier liegt auch die Lizenz-Datei („JW\_license.xml“) und hier wird die Log-Datei erstellt („JogWheelCommunicator.log“).

Besonderheit: in diesem Verzeichnis liegt auch die Definitionsdatei der Handrad-Screens. die Datei muss den Namen „LCD\_config.xml“ haben und muss die Gleiche sein die auf der SD-Karte des „LCD Touch Pendant“ Handrades zu finden ist. Das ist der Fall wenn man die Mitgelieferte Oberfläche für das Handrad verwendet.

„**JogWheelCommunicator / Plugins**“ - hier sind die verfügbaren Plugins und Module abgelegt

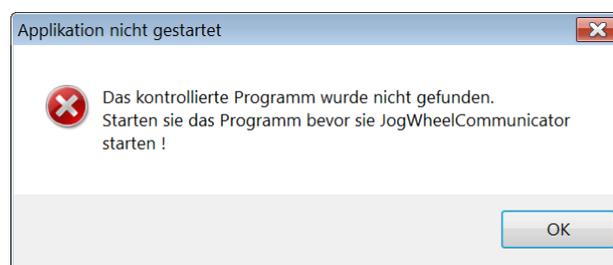
## 3.4 Start von Jog Wheel Communicator

Zum Starten von Jog Wheel Communicator doppelklicken auf der Datei „JogWheelCommunicator.exe“.

Das Programm sollte mit Administrator-Rechten gestartet werden.

Das Programm startet, lädt Konfiguration und überprüft die Lizenz. Falls zuvor bereits gesteuerte Anwendung eingerichtet worden war wird dieses aktiviert und in Vordergrund gebracht, das Hauptfenster von Jog Wheel Communicator öffnet sich dann im Hintergrund und kann über die Taskleiste aktiviert werden.

Falls gesteuerte Anwendung nicht gefunden wurde (weil es falsch eingerichtet oder nicht gestartet wurde) wird eine entsprechende Fehlermeldung angezeigt. Nach Bestätigung wird das Hauptfenster von Jog Wheel Communicator angezeigt.



**Achtung !** Wenn Jog Wheel Communicator in dem Tastatur-Simulations-Modus verwendet wird oder Funktionen die Tastatur-Befehle verwenden (z.b. um Jog-Modus zu ändern) ist es zwingend notwendig dass das CNC-Steuerungsprogramm im Vordergrund und aktiv ist, sonst gehen die Tastatur-Befehle auf das jeweils aktivierte Fenster und haben nicht die beabsichtigte Wirkung !

Beim ersten Start des Programms sind „LCD Touch Pendant“ als Handrad und „EDING CNC“ als gesteuerte Anwendung ausgewählt. Falls Sie andere existierenden Einstellungen ansehen oder verwenden möchten können Sie die auswählen, ansonsten kann eigene Konfiguration erstellt werden.

Beispiel des Hauptbildschirms des Programms:



### 3.5 Elemente des JWC Hauptfensters

Im Folgenden werden die einzelnen Elemente des Hauptfensters des Programms beschrieben. Von hier aus können sämtliche Einstellungen vorgenommen sowie Handräder und gesteuerte Applikationen verwaltet werden.

Des Weiteren gibt das Hauptfenster den Überblick über die wichtigsten Daten und Meldungen des Programms.

#### „Status“ - Bereich

Hier stehen die allgemeinen Daten zum aktuellen Programmzustand:

- **„Gesteuerte Anwendung“** - das Programm / Anwendung das aktuell gesteuert wird
- **„Lizenz“** - hier steht der Name des Lizenznehmers oder „Demo-Mode“ falls JWC ohne gültige Lizenz gestartet wurde.
- **„Maschinenmodus“** - Hier steht das aktuell verwendete Steuerungsverfahren der Anwendung (CNC Maschine). Zwei Modi werden unterschieden: **„Nativ“** - Modus zur Steuerung über Funktionen die API Schnittstelle nutzen und über Plugin implementiert sind. **„Tastatur-simuliert“** - Modus zur Steuerung über Tastenkombinationen. Das Drehen des Drehrades am Handrad wird durch Abfolge der Tastenkombinationen simuliert so dass sich die Achsen-Bewegung in etwa so aussieht wie mit einem echten MPG Handrad Anschluss.

„In etwa“ heisst dass es nicht möglich ist diese Simulation genau so fein dosierbar zu machen wie es mit einem direkten Handradanschluss möglich wäre, es sind, abhängig von der gesteuerten Anwendung bis zu drei Beschleunigungsstufen definierbar: Langsam, Mittel und Schnell die in Abhängigkeit der Drehgeschwindigkeit automatisch ausgewählt werden – das hängt aber von dem ab was die entsprechende Anwendung ermöglicht.

Der Einzelschritt-Modus funktioniert dagegen wie gewohnt.

#### „Handrad-Status“ - Bereich

Hier stehen die Verbindungsdaten und der aktuelle Zustand der Verbindung des Handrades. Die Daten sind unterschiedlich, abhängig vom Typ des Handrades.

#### „Verbindung“ - Bereich

Hier ist eine Schaltfläche „Alles neu verbinden“ platziert. Es dient dem Wiederaufbau aller Verbindungen nachdem z.b. ein Handrad aus- und wieder eingesteckt worden ist. Normalerweise werden alle Verbindungen beim Start des JWC Programms aufgebaut daher kann man stattdessen auch einfach das Programm neu starten.

**„Konfiguration“ - Bereich**

Hier sind mehrere Schaltflächen sowie Auswahl-Drop-Down Boxen platziert. Durch die können das aktuelle Handrad sowie die gesteuerte Anwendung ausgewählt werden. Die Auswahl wird automatisch gespeichert sodass beim nächsten Start die zuletzt ausgewählten Werte verwendet werden.

Durch die Schaltflächen gelangt man zu den entsprechenden Konfigurationsbereichen, so können Handräder und gesteuerte Applikationen verwaltet sowie die Konfiguration des ausgewählten Handrades bearbeitet werden.

**„Module“ - Bereich**

Hier ist eine Schaltfläche „Module konfigurieren“ platziert. Hier können Module aktiviert, deaktiviert oder deren Eigenschaften bearbeitet werden.

**„Letzte Nachricht“ - Bereich**

In diesem Bereich werden wichtige Nachrichten oder Fehler angezeigt. Das Feld zeigt nur jeweils die letzte Nachricht in Echtzeit, Wenn tiefere Fehleranalyse notwendig ist kann Logging aktiviert oder Echtzeit-Logging -Fenster geöffnet werden (Schaltfläche rechts).

**Logging**

Log-Einträge werden in die Date „JogWheelCommunicator.log“ geschrieben, es befindet sich im selben Verzeichnis wie das JWC Programm.

Logging ist standardmäßig nur für schwere Fehler aktiviert, im Falle einer Fehleranalyse ist es aber unter Umständen notwendig mehr Informationen zu bekommen. Diese Einstellung kann in der Datei „JogWheelCommunicator.exe.config“ geändert werden (ebenfalls m selben Verzeichnis wie das JWC Programm). Hier gibt es zwei Einträge:

```
<levelMin value="FATAL" />
<levelMax value="FATAL" />
```

Mögliche Werte (absteigend der Wichtigkeit sortiert):

FATAL	- schwere Fehler
ERROR	- normale Fehler
WARN	- Warnungen
INFO	- Informationen
DEBUG	- Informationen zur tieferen Fehleranalyse

Dass heisst z.b. wenn Sie alle möglichen Log-Einträge haben wollen müssen Sie den Wert von „levelMin“ wie folgt ändern :

```
<levelMin value="DEBUG" />
```

Diese Einstellung ist allerdings im Betrieb zu vermeiden denn die Grösse der Logdatei wird dadurch ständig stark anwachsen was sich negativ auf Geschwindigkeit (Performance) der Ausführung auswirkt.

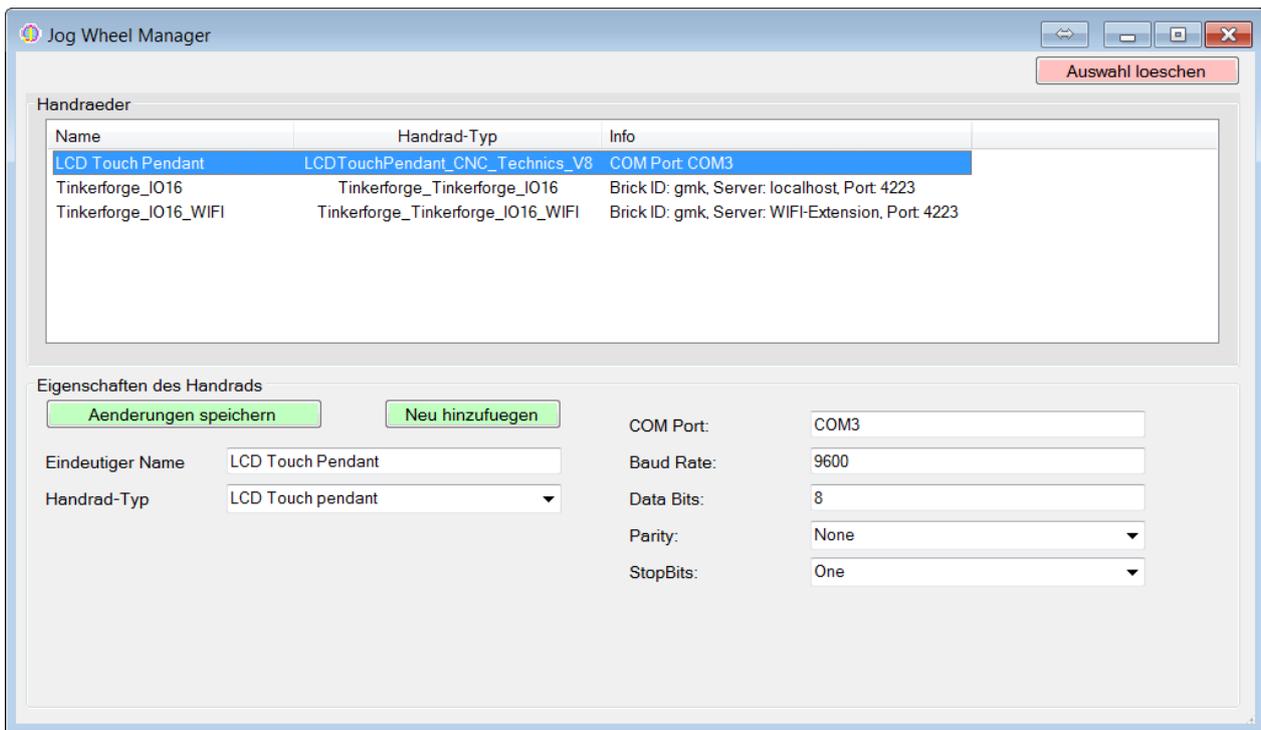
**Achtung !** Eine unbedachte manuelle Änderung der Konfigurationsdateien kann die Konsistenz der Konfiguration zerstören und sollte nur von erfahrenen Benutzern vorgenommen werden. Vor jeder Änderung wird ein Backup dringend empfohlen !

## 3.6 Verwaltung und Erstellung der Handräder

Zum Bearbeiten bereits erstellter Handräder oder zum Erstellen neuer wählen Sie die Schaltfläche


 A rectangular button with a light blue background and a thin border, containing the text "Handrad-Verwaltung".

auf dem Hauptbildschirm. Es öffnet sich ein neues Fenster:



Alle Handrad-Einstellungen können im Prinzip auch manuell bearbeitet werden und sind in der XML-Konfigurationsdatei „LCDJogWheels.xml“ abgelegt.

**Achtung !** Aufgrund der Komplexen Beziehungen innerhalb der Konfiguration kann eine unbedachte manuelle Änderung die Konsistenz der Konfiguration zerstören und sollte nur von erfahrenen Benutzern vorgenommen werden. Vor jeder Änderung wird ein Backup dringend empfohlen !

Sie können nun ein neues Handrad erstellen, ein bestehendes bearbeiten oder löschen.

Zum Erstellen eines neuen Handrades wählen Sie zuerst über „Handrad-Typ“ den Typ des neu zu erstellen Handrades. Nach der Auswahl werden die dem Typ entsprechenden Eigenschafts-Felder eingblendet. Nun setzt man die gewünschten Werte ein, trägt unter „Eindeutiger Name“ die Bezeichnung für das Handrad ein und klickt anschließend auf „Neu hinzufügen“. Das Handrad soll nun in der Liste angezeigt werden.

Zum Ändern der Werte eines Handrades soll zuerst das entsprechende Handrad in der Liste ausgewählt werden. Nach der Auswahl erscheinen die Konfigurations-Werte des Handrades in den entsprechenden Feldern die man nun bearbeiten kann. Anschließend klickt man zum Speichern auf „Aenderungen speichern“.

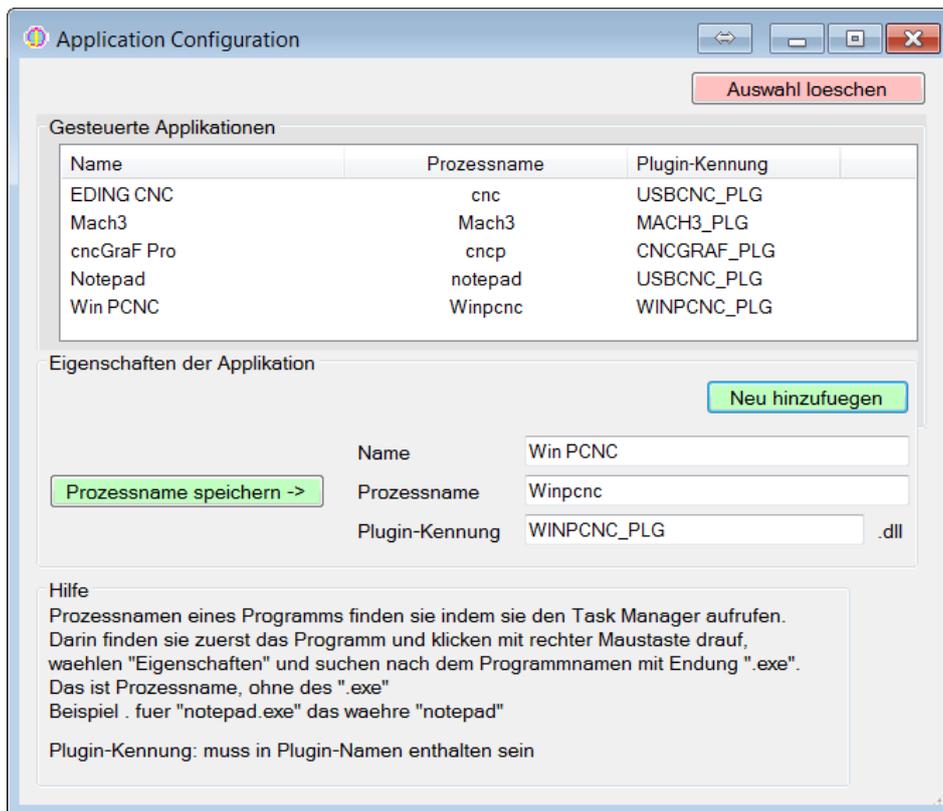
Achtung ! Typ des Handrades kann nicht bearbeitet werden! Änderungen in diesem Feld werden nicht gespeichert!

Zum Löschen eines Handrades soll zuerst das entsprechende Handrad in der Liste ausgewählt werden. Anschliessend klickt man auf „Auswahl loeschen“. Den Vorgang muss man zweimal bestätigen denn das Löschen eines Handrades bedeutet auch das Löschen aller mit dem Handrad assoziierten Funktionen und Zuordnungen, versehentliches Löschen kann also eine Menge Arbeit zunichtemachen !

### 3.7 Verwaltung und Erstellung der gesteuerten Applikationen

Zum Löschen erstellter Applikationen oder zum Erstellen neuer wählen Sie die Schaltfläche

 auf dem Hauptbildschirm. Es öffnet sich ein neues Fenster:



Zum Erstellen einer neuen gesteuerten Applikation setzt man die gewünschten Werte ein und klickt anschließend auf „Neu hinzufügen“. Die Applikation soll nun in der Liste angezeigt werden. Hinweise zum Prozessnamen und Plugin - Kennung finden sich in dem Fenster im Abschnitt „Hilfe“.

Zum Löschen einer gesteuerten Applikation soll zuerst die entsprechende Applikation in der Liste ausgewählt werden. Anschließend klickt man auf „Auswahl loeschen“. Den Vorgang muss man zweimal bestätigen denn das Löschen einer Applikation bedeutet auch das Löschen aller mit der Applikation assoziierten Funktionen und Zuordnungen, versehentliches Löschen kann also eine Menge Arbeit zunichtemachen !

Zum Ändern von Prozessnamen einer gesteuerten Applikation soll zuerst die entsprechende Applikation in der Liste ausgewählt werden. Anschließend ändert man den Prozessnamen und klickt man auf „Prozessnamen speichern ->“.

Alle Anwendungs-Einstellungen können im Prinzip auch manuell bearbeitet werden und sind in der XML-Konfigurationsdatei „LCDControlledApplications.xml“ abgelegt.

**Achtung !** Aufgrund der Komplexen Beziehungen innerhalb der Konfiguration kann eine unbedachte manuelle Änderung die Konsistenz der Konfiguration zerstören und sollte nur von erfahrenen Benutzern vorgenommen werden. Vor jeder Änderung wird ein Backup dringend empfohlen !

### 3.8 Handrad-Parameter Konfiguration

Durch Anklicken der Schaltfläche „Handrad-Parameter Konfiguration“ gelangt man in den wichtigsten Bereich der Applikation - Einstellung von Handradspezifischen Parameter und Erstellung der Funktionen. Grob kann man die Konfiguration in mehrere Bereiche unterteilen:

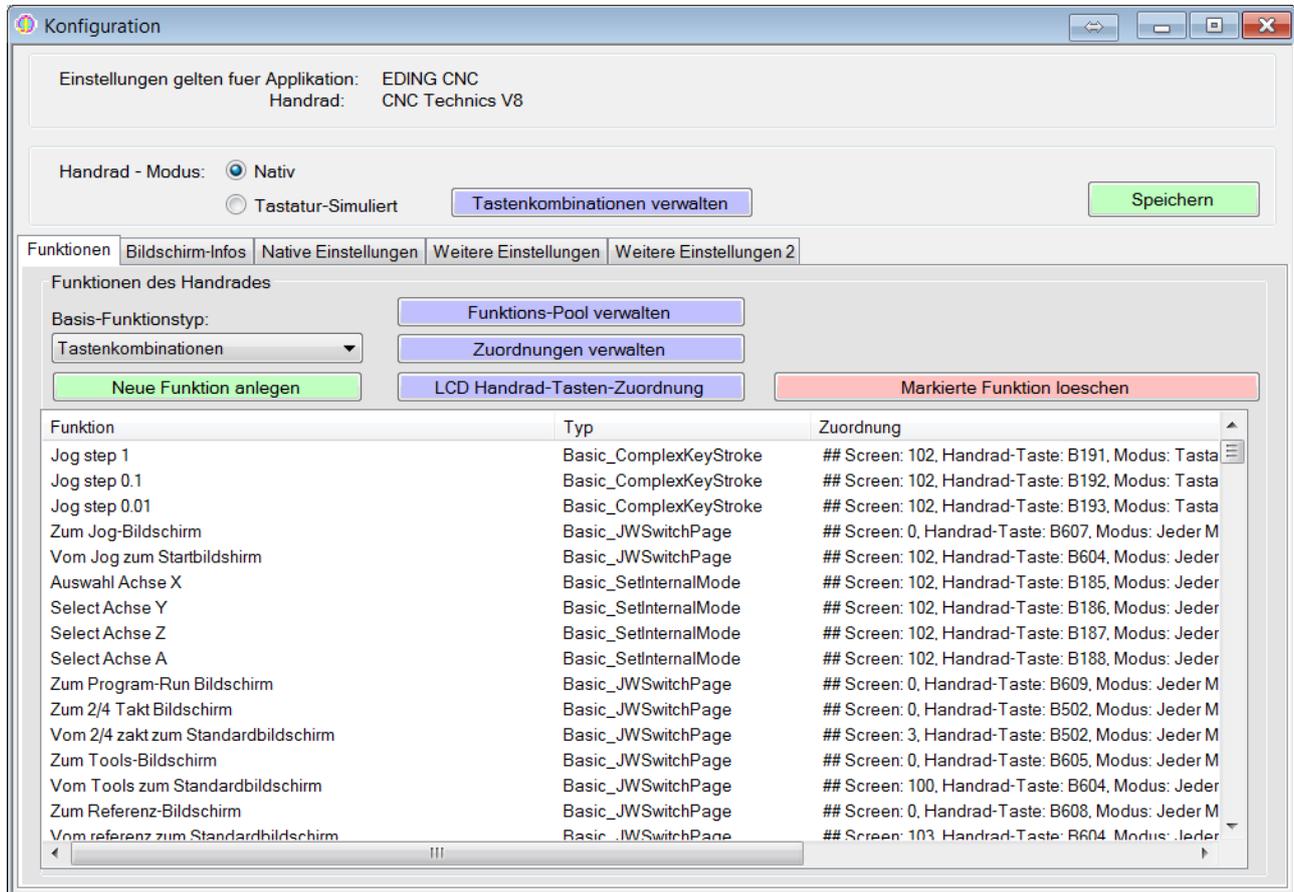
- Erstellung der Tastenkombinationen
- Erstellung der Funktionen
- Zuweisung der Funktionen
- Konfiguration des Tastensimulations-Modus
- Konfiguration des Nativ-Modus
- Zuweisung der Handrad-Elemente (Handradtyp-spezifisch: LCD touch pendant) und Variablen
- Screen-Zusatzinformationen
- Intervallbasierte Ausführung der Funktionen

Diese Bereiche werden in Reitern dargestellt, welche Reiter sichtbar sind entspricht immer dem Typ des Handrades und dem ausgewählten Handrad-Modus.

Ein neues Handrad besitzt, abgesehen von einigen Default-Werten noch keine spezifischen Einstellungen. Die vorzunehmenden Einstellungen hängen im Wesentlichen von dem Typ des Handrades ab, aktuell werden 2 Handradtypen unterstützt:

- LCD Touch pendant
- USB Schnittstellenmodul (Tinkerforge) der es erlaubt einen Drehencoder anzuschliessen (Rechtssignal, A, B, Pegel TTL 3,3V oder 5V) sowie weitere 14 Eingänge für Taster oder Schalter bietet.

Ein Beispiel des Konfigurationsdialogs sehen Sie unten:



### Funktionen / Funktionspools: Definition

Zum weiteren Verständnis wird definiert was man in JWC unter Funktion oder Funktionspool versteht.

Eine Funktion beinhaltet eine definierte Aktion die ausgeführt werden soll. Funktionen haben einen Basis Typ, je nach deren Bestimmung.

Ein Funktionspool ist eine definierte Abfolge von Funktionen und ermöglicht komplexe Abläufe zu realisieren.

Unten wird die prinzipielle Vorgehensweise beim Einstellen eines neuen Handrades beschrieben, Einzelheiten werden in den nachfolgenden Kapiteln behandelt:

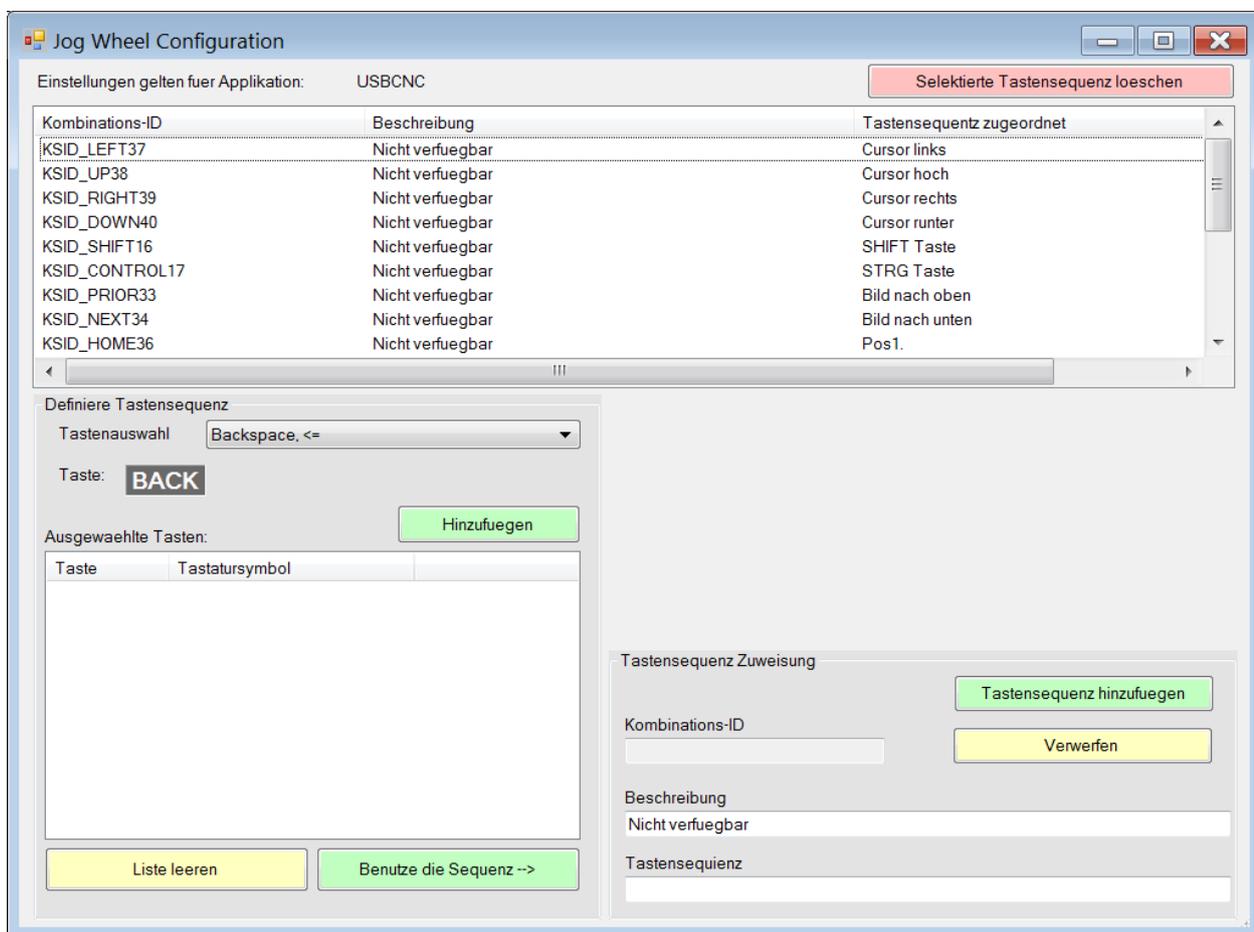
1. Wenn das Handrad im Tastensimulations-Modus betrieben werden soll müssen zuerst die verwendeten Tasten- und Kombinationen angelegt werden. Tastenkombinationen können auch im Nativ-Modus verwendet werden, falls benötigt.
2. Für Tastensimulations-Modus müssen Einstellungen für die Simulation vorgenommen werden, unter anderen die Zuordnung der Tasten zu den Achsenbewegungen.
3. Wenn das Handrad im Native-Modus verwendet wird, sind die Native-Modus Parameter einzustellen, vor allem die Plugin-Funktion die Signale vom Encoder an die API Funktion der gesteuerten Applikation weiterleitet und andere benötigte Parameter (z.b. Anzahl der Schritte des Encoders pro Umdrehung).
4. Weiter müssen die benötigten Funktionen und deren Zusammenfassungen (Funktion-Pools) erstellt werden, dabei hat man die Flexibilität auch sehr komplexe Abläufe festzulegen. Möglichkeiten dabei:
  - Variablen Werte zuweisen (z.b. aus anderen Variablen oder Rückgabewert einer Funktion)
  - Auswertungen vorzunehmen (Wenn-Dann-Sonst) und in Abhängigkeit davon Funktionen starten
  - Plugin- oder Modul-Funktionen verwenden

- MDI Eingaben (auch komplexe, mehrzeilige) erstellen
  - (LCD Touch pendant) Seiten, DROs, LEDs, Progress Balken und andere Elemente des Handrades setzen
5. Funktionen (oder Pools) müssen jetzt den Tasten/Schaltern am Handrad zugeordnet werden. Dabei kann man die entsprechende Funktion einfach per Knopfdruck am Handrad automatisch zuweisen.
  6. Anschließend können die erweiterten Einstellungen vorgenommen werden. Dazu zählen:
    - Not-Aus Funktion festlegen
    - Initialisierung und Zuordnung der Screen-Elemente (LCD Touch pendant) oder Initialisierung der Variablen
    - Zyklische Ausführung von Funktionen (Intervallbasierte Ausführung) um z.B. DRO Koordinaten auszulesen und anzuzeigen.

Die Reihenfolge der beschriebenen Schritte ist nicht zwingend und kann beliebig verändert werden, nur, logischerweise kann man erst Funktionen zuordnen wenn man die angelegt hat. Nachträgliche Bearbeitung der Reihenfolge der Ausführung und hinzufügen/löschen der Funktionen ist ebenfalls möglich, ebenso die Bearbeitung der Zuordnungen.

