



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

## ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

DEPARTMENT OF CONTROL AND INSTRUMENTATION

## DEMOKUFR R360

MOBILE CONTROL SYSTEM R360 TO THE SUITCASE

### BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

### AUTOR PRÁCE

AUTHOR

Ondřej Balga

### VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Radek Štohl, Ph.D.

BRNO 2016

# Bakalářská práce

bakalářský studijní obor **Automatizační a měřicí technika**

Ústav automatizace a měřicí techniky

**Student:** Ondřej Balga

**ID:** 158631

**Ročník:** 3

**Akademický rok:** 2015/16

**NÁZEV TÉMATU:**

**Demokufr R360**

## POKYNY PRO VYPRACOVÁNÍ:

1. Seznamte se a popište HW mobilního řídicího systému R360.
2. Vytvořte konfiguraci demonstrační úlohy do kufru.
3. Vytvořte demonstrační řídicí aplikaci s komunikací CANOpen.
4. Ověřte a vyhodnoťte své řešení.

## DOPORUČENÁ LITERATURA:

Dle vlastního literárního průzkumu a doporučení vedoucího práce.

**Termín zadání:** 8.2.2016

**Termín odevzdání:** 23.5.2016

**Vedoucí práce:** Ing. Radek Štohl, Ph.D.

**Konzultant bakalářské práce:**

**doc. Ing. Václav Jirsík, CSc., předseda oborové rady**

## UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

## **Abstrakt**

Bakalářská práce se věnuje návrhu demonstrační úlohy mobilního řídicího systému R360 do kufru. Mobilní systém R360 je zde doplněn o snímače teploty, polohy a otáček a krokový motor. Návrh má předvést možnosti mobilního systému R360.

## **Klíčová slova**

R360, PLC, systém pro mobilní přístroje, ifm electronic

## **Abstract**

This bachelor thesis deals with design of mobile control system R360 to the suitcase. Mobile system R360 is complemented by temperature sensors, position sensor, speed sensor and step motor. Proposal should show options of mobile system R360.

## **Keywords**

R360, PLC, system for mobile machine, ifm electronic

### **Bibliografická citace:**

BALGA, O. *Demokufr R360*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2016. 52s. Vedoucí bakalářské práce byl Ing. Radek Štohl, Ph.D.

## **Prohlášení**

„Prohlašuji, že svou bakalářskou práci na téma Demokufr R360 jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

V Brně dne: **23. května 2016**

.....  
podpis autora

## **Poděkování**

Děkuji vedoucímu bakalářské práce Ing. Radku Štohlovi, Ph.D. za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé bakalářské práce.

V Brně dne: **23.05.2016**

.....

podpis autora

# Obsah

|       |                                      |    |
|-------|--------------------------------------|----|
| 1     | Úvod .....                           | 11 |
| 2     | Mobilní řídicí systém R360 .....     | 12 |
| 2.1   | Kontroléry .....                     | 12 |
| 2.2   | I/O moduly .....                     | 12 |
| 2.3   | Komunikační moduly a displeje.....   | 12 |
| 2.4   | Kamerové systémy .....               | 12 |
| 2.5   | Senzory.....                         | 13 |
| 3     | Controller area network.....         | 14 |
| 3.1   | Topologie .....                      | 14 |
| 3.2   | Výměna dat v CAN.....                | 15 |
| 3.3   | Node-ID .....                        | 15 |
| 3.4   | Node Guarding Object .....           | 15 |
| 3.5   | Emergency Object.....                | 15 |
| 4     | Softwarové prostředky .....          | 16 |
| 4.1   | EPLAN.....                           | 16 |
| 4.2   | CoDeSys.....                         | 16 |
| 4.2.1 | IL – Instrumentation List .....      | 16 |
| 4.2.2 | LD – Ladder Diagram .....            | 17 |
| 4.2.3 | CFC – Continuous Function Chart..... | 18 |
| 4.2.4 | FBD – Function Block Diagram .....   | 19 |
| 4.2.5 | SFC – Sequential Function Chart..... | 19 |
| 4.2.6 | ST – Structured Text .....           | 19 |
| 5     | Hardwarová konfigurace .....         | 21 |
| 5.1   | BasicController (CR0403) .....       | 21 |
| 5.2   | BasicRelay (CR0421).....             | 21 |
| 5.3   | BasicDisplay (CR0451) .....          | 21 |
| 5.4   | Klávesnicový modul (CR1500).....     | 22 |
| 5.5   | CANfox (EC2112) .....                | 22 |
| 5.6   | Dialogový displej (CR1080) .....     | 22 |
| 5.7   | Ethernetová kamera (O2M113).....     | 22 |
| 5.8   | Snímač náklonu (CR2102).....         | 23 |
| 5.9   | Snímač náklonu (EC2082) .....        | 23 |
| 5.10  | Síťový zdroj (DN4013).....           | 23 |

|       |                                                 |    |
|-------|-------------------------------------------------|----|
| 5.11  | Digitální ukazatel (DX2002).....                | 23 |
| 5.12  | Rotační snímač (RO6343).....                    | 24 |
| 5.13  | Snímač teploty (TS2239) .....                   | 24 |
| 5.14  | Snímač polohy (IGC213) .....                    | 24 |
| 5.15  | HY-DIV168N-3.5A .....                           | 24 |
| 5.16  | Krokový motor.....                              | 25 |
| 5.17  | Hliníkový Kufr s teleskopickou rukojetí.....    | 25 |
| 6     | Praktická realizace demonstrační úlohy .....    | 26 |
| 6.1   | Napájení .....                                  | 26 |
| 6.2   | Vstupy a výstupy kontroléru .....               | 27 |
| 6.3   | Sběrnice CAN .....                              | 28 |
| 6.4   | Ethernet.....                                   | 28 |
| 6.5   | Kufr .....                                      | 29 |
| 7     | Programování demonstrační úlohy.....            | 30 |
| 7.1   | BasicController CR0403 .....                    | 30 |
| 7.1.1 | PLC konfigurace .....                           | 30 |
| 7.1.2 | Globální proměnné.....                          | 32 |
| 7.1.3 | Knihovny.....                                   | 34 |
| 7.1.4 | Programování kontroléru .....                   | 34 |
| 7.2   | Dialogový displej CR1080.....                   | 38 |
| 7.2.1 | PLC konfigurace .....                           | 39 |
| 7.2.2 | Knihovny.....                                   | 40 |
| 7.2.3 | Globální proměnné.....                          | 41 |
| 7.2.4 | Programování displeje.....                      | 42 |
| 7.2.5 | Vizualizace.....                                | 43 |
| 7.3   | BasicDisplay CR0451 .....                       | 45 |
| 7.3.1 | PLC konfigurace .....                           | 46 |
| 7.3.2 | Knihovny, Globální proměnné a programování..... | 47 |
| 7.3.3 | Vizualizace.....                                | 47 |
| 8     | Závěr.....                                      | 48 |

# Seznam obrázků

|                                                   |    |
|---------------------------------------------------|----|
| Obr. 1 – Lineární topologie [3] .....             | 14 |
| Obr. 2 – EPLAN [5] .....                          | 16 |
| Obr. 3 – CoDeSys [7] .....                        | 16 |
| Obr. 4 - Instrumentation List .....               | 17 |
| Obr. 5 - Ladder Diagram .....                     | 18 |
| Obr. 6 - Continuous Function Chart [8] .....      | 18 |
| Obr. 7 - Function Block Diagram [9] .....         | 19 |
| Obr. 8 - Sequential Function Chart [10] .....     | 19 |
| Obr. 9 - Structured Text .....                    | 20 |
| Obr. 10 – BasicController [12] .....              | 21 |
| Obr. 11 - BasicRelay [14] .....                   | 21 |
| Obr. 12 – BasicDisplay [16] .....                 | 21 |
| Obr. 13 – Klávesnicový modul [18] .....           | 22 |
| Obr. 14 - CANfox [20] .....                       | 22 |
| Obr. 15 – Dialogový displej [22] .....            | 22 |
| Obr. 16 – Ethernetová kamera [24] .....           | 22 |
| Obr. 17 - Snímač náklonu I [26] .....             | 23 |
| Obr. 18 - Snímač náklonu II [28] .....            | 23 |
| Obr. 19 - Síťový zdroj [30] .....                 | 23 |
| Obr. 20 - Digitální ukazatel [32] .....           | 23 |
| Obr. 21 - Rotační snímač [34] .....               | 24 |
| Obr. 22 - Snímač teploty [35] .....               | 24 |
| Obr. 23 - Snímač polohy [37] .....                | 24 |
| Obr. 24 - HY-DIV168N-3.5A [38] .....              | 24 |
| Obr. 25 - Krokový motor [39] .....                | 25 |
| Obr. 26 – Kufr [40] .....                         | 25 |
| Obr. 27 - Napájení demonstrační úlohy .....       | 26 |
| Obr. 28 - Zapojení pojistek .....                 | 27 |
| Obr. 29 - Ukázka zapojení driveru s motorem ..... | 27 |
| Obr. 30 - Zapojení sběrnice CAN .....             | 28 |
| Obr. 31 - Ethernet – zapojení .....               | 28 |
| Obr. 32 - CoDeSys - CR0403 target settings .....  | 30 |
| Obr. 33 - CoDeSys - CR0403 - Resources .....      | 31 |

|                                                                     |    |
|---------------------------------------------------------------------|----|
| Obr. 34 - CoDeSys CR0403 - PLC Configuration I .....                | 31 |
| Obr. 35 - CoDeSys - PLC Configuration II .....                      | 32 |
| Obr. 36- CoDeSys CR0403 - Global variables I.....                   | 32 |
| Obr. 37 - CoDeSys CR0403 - Global variables II.....                 | 33 |
| Obr. 38 - CoDeSys CR0403 - Global variables III .....               | 33 |
| Obr. 39 - CoDeSys CR0403 - Global variables IV .....                | 33 |
| Obr. 40 - CodeSys - CR0403 – Knihovny .....                         | 34 |
| Obr. 41 - CoDeSys - CR0403 - INPUT a OUTPUT.....                    | 34 |
| Obr. 42 - CoDeSys - CR0403 Převod analogového signálu .....         | 35 |
| Obr. 43 - CoDeSys - CR0403 - Tlačítka z CR1080/CR0451/ CR1500 ..... | 35 |
| Obr. 44 - CR0403 – CoDeSys - Blok motoru .....                      | 36 |
| Obr. 45 –CoDeSys - CR0403 – Ovládání motoru.....                    | 37 |
| Obr. 46 – CoDeSys - CR0403 – Přenos dat do displejů .....           | 37 |
| Obr. 47 - CoDeSys - CR0403 – Hlavní program .....                   | 38 |
| Obr. 48 - CoDeSys - CR1080 - Target Settings.....                   | 38 |
| Obr. 49 - CoDeSys - CR1080 - PLC konfigurace.....                   | 39 |
| Obr. 50 - CoDeSys - CR1080 - Parameter Manager.....                 | 39 |
| Obr. 51 - CoDeSys - CR1080 - Mapování.....                          | 40 |
| Obr. 52 - CoDeSys CR1080 - Knihovny .....                           | 41 |
| Obr. 53 - CoDeSys - CR1080 - Globální proměnné .....                | 42 |
| Obr. 54 - CoDeSys - CR1080 - Příprava dat pro vizualizaci .....     | 42 |
| Obr. 55 - CoDeSys - CR1080 - Příprava dat pro vizualizaci .....     | 43 |
| Obr. 56 - CoDeSys - CR1080 – Masterlayout .....                     | 44 |
| Obr. 57- CoDeSys - CR1080 – Masterlayout tlačítko .....             | 44 |
| Obr. 58 - CoDeSys - CR1080 – Ukázka vizualizace motoru.....         | 45 |
| Obr. 59 – CoDeSys – CR0451 - Target .....                           | 46 |
| Obr. 60 - CoDeSys - CR0451 - PLC konfigurace.....                   | 46 |
| Obr. 61 - CoDeSys - CR0451 - Masterlayout.....                      | 47 |
| Obr. 62 – CoDeSys – CR0451 – Stránky vizualizace.....               | 47 |

# 1 ÚVOD

Tématem této bakalářské práce je Demokufr R360. Jedná se tedy o demonstrační úlohu, kde je hlavním prvkem mobilní řídicí systém R360, zakomponovanou do kufru. Cílem práce je tedy předvést možnosti systému, jednoduchost zapojení a rozšiřitelnost systému.

Mobilní řídicí systém R360, vyráběný firmou IFM electronic, je modulární systém s jednoduchou montáží a obsluhou. Skládá se z kontrolérů, I/O modulů, komunikačních modulů, displejů, kamerových systémů a senzorů.

Firma IFM electronic je světovým lídrem v oblasti technologie CAN-Bus. Společnost vznikla v roce 1991 a zabývá se výrobou a prodejem komponentů pro průmyslovou automatizaci. Sortiment zahrnuje polohovou a procesní sensoriku, senzory pro kontrolu pohybu a rozpoznání objektů, bezpečnostní techniku, systémy pro prediktivní údržbu a sběrnice, identifikační a řídicí systémy.[1]

Práce je rozdělena do 6 kapitol, které jsou členěny následovně. V kapitole 2 najdete popis jednotlivých částí mobilního řídicího systému R360.

Kapitola 3 popisuje sběrnici Controller Area Network neboli CAN. Jsou zde popsány topologie, princip výměny dat na sběrnici a její základní vlastnosti.

V následující kapitole 4 jsou popsány softwarové prostředky použité při návrhu a programování demonstrační úlohy. Nejprve je popsán systém EPLAN, ve kterém bylo zakresleno zapojení demonstrační úlohy. Za ním následuje software CoDeSys, který byl použit pro programování celé demonstrační úlohy.

V kapitole 5 jsou popsány hardwarové prostředky demonstrační úlohy. Jsou zde detailně popsány všechny komponenty použité v demonstrační úloze doplněné o datasheet.

Poslední kapitola 6 popisuje programování jednotlivých zařízení. Do této kapitoly jsou zahrnuté detailní popisy nastavení a funkcí programu celé úlohy. U displejů je zde vysvětlen a ukázán princip vizualizace.

## **2 MOBILNÍ ŘÍDICÍ SYSTÉM R360**

V této kapitole jsou popsány jednotlivé části mobilního řídicího systému R360, který je hlavním prvkem této bakalářské práce. Podkapitoly jsou rozděleny podle jednotlivých typů komponentů, kde je daná skupina vždy popsána.

### **2.1 Kontroléry**

Kontroléry umožňují flexibilní konfiguraci vstupních a výstupních funkcí. Každý kontrolér je vybaven CAN rozhraními. Volně programovatelná CAN brána může být implementována pro připojení do různých sítí. Uživatel může jednoduše integrovat všechny funkce pomocí programu CoDeSyS programováním podle IEC 61131-3. Firma IFM electronic nabízí různé druhy kontrolérů od kontrolérů s vysokým stupněm ochrany, přes bezpečnostní řídicí jednotky kategorie 3, až k přístrojům pro přímé použití v kokpitu.

### **2.2 I/O moduly**

I/O moduly jsou decentralizované moduly používané ve sběrnicových systémech CANopen. Slouží ke snížení počtu přívodních kabelů do míst, kde se nacházejí signály. Univerzální použití je docíleno možností flexibilní konfigurace vstupů a výstupů. Pomáhají také snížení náklady pro udržování skladů. Je možné využít moduly s různým typem krytí a konektorové propojení M12.

### **2.3 Komunikační moduly a displeje**

Firma IFM electronic nabízí ve svém sortimentu moduly od graficky schopného dialogového monitoru PDM až k textovému displeji. Moduly umožňují uživateli volné programování podle IEC 61131. Způsob montáže je montážní výstavba nebo vestavení. Využívají se jako uživatelské rozhraní pro servis a obsluhu strojů. Je možné využít moduly s CANopen nebo s ethernetem.

### **2.4 Kamerové systémy**

Kamerové systémy mobilního řídicího systému R360 se používá k monitorování pracovního pole. Firma IFM electronic nabízí dva typy kamer O2M a O3M. V případě O2M se jedná o analogovou kameru a v případě O3M o mobilní 3D senzor. Robustní korpus je vhodný pro nasazení v extrémních podmínkách.

## **2.5      Senzory**

Řídicí systém pro mobilní stroje nabízí snímače náklonu, induktivní senzory a tlakové senzory. Senzory je možná zapojit pomocí CAN-Bus nebo analogových a digitálních výstupů. Senzory jsou z provozního hlediska bezpečné i za extrémních podmínek.

