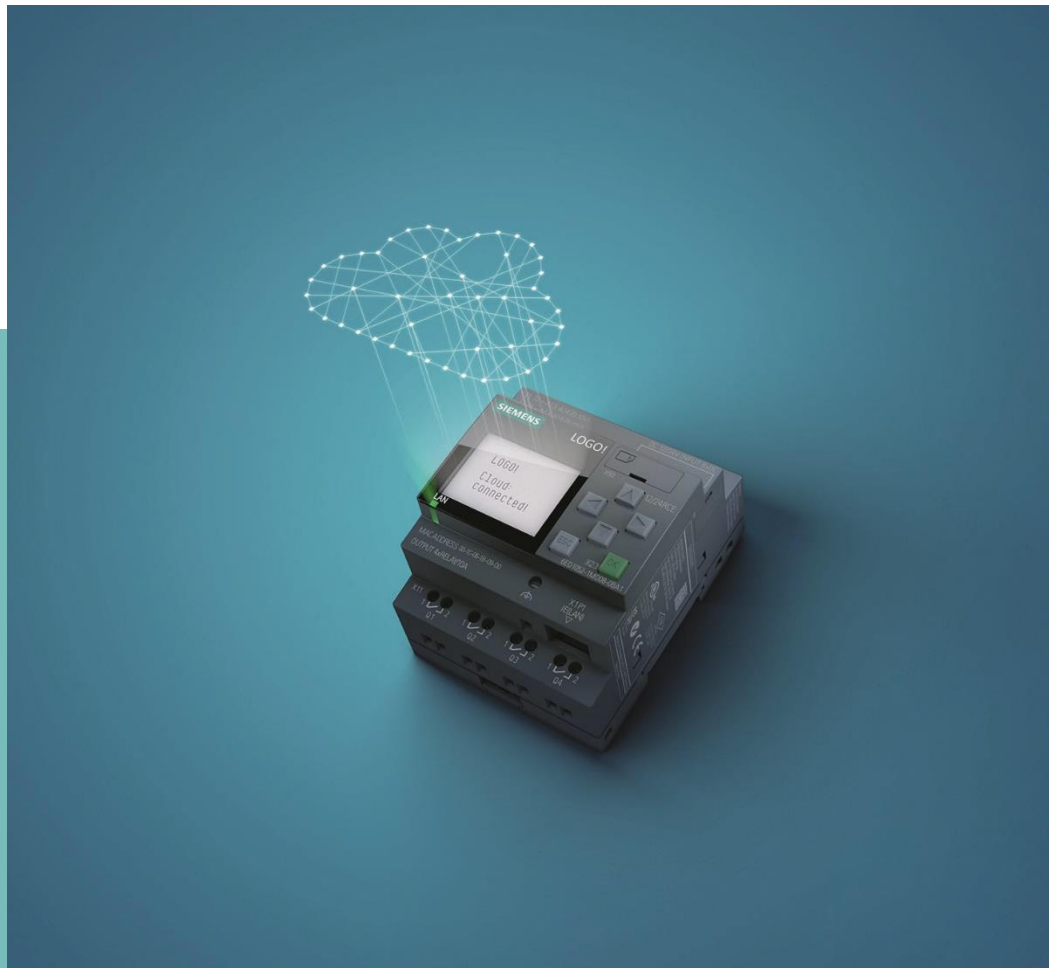


Krok za krokem III



Autor: Ing. Josef Havel

ÚVOD	5
POSTUPY	5
LOGICKÉ FUNKCE	6
Zapojení stejného příkladu pomocí bloků FBD.....	17
ASYNCHRONNÍ MOTORY.....	18
Spouštění hvězda trojúhelník v programu LAD.....	20
Reverzace asynchronního motoru varianta1	21
Popis funkce zapojených bloků.....	22
Reverzace asynchronního motoru varianta 2	23
Reverzace asynchronního motoru v programu LAD	24
Přímočarý vratný pohyb.....	25
Přímočarý vratný pohyb s prodlevou na konci a příčným posuvem.....	26
Cyklická změna směru otáčení.....	26
.....	26
Změna směru otáčení po nastavené době s prodlevou.....	27
Změna směru otáček po nastaveném počtu otáček s prodlevou při změně směru otáček	28
STROJNÍ LINKY.....	30
Spínání a vypínání tlačítka pomocí bloků "Zpožděného zapnutí" a "RS" s různou délkou postupného spínání	30
Spínání výstupů pomocí bloků "Zpožděného zapnutí" s různou délkou postupného spínání	31
Postupné spínání se stejnými intervaly pomocí dvou Posuvných registrů.....	33
Postupné spínání a vypínání	34
Spínání a vypínání tlačítka pomocí bloků "Zpožděného zapnutí" a "RS" s různou délkou postupného spínání s bloky LAD	36
Příklad linky složený z pěti strojů	37
Zapojení bloků FBD	38
Skutečné zapojení uvedeného příkladu	40
VYPÍNAČE.....	41
Křížový přepínač – řazení č.7	42
Schodišťový automat	48
Schodišťový spínač.....	49
KROKOVÝ POSUN.....	52
Časové krokování s použitím bloku "Zpožděné vypnutí".....	53
Časové krokování s použitím bloku "Zpožděné zapnutí".....	54
Časové krokování s použitím " posuvného registru" a "Asynchronního generátoru"	55
Časové krokování s použitím " Posuvného registru" a "Zpožděného zapnutí".....	58
Krokování impulzem s použitím " Čítače" a "relé RS"	59
NEPŘETRŽITĚ OPAKUJÍCÍ SE POSTUP (Cyklický).....	62

Automatický cyklický posuv pomocí asynchronního pulzního generátoru.....	65
Cyklické spínání výstupů pomocí " Hranou spouštěného impulzního relé"	66
ANALOGOVÉ BLOKY	68
Nastavení analogového zesilovače.....	69
Chlazení	71
ZMĚNA PARAMETRŮ	72
Ovládání klikáním na tlačítka	72
Nastavení textu zpráv	73
ARITMETICKÉ ÚKONY.....	75
Odečítání.....	75
Sčítání	77
Násobení.....	78
Dělení.....	79
Přičítání konstantního čísla	80
Odečítání konstantního čísla.....	81
Výpočet kombinovaných aritmetických výpočtů.....	81
PŘÍKLADY.....	88
ŠKOLNÍ ZVONEK	89
KOMPRESOR	91
SPÍNÁNÍ HLADINY.....	92
Napouštění nádrže.....	92
Řešení pomocí základních funkcí – logických bloků	92
Řešení pomocí bloku RS.....	94
Řešení pomocí bloků LAD.....	95
Vyprazdňování nádrže	96
PARKOVIŠTĚ	98
Program	98
SEMAFOR.....	103
Příklad: Semafor 1.....	103
Zapojení bloků	104
Příklad: Semafor2.....	110
Příklad: Semafor 3	114
SEMAFOR 4.....	116
Blok UDF	119
Panel HMI	120
Princip funkce:	124
VYTÁPĚNÁ BOUDA PRO KOCOURA	131
Program	132

Automatická regulace	133
OVLÁDÁNÍ DVOU PŘÍSTROJŮ LOGO JEDNÍM PANELEM HMI	136
Vytvoření programu v LOGU	136
Nastavení panelu HMI.....	139
Naprogramování panelu HMI	141
Změna parametrů na panelu HMI při ovládání dvou přístrojů LOGO.....	142
Program LOGO.....	143
Program v LOGU	144
Nastavení v panelu HMI.....	145
Pasterizace.....	147
Část programu topení.....	148
Vypnutí pasterizace	151
Ovládání jímky a zavlažování pomocí LOGO! a Node-Red	152
Nastavení Raspberry a Node-Red	152
Nastavení LOGO! A Logo Soft	158
Program	159
Ovládání jímky	160
Tlačítko STOP	161
Ovládání shromažďovací jámy	162

ÚVOD

Záměrem bylo vytvořit návody LOGO pro ty, kteří s programováním LOGA začínají, zejména žáci středních a učňovských škol obor elektro. Tento díl jsem rozdělil do dvou částí. První jsem nazval „Postupy“ a druhou „Příklady“. Část „Postupy“ obsahují krátké příklady zaměřené na jednotlivé typy úloh se kterými se můžeme v praxi setkat. Např. topení, chlazení, postupné spouštění strojní linky, spínání hladiny, aritmetické výpočty, změny parametrů pomocí tlačítek např. nastavení teploty atd. Úlohy, které máme naprogramovat budou pokaždé jiné, a zvláště pokud začínáme s programováním, budeme hledat, jak ten dílčí problém vyřešit, a v tom si myslím mohou moci postupy uvedené v tomto díle. U každého typu úloh je několik řešení, ale nic nebrání tomu, aby to programátor udělal zcela jinak.

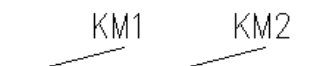
Pokud se programy mají realizovat v praxi je vždy důležité posoudit bezpečnost celého systému. Výrobce je povinen před uvedením výrobku na trh nebo do provozu, provést ověření shody a vydat EU prohlášení o shodě. To platí i pro rozváděč, jehož součástí je LOGO. Jeho náležitosti jsou stanoveny v NV č.118/2016 Sb. Pro strojní zařízení najdeme požadavky na technické požadavky a postupy posuzování shody v NV 176/2008 Sb. Za zvlášť nebezpečné je třeba považovat i automaticky řízené strojní zařízení. Obsluha nemusí rozeznat, zda část linky stojí z důvodu poruchy. např. ucpáním, nebo se jedná o technologickou prodlevu. I na to, aby poruchy byly signalizovány je nutné při programování pamatovat. Budete-li mít nějaké nápady s LOGEM, budu rád, když se o ně se mnou podělíte.
Josef Havel, revize@havelnet.cz

POSTUPY

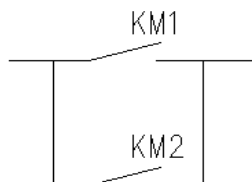
LOGICKÉ FUNKCE

V této části příkladů jde o převedení slovní formulace zadání do podoby algebry logiky. Základními funkcemi logiky jsou logický součin a logický součet a negace. V elektrotechnice logický součin představuje dva sériově řazené kontakty. Logický součet znamená dva paralelně řazené kontakty.

Logický součin $KM1 \cdot KM2$



Logický součet $KM1 + KM2$



V následujícím textu je negovaný kontakt označený čárkou: $KM1'$, a' , $M1'$... X'

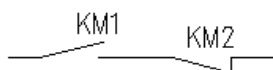
Logický součin negovaných kontaktů $KM1' \cdot KM2'$



Při převodu slovního vyjádření do algebry logiky je použití spojky "a" nebo "i", chápáno jako logický součin. Příklad: Stykač 1 sepne motor M1, bude-li sepnut stykač motoru M2 a současně stykač motoru M3. Zápis v logické algebře zní: $M1 = M2 \cdot M3$

Příklad: Motor M1 bude spuštěn, buď-li spuštěn motor M2, nebo motor M3. Zápis je: $M1 = M2 + M3$ Příklad: Motor M1 bude spuštěn, buď-li spuštěn motor M2 a bude vypnutý motor M3. Zápis je:

$$M1 = M2 \cdot M3'$$



V logice se setkáme s následujícími funkcemi

A	B	OR	NOR	AND	NAND	XOR
0	0	0	1	0	1	0
0	1	1	0	0	1	1
1	0	1	0	0	1	1
1	1	1	0	1	0	0

Prvotní zápisy v logické algebře lze zjednodušit podle následujících pravidel. Účelem je zjednodušit zapojení, při zachování stejné funkce.

Pravidla zjednodušování

Sériové řazení

$$aa = a$$

$$a' a' = a'$$

$$a \cdot 0 = 0$$

$$a \cdot 1 = a$$

$$aa' = 0$$

Při součinu nahradíme kontakty téhož prvku, jež jsou ve stejném tvaru jako sériově předřazený kontakt, **hodnotou 1**, a kontakty, jež jsou v opačném tvaru, **hodnotou 0**.

$$\begin{array}{ll} Q = a(ab + a'c) & Q = a'(ab + a'c) \\ Q = a(1b + 0c) & Q = a'(0b + 1c) \\ Q = ab & Q = a'c \end{array}$$

Příklad 1:

$$Q = a(ab + ac + b'c) + c + (ac + a'c)$$

rovnice se rozdělí na dvě skupiny (s předřazeným **a** a předř. **c**)

$$Q = [a(ab + ac + b'c)] + [c + (ac + a'c)]$$

$$Q = a(1b + 1c + b'c) + c + (a0 + a'1)$$

$$Q = a(b + c + b'c + c + a')$$

$c+c=c$

$$Q = a(b + c + b'c + a') = a(b+(c+b'c+ a'))=$$

$$Q = a(b+c +c.1 + a') = a(b+c +0) =a(b+c)$$

Paralelní řazení

$$\begin{array}{lll} a + a = a & a + 0 = a & a + a' = 1 \\ a' + a' = a' & + 1 = 1 & \end{array}$$

Při součtu nahradíme kontakty téhož prvku, jež jsou ve stejném tvaru jako paralelně přiřazený kontakt, **hodnotou 0**, a kontakty, jež jsou v opačném tvaru, **hodnotou 1**

$$\begin{array}{ll} Q = a + (ab + a'c) & Q = a' + (ab + a'c) \\ Q = a + (0b + 1c) & Q = a' + (1b + 0c) \\ Q = a+c & Q = a'+b \end{array}$$

Příklad 2 :

$$Q = a1 + a1(a2+a3').a5 + a4(a3 . a5 + a1 . a5 + a1 . a2)$$

$$Q = a1 + [0(a2 + a3').a5 + a4(a3 . a5 + 0 . a5 + 0 . a2)]$$

$$Q = a1 + a4 . a3 .a5$$

Příklad 3:

$$Q = abcd + a'bcd + abc'd + a'b c'd$$

$$Q = bd (ac + a'c + ac' + a'c')$$

$$Q = bd ((c (a+a') + c'(a+a'))$$

$$Q = bd (c + c')$$

$$Q = bd$$

Příklad 4:

Do stroje "M1" se dodává materiál třemi dopravníky M2, M3, M4. Stroj M1 bude spuštěn, pokud budou spuštěny alespoň dva ze tří dopravníků. Dopravníky i stroj jsou spouštěny stykač KM1, KM2, KM3, KM4, jejichž ovládání je dvoupolohovými spínači. Stykač stroje KM1 nelze spínačem I1 spustit, pokud není splněna výše uvedená podmínka.

Zapsané zadání do textu, ze kterého je možné sestavit výchozí vztah pro zjednodušení.

Stykač KM 1 bude sepnut, je-li zapnut stykač KM2 a KM3 nebo KM2 a KM4 nebo KM3 a KM4, nebo KM2 a KM3 a KM4.

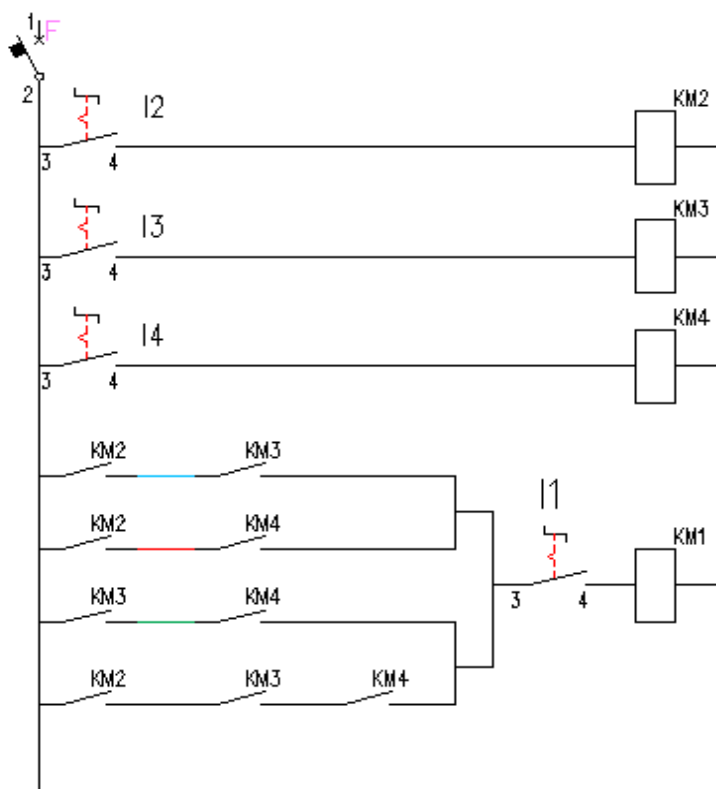
$$KM1 = KM2 \cdot KM3 + KM2 \cdot KM4 + KM3 \cdot KM4 + KM2 \cdot KM3 \cdot KM4$$

Kombinace, kdy je stroj možné spustit zapíšeme do tabulky

kombinace	KM2	KM3	KM4	KM1
1.	1	1	0	1
2.	1	0	1	1
3.	0	1	1	1
4.	1	1	1	1

Liniové schéma původního vztahu před úpravami pro zjednodušení

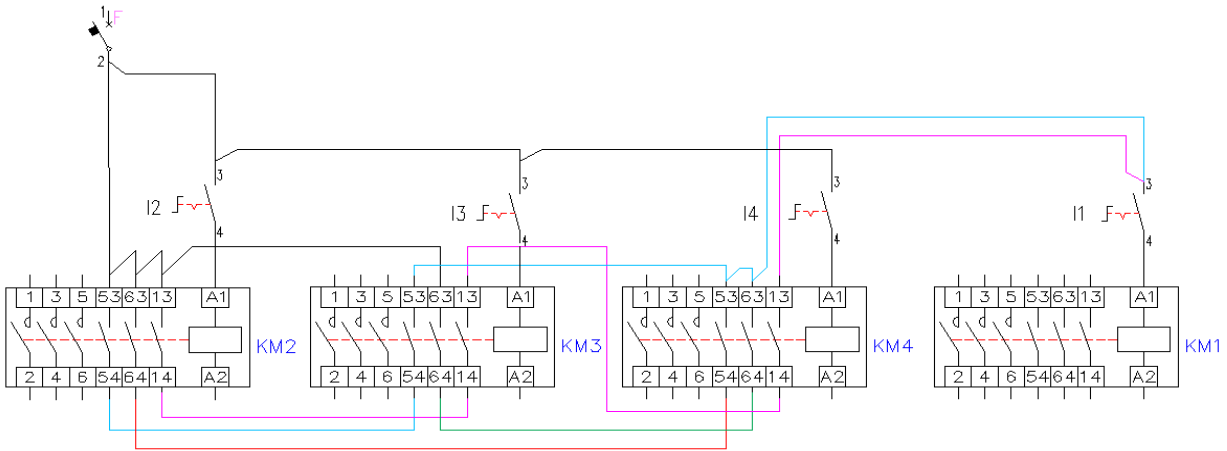
obr.1



Zapojení ovládání stykačů před zjednodušením, které odpovídá vztahu :

$$KM1 = KM2 \cdot KM3 + KM2 \cdot KM4 + KM3 \cdot KM4 + KM2 \cdot KM3 \cdot KM4$$

obr.2



Postup zjednodušování:

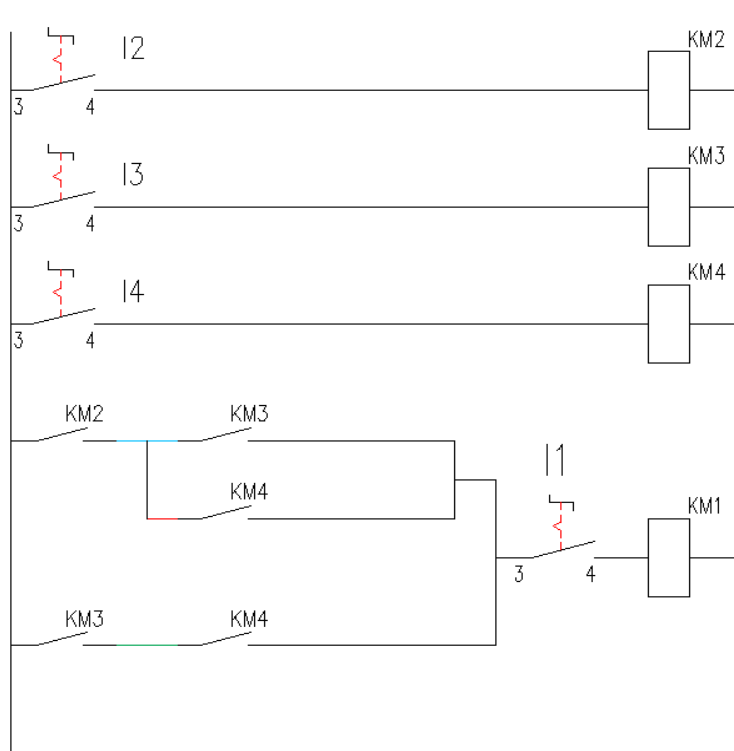
$$KM1 = (KM2 \cdot KM3 + KM2 \cdot KM4) + (KM3 \cdot KM4 + KM2 \cdot KM3 \cdot$$

$$KM4) \quad KM1 = KM2(KM3 + KM4) + KM3 \cdot KM4(1 + KM2)$$

$$KM1 = KM2(KM3 + KM4) + KM3 \cdot KM4$$

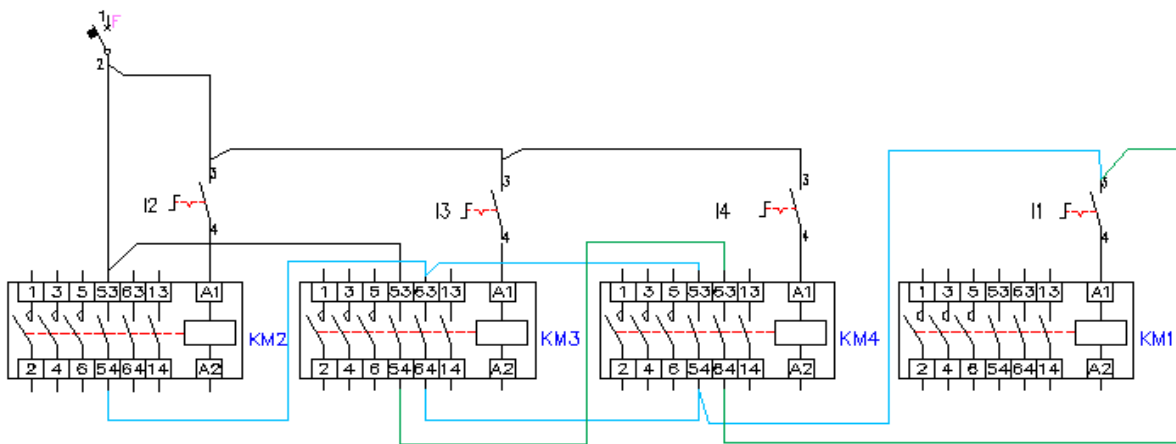
Liniové schéma zjednodušeného stavu: $KM1 = KM2(KM3 + KM4) + KM3 \cdot KM4$

obr.3



Skutečné zapojení ovládní stykačů podle zapojení liniového schématu zjednodušeného stavu

obr.4



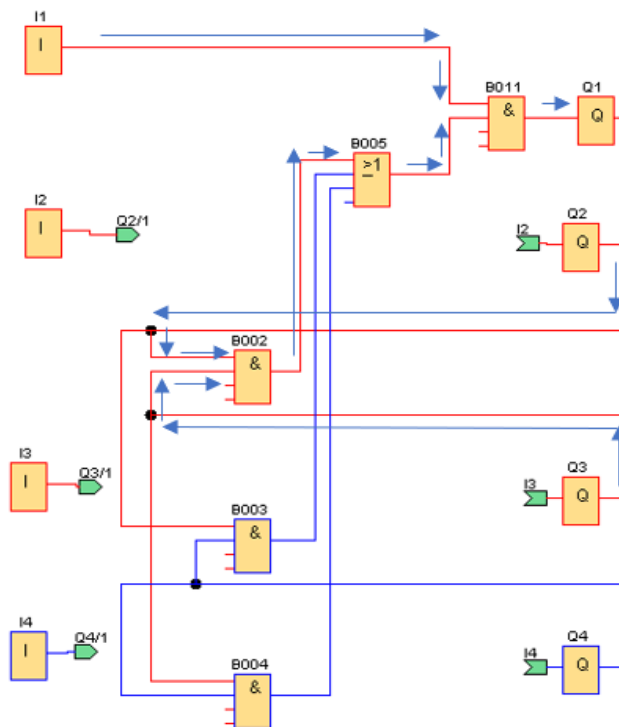
Program v LOGU podle obr.1 .

Na obrázku jsou čtyři spínače I1, I2, I3, I4 . Cívky stykačů KM představují výstupy Q.

kombinace	Q2	Q3	Q4	I1	Q1
1.	1	1	0	1	1
2.	1	0	1	1	1
3.	0	1	1	1	1
4.	1	1	1	1	1

$$Q1 = (Q2 \cdot Q3 + Q2 \cdot Q4 + Q3 \cdot Q4 + Q2 \cdot Q3 \cdot Q4) \cdot I1 \text{ sepnutí 1. kombinace } Q1 = (Q2 \cdot Q3) \cdot I1$$

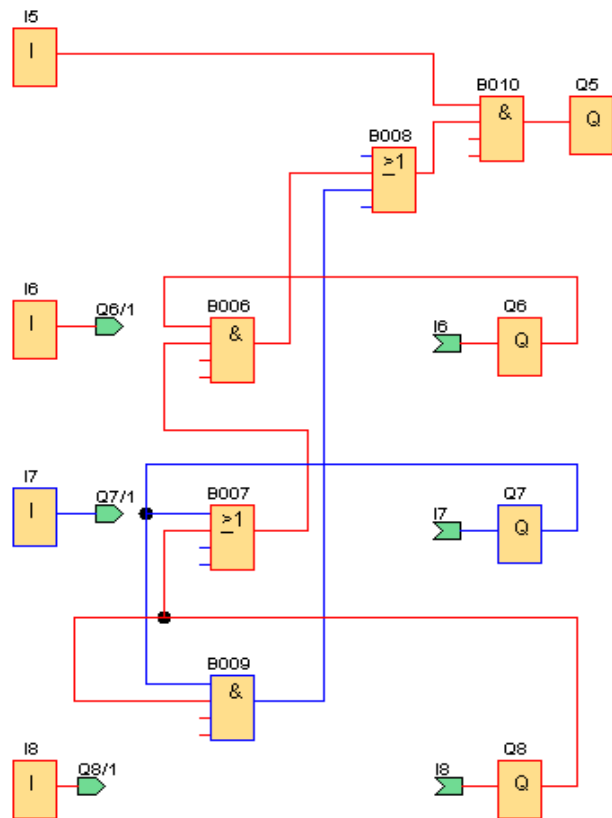
Logický součin je dán blokem AND a logický součet blokem OR. Součin $Q2 \cdot Q3$ představuje blok B002 součin závorka a I1 představuje blok B011



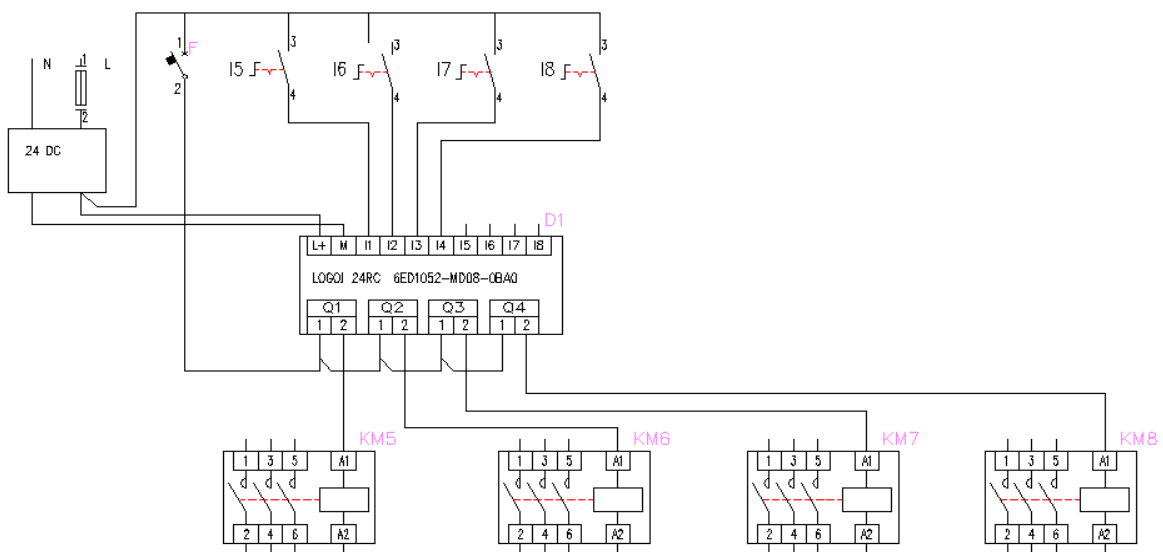
Program v LOGU podle
obr.3

$KM1 = KM2(KM3 + KM4) + KM3 \cdot KM4$ Tento vztah po doplnění o spínač I5 a náhradě KM za Q: $Q5 = ((Q6 (Q7 + Q8) + Q7 \cdot Q8) \cdot I1)$

Sepnutý je vstup I5, I6, I8




Skutečné zapojení LOGA, spínačů a stykačů

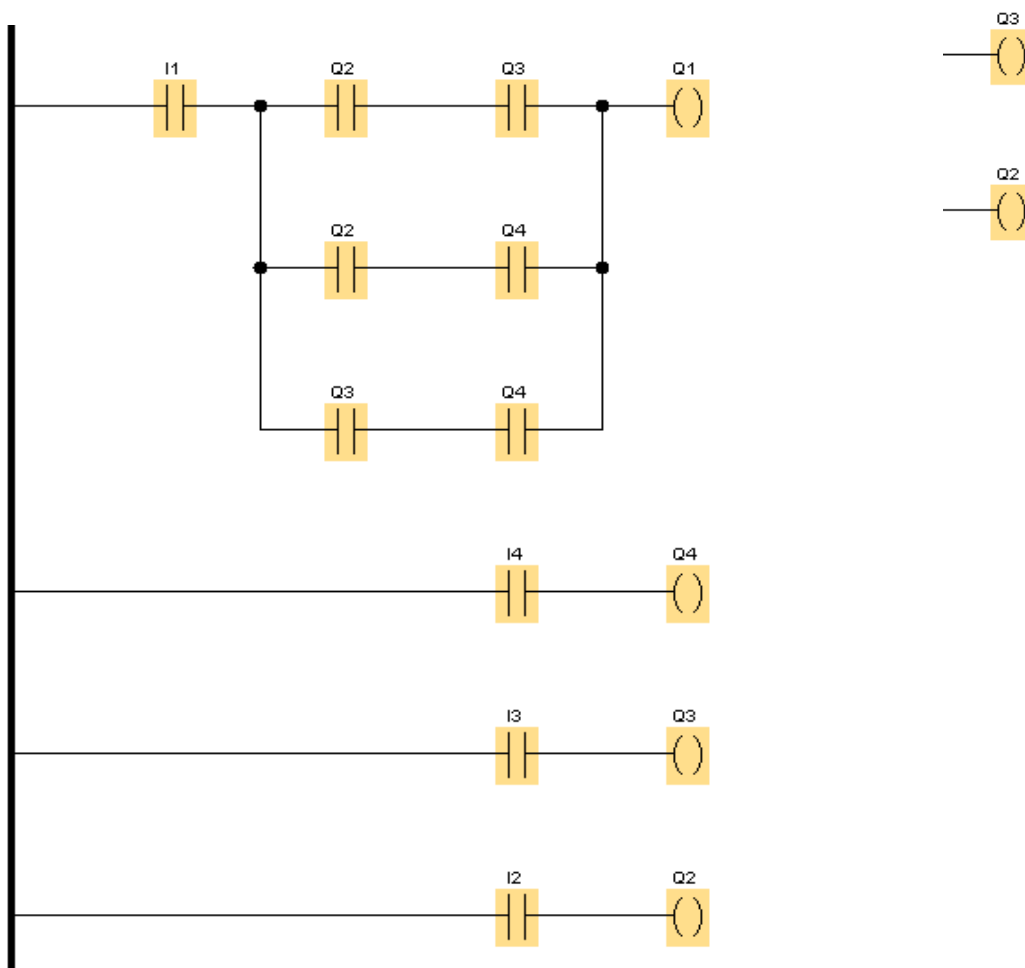


Na první pohled je patrné, že zapojení s LOGEM je jednodušší a přehlednější než zapojení pomocí propojování kontaktů stykačů. U standardního zapojení nestačí mnohdy základní kontakty stykačů a musí se použít přídatné nasazené na stykač. Pokud se použije klasické zapojení je výhodné pomocí úprav v logické algebře provést zjednodušení. Tím se sníží počet potřebných kontaktů, jak je vidět na obrázku 4.

Čistě logické příklady může být jednodušší a přehlednější naprogramovat v diagramu kontaktních schémat LAD. Příklad podle obr.1

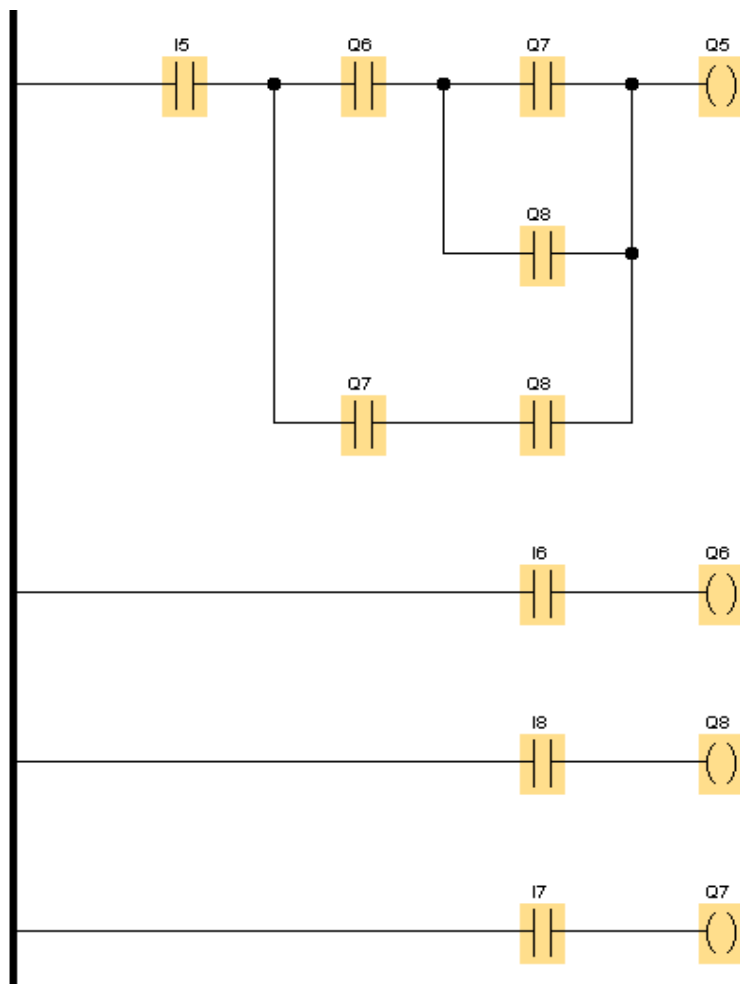
$$Q1 = (Q2 \cdot Q3 + Q2 \cdot Q4 + Q3 \cdot Q4 + Q2 \cdot Q3 \cdot Q4) \cdot I1$$

Q2, Q3  jsou pomocné kontakty, stejně označených výstupů Q2, Q3 (cívek stykačů)



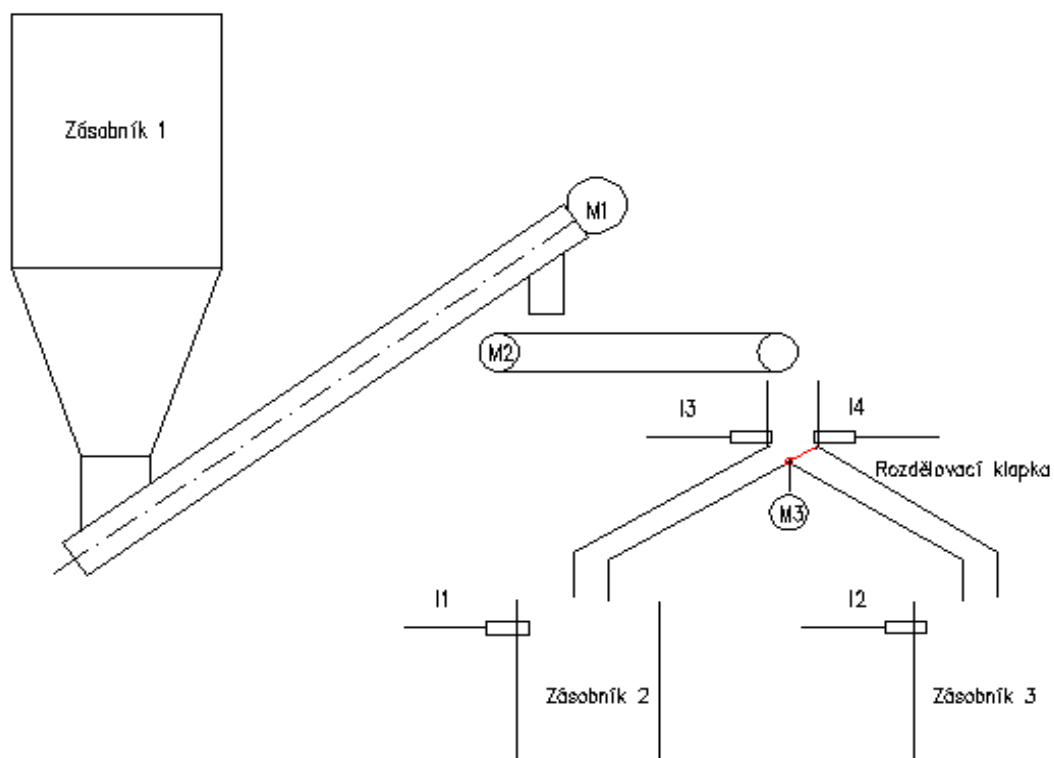
Příklad podle obr. 3 - Zjednodušení předchozího příkladu

$$Q5 = ((Q6 (Q7 + Q8) + Q7 \cdot Q8) \cdot I1)$$



Plnění zásobníků

Ze zásobníku 1 jde dopravovaný materiál šnekovým dopravníkem poháněným motorem M1 na pásový dopravník M2 a z něj se sype do dvou zásobníků. Zda se sype do zásobníku 2 nebo 3 závisí na poloze rozdělovací klapky, která je ovládána servomotorem M3. Vypnutí motoru M3 v koncových polohách zajišťuje čidlo "I3" a "I4". Zaplnění zásobníku 2 je signalizováno čidlem "I1" a zaplnění zásobníku 3 čidlem I2. Všechny motory jsou zapínány stykači. Motor M3 je ovládán dvěma stykači z důvodu změny směru otáček. Motory M1 a M2 jsou vypnuty při zaplnění obou násypků současně. Na obrázku je poloha klapky při plnění zásobníku Z2. Po jeho zaplnění se klapka přesune do polohy k čidlu "I3".



Po spuštění plnicí linky se nejprve rozběhne motor M2 a současně motor M1, jejich spuštění je však podmíněno tím, že oba zásobníky nesmí být zaplněny. Po zaplnění jednoho zásobníku se přepne klapka do druhé polohy. Jsou-li oba zásobníky plné linka se zastaví. Klapka je ovládána motorem s převodovkou. Po dosažení koncové polohy klapky je motor vypnut koncovým čidlem.

Motor M1 je zapnutý, je-li zapnutý motor M2

Dopravník Q2 je spuštěn, není-li zásobník Z2 zaplněn a zásobník Z3 je zaplněn, nebo zásobník Z2 je zaplněn a zásobník Z3 není zaplněn, nebo zásobník Z2 a Z3 nejsou zaplněny. To vyjádříme logickou rovnicí: $I1 \cdot I2' + I1' \cdot I2 + I1 \cdot I2 = Q2$, tu upravíme na tvar $I1 \cdot (I2' + I2) + I1' \cdot I2 = Q2$, protože $I2' + I2 = 1$, potom výsledný tvar po úpravě bude:

$$I1 + I1' \cdot I2 = Q2$$

Motor 1 dopravníku bude spuštěn spustí-li se motor M2:

$$Q2 = Q1$$

Klapka v poloze vpravo pro plnění zásobníku 2, bude v případě budou-li oba zásobníky prázdné nebo bude-li zásobník Z2 prázdný a Z3 plný, nebo budou oba plné.

$$I1 \cdot I2 + I1 \cdot I2' + I1' \cdot I2' = Q4$$

Klapka v poloze (vpravo) při plnění zásobníku Z2

$$I1 \cdot I2 + I2' \cdot (I1 + I1') = Q4 \quad \text{postupná úprava rovnice}$$

$$I1 \cdot I2 + I2' = Q4 \quad \text{výsledný zjednodušený tvar rovnice}$$

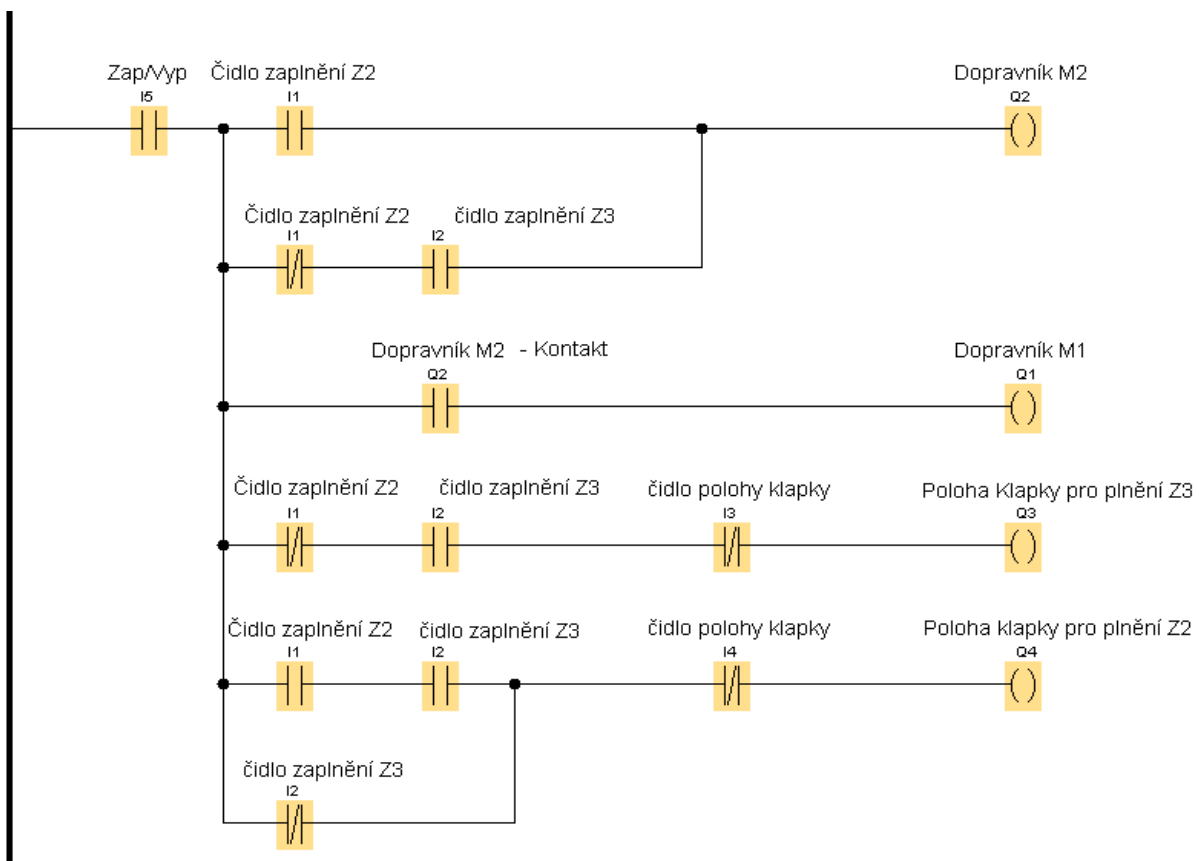
Klapka v poloze pro plnění dopravníku Z3 bude za předpokladu, že je druhý zásobník zaplněn a zásobník Z3 je prázdný

$$I1' \cdot I2 = Q3 \quad \text{klapka v poloze (vlevo) při plnění zásobníku Z3}$$

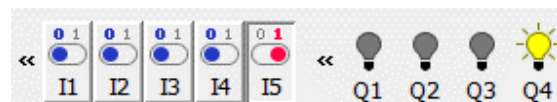
Výsledné rovnice převedeme do diagramu kontaktních schémat.