### 3.8.1 Tastenkombinationen anlegen

Um Tastenkombinationen anzulegen / bearbeiten klicken Sie in dem Konfigurationsfenster auf die Schaltfläche **Tastenkombinationen verwalten**. Dabei wird ein neues Dialogfenster geöffnet:



Zur Auswahl haben Sie alle Tasten einer normalen Tastatur zur Verfügung und können beliebig komplexe Kombinationen (Sequenzen) erstellen. Dabei entspricht z.B. eine Tastenkombination „STRG + A“ dem folgenden Ablauf:

Taste „STRG“ wird gedrückt und gehalten

Taste „A“ wird gedrückt und losgelassen

Taste „STRG“ wird losgelassen

Die Reihenfolge der Auswahl (von Oben nach Unten) entspricht der Reihenfolge der Tastenbetätigungen:

Ausgewaehlte Tasten:	
Taste	Tastatursymbol
CONTROL	STRG Taste
VK_A	Taste A

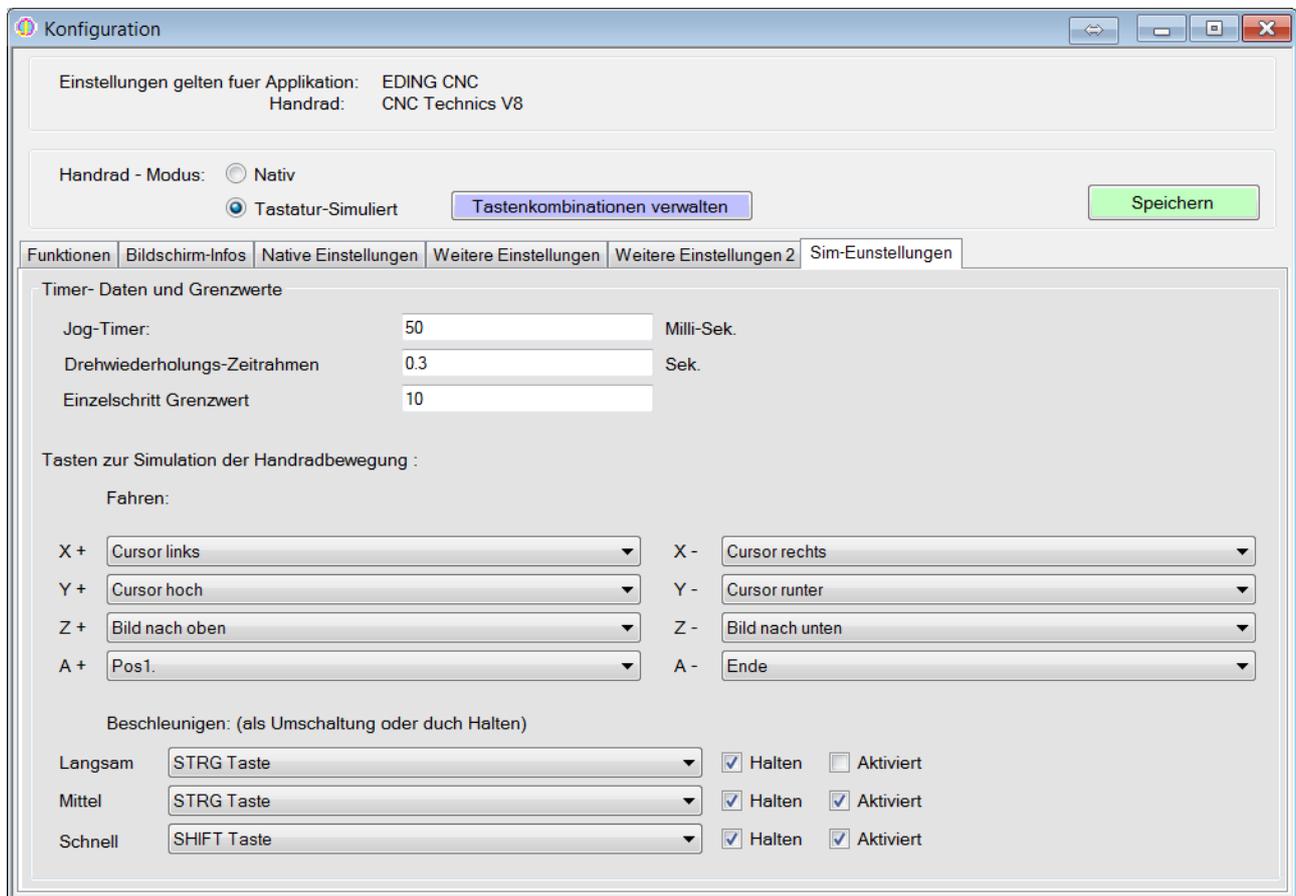
Die erstellte Tastenkombination wird dann beim Klicken auf „Tastensequenz hinzufuegen“, mit einem Namen versehen, abgespeichert.

### Löschen der Tastenkombination

Zum Löschen einer Tastenkombination wählen Sie den entsprechenden Eintrag in der Liste und klicken „Selektierte Tastensequenz loeschen“

## 3.8.2 Simulation - Tastendefinitionen

Die Einstellungen für die Simulation der Drehbewegung des Encoders mit Tasten werden in dem Reiter „Sim-Einstellungen“ festgelegt. Aktuell können 4 Achsen und 3 Beschleunigungsstufen simuliert werden. Bei Beschleunigungsstufen, wenn das Häkchen bei „Halten“ gesetzt ist wird die zugeordnete Taste gedrückt gehalten solange die Beschleunigungsstufe gilt, ansonsten wird nur ein Tastendruck simuliert (Umschalten) aber auch nur wenn die entsprechende Beschleunigungsstufe aktiviert ist (Häkchen „Aktiviert“). Unten ist ein Beispielbildschirm abgebildet:



Die Entscheidung, wann zu der jeweiligen Beschleunigungsstufe umgeschaltet wird trifft das Programm aufgrund der Einstellungen die in dem Reiter „Simulations-Einstellungen“ festgelegt werden.

### 3.8.3 Simulations-Einstellungen

Aufgrund der unterschiedlichen subjektiven Anforderungen der Benutzer und eingesetzter Hardware (Handräder) ist es oft notwendig optimale Parameter für Tastenkombinations-Simulation zu finden.

Nachfolgend ist ein Beispielbildschirm der Grundeinstellung dargestellt:

Timer- Daten und Grenzwerte		
Jog-Timer:	50	Milli-Sek.
Drehwiederholungs-Zeitrahmen	0.3	Sek.
Einzelschritt Grenzwert	10	

Die Parameter im Bereich „Timer-Daten und Grenzwerte“ haben Einfluss auf die Geschmeidigkeit der Bewegung der Achsen und die Drehgeschwindigkeit ab wann zum Einzelschritt-Modus gewechselt wird. Die Drehgeschwindigkeit des Handrades ist ein abstrakter Wert, hat realen Vorschub-Werten nichts zu tun und kann bei unterschiedlichen Handrädern variieren.

- „Jog-Timer“ - (in Millisekunden) - über diesen Wert wird festgelegt im welchen Intervall eine Funktion gestartet wird die entscheidet ob und mit welcher Geschwindigkeit die Bewegung der ausgewählten Axis der Maschine erfolgt. Dieser Wert muss in Kombination mit dem „Drehwiederholungs-Zeitrahmen“-Wert ermittelt werden und hängt von Faktoren wie PC-Leistung und dem eingesetzten Handrad ab.
- „Drehwiederholungs-Zeitrahmen“ - (in Sekunden) - dadurch wird festgelegt nach welcher Zeit die aktuelle Bewegung der Maschine gestoppt wird wenn die Drehung des Handrades aufhört (also keine Drehgeschwindigkeit-Werte mehr ankommen). Durch die Einstellung wird auch im Falle des Kabelbruches sichergestellt dass die Maschinen-Bewegung aufhört.
- „Einzelschritt Grenzwert“ - wenn der Wert der Drehgeschwindigkeit unter dem eingestellten liegt werden Einzelschritte gefahren, ansonsten geht die Maschine in Dauerlauf-Modus über.

Übrigens, über den Parameter „Aktivierungstastenkombination“ (im Reiter „Weitere Einstellungen“) kann man eine Tastenkombination definieren die beim Starten von JWC automatisch ein Dialogfenster für das manuelle Fahren aktiviert damit die Achsen über Tasten verfahren werden können (z.b. F5 bei Win PC NC). Diese Funktion kann auch abgeschaltet oder für andere Initialisierungen verwendet werden.

Reiter „Weitere Einstellungen“:

Die Parameter im Bereich „Vorschub (Feed)“ definieren die Grenzwerte der Drehgeschwindigkeit des Handrades ab wann die jeweils nächste Geschwindigkeitsstufe verwendet wird. Es besteht die Möglichkeit zur Deaktivierung der Stufen für Steuerprogramme die keine oder begrenzte Möglichkeiten diesbezüglich besitzen (z.b. Mach3 oder Win PC NC - nur langsam oder schnell).

### 3.8.4 Native Modus Einstellungen

Die Einstellungen für Native Modus werden in dem Reiter „Native Einstellungen“ festgelegt und werden im Kapitel „Native Einstellungen“ beschrieben.

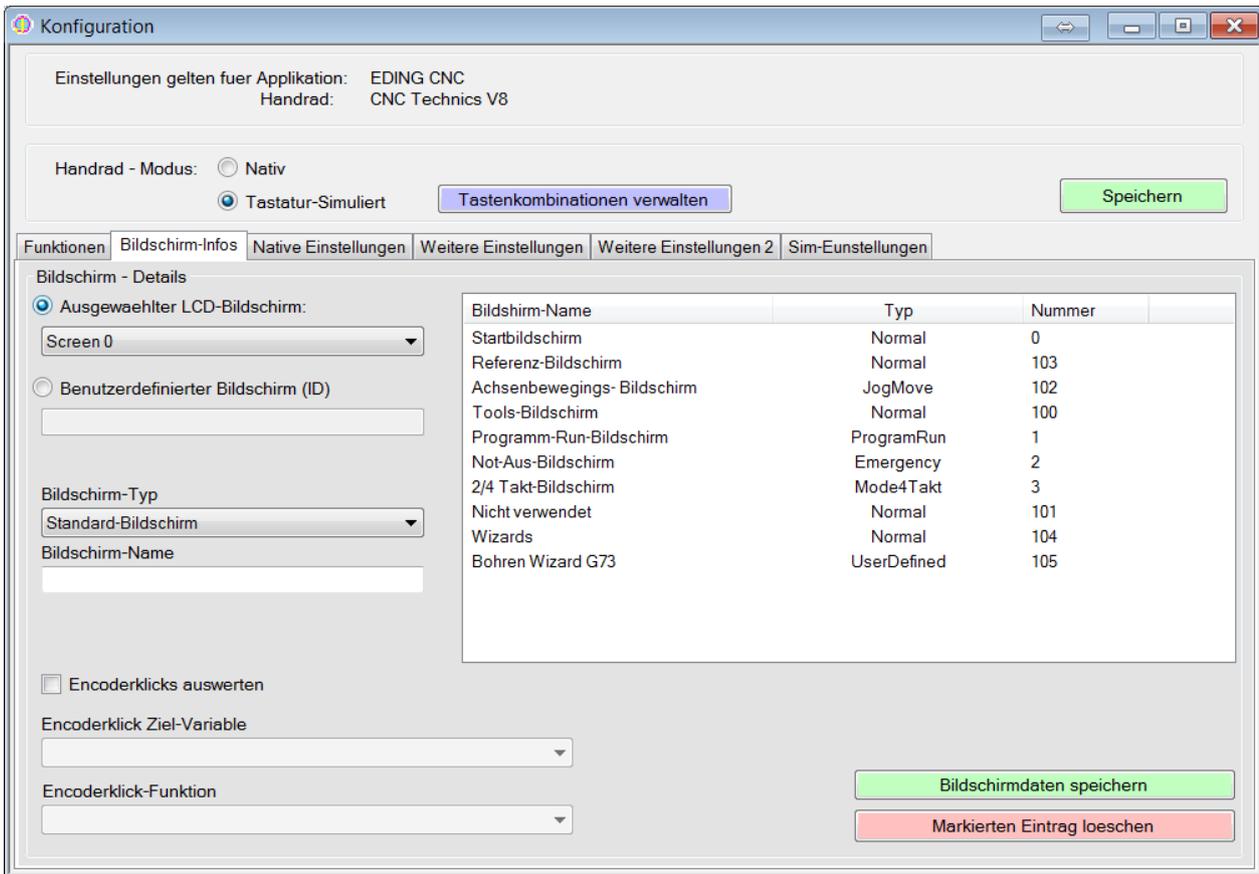
### 3.8.5 Funktionen und Pools erstellen

Die Funktionen und Funktionspools können in dem Reiter „Funktionen“ erstellt, bearbeitet und verwaltet werden. Dies ist beschrieben im Kapitel „Funktionen“

## 3.9 Andere Einstellungen

### 3.9.1 LCD Bildschirm-Zusatzinfos

Dieser Reiter beinhaltet Definitionen für logische Gültigkeitsbereiche, genannt „Screens“ sowie die „LCD Touch Pendant“ - spezifische Einstellungen für die Bildschirme des Handrades. Diese werden entweder manuell angelegt oder aus der Handrad-Konfigurationsdatei ausgelesen wo die „Screens“ nur durch Nummern identifiziert sind. Das ist unbequem bei z.b. der Zuordnung der Funktionen und hier besteht die Möglichkeit zusätzliche Informationen zu den Bildschirmen anzulegen (wie Bildschirm-Typ und Name) oder zu löschen.



Des Weiteren besteht hier die Möglichkeit eine Funktion festzulegen die aufgerufen wird wenn ein Klick von dem Handrad-Encoder auf einem bestimmten Screen ankommt (nur für benutzerdefinierte Bildschirme). Dafür muss man vor dem speichern die Variable einstellen in die Klick-Werte reingeschrieben werden (Encoderklicks-Zielvariable) und eine Funktion festlegen die diese Klicks auswertet (Encoderklick-Funktion).

### 3.9.2 Native Einstellungen

Wenn zur Steuerung der Maschine der Native-Modus verwendet wird (der Modus wo Steuerungsbefehle über Programmspezifische API-Funktionen, realisiert durch Plugins, gesendet werden) dann besteht die Notwendigkeit bestimmte Performance-Parameter einzustellen und festzulegen, welche Plugin-Funktion für die Steuerung verwendet werden soll.

Außerdem werden hier die internen Variablen definiert von wo die Plugin-Funktion ihre Parameter nehmen soll, sowie:

- Einstellungen für Handrad-Mode (Velocity / Tracing);
- Umkehren der Achsen;

**Die Plugin-Jogging -Funktion** muss zwingend den Schnittstellen-Anforderungen für die Steuerungsfunktion entsprechen. Die interne Plugin-Funktions-Schnittstelle sieht wie folgt aus (Beispiel):

```
private void PLUGINAPI_JogMove_Velocity
(int pAxis,
int pFeedAmount,
int Param1,
int Param2,
double Param3,
int pJogMode)
```

- Achse die bewegt werden soll (Automatisch ausgefüllt)
- Wert vom Handrad, das zur Berechnung der Geschwindigkeit und Richtung verwendet wird (Automatisch ausgefüllt)
- z.b. Maximalwert vom Handrad, das zur Berechnung der Geschwindigkeit verwendet wird
- z.b. Latenz, default = 0 (keine Verzögerung)
- z.b. Fließkommazahl (in Millimeter) die besagt wie gross der Schritt „Einzelschritt-Modus“ ist.
- Modus (Einzelschritt, Ununterbrochen u.s.w )

### **Tracking-Modus**

In JWC sind mehrere unterschiedliche Varianten der Auswertung der Encoder-Werte implementiert:

- „Velocity“ - dieser Modus ist sinnvoll mit der Plugin-Funktion „PLUGINAPI\_JogMove\_Velocity“ zu verwenden und implementiert den Ansatz der Drehgeschwindigkeitsauswertung des Encoders. Das hat Vorteile in manchen Bereichen, z.b. beim Fahren im Simulations-Betrieb oder 2/4 Takt Modus.
- „Tracking“ - dieser Modus implementiert die normale MPG Auswertung der Signale wie man es z.b. von Mach3 kennt – basierend auf der Anzahl der Schritte (Klicks) des Encoders. Plugin –Funktion „PLUGINAPI\_JogMove\_Track“ soll für diesen Modus verwendet werden.

Unterhalb stehen die Einstellungen für den jeweiligen Mode.

### **2/4 Takt Modus Beschleunigung**

In diesem Eingabefeld ist definiert wie schnell die Bewegung in dem 2/4 Takt Modus beschleunigt wird je länger man am Encoder dreht. Es sind drei Tupel für Minimum, Mittel und Maximale Geschwindigkeit, definiert wird hier Increment/Dectement in Abhängigkeit zu der aktuellen Geschwindigkeit.

Beim Drehen des Handrades wird die Anzahl der Impulse pro Zeiteinheit regelmäßig gemeldet. Bei jeder solchen Meldung wird die Geschwindigkeit in Abhängigkeit von der Einstellung geändert.

Beispiel:

Die Einstellung „15:1;30:3;max:5“ bedeutet:

- Bus zur Geschwindigkeit 15 wird mit Schritt „1“ hochgezählt (z.b.: 1->2->3...)
- Ab der Geschwindigkeit 15 bis 30 wird mit Schritt „3“ hochgezählt (z.b.: 17->20->23...)
- Ab der Geschwindigkeit 30 wird mit Schritt „5“ hochgezählt (z.b.: 31->36->41...)

**Reverse Richtung Achse ...**

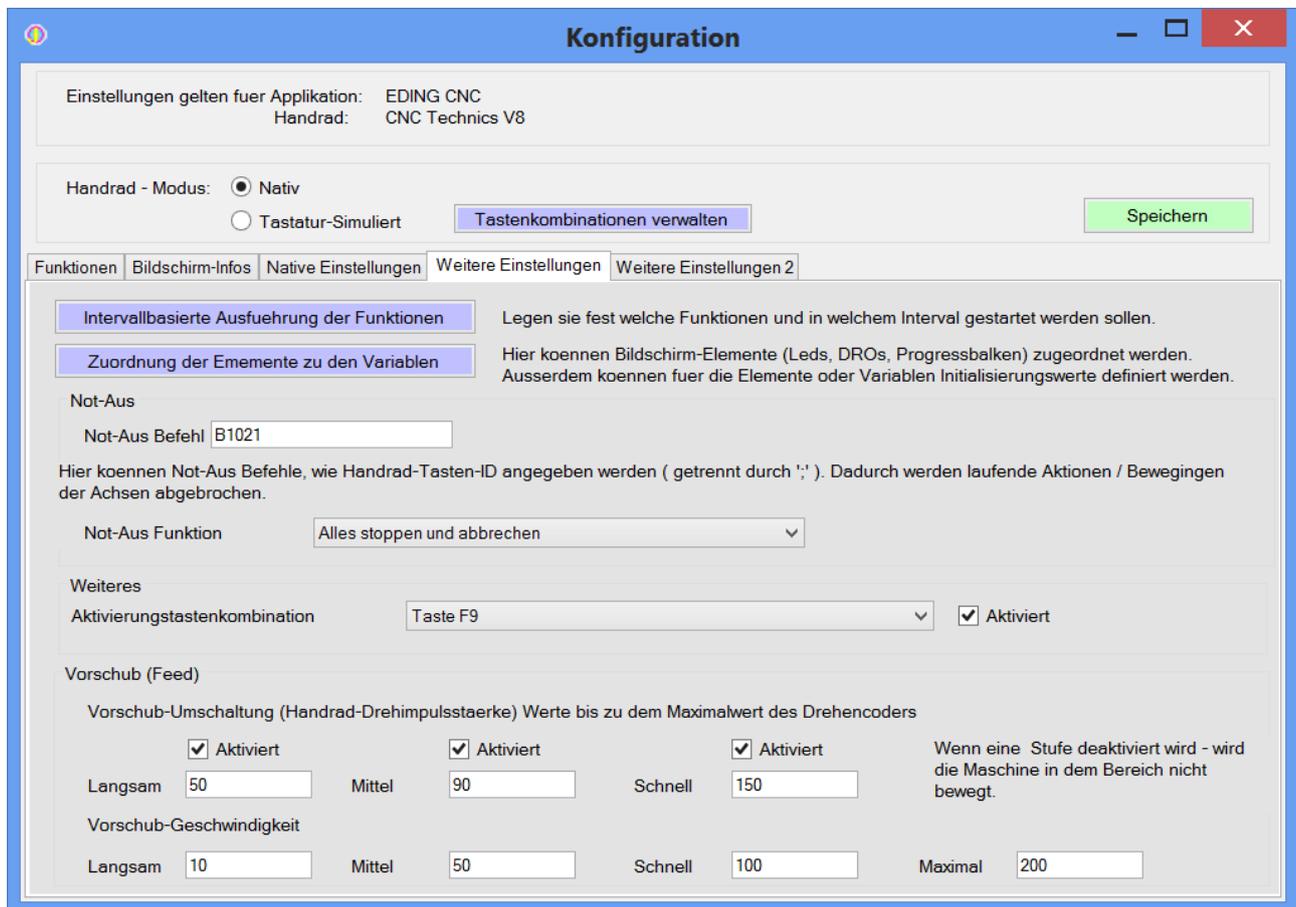
Hier kann man die Bewegung der jeweiligen Achse umkehren.

**Multiplikator**

Dieser Wert wird mit dem Handrad-Encoder-Wert multipliziert um die Geschwindigkeit der Achsenbewegung entsprechend zu steigern. **Achtung:** bei manchen nativen gesteuerten Programmen kommt es darauf an ob solch grosse Werte in der API unterstützt werden, ansonsten sollte hier Multiplikator = 1 stehen) !

**3.9.3 Weitere Einstellungen**

In diesem Reiter sind alle weiteren Einstellungen versammelt die allgemein (für die gesteuerte Applikation) gültig sind.



**Intervallbasierte Ausfuehrung:**

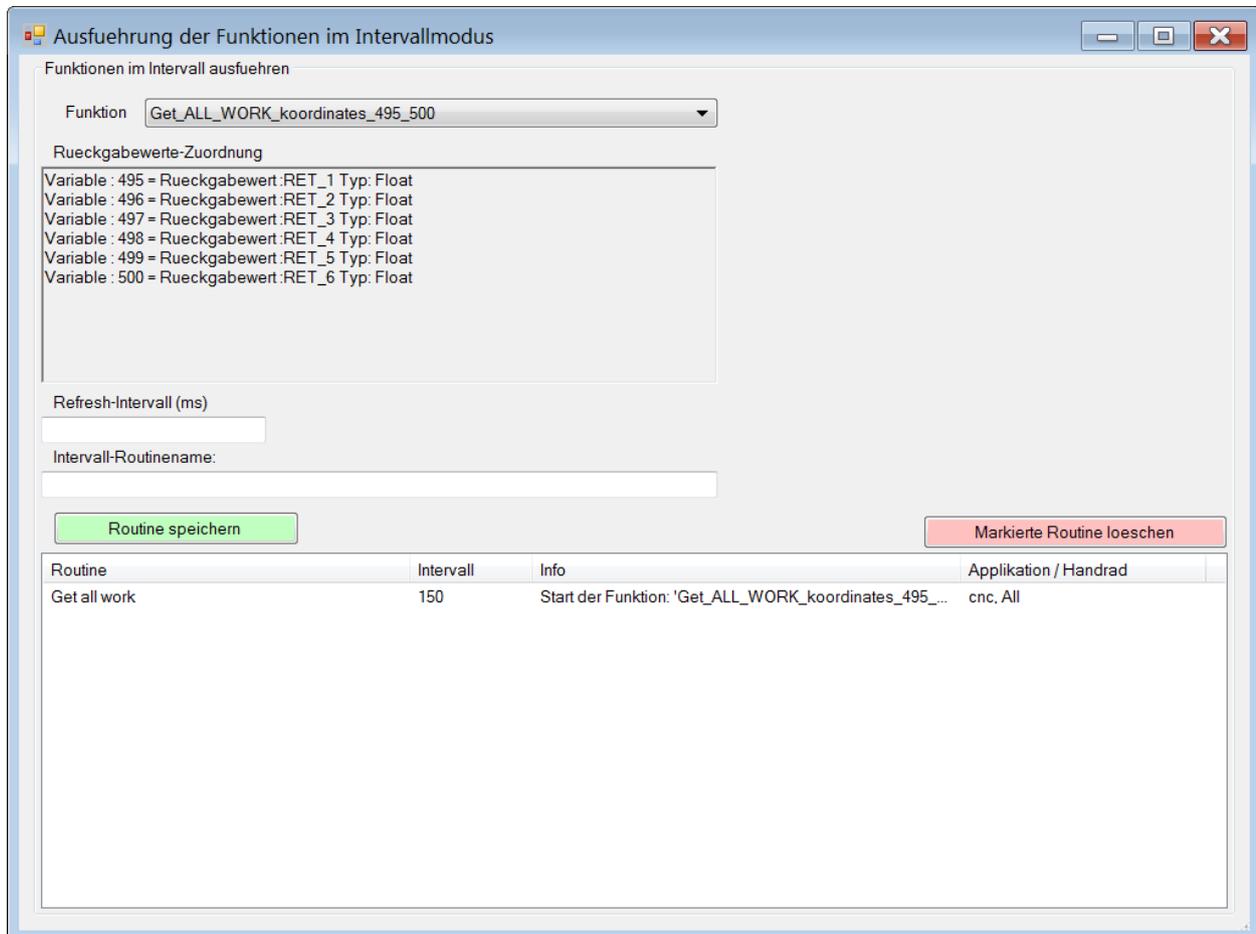


Beim Klick auf diese Schaltfläche wird ein neuer Dialog geöffnet in dem intervallbasierte Ausfuehrung der Funktionen eingestellt werden kann. Dies ist in vielen Fällen notwendig um z.b. die aktuellen Koordinaten auszulesen, eine bestimmte Prüfung vorzunehmen oder etwas zu überwachen. Wie aus dem Bild unten ersichtlich, kann ein Intervall in Millisekunden angegeben werden nach Ablauf des-

sen die zugeordnete Funktion startet.

In diesem Beispiel ist eine Funktion aktiv, die jede 150 ms die aktuellen Achsen-Koordinaten von EDING CNC ausliest damit diese dann auf dem Handrad und ggf. auf der externen Anzeige angezeigt werden können.

In dem Rückgabewerte-Feld sieht man in welche internen Variablen diese Werte geschrieben werden und vom welchen Typ die sind.



### Variablen-Elemente Zuordnung

Zuordnung der Elemente zu den Variablen

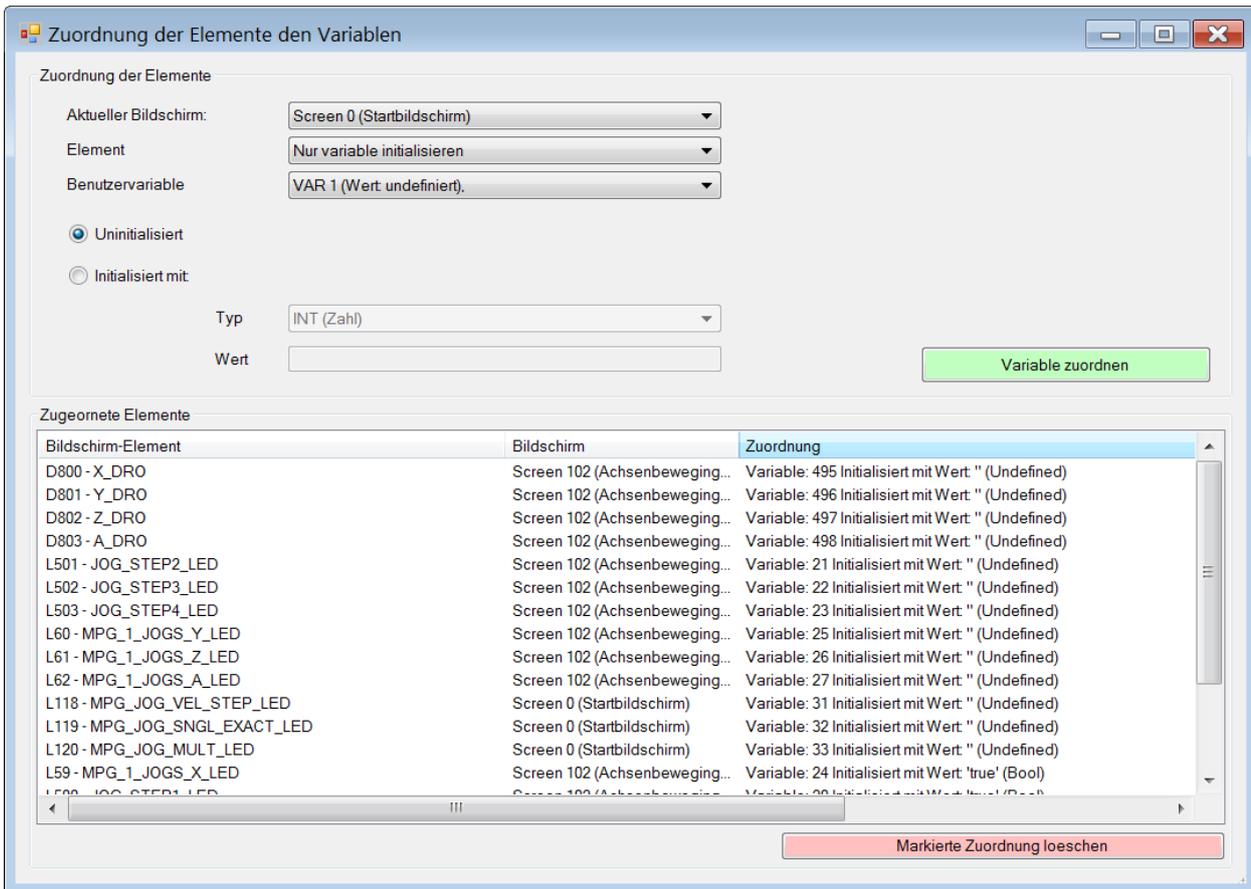
Hier koennen Bildschirm-Elemente (Leds, DROs, Progressbalken) zugeordnet werden. Ausserdem koennen fuer die Elemente oder Variablen Initialisierungswerte definiert werden.