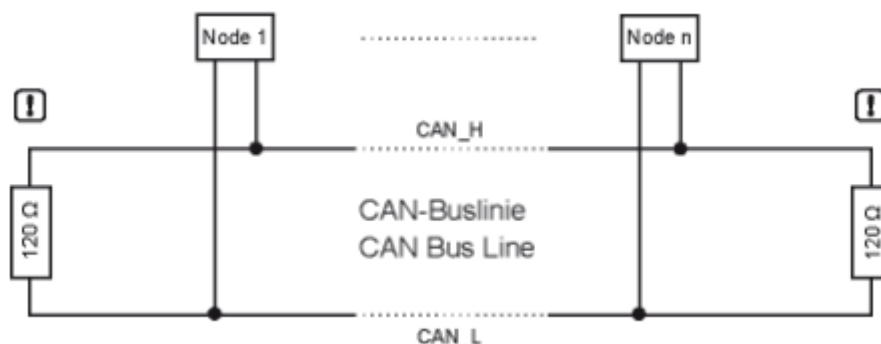
## 3 CONTROLLER AREA NETWORK

Controller Area Network neboli CAN je sériový komunikační protokol vyvinutý v polovině osmdesátých let minulého století německou firmou Robert Bosch GmbH Stuttgart. Byl původně navržen pro použití v automobilovém průmyslu a jeho vlastnosti byly definovány v CAN Specification 1.2. Nejen díky velkému vlivu německých automobilek se brzy začal prosazovat a rozšiřovat i do dalších odvětví průmyslové výroby. Proto v roce 1991 firma Bosch zveřejnila novou verzi protokolu CAN, tzv. CAN Specification 2.0, která dva roky poté byla vydána jako norma ANSI / ISO 11898. Tato nová verze se skládá ze dvou částí: CAN 2.0A odpovídající původní specifikaci 1.2 používající formát zprávy s 11 bitovým identifikátorem a CAN 2.0B používající buď standardní (tzv. Standard) nebo rozšířený formát zprávy (tzv. Extended) s 11 bitovým, resp. 29 bitovým identifikátorem. [2]

Důvodů k masivnímu rozšíření sběrnice CAN je hned několik. Sběrnice CAN má nízkou cenu, velkou spolehlivost a vysokou přenosovou rychlost (až 1Mbit/s). Je možné propojování až na vzdálenosti několika kilometrů. Sběrnice využívá nízký počet vodičů a nabízí snadnou rozšiřitelnost. Díky těmto vlastnostem se CAN rozšířil do těchto odvětví: robotika, zemědělská technika, textilní stroje, zabezpečovací zařízení.

### 3.1 Topologie

Síť CAN standardně využívá lineární topologie. Kromě toho je možné použít topologii kruh nebo hvězda. V porovnání s lineární topologií mají obě topologie jednu nevýhodu. U topologie kruh jsou všechna zařízení připojená v sérii, takže pokud jedno zařízení selže, tak selže celá sběrnice. Topologie hvězda jsou většinou řízeny centrálním procesorem, přes který musí všechny informace projít. Pokud tento procesor selže, tak nemůžou být přenášeny žádné informace. Ovšem pokud selže jiné zařízení, tak sběrnice pokračuje ve funkci. Výhodou lineární topologie je paralelní zapojení zařízení na centrální kabely. Pouze pokud centrální kabely selžou, tak sběrnice přestane fungovat.



Obr. 1 – Lineární topologie [3]

## **3.2 Výměna dat v CAN**

CAN data jsou vyměňována přes CAN protokol na linkové vrstvě sedmi vrstvého modelu ISO/OSI. Každé zúčastněné zařízení ve sběrnici může vysílat zprávy. Data jsou posílány do sběrnice bez odesílatele a bez adresy příjemce. Jsou pouze označeny identifikátorem. Každé zařízení poté přijme vysílaná data a zkontroluje podle identifikátoru, zda vysílaná data jsou pro něj.

## **3.3 Node-ID**

Zařízení v CAN sběrnici jsou rozeznávány pomocí identifikátorů neboli Node-ID. Zařízení ve stejné síti musí mít svůj jedinečný identifikátor. Rozsah Node-ID je 1 až 127.

## **3.4 Node Guarding Object**

Správce sítě periodicky monitoruje přítomnost a stav Slave zařízení zasíláním datových požadavků, dotazů podřízeným zařízením s cílem zjistit, zda jsou aktivní. Podřízené zařízení odpovídá zprávou, která obsahuje informaci o jeho stavu a doplňkový (toggle) bit měnící svoji hodnotu při každém dotazu. Doplňkový bit umožňuje zjistit, zda došla odpověď je odpovědí na aktuální dotaz. Node Guarding Object z odvráceného pohledu může podřízené zařízení provádět kontrolní činnost Life-guarding řídicího zařízení s využitím pravidelnosti dotazů Node-guarding. Jestliže podřízené zařízení neobdrží tento dotaz do plynutí určené doby, oznámí tuto okolnost nadřazenému aplikačnímu programu, který na ni může zareagovat.

## **3.5 Emergency Object**

Zpráva je zaslána v případě vzniku závažné chyby v zařízení. Díky nízké hodnotě COB-ID má zpráva vysokou prioritu. Informace o chybách je uložena v Error registru (0x1001). Zpráva obsahuje rezervovanou oblast pro rozšířenou specifikace chyby.

## 4 SOFTWAREVÉ PROSTŘEDKY

Demonstrační úlohu jsem zakreslil pomocí systému EPLAN ke kterému vytváří firma IFM electronic své knihovny. Programování úlohy bylo provedeno pomocí softwaru CoDeSys, který je dodáván přímo při zakoupení komponentů mobilního řídicího systému R360.

### 4.1 EPLAN



Obr. 2 – EPLAN [5]

EPLAN se zabývá konzultační činností a vývojem aplikačního softwaru pro urychlení projekčních procesů a snížení nákladů v oblasti inženýringu. EPLAN pomáhá společností s optimalizací jejich procesů, vyvíjí software pro elektrotechniku, mechatroniku, fluidní technologie a technologie měření a regulace. S cílem urychlit proces vývoje produktu, a tím snížit náklady projektu, zajišťuje rozhraní systémů CAD, PDM, PLM a ERP. [4]

### 4.2 CoDeSys



Obr. 3 – CoDeSys [7]

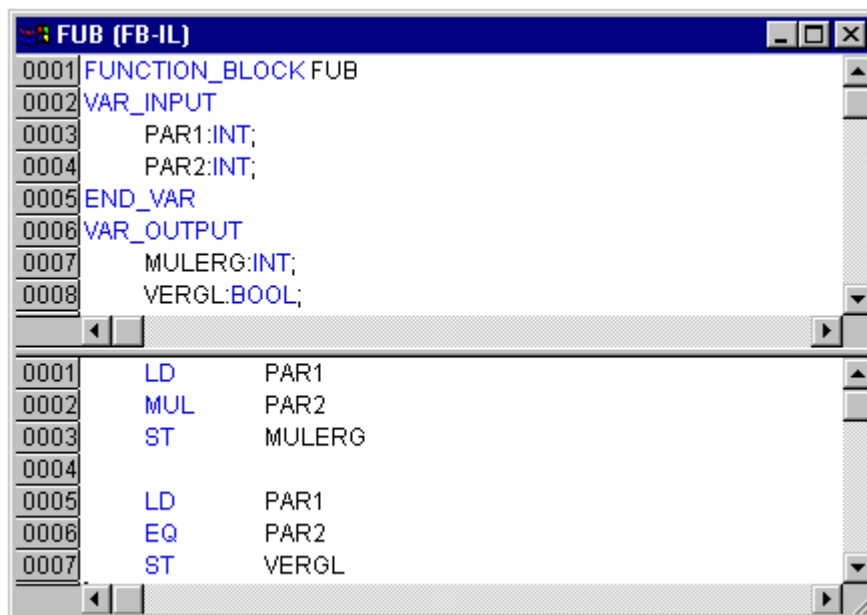
Pod označením standardu IEC 61131-3 se skrývá definice programovacích režimů (jazyků) pro programovatelné automaty (PLC). Tento soubor je v praxi spíše známý pod označením CoDeSys, což je zkratka pro Controlled Development System. To je univerzální vývojové prostředí pro aplikační programy řídicích systémů PLC vytvořené společností 3S-Smart Software Solutions GmbH právě podle standardu IEC 61131-3, bez ohledu na použitý hardware. Díky němu je tak možno vytvářet jednotné datové a programové struktury a způsoby ovládání nezávisle na výrobci a typu PLC, protože „napasování“ na daný hardware probíhá až poté při překladu. [6]

#### 4.2.1 IL – Instrumentation List

Zápis programu označovaný jako „Instruction List“ je nejzákladnější možný popis programu a více méně se podobá programování mikrokontrolérů v assembleru (ASM). Program se zde skládá z posloupnosti jednoduchých operací tvořených základními instrukcemi (např. MOV pro přesun dat mezi registry, ADD pro sečtení dvou hodnot apod.) a operandů reprezentujícími jednotlivé registry a paměťová místa.

Tento způsob programování se vyznačuje dokonalým přehledem a nadvládou programátora nad konstrukcí a strukturou programu. Umožňuje tak co nejvíce optimalizovat program jak na jeho velikost, i rychlost vykonávání, protože lze jej vytvořit bez všech zbytečných přesunů dat, odskoků a volání. [6]

Výhody toho jazyka jsou přesná definice chování programu a paměťové i na rychlost zpracování úsporný program. Nevýhodami jsou nutnost znát příkazy, mnoho psaní a horší přehlednost programu. [6]

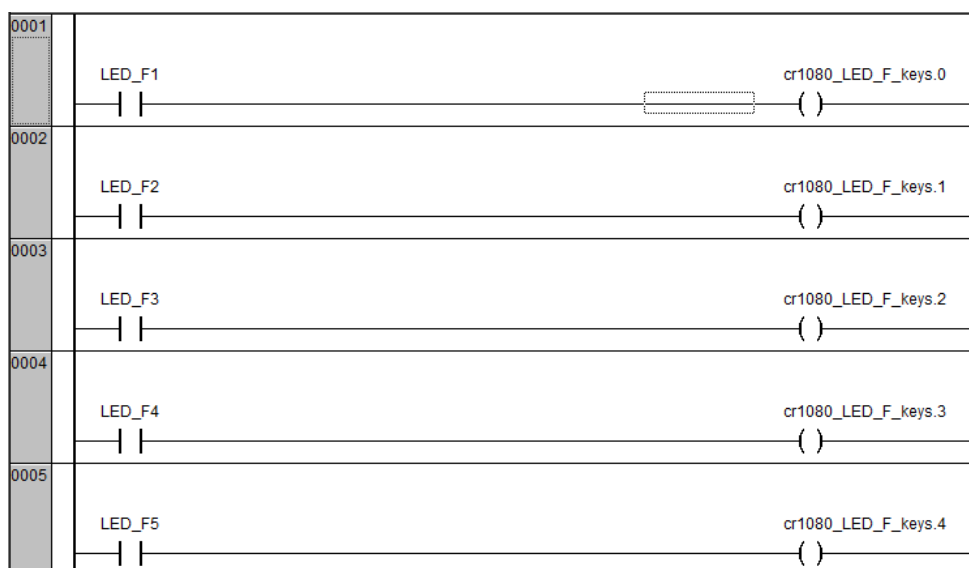


Obr. 4 - Instrumentation List

## 4.2.2 LD – Ladder Diagram

U většiny PLC, i u těch, které nepoužívají CoDeSys, je toto základní programovací prostředek a spolu s IL režimem také nejrozšířenější. Prakticky jde o grafické znázornění výše uvedeného režimu IL, kde bitové operace jsou znázorněny jako posloupnost vzájemně propojených ovládacích a výstupních kontaktů relé a složitější příkazy a funkce typu čítač, časovač, přesun dat či aritmetické operace jsou vkládány v podobě „obdélníků“ s vývody. [6]

Hlavní výhodou tohoto zápisu je však jasně definovaná posloupnost programu a tedy i velká přehlednost, která však někdy bývá na úkor kompaktnosti. U větších a složitějších programů či při masivnějším nasazení funkcí typu přesunu dat (např. funkce MOV apod.) pak může délka zápisu dost rychle narůstat až se zápis začne stávat nepřehledným. Vhodné je tak častější členění do podprogramů a masivní nasazení komentářů. Na druhou stranu u menších programů či menších částí velkého programu (podprogramů) s velkým zastoupením logických operací je LD režim výhodný. [6]



Obr. 5 - Ladder Diagram

### 4.2.3 CFC – Continuous Function Chart

Zápis programu v programovacím režimu CFC je naopak ideální pro toho, kdo vyznává skládání programu z jednotlivých „krabiček“ vzájemně propojených přes vývody. Tedy něco pro elektro konstruktéry, kteří jsou zvyklí vytvářet klasická schémata zapojení součástek. Zde je právě prostě jen plocha, na kterou se postupně vkládají a propojují jednotlivé funkce v podobě „součástek“ s daným i vlastnostmi. [6]



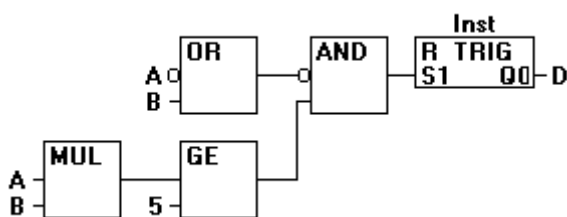
Obr. 6 - Continuous Function Chart [8]

V prostředí CoDeSys prostě máme k dispozici volnou plochu, kam se volně vkládají

jednotlivé bloky („krabičky“) reprezentující

jednotlivé funkce ať již logické, aritmetické, časování, čítání apod. Jednotlivé bloky se pak propojují čarou reprezentující vodiče po nichž se šíří signály. Tento zápis je z mého pohledu ideální pro části programu (podprogramy) zpracovávající současně analogové i logické signály. Při správném rozvržení plochy může být menší program velmi dobře přehledný zvláště z pohledu sledování „toku“ signálu od vstupu k výstupu. To je výhodné pro přehledné naprogramování zpětnovazebních regulačních algoritmů. Na druhou stranu při složitých „zapojení“, kdy se plocha již dost zaplní, se výhoda přehlednosti ztrácí a může přecházet i v chaos. Tento zápis dále není vhodný pro intenzivnější práci s daty v paměti (databáze), ASCII znaky nebo pro realizaci datové komunikace. [6]

## 4.2.4 FBD – Function Block Diagram

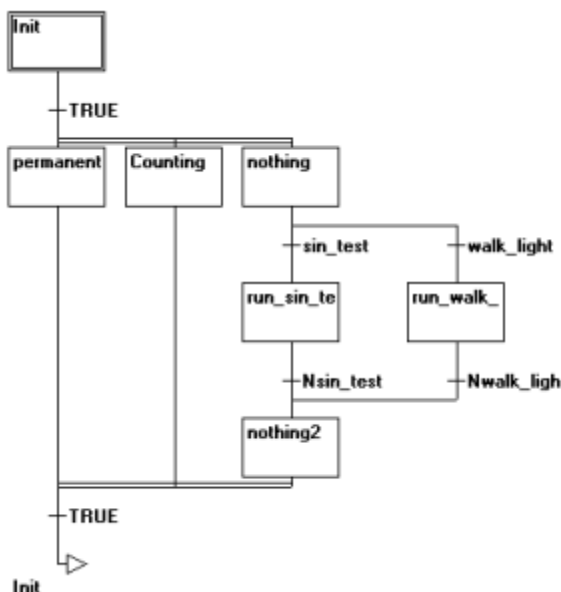


Obr. 7 - Function Block Diagram [9]

režim CFC. [6]

Stejně jako v CFC se i v FBD posloupnost programu vyjadřuje soustavou za sebou propojených bloků, které realizují různé funkce, a však zde již není volná programovací plocha, ale vše je organizováno do řádků (linií) podobně jako v LD režimu. Prakticky se dá říct, že FBD režim v prostředí CoDeSys je v podstatě LD režim, kde se u logických operací místo sériově-paralelního zapojování symbolů kontaktů relé využívá klasických značek hradel AND, OR apod. [6]

## 4.2.5 SFC – Sequential Function Chart



Obr. 8 - Sequential Function Chart [10]

Programovací režim SFC je graficky orientovaný zápis pro snadné definování chování programu, tedy jeho běh a reakce na různé rozhodovací situace. Prakticky umožňuje definovat chování programu prostřednictvím všeobecně známého vývojového schéma, tedy větvení programu na základě splnění rozhodovacích podmínek. [6]

Tento režim je ideální pro definování posloupnosti volání jednotlivých částí programu (podprogramů) napsaných některým z

výše uvedených programovacích režimů na základě hodnot stavových

proměnných. Také se s ním dobře realizuje sekvenční logika. [6]

## 4.2.6 ST – Structured Text

Programovací režim ST je ideální pro klasické programátory mikroprocesorů, protože jeho zápis je tvořený posloupností symbolických instrukcí, kde jedna instrukce

reprezentuje v „IL“ zápisu celou posloupnost základních instrukcí. Jde tedy o programovací jazyk vyšší úrovně, podobně jako například Pascal či C. Svoji strukturou (syntaxí) se také Pascalu velice podobá. [6]

Tento zápis je ideální pro práci s daty, řetězci a databázemi a pro naprogramování složitých výpočetních algoritmů. Je tak vhodný pro náročné zpracování analogových signálů, jako například pro realizaci digitálních filtrů, linearizaci signálů a složité řídicí algoritmy nebo pro práci s textovými řetězci. [6]

Výhodou je snadné a rychlé naprogramování i složitých aritmetických vzorců a relativně snadná manipulace a práce s bloky dat a databázemi (snadná realizace vícenásobných podmínkových smyček s indexací paměťových míst). Nevýhodou je horší přehlednost logických operací a „toků“ signálu strukturou programu. Také je nutné znát příkazy a syntaxi zápisu. [6]

```
0001 IF Enable AND M_Enable THEN
0002     M_Enable:= FALSE;
0003 ELSIF Enable THEN
0004     M_Enable:= TRUE;
0005 END_IF
0006 IF DIR AND M_DIR THEN
0007     M_DIR:= FALSE;
0008 ELSIF DIR THEN
0009     M_DIR:= TRUE;
0010 END_IF
0011 IF Speed_UP AND Speed_Counter.QU = FALSE THEN
0012     Speed_Plus := TRUE;
0013 ELSE
0014     Speed_Plus := FALSE;
0015 END_IF
0016 IF Speed_DOWN AND Speed_Counter.QD = FALSE THEN
0017     Speed_Minus := TRUE;
0018 ELSE
0019     Speed_Minus := FALSE;
0020 END_IF
0021
0022 IF M_Enable = FALSE THEN
0023     Speed_Counter(CU:=Speed_Plus,CD:= Speed_Minus,RESET:=M_Enable,PV:=100,CV=>M_Speed);
0024 END_IF
0025
```

Obr. 9 - Structured Text

## 5 HARDWAROVÁ KONFIGURACE

Jak už bylo zmíněno, cílem práce je sestavit demokufr R360. Na začátku je tedy dobré zmínit hardwarové prvky demonstrační úlohy. V této kapitole je popsán každý prvek demonstrační úlohy, který bude použit v její realizaci. Ke každému prvku jsou zde uvedeny hlavní specifikace doplněné o datasheet.