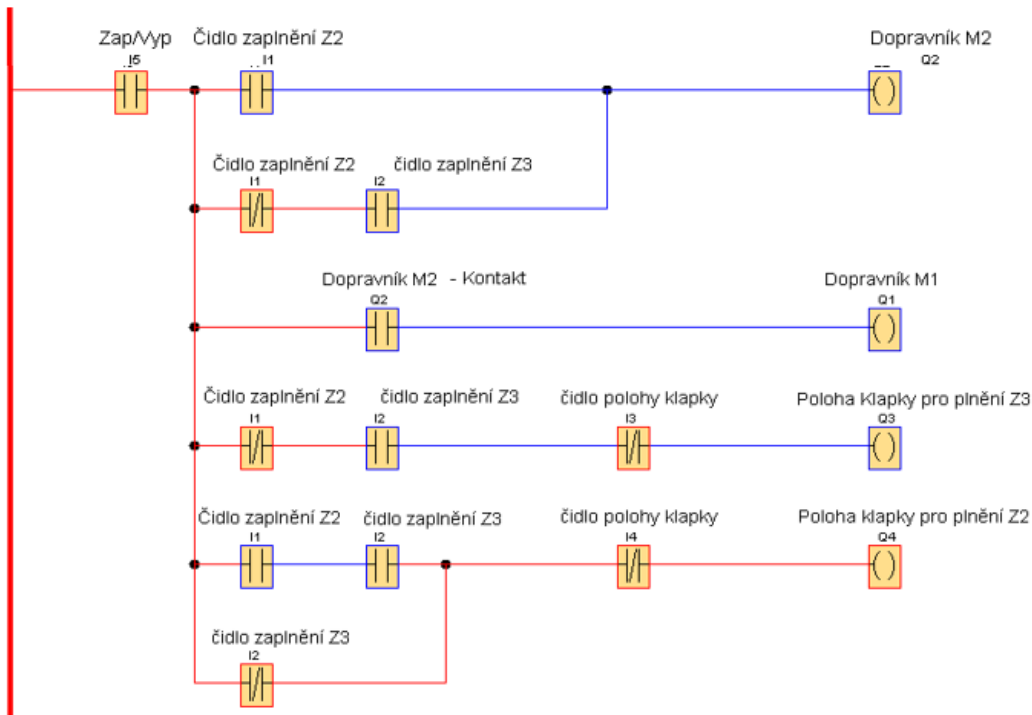
Blok "I5" má funkci vypínače, kterým se celý systém ovládní vypne nebo zapne. Do obvodu pro ovládní klapky byly navíc vloženy koncové spínače "I3" a "I4", které zastaví chod motoru po dosažení koncové polohy klapky.

Jsou-li bloky (čidla) "I1" a "I2" vypnuty je zásobník plný.

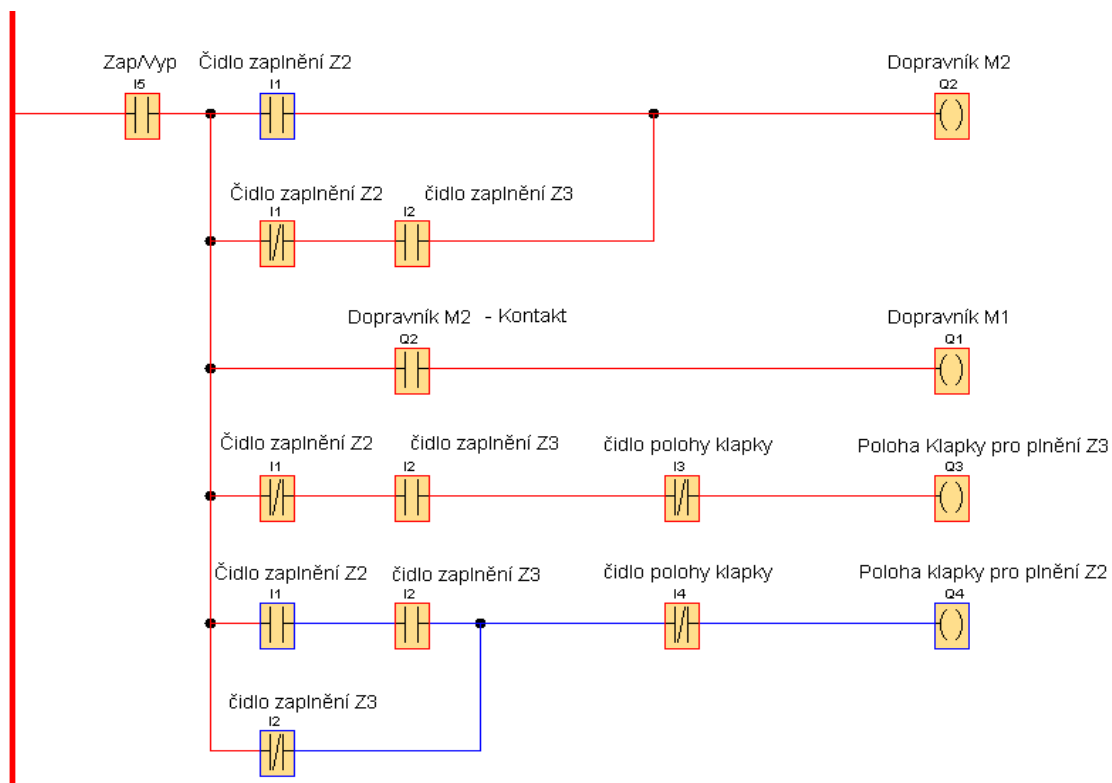
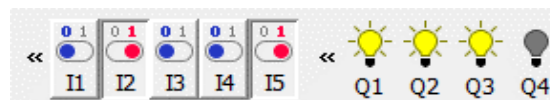


Stav, kdy jsou oba zásobníky zaplněny a dochází k přesunu klapky vpravo.

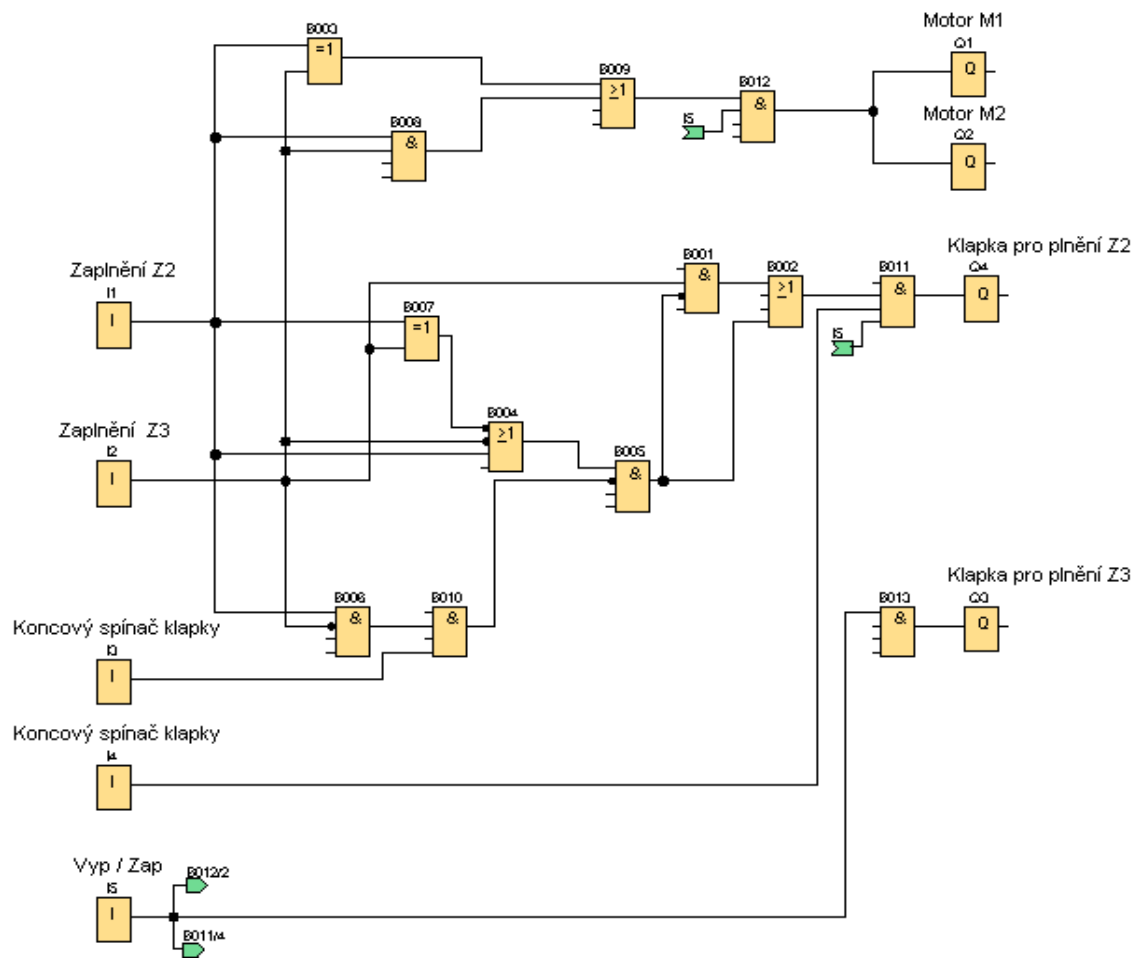




Stav, kdy je zásobník 2 zaplněn a dochází k přesunu klapky do polohy vlevo – plní se zásobník 3.



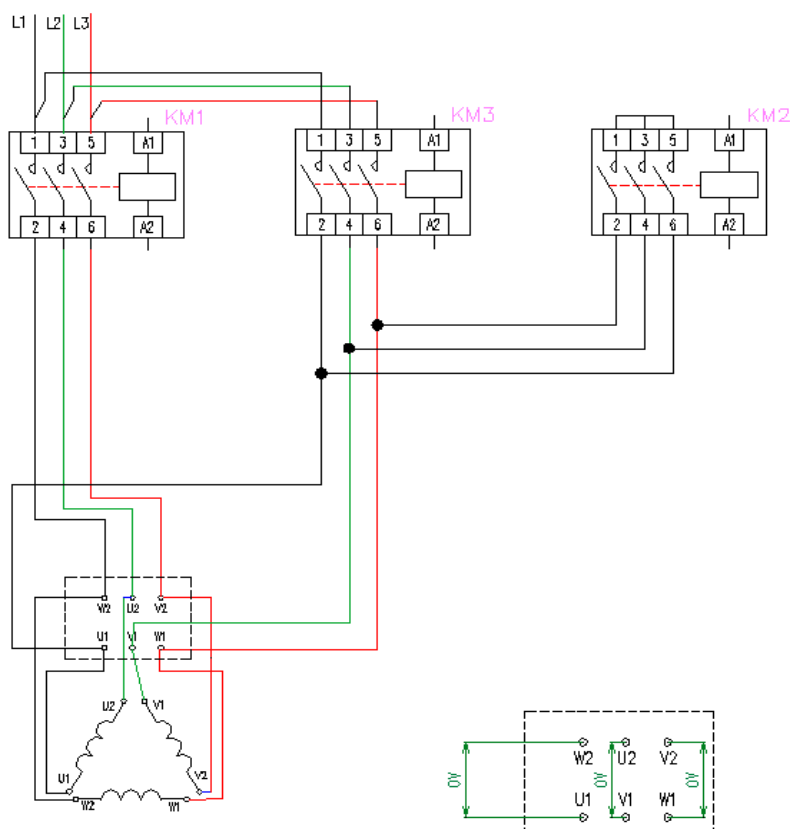
Zapojení stejného příkladu pomocí bloků FBD



ASYNCHRONNÍ MOTORY

Asynchronní motory patří mezi nejčastěji používané motory . Proto zde uvádím několik programů, které se mohou hodit v různých složitějších programech.

Přepínač hvězda trojúhelník



Pro snížení proudového nárazu při spouštění asynchronního motoru většinou od 4 kW se používá rozběh hvězda trojúhelník. Zde je na příkladu uvedeno zapojení tří stykačů, používané běžně pro uvedený rozběh.

Popis funkce:

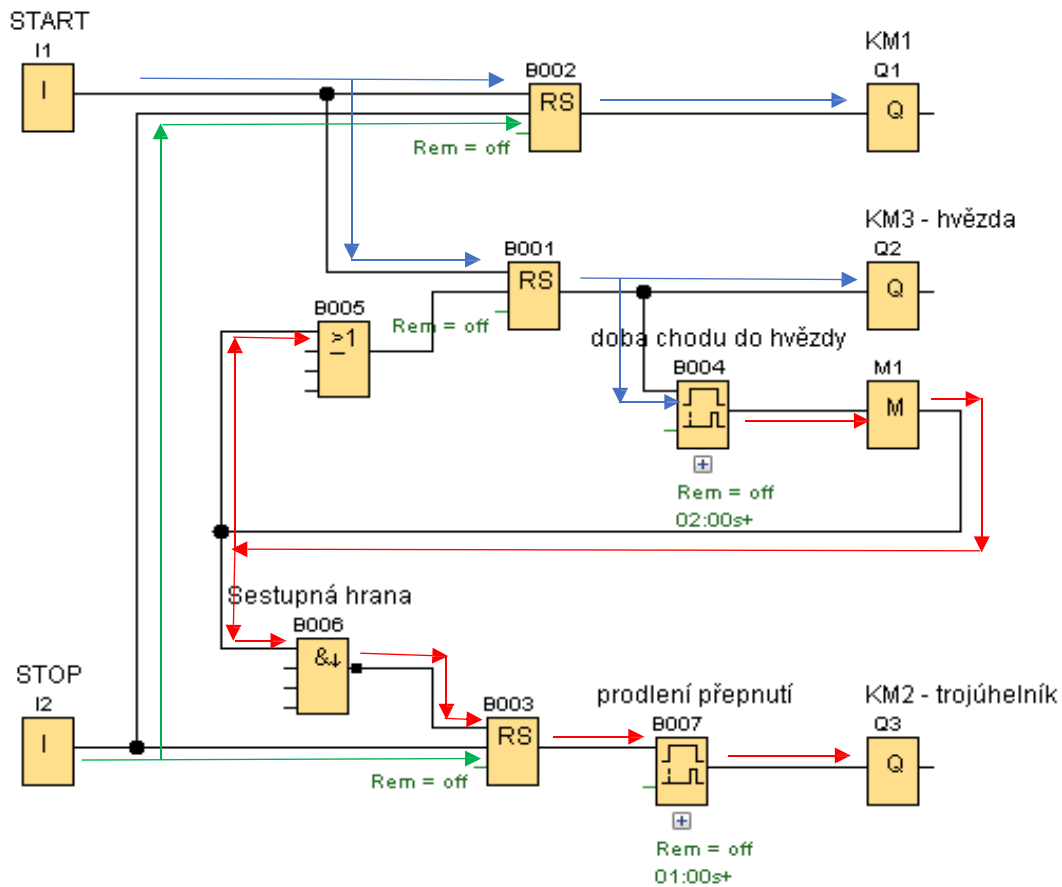
Stiskem tlačítka "I1" - Start, se sepne stykač KM1 "Q1" a KM3 "Q2", tím se spustí chod do hvězdy. Po nastavené době "B004" (zpožděné zapnutí) se vypne stykač KM3-"Q2" a s krátkým zpoždění (např. 0,2s) se sepne stykač KM2 "Q3" - trojúhelník.

Vypnutí motoru (Q1, Q3) se provede tlačítkem "I2" - STOP.

Postup sepnutí vstupů a výstupů zapsaný do tabulky

	krok	I1	I2	Q1	Q2	Q3
stisk tlačítka Start	1.	1	0	1	1	0
vypnutí stykače hvězda	2.	0	0	1	0	0
sepnutí stykače trojúhelník	3.	0	0	1	0	1
stisk tlačítka Stop	4.	0	1	0	0	0

Zapojení bloků



Zapojení do hvězdy – modré šipky:

Stiskem tlačítka START se sepne výstup relé "RS" B002 a tím výstup Q1. Současně se sepne výstup relé "RS" B001, to sepne výstup Q2. Motor běží v zapojení hvězda.

Zapojení do trojúhelníku – červené šipky:

Sepnutím výstupu relé "RS" B001 dojde k sepnutí relé zpožděného sepnutí B004. To po nastavené době přes "M1" vypne výstup relé RS B001 a tím Q2-zapojení hvězda. Blokem "NAND" (impuls na sestupnou hranu) B006 se sepne relé RS B003. Tím se sepne relé zpožděného zapnutí B007, a to sepne Q3. Motor běží do trojúhelníku.

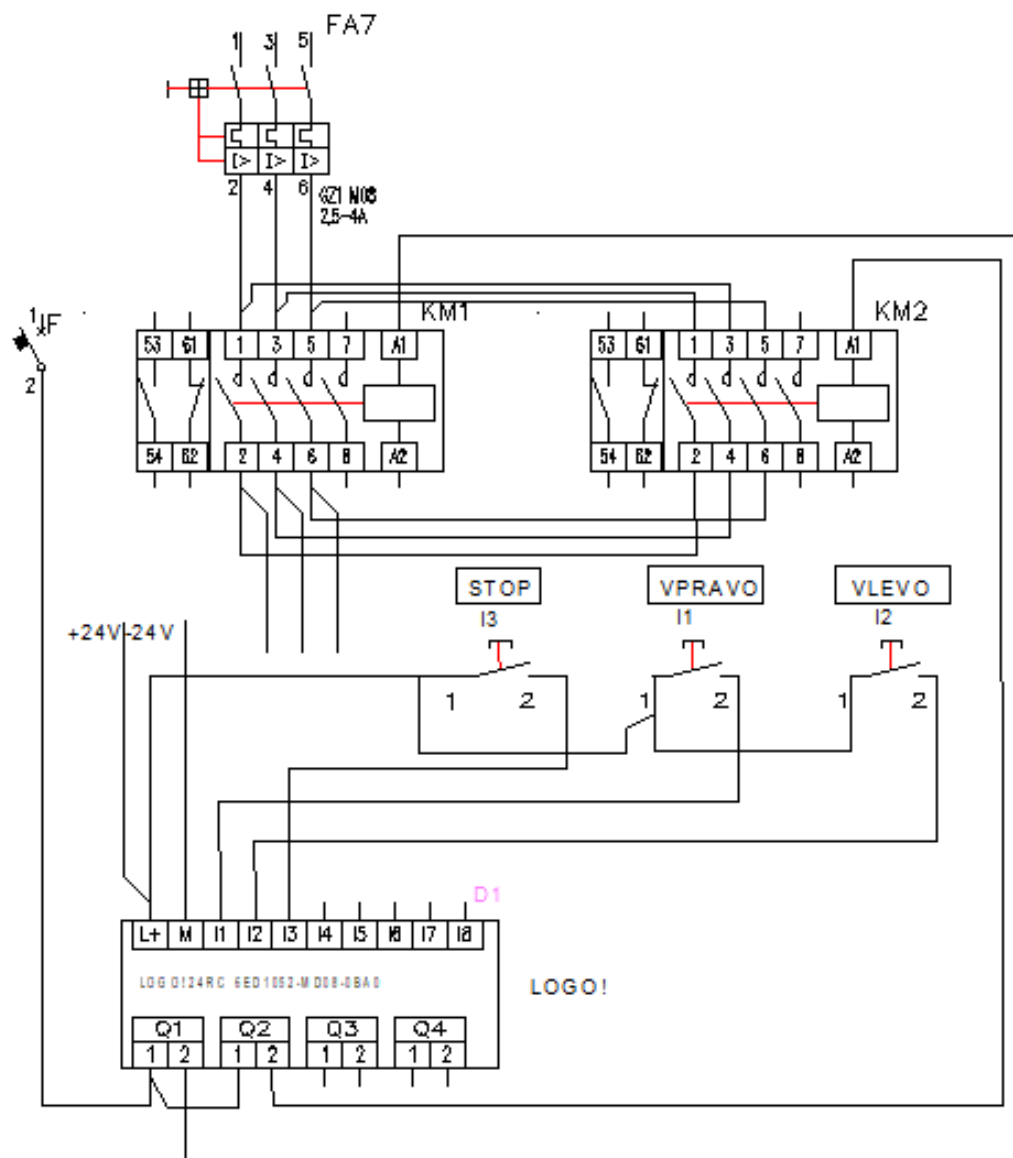
Vypnutí – zelené šipky:

Stiskem tlačítka I2 se dostane impuls 1 na vstup R, bloku B002 a B003, tím dojde k vypnutí Q1 a Q3.

Reverzace asynchronního motoru varianta1

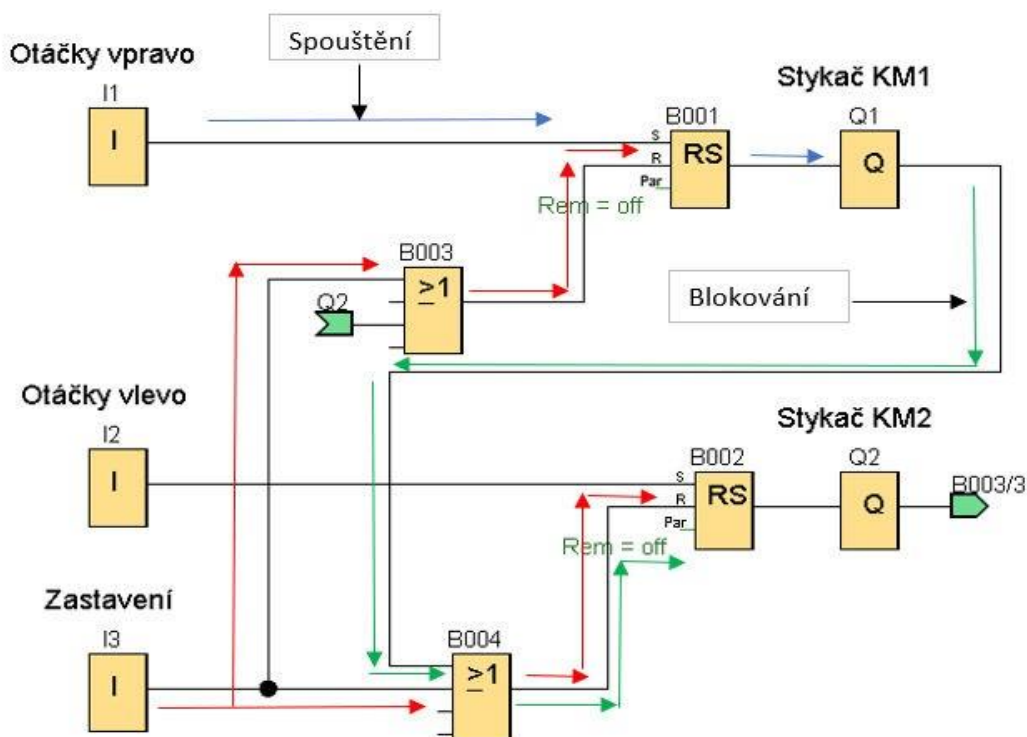
Směr otáčení asynchronního motoru se provede prohozením dvou libovolných fází. Na obrázku je na stykači KM2 prohozena fáze první a druhá.

Jeden směr otáčení se spíná stykačem KM 1 a druhý směr sepnutím stykače KM2. Když je sepnutý stykač KM1, nelze sepnout stykač KM2 a naopak, jsou navzájem. Tlačítkem I1 se spustí směr vpravo. Vypnutí se provede tlačítkem I3.



Popis funkce zapojených bloků

Šipkami je znázorněno spínání stykače KM1 a blokování stykače KM2. Zastavení je pro oba stykače shodné.



Pro spínání výstupů jsou použity Bloky "RS" Samodržné relé. Impulzem na vstup S se **sepne** výstup, impulzem na R se výstup **vypne**.

Stiskem tlačítka "I1" se sepne výstup bloku RS "B001" a tím i výstupu "Q1" (cívka stykače KM1) Současně se sepne i vstup R bloku "B002" a tím je znemožněno sepnutí bloku "Q2", při stisku tlačítka "I2". Vypnutí, kteréhokoliv výstupu "Q" se provádí tlačítkem "I3". Stiskem tlačítka "I3" se se sepne vstup R do bloků "RS".

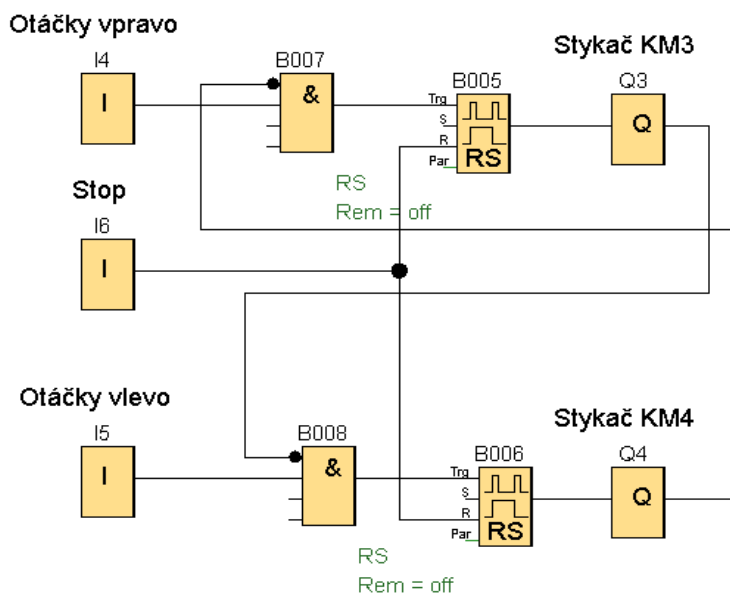
Reverzace asynchronního motoru varianta

2

Další možností pro reverzaci se vzájemným blokováním asynchronního motoru je použití bloku "Pulse Relay" (pulzní proudové relé) a bloku logického součinu "AND" s negovaným jedním vstupem. Přejde-li 1. impulz na vstup Trg sepne se výstup, při 2. impulzu na vstup Trg se výstup vypne. Zapnutí stykače KM3 se provede stiskem tlačítka "I4", jeho vypnutí se provede jeho opětovným stiskem. Tlačítko Stop "I6" slouží jako havarijní, jeho stiskem se vypne zapnutý stykač.

Zapnutí stykače KM4 se provede stiskem tlačítka "I5", jeho vypnutí se provede jeho opětovným stiskem.

Popis funkce zapojených bloků



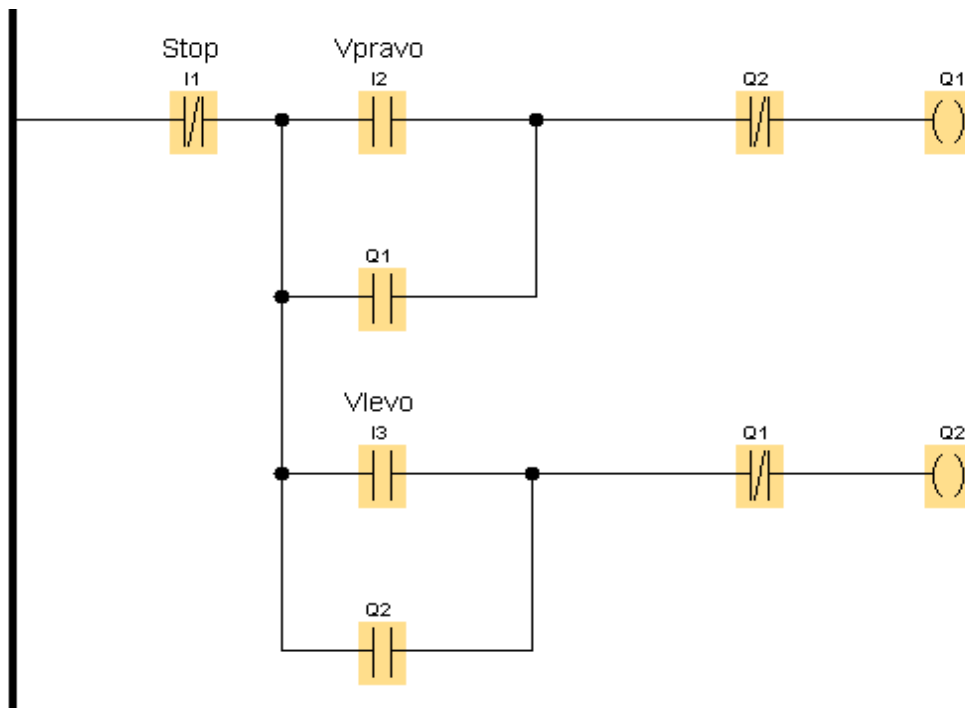
Výstup bloku "AND" je sepnutý (1), jsou-li oba vstupy sepnuté (1). Je-li jeden vstup nebo oba (0) potom i výstup je "0". Toho je využito pro vzájemnou blokadu, kdy např. při spuštění motoru výstupem Q1 nelze sepnout motor výstupem Q2.

Při vypnutém výstupu (0) "Q4" je na negovaném vstupu "AND" (1), při stisku tlačítka "I4" je na vstupu také (1) a dojde k sepnutí pulzního relé a tím i výstupu Q3. Pokud stiskneme při zapnutém výstupu "Q3" tlačítko "I5", nedojde k sepnutí výstupu Q4, protože při sepnutém výstupu "Q3" je na vstupu do bloku "B008" "AND" nula (0).

Reverzace asynchronního motoru v programu LAD

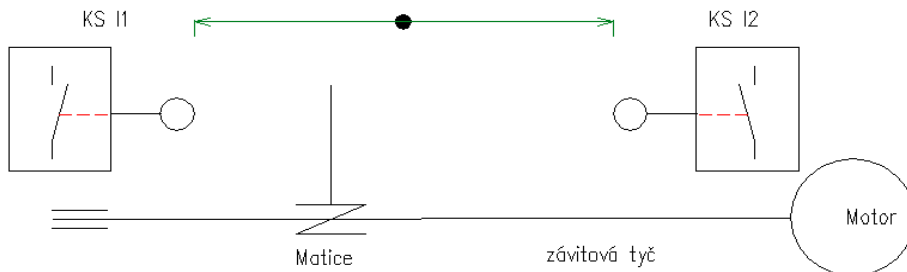
Motor podle toho, zda stiskneme tlačítko "I2" nebo "I3" se bude točit doprava nebo doleva. Změna chodu je možná jen při vypnutém stavu, stisku tlačítka "I1".

Stiskem tlačítka "I2" se sepne výstup "Q1". Spínací kontakty "Q1" překlenou "I2" a výstup "Q1" zůstane trvale sepnutý. Před výstupem "Q1" je předřazený rozpínací kontakt "Q2", ten v případě sepnutí výstupu "Q2" brání sepnutí "Q1", protože je rozpojený. Stejný postup platí při stisknutí tlačítka "I3" pro chod vlevo.

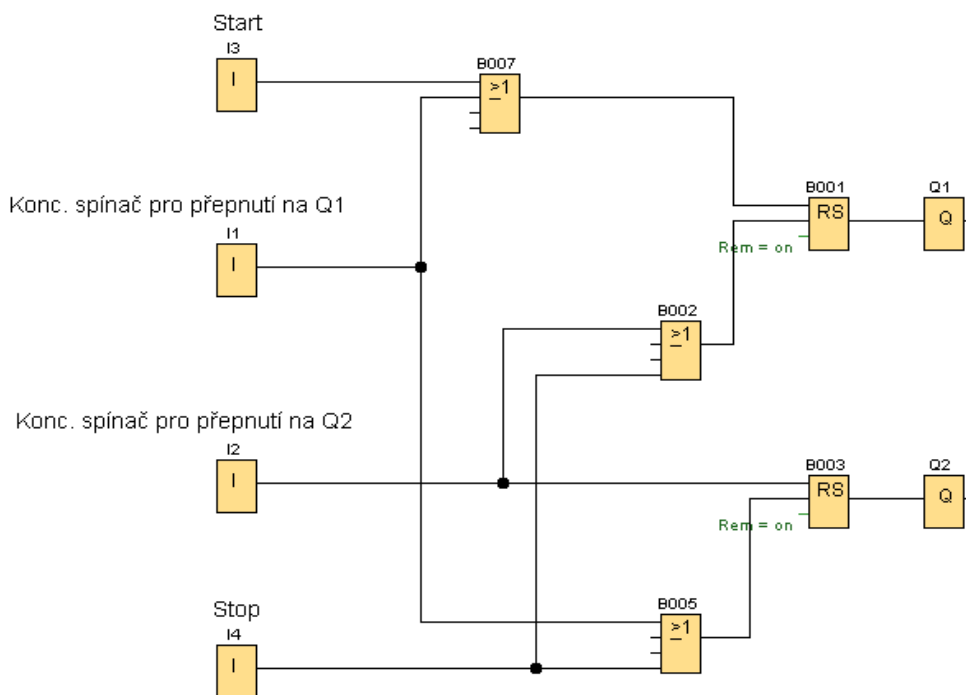


Přímočarý vratný pohyb

Je to případ, kdy se nějaký nástroj pohybuje tam a zpět, např. rovinné broušení. Kdy rotující brusný kotouč brousí rovinnou plochu, dojde na koncový spínač a pohybuje se zpět ke druhému koncovému spínači. Tam dojde k příčnému posuvu brusného kotouče a postup se neustále opakuje. Příklad je bez příčného posuvu .

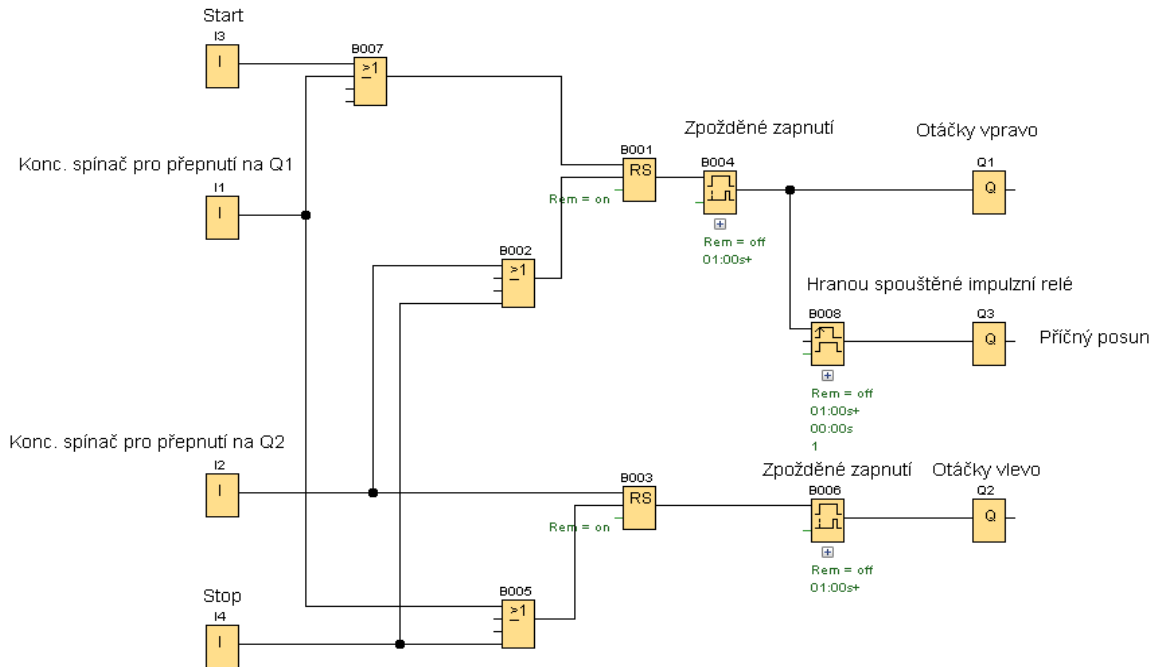


Stiskem tlačítka "I3" Start se sepne blok RS "B001" a tím výstup "Q1" (otáčení motoru vpravo) po dojetí na koncový spínač "I2" se vypne blok RS "B001", tím výstup "Q1". Současně se sepne blok RS "B003" a výstup "Q2" pro otáčení motoru vlevo. Po dojetí na koncový spínač "I1" se postup opakuje.



Přímočarý vratný pohyb s prodlevou na konci a příčným posuvem

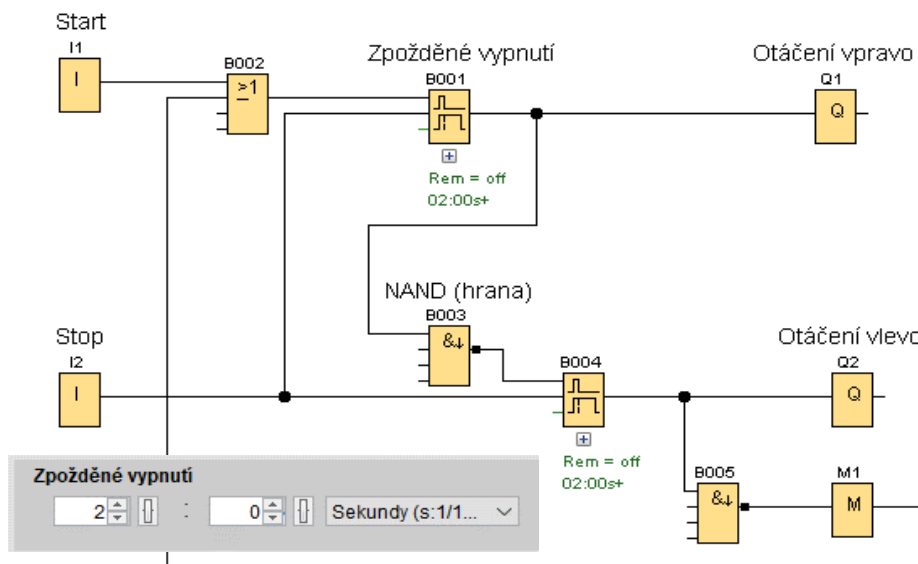
Příklad je shodný s předcházejícím, s tím že do programu jsou vloženy bloky zpožděného zapnutí "B004" a "B006" pro vytvoření prodlevy při změně směru otáčení a dále Hranou spouštěné impulzní relé "B008", které spouští výstup "Q3" pro ovládání motoru pro příčný posun.



Cyklická změna směru otáčení

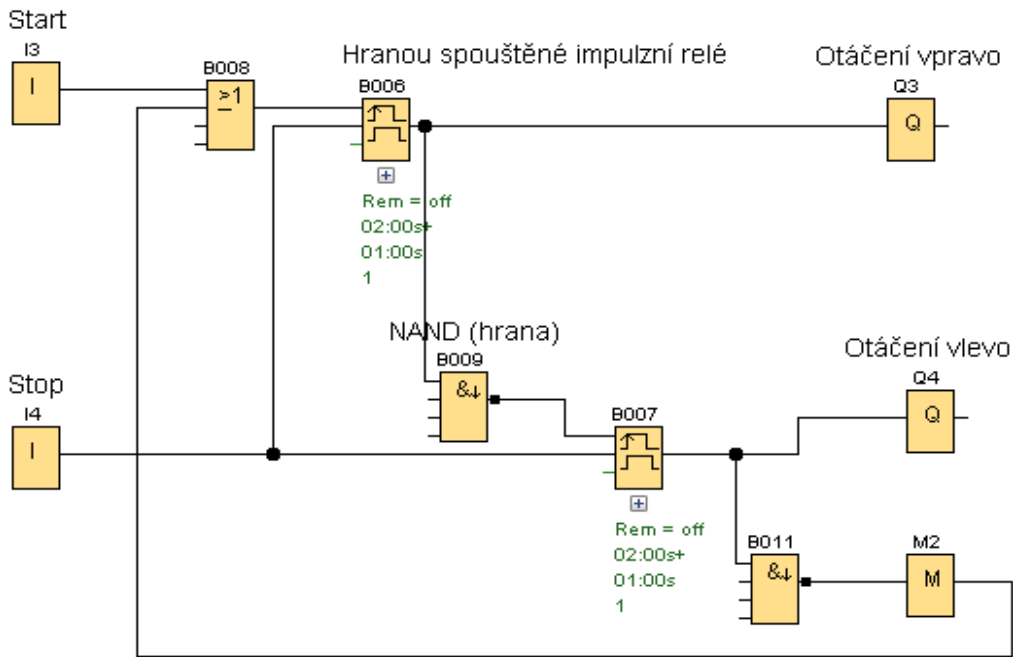
Je to případ například míchání, kdy se buben točí chvíli na jednu stranu a chvíli na druhou stranu. Pro představu je možné uvést buben automatické pračky.

Po stisku tlačítka "I1" se sepne výstup bloku Zpožděné vypnutí "B001" a tím i výstup "Q1". Po nastavené době se výstup vypne. Vypnutím dá blok NAND (hrana) impulz bloku Zpožděné vypnutí "B004", sepne se jeho výstup a tím i výstup "Q2". Po skončení časování se vypne výstup "B004", blok NAND (hrana) "B004" dá impulz k sepnutí výstupu bloku "B001" a postup se opakuje.



Změna směru otáčení po nastavené době s prodlevou

Je to obdobný příklad jako ten předešlý, ale změna je v tom, že po vypnutí "Q3" je časová prodleva a po ní se sepne výstup "Q4", po jeho vypnutí se s časovou prodlevou sepne opět výstup "Q3". Pro časování jsou použity Hranou spouštěné impulzní relé.

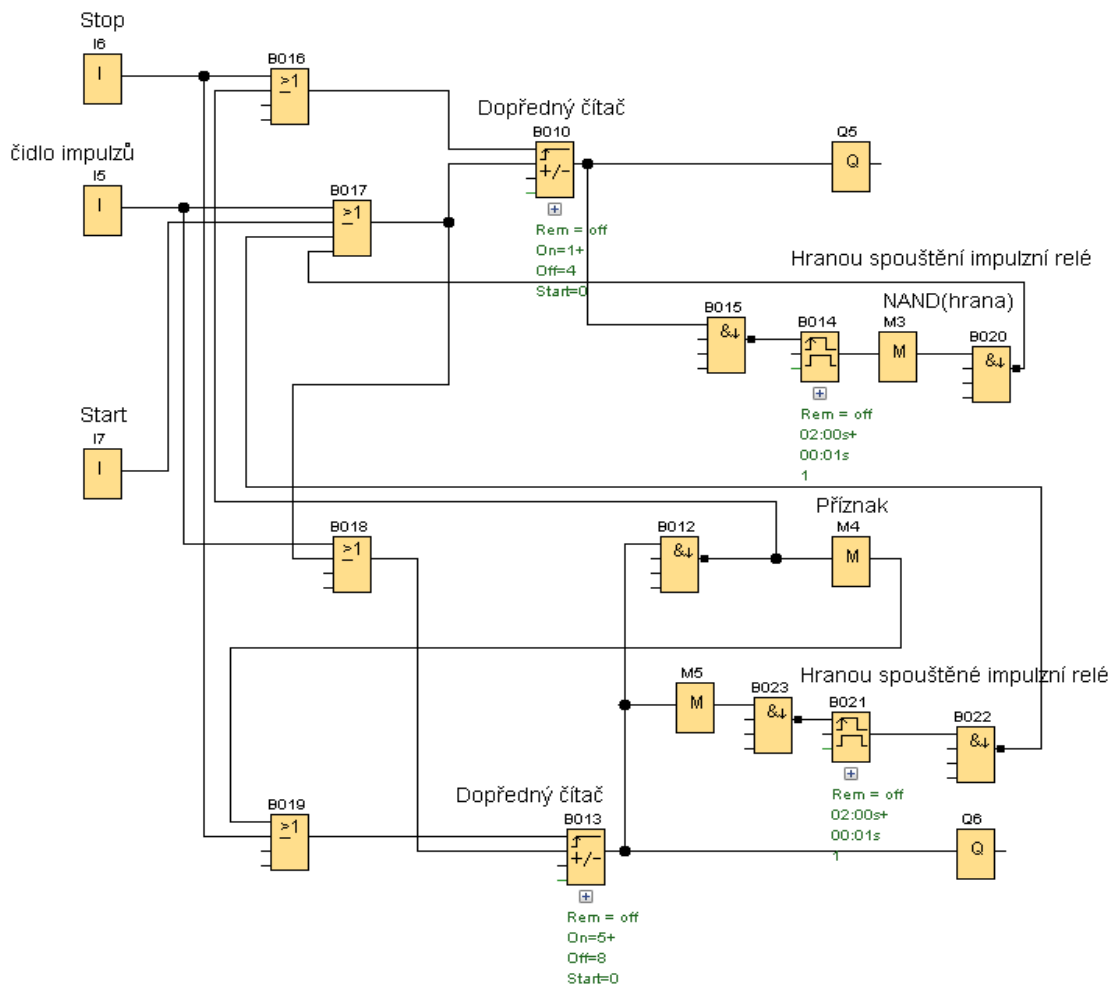


Šířka pulzu (TH)		
2	:	0
Sekundy (s:1/1...)		
Mezipulzní šířka (TL)		
1	:	0
Sekundy (s:1/1...)		

Změna směru otáček po nastaveném počtu otáček s prodlevou při změně směru otáček

Motor se točí několik otáček v jenom směru a potom zase několik otáček v druhém směru.