Beim Klick auf diese Schaltfläche wird ein neuer Dialog geöffnet in dem die Zuordnung der Bildschirm-Elemente des LCD Touch Pendant Handrades zu den internen Variablen hergestellt werden kann. Dadurch wird sichergestellt dass eine Änderung des Wertes einer zugeordneten internen Variable (z.b. durch Intervallbasierte Funktionen) automatisch das Ändern des Zustandes des entsprechenden Bildschirm -Elementes erfolgt. So können DRO's , Led's und Progress Balken die Zustandsänderung der gesteuerten Maschine widerspiegeln.

In dem Beispiel unten sieht man z.b. dass die Koordinaten, die zuvor in der Intervallbasierten Funktion ausgelesen worden sind zu den DRO's D800-D803 auf dem „Screen 102“ zugeordnet sind.

Ausserdem können in dem Dialog auch Variablen-Initialisierungen vorgenommen und ggf. mit einem Initial-Wert belegt werden. Dies ist vor allem beim Start notwendig damit alle Bildschirm-Elemente sinnvolle Werte oder Zustände anzeigen. Ausserdem können so default-Werte für Funktionen vorbelegt werden.

Die Elemente und Bildschirm-Auswahl sind nur bei einem LCD Touch Pendant Handrad sichtbar, bei den anderen Handrad-Typen sind die ausgeblendet und somit ist nur Variablen-Initialisierung möglich.



**Not-Aus Befehle**

Not-Aus Befehl

Hier können, durch Komma getrennt mehrere Befehle definiert werden die von den Tasten des Handrades kommend, alle Bewegungsabläufe stoppen und die Ausführung der Not-Aus-Funktion initiieren. In dem Beispiel oben sind es die Befehle des LCD Touch Pendant Handrades die gesendet werden wenn auf dem Bildschirm „Not-Aus“ Tastenfeld betätigt wird.

**Achtung !** Verlassen Sie sich nicht darauf, eine Software-basierte Not-Aus Abschaltung ist unsicher und muss zwingend mit einer kombiniert werden die Stromleitung zuverlässig unterbricht!

**Not-Aus Funktion**

Not-Aus Funktion

Hier kann eine (Plugin-) Funktion ausgewählt werden die im Falle von Not-Aus Aktivierung ausgeführt wird.

**Achtung !** Verlassen Sie sich nicht darauf, eine Software-basierte Not-Aus Abschaltung ist unsicher und muss zwingend mit einer kombiniert werden die Stromleitung zuverlässig unterbricht!

**Initialisierungs-Tastenkombination**

Aktivierungstastenkombination   Aktiviert

Hier kann eine Tastenkombination definiert werden die gleich nach dem fokussieren auf die gesteuerte Applikation einmal ausgeführt wird. Das ist z.b. notwendig um in ein Modus zu gelangen in dem Steuerung über Tasten möglich ist (z.b. F5 bei Win PC-NC) kann aber auch für andere Initialisierungsfunktionen verwendet werden. Diese Einstellung ist für alle Handräder gültig, ist aber Applikation-gebunden.

**Vorschub (Feed)**

Hier können Werte eingestellt werden die Einfluss auf die Geschwindigkeit und das Verhalten beim Drehen des Handrades haben.

Vorschub (Feed)					
Vorschub-Umschaltung (Handrad-Drehimpulsstaerke) Werte bis zu dem Maximalwert des Drehencoders					
<input checked="" type="checkbox"/> Aktiviert	<input checked="" type="checkbox"/> Aktiviert	<input checked="" type="checkbox"/> Aktiviert	Wenn eine Stufe deaktiviert wird - wird die Maschine in dem Bereich nicht bewegt.		
Langsam	50	Mittel	90	Schnell	150
Vorschub-Geschwindigkeit					
Langsam	10	Mittel	50	Schnell	100
				Maximal	200

Zu beachten ist dass diese Werte unterschiedlich interpretiert werden – abhängig davon welcher Modus ausgewählt ist.

Im simulierten Betrieb wird z.b. Vorschub-Geschwindigkeit nicht ausgewertet da man ja nur über Tastenkombinationen (z.b. Umschalt oder STRG) die Geschwindigkeit beeinflussen kann. Die Werte für Vorschub-Umschaltung beziehen sich teilweise auf Geschwindigkeit des Drehens (Velocity Modus) oder Anzahl der Klicks (Trace Modus).

Beispiel oben zeigt die Werte die Experimentell ermittelt worden sind können bei jedem je nach Maschine oder Empfinden verändert werden.

**3.9.4 Weitere Einstellungen 2**

In diesem Reiter werden weitere Einstellungen für den 2/4 Takt Modus sowie Variablen und Funktionen Für Override – Funktionalität bereitgestellt.

Variable für 2/4 Takt Modus wird z.b. von einer Taste gesetzt, die Feed und Speed Variablen werden z.b. durch das Drehen des Encoders verändert und werden durch die unten zugeordneten Funktionen an das gesteuerte Programm übergeben.

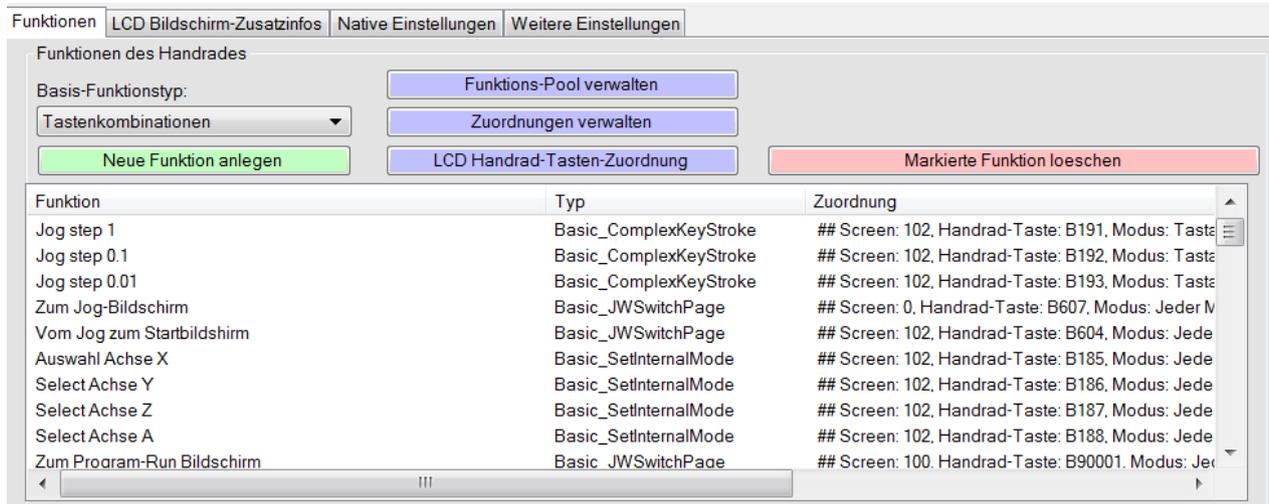
The screenshot shows a configuration window titled 'Konfiguration' with the following content:

- Header: Einstellungen gelten fuer Applikation: EDING CNC, Handrad: CNC Technics V8
- Handrad - Modus:  Nativ,  Tastatur-Simuliert,
- Tabs: Funktionen | Bildschirm-Infos | Native Einstellungen | Weitere Einstellungen | Weitere Einstellungen 2
- 4 Takt Modus-Aktivierung-Variable: VAR 60 (Wert false), Zugeordnet. Variable muss vom typ BOOL sein. Wenn die auf TRUE gesetzt ist - startet 4 Takt Betrieb.
- Feed Override Variable: VAR 72 (Wert 100), Zugeordnet. Variable muss vom typ INT oder FLOAT sein.
- Spindel Override Variable: VAR 73 (Wert 100), Zugeordnet. Variable muss vom typ INT oder FLOAT sein.
- Plugin-Feed Override-Funktion: (USBCNC\_PLG) PLUGINAPI\_SetFeedOverrideFactor
- Plugin-Spindel Override-Funktion: (USBCNC\_PLG) PLUGINAPI\_SetSpeedOverrideFactor

## 4. Funktionen

Die Erstellung der Funktionen und die Möglichkeit diese in Funktionspools zusammenzufassen ist eine der wichtigsten Produkteigenschaften von Jog Wheel Communicator und ermöglicht praktisch jeden Automatisierungsablauf zu erstellen und das ohne Programmierkenntnisse. Alle Funktionen werden mit Hilfe von „Drop-Down“ Boxen, Textfeldeingaben und Auswahllisten angelegt, Falsche Eingaben sind in den meisten Fällen ausgeschlossen.

Den Reiter für Funktionen und Funktionspools in der Konfiguration sehen Sie unten:

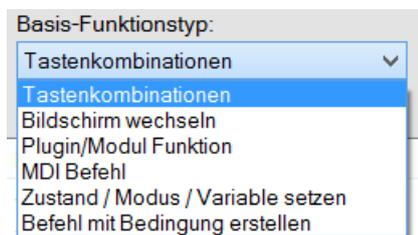


Es ist in mehrere funktionelle Gruppen aufgeteilt

- Erstellung einer neuen Funktion
- Verwaltung der Zuordnungen und Funktionspools
- Anzeigen, bearbeiten und Löschen der Funktionen

### 4.1 Funktion erstellen

Um eine Funktion zu erstellen wählt man zuerst den gewünschten Funktionstyp aus dem Drop-Down Box „Basis-Funktionstyp“, dabei gibt es mehrere Auswahlmöglichkeiten:



Aufgrund der anpassbaren Bedienoberfläche bietet LCD - touch pendant Handrad die meisten Einstellmöglichkeiten, im Folgenden werden Varianten beschrieben und Unterschiede zu dem Tinkerforge-basiertem Handrad aufgezeigt.

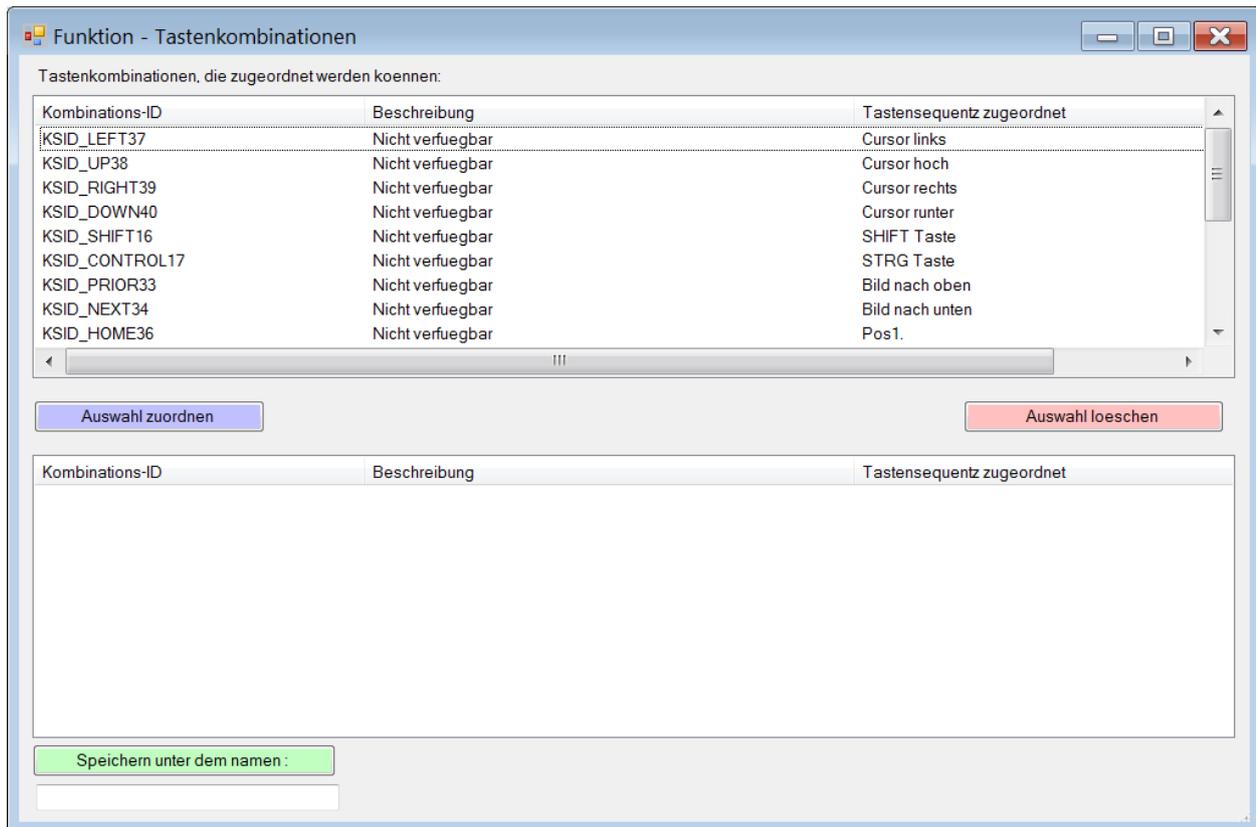
Nach der Auswahl klickt man dann auf die Schaltfläche „Neue Funktion anlegen“, daraufhin öffnet sich ein Dialogfenster für die entsprechende Funktion.

#### 4.1.1 Tastenkombinationen - Funktion

Diese Funktion bietet eine Möglichkeit beliebig komplexe Tastenkombinationen zu erstellen und unter einem Namen abzuspeichern. So können ganze Texte auf Knopfdruck ausgegeben werden. Man könnte z.B. Mak-

ros erstellen die in einen Befehl - Fenster einfügen, doch dafür existiert eine spezielle Funktion.

Das Dialog-Fenster sehen Sie unten:



die hier angezeigten, existierenden Tastenkombinationen wurden bereits wie im Abschnitt 3.8.1 angelegt. Nach der Auswahl der gewünschten Tastenkombination und beim anschliessenden Klick auf

**Auswahl zuordnen**

wird es zu der unteren Liste hinzugefügt.

**Info** : die Reihenfolge von oben nach unten ist die Ausgabereihenfolge !

**Info** : Sie können die ausgewählten Einträge beim Klick auf „Auswahl loeschen“ auch selektiv löschen.

Zur Speicherung der Funktion geben Sie den gewünschten Namen in Textfeld ein und bestätigen Sie abschliessend mit **Speichern unter dem namen :**

## 4.1.2 Bildschirm wechseln - Funktion

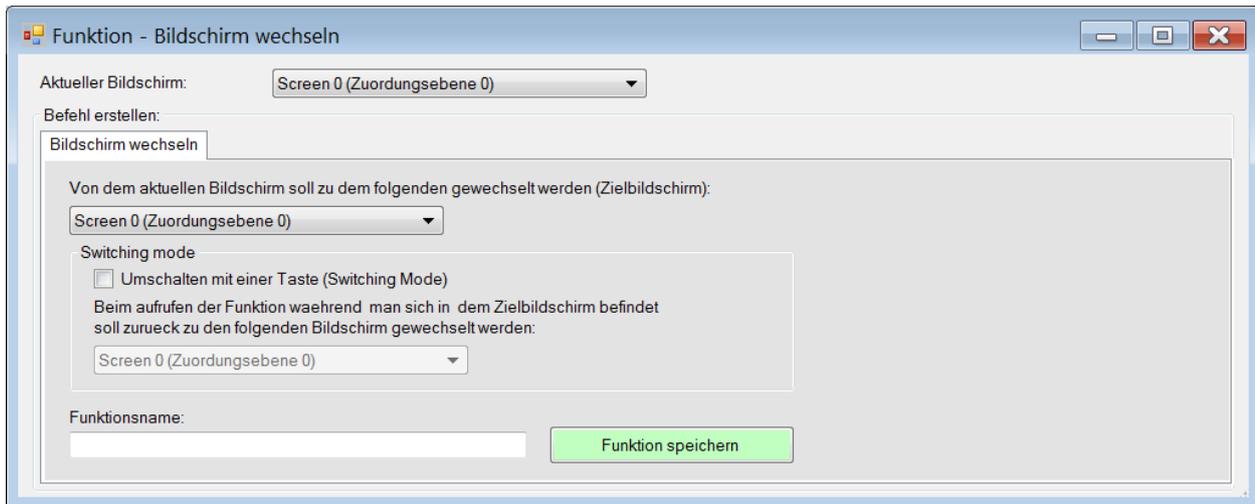
Diese Funktion gibt es vorwiegend für das Handrad „LCD touch pendant“. In dem Handrad können bis zu 20 Bildschirme definiert sein die mit Hilfe dieser Funktion aktiviert werden können.

Falls man ein anderes Handrad betreibt, funktionieren die Bildschirme als funktionsbegrenzende „Räume“ in denen z.B. Verhalten des Handrades oder Tasten Bildschirm-spezifisch definiert werden kann.

Beispiel:

Die Bewegung der Achsen ist nur in den „Jog-Move“ und „2/4 Takt Modus“ Bildschirmen möglich.

Das Dialog-Fenster sehen Sie unten:



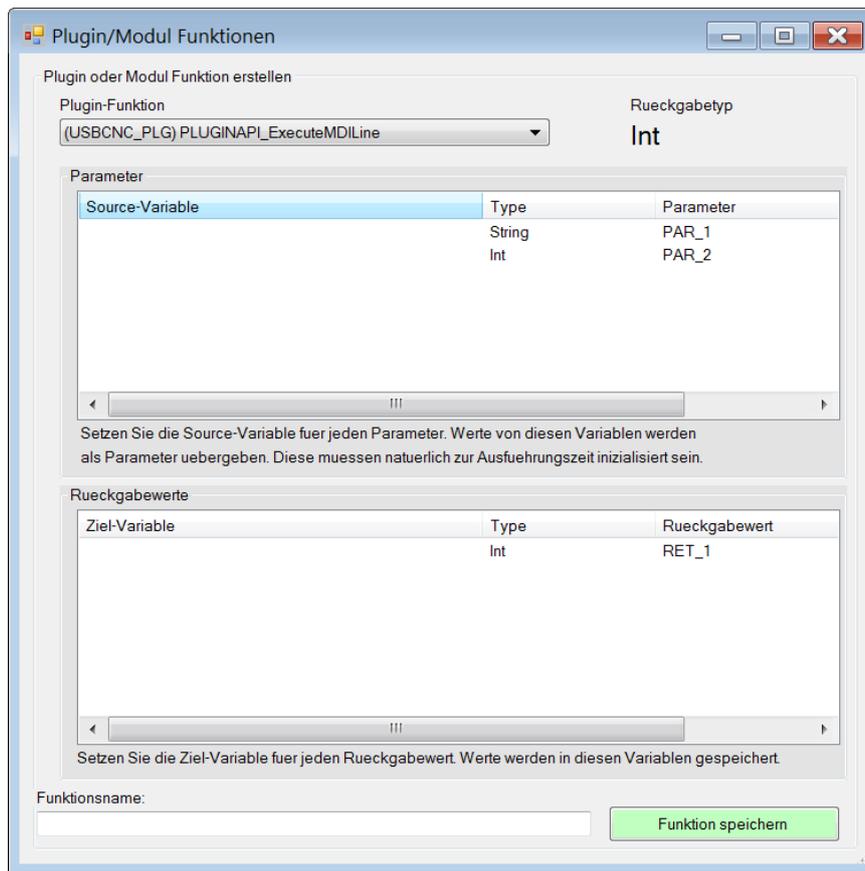
Dabei gibt es zwei Möglichkeiten. Normalerweise will man von einem bestimmten Bildschirm zu einem anderen wechseln (z.B. per Knopfdruck). Dafür wählt man den Start- und den Zielbildschirm aus und speichert die Funktion ab. Wenn man sich allerdings schon in dem Zielbildschirm befindet passiert bei der Ausführung der Funktion nichts.

Um mit derselben Funktion zu einem anderen Bildschirm zu wechseln wenn man sich bereits in dem Zielbildschirm befindet wurde „Switching Mode“ eingeführt um z.B. mit gleicher Taste zum Ausgangsbildschirm zurückzukehren. Das wird z.B. für „Emergency“ - Bildschirm verwendet.

### 4.1.3 Plugin / Modul - Funktion

Diese Funktion ist die Schnittstelle des JWC zu der Außenwelt. Damit ist es möglich die Funktionen der kompatiblen Module und Plugins zu nutzen. Da die Schnittstelle genormt und für alle Plugins/Module gleich bleibt, können Plugins / Module beliebig verändert oder sogar neu geschrieben werden ohne dass JWC selber oder die Konfiguration des Handrades verändert werden muss.

Das Dialog-Fenster sehen Sie unten:

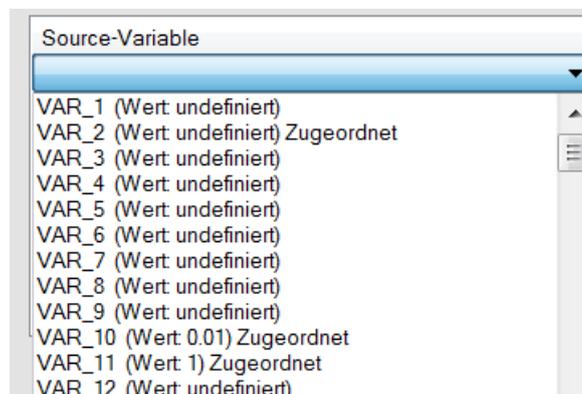


Jede externe Funktion kann Parameter, Rückgabewerte oder keins von den beiden haben. Oben, in Drop-Down Box „Plugin-Funktion“ sind alle verfügbaren Funktionen aufgelistet, vorne in Klammern steht der Name des Moduls / Plugins gefolgt von dem eigentlichen Funktionsnamen. Nach der Auswahl einer Funktion sieht man gleich gegenüber den Rückgabebetyp, falls mehrere Rückgabewerte existieren steht hier „Complex“. Des Weiteren gibt es:

- Int** - entspricht einer Ganzzahl in C# (int)
- Float** - entspricht einer Gleitkommazahl in C# (double)
- Bool** - entspricht einem boolischen Wert (true, false) in C# (bool)
- String** - entspricht einer Zeichenkette in C#

Unten stehen dann die Parameter und Rückgabewerte einzeln aufgelistet. In diesen Listen muss nun die Versorgung mit den Parameter-Werten sichergestellt bzw. Speicherort für die Rückgabewerte festgelegt werden.

Das geschieht beim Klick auf den (noch) leeren Platz in den Spalten „Source-Variable“ bzw. „Ziel-Variable“ wobei sich eine Drop-Down-Liste öffnet in der eine von z.Z. 500 internen Variablen ausgewählt werden kann.



Man soll natürlich sicherstellen dass diese Variablen vor der Verwendung gültige Werte bekommen (dem geforderten Typ entsprechend) bzw. bereits verwendete nicht überschrieben werden. Dies kann auf unter-

schiedliche Weise geschehen und wird später beschrieben.

Die bereits in anderen Funktionen zugeordnete Variablen werden in Drop-Down-Liste als „Zugeordnet“ markiert und deren aktueller Wert mit angezeigt. Dies soll helfen doppelte Belegung zu vermeiden.

Wenn alle Felder zugeordnet sind kann die Funktion unter einem, möglichst der Funktionalität entsprechenden Namen abgespeichert.

## 4.1.4 MDI Befehl - Funktion

Oft werden Abfolgen der Befehle benötigt die in einem gesteuerten Programm in speziellen Eingabefenstern eingegeben werden müssen. Bei CNC - Anwendungen wird oft von MDI Eingabefenster gesprochen, meistens kann es mit einer Tastenkombination geöffnet werden.

Des Weiteren bieten manche Programme durch API die Möglichkeit solche Befehlsfolgen auszuführen.

Für beide dieser Möglichkeiten können in JWC Funktionen erstellt werden.

Das entsprechende Dialog-Fenster sehen Sie unten:

Wie man sieht, besteht die Möglichkeit der Auswahl zwischen Tastenkürzel-basierter oder Plugin-basierter Vorgehensweise. Eine API - Variante ist vorzuziehen (falls Plugin-Funktion verfügbar) weil die schneller und zuverlässiger ist (z.B. hängt nicht vom Fenster-Fokus ab).

Bei der API-Variante muss die Plugin - Funktion bestimmten Regeln entsprechen was Parameter und Rückgabewerte angeht und zwar:

- die Plugin-Funktion muss einen Rückgabewert vom Typ INT besitzen
- die Plugin-Funktion muss 2 Parameter besitzen : der erste muss vom Typ STRING sein, der zweite vom Typ INT

Bei Tastenkürzel-Variante muss man die Tastenkombination auswählen die das MDI Fenster aufmacht.

Bei beiden Varianten kann man die auszuführenden Befehle jetzt in Textfenster eingeben. Dabei muss man entscheiden wie dieser ausgeführt werden soll: zeilenweise oder komplett, als Block.

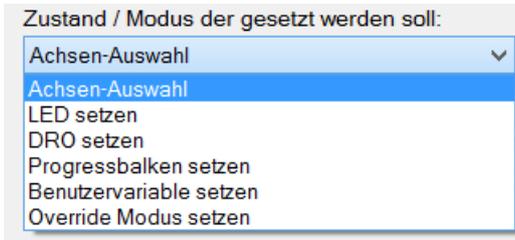
Bei zeilenweiser Ausführung (Das Häkchen bei Optionen „Befehle zeilenweise abarbeiten“ ist gesetzt) wird jede Zeile als ein Befehl ausgeführt, dabei kann man maximale Wartezeit nach jedem Befehl einstellen - es kann ja sein dass ein Befehl länger braucht weil z.B. Achsen bewegt werden oder Werkzeugwechsel stattfindet. Wenn zeilenweise Abarbeitung gewünscht ist muss man in dem Textfeld die Befehlszeilen logischerweise mit „Enter“ abschliessen damit diese erkannt werden können.

Bei der Ausführung als Block werden alle Zeilen im Textfeld zu einem Befehlsblock zusammengefügt und auf einmal zur Ausführung übergeben. Es wird jedoch empfohlen komplexe Abläufe aufzusplittern schon aus

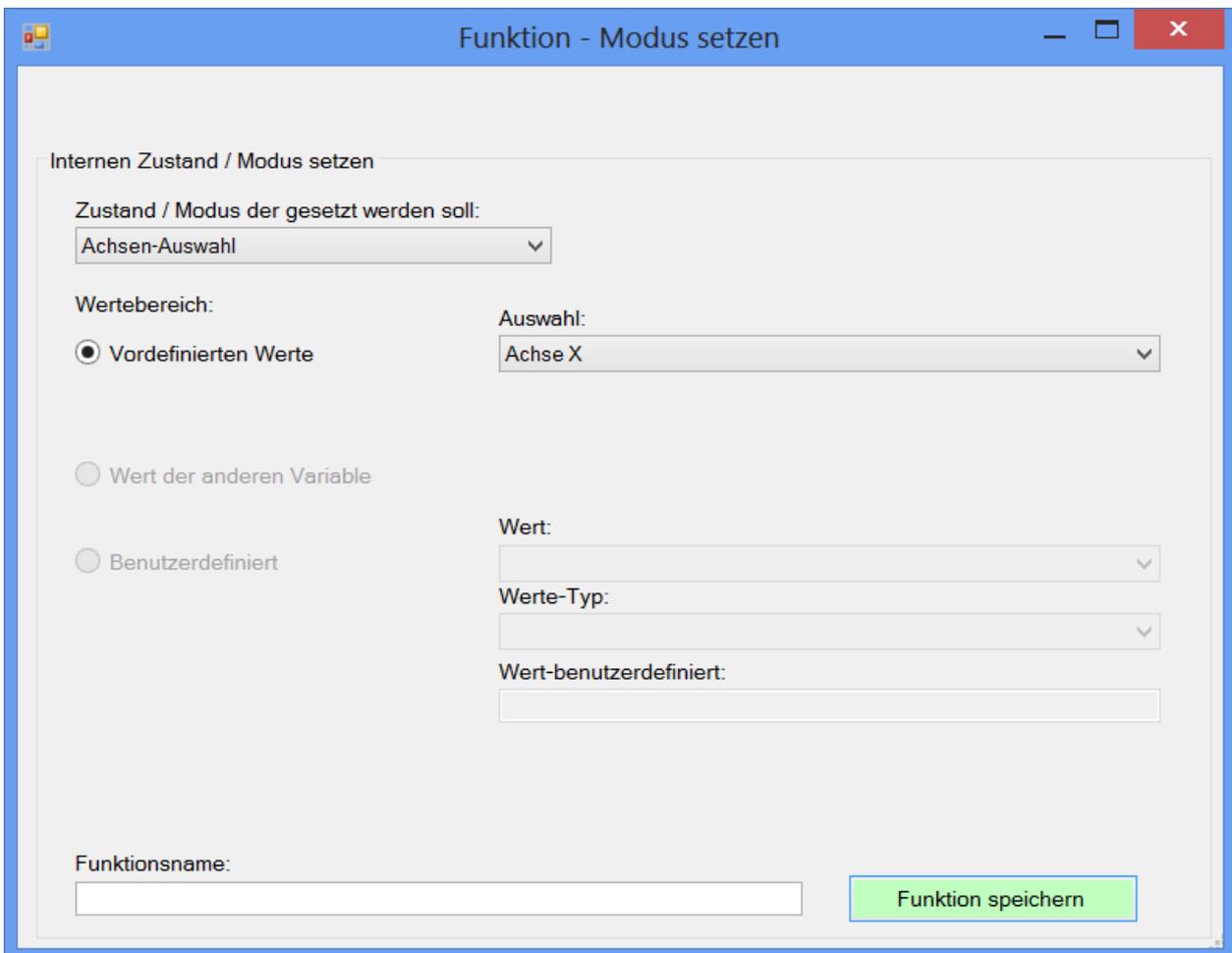
dem Grund dass z.b. die Maschinensteuerung Befehle ignorieren kann während die Achsen verfahren werden.