### 5.1 BasicController (CR0403)



Obr. 10 – BasicController [12]

Kontrolér mobilního řídicího systému R360, který má 12 vstupů, 12 výstupů a 2 CAN rozhraní. Vstupy je možné nastavit jako digitální, analogové, frekvenční a pro měření odporu. Výstupy mohou být nastaveny jako analogové, digitální nebo PWM. Kontrolér se napájí stejnosměrným napětím 8-32 V. [11]

### 5.2 BasicRelay (CR0421)



Obr. 11 - BasicRelay [14]

BasicRelay má 2 napájecí sběrnice a 6 rozvaděčů potenciálu. Je možné do něj zapojit 6 relé Kfz a 10 pojistek Kfz. [13]

### 5.3 BasicDisplay (CR0451)



Obr. 12 – BasicDisplay [16]

BasicDisplay je 2,8 palcový barevný displej s 5 volně programovatelnými podsvícenými tlačítky. Pro kurzorovou funkci má kolébkový spínač. Ke komunikaci využívá CAN rozhraní. Displej se napájí stejnosměrným napětím 8-32 V. [15]

## 5.4 Klávesnicový modul (CR1500)



Obr. 13 – Klávesnicový modul [18]

Klávesnicový modul komunikující přes rozhraní CAN jako slave zařízení. Modul má 12 tlačítek s mody toggle a push a čtyři tlačítka se šipkou. Modul se napájí stejnosměrným napětím 10-32 V. [17]

## 5.5 CANfox (EC2112)



Obr. 14 - CANfox [20]

Jedná se o převodník z rozhraní CAN na USB. Slouží k programování a diagnostiku CAN systémů. [19]

## 5.6 Dialogový displej (CR1080)



Obr. 15 – Dialogový displej [22]

s tlakovou funkcí [21]

Procesní a dialogový přístroj se sedmi palcový a barevným displejem. Je možné připojit digitální nebo analogový vstup. Jeden digitální výstup. Displej má 4 rozhraní CAN, jeden ethernet a USB. Ovládání displeje je možné pomocí 9 programovatelných tlačítek a programovatelného rotačního snímače

## 5.7 Ethernetová kamera (O2M113)



Obr. 16 – Ethernetová kamera [24]

Analogová kamera používaná u kamerových systémů pro mobilní pracovní stroje. Má čelní vyhřívané sklo. Ke komunikaci používá ethernet. Kamera se napájí stejnosměrným napětím 8-32 V. [23]

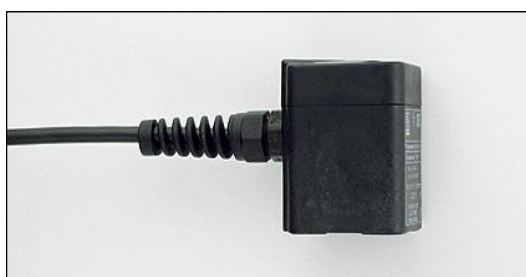
## 5.8 Snímač náklonu (CR2102)



Obr. 17 - Snímač náklonu I [26]

Snímač náklonu měří náklon ve dvou osách v rozsahu  $\pm 45$  stupňů. Přesnost snímače je 0,5 stupňů. Ke komunikaci využívá rozhraní CAN. Snímač se napájí stejnosměrným napětím 10-30 V. [25]

## 5.9 Snímač náklonu (EC2082)



Obr. 18 - Snímač náklonu II [28]

Snímač náklonu s jednoosým měření v rozsahu  $\pm 90$  stupňů. Přesnost snímače je 0,1 stupňů. Analogový výstup snímače je 4-20 mA. Snímač se napájí stejnosměrným napětím 20-30 V. [27]

## 5.10 Síťový zdroj (DN4013)



Obr. 19 - Síťový zdroj [30]

Spínaný stejnosměrný zdroj 24 V napájený střídavým napětím 230 V s výstupním výkonem 240 W. Využívá se pro regulované proudové napájení pro senzory, vyhodnocovací techniku a PLC. Má v sobě

zabudovanou ochranu proti zkratu a je odolný proti přetížení. Účinnost zdroje je 90%. [29]

## 5.11 Digitální ukazatel (DX2002)



Obr. 20 - Digitální ukazatel [32]

Přístroj slouží k univerzálnímu vyhodnocení a signalizaci pro všechny fyzikální veličiny, které se dají odvodit z impulsních sledů.

Napájí se stejnosměrným napětím 24 V. [31]

## 5.12 Rotační snímač (RO6343)



Obr. 21 - Rotační snímač [34]

Snímač určený pro měření otáček s jednostranně dutou hřídelí. Hřídel je vyrobena z nerezové oceli o průměru 12mm. Rozlišení snímače je 360. Má HTL-výstup 50mA, který je odolný proti zkratu po dobu kratší, než 1 min. Maximální počet měřitelných otáček za minutu je 12000. [33]

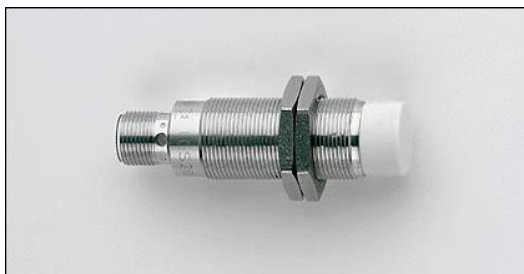
## 5.13 Snímač teploty (TS2239)



Obr. 22 - Snímač teploty [35]

Teplotní senzor určený pro připojení na vyhodnocovací jednotku. Jedná se o kabelové provedení s pozlacenými kontakty. Jako měřicí element je použit 1x Pt 100. Měřicí rozsah snímače je od -40 do 90 stupňů Celsia.

## 5.14 Snímač polohy (IGC213)



Obr. 23 - Snímač polohy [37]

Induktivní senzor se spínací vzdáleností 12mm. Provoz je možný ve 2 zapojeních: dvou-vodičově a tří-vodičově. Napájí se stejnosměrným napětím 10-30V. Výstup funguje jako spínač s proudovou zatížitelností 100mA. [36]

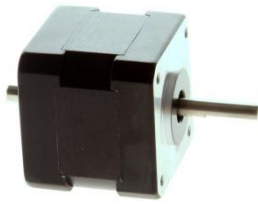
## 5.15 HY-DIV168N-3.5A



Obr. 24 - HY-DIV168N-3.5A [38]

Driver ovládaný pomocí vstupů Enable, Dir a PUL. Výstupy A a B slouží pro připojení bipolárního krokového motoru. Driver se napájí stejnosměrným napětím 24V.

## 5.16 Krokový motor



Krokový motor s osou na obě strany s přírubou Nema14. Definovaný proud má 1A a moment 0,14 Nm. Průměr osy je 5mm.

Obr. 25 - Krokový motor [39]

## 5.17 Hliníkový Kufr s teleskopickou rukojetí



Jedná se o pevný hliníkový kufr pro bezpečnou přepravu nářadí, součástek, elektroniky. Jeho rozměry jsou 566 na šířku, 477 na délku a 355 na výšku. Má uzamykatelné kovové panty a vyztužené kovové rohy. Čelní rukojeť a boční kovové rukojeti slouží k pohodlnému přenášení těžkých předmětů.

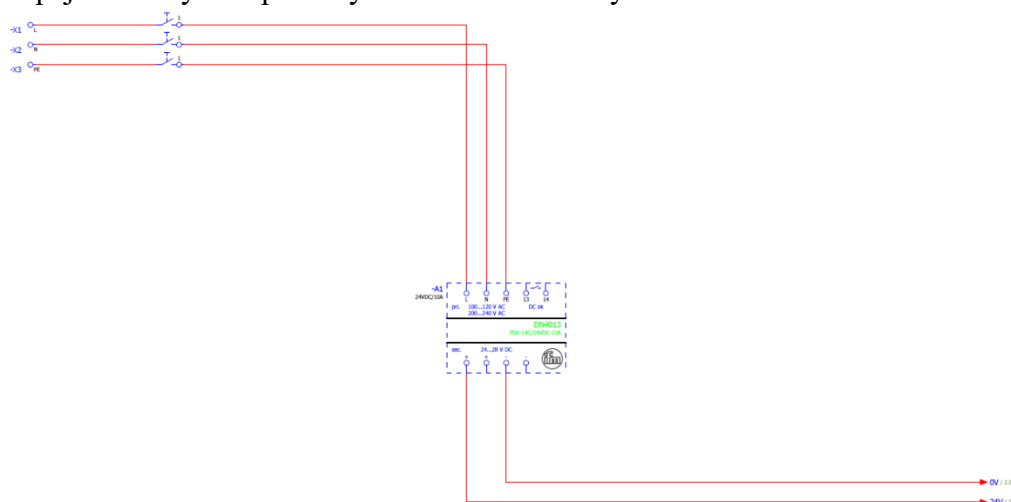
Obr. 26 – Kufr  
[40]

## 6 PRAKTICKÁ REALIZACE DEMONSTRAČNÍ ÚLOHY

Tato kapitola bude rozdělena do několika částí: napájení, vstupy a výstupy kontroléru, sběrnice CAN a ethernet. Zapojení demonstrační úlohy jsem provedl z komponentů z předchozí kapitoly. Zapojení demonstrační úlohy jsem zakreslil pomocí systému EPLAN a je možné jej nalézt v příloze 1.

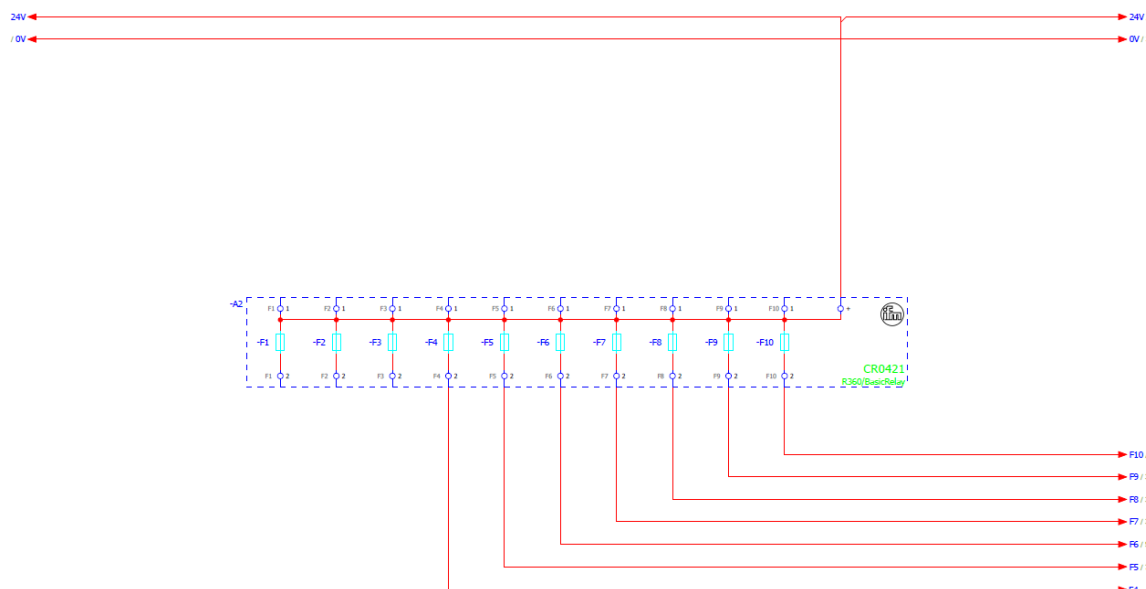
### 6.1 Napájení

Celá demonstrační úloha je napájena střídavým napětím 230 V ze zásuvky pomocí EURO kabelu. Tento kabel je připojen do EURO zástrčky vytvořené krabičky. V krabičce jde napájecí napětí přes TOTAL STOP tlačítko, které slouží k nouzovému vypnutí celé demonstrační úlohy. Napětí z krabičky pokračuje do napájecího zdroje, který převede střídavé napětí 230 V na stejnosměrné napětí 24 V. Tímto napětím se napájí všechny komponenty demonstrační úlohy.



Obr. 27 - Napájení demonstrační úlohy

Výstupní napětí napájecího zdroje je přivedeno na vstup BasicRelay. Napětí se v komponentu vede na příslušné pojistky, kterými jsou chráněny následující komponenty. Přes pojistky F10, F9 a F8 napájíme kontrolér. Kontrolér potřebuje 3 pojistky z důvodu chránění zvlášť pro vstupy, výstupy a CAN. Z pojistky F7 je napájena ethernetová kamera. Na pojistku F6 je připojen dialogový displej. Driver je napájen z pojistky F5 a digitální ukazatel z pojistky F4.

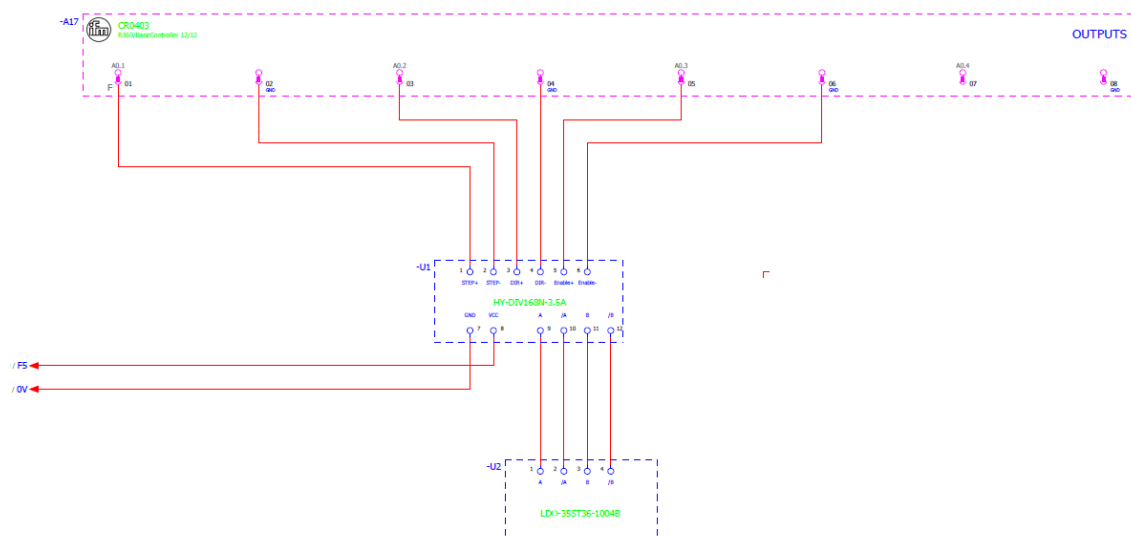


Obr. 28 - Zapojení pojistek

## 6.2 Vstupy a výstupy kontroléru

Do vstupů kontroléru jsou zapojeny jednotlivé analogové a digitální snímače. Na vstup IN01 je přiveden výstupní signál analogového snímače náklonu. Výstupní signál digitálního ukazatele je zapojen do vstupu IN02 kontroléru. Snímač teploty je zapojen do vstupu IN05 a snímač polohy do vstupu IN06.