Jejich počet závisí na nastavení Dopředných čítačů. Spuštění se provede tlačítkem "I7" Start. Jeho stiskem se sepne výstup "Q5" pro chod motoru vpravo. Jakmile se motor otočí začnou se pomocí čidla "I5" načítat pulzy na obou čítačích. Až dosáhne čítač "B010" hodnoty Vypnuto (4) výstup "Q5" se vypne. Hodnota Zapnuto (1) je z důvodu, aby po přivedení napětí do LOGA nedošlo hned ke spuštění motoru, ale až po stisku Start (načte se hodnota 1). Vypnutím čítače "B010" dá blok "B015" impuls na vstup Hranou spouštěného impulzního relé "B014", sepne se jeho výstup a dokončením časování se výstup vypne. Přechodem ze zapnutí na vypnutí dá blok NAND (hrana) "B020" impuls na vstup obou čítačů "B010" a B013". Na čítači B013 se zvýší hodnota na (5), což je hodnota zapnutí výstupu čítače. Motor se začne otáčet doleva. Udělá tři otáčky (podle nastavení) a zastaví se. Časová prodleva mezi změnou směru otáčení doprava je dána nastavením času Hranou spouštěného impulzního relé "B022". Vypnutím výstupu čítače "B013" dojde k vydání impulsu blokem "B012" k vynulování obou čítačů "B010" a "B013". Po ukončení časování Bloku "B021" dojde k impulsu na oba vstupy Cnt čítačů a načte se na nich hodnota (1), tím se sepne výstup "Q5" a postup se opakuje. Zastavení se provede stiskem tlačítka "I6" Stop (vynulují se oba čítače).



Nastavení čítače "B010"

Zapnuto
<input type="text" value="1"/>
Vypnuto
<input type="text" value="4"/>

Nastavení čítače "B013"

Zapnuto
<input type="text" value="5"/>
Vypnuto
<input type="text" value="8"/>

Nastavení hranou spouštěného impulzního relé "B014" a "B021"

Šírka pulzu (TH)
<input type="text" value="2"/> : <input type="text" value="0"/> Sekundy (s:1/1...)
Mezipulzní šírka (TL)
<input type="text" value="0"/> : <input type="text" value="0"/> Sekundy (s:1/1...)

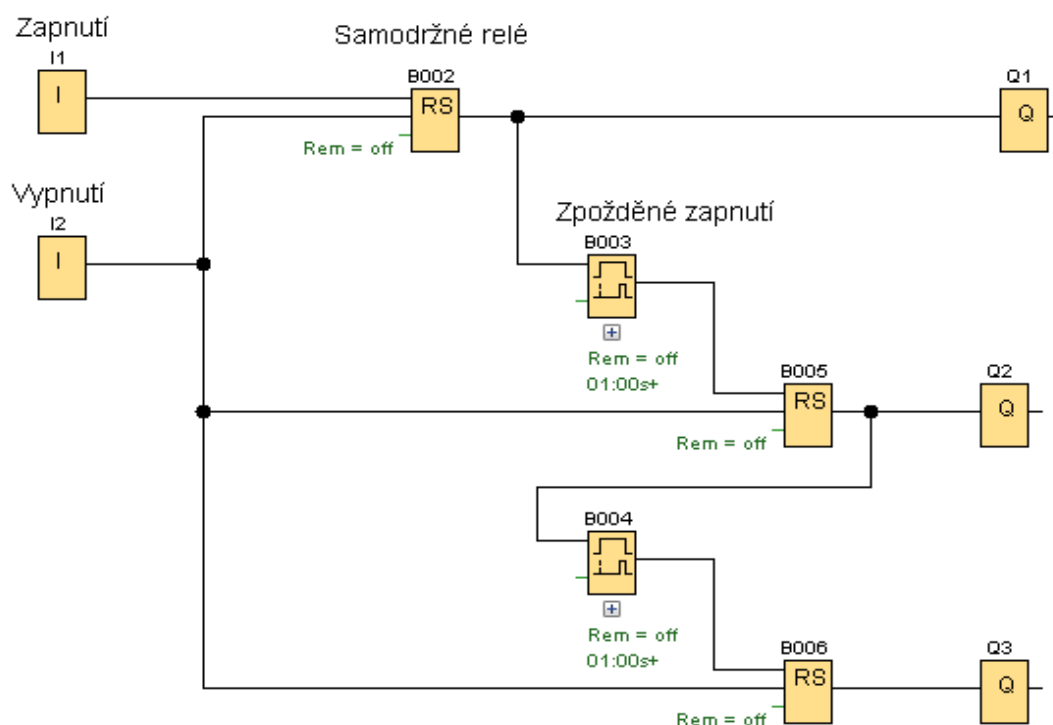
Nastavení může být v reálném případě různé.

STROJNÍ LINKY

Tímto názvem mám na mysli seřazení strojů za sebou, které budou spouštěny postupně. Význam to má především ve zmenšení proudového nárazu a někdy i z technologického hlediska. Na konci spouštění budou všechny stroje spuštěné. V části návodu Kroky se výstupy spouštěly po sobě, ale spuštěný zůstal jen jeden výstup. V následujících příkladech jsou spouštěny tři stroje (výstupy Q). Po sepnutí spínače nebo tlačítka se sepne okamžitě první stroj po časové prodlevě se spustí druhý stroj a po další časové prodlevě se spustí třetí stroj.

Spínání a vypínání tlačítka pomocí bloků "Zpožděného zapnutí" a "RS" s různou délkou postupného spínání

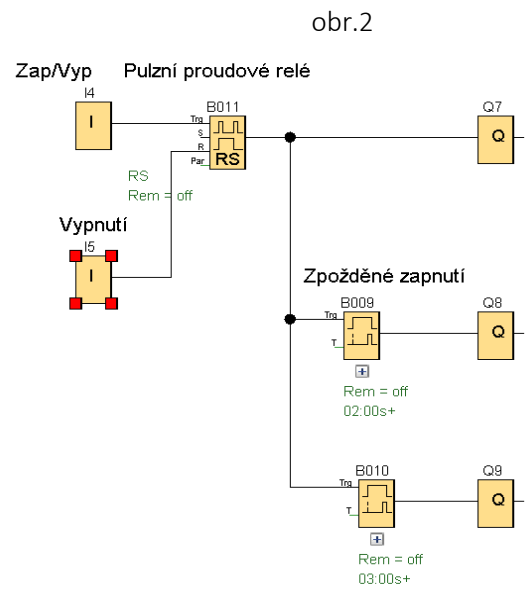
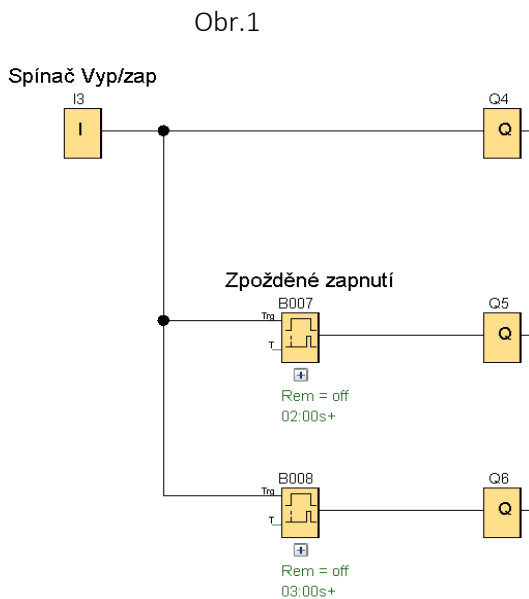
Stiskem tlačítka "I1" se sepne výstup bloku "RS" a tím výstup "Q1" a začne časovat Blok "B003". Po sepnutí jeho výstupu se sepne blok "B005" a výstup "Q2". Dále se postup opakuje.



Spínání výstupů pomocí bloků "Zpožděného zapnutí" s různou délkou postupného spínání

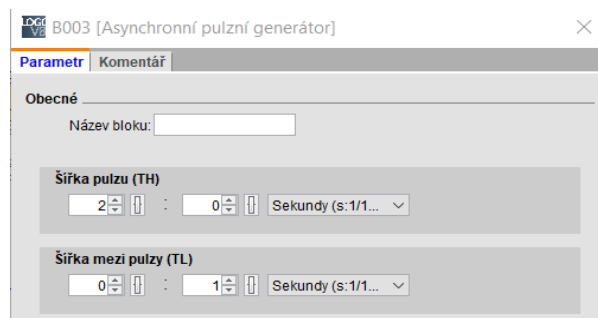
Na obr.1 je vypnutí a zapnutí řešeno spínačem. Bloky zpožděného zapnutí mají různou délku nastavení času . Po sepnutí spínače se sepne výstup Q4, po nastavených 2s bloku "B007" se sepne výstup "Q5", Po další 1s se sepne blok "Q6". Blok "B007" je nastaven na 3s. Čas nastavení musíme postupně u jednotlivých bloků zvyšovat, protože bloky začínají časovat najednou.

Na obr. 2 je zapnutí řešeno tlačítkem "I4" a vypnutí tlačítkem "I5". Pro sepnutí a vypnutí je použitý blok "B011" Pulzní proudové relé. Tlačítkem "I4" je možné výstupy jedním stiskem zapínat a druhým stiskem vypínat. Tlačítkem "I5" se výstupy jenom vypínají. Tato tlačítka může sloužit pro nouzové zastavení.



Postupné zapínání se stejnými intervaly sepnutí pomocí Posuvného registru.

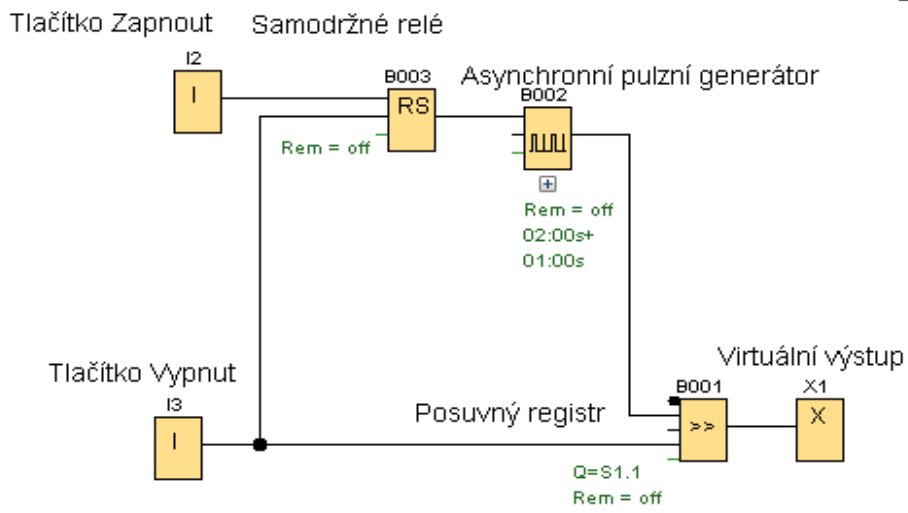
Doba mezi jednotlivými připojeními výstupů je dána nastavením Asynchronního pulzního generátoru



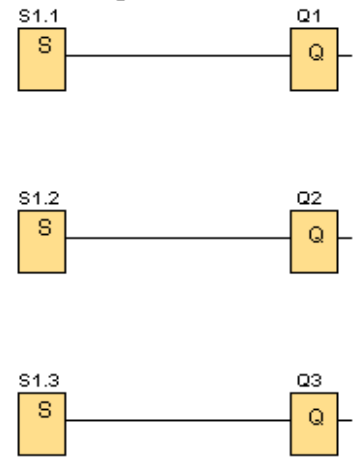
Pro postupné spínání musí být vstup posuvného registru na hodnotě 1. To je dosaženo jeho invertováním, nebo je to možné pomocí bloku High.



Posuv jednotlivých bitů zajišťuje Asynchronní pulzní generátor, zapojený do vstupu Trg Posuvného registru. Spuštění se provede tlačítkem "I2". Vypnutí tlačítkem "I3". Resetuje se relé RS a Posuvný registr.



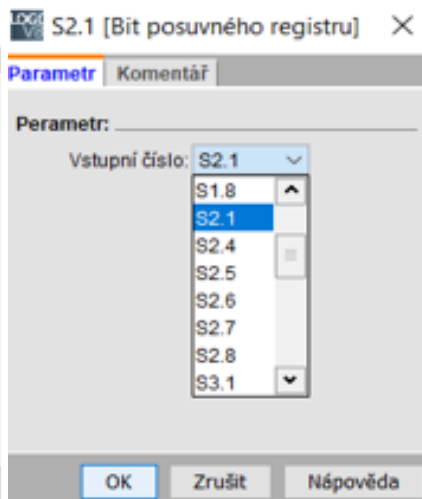
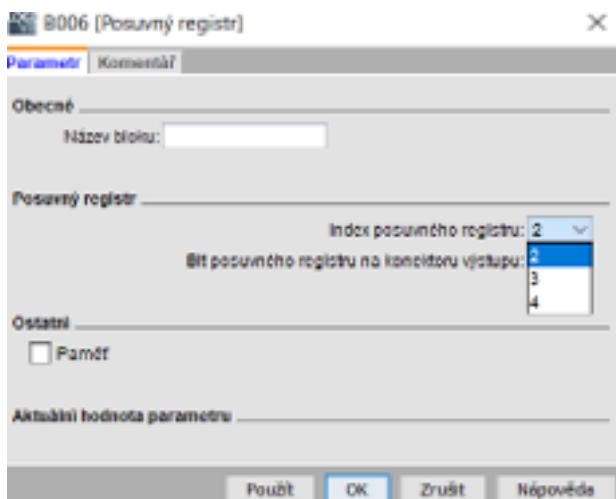
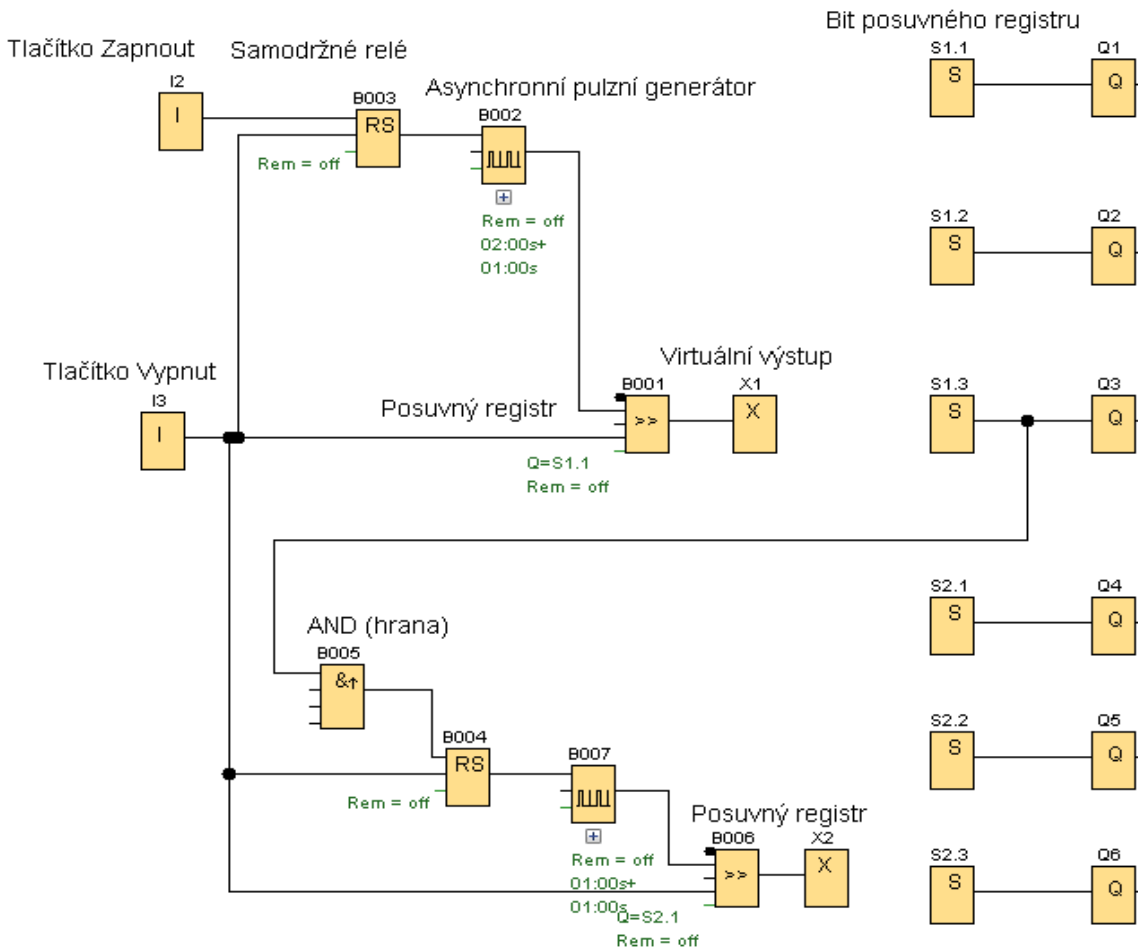
Bit posuvného registru



Postupné spínání se stejnými intervaly pomocí dvou Posuvných registrů

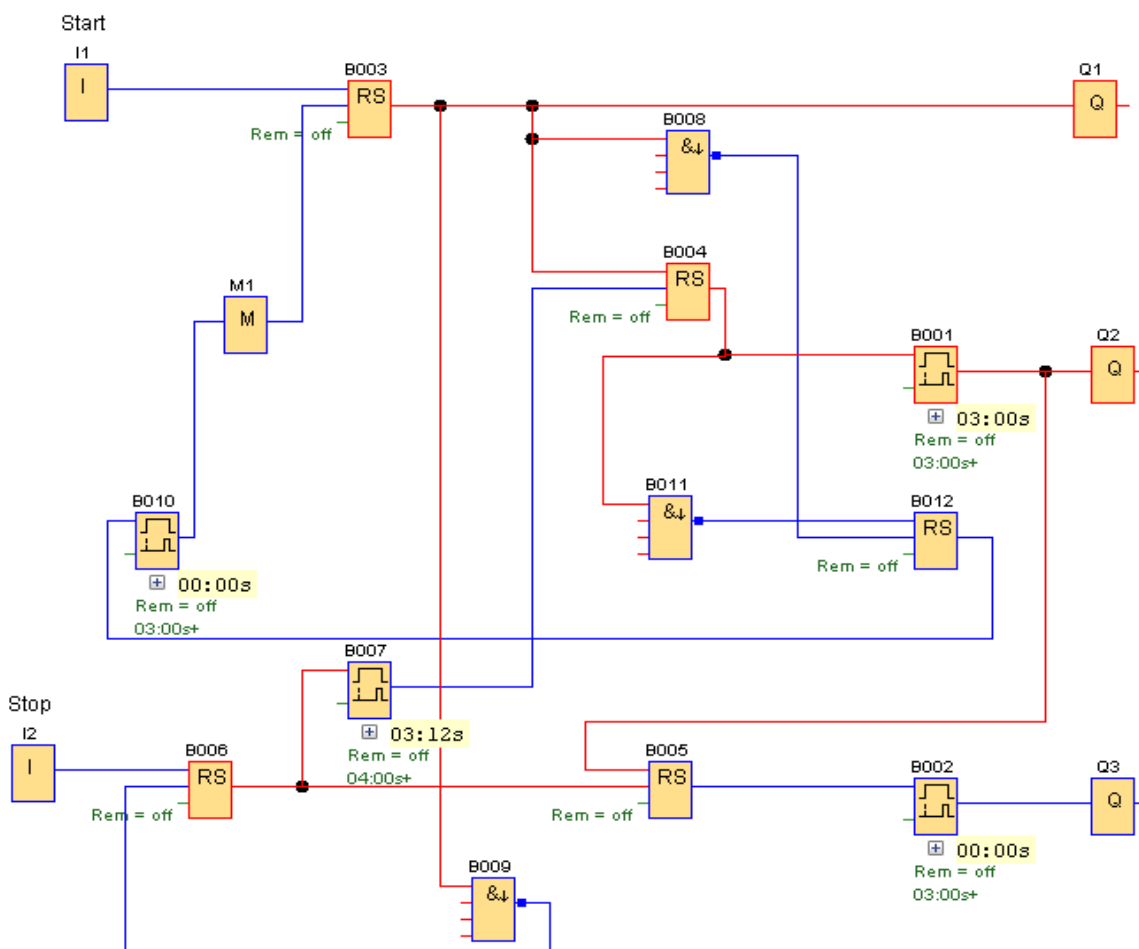
Jeden posuvný registr má osm bitů. Pokud potřebujeme více výstupů musíme přidat další posuvný registr, musíme dát pozor, aby Bit posuvného registru (první číslo S2.1) byl ve shodě s indexem Posuvného registru (2).

Zde je příklad postupného zapínání výstupů s použitím dvou posuvných registrů. Po sepnutí bitu S1.3 se impulzem na vzestupnou hranu AND (hrana) se sepnou RS relé a impulzy pulzního generátoru na vstup Posuvného registru "B006" začnou postupně spínat bity S2.1, S3.2, S2.3.

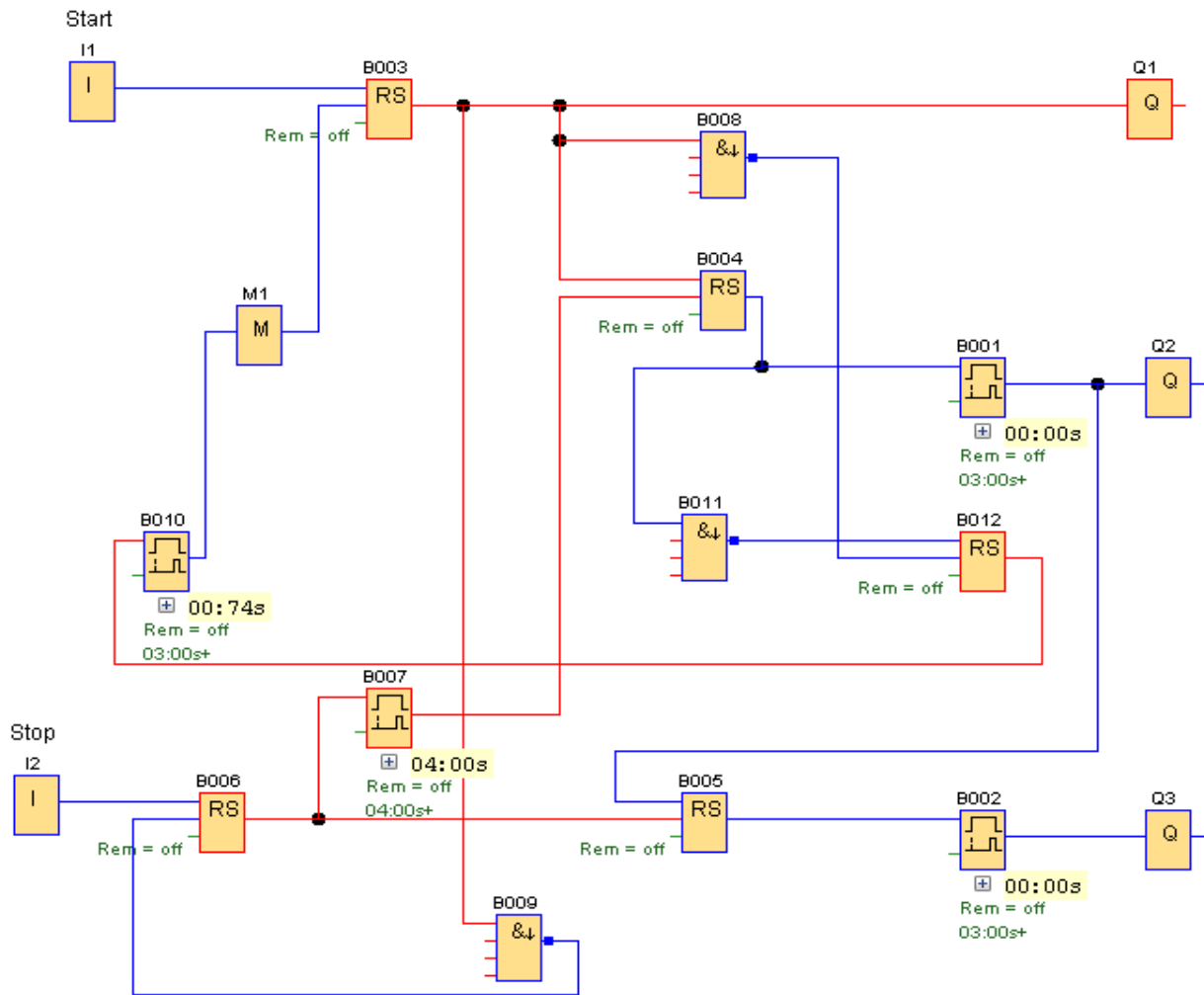


Postupné spínání a vypínání

Výchozí stav je zapnutí všech výstupů je popsán na příkladu a). Zde popíšeme postupné vypínání. Máme strojní linku s plnicím dopravníkem "Q3", strojem "Q2" a výstupním dopravníkem "Q1". Při stisku tlačítka "I2" Stop se nejprve vypne dopravník Q3 a po nastavené době stroj "Q2" a nakonec dopravník "Q1". Stiskem tlačítka "I2" se sepnou blok RS "B006" ten vypne blok RS "B005" a tím výstup "Q3". Současně začne časovat blok "B007" jehož výstup je napojený na blok RS "B004".



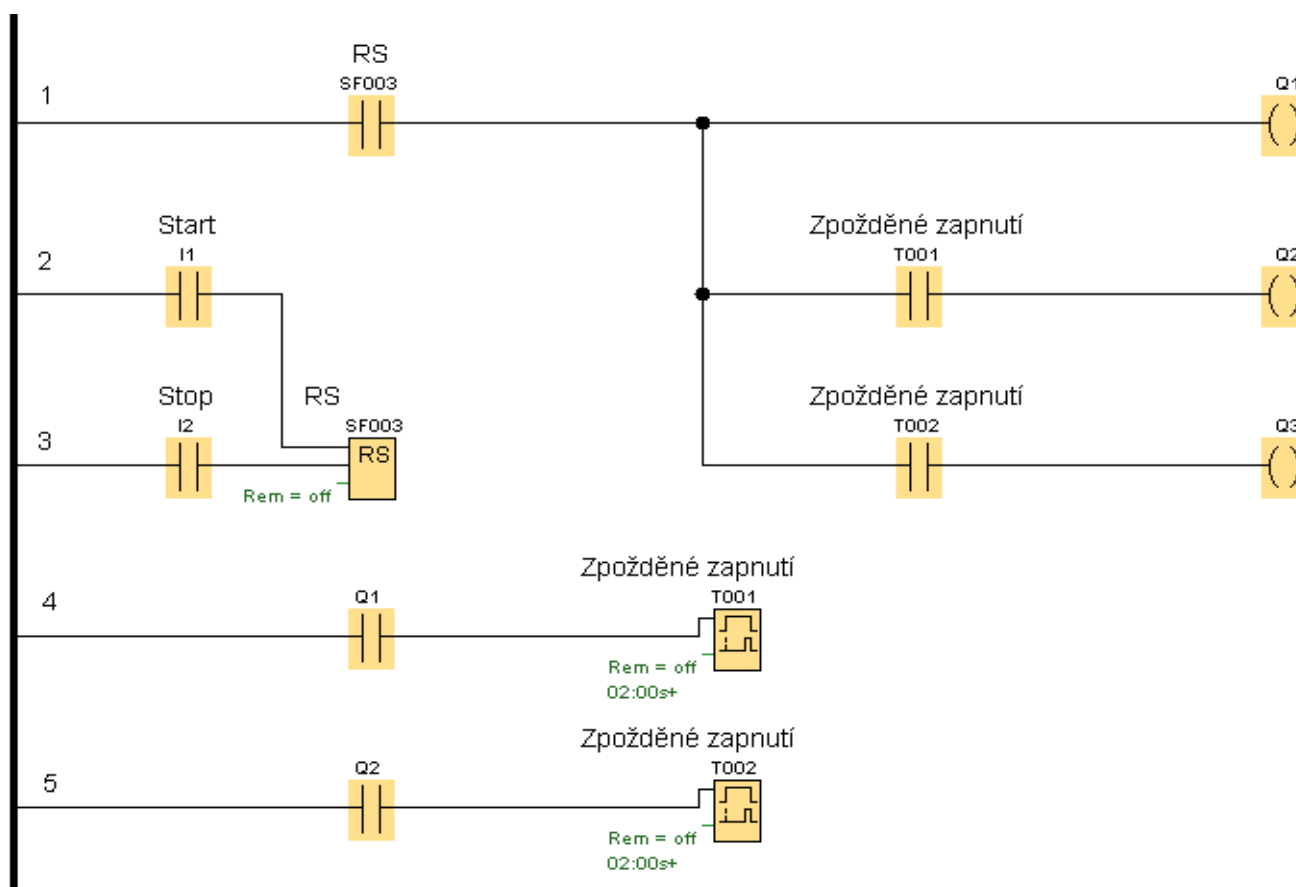
Po ukončení časování bloku "B007" se sepne jeho výstup a ten vypne blok "B004" a tím se vypne výstup "Q2". Vypnutím bloku "B004" dojde k impulzu NAND (hrana) a sepne se blok RS "B012" a dojde k sepnutí časovače "B010". Po dokončení časování dojde k vypnutí bloku RS "B003" a tím i výstupu "Q1". Vypnutím bloku RS "B003" dá blok "B009" impulz na vstup R bloku RS "B006" a vše se vypne.



Spínání a vypínání tlačítka pomocí bloků "Zpožděného zapnutí" a "RS" s různou délkou postupného spínání s bloky LAD

Stiskem tlačítka Start se sepne blok RS "SF003" a tím jeho spínací kontakt v 1. řádku. Jeho sepnutím se sepne výstup "Q1". Současně se sepne spínací kontakt "Q1" ve 4. řádku a tím časování bloku "T001". Po sepnutí výstupu časovače se sepne jeho spínací kontakt "T001" a tím i výstup "Q2". V 5. řádku se sepne kontakt výstupu "Q2" a začne časovat blok "T002". Po dokončení časování se sepne spínací kontakt "T002" a tím výstup "Q3".

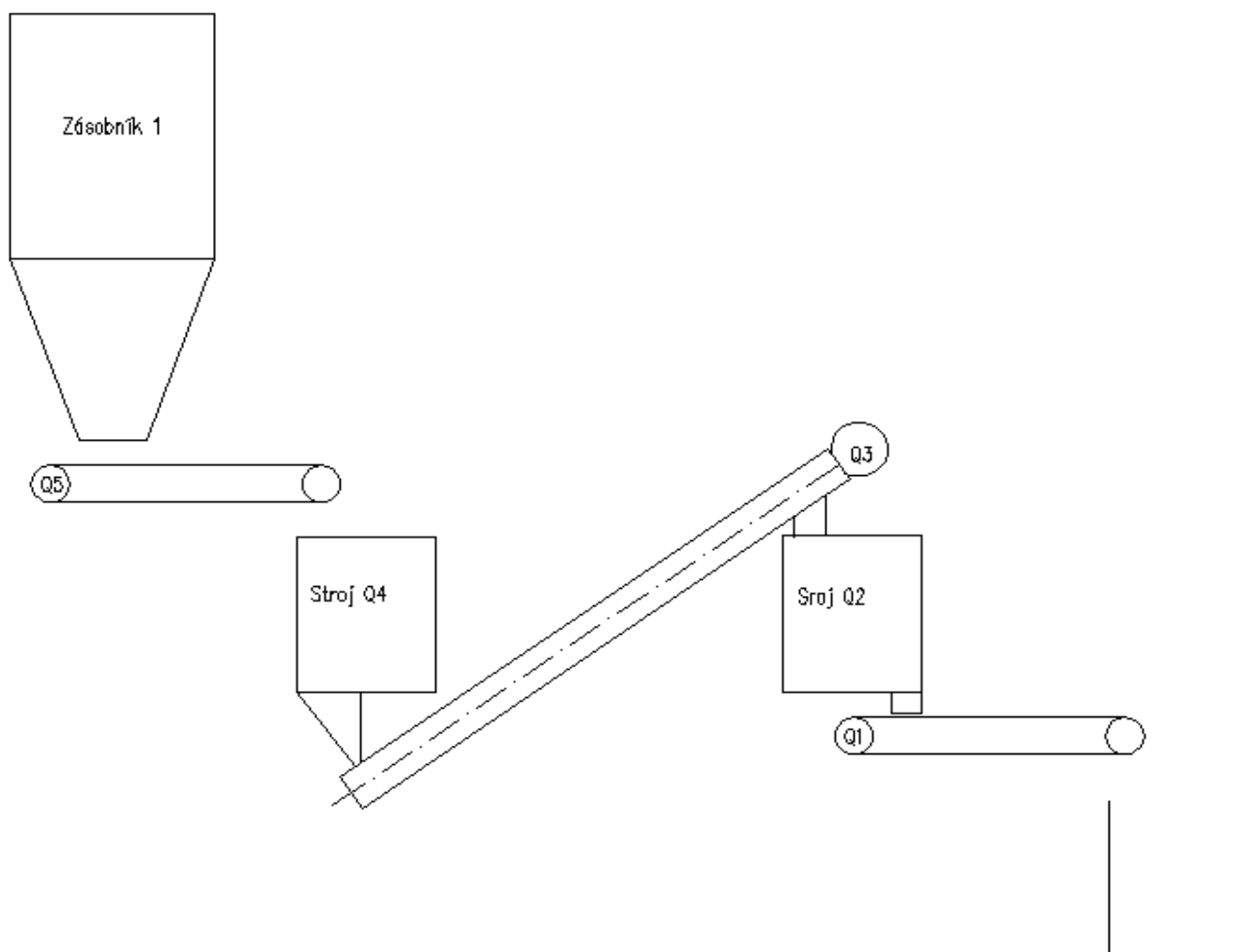
Vypnutí se provede stiskem tlačítka Stop (řádek 3). Tím se rozpojí kontakty "SF003" a vypnou se všechny výstupy (Q1, Q2, Q3). Jejich vypnutím se rozpojí kontakt Q1 ve 4. řádku a Q2 v pátém řádku a vypnou se oba bloky Zpožděného zapnutí.



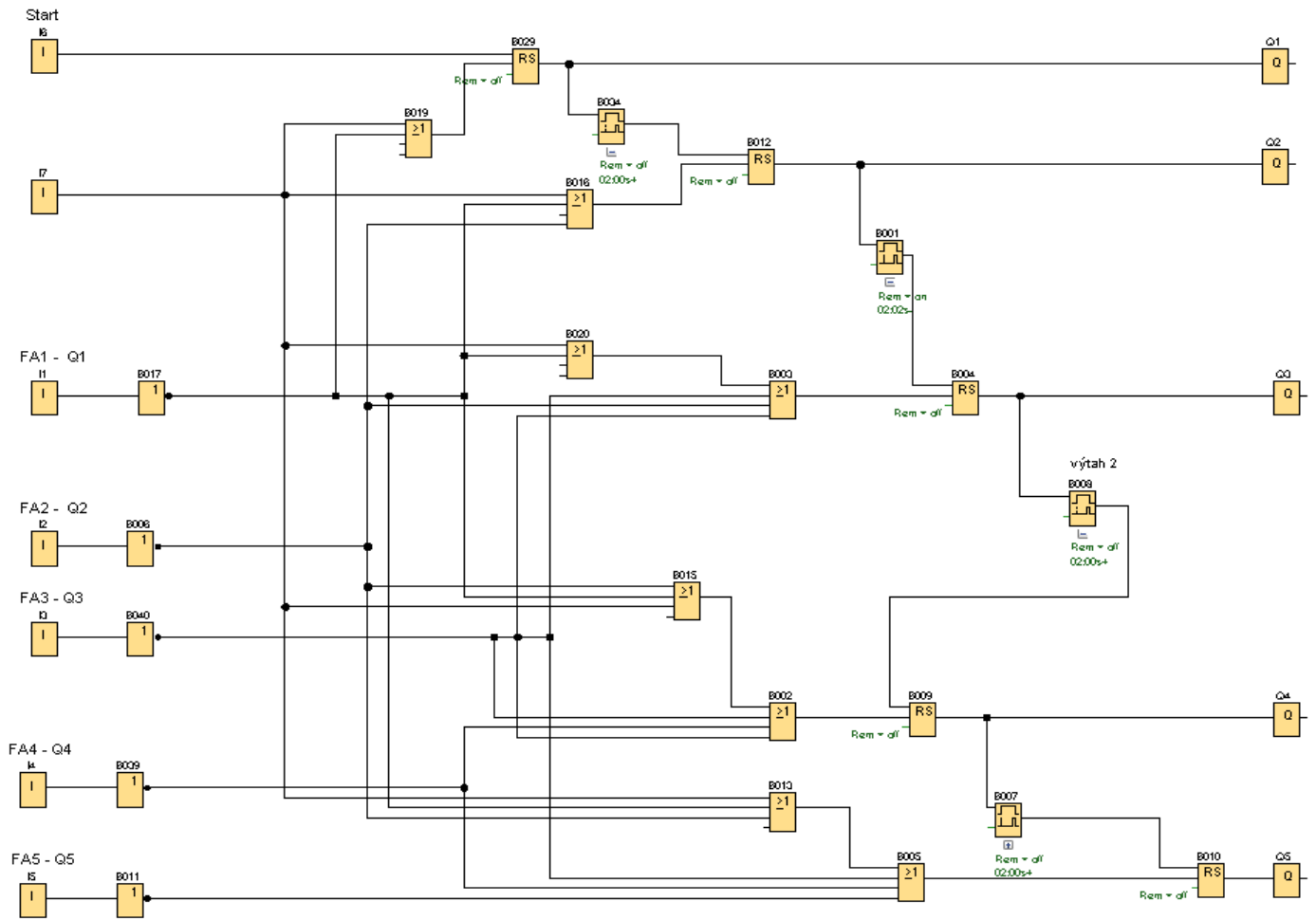
Příklad linky složený z pěti strojů

Po spuštění linky se nejprve rozeběhne dopravník Q1, poslední v řadě, po něm se spustí s časovou prodlevou stroj Q2, dále s prodlevou dopravník Q3, následuje spuštění stroje Q4, a nakonec se spustí plnicí dopravník Q5 oba s časovým odstupem. Každý motor má ochranu proti přetížení. Při vypnutí ochrany motoru např. Q3 se vypne motor Q3 a motory Q4 a Q5. Vždy před ním, aby se přerušil tok materiálu a docházelo k ucpání, nebo přepadávání materiálu.

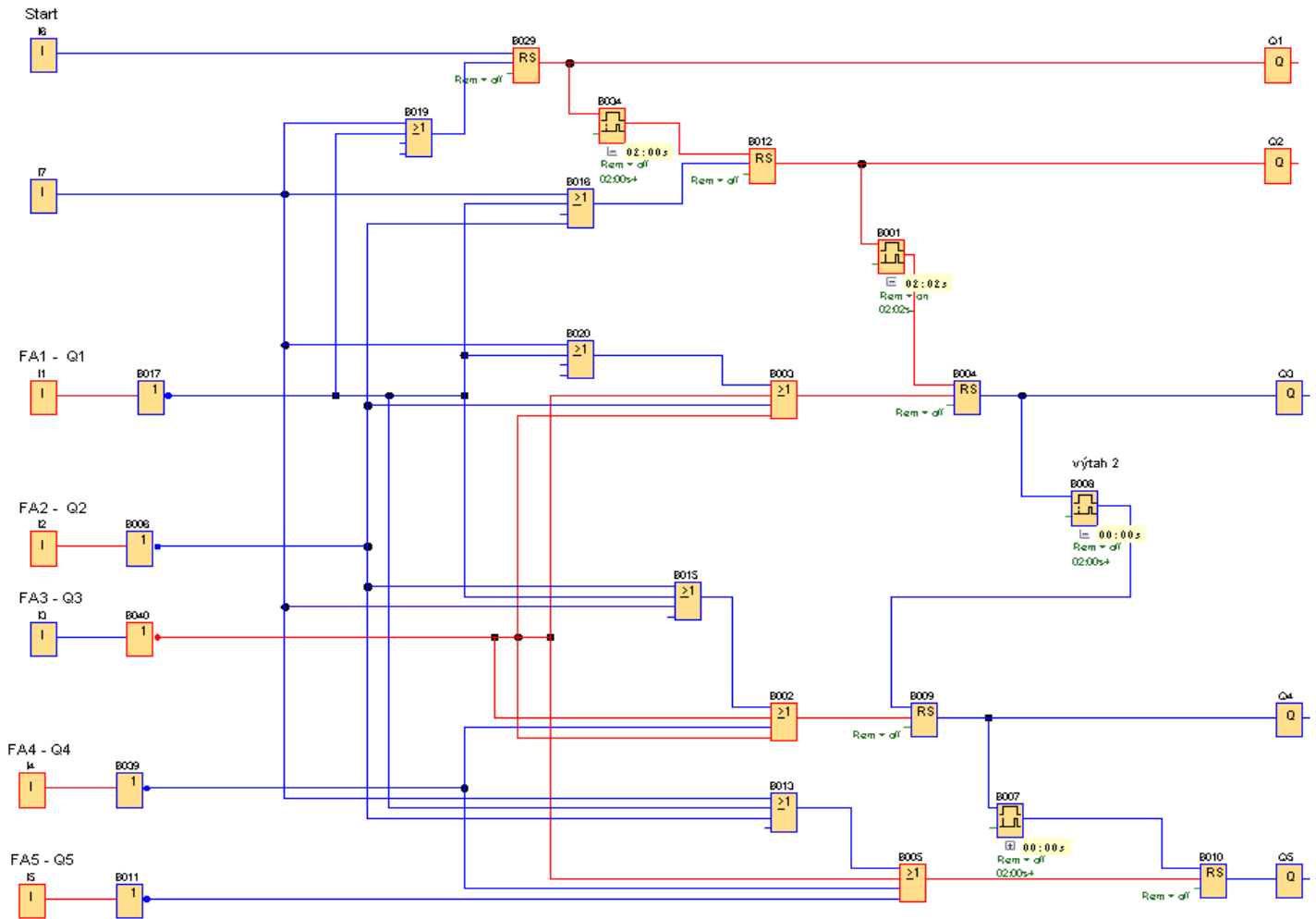
Motory za místem vypnutí motoru zůstávají v chodu.



Zapojení bloků FBD

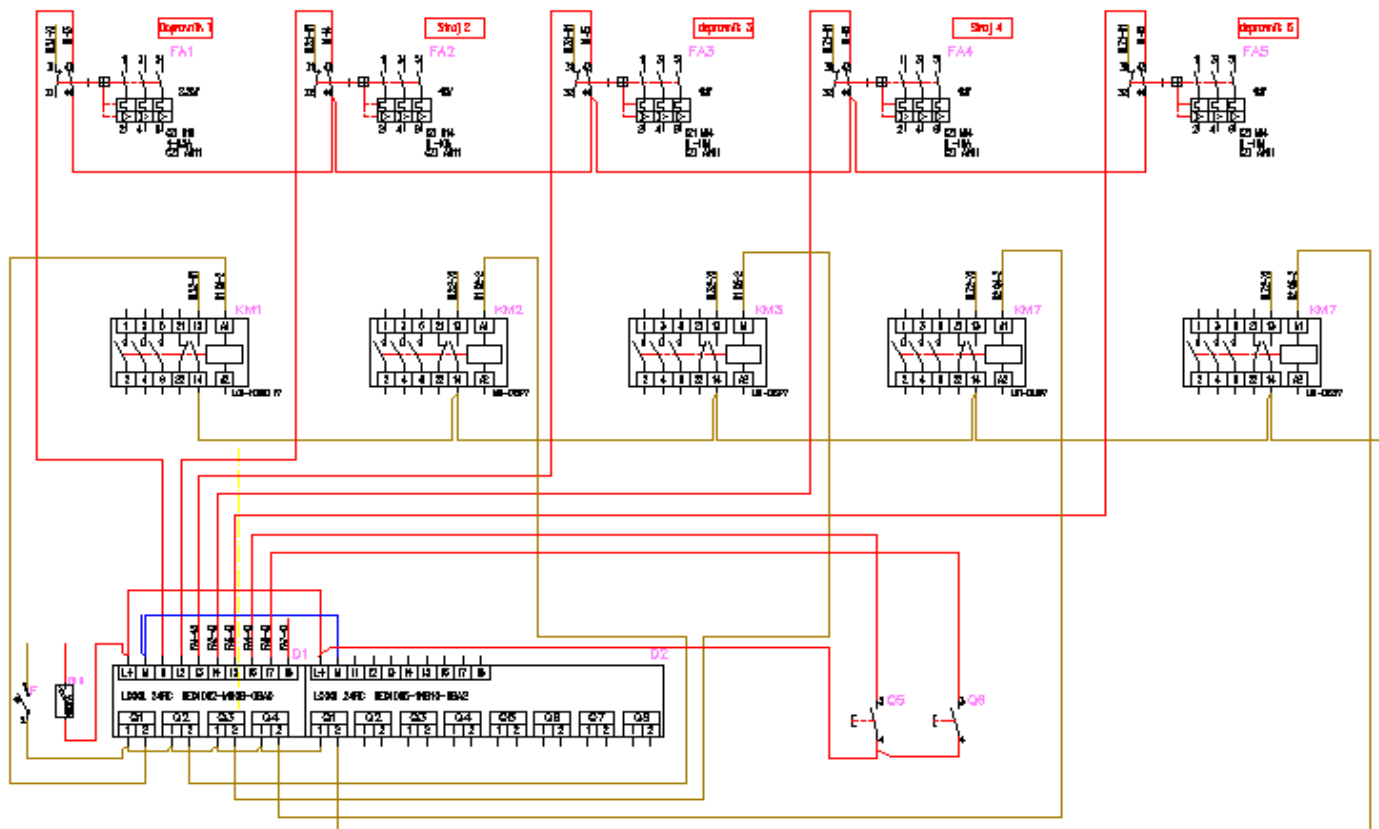


Stav při vypnutí tepelného relé FA3 pro motor Q3. Tepelné nadproudové relé má sepnuté kontakty, pokud není motor přetížen. Při přetížení dojde jejich rozpojení. Vstupy z FA (I) vypínají relé RS, a to se vypne má-li jeho vstup R hodnotu 1 (je-li sepnutý). Proto jsou vstupy z nadproudových relé negovány.