### 4.1.5 Zustand / Modus / Variable setzen - Funktion

Mit Hilfe dieser Funktion können Werte für unterschiedliche Elemente des Handrads (LCD - touch pendant Handrad) und Benutzer-Variablen gesetzt werden. So erstellte Funktionen können dann zusammengefasst und den Handrad-Tasten zugeordnet werden. Angezeigt werden immer nur die aktuell möglichen Einstellungen, wobei einige (z.b. Progress Balken setzen) nur Sinn in LCD - touch pendant Handrad haben:



Die Funktion zur Auswahl der Achsen kann wie folgt erstellt werden:



Eine Funktion zum Ein- / Ausschalten einer LED auf LCD Touch Pendant Screen kann erstellt werden wobei es die Möglichkeit gibt den Wert aus einer zuvor gesetzten Variable zu verwenden:

**Funktion - Modus setzen**

Aktueller Bildschirm: Screen 100 (Tools-Bildschirm)

Internen Zustand / Modus setzen

Zustand / Modus der gesetzt werden soll:  
LED setzen

Wertebereich:

Vordefinierten Werte

Wert der anderen Variable

Benutzerdefiniert

Auswahl:

LED Auswahl:  
L13 - FLOOD\_LED

Wert:  
Wert der Variable  
LED an  
LED aus  
Wert der Variable  
VAR 1 (Wert undefiniert)

Funktionsname:

Funktion speichern

Eine Funktion zum Setzen eines DRO - Wertes auf LCD Touch Pendant Screen kann ebenso erstellt werden. Dabei hat man die Auswahl

- ob das ein definierter Wert sein soll oder
- ob ein Wert addiert werden soll oder
- ob ein Wert subtrahiert werden soll oder
- ob ein Wert aus einer zuvor zugewiesenen Variablen verwendet werden soll.

**Funktion - Modus setzen**

Aktueller Bildschirm: Screen 102 (Achsenbewegungs- Bildschirm)

Internen Zustand / Modus setzen

Zustand / Modus der gesetzt werden soll:  
DRO setzen

Wertebereich:

Vordefinierten Werte

Wert der anderen Variable

Benutzerdefiniert

Auswahl:  
GetHeartBeat (Ret = VAR\_2)

DRO Auswahl:  
D800 - X\_DRO

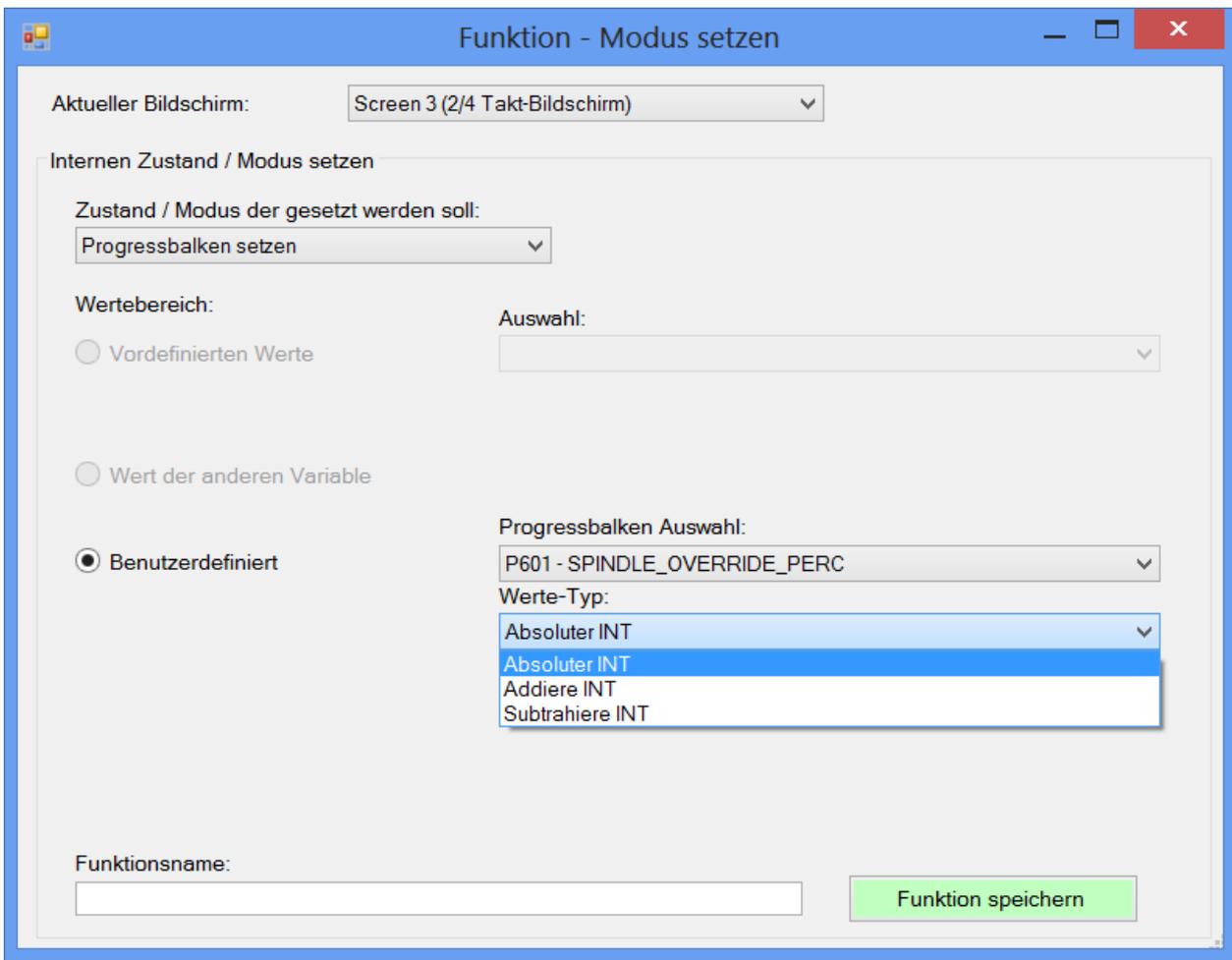
Werte-Typ:  
Wert der Variable  
Absoluter FLOAT  
Addiere FLOAT  
Subtrahiere FLOAT  
Wert der Variable

Funktionsname:

Funktion speichern

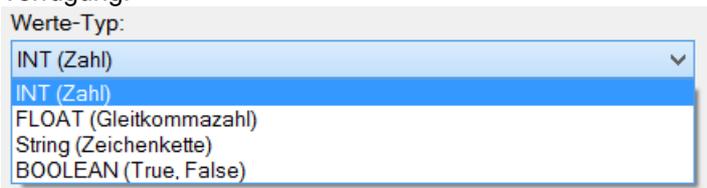
Ebenso ist es möglich eine Funktion zu erstellen die Progress Balken Werte auf LCD Touch Pendant Screen setzt. Hier hat man die Auswahl:

- ob das ein definierter Wert sein soll oder
- ob ein Wert addiert werden soll oder
- ob ein Wert subtrahiert werden soll.

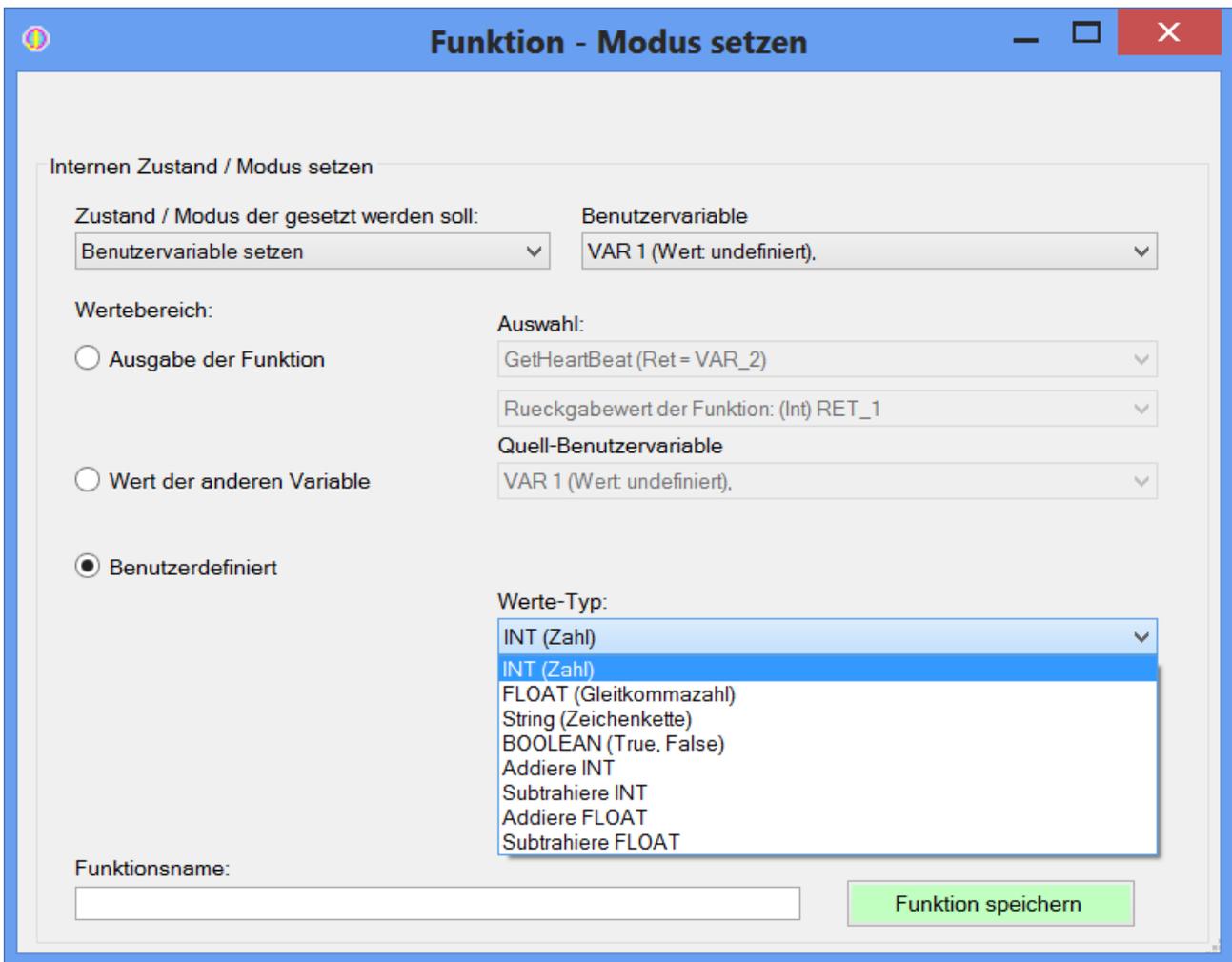


Schliesslich kann man eine Funktion erstellen die Benutzervariablen Werte setzt. Hier hat man viele Möglichkeiten:

- Variable kann auf den Rückgabewert einer Plugin-Funktion gesetzt werden. Dabei kann, bei mehreren Rückgabewerten einer Plugin-Funktion, ein bestimmter Rückgabewert ausgewählt werden.
- Ebenso kann einer Variablen der Wert einer anderen Variablen zugewiesen werden.
- Ein beliebiger Zahlenwert kann addiert oder subtrahiert werden
- Auch Zuweisung eines benutzerdefinierten Wertes ist möglich, hier stehen verschiedene Datentypen zur Verfügung:



Beim Speichern werden alle eingegebenen Werte auf das Entsprechen dem ausgewählten Datentyp geprüft.

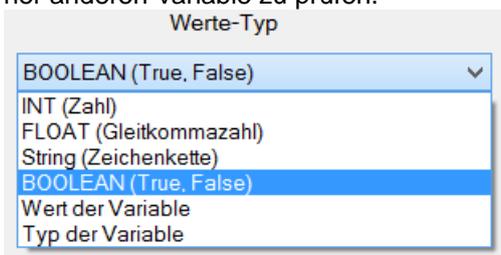


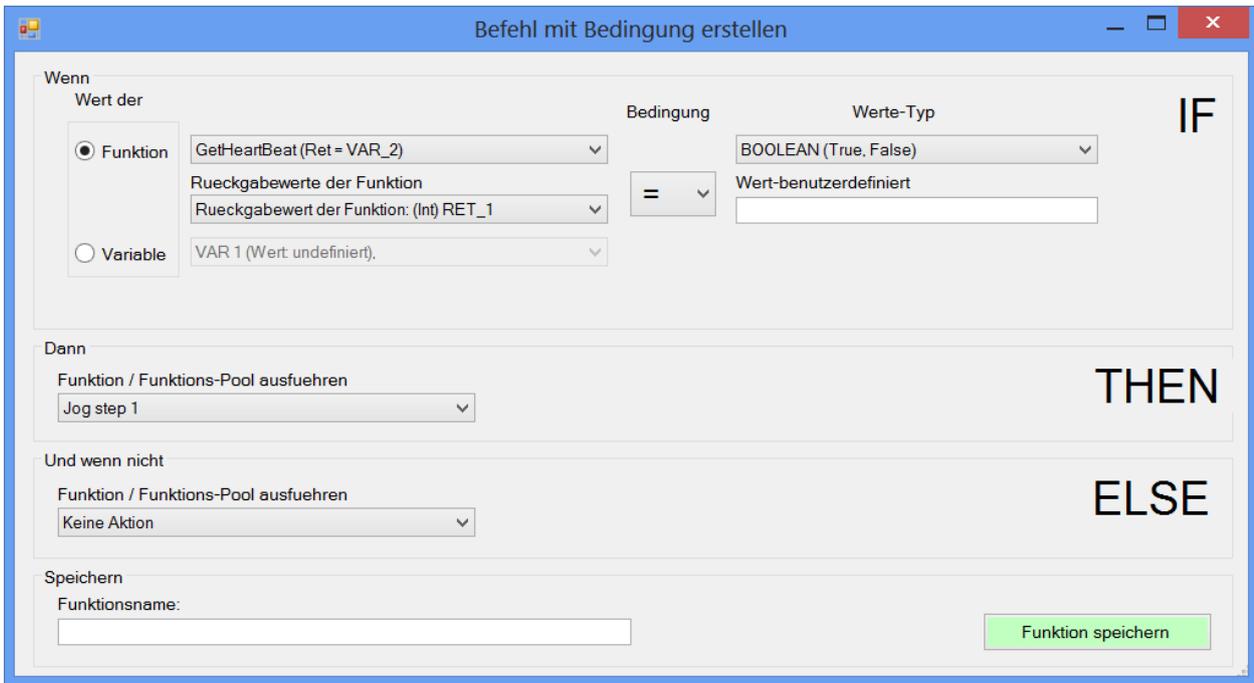
### 4.1.6 Befehl mit Bedingung erstellen - Funktion

Eis ist immer wieder notwendig Aktionen in Abhängigkeit eines vorher berechneten Zustandes durchzuführen. Diese bedingungsabhängige Ausführung ist z.B. in vielen Programmiersprachen als Konstrukt „IF-THEN-ELSE“ bekannt und ist in JWC ebenfalls vorhanden.

Dialog unten ermöglicht es eine Funktion als eine Bedingung zu definieren, die Aktion festzulegen wenn diese wahr ist und optional eine Aktion durchzuführen wenn die Bedingung falsch war.

Als Operanden können Plugin-Funktionen (Rückgabewerte deren) und Variablen verwendet werden, dabei besteht die Möglichkeit nicht nur die festen Werte zu vergleichen sondern auch den Datentyp oder Wert einer anderen Variable zu prüfen.





### 4.1.7 Funktions-Pool erstellen

Um einen Funktions-Pool zu erstellen klickt man in dem Reiter „Funktionen“ (im Konfigurations-Dialog) auf die Schaltfläche



daraufhin öffnet sich ein Dialog wo man entweder die Zuordnung der Funktionen einer Handrad-Taste oder die Speicherung eines Funktions-Pools vornehmen kann. Dabei :

- man selektiert die Funktion

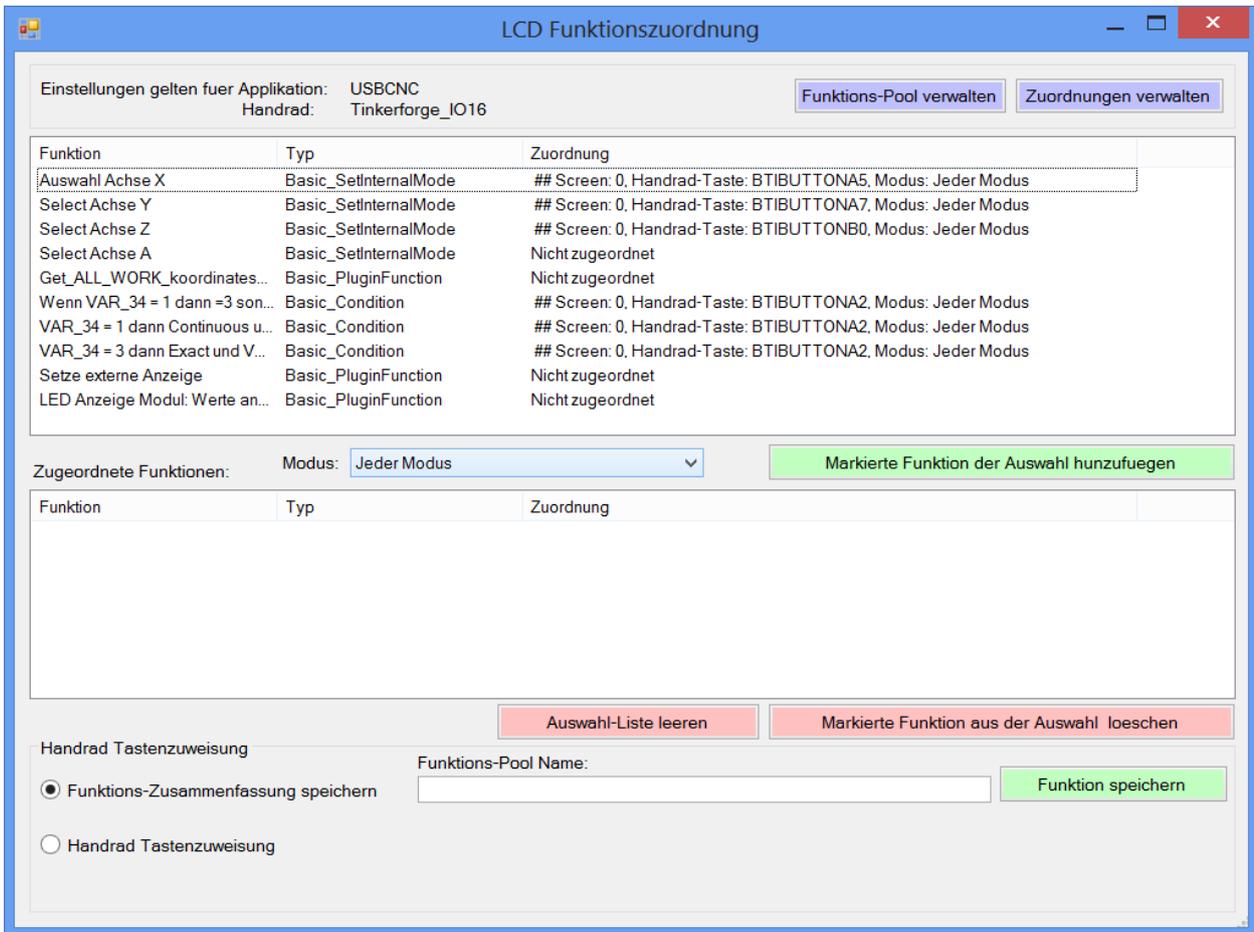
- man klickt auf

- man wiederholt den Vorgang bis alle gewünschten Funktionen in der Reihenfolge der Ausführung angeordnet sind

- man selektiert den Modus (Native oder Tastatur-Simuliert)

- man wählt ,gibt den Namen für Funktions-Pool ein und klickt auf

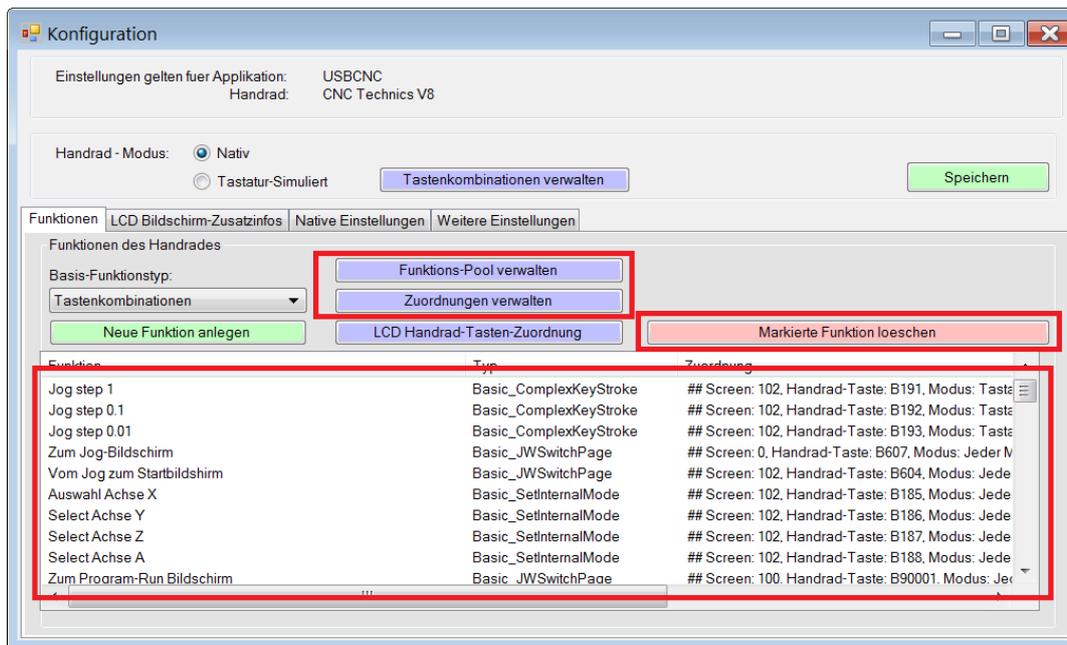




## 4.2 Verwaltung der Funktionen

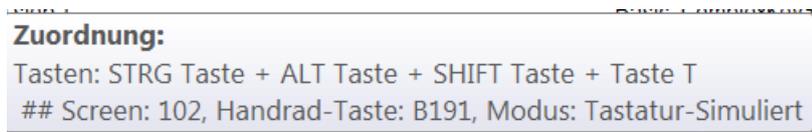
Die erstellten Funktionen sind ein Grundstein mit Hilfe dessen sich eine beliebig komplexe Funktionalität realisieren lässt. Das erfordert allerdings auch eine flexible und übersichtliche Verwaltung um Funktionen zu löschen, verschieben und Zuordnungen zu bearbeiten.

Die meisten Verwaltungsmöglichkeiten sind aus dem Reiter „Funktionen“ zugänglich. Die im Bild rot markierten Bereiche werden im folgendem erklärt:



### 4.2.1 Funktion löschen

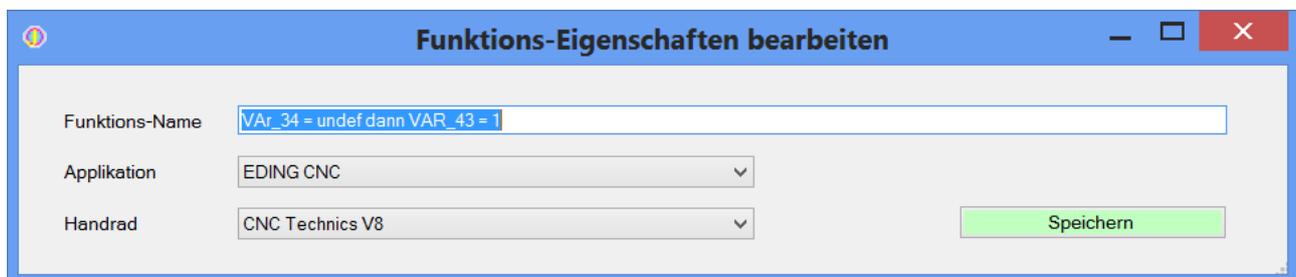
Um eine Funktion zu löschen markieren Sie die Funktion und klicken anschliessend auf „Markierte Funktion löschen“. Um erst zu sehen was die Funktion macht und wo die zugeordnet ist platzieren Sie den Mauszeiger über die gewünschte Funktion , dann erscheint eine kurze Beschreibung der eintsprechenden Funktionalität und der Zuordnung der Funktion:



Beim Löschen der Funktion wird auch eine Sicherheitsabfrage initiiert die unter anderem darauf hinweist dass beim Löschen einer Funktion auch alle Zuordnungen gelöscht werden. Das ist wichtig zu wissen denn sonst würden alle anderen Funktionen ungültig wenn Sie die gelöschte Funktion verwendet haben.

### 4.2.2 Eigenschaften der Funktion bearbeiten

Beizeiten ist es notwendig eine Funktion umzubenennen, Zuordnung zu der Applikation zu ändern oder Geltungsbereich (Handrad) anzupassen denn nicht alle Funktionen machen überall Sinn (z.b. Spezifische Funktionen die nur für EDING CNC verwendet werden können). Das kann man machen indem man mit der Maus ein Linksklick auf der Funktion ausführt. Danach erscheint ein Fenster wo man die Änderungen vornehmen kann:



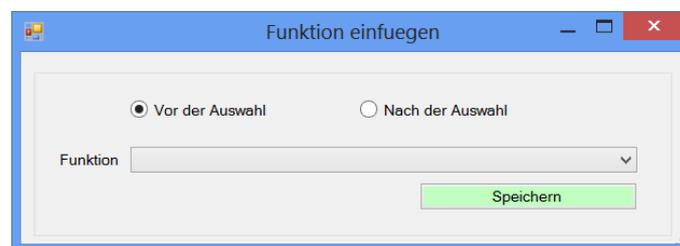
Dabei kann man auch „Alle Applikation“ und „Alle Handräder“ auswählen damit eine Funktion überall verfügbar wird und zugeordnet werden kann.

## 4.2.3 Zuordnungen verwalten

Hier kann man die bestehenden Zuordnungen zu den Tasten des Handrades anschauen und bearbeiten.

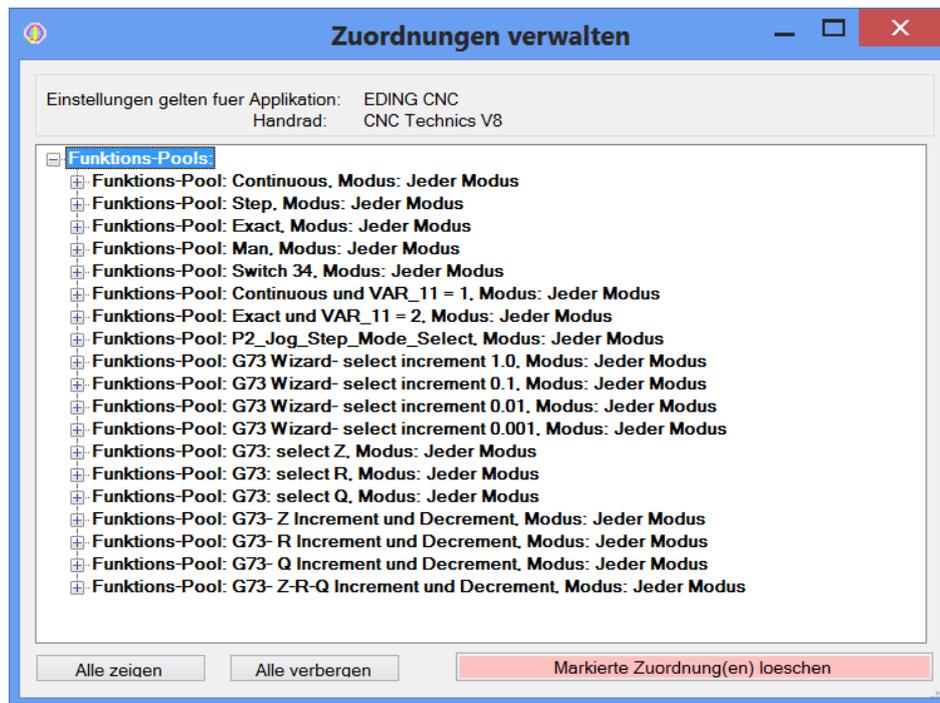


Hier am Beispiel des LCD Touch Pendant Handrades sieht man welche Funktionen welcher Taste auf welchem Bildschirm zugeordnet sind. Diese Zuordnungen kann man löschen (Klick auf „Markierte Zuordnung(en) loeschen“), verschieben oder eine neue Funktion hinzufügen (dies ist durch ein Kontextmenu möglich - Maus Linksklick -):



## 4.2.4 Funktions-Pool verwalten

Wie schon erwähnt, können Funktionen in Gruppen zusammengefasst werden zu sog. Funktions-Pools.



Genauso wie bei den Zuordnungen kann man Funktionen und Pools löschen (Klick auf „Markierte Zuordnung(en) loeschen“), verschieben oder eine neue Funktion hinzufügen (dies ist durch ein Kontextmenu möglich - Maus Linksklick -). Ebenfalls aus dem Kontextmenu ist es möglich Modus einer Zuordnung / eines Funktionspools zu ändern.