Výstupy kontroléru jsou přivedeny na driver. Tento driver je ovládán pomocí 3 výstupních signálů přes rezistory o hodnotě 2 kΩ. První ze signálů slouží k zapínání a vypínání driveru. Druhý ke změně směru otáčení krokového motoru. Posledním signálem se pomocí pulsů ovládá krokový motor.

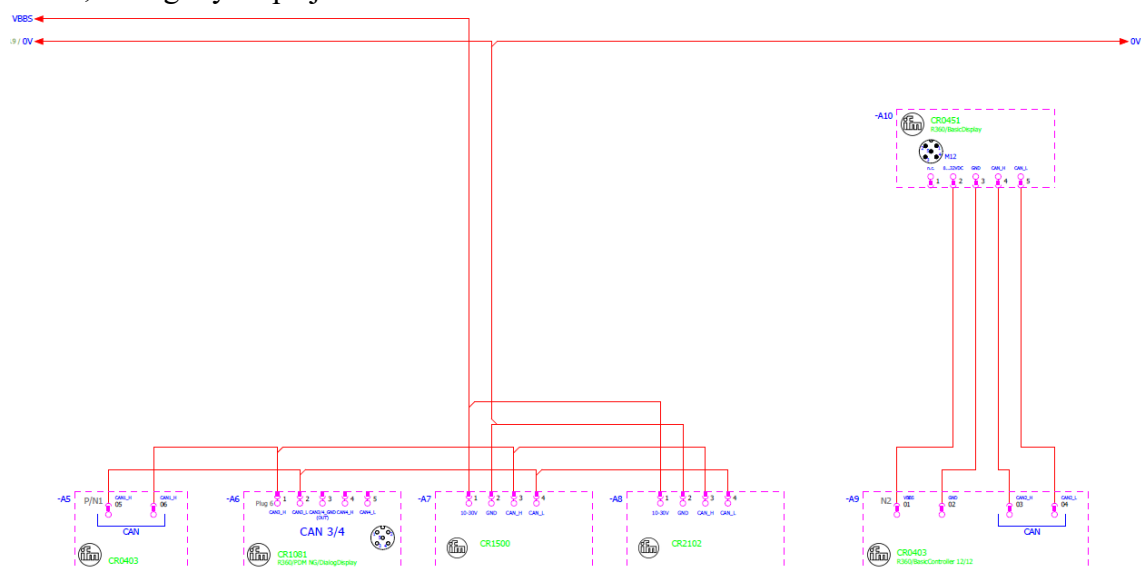


Obr. 29 - Ukázka zapojení driveru s motorem

## 6.3 Sběrnice CAN

Do prvního rozhraní CAN kontroléru je připojen BasicDisplay, který slouží k vizualizování demonstrační úlohy a k ovládání krokového motoru.

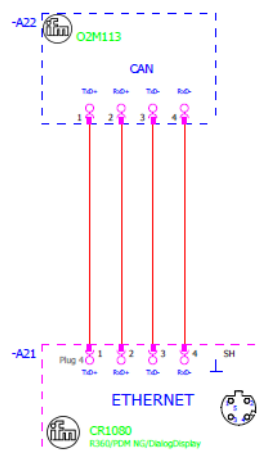
Do druhého rozhraní CAN kontroléru je připojeno více komponentů. Zařízení jsou připojeny pomocí lineární topologie. Připojené zařízení do CAN sítě jsou klávesnicový modu, dialogový displej a snímač náklonu.



Obr. 30 - Zapojení sběrnice CAN

## 6.4 Ethernet

Pomocí ethernetu je spojena ethernetová kamera a dialogový displej. Kamera přenáší obraz do dialogového displeje, kde je poté obraz vizualizován.



Obr. 31 - Ethernet – zapojení

## **6.5      Kufr**

Demonstrační úloha je zapojena do kufru. Ke spodní straně kufru je přišroubován podklad. Na tomto podkladu jsou přidělány tyto zařízení: krabice s TOTAL STOP, BasicRelay CR0451, BasicController CR0403 s BasiDisplay CR0451, klávesnicový modul, digitální ukazatel, rotační snímač, driver a motor. Na víko kufru je připevněný dialogový displej a gumičkou upevněné snímače, se kterými je možné volně manipulovat.

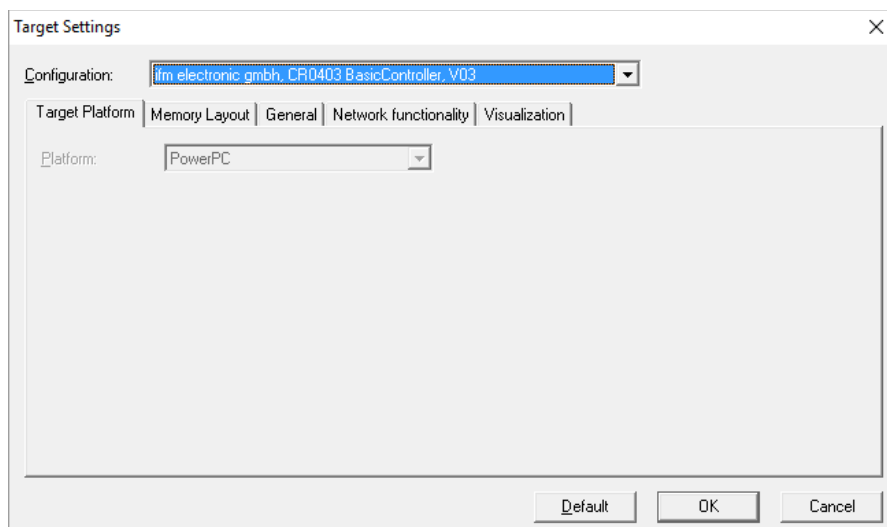
## 7 PROGRAMOVÁNÍ DEMONSTRAČNÍ ÚLOHY

V této kapitole je popsán postup vývoje programu pro demonstrační úlohu. Celý program je napsán ve vývojovém prostředí CoDeSys, do kterého jsou vloženy soubory pro jednotlivá zařízení společnosti IFM electronic. Kapitola je rozdělena na podkapitoly zařízení, které jsou programovány. Každá podkapitola má své další podkapitoly, ve kterých jsou popsány PLC konfigurace, globální proměnné, knihovny, samotný program a případně vizualizace u displejů.

Jako první je popsán BasicController, který je hlavou celé úlohy a funguje jako master v síti CANopen. Dále jsou popsány dvě slave zařízení: dialogový displej a BasicDisplay.

### 7.1 BasicController CR0403

Při tvorbě programu v CoDeSys je nejprve potřeba vybrat Target Settings pro náš kontrolér. Pro demonstrační úlohu je potřeba zvolit nastavení podle Obr.32 a potvrdit tlačítkem OK.

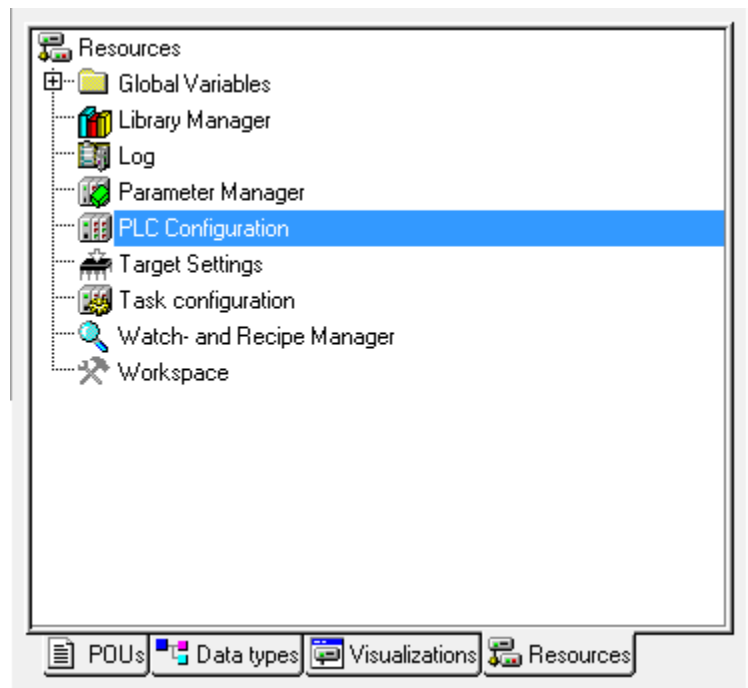


Obr. 32 - CoDeSys - CR0403 target settings

V další kroku se program zeptá, jestli má vytvořit nový soubor projektu a jak ho má pojmenovat. Sám navrhne název PLC\_PRG, kterým se označuje hlavní část programu. Znovu potvrdíme tlačítkem OK.

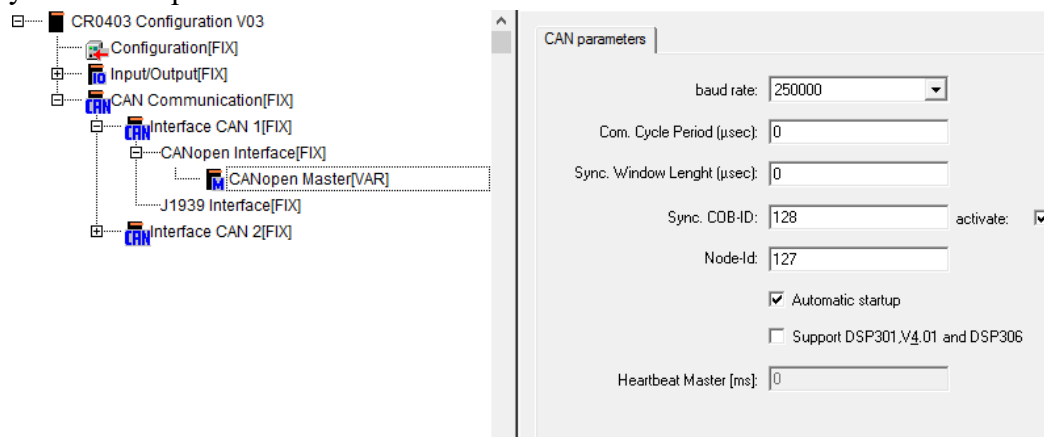
#### 7.1.1 PLC konfigurace

V objektovém organizéru poté otevřeme záložku Resources. V této záložce otevřeme položku PLC Configuration.



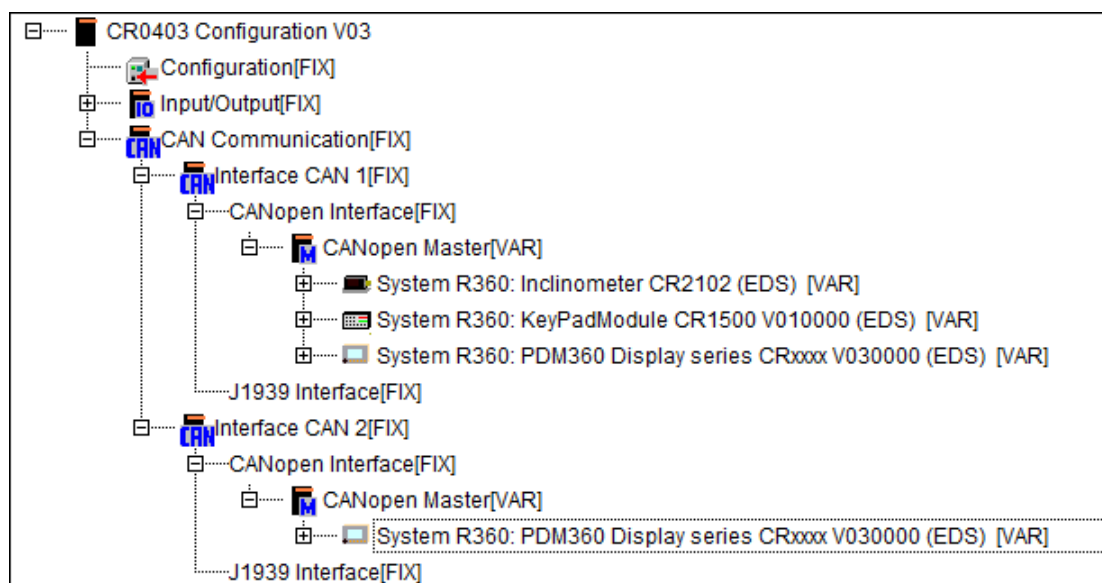
Obr. 33 - CoDeSys - CR0403 - Resources

V položce PLC Configuration je poté potřeba nastavit CAN rozhraní. Nastavení provedeme tak, že rozklikneme položku CR0403 Configuration V03 a poté položku CAN communication. Nejdříve je potřeba u položky Interface CAN 1 nastavit, že chceme, aby kontrolér fungoval jako CANopen Master. To provedeme tak, že klikneme pravým tlačítkem na CANopen interface vybereme možnost Append Subelement a z nabídky vybereme CANopen Master. V pravé části programu se nám ukáží parametry CAN rozhraní. Zde můžeme nastavit patřičný baud rate, Node-ID a automatické zapínání při zapnutí kontroléru. Pro naše potřeby zvolíme baud rate 2500000, Node-Id nastavíme na 127. Nastavení je vidět na Obr. 34. Tím máme nastaveno, že CAN rozhraní kontrolér bude fungovat jako Master. Teď je potřeba přidat zařízení, se kterými chceme po CAN síti komunikovat.



Obr. 34 - CoDeSys CR0403 - PLC Configuration I

Nejprve tedy přidáme snímač náklonu CR2102. Klikneme pravým tlačítkem myši na CANopen Master a vybereme Append Subelement. Z nabídky možných zařízení vybereme Inclinometer CR2102. V nastavení parametru snímače nastavíme Node-ID na 2. Pomocí záložky Service Data Object můžeme měnit parametry snímače, které jsou definované v datasheetu. Dále opakujeme akci pro klávesnicový modul CR1500 a dialogový displej CR1080. Klávesnicovému modulu nastavíme Node-ID na 4 a dialogovému displeji na 1. Tímto je provedeno nastavení CAN 1 rozhraní. Tuto akci opakujeme i pro položku Interface CAN 2, kde nastavíme baud rate na 250000 a vložíme Display series CRxxxx pro BasicDisplay CR0451. Výsledné nastavení PLC konfigurace je možné vidět na Obr. 35.



Obr. 35 - CoDeSys - PLC Configuration II

## 7.1.2 Globální proměnné

Nejprve vytvoříme globální proměnné pro potřebné vstupy a výstupy, abychom mohli dále v programu snadno používat. Začneme s analogovými vstupy a výstupy kontroléru. Jelikož tyto hodnoty se získávají přímo z bloků, tak je nebudeme definovat pomocí %I nebo %Q, ale jen jako hodnoty v paměti. Definované hodnoty vypadají podle Obr. 36.