Skutečné zapojení uvedeného příkladu

V první řadě jsou motorové spouštěče (nadproudová ochrana a ochrana při zkratu), ve druhé stykače a ve třetí LOGO s rozšiřujícím modulem. Ve schématu jsou zakresleny pouze ovládací obvody.

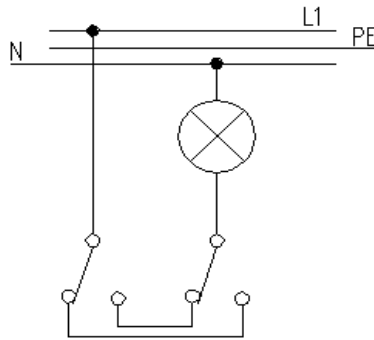


VYPÍNAČE

LOGO! Nebude asi nikdo samostatně používat jako spínač osvětlení v domě, ale uvedené jednoduché programy mohou být použity jako dílčí část složitějších programů, a to nejen pro osvětlení.

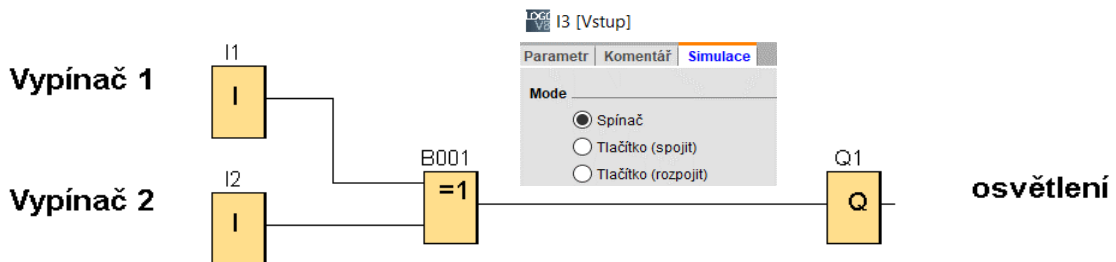
Střídavý přepínač – řazení č.6

Zapojení s vypínači



Toto zapojení umožňuje sepnout a vypnout jeden světelný obvod ze dvou míst.

Jedno z možných zapojení v LOGU

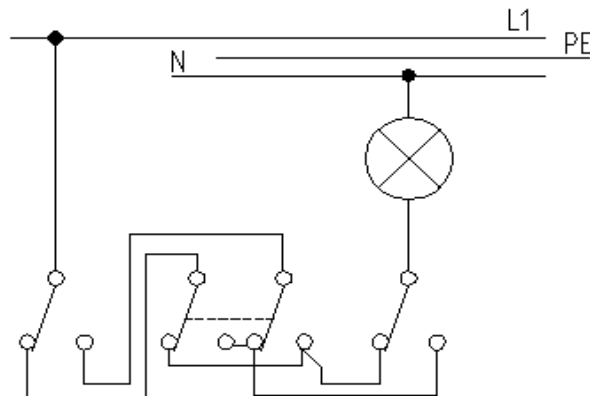


Spínače "I1" a "I2" mají výstupy zapojeny na vstupy do bloku XOR "B001". V následující tabulce jsou všechny kombinace, které mohou nastat. Výstup "Q1" bude sepnutý při rozdílných hodnotách na vstupu do "B001"

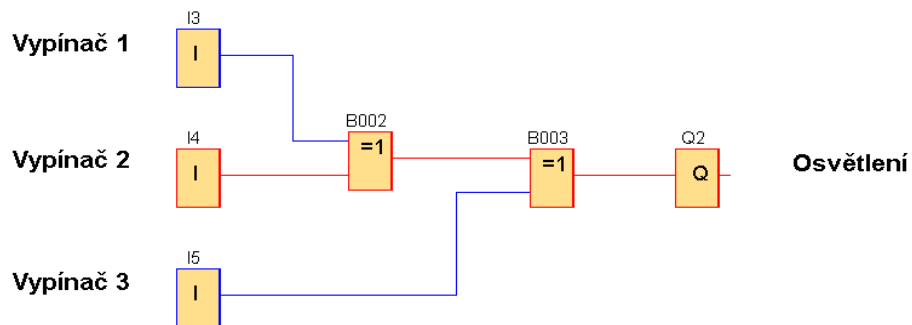
kombinace	I1	I2	Q
1	0	0	0
2	0	1	1
3	1	0	1
4	1	1	0

Křížový přepínač – řazení č.7

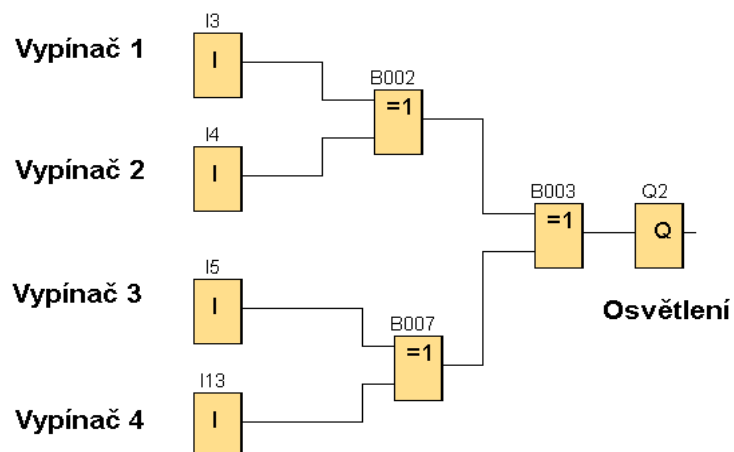
Zapojení ze tří míst pomocí dvou střídaných přepínačů a jednoho křížového přepínače.



S tímto zapojením můžeme ovládat okruh osvětlení ze tří míst. Vstupy do "B001" jsou rozdílné, proto je výstup z "B002" sepnutý. Výstup z "I5" je vypnutý. Vstupy do XOR "B003" jsou rozdílné, proto je výstup "Q2" sepnutý. Sepnutím vypínače 1 nebo 3, nebo vypnutím vypínače 2 se výstup vypne.



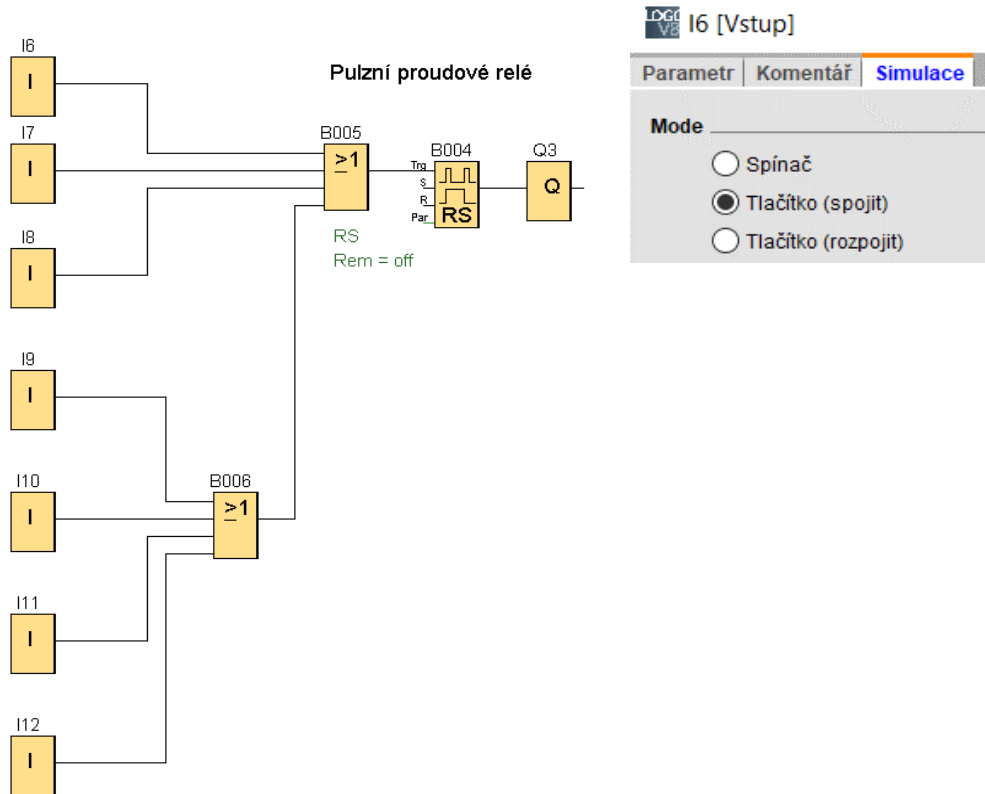
Na stejném principu můžeme provést spínání ze čtyřech míst. Kterýmkoliv vypínačem můžeme osvětlení "Q2" zapnout nebo vypnout.



Jednoduší spínání jednoho výstupu z více míst je pomocí "Pulzního proudového relé".

Stiskem libovolného tlačítka můžeme zapnout a vypnout výstup "Q3". Na příkladu je ovládání ze sedmi míst.

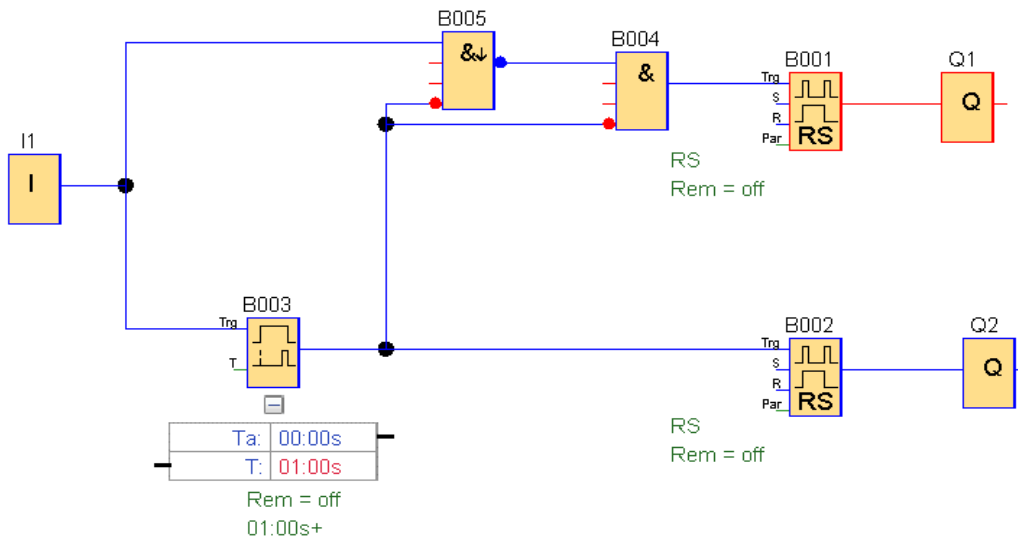
V předchozích příkladech byly na vstupech použity spínače (trvale zapnuto nebo vypnuto), v tomto případě jsou použita tlačítka (sepnuto jen při stisku)



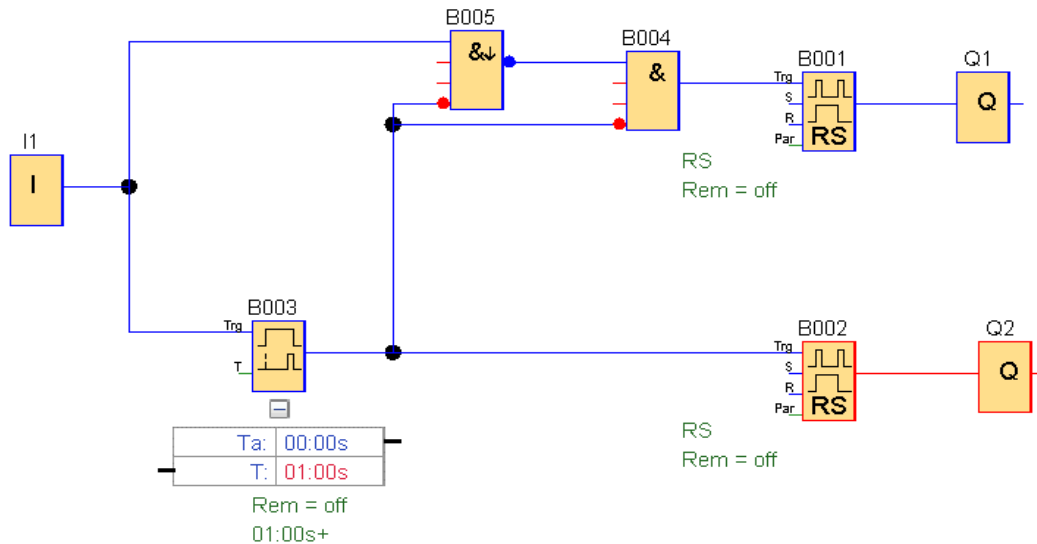
Jedním stiskem tlačítka se výstup pulzního proudového relé sepne a druhým stiskne se výstup vypne.

Samostatné ovládání dvou obvodů jedním tlačítkem

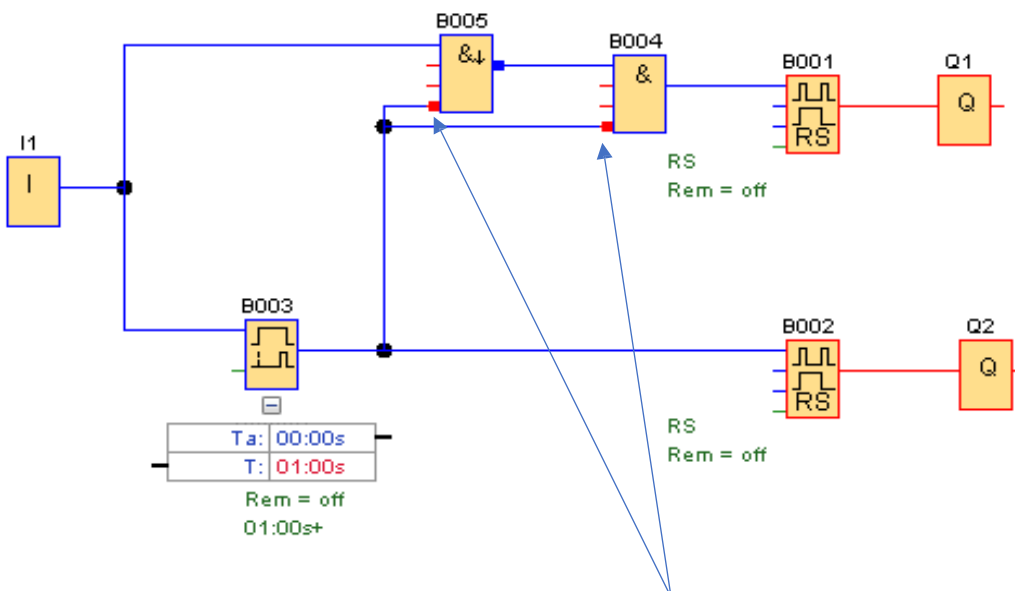
1. Krátkým stiskem (menším než 1 vteřina) se sepne výstup "Q1". Opětovným krátkým stiskem se výstup vypne.



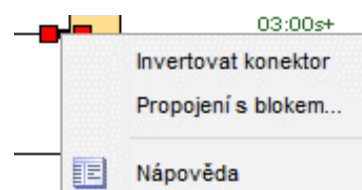
2. Stiskem delším než 1 vteřina (nastavená doba může být je libovolná) se při vypnutém výstupu "Q1" sepne výstup "Q2". Opětovným stiskem delším než 1s se výstup "Q2" vypne.



3. Sepnutí obou výstupů. Krátkým stiskem sepne výstup "Q1" a dlouhým stiskem výstup "Q2" můžeme však volit i opačný sled. Dlouhým stiskem sepne "Q2" a následně krátkým stiskem "Q1". Stejně můžeme vypnout jeden nebo oba výstupy.

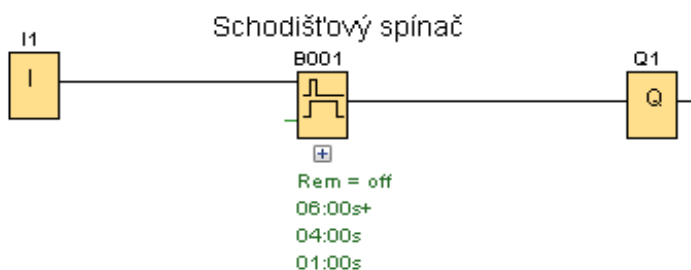


U bloku NAND (hrana) "B005" a "B004" jsou vstupy "4" negovány (klik PT na vstup bloku a Invertovat)



Schodišťový spínač

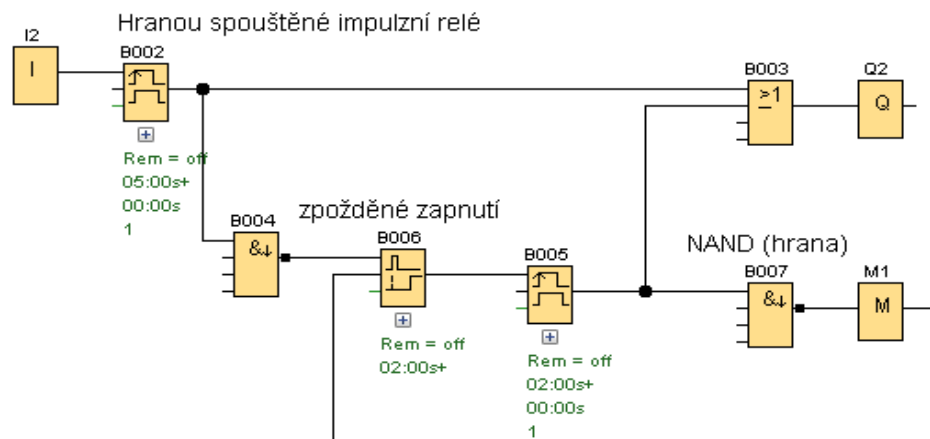
LOGO má samostatný blok "Schodišťový spínač". Po stisku tlačítka "I1" se sepne výstup, na nastavenou dobu např. 6s, světlo se rozsvítí, před ukončením časování světlo problikne (zhasne) a zase se rozsvítí, to je signál, že dojde k definitivnímu vypnutí.



„*Zpožděné vypnutí*“ je doba, za kterou světlo definitivně vypne „*Čas varovného signálu*“ je doba, za níž světlo krátce zhasne, „*Interval varovného signálu* „ je doba, kdy se světlo opět rozsvítí.

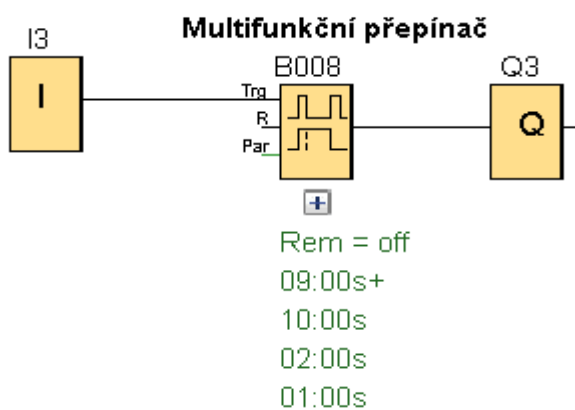
Podle nastavených hodnot světlo po rozsvícení po 4s zhasne, po 1s se rozsvítí a po další 1s zhasne trvale.

Zapojení stejné funkce jako má blok "Schodišťový spínač", ale tato funkce je složena z více časových bloků.

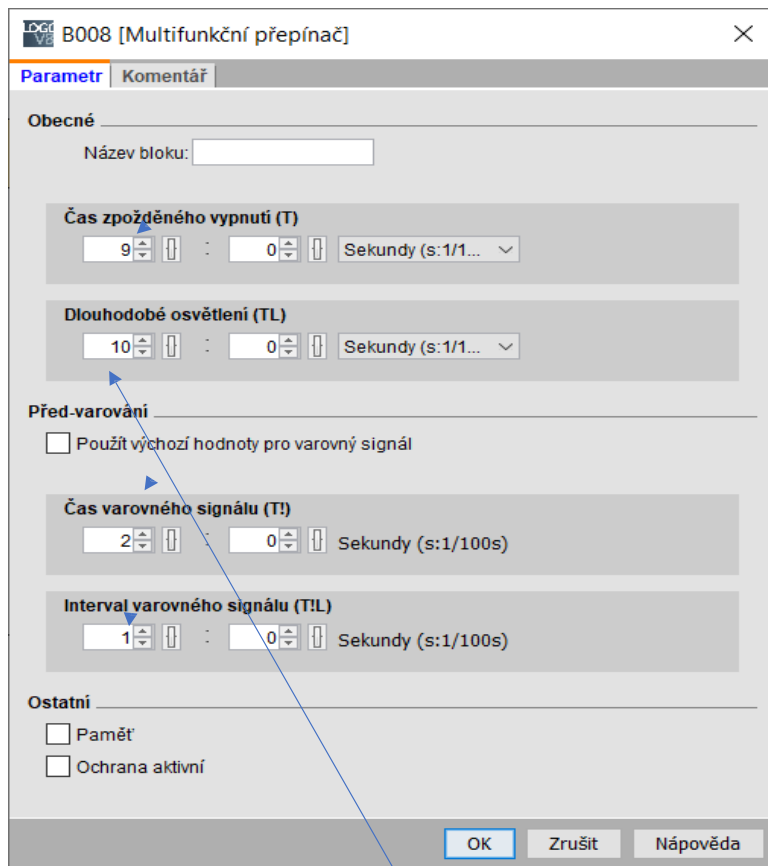


Stejnou funkci schodišťového spínače můžeme dosáhnout i multifunkčním přepínačem Vstup

"I3" musí mít funkci tlačítka.

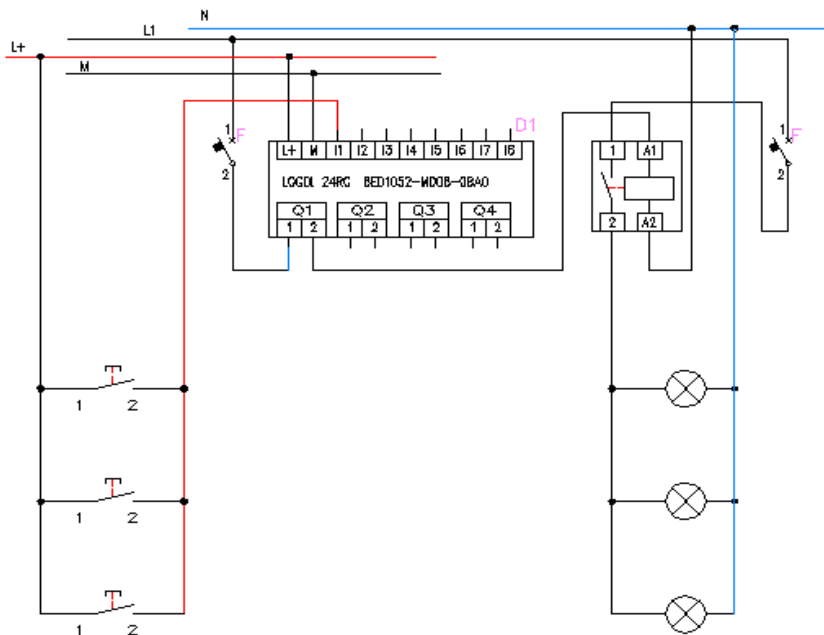


Podle následujícího nastavení výstup "Q3" trvale vypne po 9s. Dvě vteřiny před trvalým vypnutím (od začátku na 7s) výstup vypne na 1s. V osmé vteřině zapne a v deváté vteřině trvale vypne.



Hodnota dlouhodobého osvětlení musí být větší než "0"

Příklad zapojení schodišťového spínače při použití LOGA! Napájení LOGA může být i 230V.



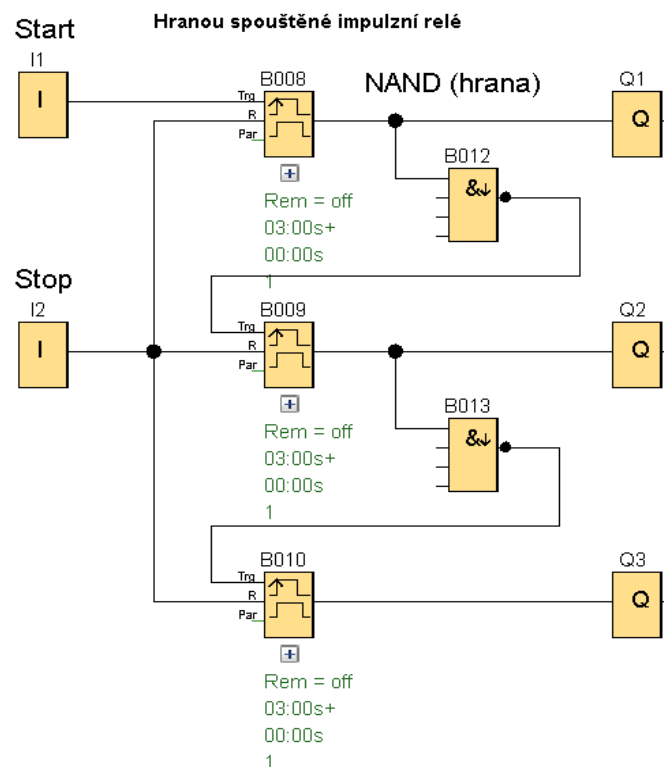
KROKOVÝ POSUN

Jedná se přechod kdy ovládání je provedeno jedním tlačítkem. Krokování je automatické a v časovém intervalu se zapínají ve vybraném pořadí výstupy, nebo druhá varianta je postupného spouštění je opětovným stiskem tlačítka. Při sepnutí výstupu se předchozí výstup vypne. Můžeme tento způsob nazvat sekvenční. Funguje to jako převodovka u motocyklu. Stupně se mění v řadě za sebou.

Časové krokování s použitím bloku "Hranou spouštěného impulzního relé"

V následujícím příkladu se po stisku tlačítka "I1" sepne výstup Q1, po 3s se vypne a sepne se výstup "Q2", ten je sepnutý také 3s poté se vypne a sepne se "Q3". Po 3s se vypne. Kdykoliv je možné výstupy vypnout tlačítkem "I2".

Stiskem tlačítka "I1" se sepne výstup Hranou spouštěného impulzního relé "B008" a relé začne časovat. Po nastaveném čase (3s) vypne. Přejdem ze zapnuto na vypnuto dá blok NAND (hrana) impulz druhému impulznímu relé "B009". Tím se sepne jeho výstup a začne časovat. Po ukončení časování se výstup vypne a blok NAND (hrana) "B013" dá impulz k sepnutí relé "B010" a tím se sepne poslední výstup "Q3" a po nastavené době (3s) se vypne.

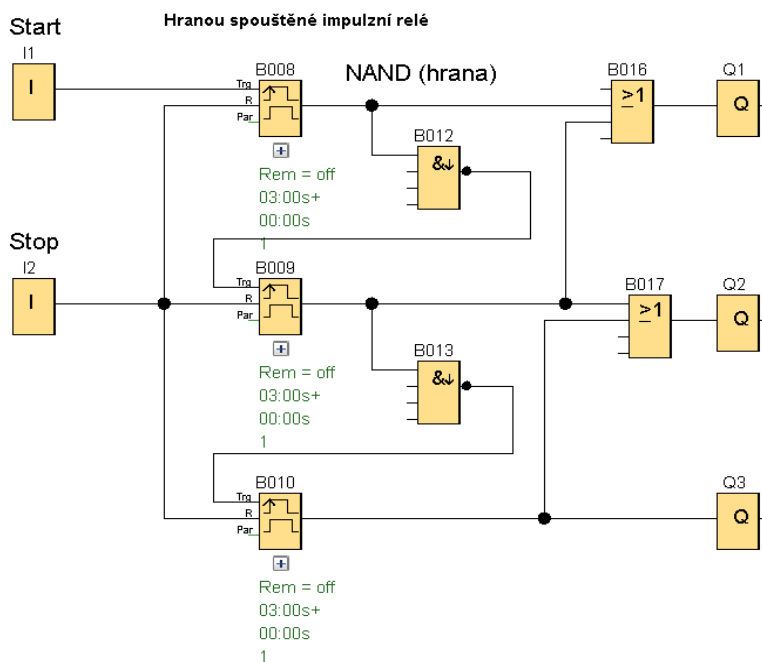


Zapnutí výstupů v jednotlivých krocích

	Q1	Q2	Q3
1.krok	1	0	0
2.krok	0	1	0
3.krok	0	0	1

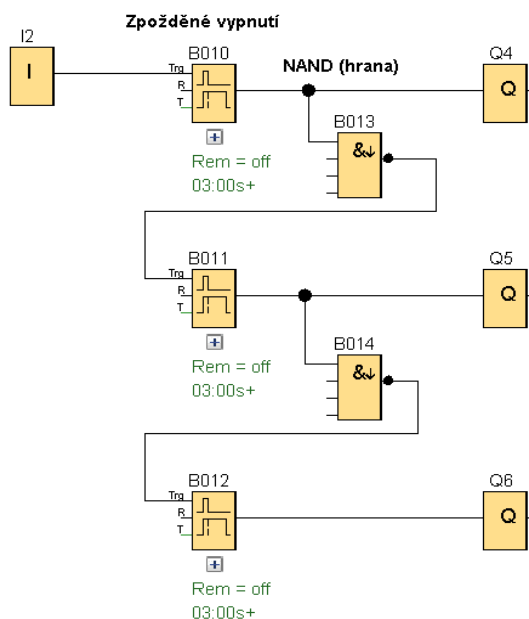
V jednotlivých krocích nemusíme spínat výstupy v pořadí, jak jdou za sebou, ale v různých kombinacích. To dosáhneme propojením výstupů z časových relé na jednotlivé výstupy,

	Q1	Q2	Q3
1.krok	1	0	0
2.krok	1	1	0
3.krok	0	1	1



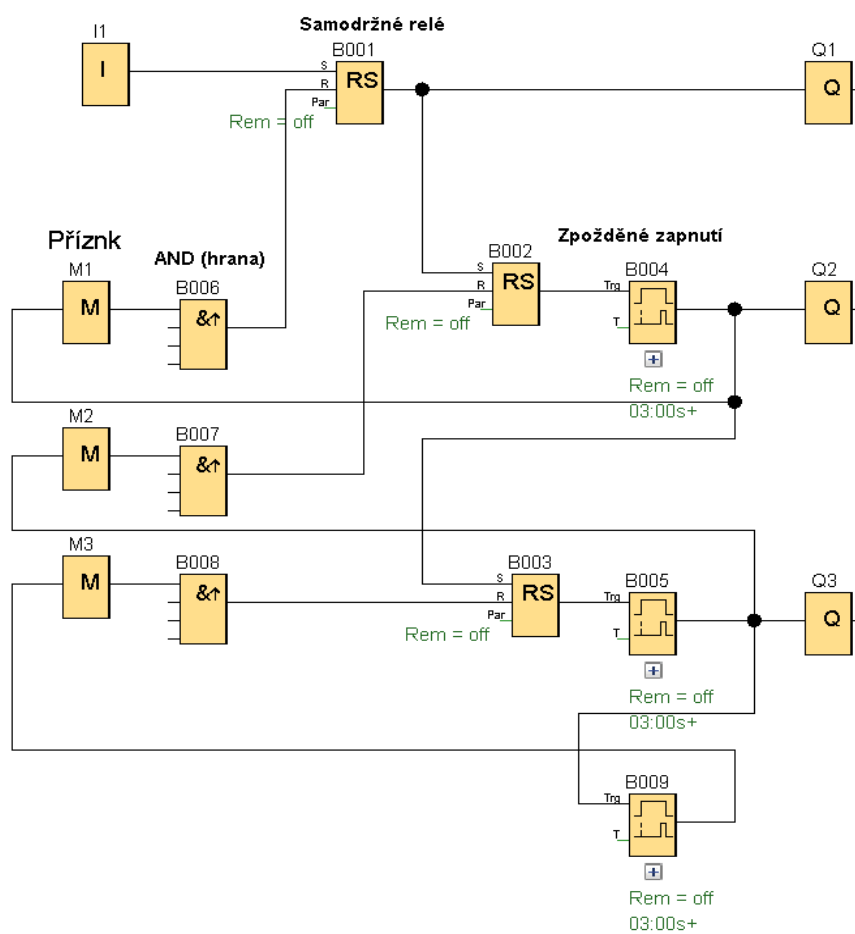
Časové krokování s použitím bloku "Zpožděné vypnutí"

Stiskem tlačítka "I2" se sepne výstup z bloku Zpožděné vypnutí "B010" a tím výstup "Q4". Po nastavené době blok "B010" vypne výstup. Vypne se výstup "Q4" a přechodem ze zapnutí na vypnutí dá blok NAND(hrana) B013 impuls na vstup Zpožděného vypnutí "B0011" a ten sepne svůj výstup. Dále se postup opakuje.



Časové krokování s použitím bloku "Zpožděné zapnutí"

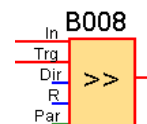
Trochu náročnější na počet bloků je varianta s použitím bloku "Samodržné relé" a "Zpožděné zapnutí". Stiskem tlačítka "I1" se sepne blok RS "B001" ten sepne výstup "Q1" a současně blok RS "B002", ten je napojen na blok Zpožděného zapnutí "B004". Blok začne odpočítávat čas do sepnutí výstupu. Po sepnutí jeho výstupu se sepne "Q2" a současně se vypne přes blok "M1" a AND (hrana) výstup z RS "B001" a tím předchodzí výstup "Q1".



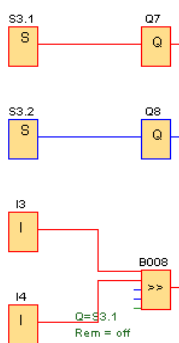
Časové krokování s použitím " posuvného registru" a "Asynchronního generátoru"

Zde je pro krokování použitý Asynchronní pulzní generátor. Doba mezi jednotlivými kroky je stejná. Stisknutím tlačítka "I2" se sepne "Q4" po době nastavené na pulzním generátoru se zapne výstup "Q5" a vypne, po uplynutí stejné doby se zapne výstup "Q6" a vypne "Q5". Po dalším sepnutím vzestupné hrany se krokování zastaví. Opětovným stiskem tlačítka "I2" se bude krokování opakovat. Doba sepnutí výstupů Q je dána součtem doby pulzu a mezi pulzy.

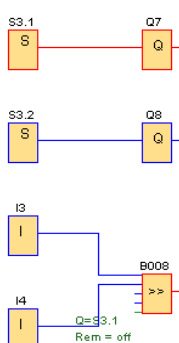
Na následujících obrázcích je zachyceno postupné spínání a vypínání dvou vstupů, závislých na stavu vstupů "In" a "Trg" do posuvného registru. Následující obrázky jsou zde pro znázornění, jak postupné spínání a vypnutí postupuje, nejsou součástí popsání programu.



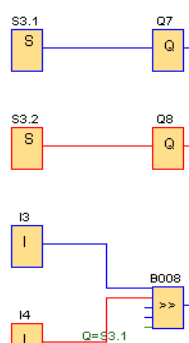
obr.1



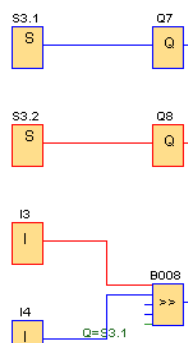
obr.2



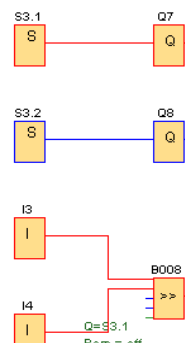
obr.3



obr.4

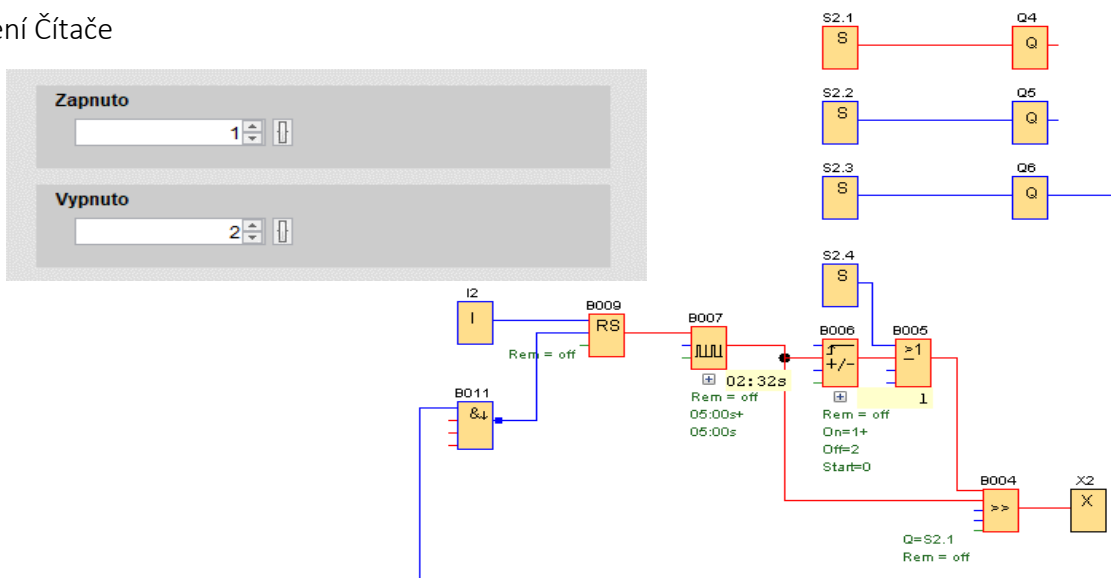


obr.5

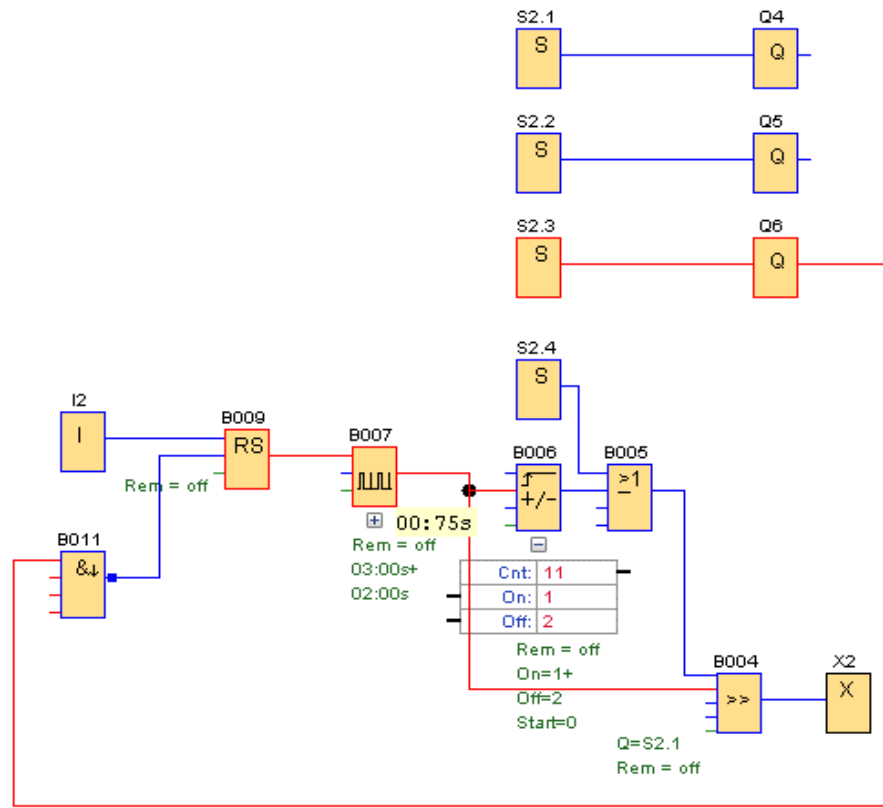


Celý program v 1. kroku. Pulzní generátor dá impuls čítači, ten načte hodnotu 1 (zapnutí výstupu). Tím se sepne vstup "In" Posuvného registru a současně se sepne vstup "Trg". Tím dojde k sepnutí prvního výstupu "Q4".

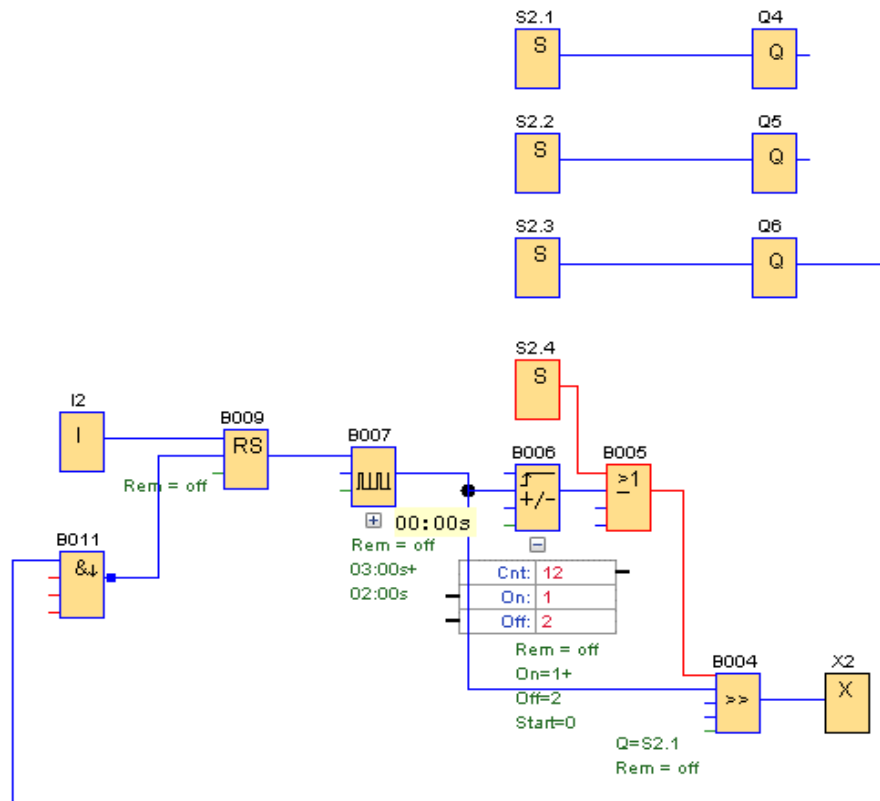
Nastavení čítače



Postup do dalšího přepnutí se opakuje. Sepnutím výstupu "Q6" se sepne vstup do bloku NAND (hrana). Současně s sepnutím Bitového registru "S2.4" se vypne výstup "Q6", tím dá blok NAND (hrana) impulz do bloku RS "B009", ten vypne výstup, ukončí se pulzování a výstupy zůstanou ve vypnutém stavu.