## 5. Module

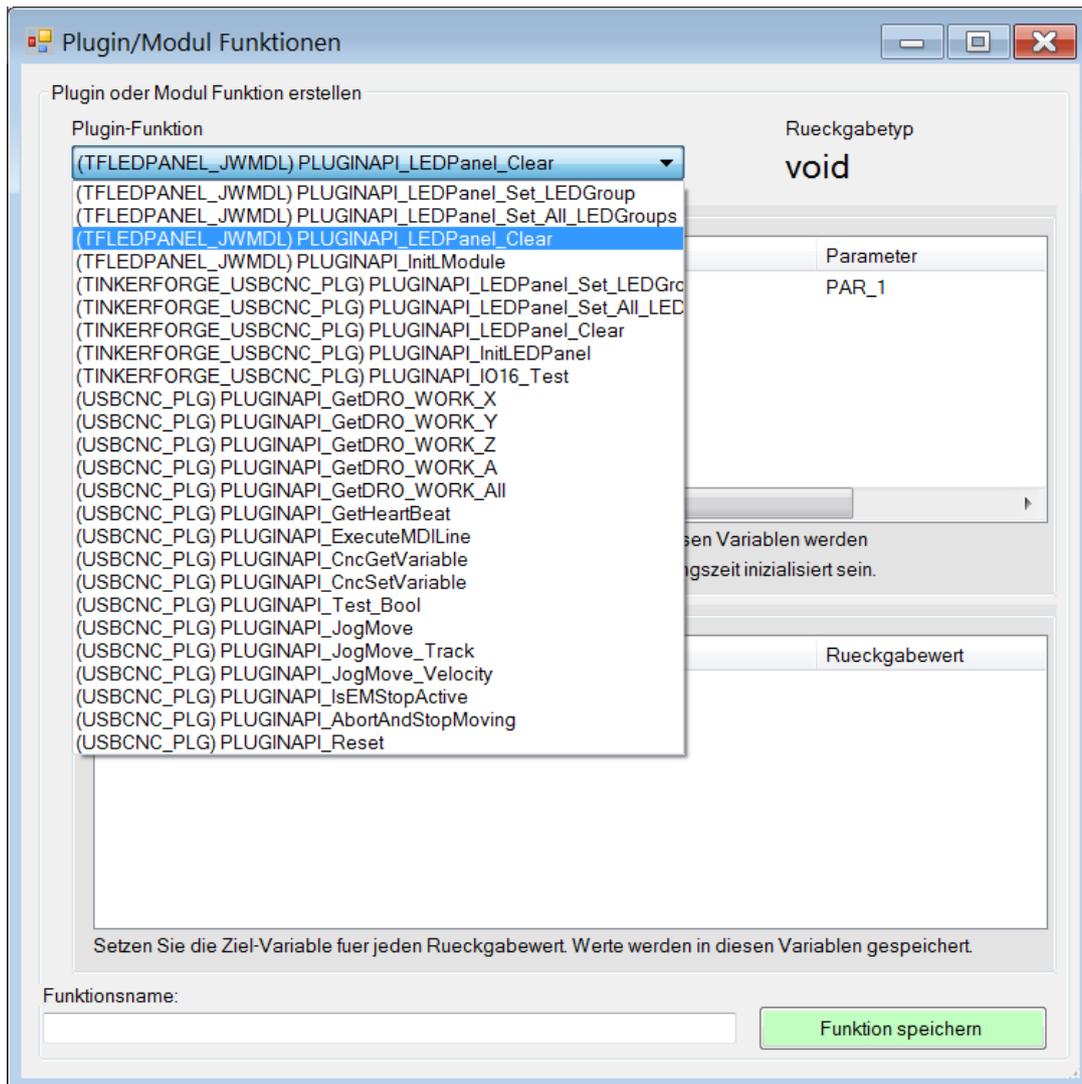
### 5.1 Installation, Aktivierung und Konfiguration eines Moduls

Module werden als vorkompilierte Bibliotheken (Dll's) geliefert. Um ein Modul zu installieren:

1. Installieren Sie die für den Betrieb des Moduls notwendige Komponenten (Beachten Sie die mitgelieferten Hinweise zu dem Modul)
2. Kopieren Sie das Modul ins Verzeichnis „**JogWheelCommunicator / Plugins**“
3. Öffnen Sie die Module-Konfigurationsdatei „**LCDJogWheelExtensionsConfiguration.xml**“ und kopieren Sie die mit dem Modul mitgelieferte Konfigurations- XML Sektion als Unterknoten zu dem Gruppenknoten z.B. „<TinkerforgeExtensions>“ (Beachten Sie die mitgelieferten Hinweise zu dem Modul)
4. Starten Sie JWC
5. Wählen Sie auf dem Startbildschirm  „Module konfigurieren“
6. In dem Dialog der sich daraufhin öffnet kann man nun die Installierten Module sehen. Vergewissern Sie sich ob das neue Modul in der Liste aufgeführt ist. Neue Module sind deaktiviert und erscheinen auf rotem Hintergrund.
7. Klicken Sie mit der rechten Maustaste auf dem Modul. Es erscheint ein Kontextmenu mit den Einträgen „Aktivieren“, „Deaktivieren“ und „Bearbeiten der Eigenschaften“
8. Wählen Sie „Bearbeiten der Eigenschaften“ und nehmen Sie die notwendigen Einstellungen vor (Beachten Sie die mitgelieferten Hinweise zu dem Modul). Speichern Sie die Einstellungen.
9. Klicken Sie erneut mit der rechten Maustaste auf dem Modul und wählen Sie „Aktivieren“. Hintergrundfarbe des Moduls soll sich auf „grün“ ändern, das Modul ist somit aktiviert.

### 5.2 Verwendung

Nachdem ein Modul aktiviert wurde stehen die Modul-Funktionen zur Verwendung bereit und sollten zuerst in JWC eingebunden werden. Das geschieht indem man die Parameter / Rückgabewerte zu den internen Variablen zuordnet (siehe Kapitel 4.1.3 „Plugin / Modul - Funktion“)



## 6. Anwendungsbeispiele

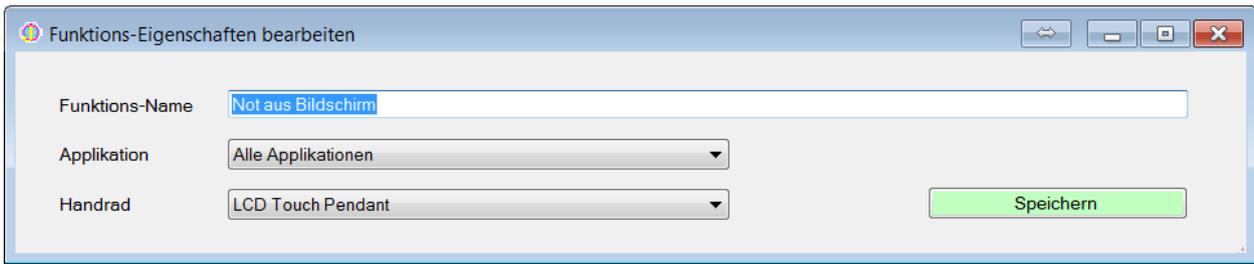
JWC ist eine sehr flexibel angelegte Anwendung mit deren Hilfe viele Aufgaben der Automatisierung und CNC Steuerung erleichtert oder erst möglich gemacht werden. Um Anregungen für eigene Entwicklungen zu geben und grundlegende Abläufe zu verdeutlichen werden im Folgenden ein Paar Anwendungsbeispiele in Detail erklärt.

### 6.1 Verwenden einer Funktion in mehreren Profilen

Das Nutzen der Funktionen eines anderen Profils hat den Vorteil dass diese Funktionen bereits in voller Komplexität erstellt und getestet worden sind – und somit die Einrichtung sehr viel schneller geht und die doppelte Erstellung von gleichen Funktionen vermieden wird. Es hat allerdings auch den Nachteil dass wenn diese Funktion versehentlich oder absichtlich gelöscht wird, sind alle Profile die diese Funktion verwenden betroffen.

Doch zuerst muss man eine Funktion global verfügbar machen. Normalerweise gelten die neu erstellten Funktionen für die aktuell ausgewählte Applikation und das Handrad und sind für die anderen Profile nicht

sichtbar. Das kann man ändern indem man im Reiter „Funktionen“ auf der gewünschten Funktion mit rechter Maustaste klickt und „Bearbeiten der Eigenschaften“ auswählt. Dann erscheint ein Dialog:

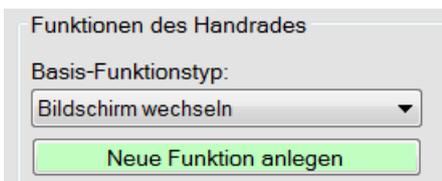


in dem man dann den Gültigkeitsbereich der Funktion einstellen kann. Hier sollte man „Alle Applikationen“ und / oder „Alle Handraeder“ damit man die in dem entsprechenden Profil sehen und zuordnen kann.

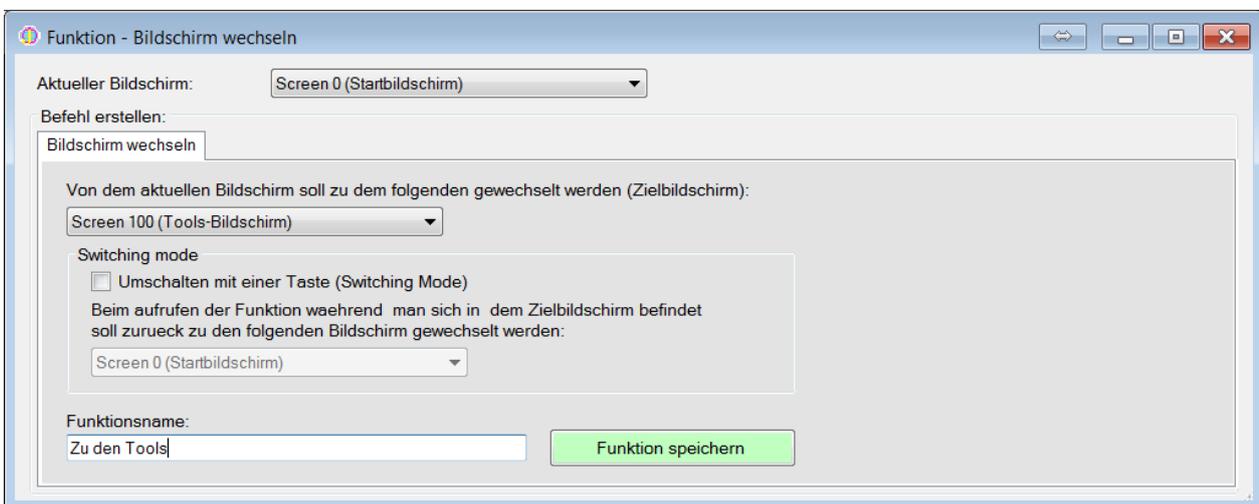
## 6.2 Umschalten von Bildschirmen auf dem Handrad

Um beim Berühren einer Schaltfläche auf dem Handrad-Bildschirm zu einem anderen Bildschirm zu gelangen geht man folgendermaßen vor:

- in der Konfiguration des Handrades gehe man auf den Reiter „Funktionen“ und wähle unter „Basis-Funktionstyp“ „Bildschirm wechseln“, dann klicke man auf „Neue Funktion anlegen“:



Dabei erscheint ein Dialogfenster in dem Start- und Zielbildschirme definiert werden können. In dem Beispiel wollen wir von dem „Startbildschirm“ zu dem „Tools-Bildschirm“ gelangen:



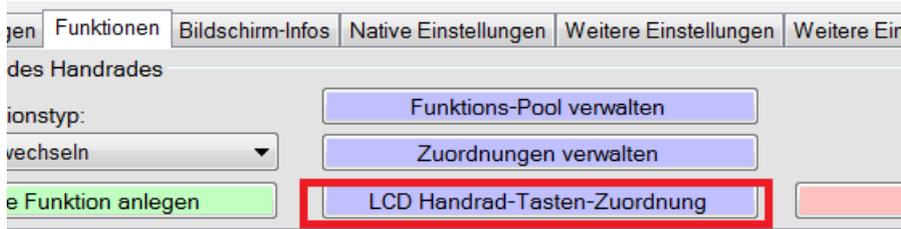
Beim Klick auf „Funktion speichern“ wird die neue Funktion angelegt. Alternativ hätte man eine bereits existierende Funktion eines anderen Profils freigeben und verwenden können.

Die Funktion soll nun in der Funktionsliste sichtbar sein:



Man sieht hier dass die Funktion noch nicht in dem Profil zugeordnet wurde.

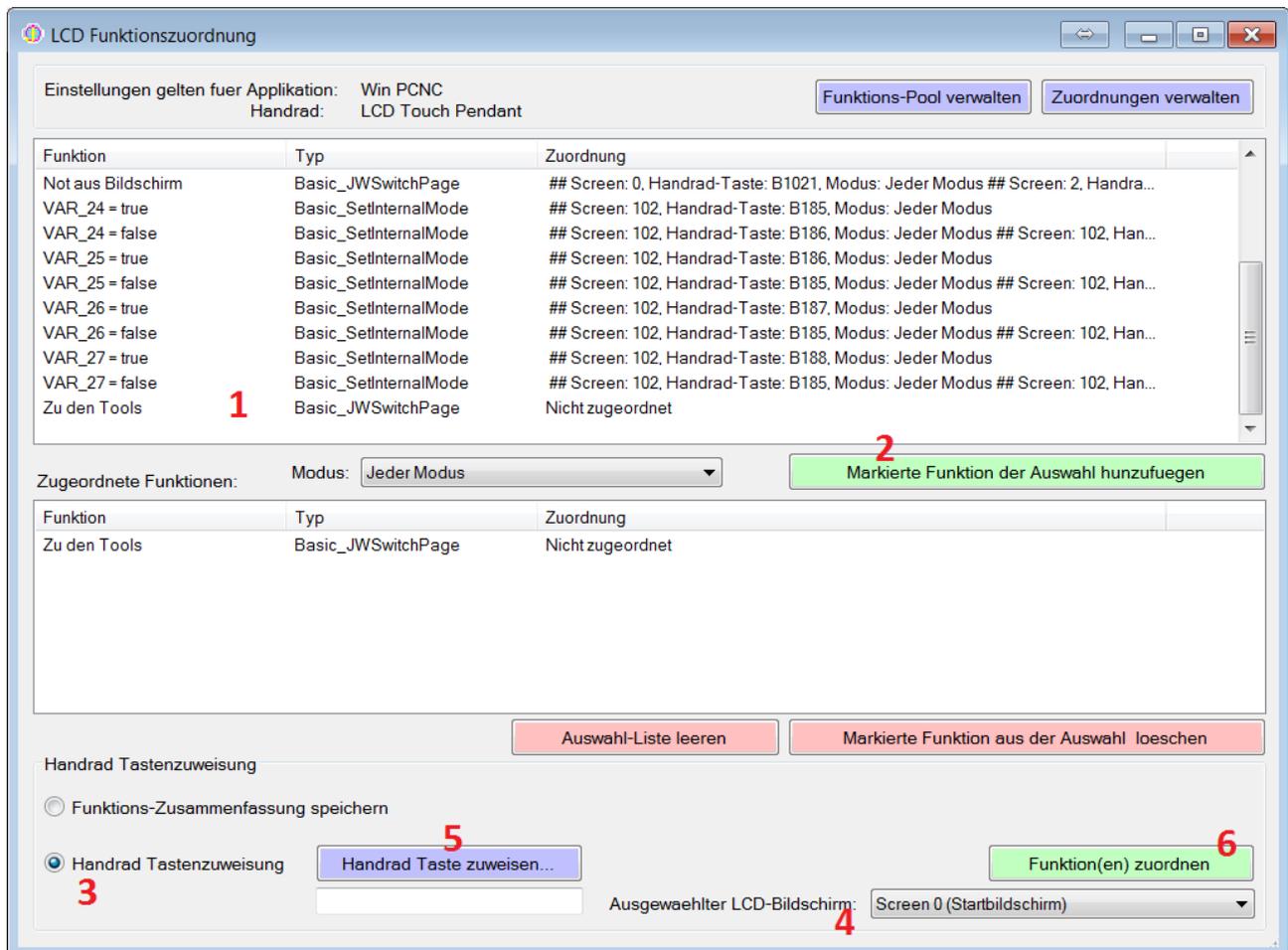
Nun müssen diese Funktion einer Taste des Startbildschirms zuordnen um beim Klick darauf zu dem eingestellten Bildschirm zu gelangen. Dazu klicke man nun auf die Taste „LCD Handrad Tasten-Zuordnung“:



Daraufhin wird ein Dialog geöffnet in dem man nun folgende Aktionen durchführen soll (das Handrad muss währenddessen bereits richtig konfiguriert und angeschlossen worden sein):

- (1) Die gerade erstellte Funktion auswählen
- (2) Die Funktion der Liste der ausgewählten Funktionen hinzufügen
- (3) Handrad-Tastenzuweisung Option auswählen
- (4) Das gewünschte Bildschirm auswählen
- (5) Auf die Taste „Handrad Taste zuweisen...“ klicken. Nach ein paar Sekunden erscheint ein Dialog und auf dem Handrad soll der gewünschte Bildschirm angezeigt werden. Man soll nun auf die gewünschte Taste auf dem Handrad-Bildschirm klicken die zu dem „Tools“-Bildschirm führen soll und nachdem eine Nummer in Rot im Dialog erscheint – auf „Verwenden“ klicken. Diese Nummer erscheint dann in dem Textfeld unter der Taste „Handrad Taste zuweisen...“.
- (6) Jetzt soll auf die Taste „Funktion(en) zuordnen“ klicken und die Zuordnung wird gespeichert. Man kann die Zuordnung im Reiter „Funktionen“ jetzt überprüfen und wenn Konfiguration geschlossen worden ist – die gerade eingerichtete Funktionalität auf dem Handrad prüfen.

Analog dazu kann nun eine Verbindung zu jedem verfügbaren Bildschirm hergestellt werden.



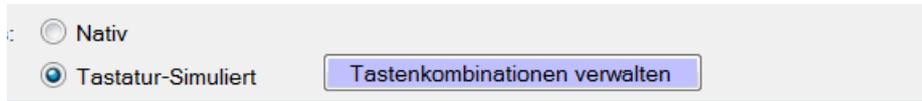
## 6.3 Anlegen einer Tastenkombination

Oft ist es notwendig verschiedene Tastenkombinationen auszulösen. Dazu muss man:

- eine Tastenkombination erstellen
- eine Funktion für diese Tastenkombination erstellen

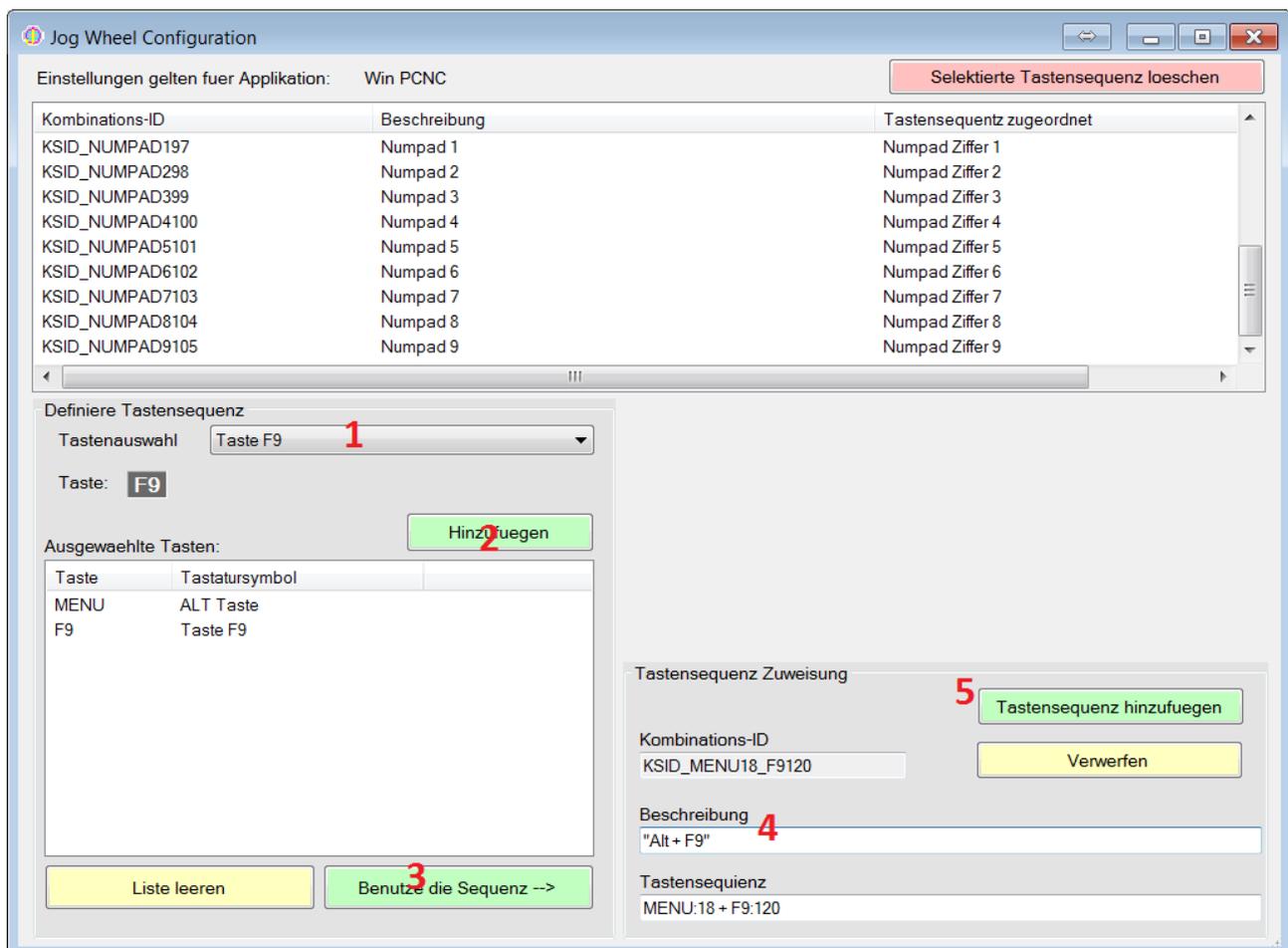
Unten wird gezeigt wie man Tastenkombination „Alt + F9“ erstellt.

Zum erstellen (falls noch nicht erstellt) der gewünschten Tastenkombination klicke man in der Konfiguration auf die Taste „Tastenkombinationen verwalten“:

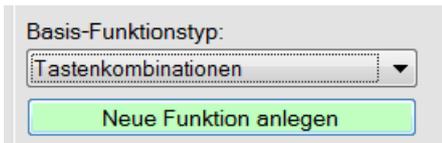


Es erscheint ein Dialog wo man die Tastenkombination erstellen kann und zwar so:

- (1) Unter „Tastenauswahl“ wähle man Taste „Alt“
- (2) Man füge die Taste der Auswahlliste hinzu. Der Vorgang muss für die Taste „F9“ wiederholt werden.
- (3) Wenn das Zusammenstellen der Tastenkombination abgeschlossen ist soll auf „Benutze die Sequenz ->“ geklickt werden. Dadurch wird Tastensequenz in dem entsprechenden Textfeld erscheinen.
- (4) Man Ergänze die Beschreibung
- (5) Jetzt kann die Tastensequenz mit dem Klick auf „Tastensequenz hinzufügen“ angelegt werden.



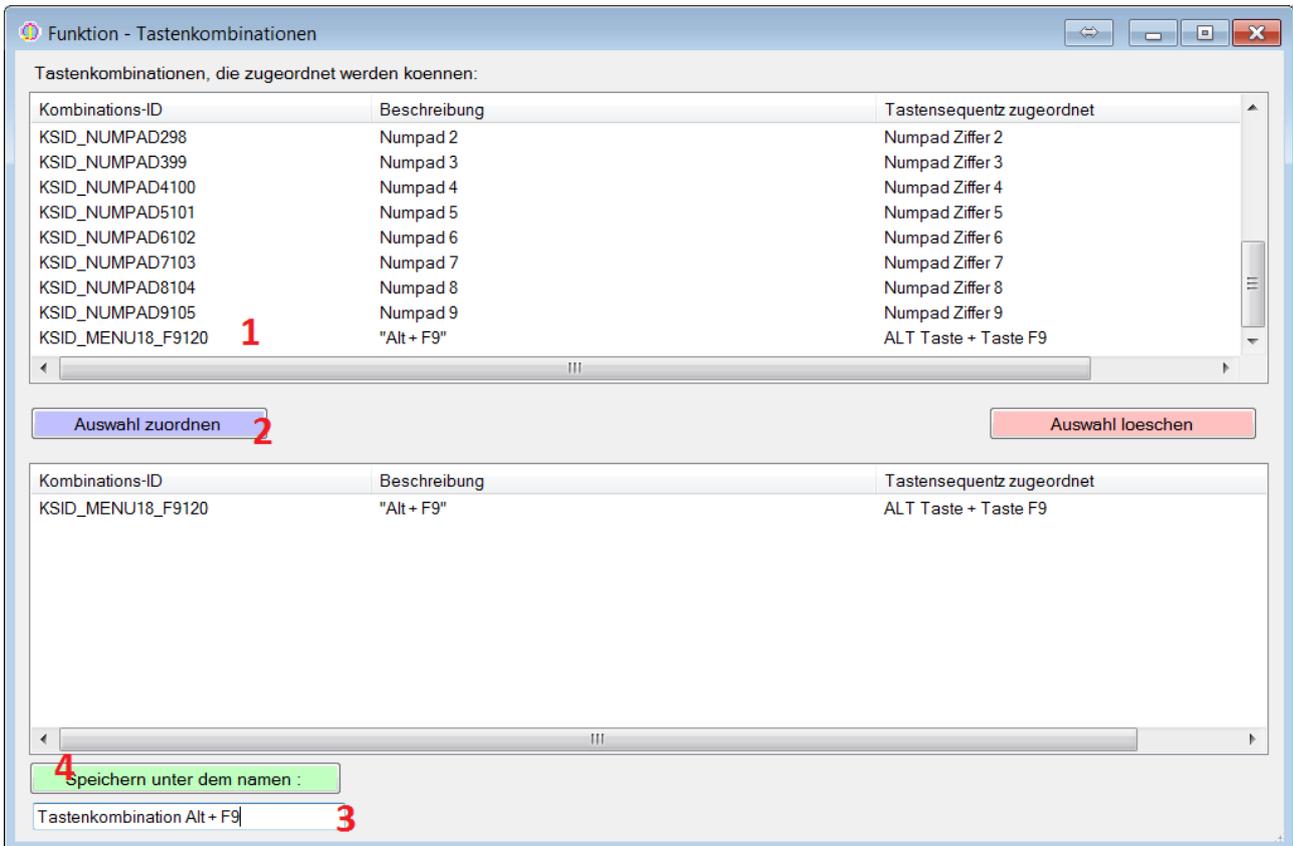
Nun soll die Funktion für die Tastenkombination erstellt werden. Dazu den entsprechenden „Basis-Funktionstyp“ auswählen:



und in dem neuen Dialog die Funktion erstellen:

- (1) Tastenkombination auswählen
- (2) Auswahl zuordnen
- (3) Beschreibung ergänzen
- (4) Speichern

Die so erstellte Funktion kann nun zugeordnet oder im Rahmen eines Funktionspools verwendet werden.



## 6.4 Realisierung einer Tinkerforge-basierten DRO Anzeige für 4 Achsen



Mit Hilfe des Moduls „LED Panel 4 Achsen“ (Datei TFLEDPANEL\_JWMDL.dll“ im Ordner „Plugins“) kann eine digitale Anzeige für Achsen-Koordinaten mit 4 Vorkommastellen und 4 Nachkommastellen realisiert werden. Die DRO Anzeige kann mit jeder Steuersoftware funktionieren die eine Schnittstelle zum Rauslesen der Koordinaten bietet die über ein Plugin realisierbar ist - diesem Beispiel wird die Anbindung an Eding CNC beschrieben.

Im Folgenden wird beschrieben wie die Hardware aufgebaut und Software für Eding CNC konfiguriert werden kann. Dies ist nur mit der freigeschalteten Version möglich, im Demo-Modus sind Module abgeschaltet.