V dalším kroku nadefinujeme hodnoty pro dialogový displej CR1080 a BasicDisplay CR0451. Pro demonstrační úlohu je potřeba nadefinovat adresy tlačítek, led diod a aktuální stránky dialogového displeje. Dále je potřeba nadefinovat výstupní hodnoty, které budou do dialogového displeje odesílány. Definici je možné vidět na Obr. 37.

```
0001 VAR_GLOBAL
0002 (*Vstupy*)
0003 Otacky: INT;
0004 Naklon_analog: INT;
0005 Teplota: INT;
0006 Poloha: BOOL;
0007
0008 (*Výstupy*)
0009 M_Enable: BOOL:= TRUE;
0010 M_DIR: BOOL;
0011 M_STEP: WORD;
0012 M_Speed: WORD:= 0;
0013
0014 (*Ovládání motoru*)
0015 CR1500_Enable: BOOL;
0016
0017 END_VAR
0018
```

Obr. 36- CoDeSys  
CR0403 - Global  
variables I

|      |                                     |      |                                     |
|------|-------------------------------------|------|-------------------------------------|
| 0001 | VAR_GLOBAL                          | 0001 | VAR_GLOBAL                          |
| 0002 | cr1080_LED_F_keys AT %QB22:BYTE;    | 0002 |                                     |
| 0003 | cr1080_LED_F2_keys AT %QB37:BYTE;   | 0003 | CR0451_F1: BOOL;                    |
| 0004 | cr1080_LED_C_keys AT %QB23:BYTE;    | 0004 | CR0451_F2: BOOL;                    |
| 0005 |                                     | 0005 | CR0451_F3: BOOL;                    |
| 0006 | CR1080_F1: BOOL;                    | 0006 | CR0451_F4: BOOL;                    |
| 0007 | CR1080_F2: BOOL;                    | 0007 | CR0451_UP: BOOL;                    |
| 0008 | CR1080_F3: BOOL;                    | 0008 | CR0451_LEFT: BOOL;                  |
| 0009 | CR1080_F4: BOOL;                    | 0009 | CR0451_DOWN: BOOL;                  |
| 0010 | CR1080_F5: BOOL;                    | 0010 | CR0451_RIGHT: BOOL;                 |
| 0011 | CR1080_F6: BOOL;                    | 0011 | CR0451_OK: BOOL;                    |
| 0012 | CR1080_F7: BOOL;                    | 0012 |                                     |
| 0013 | CR1080_F8: BOOL;                    | 0013 | (*DATAZ PLC DO PDM*)                |
| 0014 | CR1080_F9: BOOL;                    | 0014 | CR0451_Dat AT %QB74:BYTE;           |
| 0015 | CR1080_left: BOOL;                  | 0015 | CR0451_Teplota AT %QB75:BYTE;       |
| 0016 | CR1080_right: BOOL;                 | 0016 | CR0451_Speed AT %QB76:BYTE;         |
| 0017 | CR1080_PUSH: BOOL;                  | 0017 | CR0451_Naklon_analog AT %QB77:BYTE; |
| 0018 |                                     | 0018 | CR0451_Naklon_canX AT %QB78:BYTE;   |
| 0019 | (*DATAZ PLC DO PDM*)                | 0019 | CR0451_Naklon_canY AT %QB79:BYTE;   |
| 0020 | CR1080_Dat AT %QB38:BYTE;           | 0020 | (*DATAZ PDM DO PLC*)                |
| 0021 | CR1080_Teplota AT %QB39:BYTE;       | 0021 | CR0451_F_keys AT %IB74: BYTE;       |
| 0022 | CR1080_Speed AT %QB40:BYTE;         | 0022 | CR0451_C_keys AT %IB75: BYTE;       |
| 0023 | CR1080_Naklon_analog AT %QB41:BYTE; | 0023 | CR0451_PRESENT_PAGE AT %IW35:WORD;  |
| 0024 | CR1080_Naklon_canX AT %QB42:BYTE;   | 0024 |                                     |
| 0025 | CR1080_Naklon_canY AT %QB43:BYTE;   | 0025 | END_VAR                             |
| 0026 |                                     |      |                                     |
| 0027 | (*DATAZ PDM DO PLC*)                |      |                                     |
| 0028 | cr1080_F_keys AT %IB40: BYTE;       |      |                                     |
| 0029 | cr1080_F2_keys AT %IB53: BYTE;      |      |                                     |
| 0030 | cr1080_C_keys AT %IB41: BYTE;       |      |                                     |
| 0031 | CR1080_PRESENT_PAGE AT %IW18:WORD;  |      |                                     |
| 0032 | END_VAR                             |      |                                     |

Obr. 37 - CoDeSys CR0403 - Global variables II

Podobně je potřeba nadefinovat tlačítka a led diody pro klávesnicový modul CR1500 a snímač náklonu CR2102. Definici je možné vidět na Obr. 38. A Obr. 39.

|      |                             |      |                      |
|------|-----------------------------|------|----------------------|
| 0001 | VAR_GLOBAL                  | 0001 | VAR_GLOBAL           |
| 0002 | (*CR1500*)                  | 0002 |                      |
| 0003 | cr1500_led AT %QD3:UDINT;   | 0003 | (*CR2102 vstupy*)    |
| 0004 | cr1500_bar1 AT %QB16:USINT; | 0004 | xaxis AT %IW11: INT; |
| 0005 | cr1500_bar2 AT %QB17:USINT; | 0005 | yaxis AT %IW12: INT; |
| 0006 | cr1500_keys AT %ID8:UDINT;  | 0006 |                      |
| 0007 |                             | 0007 | END_VAR              |
| 0008 | (*CR1500 tlačítka*)         |      |                      |
| 0009 | CR1500_key1: BOOL;          |      |                      |
| 0010 | CR1500_key2: BOOL;          |      |                      |
| 0011 | CR1500_key3: BOOL;          |      |                      |
| 0012 | CR1500_key4: BOOL;          |      |                      |
| 0013 | CR1500_key5: BOOL;          |      |                      |
| 0014 | CR1500_key6: BOOL;          |      |                      |
| 0015 | CR1500_key7: BOOL;          |      |                      |
| 0016 | CR1500_key8: BOOL;          |      |                      |
| 0017 | CR1500_key9: BOOL;          |      |                      |
| 0018 | CR1500_key10: BOOL;         |      |                      |
| 0019 | CR1500_key11: BOOL;         |      |                      |
| 0020 | CR1500_key12: BOOL;         |      |                      |
| 0021 | CR1500_ledbar1_value: BYTE; |      |                      |
| 0022 | CR1500_ledbar2_value: BYTE; |      |                      |
| 0023 | CR1500_keyleft1: BOOL;      |      |                      |
| 0024 | CR1500_keyright1: BOOL;     |      |                      |
| 0025 | CR1500_keyleft2: BOOL;      |      |                      |
| 0026 | CR1500_keyright2: BOOL;     |      |                      |
| 0027 | END_VAR                     |      |                      |

Obr. 38 - CoDeSys CR0403 - Global variables III

Obr. 39 - CoDeSys CR0403 - Global variables IV

### 7.1.3 Knihovny

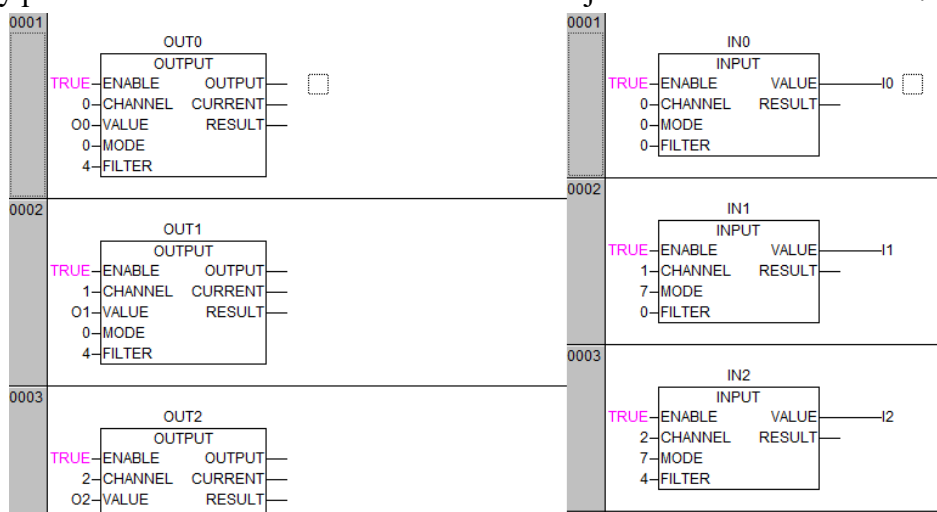
Abychom byli schopni správně ovládat všechny možnosti kontroléru, tak musíme vložit knihovny s předdefinovanými bloky a nastaveními. Pro kontrolér CR0403 musíme vložit knihovny pro CAN, CANopen a knihovnu samotného kontroléru. Výsledek by měl vypadat podobně jako na Obr. 40.

```
ifm_CANopen\ifm_CANopen_NT_V020004.lib 21.3.16 00:48:18
3S_CanDrv.lib 21.3.16 00:48:18
standard.lib 4.10.05 12:14:46
SysLibCallback.lib 26.2.16 17:15:25
ifm_RAWCan\ifm_RAWCan_NT_V020004.lib 21.3.16 00:48:18
ifm_CR0403\ifm_CR0403_V030100.lib 21.3.16 00:48:18
```

Obr. 40 - CodeSys - CR0403 – Knihovny

### 7.1.4 Programování kontroléru

Nejprve je potřeba nastavit vstupy a výstupy pomocí bloků, které jsou definované v knihovně pro kontrolér. Vstupy se nastavují pomocí bloku INPUT a výstupy pomocí bloku OUTPUT. Ukázku nastavení je možné vidět na Obr. 41.

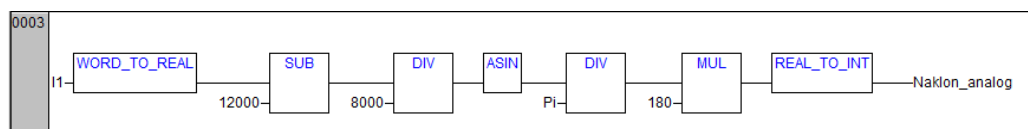


Obr. 41 - CoDeSys - CR0403 - INPUT a OUTPUT

Tyto bloky jsou nastaveny pro všechny vstupy a výstupy v blocích Inputs a Outputs, které mají jako vstupy/výstupy pouhé hodnoty signálů. Vstupní signály se poté dále převádí ze standardizované hodnoty na hodnotu měřenou.

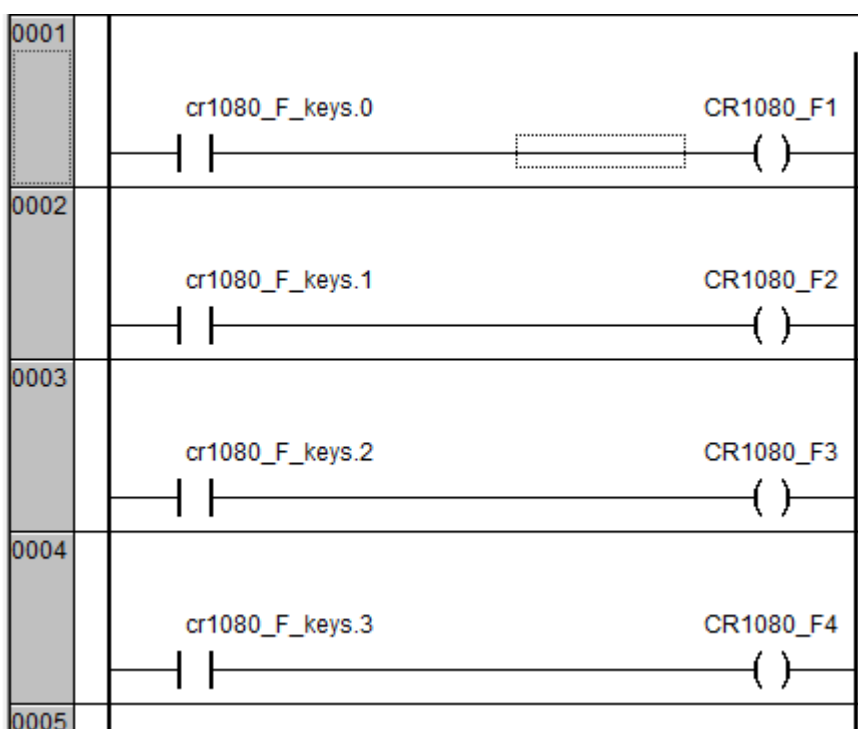
Převod standardizované veličiny na měřenou veličinu závisí na výstupní charakteristice snímače. Analogový snímač náklonu má výstupní funkci rovnu  $I_A = 12\text{mA} + \sin(\alpha) * 8\text{mA}$ . Díky této funkci jsem vytvořil přepočít pro úhel náklonu snímače. Výsledný převod v programu Obr. 42. Stejným způsobem jsem poté

pokračoval pro snímač teploty a snímač otáček, které mají výstupní funkci přímky. Tímto je převod hotový a hodnoty snímačů máme v kontroléru ve správných veličinách.



Obr. 42 - CoDeSys - CR0403 Převod analogového signálu

Nyní je potřeba přiřadit tlačítkům BasicDisplaye, klávesnicového modulu a dialogového displeje srozumitelný název, aby se s nimi dalo srozumitelně a jednoduše pracovat. Hodnoty tlačítek se ze zařízení posílají po bytech, kde každý bit představuje hodnotu daného tlačítka. Stačí tedy využít tuto hodnotu a přiřadit ji patřičnému tlačítku ve formátu bool. Ukázku pro dialogový displej CR1080 můžeme vidět na Obr. 43. Stejným způsobem se postupuje i u displeje BasicDisplay a klávesnicového modulu.



Obr. 43 - CoDeSys - CR0403 - Tlačítka z CR1080/CR0451/ CR1500

Díky předcházejícím krokům jsme schopni v kontroléru sledovat jakékoliv zmáčknutí tlačítka na obou displejích a klávesnicovém modulu. Těmito tlačítky má jít ovládat krokový motor ze všech tří zařízení. Je tedy potřeba vytvořit blok, který bude ovládat krokový motor.

Tento blok vytvoříme podle signálů driveru. Vstupy bloku motoru tedy budou Enable, DIR, Speed\_UP, Speed\_DOWN. Vstup při příchodu signálu na vstup DIR se změní směr otáčení krokového motorku. Vstupy Speed\_UP a Speed\_DOWN jsou závislé na vstupu Enable. Pokud tento vstup není sepnutý, tak těmito vstupy sice projde signál, ale čítač jen nezapočítá, jelikož je restartován vstupem Enable. To znamená, že

změna rychlosti není možná, pokud driver motoru není zapnutý. Rychlost motorku je potom zaznamenána do výstupní proměnné Speed. Kód ovládacího bloku je možné vidět na Obr. 44.

```

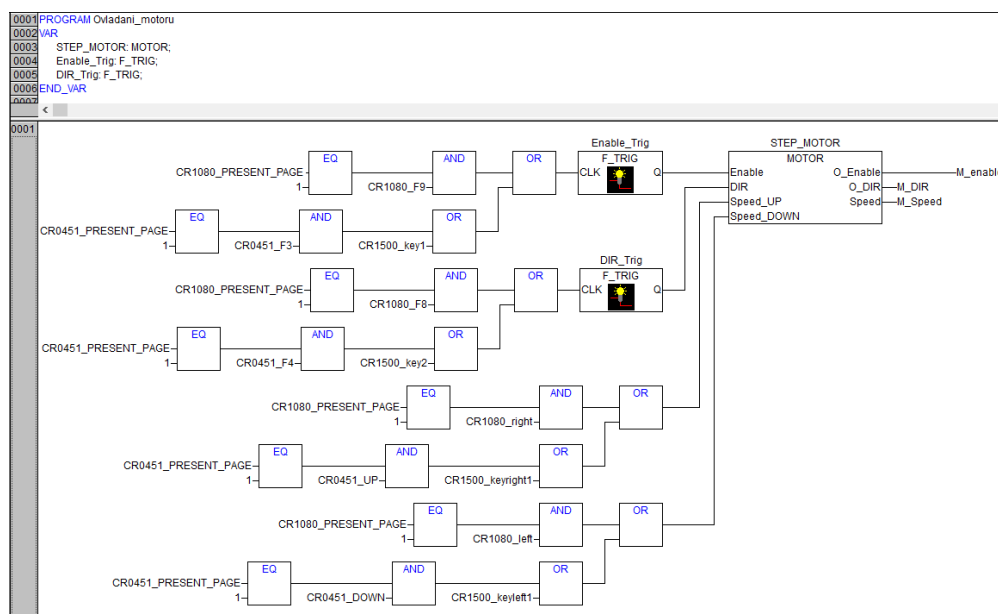
0001 FUNCTION_BLOCK MOTOR
0002 VAR_INPUT
0003     Enable: BOOL;
0004     DIR: BOOL;
0005     Speed_UP: BOOL;
0006     Speed_DOWN: BOOL;
0007 END_VAR
0008 VAR_OUTPUT
0009     O_Enable: BOOL;
0010     O_DIR: BOOL;
0011     Speed: WORD;
0012 END_VAR
0013 VAR
0014     Speed_Counter: CTUD;
0015     Speed_Plus: BOOL;
0016     Speed_Minus: BOOL;
0017 END_VAR
0001 IF Enable AND O_Enable THEN
0002     O_Enable := FALSE;
0003 ELSIF Enable THEN
0004     O_Enable := TRUE;
0005 END_IF
0006 IF DIR AND O_DIR THEN
0007     O_DIR := FALSE;
0008 ELSIF DIR THEN
0009     O_DIR := TRUE;
0010 END_IF
0011 IF Speed_UP AND Speed_Counter.QU = FALSE THEN
0012     Speed_Plus := TRUE;
0013 ELSE
0014     Speed_Plus := FALSE;
0015 END_IF
0016 IF Speed_DOWN AND Speed_Counter.QD = FALSE THEN
0017     Speed_Minus := TRUE;
0018 ELSE
0019     Speed_Minus := FALSE;
0020 END_IF
0021
0022 Speed_Counter(CU:=Speed_Plus,CD:=Speed_Minus,RESET:=M_Enable,PV:=100,CV=>Speed);
0023

```

Obr. 44 - CR0403 – CoDeSys - Blok motoru

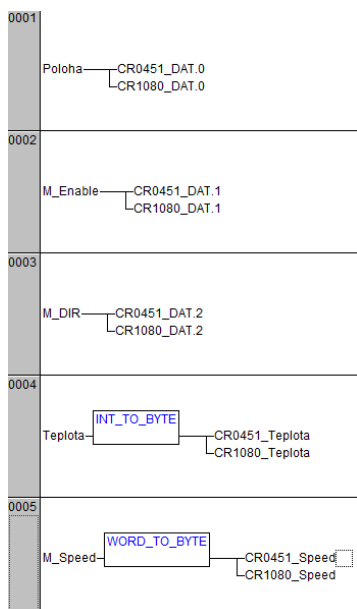
Ted' je potřeba tento blok použít k ovládání pomocí tlačítek displejů a klávesnicového modulu. U displejů je potřeba si dát pozor na to, na jaké straně vizualizace se nacházíme.