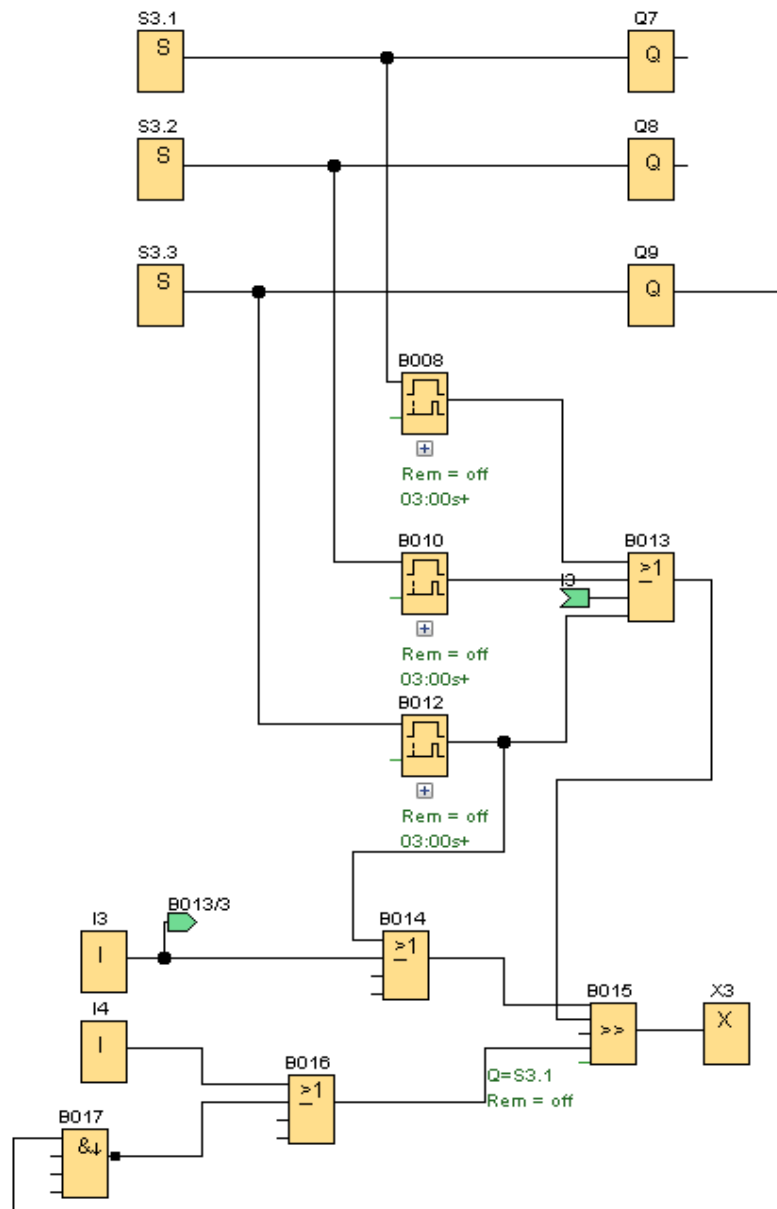


Stav po vypnutí



Časové krokování s použitím "Posuvného registru" a "Zpožděného zapnutí"

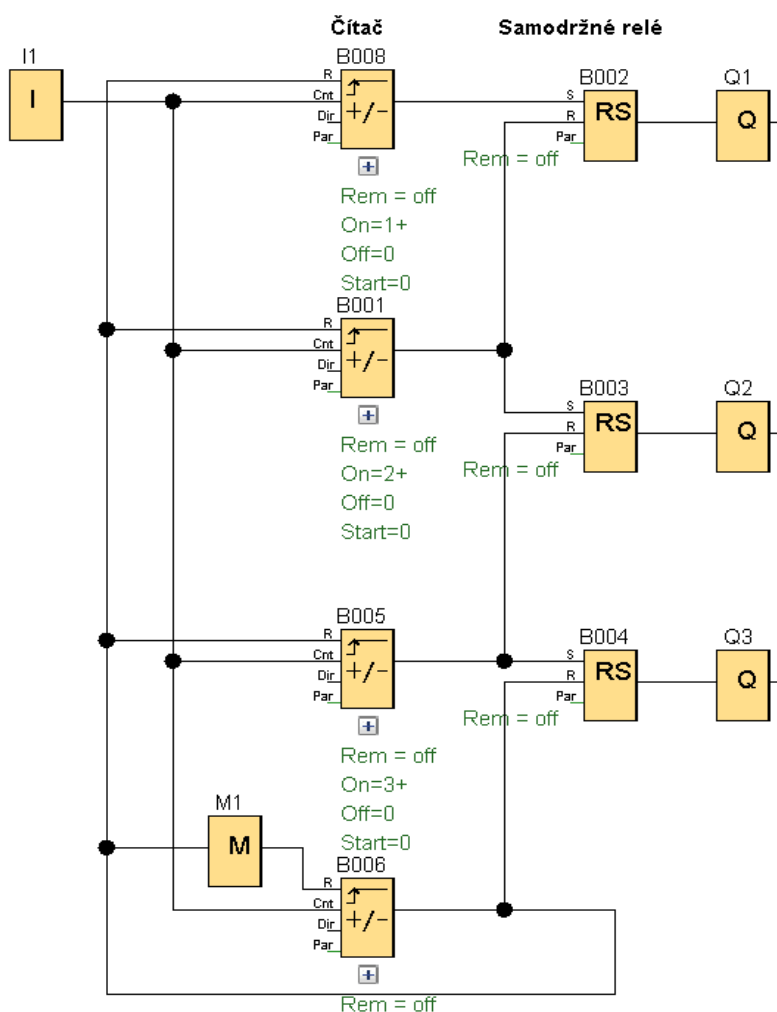
Stiskem tlačítka "I3" se sepne první výstup "Q7". Po nastavené době zpožděného zapnutí se vypne a současně se zapne výstup "Q8". Stejně je to i v dalším kroku. Po vypnutí posledního výstupu "Q9" se krokování zastaví. Krokování lze zastavit v kterékoliv fázi tlačítkem "I3".



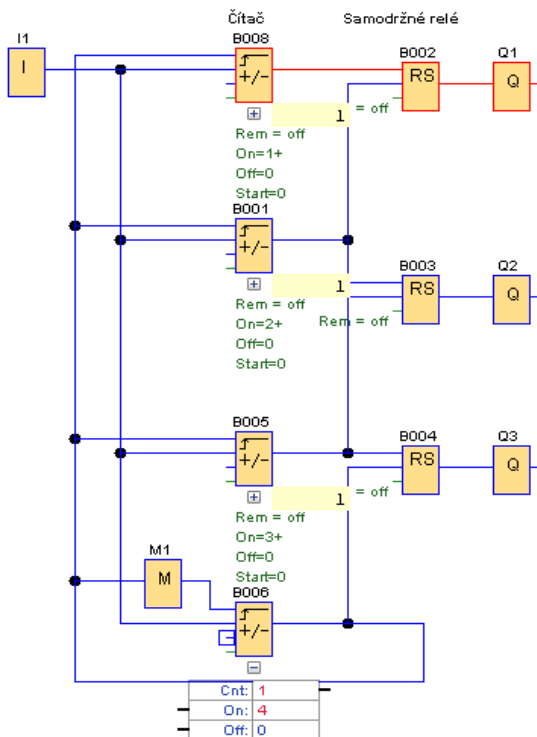
Krokování impulzem s použitím " Čítače" a "relé RS"

Při tomto způsobu krokování se posun na další krok provádí impulzem "I". V praxi to může být tlačítkem, koncovým spínačem apod. V následujícím příkladu je krokování provedeno čítačem a samodržným relé. Čítače mají v stav zapnutí nastavený v pořadí jak jdou za sebou od 1 do 4. (B008 = 1, B001 = 2, B005 = 3, B006 = 4). Při prvním stisku tlačítka se načte hodnota 1 a sepne se výstup bloku RS "B002". Při druhém stisku se načte ve všech čítačích hodnota 2, sepne se výstup druhého čítače "B001" a ten současně vypne předchozí relé RS "B002" a sepne výstup relé RS "B003". Stejným způsobem probíhá i třetí krok. Čtvrtý krok je určen k vypnutí výstupu relé RS "B004" a tím výstupu "Q3". Současně dojde k vynulování i všech čítačů.

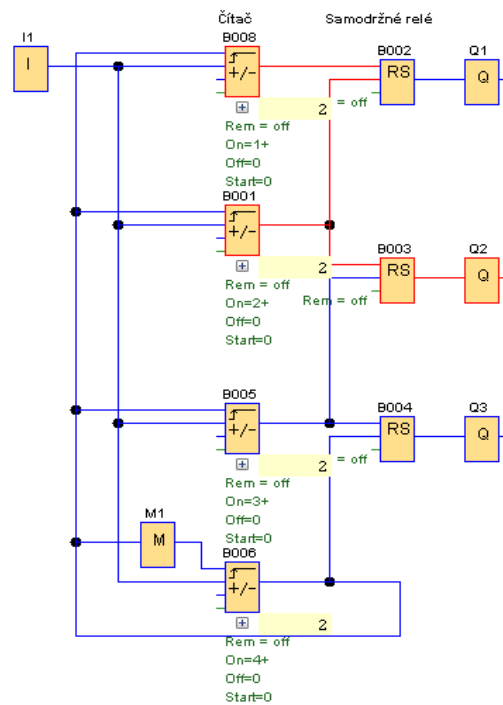
Pokud se objeví při propojování bloků nápis: ⊘ Rekurze je dovolená pouze pomocí výstupů a příznaků musí se použít Příznak M, jako v tomto případě "M1"



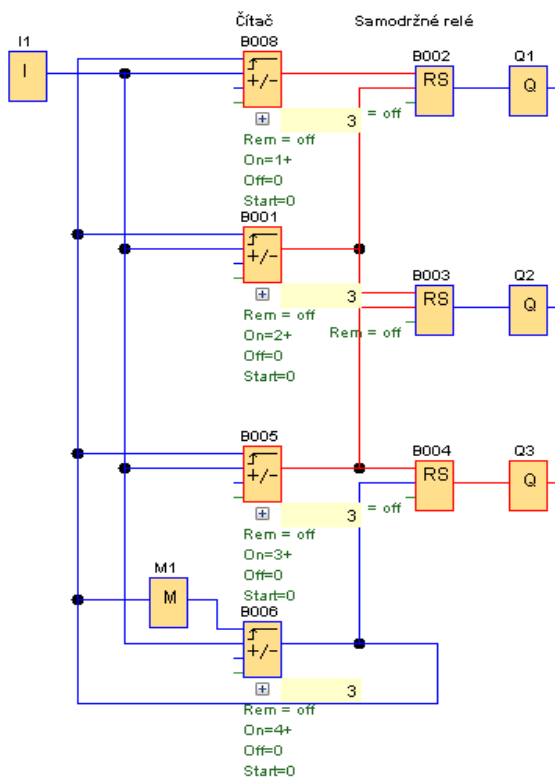
Stav v prvním kroku



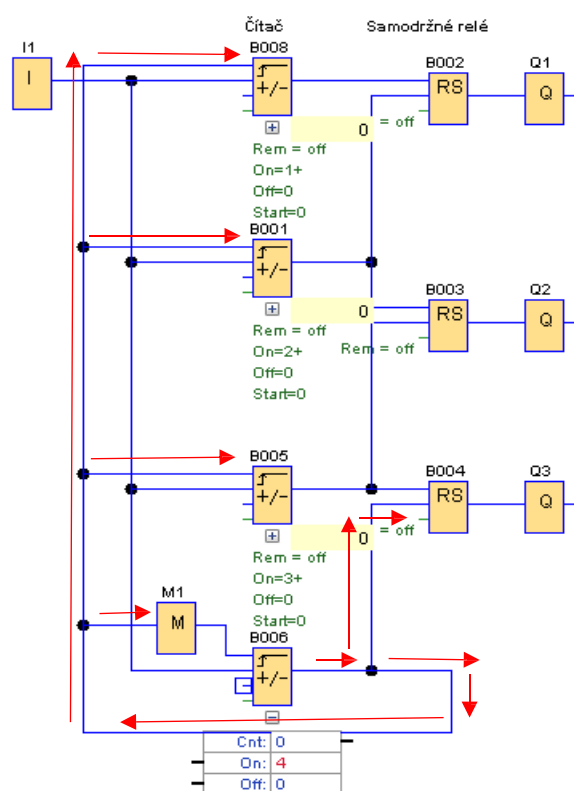
Stav ve druhém kroku



Stav ve třetím kroku



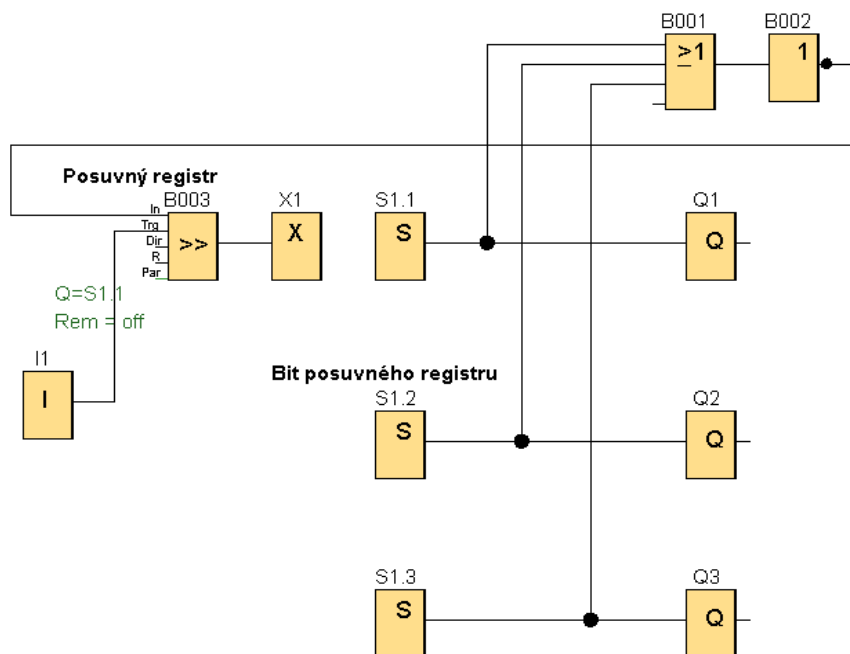
Stav ve čtvrtém kroku



Krokování impulzem s použitím "Posuvného registru".

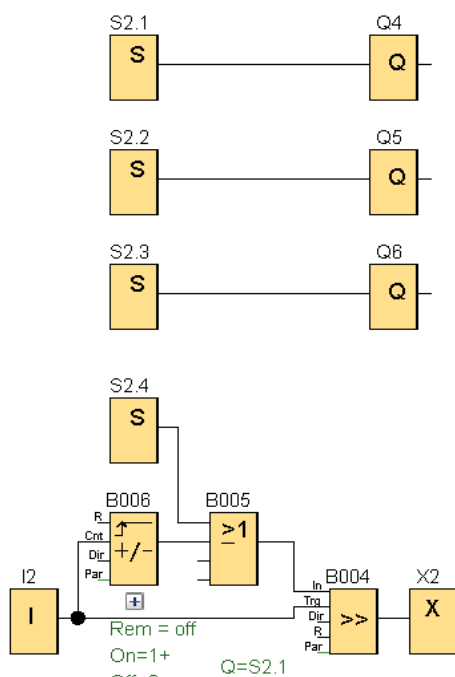
Ke krokování je určen posuvný registr v kombinaci s bitem posuvného registru.

V tomto zapojení se stiskem tlačítka "I1" sepne bit S1.1 a tím výstup "Q1". Dalším stiskem tlačítka se bit S1.1 vypne a sepne se bit S1.2. Dalším stiskem se vypne bit "S1.2" a sepne se bit "S1.3". Třetím stiskem tlačítka se bit "S1.3" vypne a tím jsou všechny výstupy "Q" vypnuty.



b) Krokování impulzem s použitím " Posuvného registru".

Další možné zapojení bloků s posuvným registrem pro výstupy "Q" podle níže uvedené tabulky.



	Q1	Q2	Q3
1.krok	1	0	0
2.krok	0	1	0
3.krok	0	0	1
4. krok	0	0	0

NEPŘETRŽITĚ OPAKUJÍCÍ SE POSTUP (Cyklický)

Běžně se s tímto postupem setkáme třeba u semaforu, kde se pravidelně střídají barvy, ale můžeme s tímto postupem přijít do styku např. ve výrobě, kde se automaticky střídají jednotlivé operace. Např. při lisování. 1. Plech se posune pod lis, 2. upevní se, 3. spustí se lis, 4. uvolní se uchycení a cyklus se opakuje.

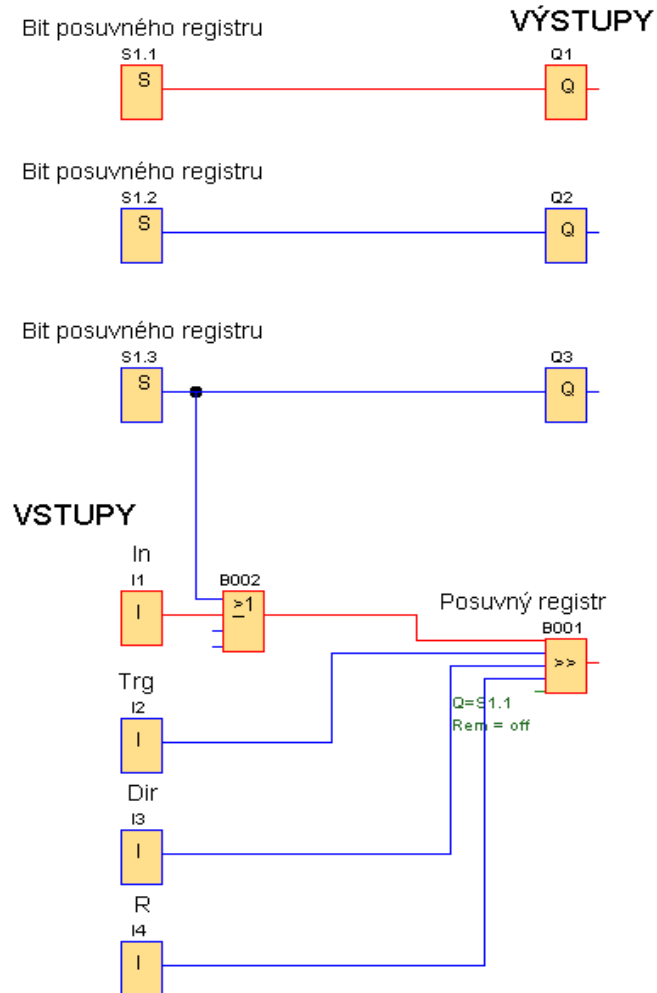
Použití posuvného registru – posun se provede stiskem tlačítka

Pro snazší pochopení je zvolen jednoduchý příklad s ovládáním pomocí tlačítek.

	I1	I2	Q1	Q2	Q3
1. krok	1	1	1	0	0
2. krok	0	1	0	1	0
3. krok	0	1	0	0	1
4. krok	0	1	1	0	0
5. krok	0	1	0	1	0
6. krok	0	1	0	0	1

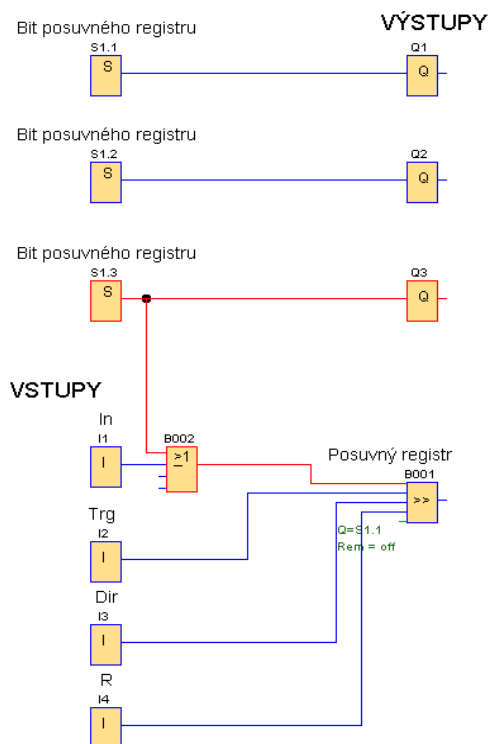
- 1.krok Sepneme spínač I1 (vstup In do posuvného registru) a sepneme tlačítko I2 (vstup Trg do posuvného registru) - tím se sepne výstup Q1.
2. krok Vypne se spínač I1 a sepne se tlačítko I2. Dojde k vypnutí výstupu Q1 a sepne se výstup Q2.

3. krok Ve třetím kroku a v dalších, zůstane vstup I1 vypnutý a sepne se tlačítko I2, tím se vypne výstup Q2, sepne se vstup Q3 a současně vstup In posuvného registru.



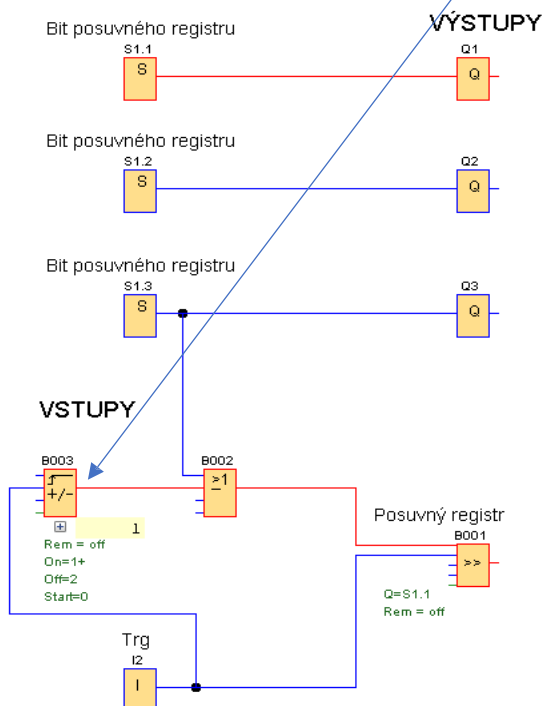
4. krok Ve čtvrtém kroku se stiskne tlačítko I2, tím se vypne vstup In do posuvného registru, vypne se výstup Q3 a současně se posune Bit na S1.1 a sepne výstup Q1. Tím se dostane cyklus na začátek a stiskáním tlačítka I2 se budou výstupy Q1 až Q3 postupně zapínat a vypínat, a to do té doby, dokud budeme klikat na tlačítko I2.

Stav při třetím kroku



Náhrada tlačítka I1 dopředným čítačem

Pro ovládání jedním tlačítkem použijeme dopředný čítač. Zapnutí nastavíme na hodnotu 1 a vypnutí na hodnotu 2. Posuv postupného sepnutí jednotlivých výstupů provádíme tlačítkem.



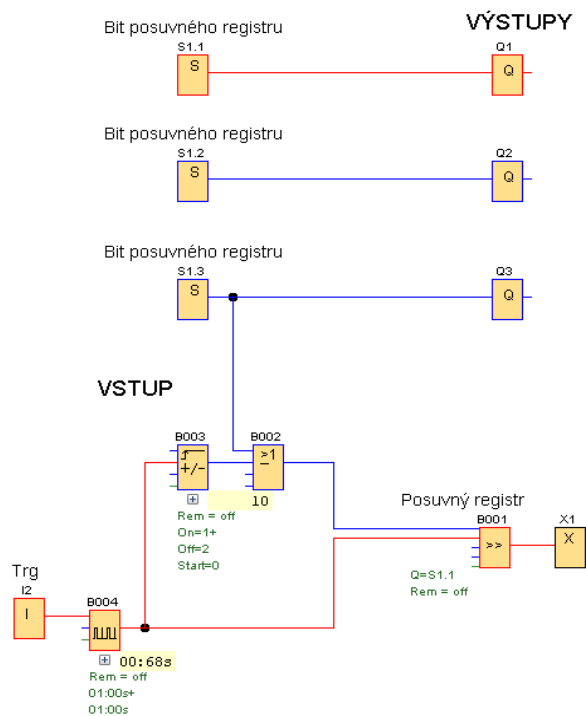
nastavení čítače

LDGr B003 [Dopředný a zpětný čítač]

Parametr	Komentář
Obecné	
Název bloku:	<input type="text"/>
Počáteční hodnota:	<input type="text" value="0"/>
Zapnuto	
	<input type="text" value="1"/>
Vypnuto	
	<input type="text" value="2"/>

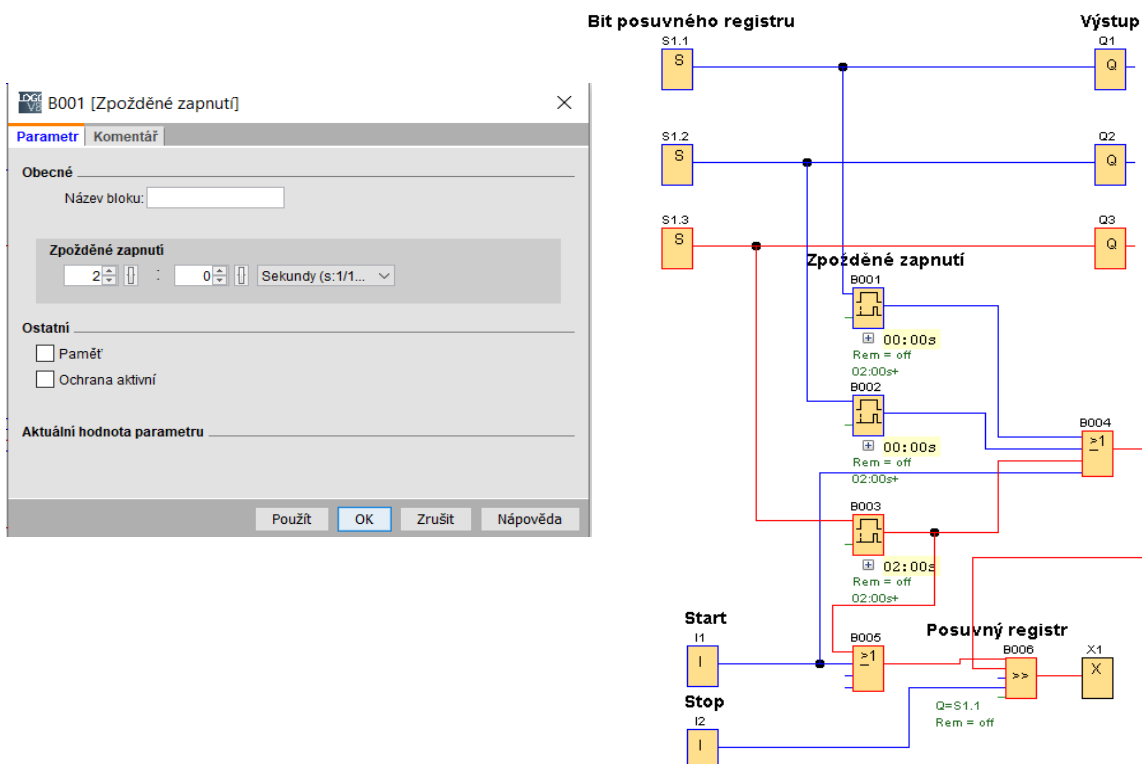
Automatický cyklický posuv pomocí asynchronního pulzního generátoru

Za spínač I2 vložíme asynchronní pulzní generátor. V tomto zapojení bude možné nastavit dobu sepnutí výstupů, které bude u všech výstupů shodná.



Automatický cyklický posuv pomocí jednotlivých časových spínačů se zpožděným zapnutím

Toto zapojení umožňuje nastavení sepnutí jednotlivých výstupů na různou dobu.



B001 [Zpožděné zapnutí]

Parametr | Komentář

Obecné

Název bloku: _____

Zpožděné zapnutí

2 : 0 Sekundy (s:1/1...)

Ostatní

Pamět

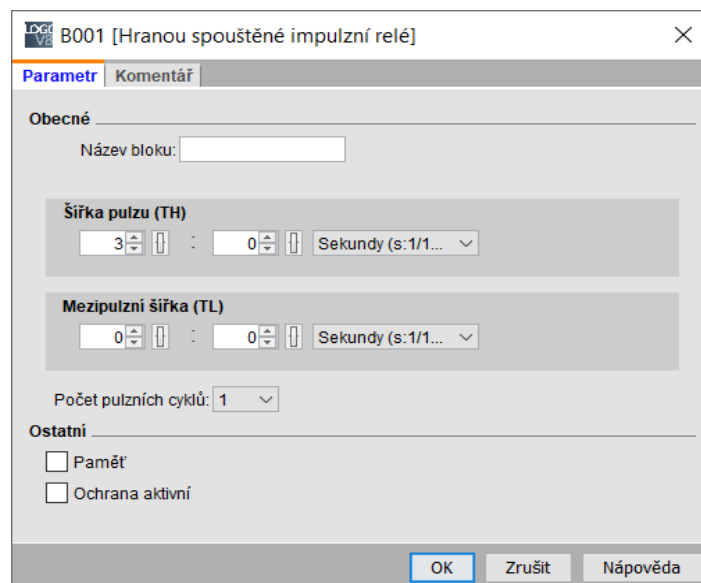
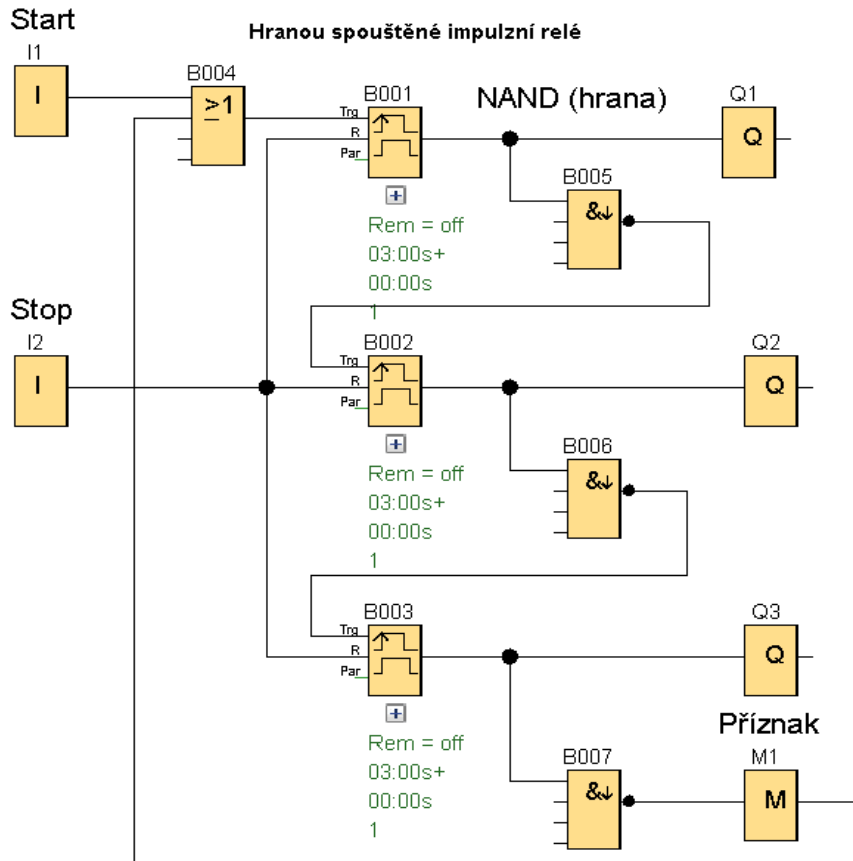
Ochrana aktivní

Aktuální hodnota parametru _____

Použít OK Zrušit Nápověda

Cyklické spínání výstupů pomocí "Hranou spouštěného impulzního relé"

Impulzem "I1" se sepne okamžitě (nastavení TL = 0) hranou spouštěného impulzního relé. To je sepnuté po dobu "TH" (v tomto případě 3s), tím je sepnutý i výstup Q1. Po vypnutí relé "B001" dá "B005" (NAND (hrana) impulz na relé "B002" to sepne po dobu "TH" výstup Q2. Stejně se sepne i výstup Q3. Po jeho vypnutí se přes příznak "M1" a blok OR "B004" opět sepne hranou spouštěného impulzního relé a tím i výstup Q1 a celý cyklus se opakuje do doby, než stiskneme "I2" Stop.



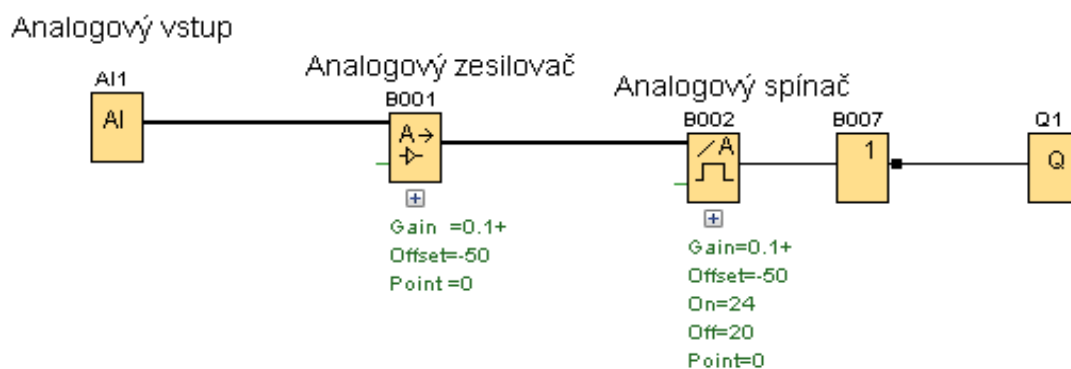
ANALOGOVÉ BLOKY

Analogové vstupy jsou spojitě napěťové a jejich hodnota se pohybuje v intervalu 0V až 10V. Je výstup může být digitální (0 nebo 1) nebo analogový (0 - 10V)

Samostatná část s použitím analogových bloků je v kapitole Aritmetické úlohy a Změna p Asi nejčastěji použijeme analogové bloky při řízení teploty.

Vytápění

Na tomto příkladu je program termostatu. Při teplotě vyšší než v tomto případě "-50°C" se sepne vytápění. Po dosažení teploty 24°C se vytápění vypne. Opětovně se zapne při snížení teploty na 2



Nastavení bloků

B001 [Analogový zesilovač]

Parametr | Komentář

Obecné

Název bloku:

Senzor

Snímač: 0 ... 10 V

Analogová nastavení

Rozsah měření	Parametr
Minimum: -50	Přírůstek: 0,10
Maximum: 50	Offset: -50

B002 [Analogový spínač]

Parametr | Komentář

Obecné

Název bloku:

Senzor

Snímač: 0 ... 10 V

Analogová nastavení

Rozsah měření	Parametr
Minimum: 0	Přírůstek:
Maximum: 1000	Offset:

Prahové hodnoty

Zapnuto

24

Vypnuto

20

Spínání bloků v průběhu stoupající a klesající teploty.

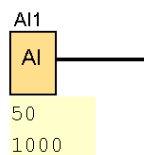
teplota	B002	Q1
$t > -50 < 24$	0	1
$t \geq 25$	1	0
$t \leq 20$	0	1

V příkladu je důležitý blok negace "B007". Výstup v bloku Analogový spínač "B002" sepne při do teploty 25°C, ale protože je vstup do bloku Výstup "Q1" negován (blok B007), je výstup "Q1" sepne do teploty 24°C a topení topí. Při teplotě 25°C a vyšší zůstane výstup "Q1" vypnutý. Při poklesu t na 20°C, blok "B002" vypne a negací vstupu do bloku "Q1" se blok sepne a topení začne topit.

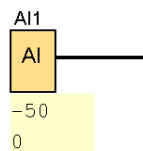
Nastavení analogového zesilovače

Na analogovém zesilovači nastavíme předpokládaný rozsah teplot, jejichž teplotu budeme měřit. Na čidle PT 100 se bude pohybovat kolem 100 Ω. S rostoucí teplotou se bude odpor zvětšovat odpovídá 119,4 Ω, pro 0°C je to 100 Ω a pro teploty pod 0°C se odpor zmenšuje, pro -50°C má hod 119,4 Ω. Odpor se pomocí převodníku převede na napětí v rozsahu 0 až 10V. V LOGU se zob hodnota 0 až 1000, což vidíme na spodním řádku Analogového vstupu. Na prvním řádku je vidět maximální hodnota nastaveného rozsahu analogového zesilovače. Po nastavení rozsahu se přepočítá a simulaci se nastavuje již požadovaná hodnota ve stupních a nemusí se nic přepočítávat.

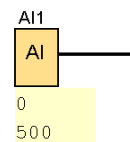
Analogový vstup



Analogový vstup



Analogový vstup



V tabulce pro nastavení Analogového zesilovače se zobrazuje kromě nastavitelných hodnot "maxi a minimum" i hodnoty "Přírůstek a Offset", ty se vypočítávají automaticky, ale funguje ti obrácen. Zadáme-li "Přírůstek a Offset" vypočítá se nám "Maximum a Minimum."

Přírůstek je podíl nastaveného rozsahu (-50 až +50 což je v absolutní hodnotě 100) a hodnotou 1000

P Přírůstek v tomto případě $100/1000 = 0,1$.

Off Offset je rozdíl od nulové hodnoty. Zde je to -50.

Ax hodnota z rozsahu 0–1000, která odpovídá odporu čidla Pt 100

A Skutečná teplota

$$A = Ax * P + Off$$

$$A = 1000 * 0,1 - 50 = 50$$

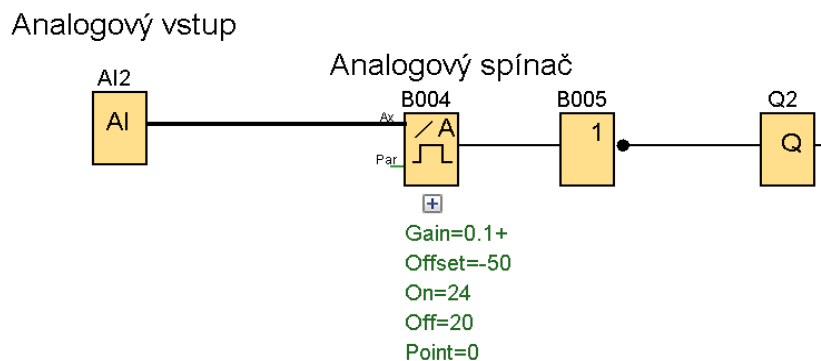
$$A = 0 * 0,1 - 50 = -50$$

$$A = 500 * 0,1 - 50 = 0$$

$$Ax = (A - Off) / P$$

$$Ax = ((25 - (-50)) / 0,1) = 750$$

Analogová zesilovač můžeme v uvedeném příkladu vynechat, nastavení rozsahu teplot se nastaví v bloku Analogový spínač "B004".



Použije-li se Analogový zesilovač ponechá se v Analogovém spínači základní nastavení min. =0 a max. = 1000. Protože vstupní hodnota by se počítala již z upraveného výstupu Analogového zesilovače.

LOGIX 2 B002 [Analogový spínač]

Parametr Komentář

Obecné

Název bloku:

Senzor

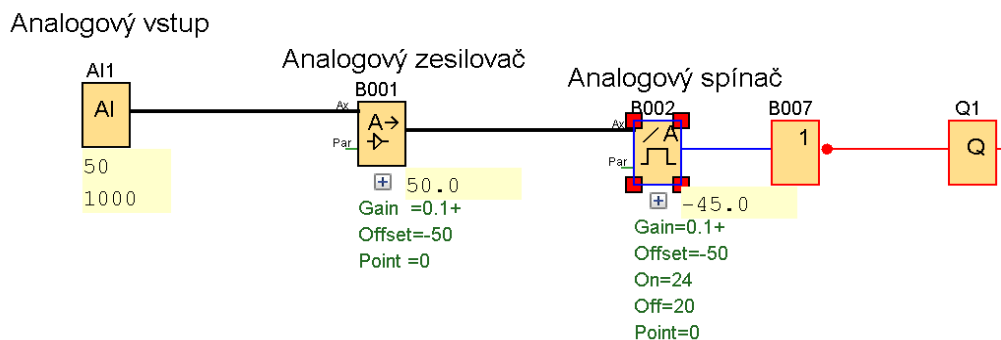
Snímač: 0 ... 10 V

Analogová nastavení

Rozsah měření	Parametr
Minimum: -50	Přírůstek: 0,10
Maximum: 50	Offset: -50

Pokud bychom nastavili Analogový zesilovač a Analogový spínač shodně podle výše uvedeného na vyjde nám na Analogovém spínači hodnota -45 °C při nastaveném maximum 50°C, protože vstup je 50.

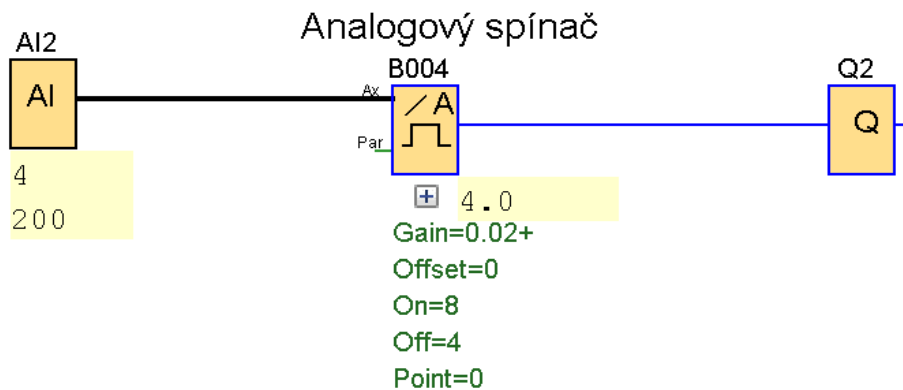
$$A = 50 * 0,1 - 50 = 45 \text{ °C}$$



Chlazení

V chladírně budeme udržovat teplotu mezi 4°C a 8°C.

Analogový vstup



Chlazení – výstup "Q2" bude spínat při dosažení teploty 9°C a vypínat až se teplota sníží na 4°C.

Logo [B004 [Analogový spínač]

Parametr Komentář

Obecné

Název bloku:

Senzor

Snímač: 0 ... 10 V

Analogová nastavení

Rozsah měření	Parametr
Minimum: 0	Přírůstek: 0,02
Maximum: 20	Offset: 0

Prahové hodnoty

Zapnuto

Vypnuto

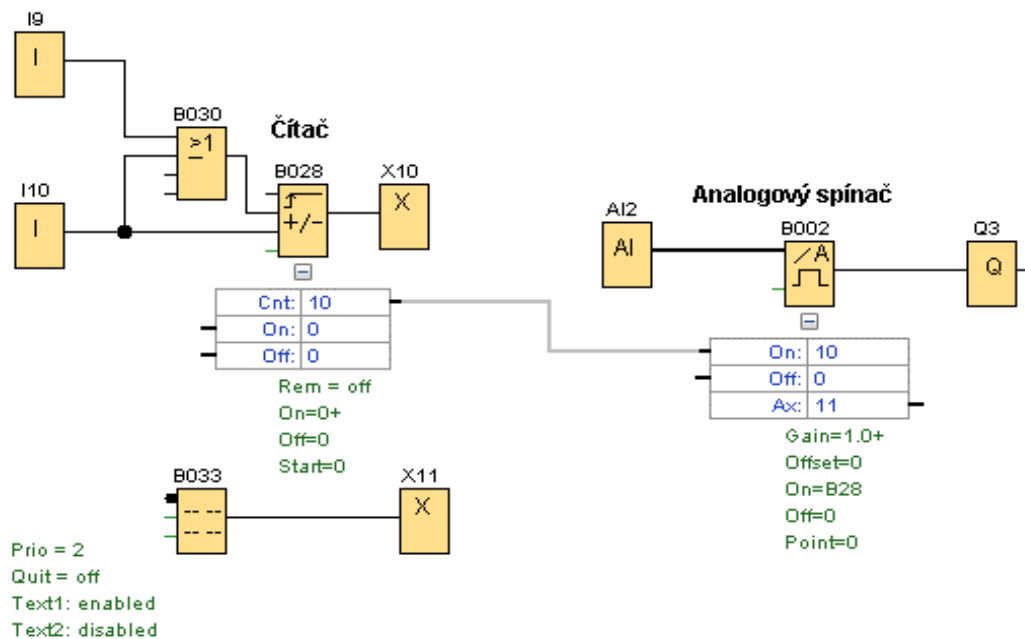
ZMĚNA PARAMETRŮ

V následujících třech příkladech je změna parametrů provedena tlačítky. V praxi je vždy důležité posoudit v jakém prostředí technologii bude kdo ji bude obsluhovat a podle toho zvolit ovládání. Tlačítka vyhovují horším podmínkám i méně technicky založené obsluze. Pokud se využije pro sledování displej LOGA je výhodné ho mít v rozváděči s průhledným krytem.

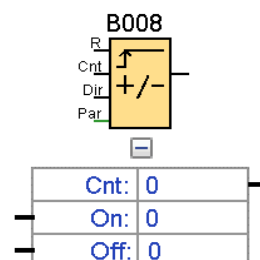
Ovládání klikáním na tlačítka

V příkladu je použitý "Analogový spínač". U něj se pomocí tlačítek mění hodnota parametru pro zapnutí výstupu. V příkladu je nastavena pomocí čítače na hodnotu 10. Analogovým vstupem "AI2" se mění hodnota vstupu (např. teplota dosáhla 11°C). Protože je dosažená teplota vyšší, než nastavená došlo k sepnutí výstupu. Vypnutí je v tomto případě nastaveno na nulu. Při poklesu "AI2" na 0°C se výstup vypne.