### Hardware

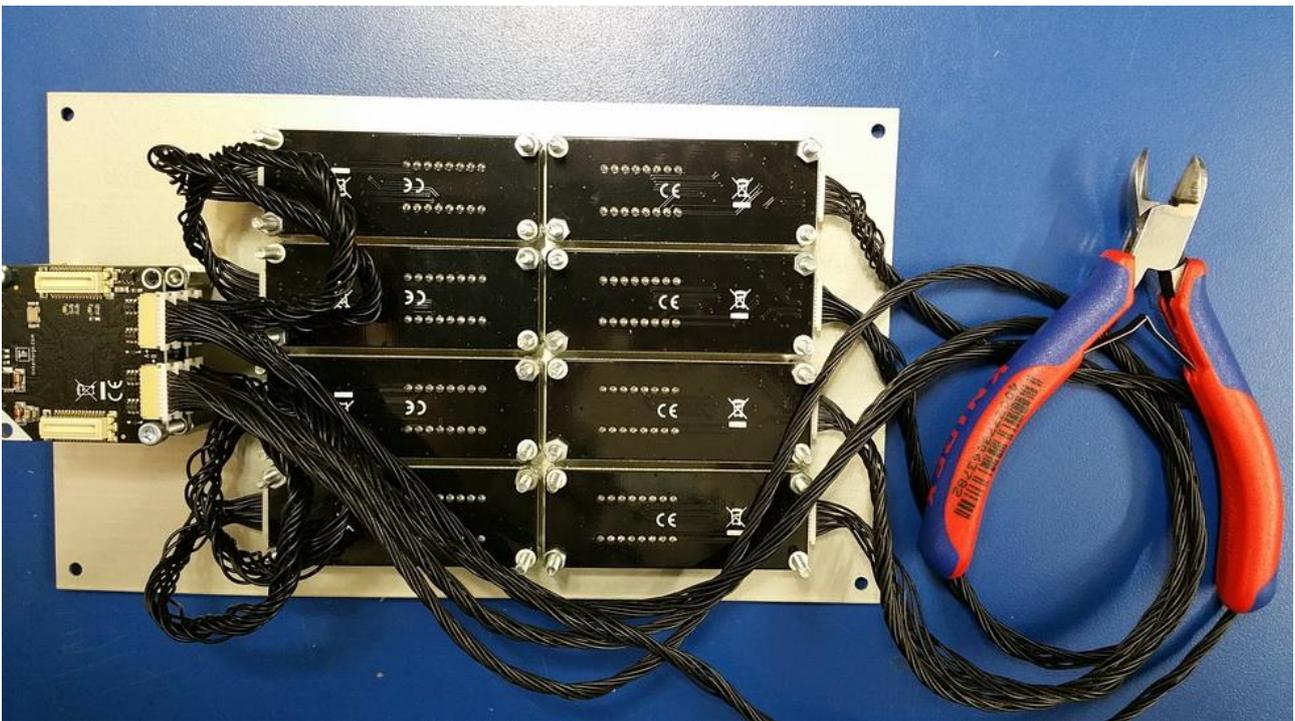
Die Elektronik der DRO Anzeige ist komplett mit den Bausteinen aus dem Tinkerforge Sortiment realisiert. Es werden keine Elektronikkenntnisse benötigt, es wird auch nichts gelötet – alle Module werden über Kabel mit Steckverbindern miteinander verbunden. Eine Externe Stromversorgung ist ebenfalls nicht erforderlich, Anschluss über USB versorgt die Elektronik auch mit Strom (ein normaler 500mA USB Anschluss vorausgesetzt).

Falls allerdings eine DRO Anzeige über WIFI mit dem Rechner verbunden wird, dann muss eine externe Stromquelle (5V) angeschlossen werden.

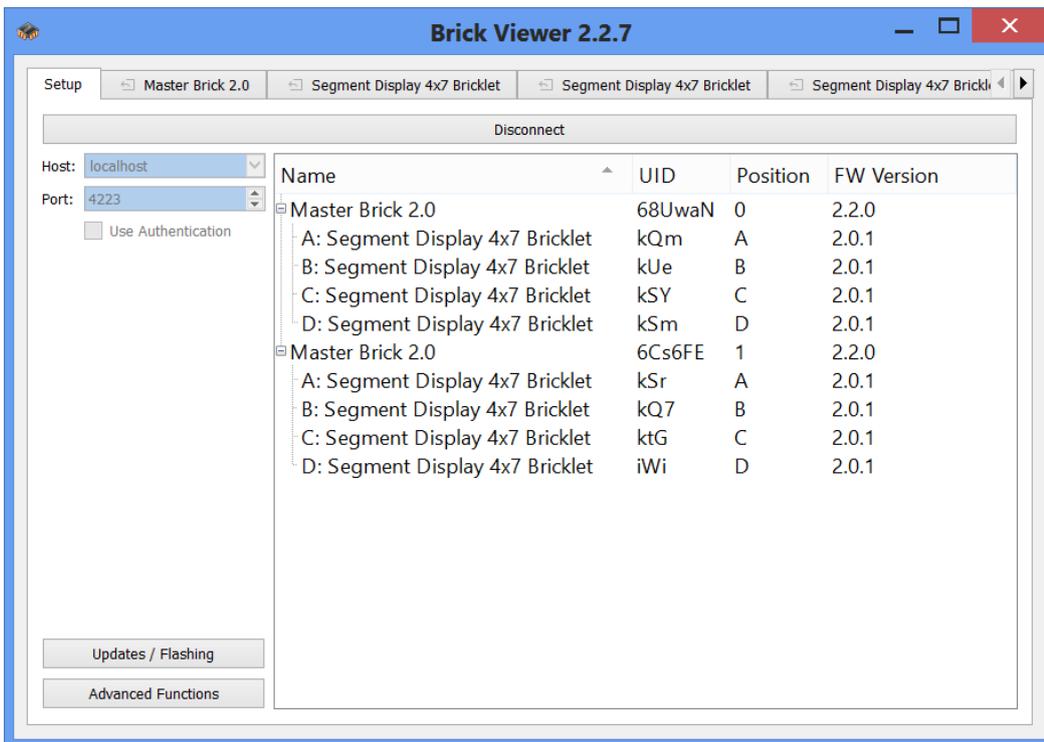
Folgende Tinkerforge-Komponenten werden benötigt:

- Master Brick (<https://www.tinkerforge.com/de/shop/bricks/master-brick.html>) – 2 St.
- Segment Display 4x7 Bricklet (<https://www.tinkerforge.com/de/shop/bricklets/segment-display-4x7-bricklet.html>) – 8 St.
- Verbindungskabel, z.b. 50 cm (<https://www.tinkerforge.com/de/shop/accessories/bricklet-cable-black-50cm.html>) – 8 St.
- Mini USB Kabel (z.b.: <https://www.tinkerforge.com/de/shop/accessories/mini-usb-cable-180cm.html>)
- Falls über WIFI angeschlossen werden soll: 1 St. WIFI Master Extension (<https://www.tinkerforge.com/de/shop/master-extensions/wifi-master-extension.html>) und eine externe Stromversorgung, z.b. USB Netzteil.

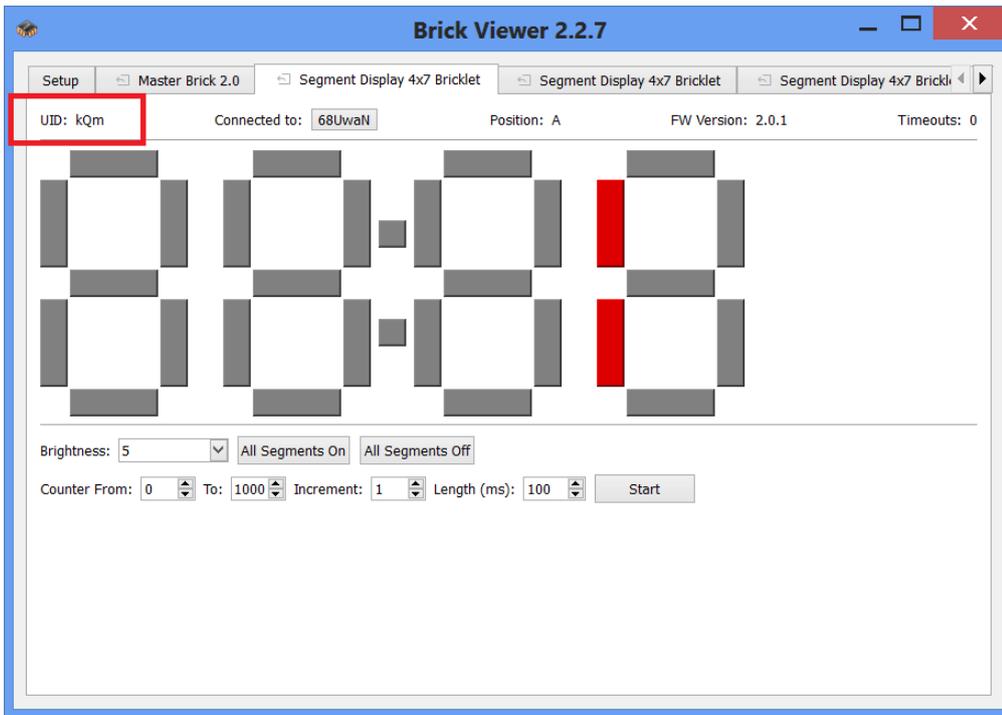
Verbinden Sie die Komponenten miteinander wie in dem Bild unten dargestellt so dass beide Master Bricks aufeinander gesteckt sind und an jeden Master Brick 4 Segment Display Bricks angeschlossen sind. Dabei ist die Reihenfolge irrelevant – später kann man die Anordnung über die Softwarekonfiguration vornehmen. Es ist allerdings jetzt schon vom Vorteil die Segment Bricklets auf einem Gestell so anzuordnen wie die später in der fertigen DRO Einheit verwendet werden sollen. Wichtig ist zu wissen dass die Segments einer Achse mit den Anschlussfreien kurzen Seiten zueinander angeordnet werden müssen (weil die Werte auf den Segmenten für Nachkommastellen dann gedreht dargestellt werden)



Bitte laden und installieren Sie Brick Deamon und Brick Viewer Software von der Tinkerforge-Seite (<http://www.tinkerforge.com/de/doc/Downloads.html>). Schließen Sie den USB Kabel an den unteren Master Brick und dann an den Rechner und starten Sie Brick Viewer. Die Master Bricks und Segment Bricklets sollten erkannt und angezeigt werden, z.b.:

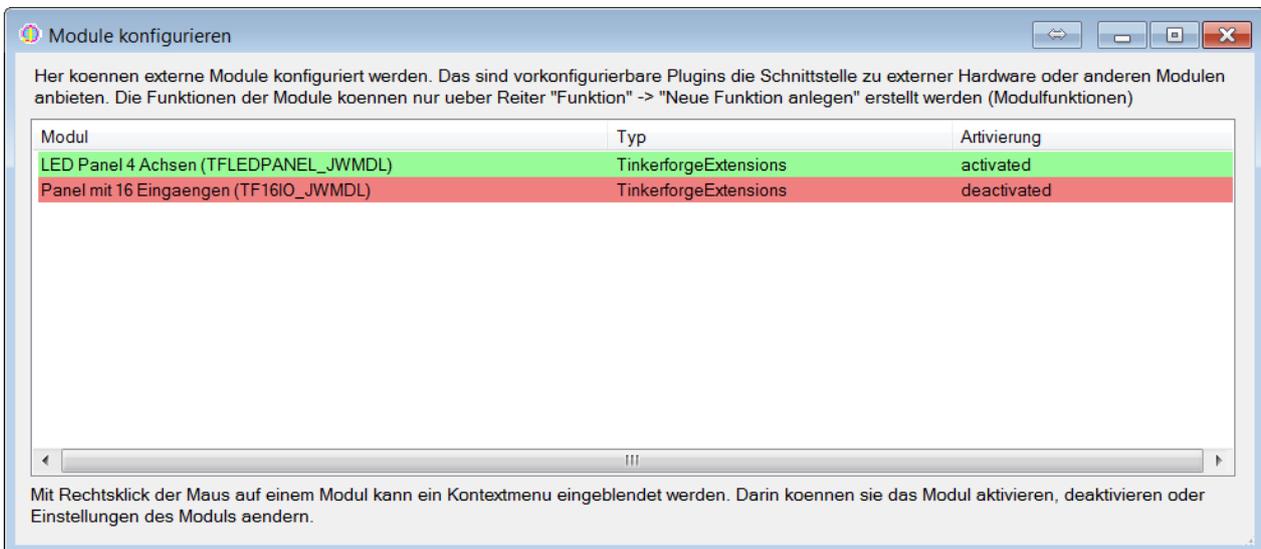


Notieren Sie die UID's von jedem der Segment Bricklets (das wird später für Konfiguration gebraucht). Mit Hilfe der Testfunktion des Brick Viewers können Sie jetzt schon anordnen welcher Bricklet zu welcher Achse gehört (X, Y, Z oder A) und ob es Vor- oder Nachkommastellen darstellen soll:

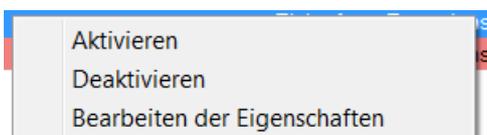


## Software- Konfiguration

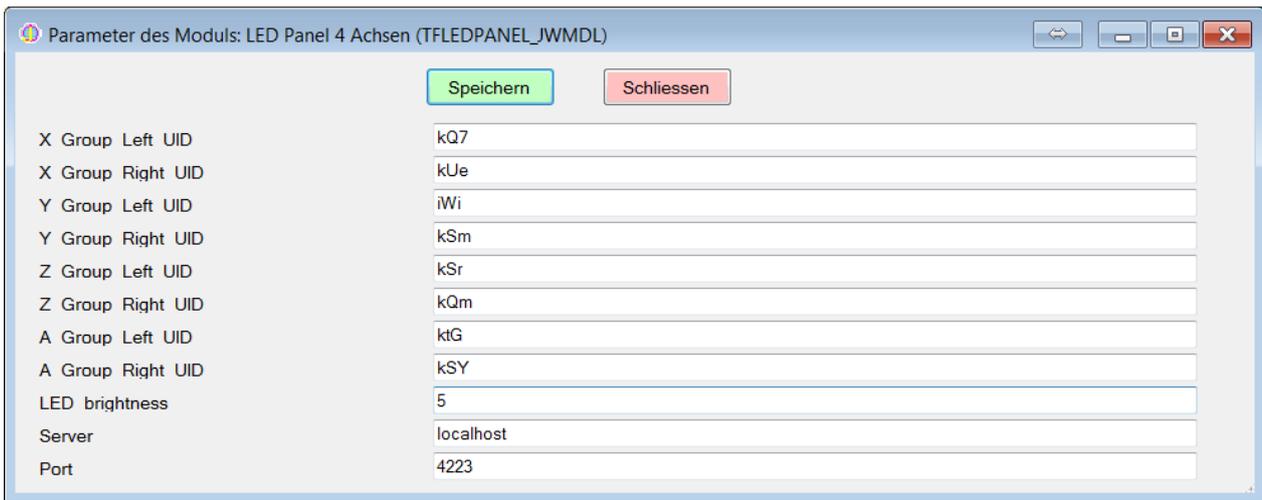
Um das Modul zu konfigurieren starten Sie JWC und klicken auf „Module Konfigurieren“. Es erscheint ein Dialog in dem Sie alle verfügbaren Module sehen können. Die Farbe der Einträge in der Liste entspricht dem Aktivierungsstatus des Moduls: Rot - deaktiviert, Grün – aktiviert:



Um ein Modul zu konfigurieren oder zu aktivieren/deaktivieren klicken Sie auf das entsprechende Modul mit der rechten Maustaste und wählen Sie die gewünschte Aktion:



Nachdem das Modul aktiviert worden ist, klicken Sie in dem Kontextmenu auf „Bearbeiten der Eigenschaften“. Daraufhin erscheint ein Dialog wo Sie die zuvor notierten UID's der Segments der Anzeige eintippen können:



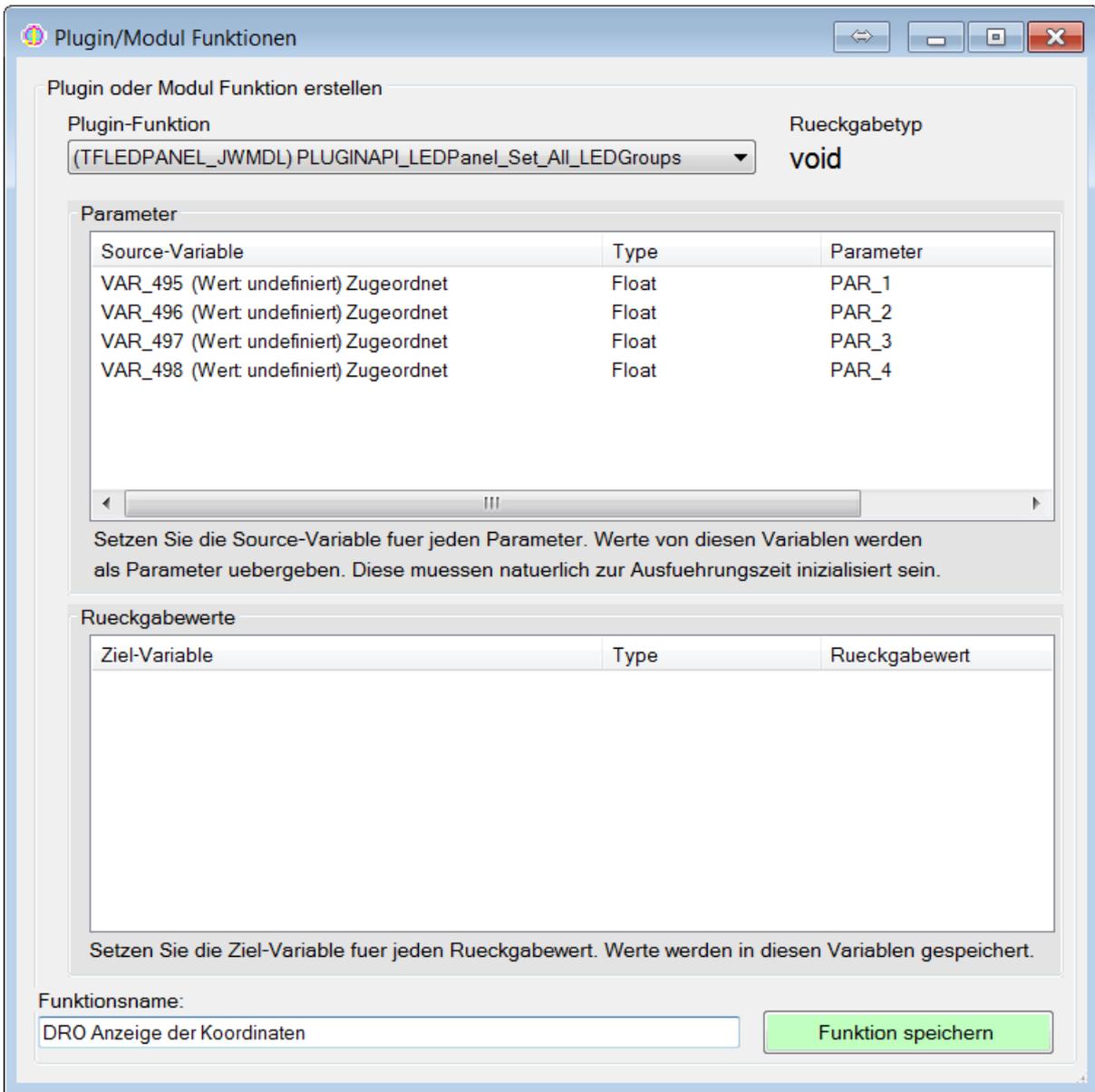
„Group Left UID“ enthält Vorkommastellen, „Group Right UID“ enthält Nachkommastellen der jeweiligen Achse. Stellen Sie sicher dass „Server“ und „Port“ richtig gesetzt sind. Default – Werte dafür sind „localhost“ und „4223“, stellen Sie sicher dass dieser Port nicht vom Firewall blockiert wird (Sie können das mit dem „Brick Viewer“ testen).

Mit „LED Brightness“ können Sie die Helligkeit der Anzeige einstellen. Klicken Sie Anschließend auf „Speichern“ und Starten Sie den JWC neu.

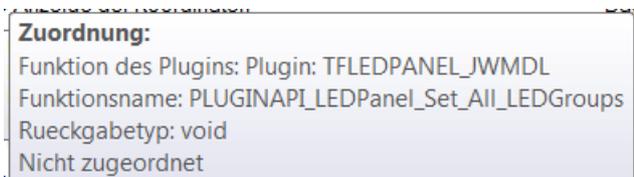
Anschließend klicken Sie auf „Handrad-Parameter Konfiguration“ und im Reiter „Funktionen“ soll nun eine Plugin-Funktion erstellt werden die Koordinaten anzeigt. Diese Funktion ist bereits unter dem Namen „LED Anzeige Modul: Werte anzeigen“ angelegt aber unten soll nochmal der Erstellungsprozess gezeigt werden:



Klicken Sie auf „Neue Funktion anlegen“. Erscheint ein Dialog indem Sie die Felder wie folgt setzen sollen:

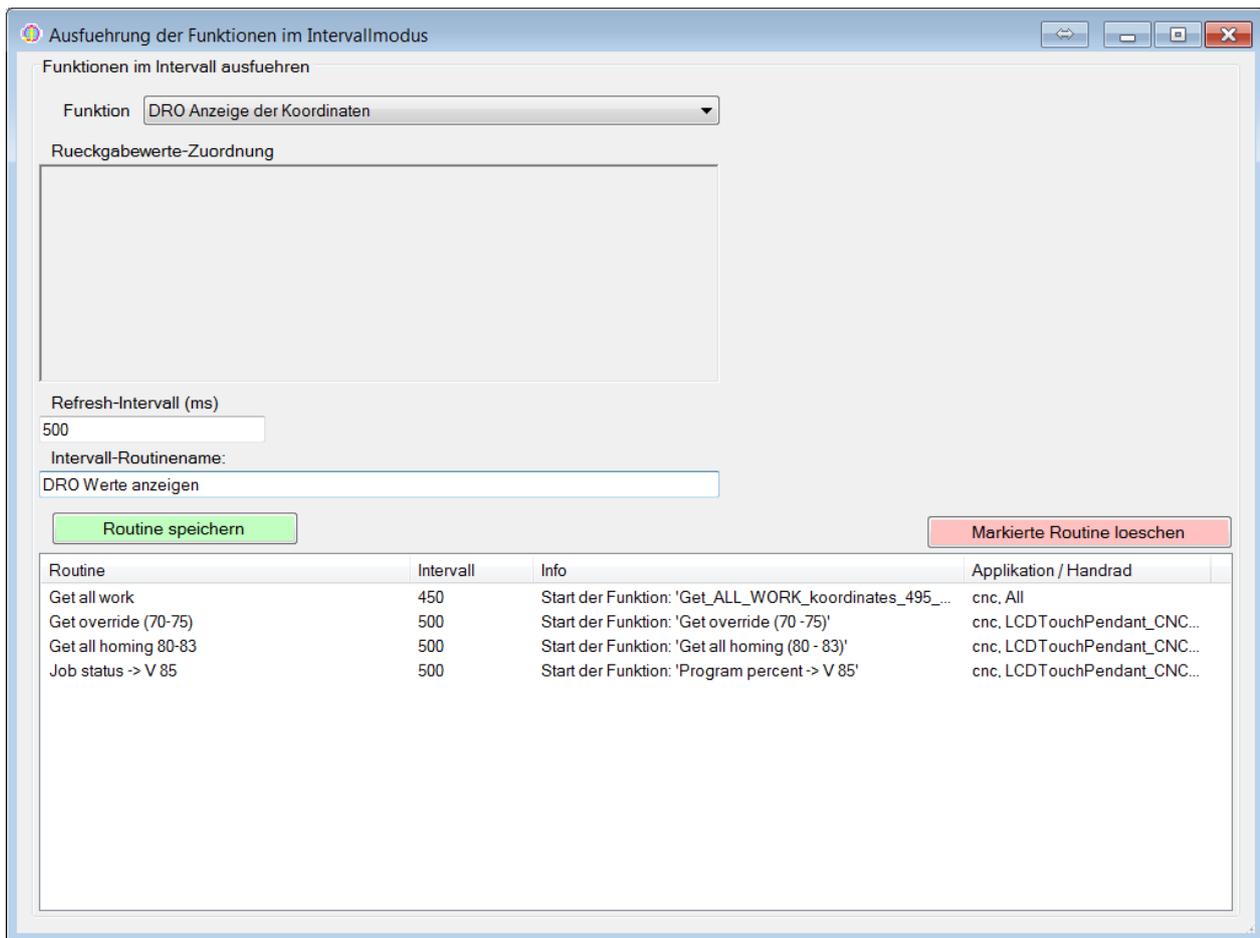


Setzen Sie bei der Erstellung der Funktion für die Parameter die Variablen aus denen die Koordinaten für Var\_495 = X, Var\_496 = Y, Var\_497 = Z, Var\_498 = A herausgelesen werden sollen. Vergeben Sie dann einen Funktionsnamen und Klicken Sie auf „Funktion speichern“. Wenn alles geklappt hat, werden Sie die Funktion und deren Zuordnungsdaten in der Funktionsliste sehen:



Nun muss die Funktion mit den Parametern versorgt werden. Man kann sich eine neue Funktion anlegen doch wir verwenden in diesem Beispiel eine bereits existierende Funktion „Get\_ALL\_WORK\_koordinates\_495\_498“ die auch schon eingerichtet ist und über „Intervallbasierte Ausführung der Funktionen“ jede 450 ms. ausgeführt wird sodass die definierten Variablen mit aktuellen Werten von Eding CNC gefüllt werden.

Damit nun diese Werte auf der Anzeige angezeigt werden muss unsere, zuvor erstellte Funktion „DRO Anzeige der Koordinaten“ auch in einem passenden Intervall ausgeführt werden. Dafür klicken Sie auf den Knopf „Intervallbasierte Ausführung der Funktionen“ dass im Reiter „Weitere Einstellungen“ zu finden ist und nehmen Sie die Einstellungen wie folgt vor:



Abschließend klicken Sie auf „Routine speichern“ und starten Sie den JWC neu. Wenn alles funktioniert hat werden Sie die aktuellen Koordinaten auf der Anzeige sehen können.

## 6.5 Handrad-Schrittweite Modus rotierend umschalten mit nur einer Taste

In JWC kann man mit nur einer Taste zwischen verschiedenen Modi umschalten. Wie das funktioniert wird in diesem Beispiel veranschaulicht. Wir verwenden das Profil für „Eding CNC“ und „LCD Touch Pendant“ im „Native“ Modus.

Die Erstellung jeder komplexen Funktion fängt immer mit der Erstellung von einfachen Funktionen. Diese werden dann zusammengefasst und in Kombinationen verwendet wobei die Komplexität mit jeder Stufe ansteigt. Dabei gibt es viele Möglichkeiten das Gleiche zu erreichen, hier wird eine der Möglichkeiten präsentiert.