Aby se nám neprovedla operace několikrát, tak musíme vstupy Enable a DIR připojit přes blok F\_TRIG nebo R\_TRIG, které zajistí, že za jedno zmáčknutí se nám provede změna těchto vstupů jen jednou. Vstup Enable je ovládám ze tří zařízení, takže pomocí dvou bloků OR za sebou vytvoříme tři možné cesty pro sepnutí vstupu. Z dialogového displeje jde ovládat vstup Enable pouze, pokud vizualizace je na první stránce neboli CR1080\_PRESENT\_PAGE = 1 a je sepnuté tlačítko CR1080\_F9. Obdobně platí nastavení pro BasicDisplay, jen musí být sepnuté tlačítko CR0451\_F3 a na klávesnicovém modulu stačí, aby bylo sepnuto tlačítko CR1500\_F1. Stejně pokračujeme i pro vstup DIR jen s jinými tlačítky. Vstupy Speed\_UP a Speed\_DOWN jsou identické, jen už nepotřebují F\_TRIG nebo R\_TRIG, jelikož tuto funkci zajistí čítač CTUD. Použití bloku je vidět na Obr. 45.



Obr. 45 – CoDeSys - CR0403 – Ovládání motoru

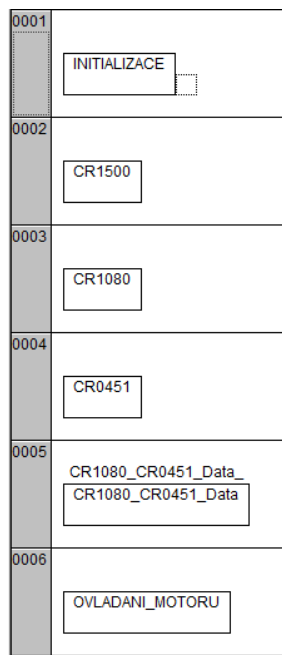
Poslední věcí, kterou je potřeba udělat v kontroléru, je odeslat příslušné hodnoty snímačů a stav motoru do displejů, aby mohly být vizualizovány. Toho docílíme tak, že využijeme předdefinované rezervované byty z PLC konfigurace. Tyto byty už jsme nadefinovali pomocí globálních proměnných (viz Obr. 37). Nyní jen do těchto hodnot vložíme správné hodnoty a knihovny nám sami zajistí jejich odeslání po sběrnici CAN do správného zařízení.



Obr. 46 – CoDeSys - CR0403 – Přenos dat do displejů

Budeme potřebovat 7 bytů, do kterých vložíme přenášené hodnoty. Do prvního bitu prvního bytu vložíme informaci o sepnutí snímače polohy. Do druhého a třetího bitu vložíme informaci o zapnutí a vypnutí driveru a směru otáčení motoru. Druhý byte nám přenáší hodnotu teploty. Třetí byte bude sloužit k přenosu hodnoty rychlosti motoru. Čtvrtý byte k přenosu hodnoty analogového snímače náklonu. Pátý a šestý byte k přenosu hodnot x a y snímače náklonu. Poslední sedmý byte bude sloužit k přenosu hodnoty otáček ze snímače otáček. Ukázku kódu je možné vidět na Obr. 46.

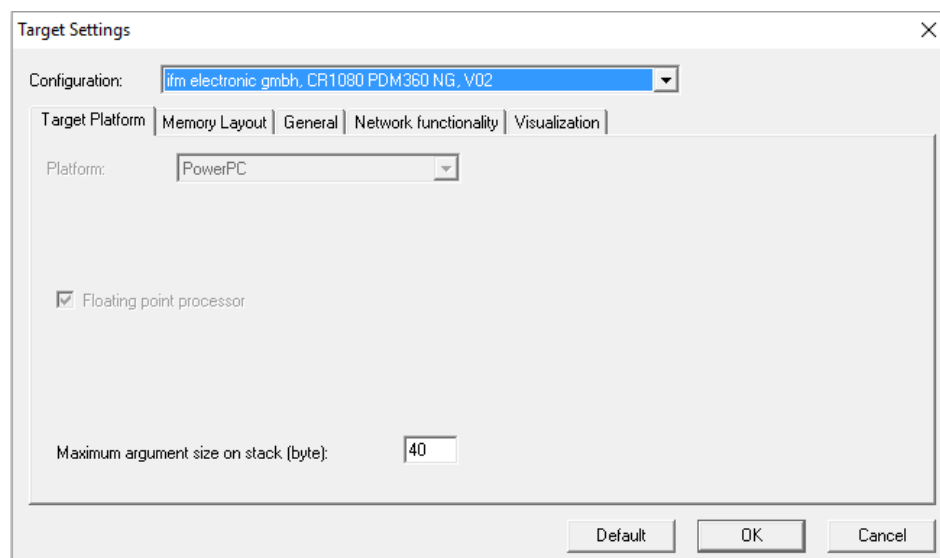
Tímto je kontrolér plně naprogramován a stačí už jen vytvořené podprogramy zavolat v hlavním programu (viz Obr. 47.). Dále stačí naprogramovat displeje a celý systém bude plně fungovat.



Obr. 47 - CoDeSys - CR0403 – Hlavní program

## 7.2 Dialogový displej CR1080

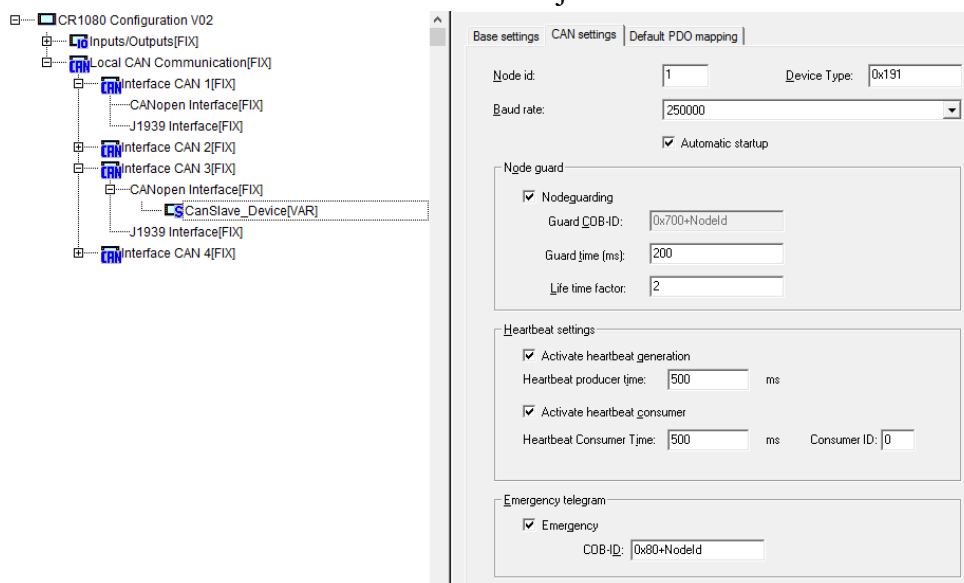
Stejně jako u kontroléru je nejprve potřeba vybrat patřičné nastavení v Target Settings. Pro náš dialogový displej vybereme možnost podle Obr. 48. V záložce Network functionality je potřeba ještě upravit položku Names of supported network interfaces na CAN;UDP. Díky této úpravě budeme schopni programovat dialogový displej pomocí ethernetového kabelu. Po odkliknutí tlačítka OK se nás program opět zeptá, jestli chceme vytvořit hlavní program, tak potvrdíme tlačítkem OK.



Obr. 48 - CoDeSys - CR1080 - Target Settings

## 7.2.1 PLC konfigurace

Stejně jako u kontroléru, tak i tady musíme nastavit CAN rozhraní. Rozdíl bude v tom, že toto zařízení bude fungovat jako slave. Pro naše účely využijeme Interface CAN 3, kde u položky CANopen přidáme položku CanSlave\_Device. Node-ID nastavíme na 1 a baud rate na 250000. Nastavení je možné vidět na obrázku Obr. 49.



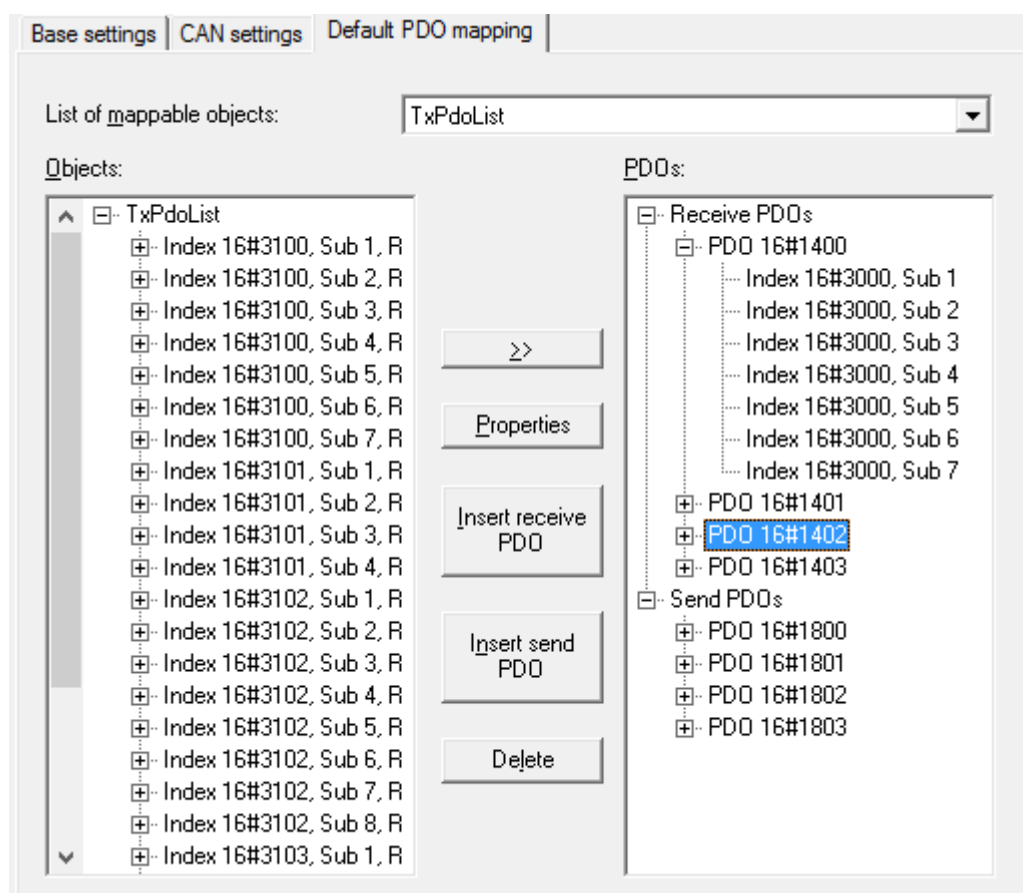
Obr. 49 - CoDeSys - CR1080 - PLC konfigurace

Dále je potřeba nastavit v Parameter Manager byty, které se mají přes sběrnici CAN posílat. Firma IFM electronic dodává template s přednastavenými hodnotami pro slave zařízení. Díky tomuto template nám budou data posílaná přes sběrnice přesně sedět s daty, které očekáváme v PLC konfiguraci v kontroléru. Tuto nabídku je také možno rozšířit a posílat i další data, pokud je potřeba. Ukázku nastavení je možné vidět na Obr. 50.

| Name                  | Index   | SubIndex | Accessright | Variable              | Value |
|-----------------------|---------|----------|-------------|-----------------------|-------|
| RdDataPdo1B/Id0_1     | 1583000 | 1581     | write only  | DESIRED_PAGE          |       |
| RdDataPdo1B/Id2       | 1583000 | 1582     | write only  | DESIRED_MESSAGE       |       |
| RdDataPdo1B/Id3       | 1583000 | 1583     | write only  | DESIRED_MESSAGE_GROUP |       |
| RdDataPdo1B/Id4       | 1583000 | 1584     | write only  | LED_F_VKEYS           |       |
| RdDataPdo1B/Id5       | 1583000 | 1585     | write only  | LED_C_KEYS            |       |
| RdDataPdo1B/Id6       | 1583000 | 1586     | write only  | DESIRED_LEDVALUE_MAX  |       |
| RdDataPdo1B/Id7       | 1583000 | 1587     | write only  | DESIRED_LEDVALUE_MIN  |       |
| RdDataPdo2B/Id0       | 1583001 | 1581     | write only  | DESIRED_BACKLIGHT     |       |
| RdDataPdo2B/Id1_2     | 1583001 | 1582     | write only  | HANDLE_BACK           |       |
| RdDataPdo2B/Id2_4_5_6 | 1583001 | 1583     | write only  | INPUT_VALUE_BACK      |       |
| RdDataPdo2B/Id7       | 1583001 | 1584     | write only  | LED_F_VKEYS2          |       |
| RdDataPdo3B/Id0       | 1583002 | 1581     | write only  | TOPdmPdo3ReserveByte1 |       |
| RdDataPdo3B/Id1       | 1583002 | 1582     | write only  | TOPdmPdo3ReserveByte2 |       |
| RdDataPdo3B/Id2       | 1583002 | 1583     | write only  | TOPdmPdo3ReserveByte3 |       |
| RdDataPdo3B/Id3       | 1583002 | 1584     | write only  | TOPdmPdo3ReserveByte4 |       |
| RdDataPdo3B/Id4       | 1583002 | 1585     | write only  | TOPdmPdo3ReserveByte5 |       |
| RdDataPdo3B/Id5       | 1583002 | 1586     | write only  | TOPdmPdo3ReserveByte6 |       |
| RdDataPdo3B/Id6       | 1583002 | 1587     | write only  | TOPdmPdo3ReserveByte7 |       |
| RdDataPdo4B/Id0       | 1583003 | 1581     | write only  | TOPdmPdo4ReserveByte1 |       |
| RdDataPdo4B/Id1       | 1583003 | 1582     | write only  | TOPdmPdo4ReserveByte2 |       |
| RdDataPdo4B/Id2       | 1583003 | 1583     | write only  | TOPdmPdo4ReserveByte3 |       |
| RdDataPdo4B/Id3       | 1583003 | 1584     | write only  | TOPdmPdo4ReserveByte4 |       |
| RdDataPdo4B/Id4       | 1583003 | 1585     | write only  | TOPdmPdo4ReserveByte5 |       |
| RdDataPdo4B/Id5       | 1583003 | 1586     | write only  | TOPdmPdo4ReserveByte6 |       |
| RdDataPdo4B/Id6       | 1583003 | 1587     | write only  | TOPdmPdo4ReserveByte7 |       |

Obr. 50 - CoDeSys - CR1080 - Parameter Manager

Takto vytvořený list v Parameter Manager je ještě nutné namapovat v nastavení slave zařízení. To provedeme tak, že otevřeme PLC konfiguraci a klikneme na CanSlave\_Device. V pravé části vybereme záložku Default PDO mapping a zde je potřeba vložit správně údaje, aby posílaná a přijímaná data odpovídala těm co očekáváme v kontroléru. Ukázku provedení je možné vidět na Obr. 51.



Obr. 51 - CoDeSys - CR1080 - Mapování

## 7.2.2 Knihovny

Stejně jako u kontroléru je potřeba vložit patřičné knihovny pro správný chod displeje. Je zde potřeba vložit vlastní knihovnu pro displej, CAN knihovny a knihovny pro ethernet. Všechny vložené knihovny je vidět na Obr. 52.