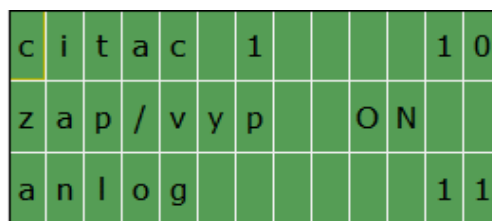
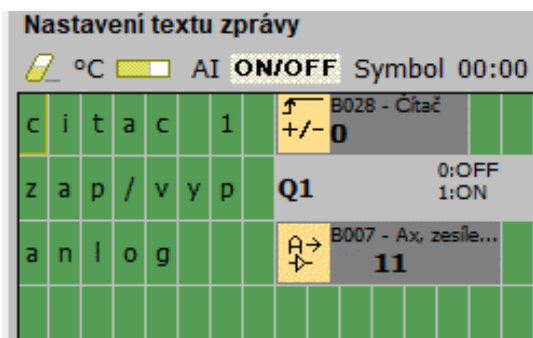
Změnu parametru měníme propojením parametru "Cnt" Dopředného a zpětného čítače s parametrem "ON" Analogového spínače. Klikáním na tlačítko "I9" se hodnota čítače zvyšuje a klikáním na "I10" se hodnota snižuje.



Tlačítko "I9" je zapojeno na vstup čítače Cnt – impulzem se hodnota zvyšuje. Tlačítko "I10" je zapojeno do vstupu "Cnt" a "Dir". Při hodnotě 1 na obou vstupech "Dir" přepne čítač na odečítání a sepnutím (1) "Cnt" se odečte hodnota čítače o jednu.



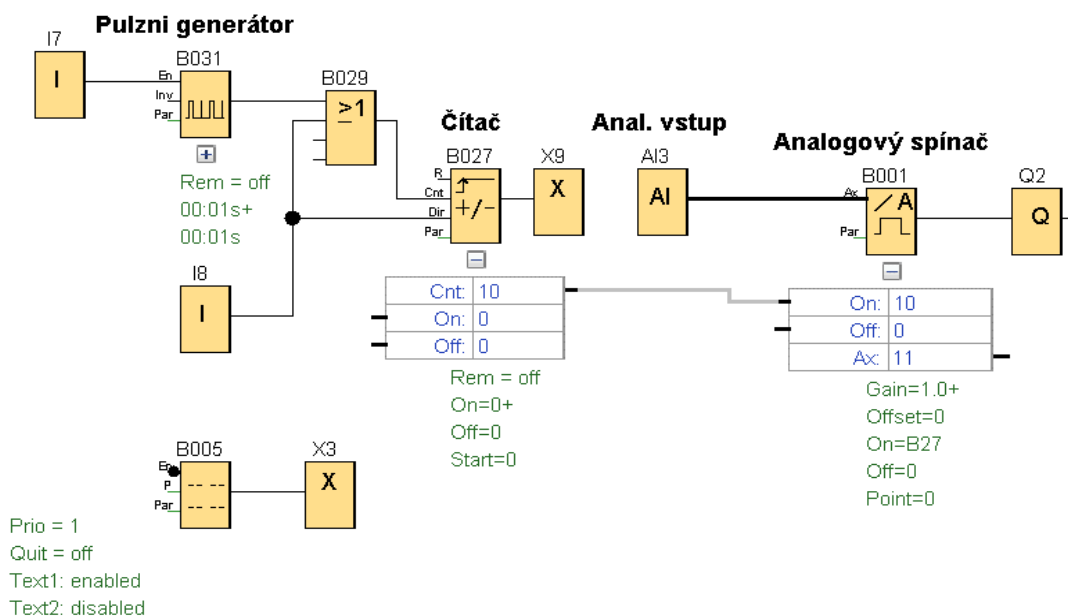
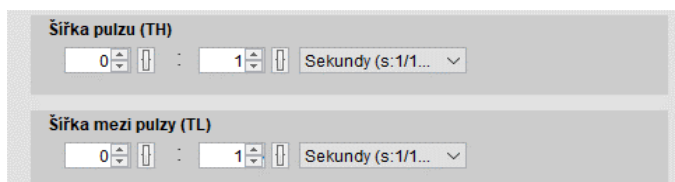
Nastavení textu zpráv



1. Automatické zvyšování hodnoty čítače držením tlačítka "I7" a snižování hodnoty klikáním na tlačítko "I8"

Funkce příkladu je shodná s předchozím. Rozdíl je vložení Asynchronního pulzního generátoru za tlačítko "I7". Držením tlačítka je aktivní pulzní generátor a střídáním sepnutí a vypnutí výstupu se automaticky zvyšuje hodnota čítače.

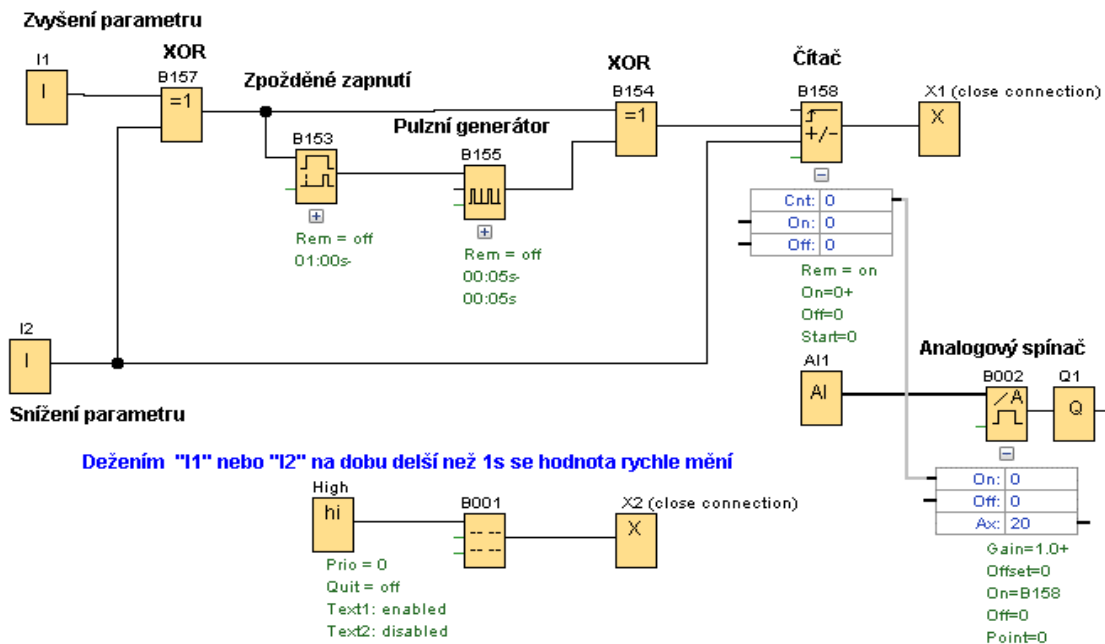
Nastavení pulzního generátoru



2. Klikání na příslušné tlačítko se zvyšuje nebo snižuje hodnota čítače. Pokud tlačítko podržíme na dobu delší než jednu 1s (podle nastavení čas. spínače) bude se hodnota čítače měnit rychle automaticky.

Výstup bloku XOR je sepnutý, mají-li vstupu různou hodnotu. Krátkým stiskem tlačítka "I1" projde jeho hodnota "1" do čítače, přes dva bloky XOR, protože jejich druhé vstupy jsou "0". Čítač zvýší hodnotu o jednu. Při stisku "I2" se na vstup Dir čítače dostane hodnota "1" (odečítání) a současně se hodnota (1) dostane na vstup Cnt a hodnota čítače se o jednu sniží.

Dlouhý stisk – Držením tlačítka "I1" začne časovat zpožděné zapnutí "B153" po sepnutí výstupu se sepne asynchronní pulzní generátor a ten bude zvyšovat hodnotu čítače. Držením tlačítka "I2" dojde přes vstup Dir čítače k přepnutí do odečítání a pomocí asynchronního pulzního generátoru se bude měnit vstup Cnt čítače a hodnoty čítače se budou snižovat.

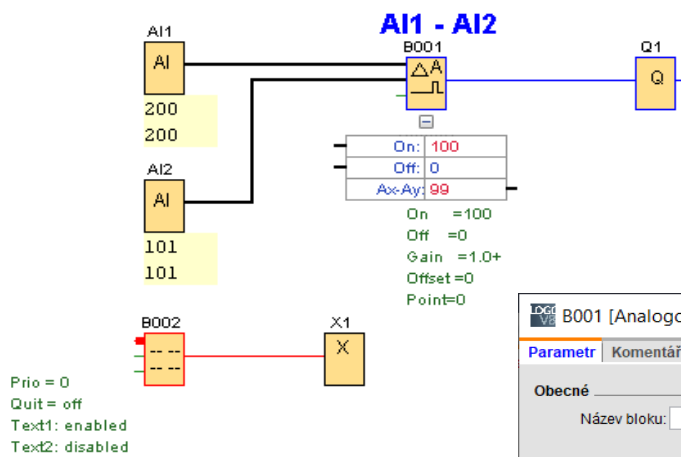


ARITMETICKÉ ÚKONY

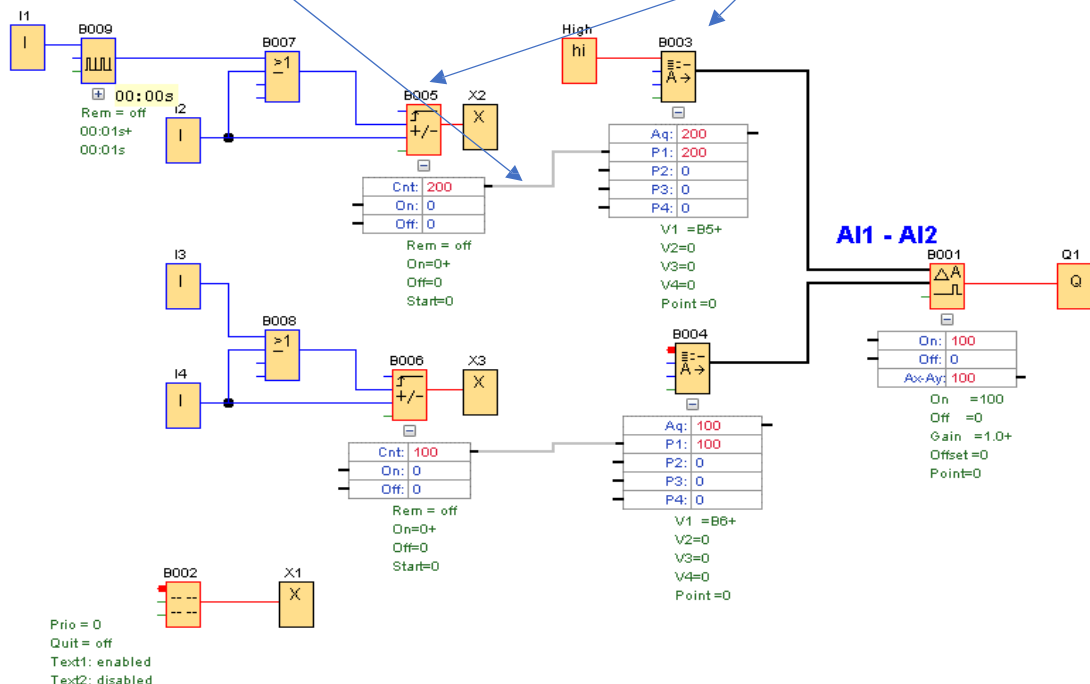
Pomocí analogových bloků je možné provést aritmetické operace sčítání, odečítání, násobení a dělení, které se mohou v některých aplikacích hodit.

Odečítání

Pro odečítání použijeme analogový komparátor. Samotný blok má tuto funkci ve svém programu. Dělá rozdíl mezi vstupem Ax a Ay. Na obrázku je Ax = 200, Ay = 101, rozdíl je 99, protože je hodnota rozdílu nižší než 100 výstup je vypnutý. Při dosažení rozdílu 100 se výstup sepne. Sepnutý výstup se vypne při rozdílu Ax - Ay = 0. Bude-li nastaven na jinou hodnotu vypne se při dosažení tohoto rozdílu. Hodnoty Ax a Ay se mění analogovým vstupem

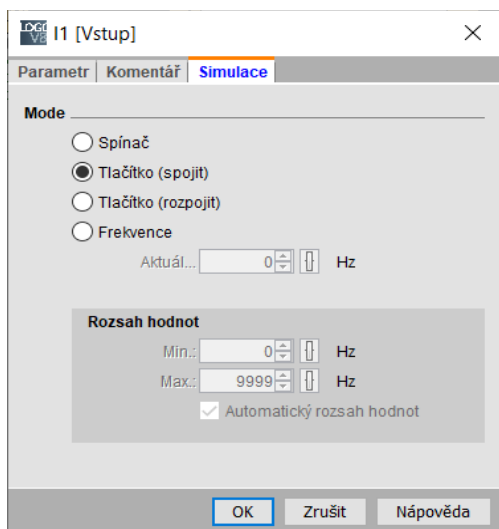


Chceme-li mít na vstupu digitální hodnoty vložíme do programu blok "Analogový MUX" a čítač. Parametry čítače propojíme s parametry analogového MUX.

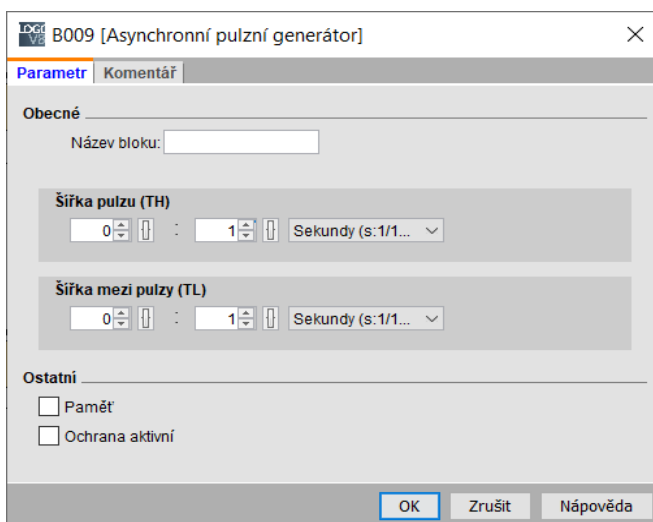


V tomto příkladu je zapojený asynchronní generátor pulzů "B009", a to z důvodu, abychom nemuseli klikat na tlačítko "I1", ale stačí ho podržet a hodnoty se budou rychle zvyšovat. Překročíme-li požadovanou hodnotu, snížíme ji postupně klikáním na tlačítko "I2".

Nastavení tlačítka

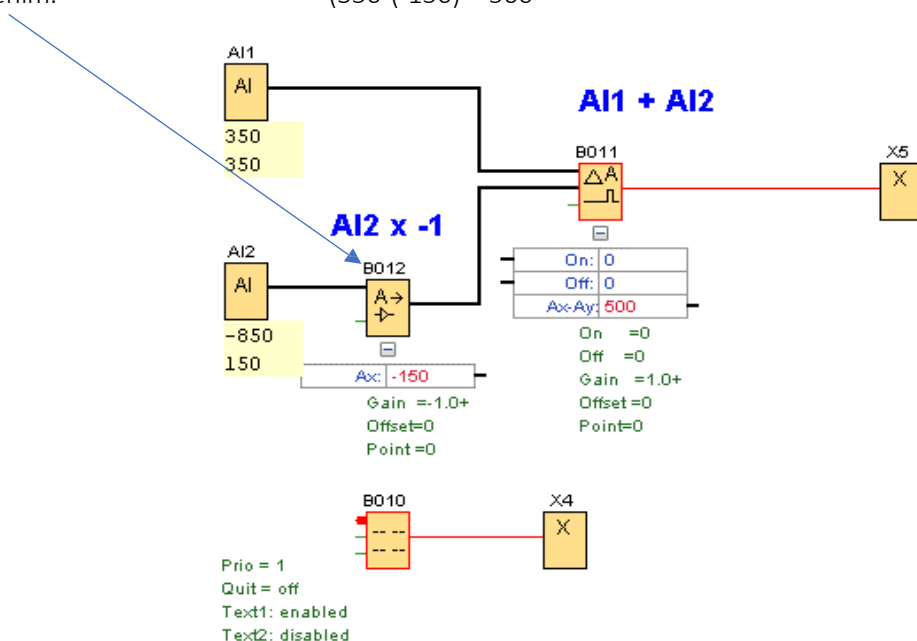


Nastavení pulzního generátoru



Sčítání

Pro sčítání se použije opět "Analogový komparátor", ale protože se hodnoty Ay odečítají od hodnoty Ax, musí mít hodnoty Ay záporné hodnoty, to se dosáhne vložením "Analogového zesilovače" a jeho nastavením.
 $(350 - (-150)) = 500$



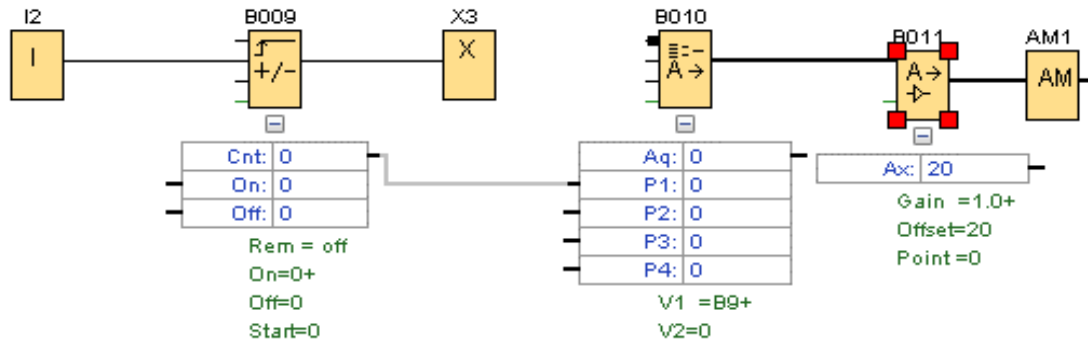
Nastavení "Analogového zesilovače" pro operaci sčítání

sečtení hodnot 350+150

A x									3	5	0	
A y									-	1	5	0
A x -	A y								5	0	0	

Přičítání konstantního čísla

Zapojení zůstane stejné, pouze se změní hodnota parametru "Minimum" u analogového zesilovače.



V tomto případě přičítáme hodnotu čítače ke konstantnímu číslu "20"

$$15 + 20 = 35$$

B011 [Analogový zesilovač]

Parametr Komentář

Obecné

Název bloku:

Senzor

Snímač: 0 ... 10 V

Analogová nastavení

Rozsah měření	Parametr
Minimum: 20	Přírůstek: 1,00
Maximum: 1020	Offset: 20

Desetinná místa

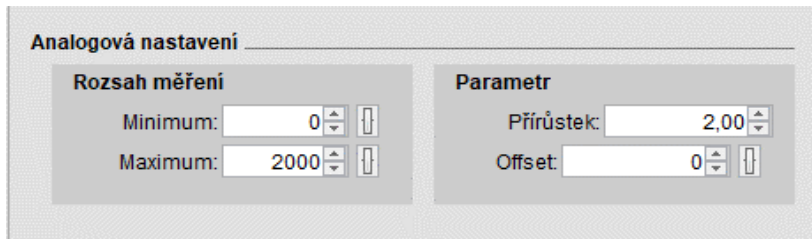
Počet desetinných míst v textu zprávy: 0 +12345

Ostatní

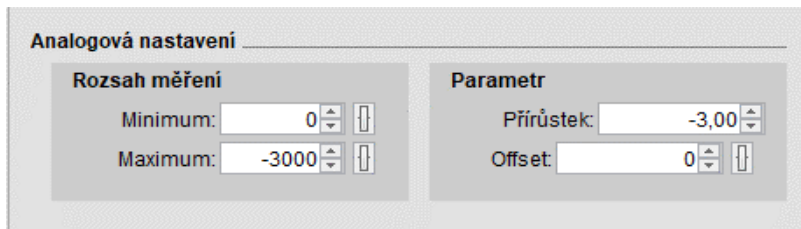
Ochrana aktivní

OK Zrušit nápověda

c	i	t	a	c																1	5	
m	i	n	i	m	u	m															2	0
v	ý	s	l	e	d	e	k														3	5



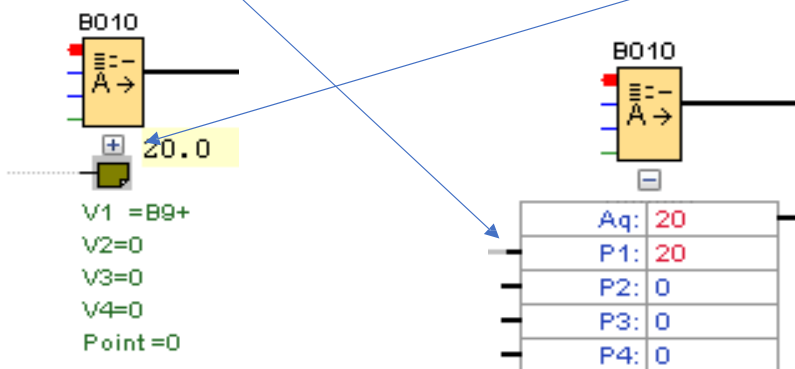
Nastavení analogového zesilovače B002



$$[20 \times 2] - [5 \times (-3)] = 40 + 15 = 55$$

c	i	t	a	c	1				2	0	
c	i	t	a	c	2					5	
A	x								4	0	
A	y								-	1	5
A	x	-	A	y					5	5	

Analogový MUX "B032" a "B026" převádí digitální hodnoty na analogové. Propojení mezi digitálním blokem (čítačem) je provedeno pomocí parametrů (šedá spojovací čára). Kliknutím na "+" bloku se rozvine tabulka s výstupy.



Nastavení displeje

c	i	t	a	c	1	B027 - Čítač			
						+/-	10		
c	i	t	a	c	2	B028 - Čítač			
						+/-	20		
A	n	a	l	o	g	1	B001 - Ax, zesíle...		
						A→	30		
A	n	a	l	o	g	2	B002 - Ax, zesíle...		
						A→	40		
V	y	s	l	e	d	e	k	B003 - Aq zesílené	
						+=	100		

2. Ve stejném příkladu provedeme součet hodnot čítačů a vydělíme ho rozdílem analogových hodnot. $(V1+V2)/(V3-V4) = (70+30)/(60-10)=2$

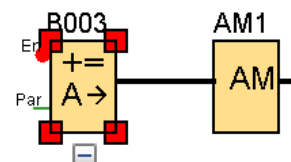
V1
 B032 [Analogový MU...]
 Operátor 1: + Priorita 1: H

V2
 B026 [Analogový MU...]
 Operátor 2: ÷ Priorita 2: L

V3
 B001 [Analogový zes...]
 Operátor 3: - Priorita 3: M

V4
 B002 [Analogový zes...]

c	i	t	a	c	1				7	0
c	i	t	a	c	2				3	0
A	n	a	l	o	g	1			6	0
A	n	a	l	o	g	2			1	0
V	y	s	l	e	d	e	k			2



Aq:	2
P1:	70
P2:	30
P3:	60
P4:	10

V1=B32+
 V2=B26
 V3=B1
 V4=B2
 Point=0
 (B32+B26)/(B1-B2)

Dole pod blokem vidíme vzorec, podle kterého probíhá výpočet

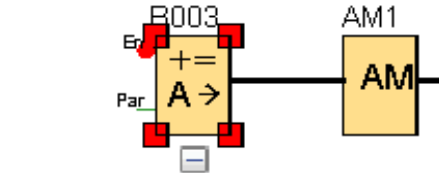
3. Kombinací Operátorů a Priorit nastavíme početní úkon $(V1/V2) + (V3 \times V4)$
 $(20/10) + (10 \times 3) = 32$

V1
 B032 [Analogový MU...]
 Operátor 1: ÷ Priorita 1: M

V2
 B026 [Analogový MU...]
 Operátor 2: + Priorita 2: L

V3
 B001 [Analogový zes...]
 Operátor 3: × Priorita 3: H

V4
 B002 [Analogový zes...]



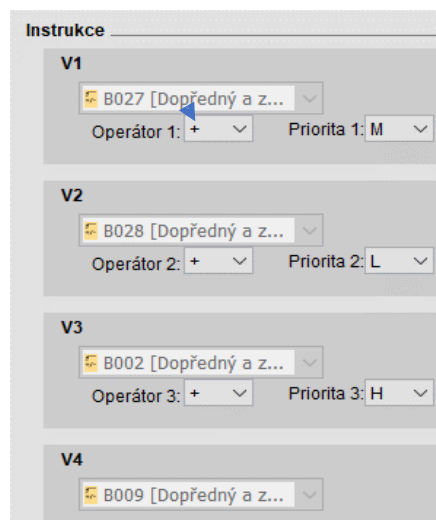
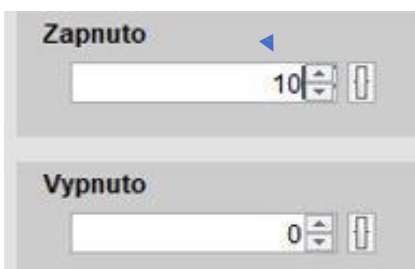
Aq:	32
P1:	20
P2:	10
P3:	3
P4:	10

V1=B32+
 V2=B26
 V3=B1
 V4=B2
 Point=0
 $(B32/B26)+(B1*B2)$

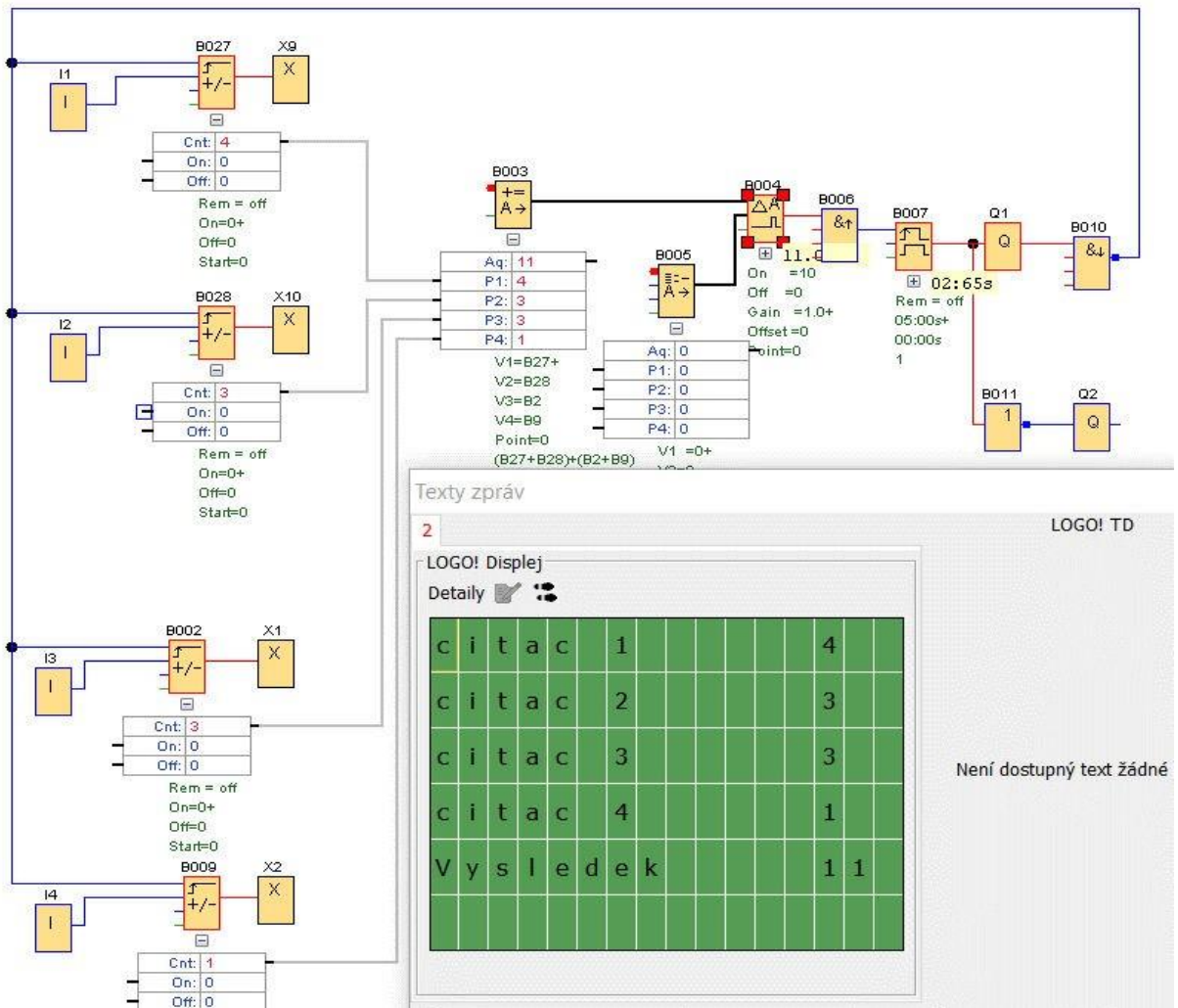
Následující příklad představuje situaci, kdy jsou vedle sebe čtyři stroje, které vyrábějí stejné výrobky, a ty padají do jedné bedny. Každý stroj má čidlo, které vydá impuls, když výrobek opustí stroj. Čidla jsou v programu napojena na vstupy I1, I2, I3, I4. Pokud součet součástek ze všech strojů dosáhne požadovaný počet, stroj se zastaví "Q2", plná bedna se odsune a přisune se nová. Čítače pro jednotlivé stroje se vynulují a proces se opakuje.

Vysvětlení funkce bloků:

Na vstupy I1, I2, I3, I4 navazují čítače. Ty jsou pomocí parametrů propojeny s parametry bloku "Matematické instrukce" "B003". Všechny vstupy se sčítají. Protože blok mat. instrukcí má jen analogový výstup a pro ovládání výstupu "Q" potřebujeme digitální, vložíme za něj "Analogový komparátor" "B004" s digitálním výstupem. Na něm nastavíme, při jakém součtu má výstup Zapnout a Vypnout.



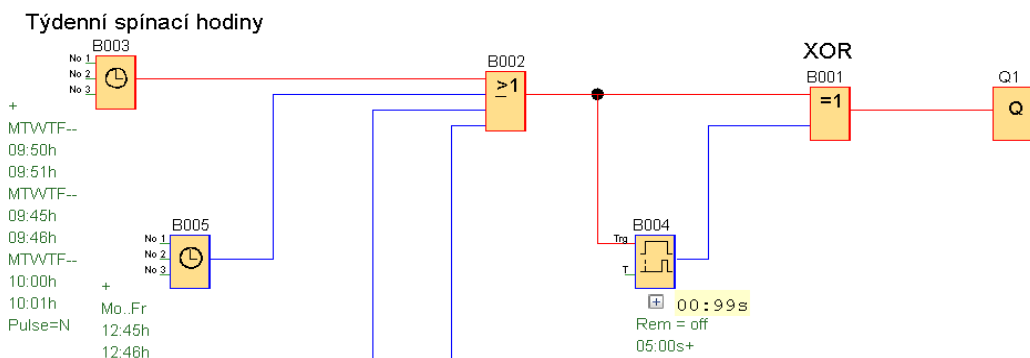
Po sepnutí výstupu komparátoru dostane hranou spouštěné relé "B007" impuls z bloku AND (hrana) "B006" a sepne se jeho výstup. Současně se začíná odpočítávat čas k jeho vypnutí. Po sepnutí výstupu "B007" se vypne výstup "Q2" (stroje) a sepne se "Q1"(posun bedny). Po vypnutí výstupu "Q1" se impulsem z NAND (hrana) "B010" resetují čítače a tím výstup Analogového komparátoru.



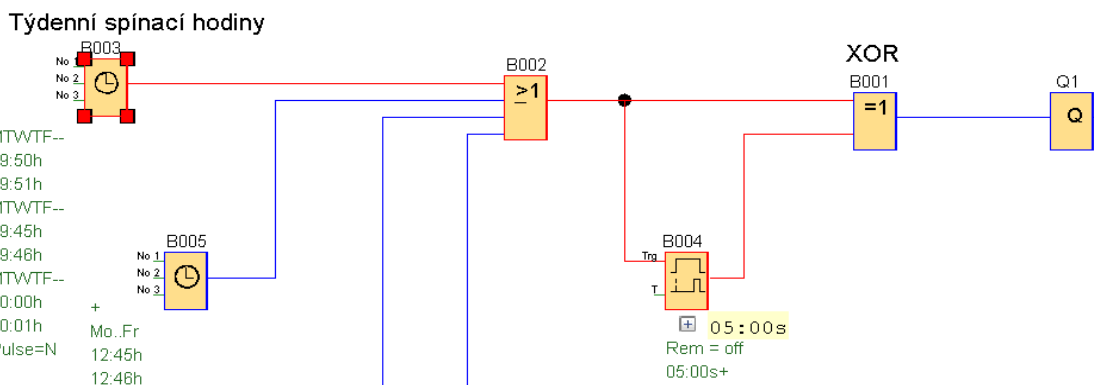
PŘÍKLADY

Před výstupem "Q1" je vložený logický blok XOR "B001". Jeho výstup je sepnutý, nabývají-li vstupy rozdílnou hodnotu (1–0 nebo 0–1), jsou oba vstupy sepnuté nebo vypnuté je výstup vypnutý. Spínací hodiny mají nejmenší interval 1 minutu, aby zvonek zvonil kratší dobu, je proto do programu vložen blok zpožděné zapnutí "B004" a je nastaven na 5s.

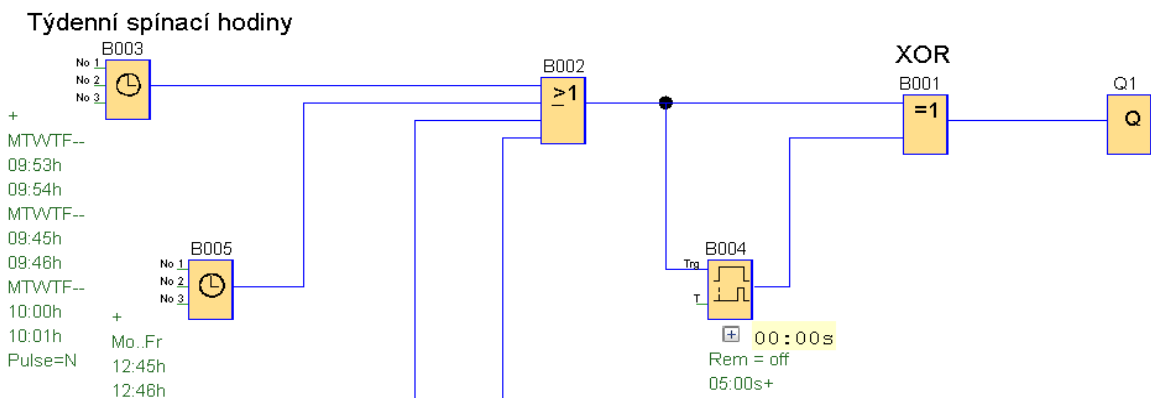
Na tomto obrázku je vidět sepnuté spínací hodiny, časovač ještě nedosáhl 5 s, a tak je jeho výstup vypnutý. Protože jsou vstupy do bloku XOR rozdílné je sepnutý výstup.



Zde dosáhlo časování 5s a výstup bloku "B004" se sepnul. Protože jsou oba vstupy do "B001" sepnuté, vypne se výstup.



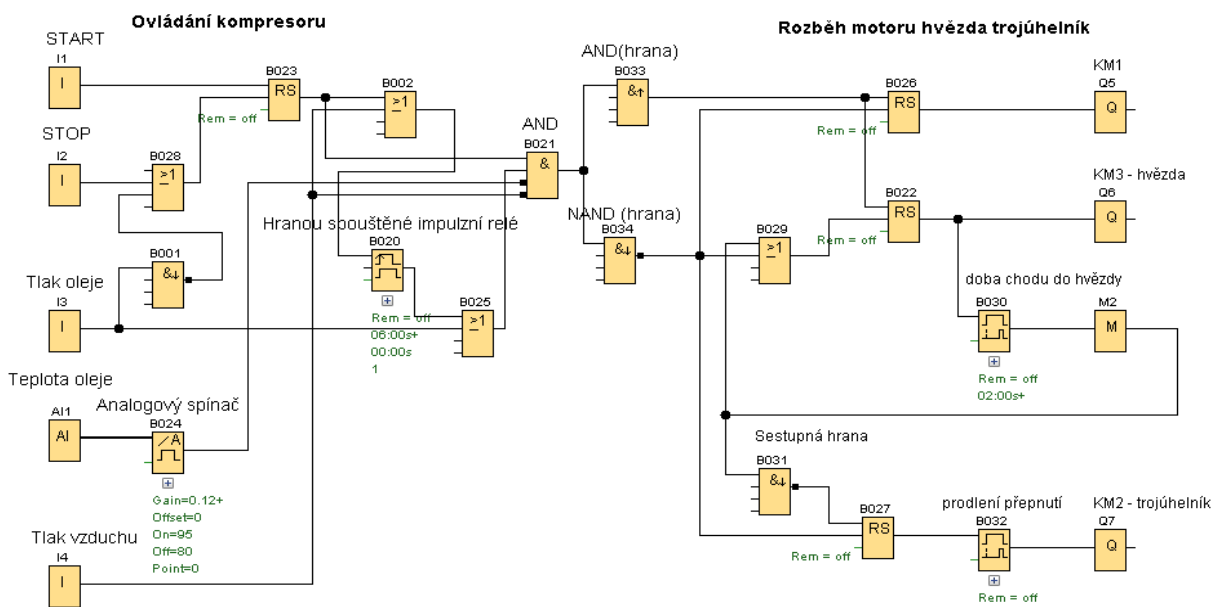
Zde je situace, kdy vypnuly spínací hodiny. Oba vstupy do XOR "B001" jsou vypnuté, a proto je vypnutý i jejich výstup.



KOMPRESOR

Kompresor je poháněn elektromotorem, který se rozbíhá nejprve v zapojení do hvězdy a po rozběhu se přepne trvale do trojúhelníku. Kompresor se spouští tlačítkem START. Motor kompresoru se zastaví v případě natlakování, což zajistí tlakový spínač "I4", dále při poklesu tlaku mazacího oleje "I3" a při překročení teploty mazacího oleje "AI1". Kompresor je možné kdykoliv zastavit tlačítkem STOP "I2". Při poklesu tlaku vzduchu se motor spustí.

- I1 START – spínací tlačítko
- I2 STOP – rozpínací tlačítko



- I3 tlak oleje- spínací kontakty
- AI1+B024 teplota oleje
- I4 Tlak vzduchu – rozpínací kontakty

Trvalé zapnutí motoru "Q5, Q7, je závislé na výstupu bloku AND "B021"

RS (B023)	I2 – STOP	I3-olej	AI1+B024	I4 –vzduch	AND (B021) - výstup
1	0	1	0	0	1

Tlak motoru je ve vypnutém stavu nulový. Tím je hodnota "I3" nula a motor by se nespustil, proto je na dobu delší, než je rozběh motoru zajištěno, aby na vstupu 2 And "B021", kde vstupuje "I3" byla hodnota 1, a to pomocí hranou spouštěného impulzního relé "B020". Za předpokladu, že není vzdušník kompresoru natlakován se spustí motor. Spuštěním motoru se zvýší tlak oleje v mazacím systému a sepne tlakový spínač "I3", pokud by se tlak oleje nebyl dostatečný, motoru se po ukončení časování "B020" vypne. Motor se automaticky vypne po dosažení nastaveného tlaku "I4" ve vzdušníku kompresoru. Po poklesu tlaku vzduchu, který je dán nastavením mechanického tlakového spínače se motor spustí.

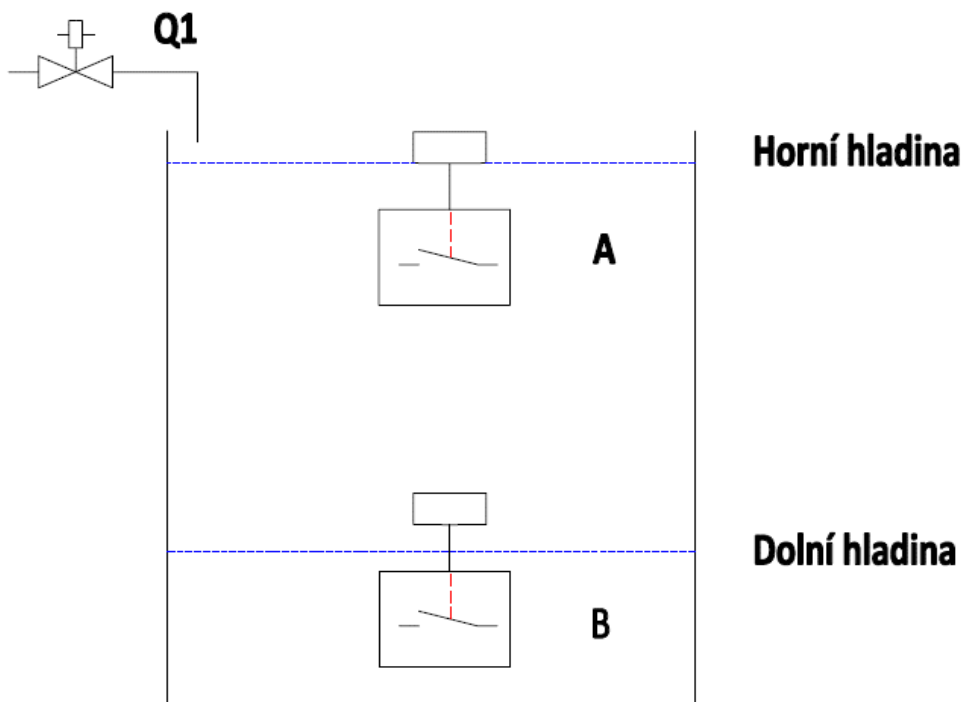
Spuštění motoru s přepínáním hvězda trojúhelník bylo popsáno v části tohoto návodu Motory. Tlačítko Start v tomto případě nahrazuje blok AND(hrana) "B033". Kdy po změně stavu vstupu z vypnutí na zapnutí dá impuls bloku RS "B026", a ten začne spouštět motor. Vypnutí motoru je provedeno náhradou tlačítka Stop blokem NAND (hrana) "B034", kdy při změně stavu ze zapnutí na vypnutí se resetují všechny bloky RS.

SPÍNÁNÍ HLADINY

Napouštění nádrže

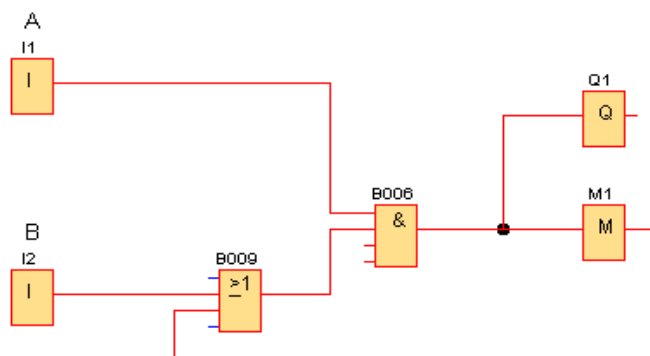
Voda natéká do nádrže při otevřeném elektromagnetickém ventilu. Při dosažení horní hladiny se ventil uzavře. Při klesání hladiny zůstává ventil uzavřený až do dosažení dolní hladiny. Během klesání hladiny z horní hladiny do dolní zůstává ventil Q1 zavřený. Až hladina vody klesne na dolní hladinu, ventil se opět otevře a voda začne natékat do nádrže.

Plovákový spínač sepne při poklesu hladiny .

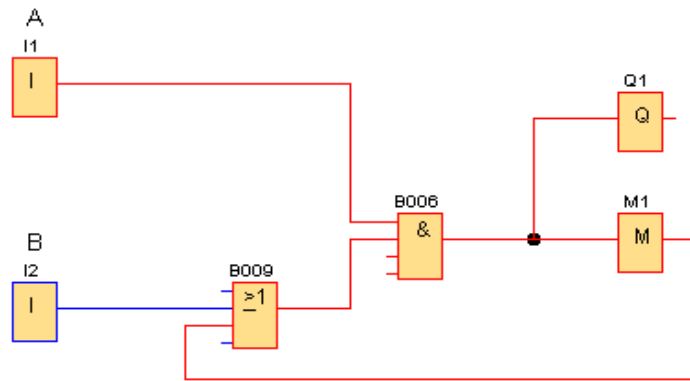


Řešení pomocí základních funkcí – logických bloků

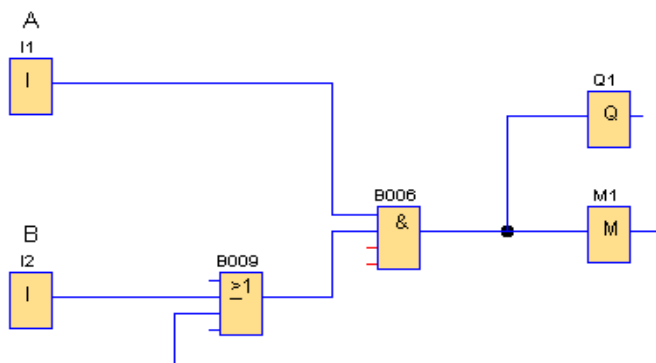
1. Stav, kdy je nádrž prázdná. Oba kontakty plovákových spínačů jsou sepnuty . Je sepnutý ventil Q1 a voda natéká.