Um mit einer Taste die Schrittweite in „Single-Step“ Betrieb umzuschalten müssen zuerst Variablen definiert werden die für das jeweilige Zustand / Schrittwert stehen. Dazu sucht man vier unbenutzte Variablen. Bei Eding CNC gibt es 4 Stufen die wir hier nutzen, wir definieren also die Zustandsvariablen für einzelne Schrittwerte wie folgt:

```
Var_20 -1.0
Var_21 - 0.1
Var_22 - 0.01
Var_23 - 0.001
```

Diese Variablen sollen:

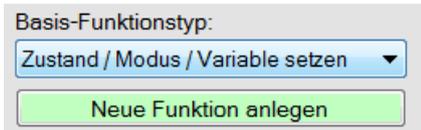
- von Typ „Bool“ sein (also nur An oder Aus)
- Mit initial-Werten initialisiert werden;

- Mit Hilfe einer Handrad-Taste umgeschaltet werden;
- Ggf. mit Bildelementen verknüpft werden sodass eine Zustandsänderung sogleich auf dem Bildschirm des Handrades angezeigt wird;

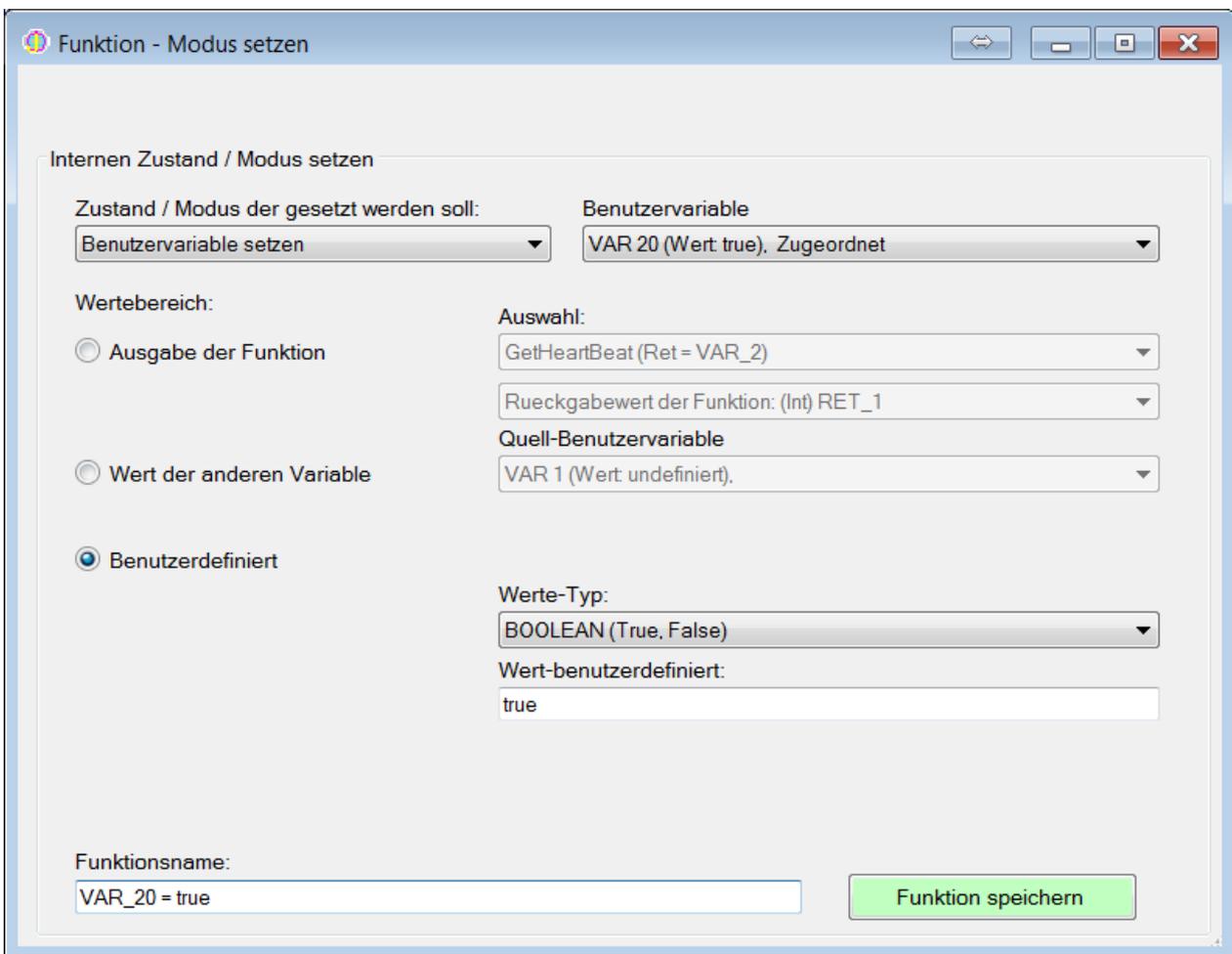
**Anlegen der Elementar-Funktionen**

Zuerst müssen (wenn nicht schon vorhanden) die Funktionen angelegt werden die überhaupt eine Variable auf einen definierten Wert setzen. Am Beispiel der Variable Var\_20 (zu den anderen analog verfahren) :

wir legen jeweils eine Funktion an die den Wert auf „True“ (An) oder „False“ (Aus) setzt. Das kann man über den Basis-Funktionstyp „Zustand / Modus / Variable setzen“ realisieren:



Hier ein Beispiel der Funktion für das Setzen der Variable 20 auf „true“:



Analog dazu erstellt man alle anderen Funktionen für Variablen 20- 23. Es soll dann in Funktionstabelle in etwa so aussehen:

VAR_20 = true	Basic_SetInternalMode
VAR_20 = false	Basic_SetInternalMode
VAR_21 = true	Basic_SetInternalMode
VAR_21 = false	Basic_SetInternalMode
VAR_22 = true	Basic_SetInternalMode
VAR_22 = false	Basic_SetInternalMode
VAR_23 = true	Basic_SetInternalMode
VAR_23 = false	Basic_SetInternalMode

**Anlegen der Funktionalität**

Die Information welche Schrittstufe ausgewählt ist muss auch an die Steuersoftware übergeben werden. Da wir uns ja im „Native“ Modus befinden – werden die Befehle an die Steuersoftware über Plugin-Funktionen übergeben. In unserem Fall ist es im Reiter „Native Einstellungen“ so festgelegt dass Variable 10 den Einzelschrittwert beinhalten muss:

Plugin-Jogging-Funktion		
<input type="text" value="(USBCNC_PLG) PLUGINAPI_JogMove_Track"/>		
Schritt-Variable	<input type="text" value="VAR 10 (Wert 1.0), Zugeordnet"/>	Fließkommazahl in Millimeter.
Modus-Auswahl-Variable	<input type="text" value="VAR 11 (Wert 1), Zugeordnet"/>	Muss vom Typ INTEGER sein.

daher müssen wir auf die oben gezeigte Art vier Funktionen erstellen die den Wert der Variable 10 auf den jeweils gewünschten Schrittwert setzen (Achtung, diesmal vom Typ FLOAT !). Dann soll es in der Funktionstabelle in etwa wie folgt aussehen:

Setze VAR_10 = 1.0	Basic_SetInternalMode
Setze VAR_10 = 0.1	Basic_SetInternalMode
Setze VAR_10 = 0.01	Basic_SetInternalMode
Setze VAR_10 = 0.001	Basic_SetInternalMode

**Kombination der Funktionen**

Jetzt soll überlegt werden wie bei einem Antippen einer Taste die Modi umgeschaltet werden. Hierfür wird noch eine Funktion benötigt um den Wert Variable 10 auf 0 zu setzen. Erstellen Sie eine solche Funktion analog zu den anderen:

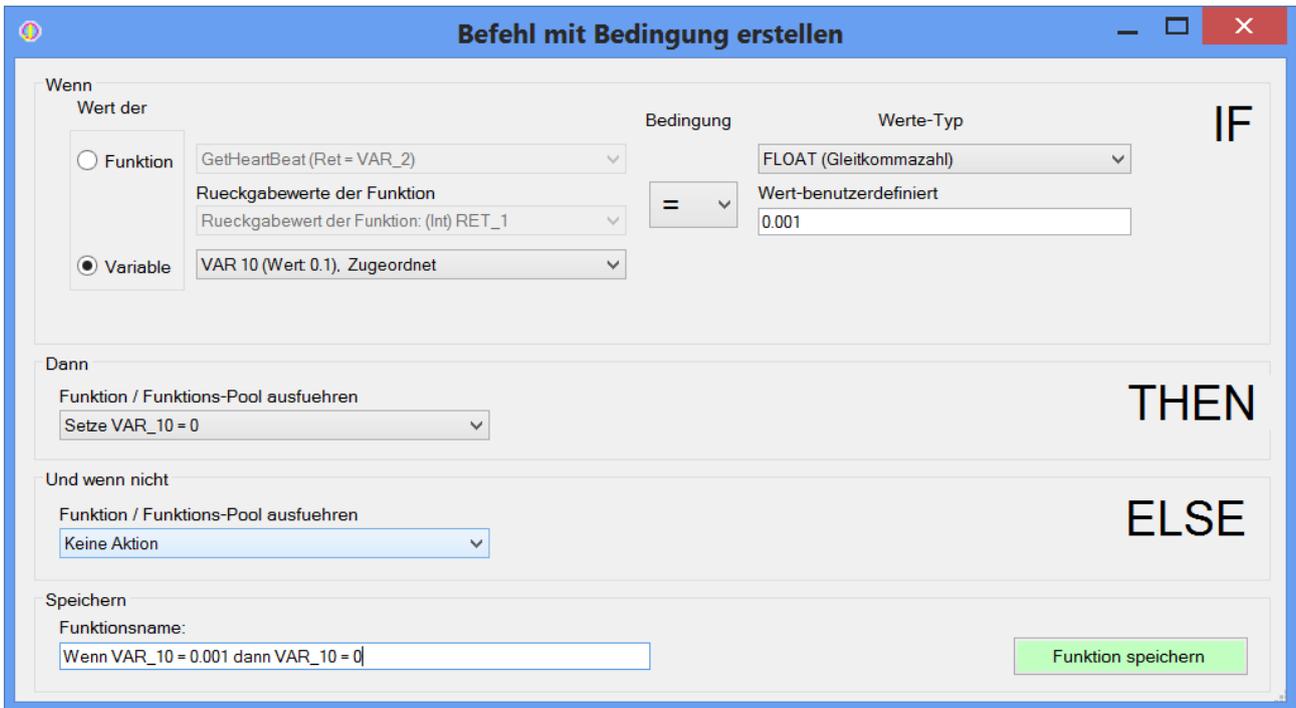
Setze VAR_10 = 0	Basic_SetInternalMode
------------------	-----------------------

Jetzt werden Funktionen mit Bedingungen gebraucht um die Rotation zu ermöglichen:

Basis-Funktionstyp:
<input type="text" value="Befehl mit Bedingung erstellen"/>
<input type="button" value="Neue Funktion anlegen"/>

Die Idee ist: wenn Variable 10 einen Wert grösser 0 enthält – wird zu dem jeweils nächsten Wert umgeschaltet. Wenn der letzte Wert erreicht ist (0.001) dann wird zuerst zu 0 und im gleichen Zug zu dem ersten Wert umgeschaltet (1.0). Unten ist die erste Funktion dargestellt:

„Wenn VAR\_10 = 0.001 dann VAR\_10 = 0“



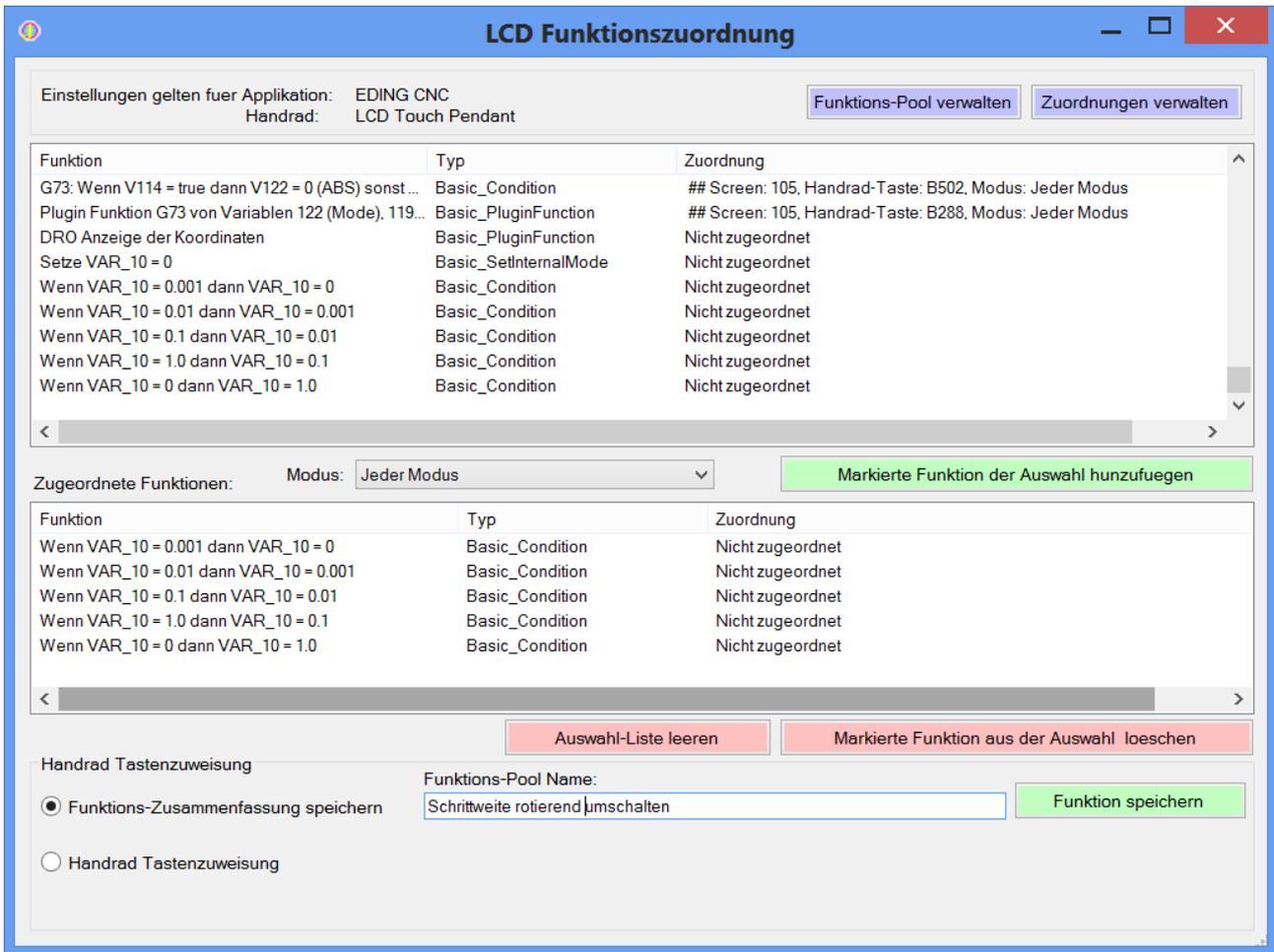
Nun sollen analog alle anderen erstellt werden:

- Wenn VAR\_10 = 0.01 dann VAR\_10 = 0.001
- Wenn VAR\_10 = 0.1 dann VAR\_10 = 0.01
- Wenn VAR\_10 = 1.0 dann VAR\_10 = 0.1
- Wenn VAR\_10 = 0 dann VAR\_10 = 1.0

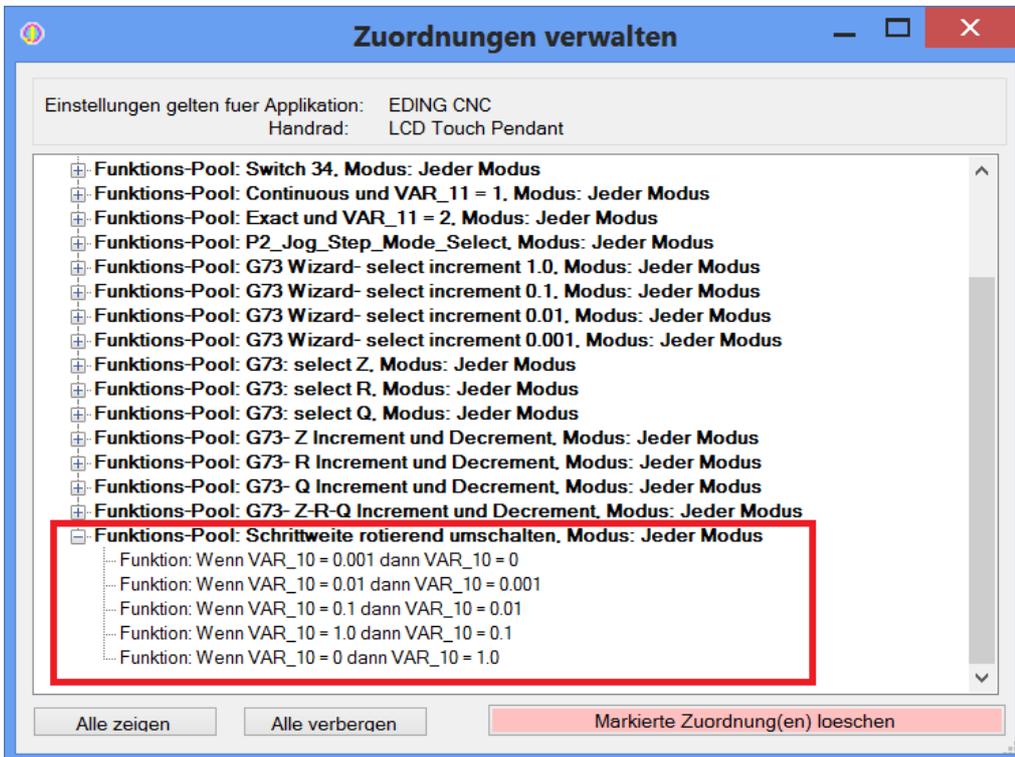
Die so erstellten Funktionen werden wir nun zu einem Funktionspool zusammenfassen. Dazu klicken Sie im Reiter „Funktionen“ die Taste



Daraufhin in dem Nachfolgenden Dialog setzen Sie die Funktionen wie folgt zusammen (Reihenfolge ist wichtig !):



Klicken Sie anschließend auf „Funktion speichern“ und prüfen Sie ob Funktionspool korrekt angelegt worden ist:



Jetzt ist es notwendig sich um die LED-Anzeigen zu kümmern – damit die Anzeige auch rotierend umgeschaltet wird.

Wir haben ja die Funktionen um die Variablen 20 – 23 zu setzen, erstellen wir die notwendigen Rotations-funktionspools:

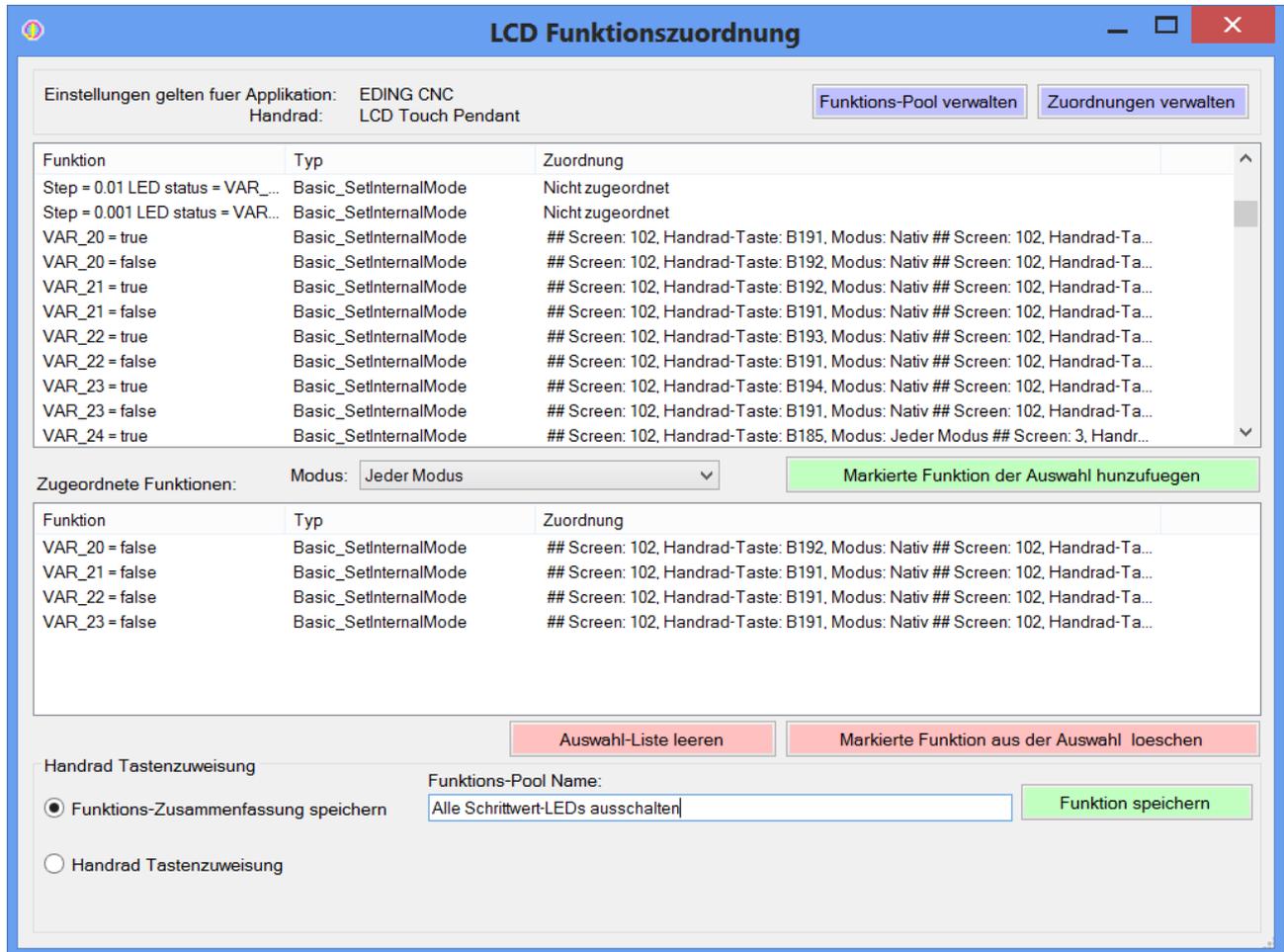
Einmal „Alle Schrittwert-LEDs ausschalten“

VAR\_20 = false

VAR\_21 = false

VAR\_22 = false

VAR\_23 = false



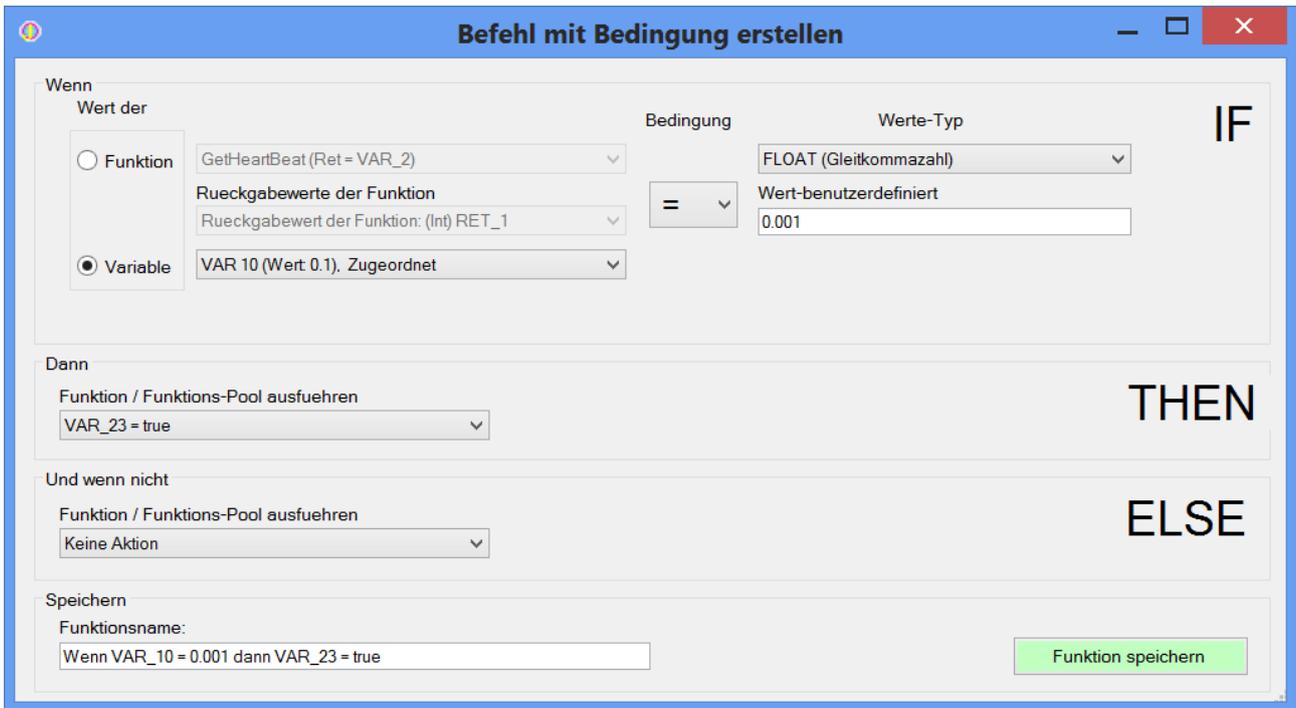
und „Aktuellen Schrittwerte LED setzen“

Wenn VAR\_10 = 0.001 dann VAR\_23 = true

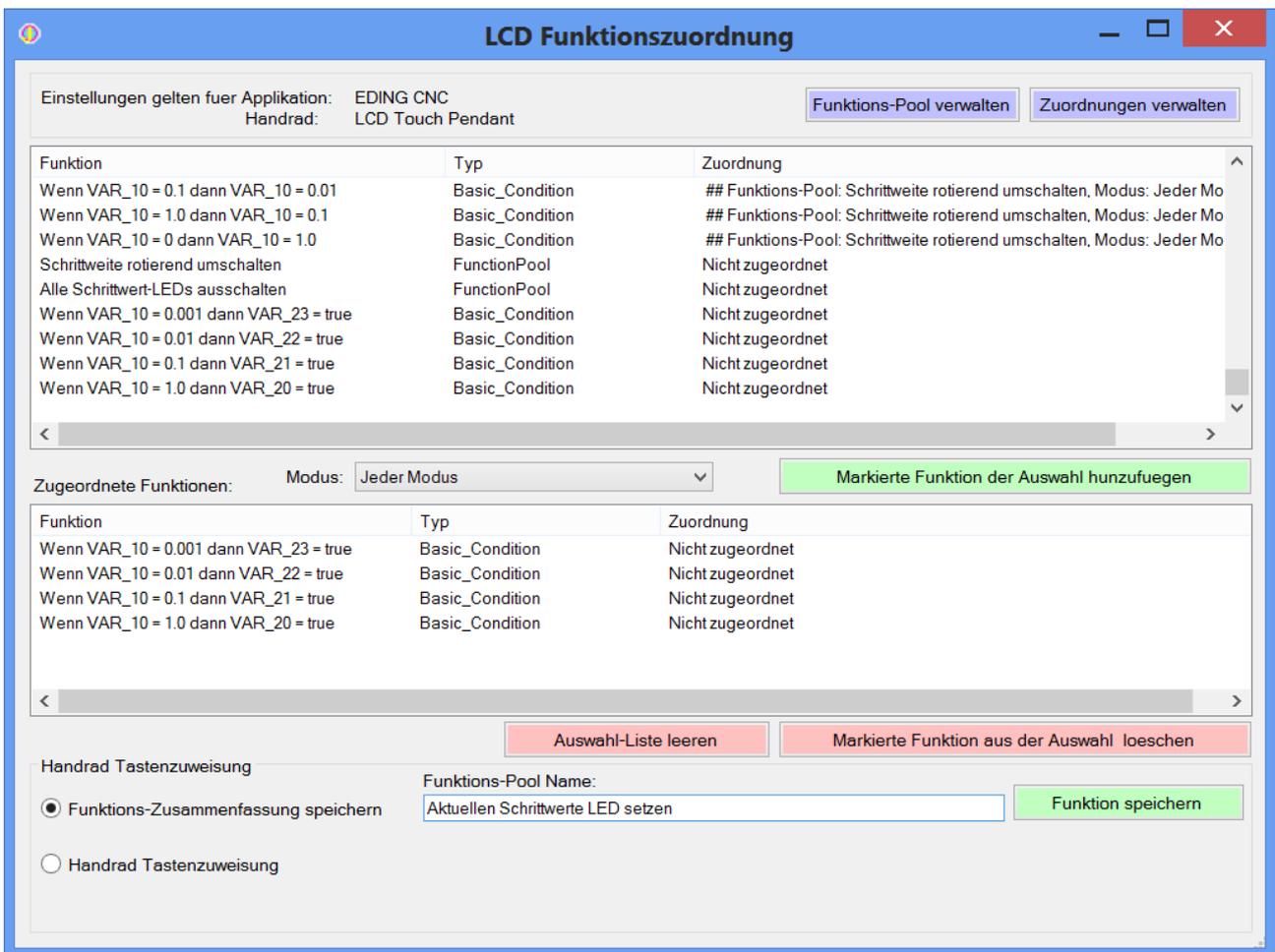
Wenn VAR\_10 = 0.01 dann VAR\_22 = true

Wenn VAR\_10 = 0.1 dann VAR\_21 = true

Wenn VAR\_10 = 1.0 dann VAR\_20 = true



Und



**Zuordnung der so erstellten Funktionspools zu einer Taste des Handrades.**

Jetzt haben wir alles was wir benötigen. Ordnen wir die Funktionspools der ausgewählten Handradtaste zu:

LCD Funktionszuordnung

Einstellungen gelten fuer Applikation: EDING CNC  
 Handrad: LCD Touch Pendant

Funktions-Pool verwalten
Zuordnungen verwalten

Funktion	Typ	Zuordnung
Wenn VAR_10 = 1.0 dann VAR_10 = 0.1	Basic_Condition	## Funktions-Pool: Schrittweite rotierend umschalten, Modus: Jeder M
Wenn VAR_10 = 0 dann VAR_10 = 1.0	Basic_Condition	## Funktions-Pool: Schrittweite rotierend umschalten, Modus: Jeder M
Schrittweite rotierend umschalten	FunctionPool	Nicht zugeordnet
Alle Schrittwert-LEDs ausschalten	FunctionPool	Nicht zugeordnet
Wenn VAR_10 = 0.001 dann VAR_23 = true	Basic_Condition	## Funktions-Pool: Aktuellen Schrittwerte LED setzen, Modus: Jeder I
Wenn VAR_10 = 0.01 dann VAR_22 = true	Basic_Condition	## Funktions-Pool: Aktuellen Schrittwerte LED setzen, Modus: Jeder I
Wenn VAR_10 = 0.1 dann VAR_21 = true	Basic_Condition	## Funktions-Pool: Aktuellen Schrittwerte LED setzen, Modus: Jeder I
Wenn VAR_10 = 1.0 dann VAR_20 = true	Basic_Condition	## Funktions-Pool: Aktuellen Schrittwerte LED setzen, Modus: Jeder I
Aktuellen Schrittwerte LED setzen	FunctionPool	Nicht zugeordnet

<
>

Zugeordnete Funktionen: Modus: Jeder Modus Markierte Funktion der Auswahl hinzufuegen

Funktion	Typ	Zuordnung
Schrittweite rotierend umschalten	FunctionPool	Nicht zugeordnet
Alle Schrittwert-LEDs ausschalten	FunctionPool	Nicht zugeordnet
Aktuellen Schrittwerte LED setzen	FunctionPool	Nicht zugeordnet

<
>

Auswahl-Liste leeren
Markierte Funktion aus der Auswahl loeschen

Handrad Tastenzuweisung

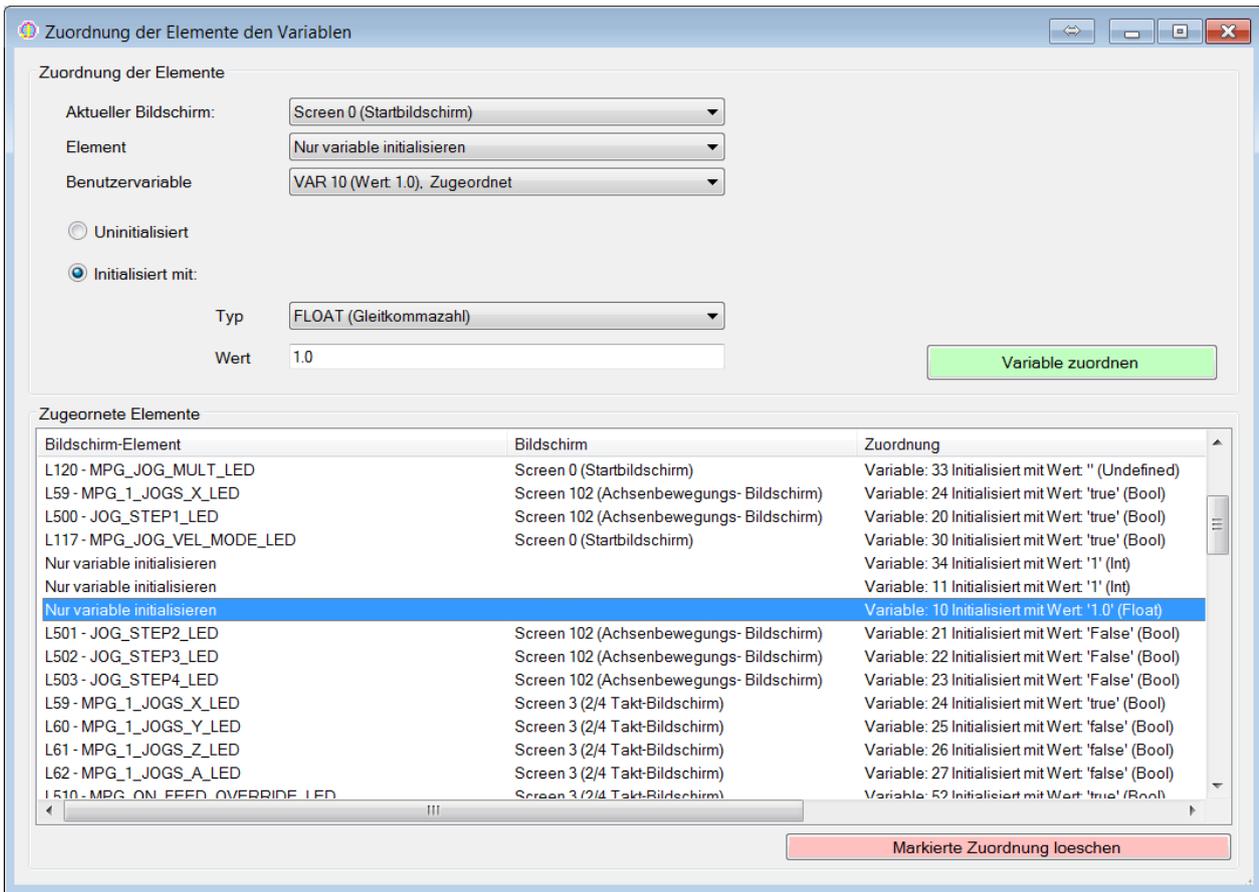
Funktions-Zusammenfassung speichern  
 Handrad Tastenzuweisung
 Handrad Taste zuweisen...
Funktion(en) zuordnen