---

```
ifm_PDMng_util_V010301.lib 26.2.16 17:16:19
ifm_CR1080\ifm_CR1080_V020200.lib 26.2.16 17:16:21
ifm_PDM360NG_linux_syscall_asynch_V010100.lib 26.2.16 17:16:19
SysLibCallback.lib 26.2.16 17:15:25
ifm_CANopen\3S_CanOpenDevice.lib 21.3.16 00:48:18
ifm_CANopen\ifm_NetVarLib_NT_V020004.lib 21.3.16 00:48:18
ifm_RAWCan\ifm_RAWCan_NT_V020004.lib 21.3.16 00:48:18
ifm_CANopen\ifm_CANopen_NT_V020004.lib 21.3.16 00:48:18
lecsfc.lib 13.4.06 16:51:28
Util.lib 26.2.16 17:15:25
SysLibFile.lib 26.2.16 17:15:25
SysLibMem.lib 26.2.16 17:15:25
ifm_pdm_input_V010004.lib 26.2.16 17:16:19
SysLibTime.lib 26.2.16 17:15:25
SysLibRtc.lib 26.2.16 17:15:25
ifm_CANopen\3S_CANopenManager.lib 21.3.16 00:48:18
ifm_CANopen\3S_CanDrv.lib 21.3.16 00:48:18
3S_CANOPENNETVAR.LIB 21.3.16 00:48:18
SysLibSockets.lib 26.2.16 17:15:25
NETVARUDP_LIB_V23.LIB 26.2.16 17:15:25
LinuxSystemLib.Lib 26.2.16 17:15:25
standard.lib 4.10.05 12:14:46
```

Obr. 52 - CoDeSys CR1080 - Knihovny

### 7.2.3 Globální proměnné

Nejdříve ze všeho musíme nadefinovat globální proměnné, které jsme použili v Parameter manager. Díky tomu, že jsme použili template od firmy IFM, tak už jsou automaticky nadefinovány proměnné pro přenášená data, všechny tlačítka a proměnné pro texty k jednotlivým tlačítkům. Hidden tlačítka budou sloužit ke skrývání jednotlivých tlačítek, pokud nejsou potřeba na dané stránce vizualizace. Proměnné textů budou sloužit k popsání tlačítek a popsání celé stránky. Globální proměnné je možné vidět na Obr. 52 nalevo.

|      |                                |      |                       |
|------|--------------------------------|------|-----------------------|
| 0001 | VAR_GLOBAL                     | 0001 | VAR_GLOBAL            |
| 0002 |                                | 0002 |                       |
| 0003 |                                | 0003 | SoftKey_F1: BOOL;     |
| 0004 | InputModeDt: BOOL;             | 0004 | SoftKey_F2: BOOL;     |
| 0005 | TextColor: DWORD := 16#FFFFFF; | 0005 | SoftKey_F3: BOOL;     |
| 0006 | TextSize: BYTE := 12;          | 0006 | SoftKey_F4: BOOL;     |
| 0007 |                                | 0007 | SoftKey_F5: BOOL;     |
| 0008 | Hidden_F1: BOOL;               | 0008 | SoftKey_F6: BOOL;     |
| 0009 | Hidden_F2: BOOL;               | 0009 | SoftKey_F7: BOOL;     |
| 0010 | Hidden_F3: BOOL;               | 0010 | SoftKey_F8: BOOL;     |
| 0011 | Hidden_F4: BOOL;               | 0011 | SoftKey_F9: BOOL;     |
| 0012 | Hidden_F5: BOOL;               | 0012 | SoftKey_F10: BOOL;    |
| 0013 | Hidden_F6: BOOL;               | 0013 | SoftKey_Up: BOOL;     |
| 0014 | Hidden_F7: BOOL;               | 0014 | SoftKey_Esc: BOOL;    |
| 0015 | Hidden_F8: BOOL;               | 0015 | SoftKey_Ok: BOOL;     |
| 0016 | Hidden_F9: BOOL;               | 0016 | SoftKey_Left: BOOL;   |
| 0017 | Hidden_Ctrl: BOOL;             | 0017 | SoftKey_Down: BOOL;   |
| 0018 | Hidden_Up: BOOL;               | 0018 | SoftKey_Right: BOOL;  |
| 0019 | Hidden_Down: BOOL;             | 0019 | SoftKey2_F1: BOOL;    |
| 0020 | Hidden_Left: BOOL;             | 0020 | SoftKey2_F2: BOOL;    |
| 0021 | Hidden_Right: BOOL;            | 0021 | SoftKey2_F3: BOOL;    |
| 0022 | Hidden_Ok: BOOL;               | 0022 | SoftKey2_F4: BOOL;    |
| 0023 | Hidden_HeadLine: BOOL;         | 0023 | SoftKey2_F5: BOOL;    |
| 0024 |                                | 0024 | SoftKey2_F6: BOOL;    |
| 0025 | TXT_F1: STRING[20];            | 0025 | SoftKey2_F7: BOOL;    |
| 0026 | TXT_F2: STRING[20];            | 0026 | SoftKey2_F8: BOOL;    |
| 0027 | TXT_F3: STRING[20];            | 0027 | SoftKey2_F9: BOOL;    |
| 0028 | TXT_F4: STRING[20];            | 0028 | SoftKey2_F10: BOOL;   |
| 0029 | TXT_F5: STRING[20];            | 0029 | SoftKey2_Up: BOOL;    |
| 0030 | TXT_F6: STRING[20];            | 0030 | SoftKey2_Esc: BOOL;   |
| 0031 | TXT_F7: STRING[20];            | 0031 | SoftKey2_Ok: BOOL;    |
| 0032 | TXT_F8: STRING[20];            | 0032 | SoftKey2_Left: BOOL;  |
| 0033 | TXT_F9: STRING[20];            | 0033 | SoftKey2_Down: BOOL;  |
| 0034 | TXT_Ctrl: STRING[20];          | 0034 | SoftKey2_Right: BOOL; |
| 0035 | TXT_HeadLine: STRING[30];      | 0035 | InputMode: BOOL;      |
| 0036 |                                | 0036 |                       |
| 0037 | END_VAR                        | 0037 | END_VAR               |

Obr. 53 - CoDeSys - CR1080 - Globální proměnné

Dále jsou v tomto template vytvořeny tlačítka Softkey1 a Softkey2. Tlačítka Softkey1 reagují pouze na nástupnou hranu, zatímco tlačítka Softkey2 jsou statická a reagují na zmáčknutí. Definici proměnných je možné vidět na Obr. 52 napravo.

## 7.2.4 Programování displeje

V našem případě bude displej sloužit jako slave zařízení, které bude mít za úkol přijmout data z kontroléru a kamery a vizualizovat je. Nejprve tedy vytvoříme soubor „DISPLAY\_SETTINGS“, kde hodnoty získané z kontroléru převedeme na přívětivější názvy pro pozdější použití ve vizualizaci. Ukázka je na Obr. 53.

|      |                                    |
|------|------------------------------------|
| 0001 | ToPdmPdo3ReserveByte1.0 — Poloha   |
| 0002 | ToPdmPdo3ReserveByte1.1 — M_Enable |
| 0003 | ToPdmPdo3ReserveByte1.2 — M_DIR    |

Obr. 54 - CoDeSys - CR1080 - Příprava dat pro vizualizaci

Nyní už jen vytvoříme program pro přepínání stránek vizualizace. Nejdříve vytvoříme blok SELECT\_PAGE. Tento blok bude mít za úkol vyhodnocovat co má být na dané stránce zobrazeno, na kterou stránku je možné se dále přepnout pomocí tlačítek a jaká stránka se má právě zobrazit. Musí zde být tedy vypsány všechny stránky, které jsou používány. Jednotlivé stránky jsou rozděleny podle identifikačního čísla, aby byla možná jejich identifikace. Ukázku nastavení úvodní stránky je možné vidět na Obr. 54.

```
1: (*Set txt for softkeys*)
TXT_F1:=MOTOR;
TXT_F2:=NÁKLON;
TXT_F3:=POLOHA;
TXT_F4:=KAMERA;
TXT_F5:=;
TXT_F6:=;
TXT_F7:=;
TXT_F8:=DIR;
TXT_F9:=ON/OFF;
TXT_HeadLine:=Demokutr R360;
TextColor:=16#FFFFFF;
TextSize:=18;
Hidden_F1:=FALSE;
Hidden_F2:=FALSE;
Hidden_F3:=FALSE;
Hidden_F4:=FALSE;
Hidden_F5:=TRUE;
Hidden_F6:=TRUE;
Hidden_F7:=TRUE;
Hidden_F8:=FALSE;
Hidden_F9:=FALSE;
Hidden_Ctrl:=FALSE;
Hidden_Up:=FALSE;
Hidden_Down:=TRUE;
Hidden_Left:=TRUE;
Hidden_Right:=TRUE;
Hidden_Ok:=TRUE;
Hidden_HeadLine:=FALSE;

IF SoftKey_F2 THEN
    PAGE_INTERN:=2;
END_IF;
IF SoftKey_F3 THEN
    PAGE_INTERN:=3;
END_IF;
IF SoftKey_F4 THEN
    PAGE_INTERN:=4;
END_IF;
```

Obr. 55 - CoDeSys - CR1080 - Příprava dat pro vizualizaci

Jakmile máme předchozí nastavování stránek hotové pro všechny stránky, které chceme použít, tak je potřeba nastavit blok PDM\_PAGECONTROL. Do vstupu INIT\_PAGE je potřeba přivést úvodní stránku neboli 1. Do BASENAME zase písmeno „P“ pro identifikaci stránek. Výstupem je poté stránka, kterou chceme zobrazit neboli PRESENT\_PAGE.

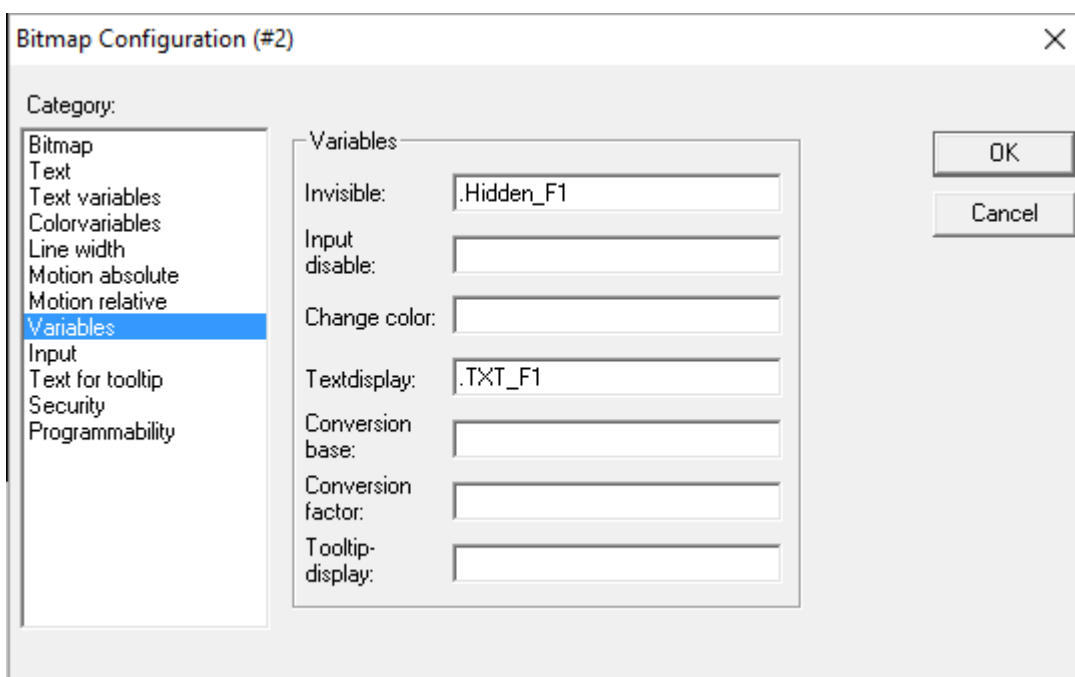
## 7.2.5 Vizualizace

Vizualizace v prostředí CoDeSys je velmi jednoduchá a intuitivní. Přejdeme tedy do záložky Visualization v objektovém organizéru. Nejdříve je potřeba vytvořit MASTERLAYOUT\_1, který se bude zobrazovat pod všemi ostatními stránkami. Využijeme opět layout z template od firmy IFM.



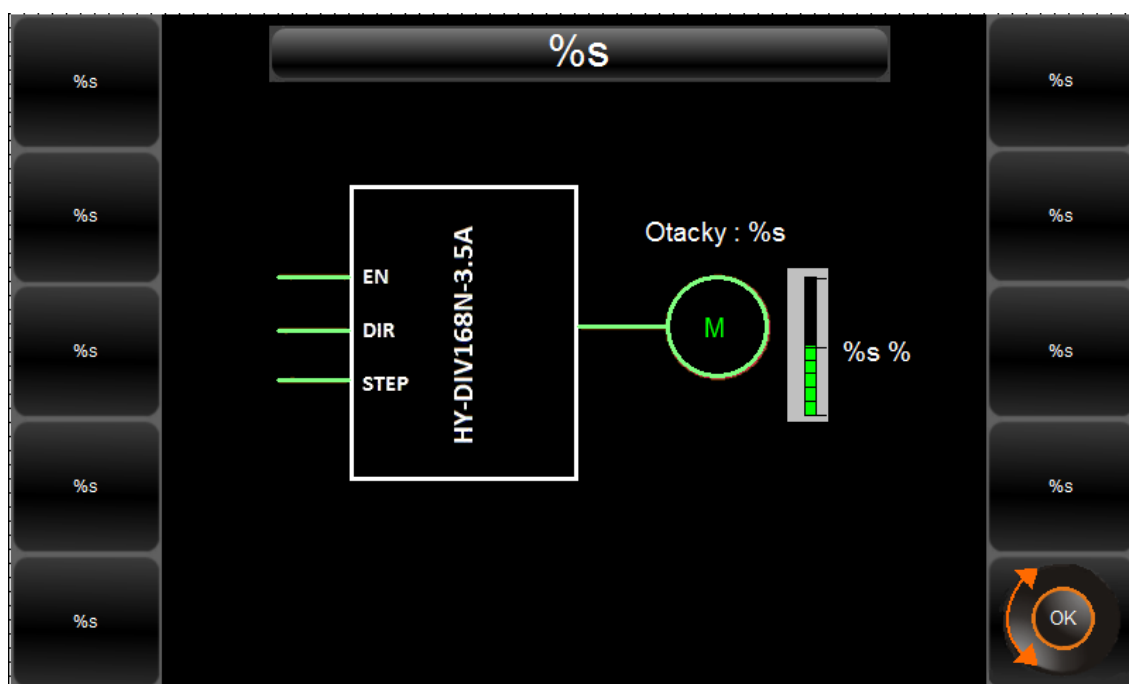
Obr. 56 - CoDeSys - CR1080 – Masterlayout

Viditelnost jednotlivých tlačítek po levé a pravé straně je ovládána pomocí proměnných hidden tlačítek nadefinovaných v globálních proměnných. Pokud je toto tlačítko v hodnotě TRUE, tak dané tlačítko je neviditelné. Texty jednotlivých tlačítky se nastavují pomocí proměnných TXT\_X. Ukázku nastavení jednoho tlačítka je vidět na Obr. 56, kde položka Invisible slouží ke skrytí tlačítka, pokud je hodnota v TRUE a položka Textdisplay slouží k vypsání textu řetězce na místo %s v textu objektu.



Obr. 57- CoDeSys - CR1080 – Masterlayout tlačítko

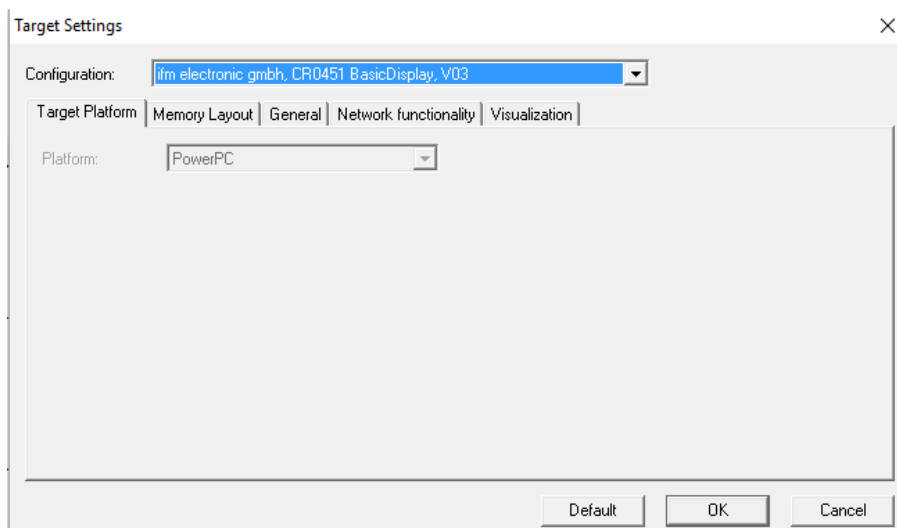
Ted' už stačí vytvořit 4 stránky vizualizace, které budou zobrazovat motor, náklon, teplotu a polohu a kameru. Každá stránka nabízí 4 tlačítka na levé straně, aby bylo snadné přepínat mezi stránkami vizualizace. Ukázku vizualizace motoru je vidět na Obr 57. Vstupní zelené přívody do driveru signalizují sepnutí signálu. Pokud je například driver vypnut, tak je vstup Enable červený. Při zapnutém vstupu Enable je možné pomocí kolečka ovládat rychlost motoru, kterou je možné sledovat na svislém posuvníku. Výstupní motor zezelená až v době, kdy se začne reálně otáčet. Otáčky motoru jsou zobrazovány nad motorem. Ukázku stránky motoru je možné vidět na Obr. 57. Postup pro ostatní stránky vizualizace je stejný.