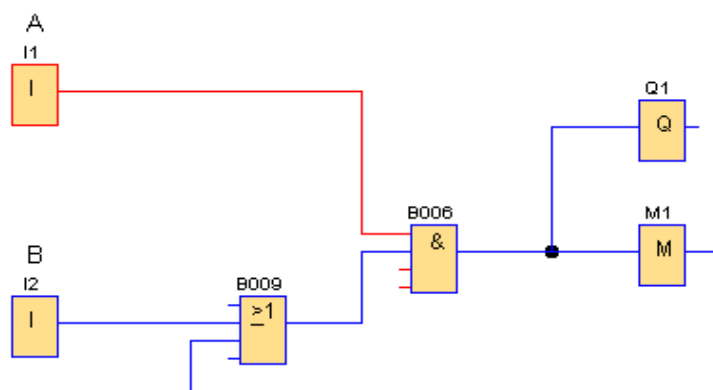
2. Hladina stoupla nad dolní hladinu. Vypne se kontakt plovákového spínače B. Ventil Q1 zůstává sepnutý, voda natéká.



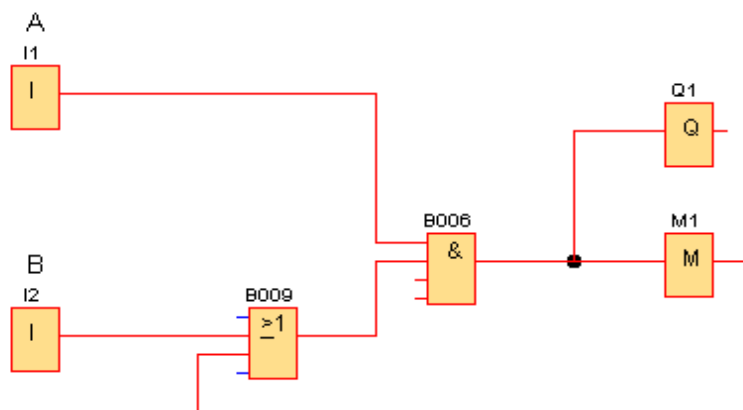
3. Voda dosáhla horní hladiny. Rozpojí se kontakty plovákového spínače A, ventil Q1 se vypne.



4. Hladina klesne pod horní hladinu. Sepne se kontakt plovákového ventilu A. Ventil Q1 zůstává vycpaný až do poklesu na dolní hladinu.



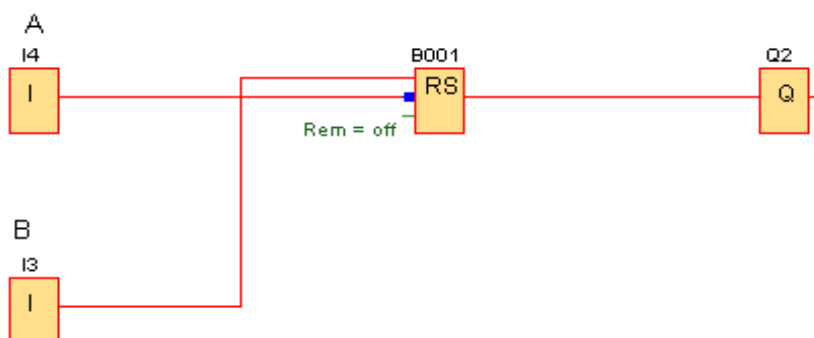
5. Voda je na úrovni dolní hladiny. Sepne se kontakt B a tím i ventil Q1. Voda natéká a postup se opakuje.



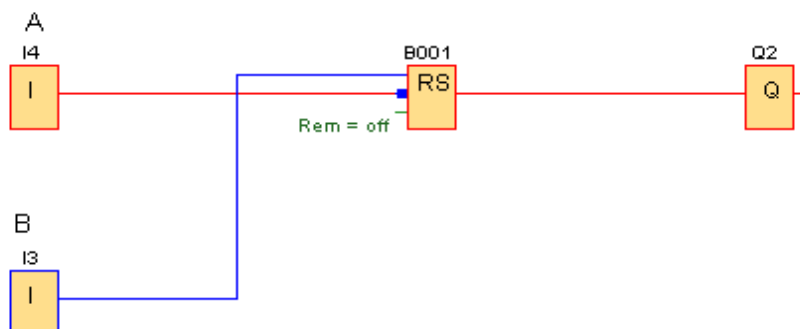
Řešení pomocí bloku RS

Vstup R bloku RS "B001" je negovaný

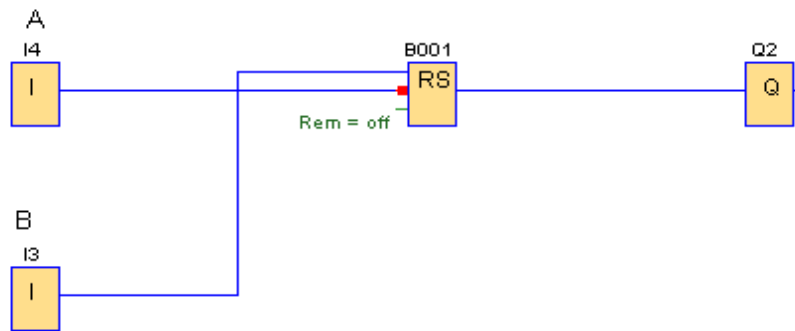
1. Stav, kdy je nádrž prázdná. Oba kontakty plovákových spínačů jsou sepnuty. Je sepnutý ventil Q1 a voda natéká.



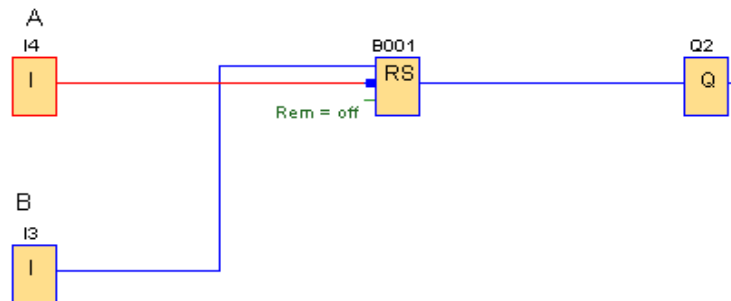
2. Hladina stoupla nad dolní hladinu. Vypne se kontakt plovákového spínače B. Ventil Q1 zůstává sepnutý, voda natéká.



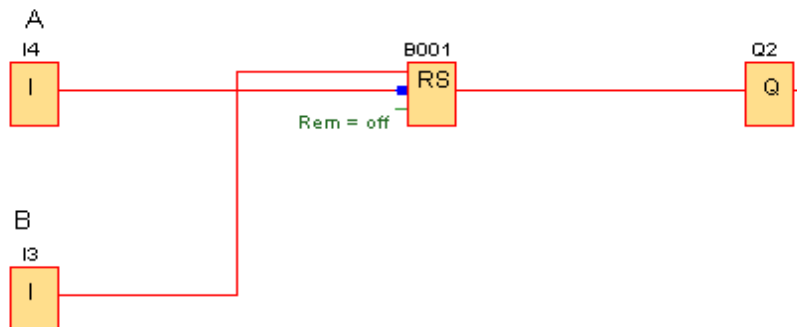
3. Voda dosáhla horní hladiny. Rozpojí se kontakty plovákového spínače A, ventil Q1 se vypne.



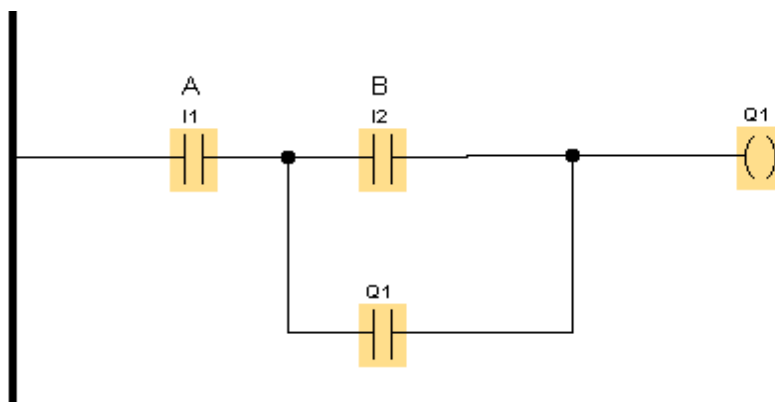
4. Hladina klesne pod horní hladinu. Sepne se kontakt plovákového ventilu A. Ventil Q1 zůstává vypnutý až do poklesu na dolní hladinu.



5. Voda je na úrovni dolní hladiny. Sepne se kontakt B a tím i ventil Q1. Voda natéká a postup se opakuje.

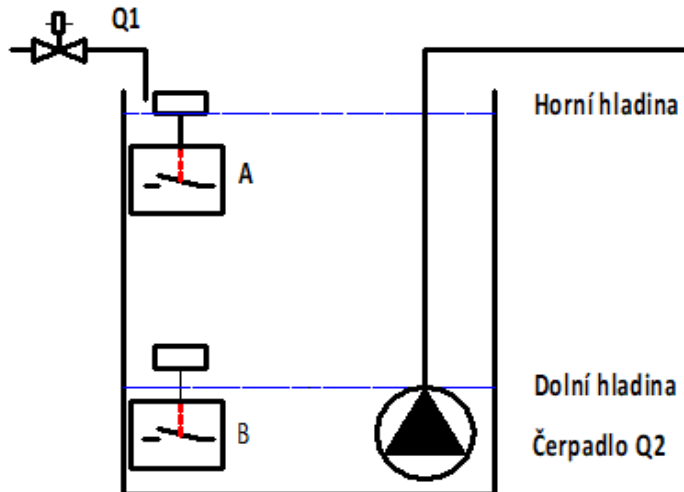


Řešení pomocí bloků LAD

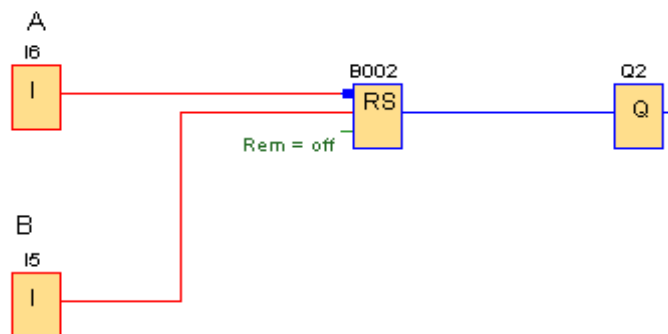


Vyprazdňování nádrže

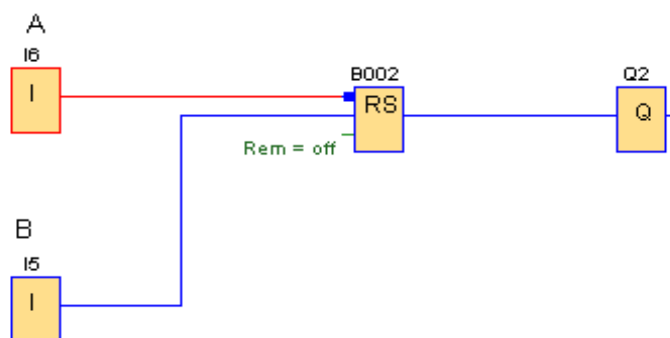
Až se nádrž naplní na horní hladinu, sepne se čerpadlo Q2. Po odčerpání vody na dolní hladinu se čerpadlo vypne a zůstane vypnuté až do zaplnění nádrže do horní hladiny.



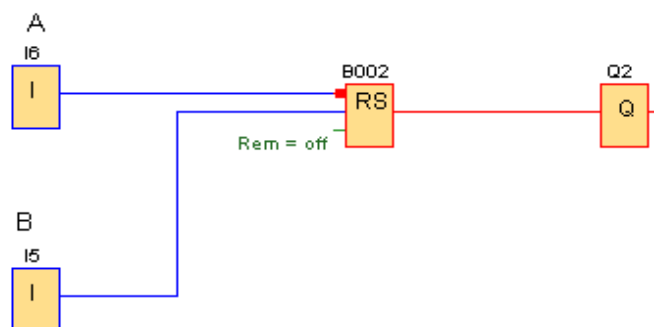
1. Nádrž je prázdná, oba plovákové spínače mají sepnuté kontakty. Čerpadlo je vypnuté.



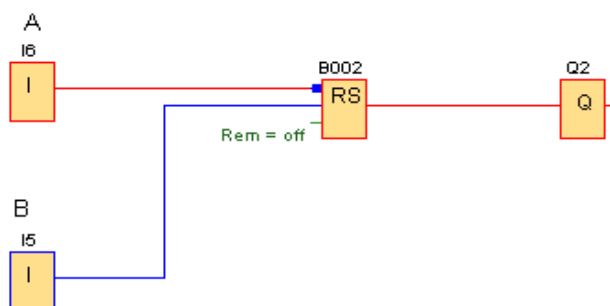
2. Nádrž se zaplní nad dolní hladinu. Plovákový spínač B sepne. Čerpadlo Q2 zůstává vypnuté.



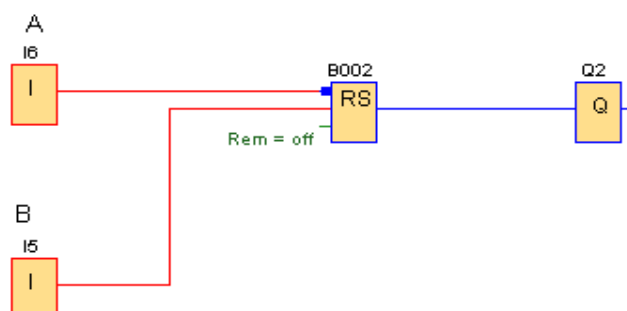
3. Kapalina v nádrži dosáhne horní hladiny. Plovákový spínač A vypne. Protože je vstup S bloku RS negován, sepne se výstup bloku RS "B002" a tím i čerpadlo Q2.



4. Zapnutím čerpadla se začne snižovat hladina, sepne se kontakt plovákového spínače A. Čerpadlo zůstane sepnuté.



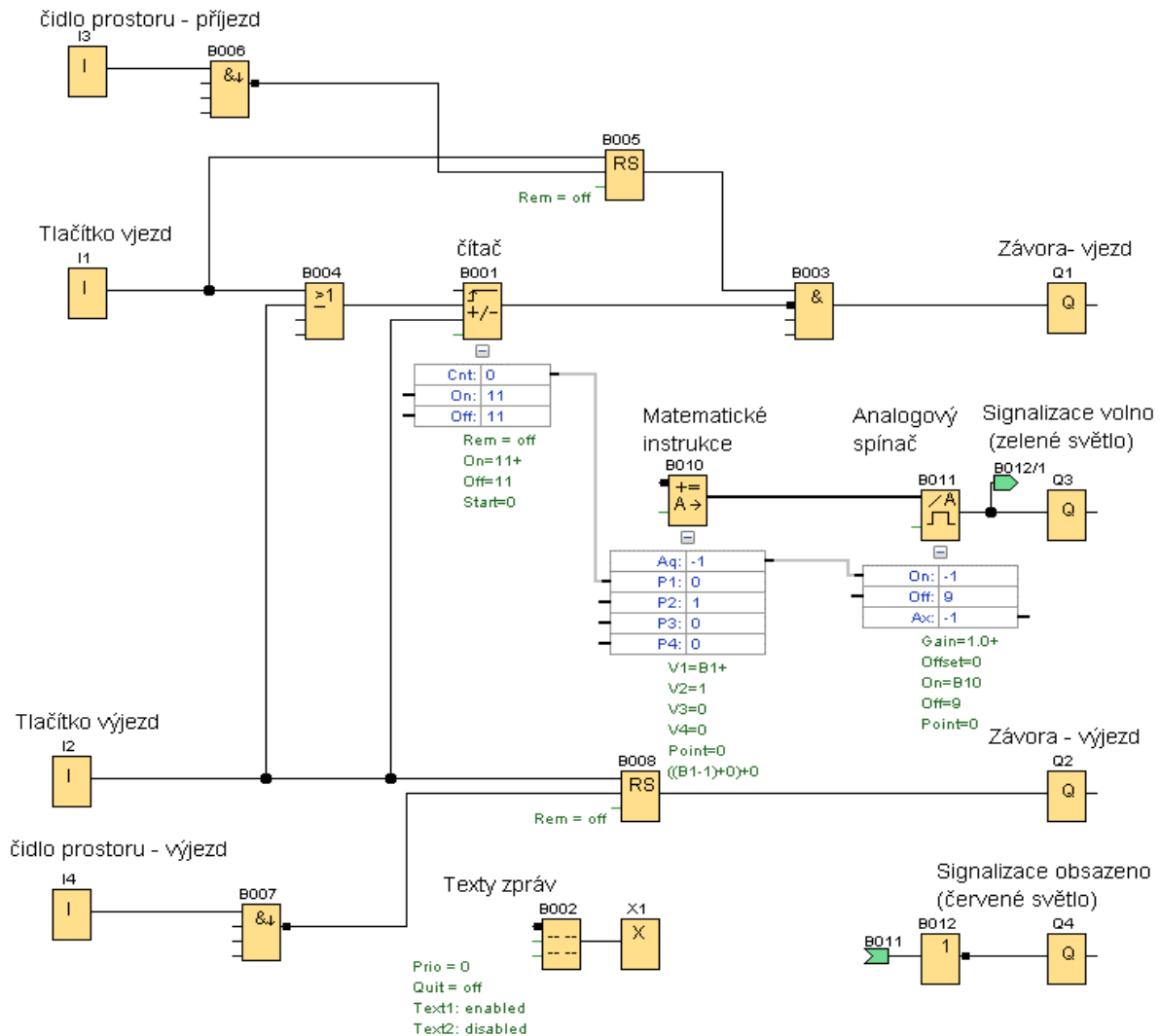
5. Při poklesu kapaliny na dolní hladinu se sepne kontakt plovákového spínače B, tím se vypne výstup bloku RS "B002" a tím i čerpadlo Q2.



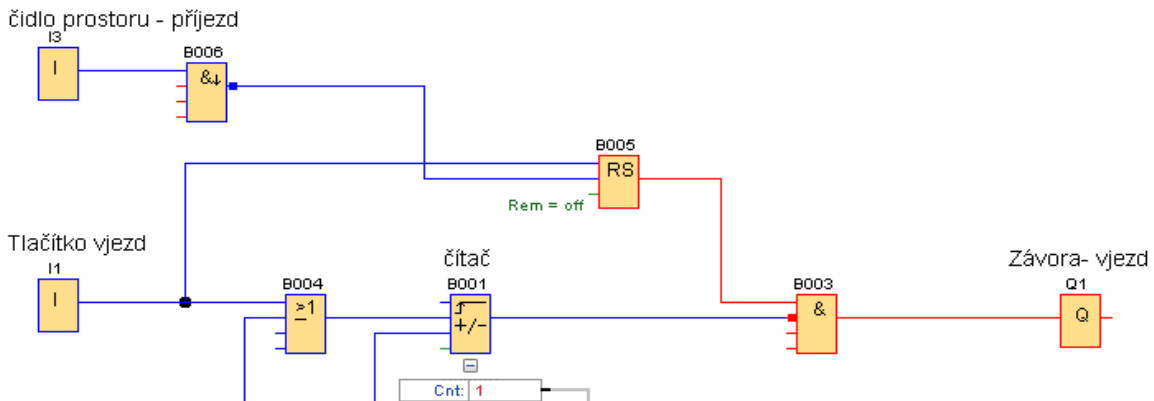
PARKOVIŠTĚ

Parkoviště má dvě závory. Jednu pro příjezd vozidel a druhou pro jejich odjezd. Při příjezdu se stiskne tlačítko vyjede parkovací lístek a závara se otevře. Čidlo zaznamená v prostoru závory vozidlo. Po opuštění vozidla z prostoru závorou se závara automaticky zavře. Při výjezdu se vloží parkovací lístek do čtečky, závara otevře. Po opuštění prostoru pod závorou se závara automaticky zavře. Je-li na parkovišti volné místo svítí zelený semafor, při jeho obsazení se rozsvítí červená. Při obsazení všech parkovacích míst zůstane závara zavřená. Po uvolnění parkovacího je umožněn jezd dalšího vozidla.

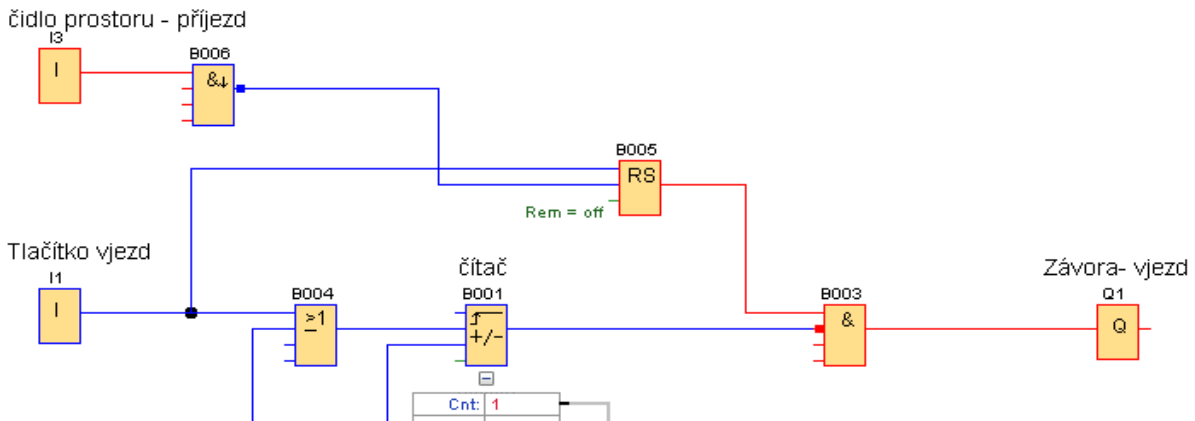
Program



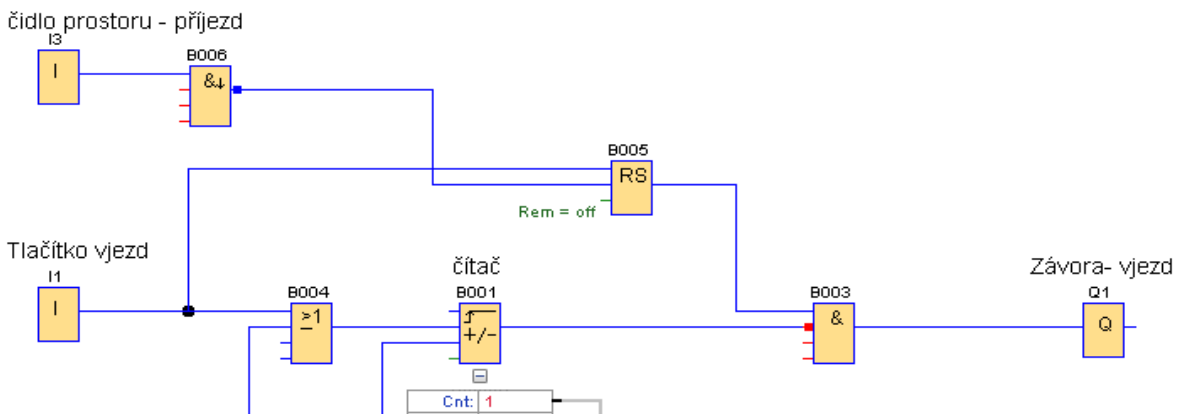
1. Při příjezdu k závoře se stiskne tlačítko "I1", tím se sepne výstup RS "B005" a současně se zvýší hodnota čítače "B001" o 1. Pokud nedosáhne čítač nastavenou hodnotu pro sepnutí, zůstává výstup čítače vypnutý, ale jeho vstup do bloku AND "B003" je negován. Na vstupu do bloku AND jsou oba vstupy 1, proto je jeho výstup sepnutý a závořa se otevře.



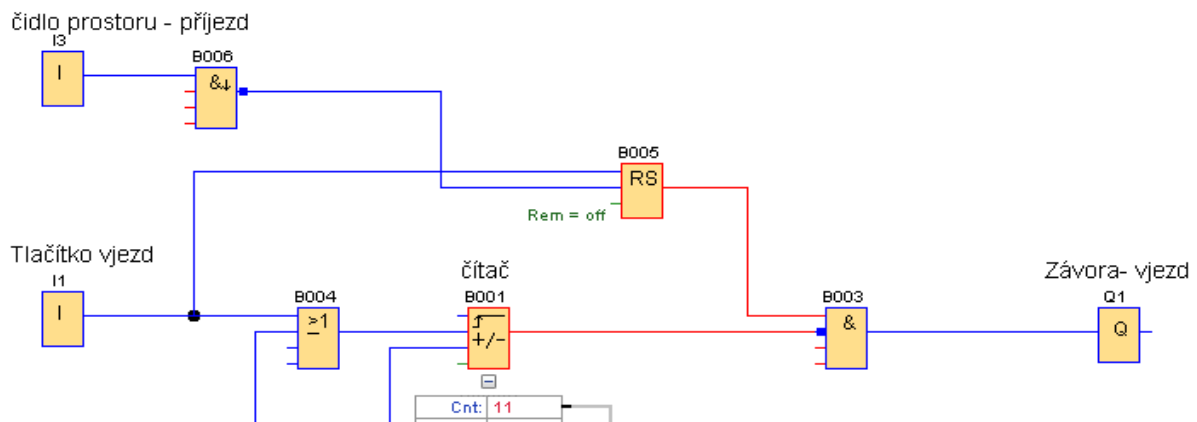
2. Po pojetí vozidla do prostoru závoře se sepne čidlo "I3". Závořa zůstane otevřena.



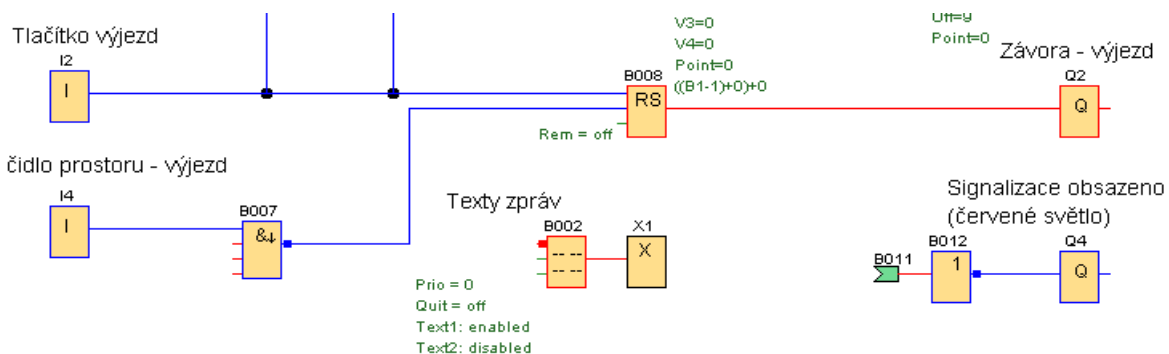
3. Po opuštění prostoru pod závořou se čidlo "I3" vypne. Vypnutím dá blok NAND (hrana) "B006" impuls na vstup R, bloku RS "B005" a tím dojde k vypnutí jeho výstupu. Tím jsou na vstupu "B003" hodnoty 0 a 1 a jeho výstup se vypne. Závořa "Q1" se zavře.



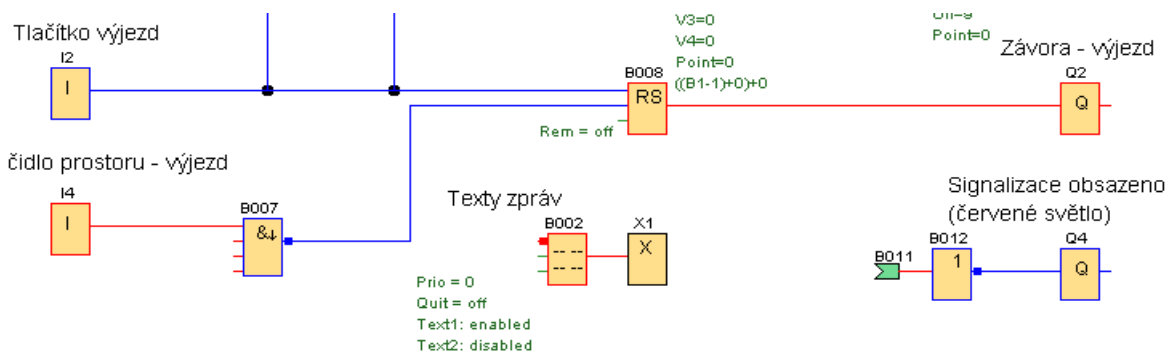
4. Po obsazení parkoviště, což je v tomto případě 10 a stisknutí tlačítka "I1" se načte hodnota 11, na ní je nastaveno sepnutí čítače, čítač "B001" sepne výstup a vstup do bloku AND "B003" má hodnoty 1 a 0 (negace výstupu čítače) a jeho výstup se vypne, tím zůstane spuštěná závara.



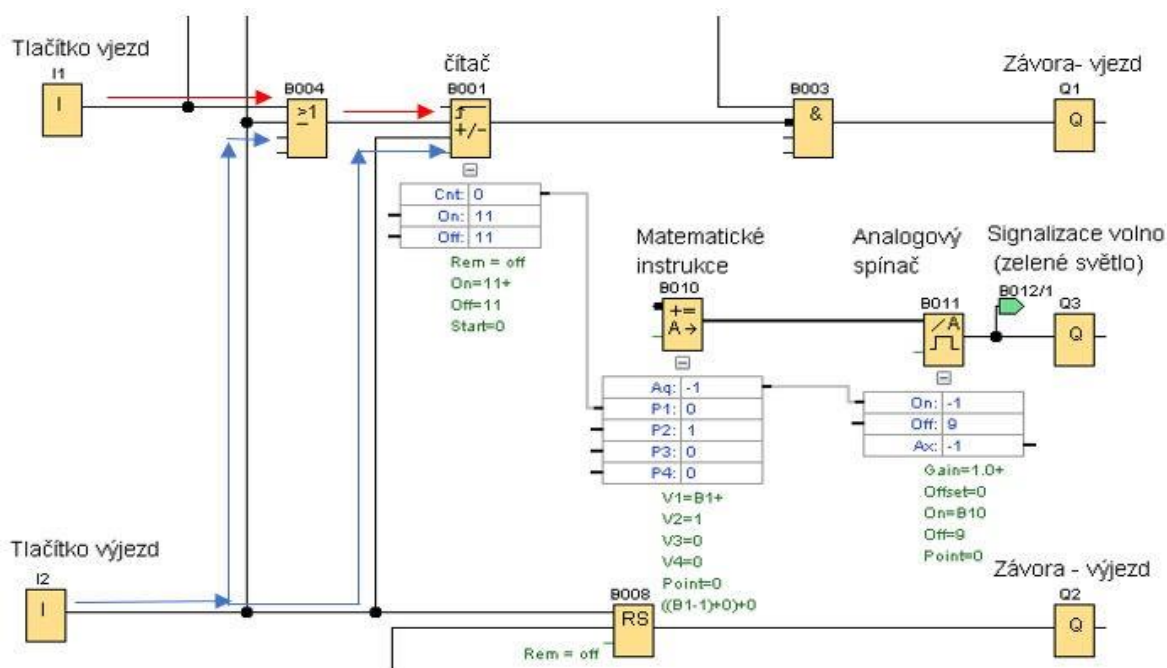
Odjezd vozidel je řešen podobně. Po příjezdu k závoře se přiloží nebo vloží parkovací lístek do čtečky, ta vyhodnotí zaplacení parkovného a pokud je vše v pořádku vydá impuls "I2", ten sepne výstup bloku RS "B008" a otevře se závara.



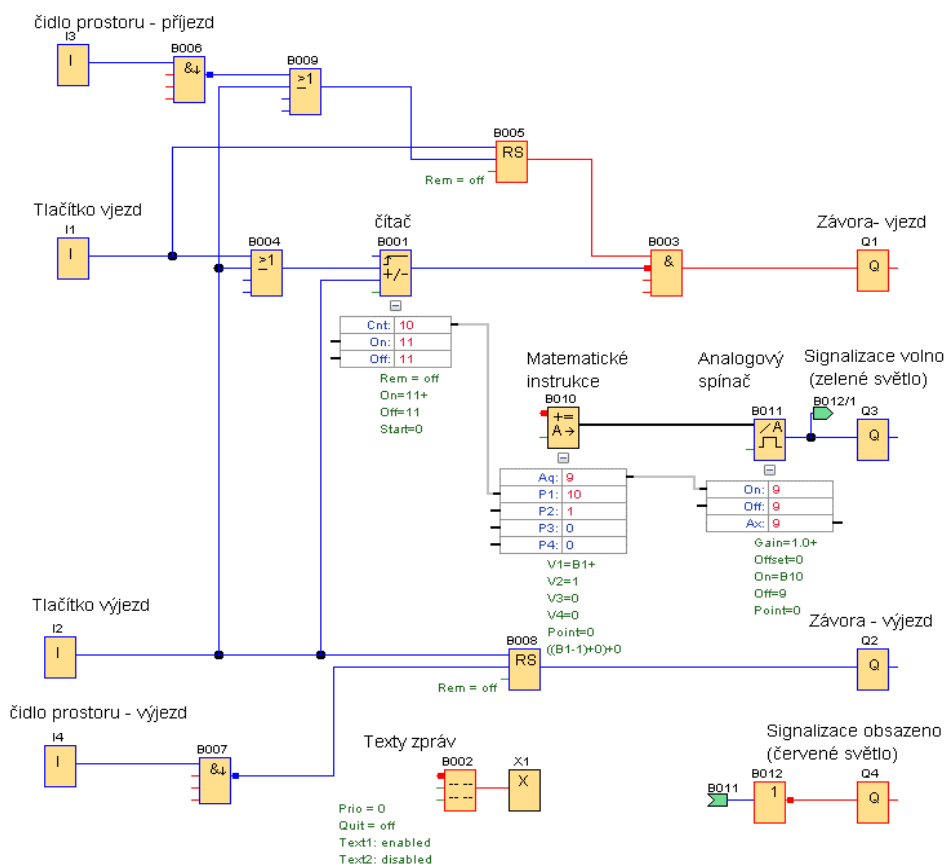
Dále se situace opakuje jako při příjezdu. Sepne se čidlo "I4", které sleduje, zda je v prostoru pod závorou vozidlo a po jeho opuštění se závara spustí. Impulzem vstupu "I2" se kromě zdvižení závoře ještě sníží hodnota čítače "B001".



Červeně je označena cesta při načítání, modře při odečítání. Kapacita parkoviště je 10 vozidel.



Pro signalizaci obsazeno a volno, je do programu vložen blok Matematické instrukce a analogový spínač. Je to z důvodu, že po příjezdu 10. vozidla se musí na semaforu musí objevit červená, ale 10. vozidlo musí být vpuštěno na parkoviště. Proto se od hodnoty čítače odečítá jednička. Na následujícím obrázku je vidět, že při 10. vozidle je závora otevřena, ale svítí červený semafor.



Nastavení bloků pro parkoviště s kapacitou 10 vozidel

B001 [Dopředný a zpětný čítač]

Parametr | Komentář

Obecné

Název bloku:

Počáteční hodnota:

Zapnuto

Vypnuto

B010 [Matematické instrukce]

Parametr | Komentář

Obecné

Název bloku:

Instrukce

V1

Operátor 1: Priorita 1:

V2

Operátor 2: Priorita 2:

V3

Operátor 3: Priorita 3:

V4

SEMAFOR

Příklad: Semafor 1

Na rovné vozovce je přechod pro chodce řízený semaforem. Po přivedení napětí do LOGA se automaticky rozsvítí červená pro řidiče, ta svítí 45 s a současně svítí zelená svítí pro chodce. Po 35s se pro chodce rozsvítí červená a současně svítí červená pro řidiče. Po 45s se sepne žlutá a ta svítí společně s červenou 2s. Po 2s se rozsvítí zelená pro řidiče a svítí 45s, pro chodce svítí červená. Po 45s se rozsvítí žlutá pro řidiče na 5s, pro chodce svítí stále červená. Po 5s zhasne žlutá a rozsvítí se červená pro řidiče, červená pro chodce stále svítí. Po 3s se pro chodce rozsvítí zelená, ta svítí 35 s a cyklus se opakuje.

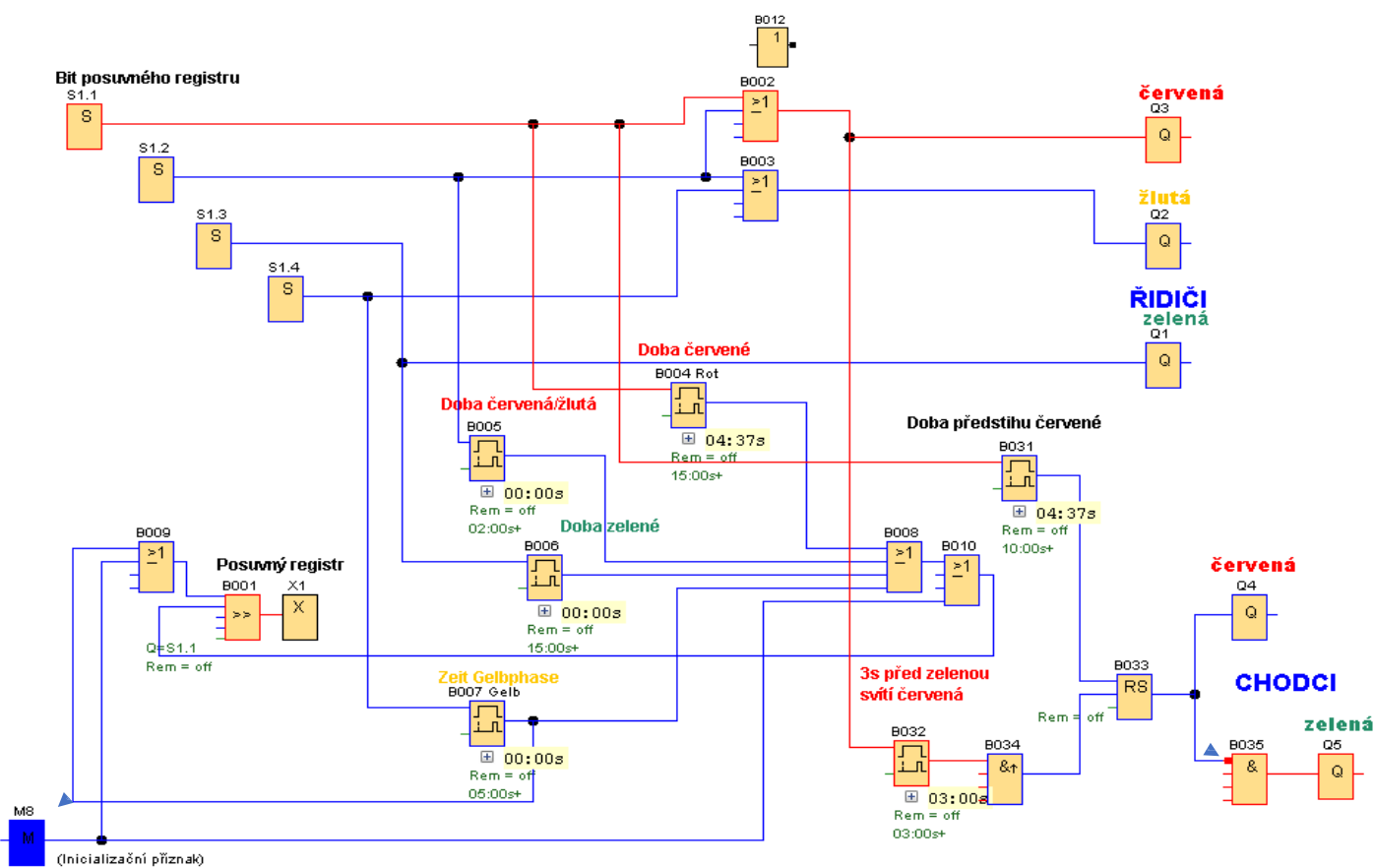
Text zadání příkladu převedený do tabulky :

postup	řidiči			chodci		doba trvání kroku (s)
	červená	žlutá	zelená	červená	zelená	
1.krok	1	0	0	0	1	35
2. krok	1	0	0	1	0	10
3.krok	1	1	0	1	0	2
4.krok	0	0	1	1	0	45
5.krok	0	1	0	1	0	5
6.krok	1	0	0	1	0	3
7.krok	1	0	0	0	1	35

Zapojení bloků

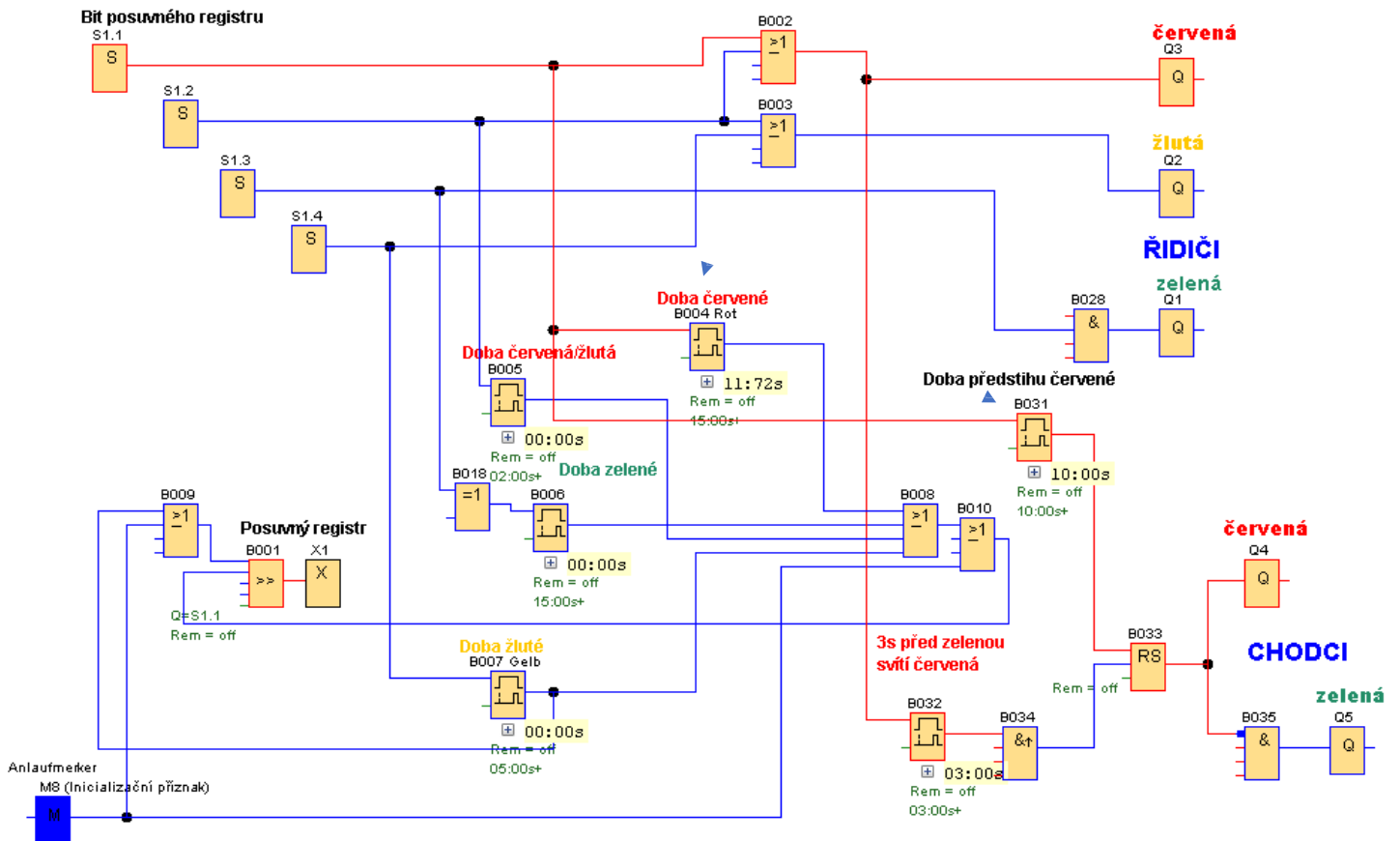
1. krok

Čas je u bloků zpožděného zapnutí zkrácen z důvodu simulace. Na obrázku je stav při prvním kroku. Střídání barev na semaforu je provedeno bitovým registrem, který byl popsán v předchozí části. Program se aktivuje připojením na napájení, a to "Příznakem M8". Po spuštění je sepnutý bit pos.registru S1.1. Ten sepne výstup červené pro řidiče. Doba trvání červené "B004" je nastavena na 15s (v zadání 45s). Současně s časovačem "zpožděné zapnutí" je aktivován i časovač "B031" (10s). Dokud nedosáhne čas u "B031" hodnotu 10s svítí pro chodce zelená. Výstup bloku "RS" "B033" není sepnutý a zelená svítí, protože je vstup do "B035" negován. Místo bloku AND může být použitý blok negace NOT.