B192
Ausgewaehelter LCD-Bildschirm: Screen 102 (Achsenbewegungs- Bildschirm)

**Initialisierung der Variablen**

Als letztes muss man noch sich darum kuummern dass die Variable 10 beim Start mit dem richtigen Initialwert (1.0) belegt wird. Das ist in dem mitgelieferten Profil bereits der Fall, doch wenn man einen anderen Initialwert vorzieht – hier die Vorgehensweise:

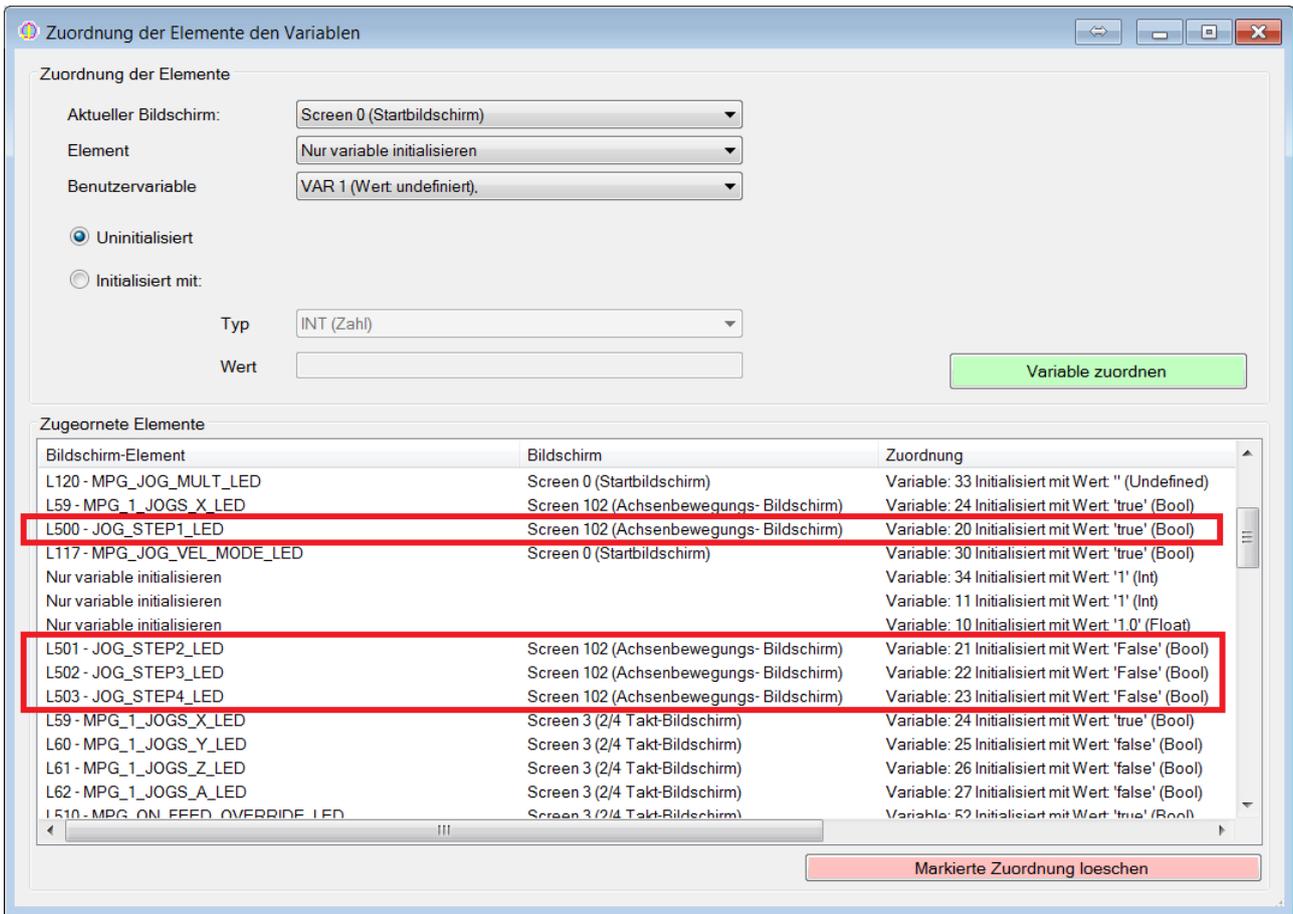
In den Einstellungen auf dem Reiter „Weitere Einstellungen“ klickt man auf den Knopf „Zuordnung der Elemente zu den Variablen“. In dem geoeffneten Dialog kann man die Zuordnung dann wie folgt vornehmen:



Wie Sie in dem markierten Text sehen – ist diese Zuordnung bereits existent. Wenn Sie jetzt auf „Variable zuordnen“ klicken würden – erscheint eine Fehlermeldung die besagt dass man zuerst die bestehende Zuordnung löschen soll.

**Zuordnung der Variablen zu den Bildschirmelementen**

Auch die Zuordnung der LEDs zu den Variablen 20-23 existiert bereits, wenn der Wert der zugeordneten Variable sich ändert, wird das zugleich auf dem Handrad-Bildschirm angezeigt:



falls man andere LEDs verwenden will, soll man die bestehenden Zuordnungen löschen und wie gewünscht neu anlegen.

## 6.6 Realisierung eines Tinkerforge-basierten Handrades mit Drehencoder und 4 Tasten

Neben LCD Touch Pendant Handrad ist es möglich ein selbstgebautes (oder ein bereits verfügbares) Handrad an JWC anzuschließen. Die hierfür notwendigen elektronischen Komponenten sind:

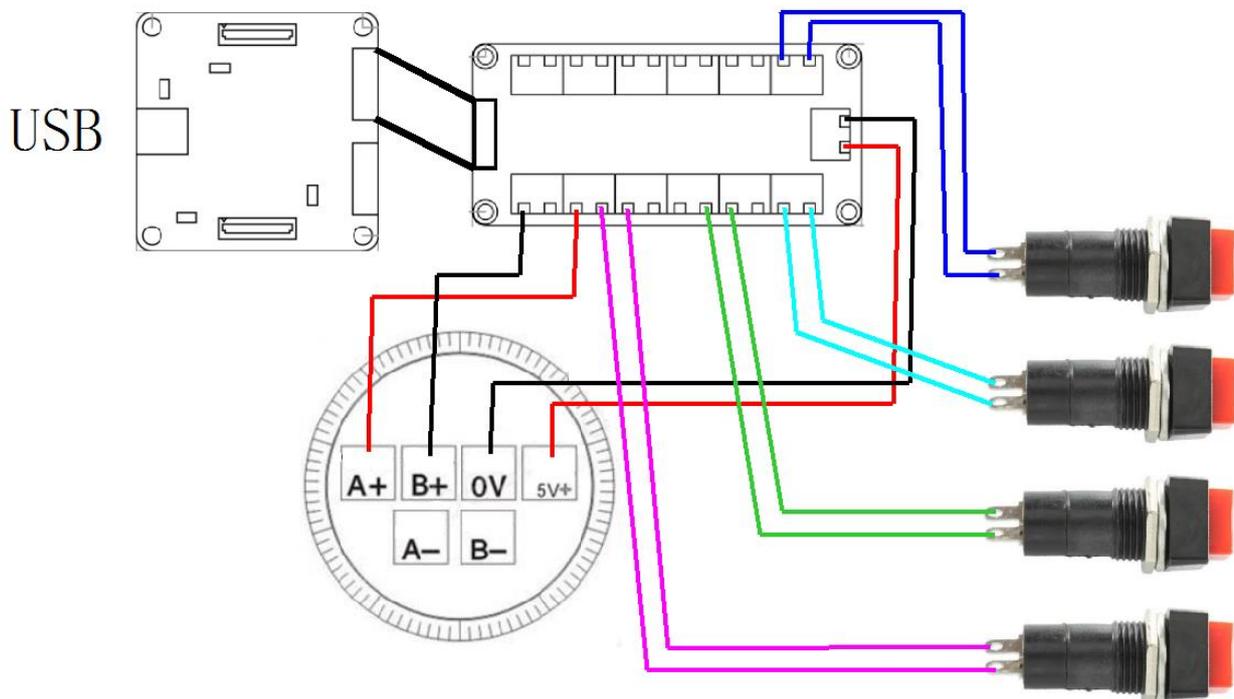
- Master Brick (<https://www.tinkerforge.com/de/shop/bricks/master-brick.html>) – 1 St.
- IO-16 Bricklet Bricklet (<https://www.tinkerforge.com/de/shop/bricklets/io16-bricklet.html>) – 1 St.
- Verbindungskabel, z.B. 50 cm (<https://www.tinkerforge.com/de/shop/accessories/bricklet-cable-black-50cm.html>) – 1 St.
- Mini USB Kabel (z.B.: <https://www.tinkerforge.com/de/shop/accessories/mini-usb-cable-180cm.html>)
- Falls über WIFI angeschlossen werden soll: 1 St. WIFI Master Extension (<https://www.tinkerforge.com/de/shop/master-extensions/wifi-master-extension.html>) und eine externe Stromversorgung, z.B. USB Netzteil.

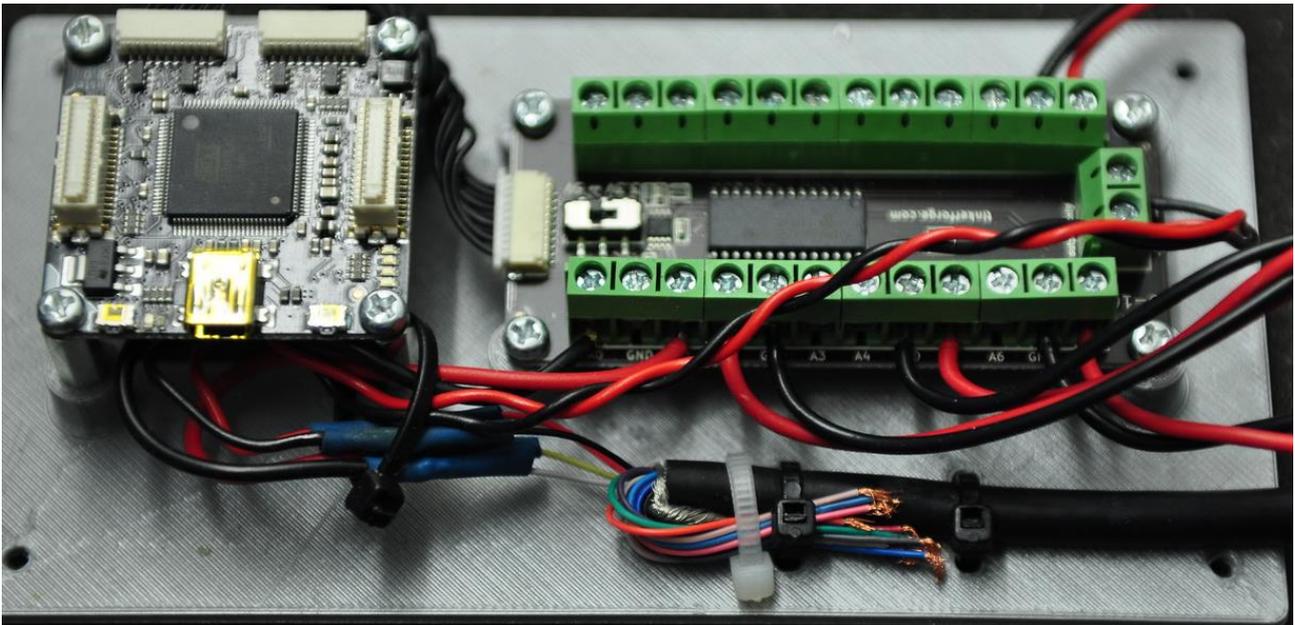
Der Encoder des anzuschließenden Handrades muss A und B Rechtecksignale ausgeben und mit 5V mit Strom versorgt werden können. IO-16 Tinkerforge Modul hat 16 Eingänge, von denen 2 für den Encoder reserviert sind und 14 für Tasten verwendet werden können.

Bitte zu beachten: Encoder muss zwingend an die A0 und A1 Eingänge angeschlossen werden, für die Tasten ist es irrelevant an welche Eingänge die angeschlossen werden, man soll nur aufpassen dass für eine Taste immer ein „A“-Eingang zusammen mit einer „GND“ Klemme verwendet wird.

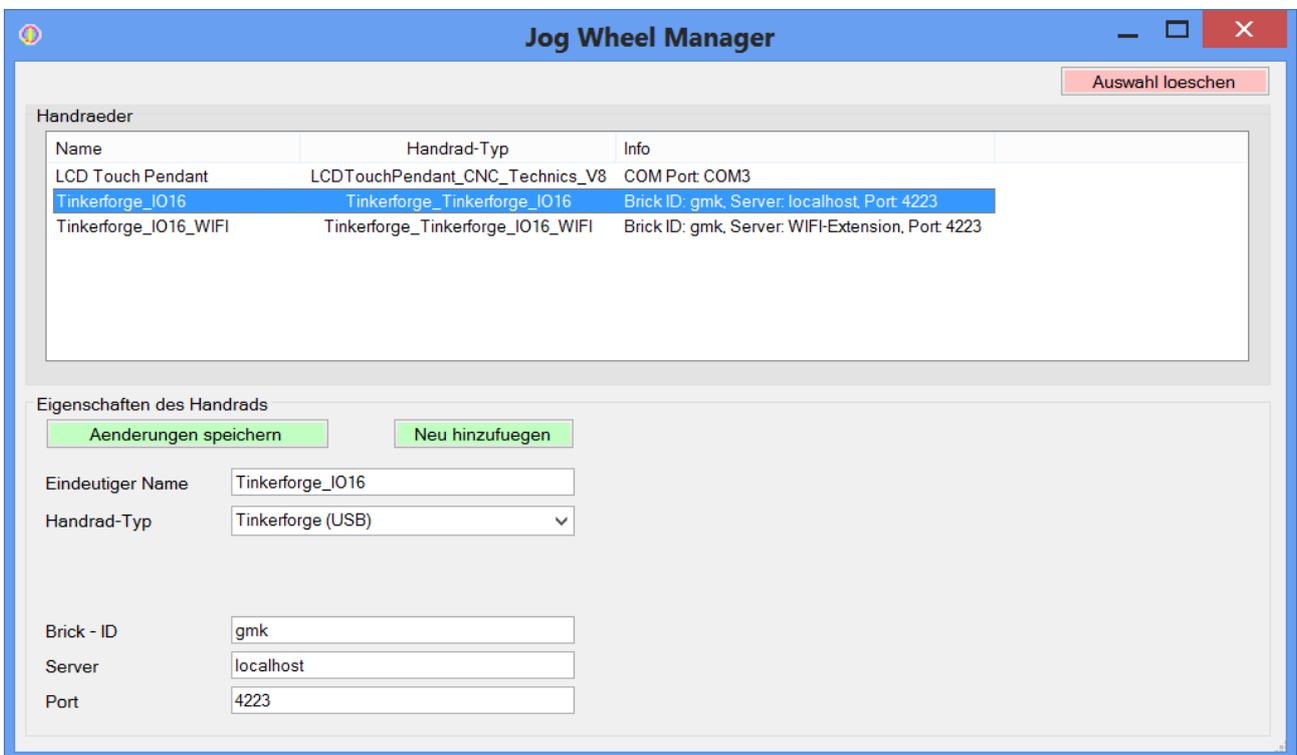
Der Spannungsschalter der IO16 Bricklets muss auf „5V“ stehen !

Unten ist ein Beispiel einer Belegung für ein Handrad mit 4 Tasten abgebildet:





Das so angeschlossene Handrad muss nun in JWC konfiguriert werden. Dafür soll das entsprechende Handrad in JWC zuerst ausgewählt und mit dem Klick auf die Schaltfläche „Handrad-Verwaltung“ konfiguriert werden:



In dem Dialog soll man nun den Eintrag „Tinkerforge\_IO16“ auswählen und Brick-ID anpassen. Welcher ID ihr IO-16 Bricklet hat können Sie über Tinkerforge Brick Viewer erfahren.

Wenn Sie das Handrad über WLAN angeschlossen haben – können Sie den anderen Eintrag auswählen „Tinkerforge\_IO16\_WiFi“ oder die Werte für „Server“ und „Port“ selber anpassen:

Brick - ID	<input type="text" value="gmk"/>
Server	<input type="text" value="WIFI-Extension"/>
Port	<input type="text" value="4223"/>

Wenn alles funktioniert hat, können sie nun die einzelnen Tasten genauso zuordnen wie beim LCD Touch Pendant – mit dem Unterschied dass Sie keinen Bildschirm zur Verfügung haben was die Navigation auf den Screens (die im Hintergrund genauso funktionieren) erschwert und gut durchdacht werden soll.

Beispiel: wenn JWC startet befindet man sich auf dem Startbildschirm des Handrades (das nicht sichtbar ist) wo es keine Bewegung der Achsen erlaubt ist. Hier soll man zuerst für die gewünschten Betriebsmodi die passenden Tasten zuordnen (Achsenbewegungsbildschirm, 2/4 Takt,...) und dann auf den entsprechenden Bildschirmen mit der Definition der Funktionen fortfahren.

## Beispiele in späteren Versionen geplant

### **Zuordnen einer Funktion zu einer Schaltfläche auf dem Handrad-Bildschirm**

See Beispiel 6.10

### **Ein- und Ausschalten einer LED auf dem Handrad-Bildschirm mit nur einer Taste**

Geplant

### **Steuern des Wertes eines DRO auf dem Handrad-Bildschirm mit dem Encoder (+ / -)**

Geplant

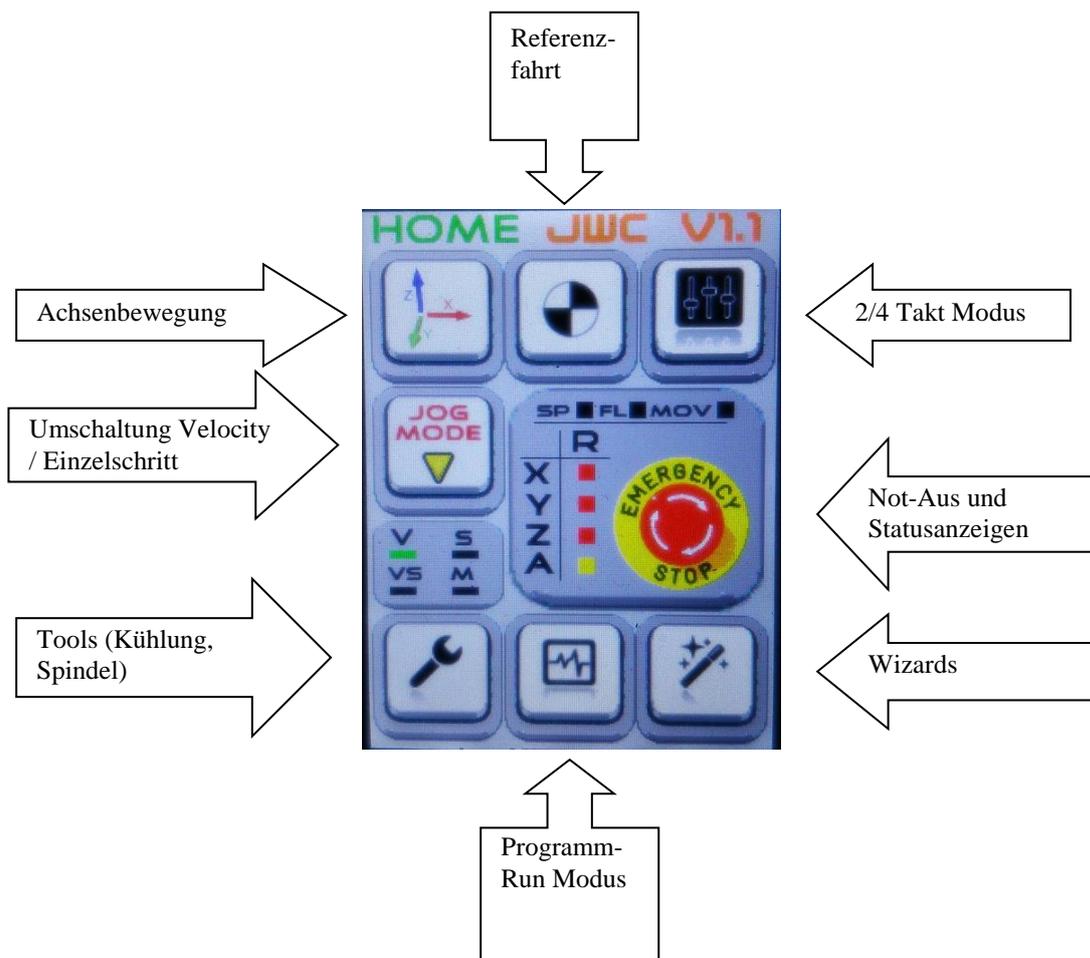
### **Anzeige der Werte aus dem Plugin im DRO auf dem Handrad-Bildschirm**

Geplant

# 7 Profilbeschreibung für Eding CNC und LCD Touch Pendant

JWC kommt mit einem bereits vollständig vorkonfigurierten Profil für das CNC Steuerungsprogramm Eding CNC um es zusammen mit dem LCD Touch Pendant Handrad verwenden zu können. Im Folgenden finden Sie die Beschreibung der Funktionen des Profils.

## 7.1 Startbildschirm

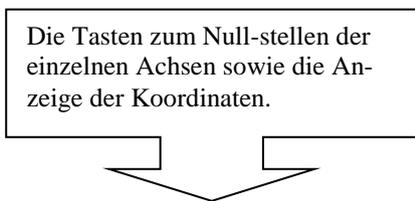


Dieser Bildschirm ist eine Startzentrale von wo jeder weitere Modus erreicht werden kann. Beim Antippen auf „Not-Aus“ (Emergency Stop) kommt man in ein Notstop-Bildschirm indem keine Bewegung der Achsen oder Sonstige Aktionen der CNC Steuerung möglich sind. Beim erneuten Antippen auf den Bildschirm kehrt man wieder zurück.

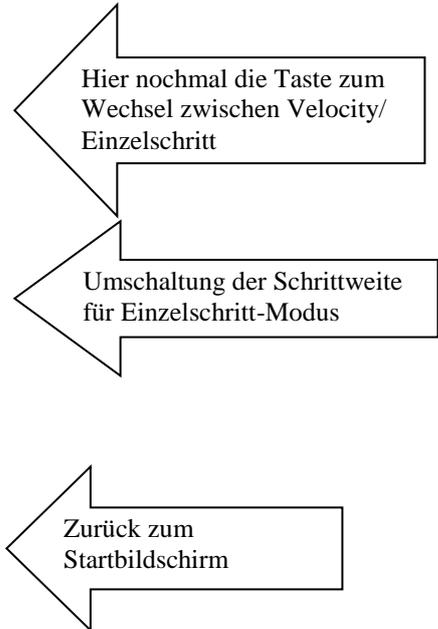
Die äußeren Hardware-Tasten des Handrades sind mit „Stop“ (Rot) und „Reset“ (Grün) belegt.

## 7.2 Achsenbewegungsbildschirm

Dies ist einer der wichtigsten Bildschirme auf dem man sich die meiste Zeit aufhalten wird.



Die Tasten zur Auswahl der Achse die beim Drehen des Handrades bewegt wird.



Auf diesem Bildschirm kann man die ausgewählte Achse durch das Drehen des Handrades ununterbrochen (Im Velocity Modus) oder schrittweise (Im Einzelschritt-Modus) verfahren. Die äusseren Hardware-Tasten des Handrades funktionieren als Schalter für 2-Takt Modus: beim Halten einer Taste und dem Drehen des Handrades im Uhrzeigersinn fängt die ausgewählte Achse in die entsprechende Richtung zuerst langsam und dann immer schneller zu fahren, beim Drehen gegen Uhrzeigersinn verringert sich die Geschwindigkeit bis die Bewegung stoppt. Beim Loslassen der Taste stoppt die Bewegung ebenfalls.

### 7.3 Referenz-Bildschirm



Von diesem Bildschirm aus kann man die Referenzfahrt für die einzelnen Achsen (oder auch für alle nacheinander) auslösen.

## 7.4 2/4 Takt Modus und Override-Bildschirm



Das ist der spezialisierte Bildschirm für 2/4 Takt Modus. Von diesem Bildschirm aus kann man aber auch Override-Einstellungen für die Spindeldrehzahl und Feed verändern oder zurücksetzen. In dem Fall wenn keine der Tasten des Handrades gehalten wird – verändert sich der jeweils ausgewählte Override-Wert beim Drehen des Handrades.

Wenn eine der Hardware-Tasten des Handrades gehalten wird verhält sich das Handrad wie im Achsenbewegungsbildschirm beschrieben (2 Takt Modus).

Wenn man den 4-Takt Modus aktiviert (Taste 2/4) – leuchtet die LED erst mal grün und es bewegt sich nichts. Wenn man nun das Handrad in eine der beiden Richtungen dreht, fängt die ausgewählte Achse in diese Richtung zu laufen – zuerst langsam und dann beim Weiterdrehen immer schneller, dabei muss keine Taste gehalten werden. Die Bewegung stoppt erst wenn man wiederholt die „2/4“ Taste antippt.

## 7.5 Tools-Bildschirm



Auf diesem Bildschirm sind nicht alle Tasten mit Funktionen belegt. Sie können Sprühnebelkühlung, Flutkühlung und Spindel ein – und ausschalten. Andere Tasten können je nach Bedarf selber belegt werden. Der Zustand wird durch die LED's signalisiert.

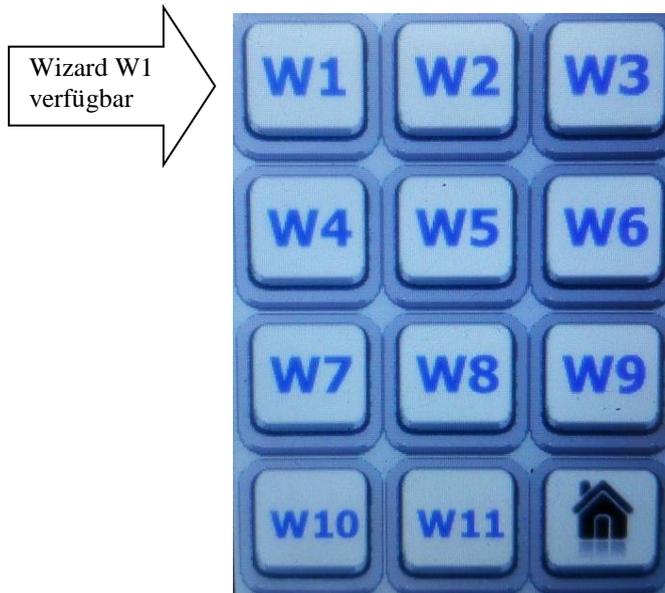
## 7.6 Programm-Run Bildschirm



Dieser Bildschirm ist für die Programmausführung gedacht. Hier können Sie genauso wie in dem „2/4 Takt“ Bildschirm beim Drehen des Handrades die Override-Werte anpassen oder zurücksetzen. Zusätzlich wird hier der geschätzte Verlauf des Programms (in %) dargestellt.

Man kann das Programm starten und pausieren (Taste START) oder abbrechen (STOP). Ob das Programm läuft sieht man an dem grünen LED auf der START-Taste.

## 7.7 Wizards-Bildschirm



Dieser Bildschirm ist dazu gedacht Tasten mit speziellen Wizards (Hilfsprogrammen) zu belegen die der Benutzer auch selber nach Bedarf entwickeln und einsetzen kann.

Als Beispiel hier ist die Taste „W1“ mit dem Wizard für G73 Bohren belegt. In diesem Wizard können Sie alle notwendigen Parameter des G73 Bohrzyklus einfach durch die Auswahl der Schrittweite und das Drehen des Handrades einstellen – und direkt bohren ohne zu dem PC zurückzukehren.

Die eingestellten Werte bleiben erhalten auch wenn der Wizard verlassen wird – sodass Sie zu einer anderen Position fahren können und beim Zurückkehren zu dem Wizard mit den gleichen Werten bohren können.