Obr. 58 - CoDeSys - CR1080 – Ukázka vizualizace motoru

### 7.3 BasicDisplay CR0451

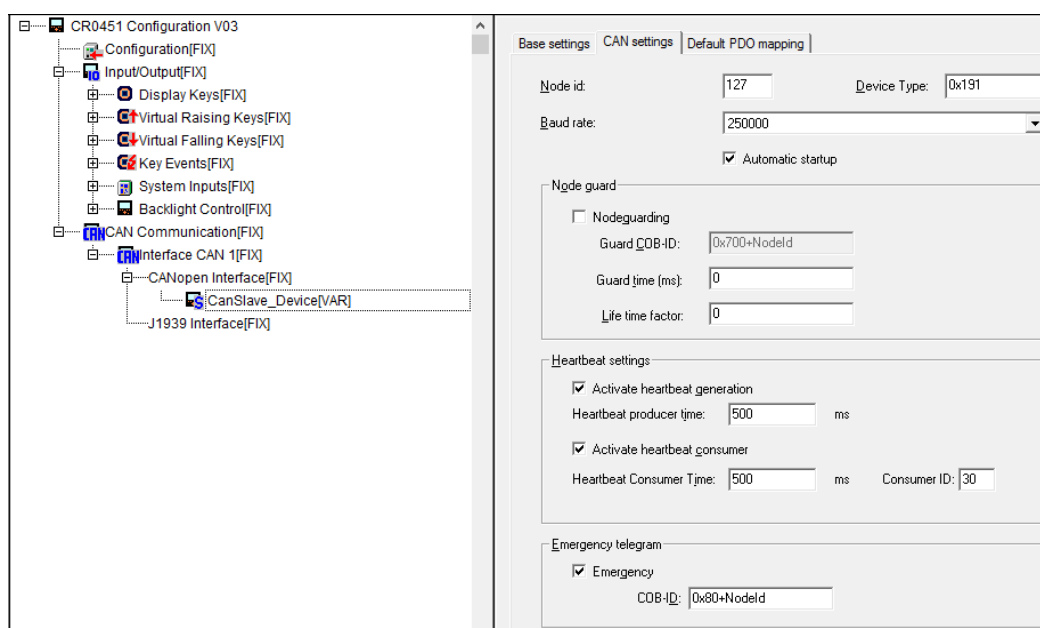
Postup nastavení tohoto zařízení je úplně stejný jako u dialogového displeje. Nejprve je zase potřeba vybrat patřičný Target Settings. V seznamu opět najdeme naše zařízení a klikneme na OK. Opět rovnou vytvoříme i hlavní program PLC\_PRG.



Obr. 59 – CoDeSys – CR0451 - Target

### 7.3.1 PLC konfigurace

Tento displej nabízí pouze jedno rozhraní CAN. Jelikož počítáme, že bude opět fungovat jako slave zařízení, tak je potřeba to nastavit, jako u dialogového displeje. Počítáme také, že zařízení bude fungovat s baud rate 250000 a node id 127. Nastavení je možné vidět na Obr. 60.



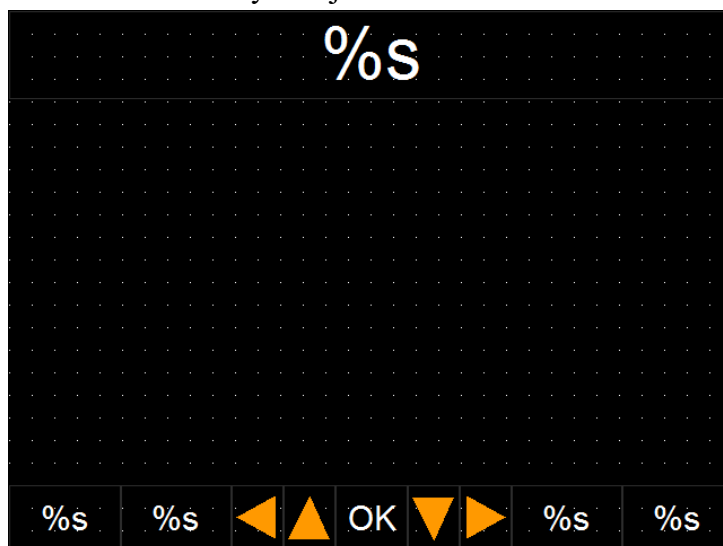
Obr. 60 - CoDeSys - CR0451 - PLC konfigurace

### 7.3.2 Knihovny, Globální proměnné a programování

Nastavení těchto věcí je úplně stejné jako u dialogového displeje. Nastavení knihoven bude stejný, jen je potřeba použít knihovnu CR0451 místo knihovny CR1080. Globální proměnné budou taktéž stejné, jelikož předávání dat po sběrnici CAN funguje s úplně stejně jako v CR1080. Jediný malý rozdíl je v tom, že BasicDisplay má jiné uspořádání tlačítek, ale postup nastavení je úplně stejný. Dokonce i postup programování jednotlivých stránek.

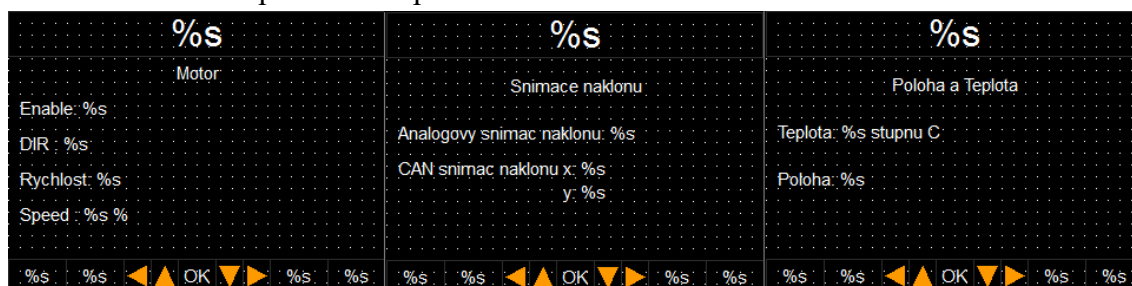
### 7.3.3 Vizualizace

Vizualizace je podobná té z dialogového displeje. Nejprve je potřeba nastavit vytvořit MasterLayout\_1, který je potom zobrazován na všech stránkách. Jelikož tento displej je mnohem menší a nabízí menší možnosti nastavení, tak i výsledná vizualizace bude jednodušší. Ukázku master layoutu je možné vidět na Obr. 61.



Obr. 61 - CoDeSys - CR0451 - Masterlayout

Každé tlačítko je opět ovládáno proměnnými hidden, kterými je možné jednotlivá tlačítka zobrazovat/skrývat. U nás bude pomocí šipek možné posouvat mezi 3 stránkami vizualizace. První stránka slouží k zobrazení důležitých dat o motoru, druhá o náklonu a třetí o poloze a teplotě. Ukázku vizualizace můžeme vidět na Obr. 62.



Obr. 62 – CoDeSys – CR0451 – Stránky vizualizace

## 8 ZÁVĚR

Tato bakalářská práce byla zaměřena na návrh a realizaci demokufu R360, kde mobilní řídicí systém R360 byl doplněn o snímače, driver a krokový motor.

Z důvodu předvedení možnosti mobilního řídicího systému R360 byl systém doplněn o standartní snímače teploty, polohy a náklonu. Jako ukázkový výstup byl použit driver s krokovým motorem. Celé zapojení je zakresleno v systému EPLAN a je k vidění v příloze 1.

V průběhu práce bylo navrženo celé zapojení demonstrační úlohy do kufru. Demonstrační úloha byla zapojena a naprogramována. V poslední fázi byla demonstrační úloha otestována a zabudována do kufru.

Programování a vizualizace byly provedeny ve vývojovém prostředí CoDeSys. Vizualizace umožňuje sledování hodnot snímačů a ovládání krokového motoru. Ovládání motoru je také možné z klávesnicového modulu. Všechny soubory vytvořené během programování se nachází v příloze 2.

# Literatura

- [1] Profil firmy – ifm electronic, spol. s r.o.. ABC. [online]. [cit. 2016-01-03]. Dostupné z: <http://www.abc.cz/firma/ifm-electronic-senzory/>
- [2] Historie protokolu CAN. [online]. [cit. 2016-05-22]. Dostupné z: [http://simple\\_can\\_node.sweb.cz/historie.htm](http://simple_can_node.sweb.cz/historie.htm)
- [3] Systém manual Basic controler – ifm electronic, spol. s r.o. [online]. [cit. 2016-05-22]. Dostupné z: <https://www.ifm.com/mounting/7390709UK.pdf>
- [4] Společnost. *EPLAN*. [online]. [cit. 2016-05-22]. Dostupný na WWW: <http://www.eplan.cz/cz/spolecnost/>
- [5] Logo, *EPLAN*. [online]. [cit. 2016-05-22]. Dostupný na WWW: [http://www.navachot.com/images/logo\\_eplan.jpg](http://www.navachot.com/images/logo_eplan.jpg)
- [6] CoDeSys. [online]. [cit. 2016-05-22]. Dostupný na WWW: <http://automatizace.hw.cz/programovaci-rezimy-pro-plc-dle-iec-611313-codesys>
- [7] Logo, *CoDeSys*. [online]. [cit. 2016-05-22]. Dostupný na WWW: [http://automatizace.hw.cz/files/images/image/htuk\\_codesys\\_CoDeSys30\\_Logo.jpg](http://automatizace.hw.cz/files/images/image/htuk_codesys_CoDeSys30_Logo.jpg)
- [8] CFC. [online]. [cit. 2016-05-22]. Dostupný na WWW: [http://automatizace.hw.cz/files/images/image/smallrezim\\_CFC2.png](http://automatizace.hw.cz/files/images/image/smallrezim_CFC2.png)
- [9] FBD. [online]. [cit. 2016-05-22]. Dostupný na WWW: [http://automatizace.hw.cz/files/images/image/smallrezim\\_FBD2.png](http://automatizace.hw.cz/files/images/image/smallrezim_FBD2.png)
- [10] SFC. [online]. [cit. 2016-05-22]. Dostupný na WWW: [http://automatizace.hw.cz/files/images/image/smallrezim\\_SFC2.png](http://automatizace.hw.cz/files/images/image/smallrezim_SFC2.png)
- [11] CR0403. ifm electronic. [online]. [cit. 2016-05-22]. Dostupný na WWW: <http://www.ifm.com/mounting/80010113UK.pdf>
- [12] AUTOR NEUVEDEN. ifm electronic [online]. [cit. 22.5.2016]. Dostupný na WWW: [http://www.ifm.com/tedo/foto/e100\\_0075.gif](http://www.ifm.com/tedo/foto/e100_0075.gif)
- [13] CR0421. ifm electronic. [online]. [cit. 2016-05-22]. Dostupný na WWW: <http://www.ifm.com/mounting/80010113UK.pdf>
- [14] AUTOR NEUVEDEN. ifm electronic [online]. [cit. 22.5.2016]. Dostupný na WWW: [http://www.ifm.com/tedo/foto/e100\\_0076.gif](http://www.ifm.com/tedo/foto/e100_0076.gif)
- [15] CR0451. ifm electronic. [online]. [cit. 2016-05-22]. Dostupný na WWW: <http://www.ifm.com/mounting/7390823UK.pdf>
- [16] AUTOR NEUVEDEN. ifm electronic [online]. [cit. 22.5.2016]. Dostupný na WWW: [http://www.ifm.com/tedo/foto/e100\\_0077.gif](http://www.ifm.com/tedo/foto/e100_0077.gif)
- [17] CR1500. ifm electronic. [online]. [cit. 2016-05-22]. Dostupný na WWW: <http://www.ifm.com/mounting/7390289DEUK.pdf>
- [18] AUTOR NEUVEDEN. ifm electronic [online]. [cit. 22.5.2016]. Dostupný na WWW: [http://www.ifm.com/tedo/foto/e100\\_0010.gif](http://www.ifm.com/tedo/foto/e100_0010.gif)
- [19] EC2112. ifm electronic. [online]. [cit. 2016-05-22]. Dostupný na WWW: <http://www.ifm.com/mounting/7390837UK.pdf>
- [20] AUTOR NEUVEDEN. ifm electronic [online]. [cit. 22.5.2016]. Dostupný na WWW: [http://www.ifm.com/tedo/foto/e100\\_0074.gif](http://www.ifm.com/tedo/foto/e100_0074.gif)
- [21] CR1080. ifm electronic. [online]. [cit. 2016-05-22]. Dostupný na WWW: <http://www.ifm.com/mounting/7390965UK.pdf>

- [22] AUTOR NEUVEDEN. ifm electronic [online]. [cit. 22.5.2016]. Dostupný na WWW:  
[http://www.ifm.com/tedo/foto/e100\\_0079.gif](http://www.ifm.com/tedo/foto/e100_0079.gif)
- [23] O2M113. ifm electronic. [online]. [cit. 2016-05-22]. Dostupný na WWW:  
<http://www.ifm.com/mounting/7390810UK.pdf>
- [24] AUTOR NEUVEDEN. ifm electronic [online]. [cit. 22.5.2016]. Dostupný na WWW:  
[http://www.ifm.com/tedo/foto/e100\\_0051.gif](http://www.ifm.com/tedo/foto/e100_0051.gif)
- [25] CR2102. ifm electronic. [online]. [cit. 2016-05-22]. Dostupný na WWW:  
<http://www.ifm.com/mounting/7390398DEUK.pdf>
- [26] AUTOR NEUVEDEN. ifm electronic [online]. [cit. 22.5.2016]. Dostupný na WWW:  
[http://www.ifm.com/tedo/foto/e100\\_0018.gif](http://www.ifm.com/tedo/foto/e100_0018.gif)
- [27] EC2082. ifm electronic. [online]. [cit. 2016-05-22]. Dostupný na WWW:  
<http://www.ifm.com/mounting/7390445DEUK.pdf>
- [28] AUTOR NEUVEDEN. ifm electronic [online]. [cit. 22.5.2016]. Dostupný na WWW:  
[http://www.ifm.com/tedo/foto/e100\\_0020.gif](http://www.ifm.com/tedo/foto/e100_0020.gif)
- [29] DN4013. ifm electronic. [online]. [cit. 2016-05-22]. Dostupný na WWW:  
<http://www.ifm.com/mounting/7390980DEESFRITPTUK.pdf>
- [30] AUTOR NEUVEDEN. ifm electronic [online]. [cit. 22.5.2016]. Dostupný na WWW:  
[http://www.ifm.com/tedo/foto/e200\\_0077.gif](http://www.ifm.com/tedo/foto/e200_0077.gif)
- [31] DX2002. ifm electronic. [online]. [cit. 2016-05-22]. Dostupný na WWW:  
<http://www.ifm.com/mounting/7390980DEESFRITPTUK.pdf>
- [32] AUTOR NEUVEDEN. ifm electronic [online]. [cit. 22.5.2016]. Dostupný na WWW:  
[http://www.ifm.com/tedo/foto/e200\\_0004.gif](http://www.ifm.com/tedo/foto/e200_0004.gif)
- [33] RO6343. ifm electronic. [online]. [cit. 2016-05-22]. Dostupný na WWW:  
<http://www.ifm.com/mounting/701858DEFUK.pdf>
- [34] AUTOR NEUVEDEN. ifm electronic [online]. [cit. 22.5.2016]. Dostupný na WWW:  
[http://www.ifm.com/tedo/foto/400\\_0006.gif](http://www.ifm.com/tedo/foto/400_0006.gif)
- [35] AUTOR NEUVEDEN. ifm electronic [online]. [cit. 22.5.2016]. Dostupný na WWW:  
[http://www.ifm.com/tedo/foto/600\\_0066.gif](http://www.ifm.com/tedo/foto/600_0066.gif)
- [36] IGC213. ifm electronic. [online]. [cit. 2016-05-22]. Dostupný na WWW:  
<http://www.ifm.com/mounting/1D2226UK.pdf>
- [37] AUTOR NEUVEDEN. ifm electronic [online]. [cit. 22.5.2016]. Dostupný na WWW:  
[http://www.ifm.com/tedo/foto/100\\_0172.gif](http://www.ifm.com/tedo/foto/100_0172.gif)
- [38] AUTOR NEUVEDEN. ifm electronic [online]. [cit. 22.5.2016]. Dostupný na WWW:  
[http://www.soonwell.com/ebay\\_file/pretty-pic/siyouka-5.JPG](http://www.soonwell.com/ebay_file/pretty-pic/siyouka-5.JPG)
- [39] AUTOR NEUVEDEN. ifm electronic [online]. [cit. 22.5.2016]. Dostupný na WWW:  
<https://cnc.inshop.cz/inshop/catalogue/products/pictures/35T36-1004B.jpg>
- [40] AUTOR NEUVEDEN. ifm electronic [online]. [cit. 22.5.2016]. Dostupný na WWW:  
<http://www.all4.cz/deploy/files/1140984.jpg>

## **Seznam příloh**

Příloha 1. Schéma zapojení

Příloha 2. DVD