2. krok

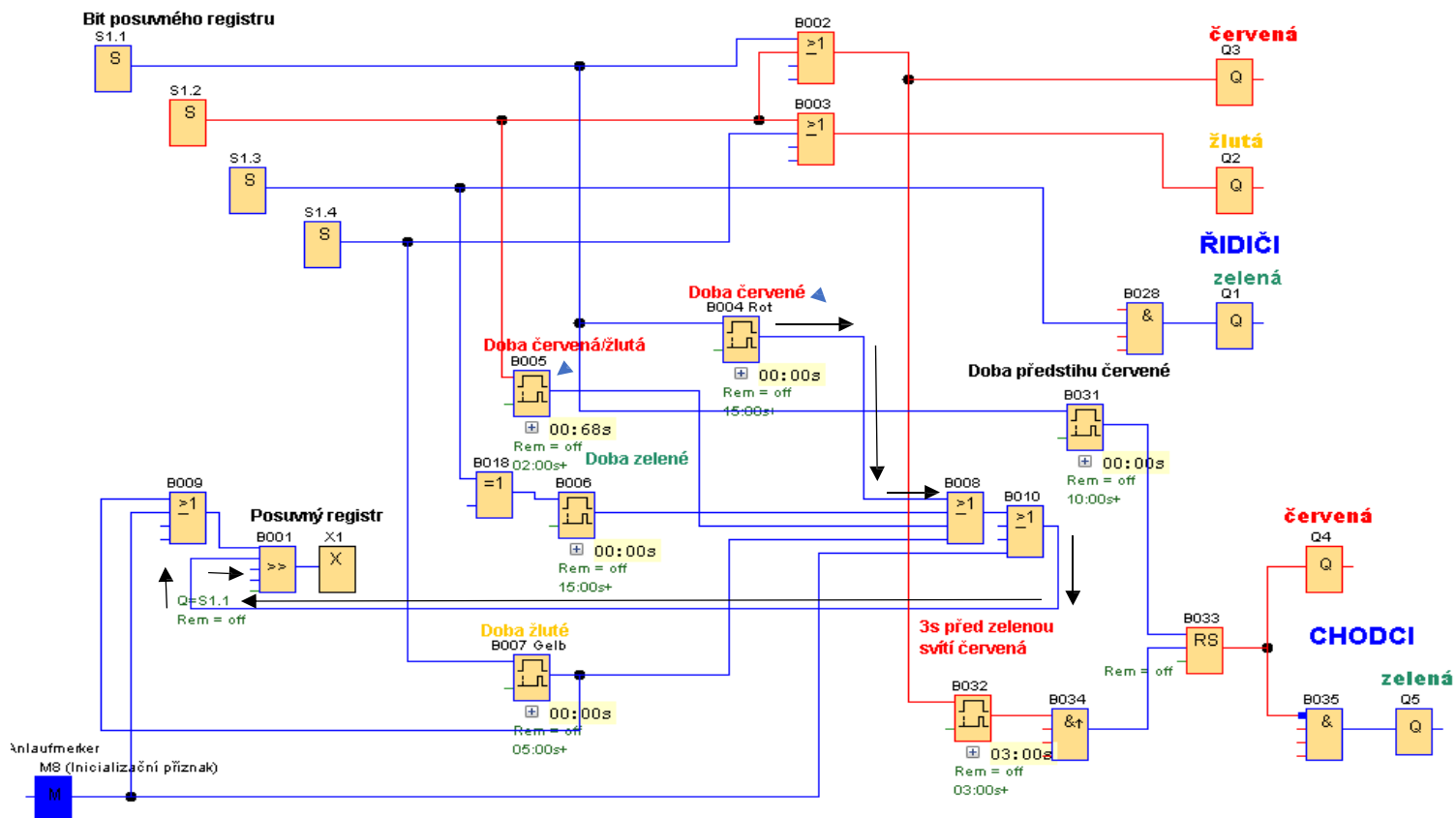
Časovač "B031" dosáhl nastavenou hodnotu 10s a sepnul výstup. Tím se sepnul výstup bloku RS "B033", tím se sepnula červená pro chodce a vypnula zelená pro chodce (negace vstupu "B035"). Současně ještě nedokončil časování blok "B004", a tak pro chodce i řidiče svítí červená po dobu (15-10 = 5s).



3. Krok

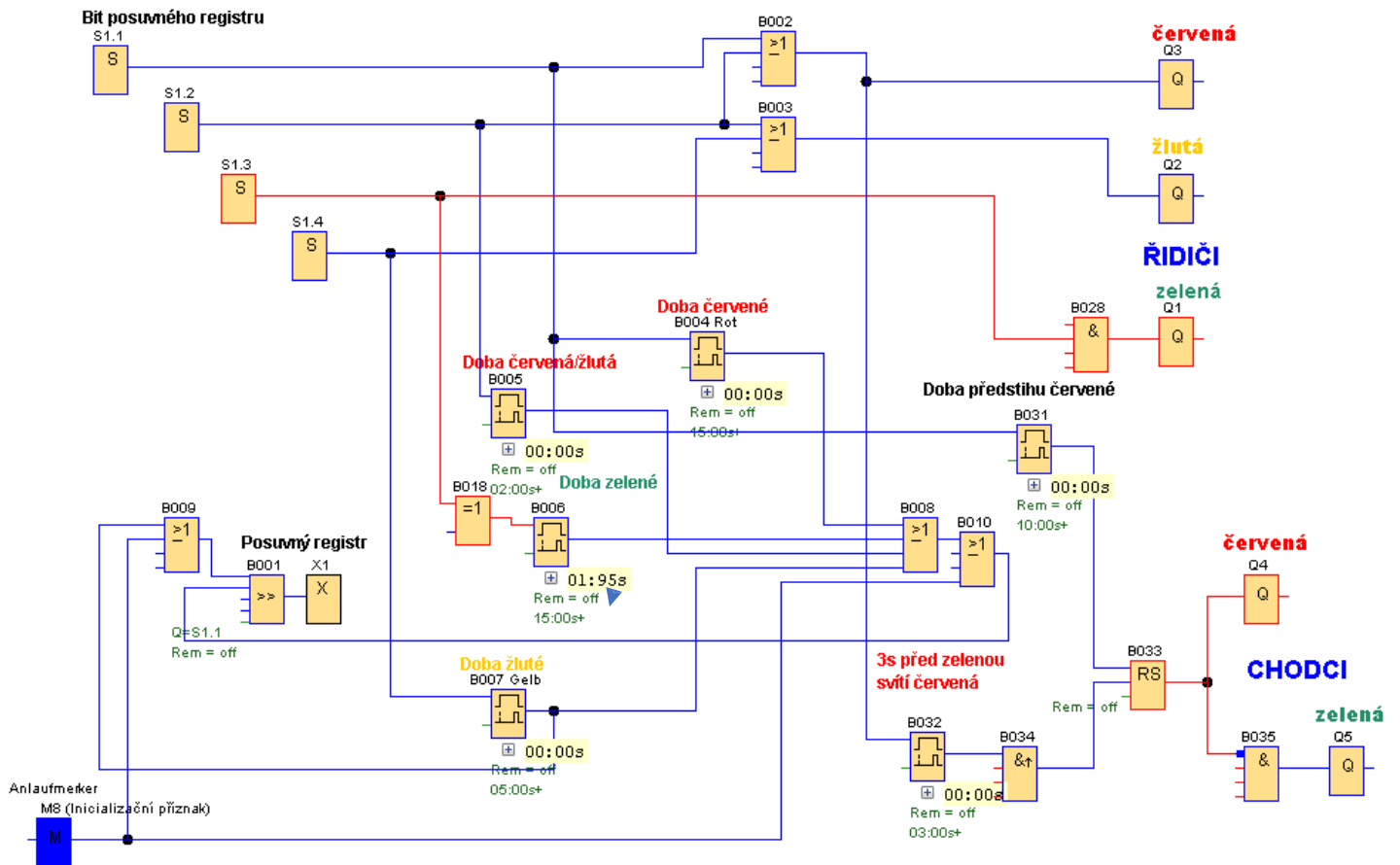
Po ukončení časování "B004" (15s) došlo k posunu bitu posuvného registru "S1.2" a svítí pro řidiče současně červená a žlutá. Protože se vypnul bit "S1.1" vypnulo se i napájení časovače červeného světla "B004" a začal časovat blok "B005" (doba červená/žlutá).

Cesta k posuvnému registru po sepnutí časovače "B004"



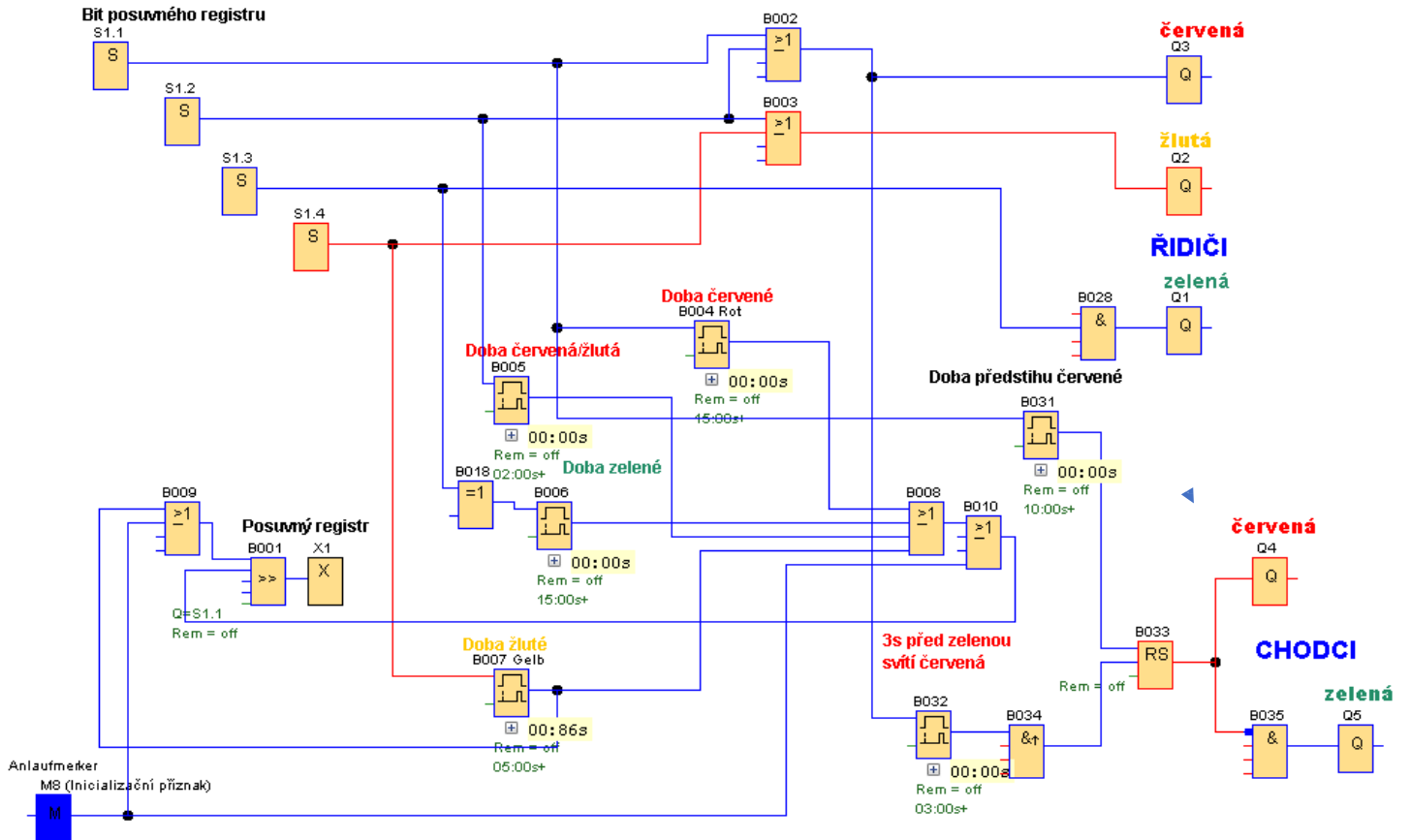
4.Krok

Po ukončení časování bloku "B005" se sepne jeho výstup a tím dojde k podnětu posuvného registru "B001" a přepne se bit posuvného registru na S1.3. a rozsvítí se zelená pro řidiče. Ta zůstane svítit dobu nastavenou časovačem "B006" (Doba zelené).



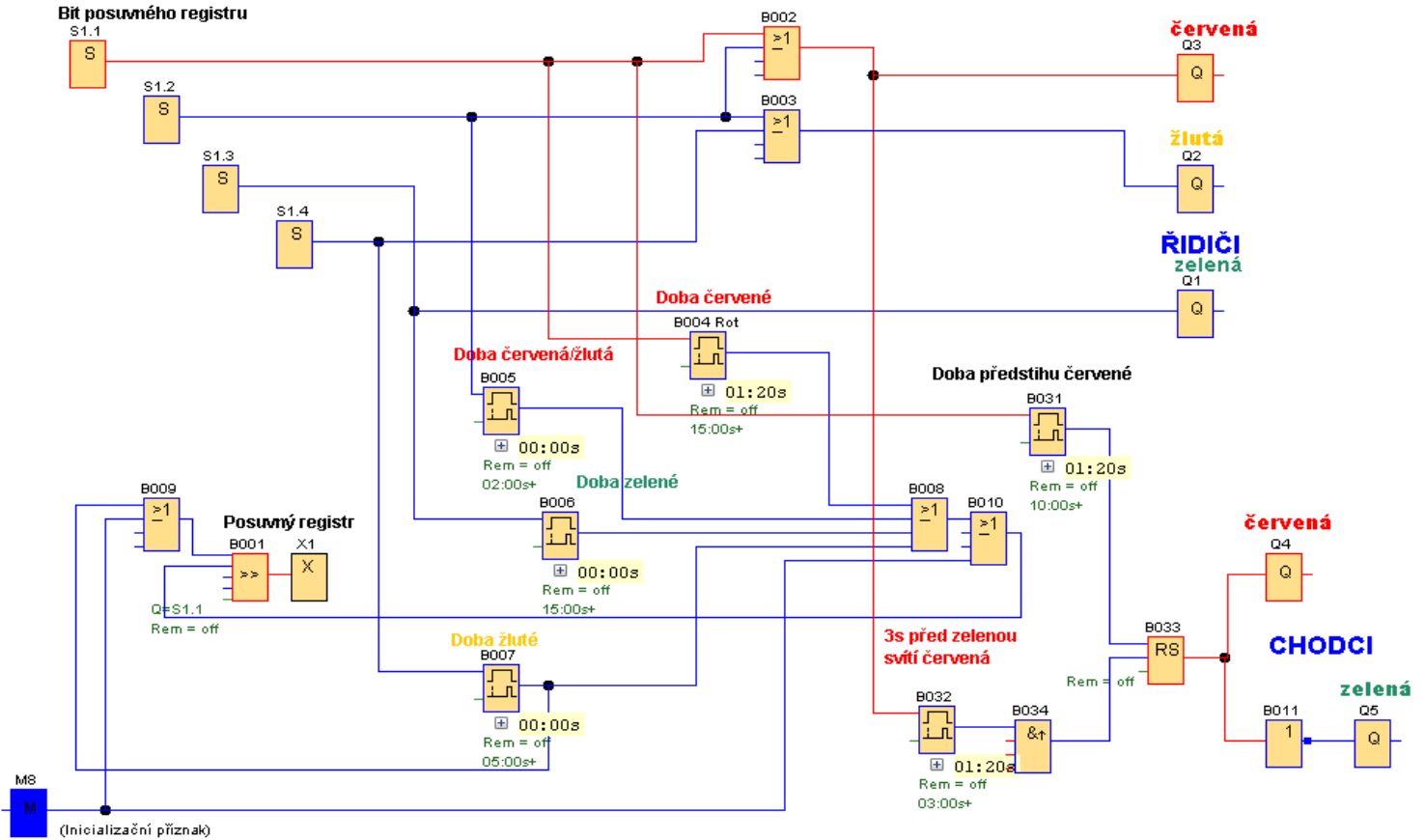
5. Krok

Po ukončení časování bloku "B006" dojde k posunu registru "B001" na bit "S1.4", zhasne zelená a rozsvítí se žlutá.



6. Krok

Po ukončení časování "B007" (žluté) dojde posunu registru na první bit "S1.1". Rozsvítí se červená pro řidiče. Stále ještě svítí červená pro chodce. S aktivací bitu "S1.1" začne časovat blok "B032". Po ukončení časování (3s) dojde přes blok "B034" AND (hrana) k vypnutí výstupu z bloku RS "B033". Tím zhasne červená pro chodce a rozsvítí se zelená. Červená pro řidiče zůstává svítit. Tím bylo docíleno počátečního stavu a cyklus se opakuje.

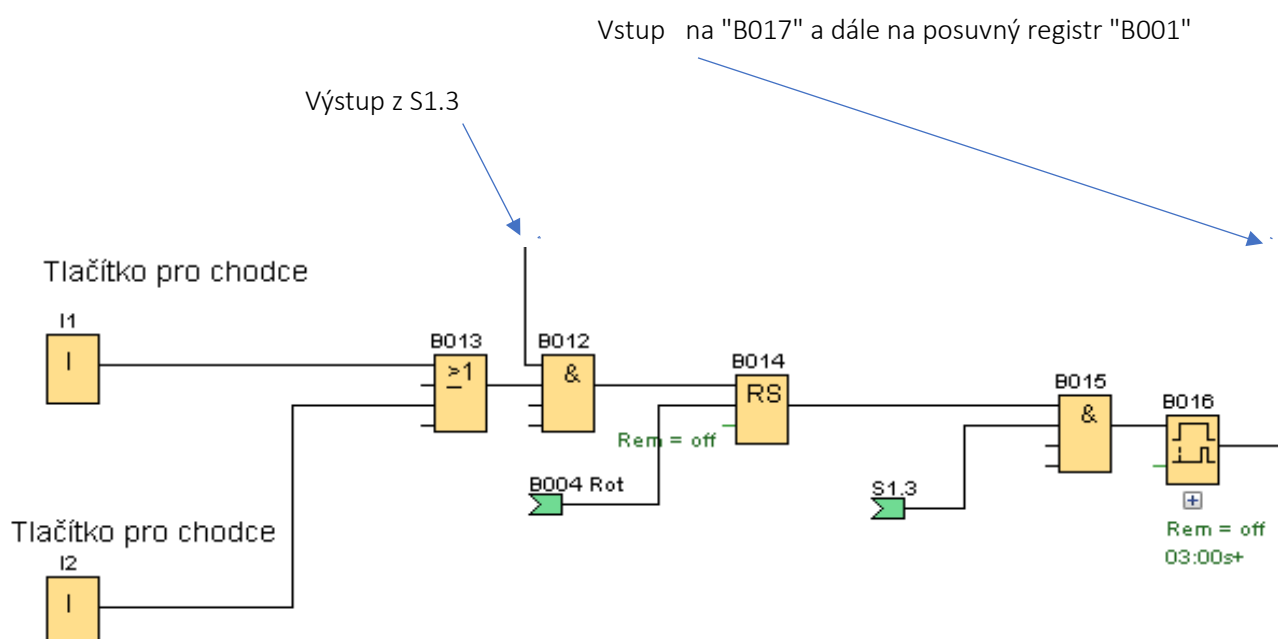


Příklad: Semafor2

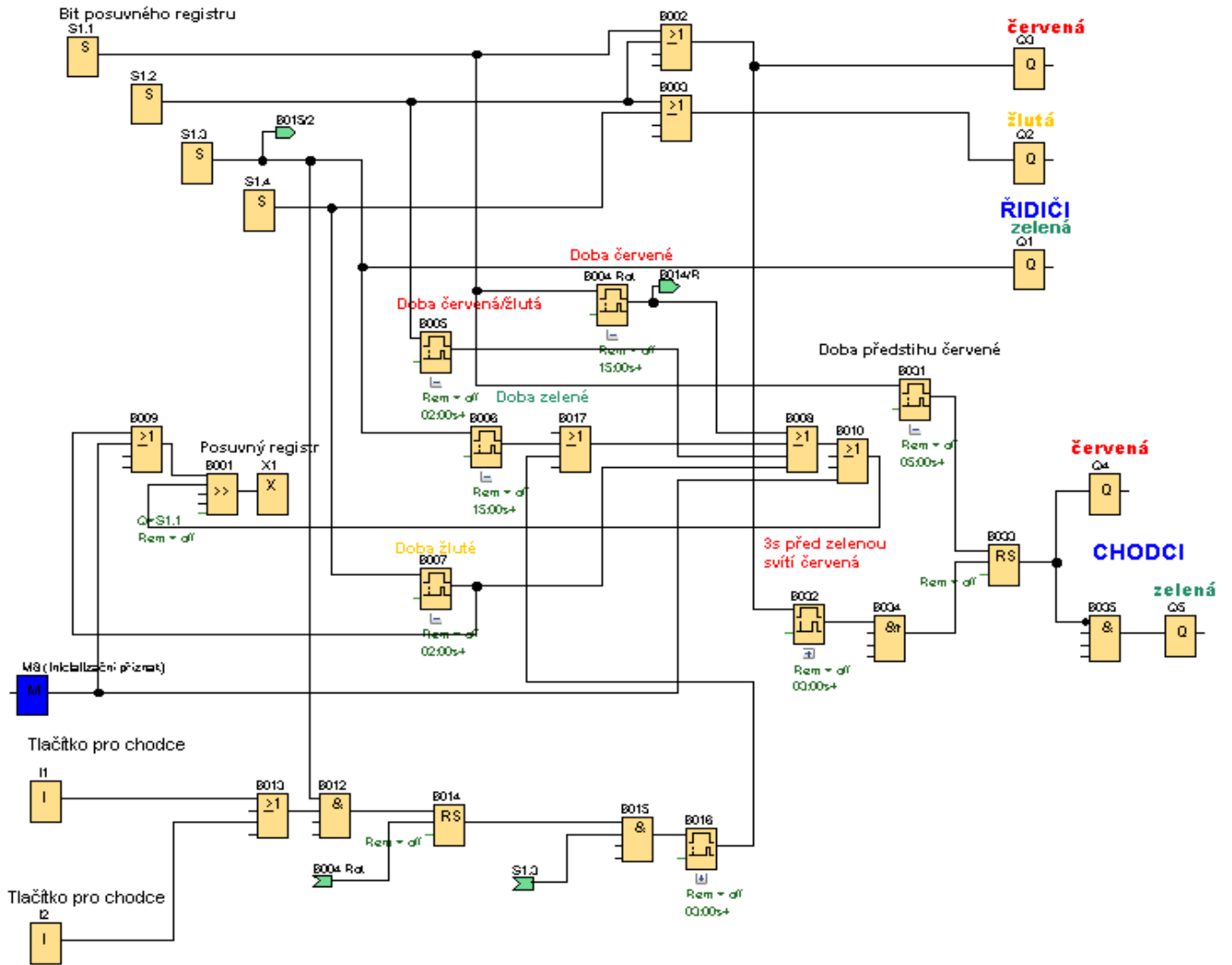
Příklad se semaforem pro chodce rozšíříme o tlačítka pro chodce. Tlačítka jsou na každé straně přechodu pro chodce. Svítí-li zelená pro řidiče, stiskem tlačítka se doba zelené pro řidiče zkrátí. Po sepnutí "I1" nebo "I2" se sepne výstup bloku "B016" (3s) a sepne se žluté světlo "B007" (2s), po něm červené pro řidiče a po 3s "B032" se sepne pro chodce zelená. Od stisku tlačítka pro chodce po rozsvícení zelené uběhne 8s.

Předchozí příklad rozšíříme o následující bloky. Zrychlení přechodu ze zelené pro řidiče na červenou se dosáhne posunem z registru "S1.3" na registr "S1.4". Tím se dostaneme do 5. kroku předchozího příkladu.

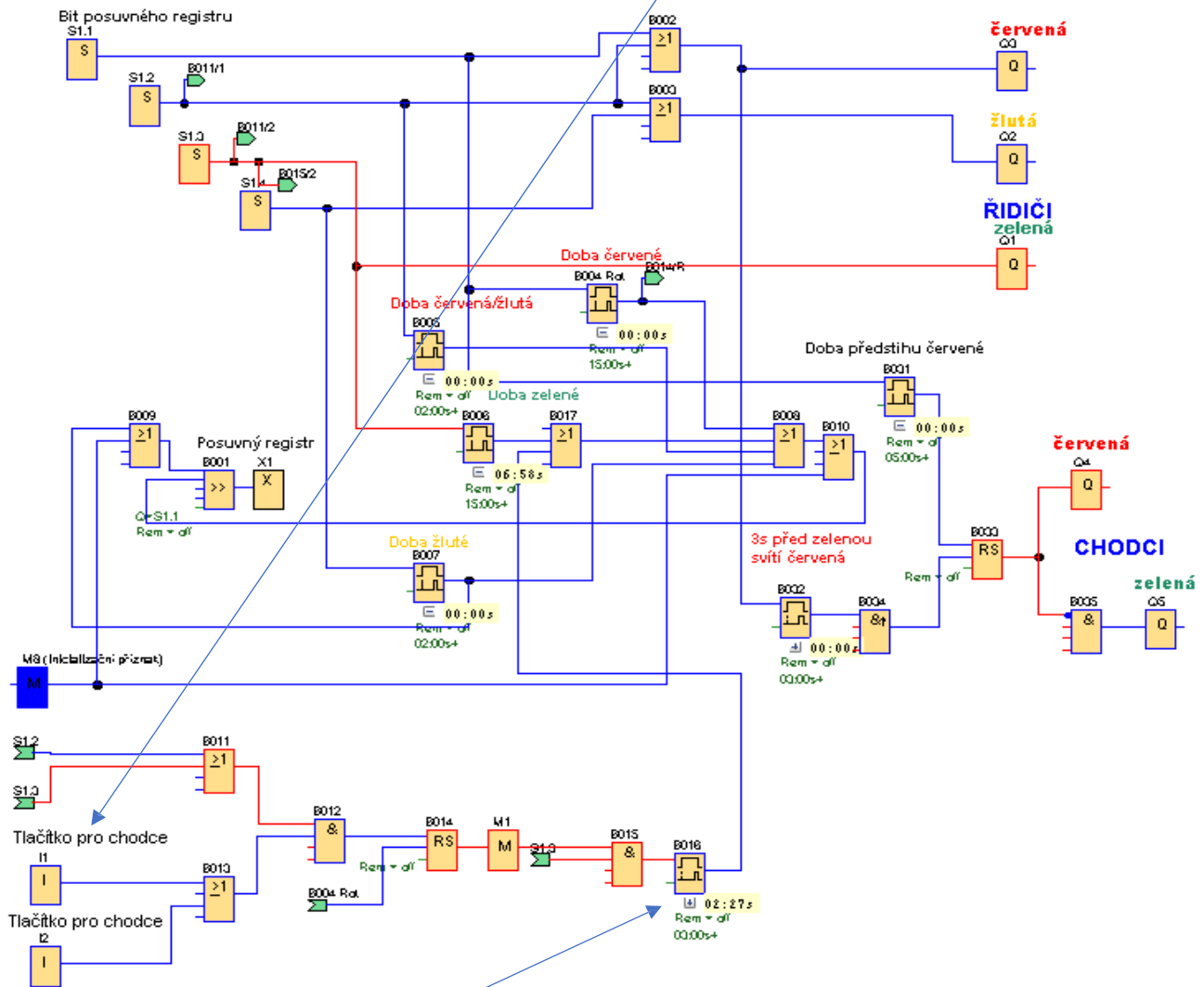
Svítí-li zelená je vstup do bloku AND "B012" jedna. Stiskem jednoho z tlačítek "I1, I2" se sepne výstup AND "B012", tím dojde k sepnutí výstupu z bloku RS "B014". Tak jsou na vstupu bloku AND "B015" 2 hodnoty 1 a výstup se sepne časování "B016". Po dokončení časování dostane impuls posuvný registr "B001" a posune bit na "S1.4". Zhasne zelená a rozsvítí se žlutá. Blok RS "B012" se resetuje po ukončení časování bloku "B004" (doba červené pro řidiče).



Celkové zapojení bloků

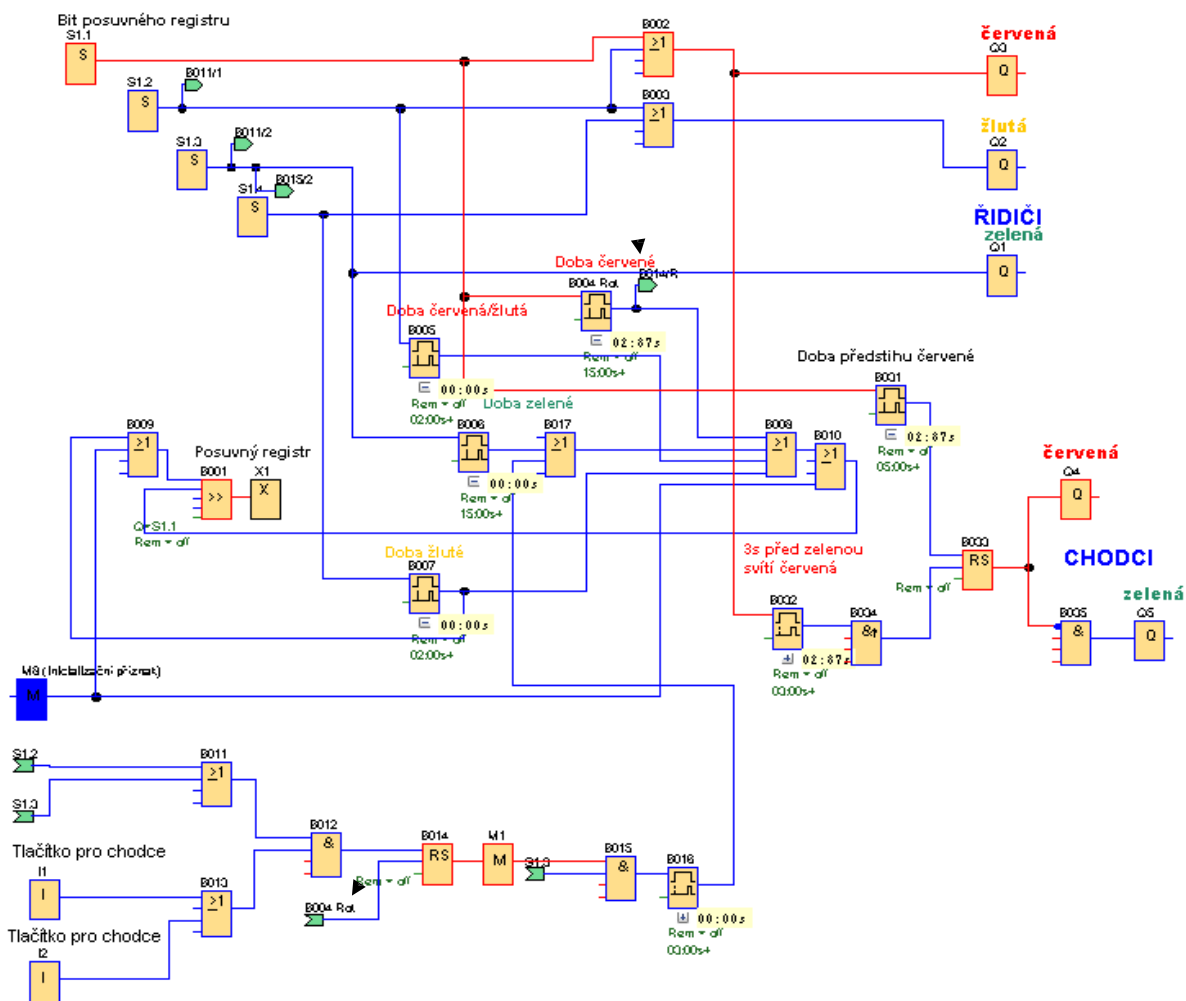


Stav po stisku tlačítka "I1" nebo "I2"



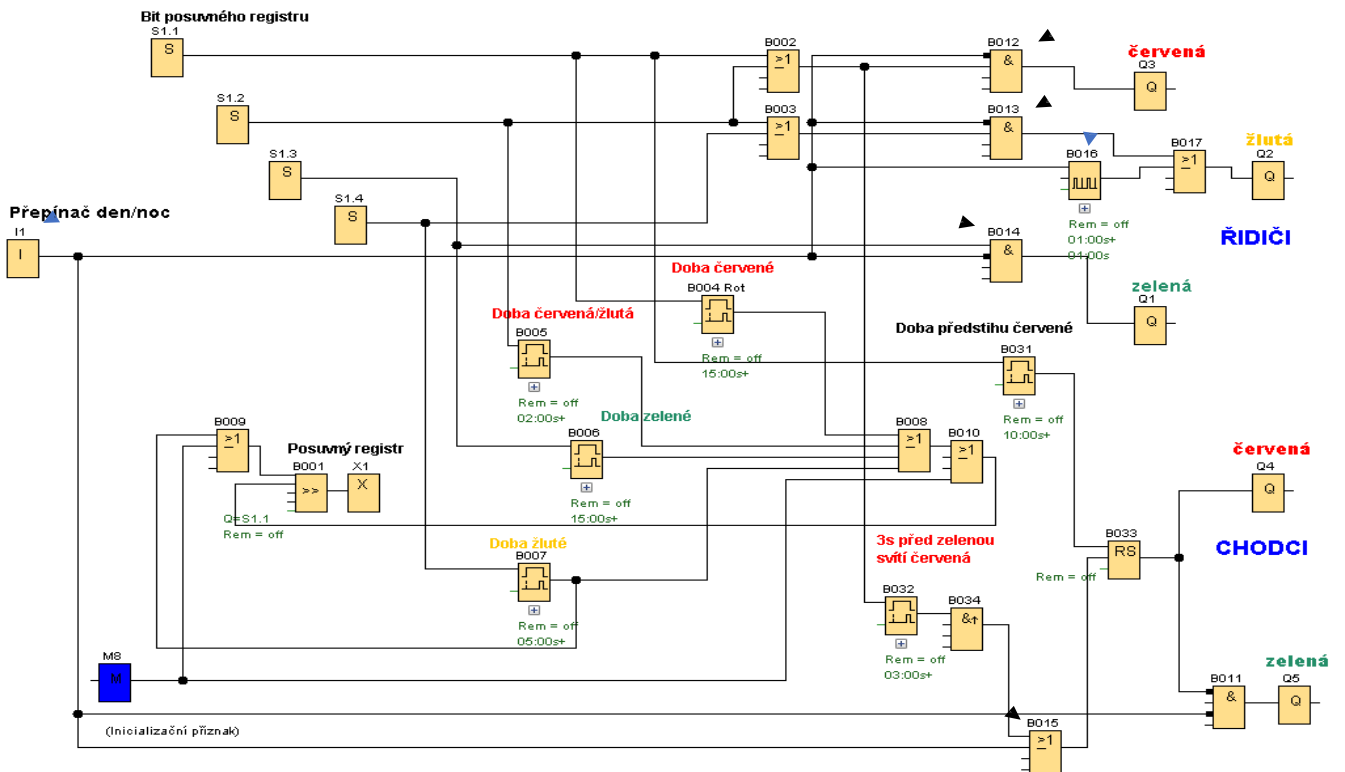
Blok zpožděného zapnutí "B016" časuje, jeho výstup ještě neseplul. Pro chodce svítí červená.

Posuvný registr byl přepnut z bitu S1.3 na bit S1.4 rozsvítla se žlutá a po ukončení časování žluté "B0007" se posune bit na S1.1 a rozsvítí se červená pro řidiče. To je stav na následujícím obrázku. Po dokončení časování červené "B004" se resetuje blok RS "B014".

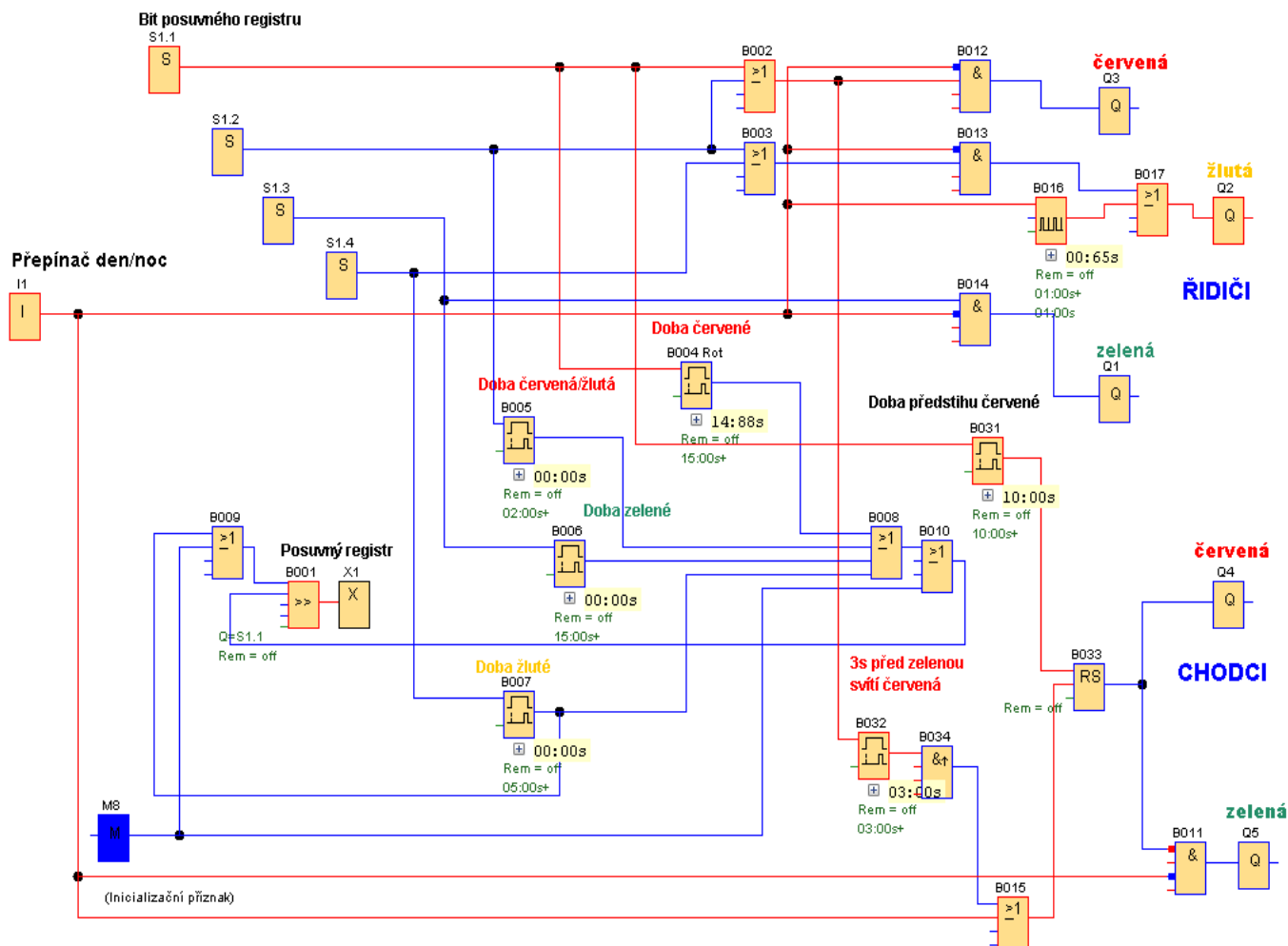


Příklad: Semafor 3

Příklad 1 je doplněn o přepínání pro denní a noční provoz. V noci je provoz nižší, proto se vypne střídání barev na semaforu a zapne se přerušované žluté světlo. Světla pro chodce budou vypnuta. Program 1 doplníme o přepínač "I1" o asynchronní generátor "B016" a bloky AND "B12", "B13", "B014", "B015". Nově přidané bloky AND blokují časovou změnu světel semaforu a vypnou je. Přepnutím přepínače "I1" se kromě blokace světel, aktivuje asynchronní generátor propojená přes blok "OR" a žluté světlo začne blikat.

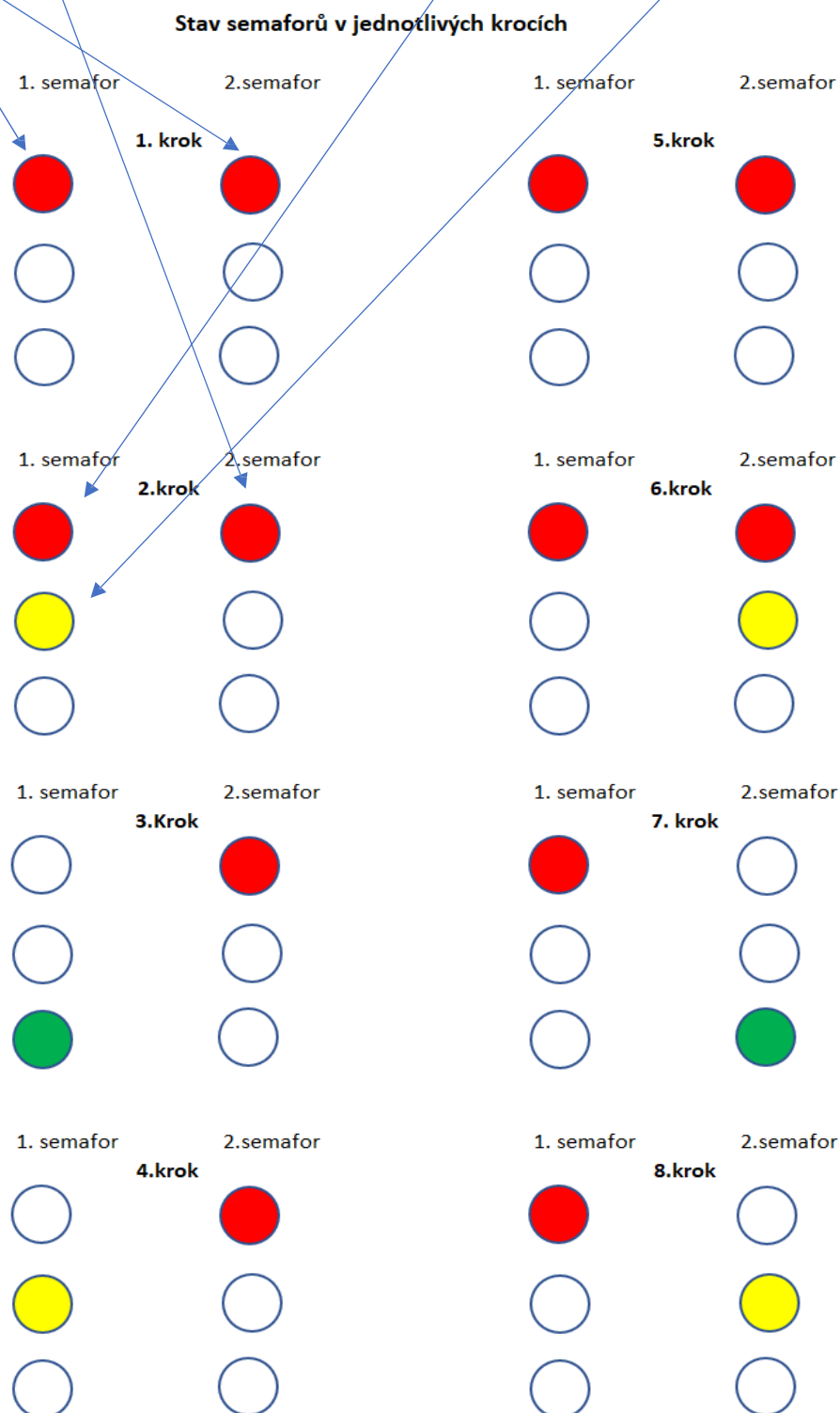


Stav po přepnutí přepínače "I1" do polohy noc. Bliká žluté světlo.



SEMAFOR 4

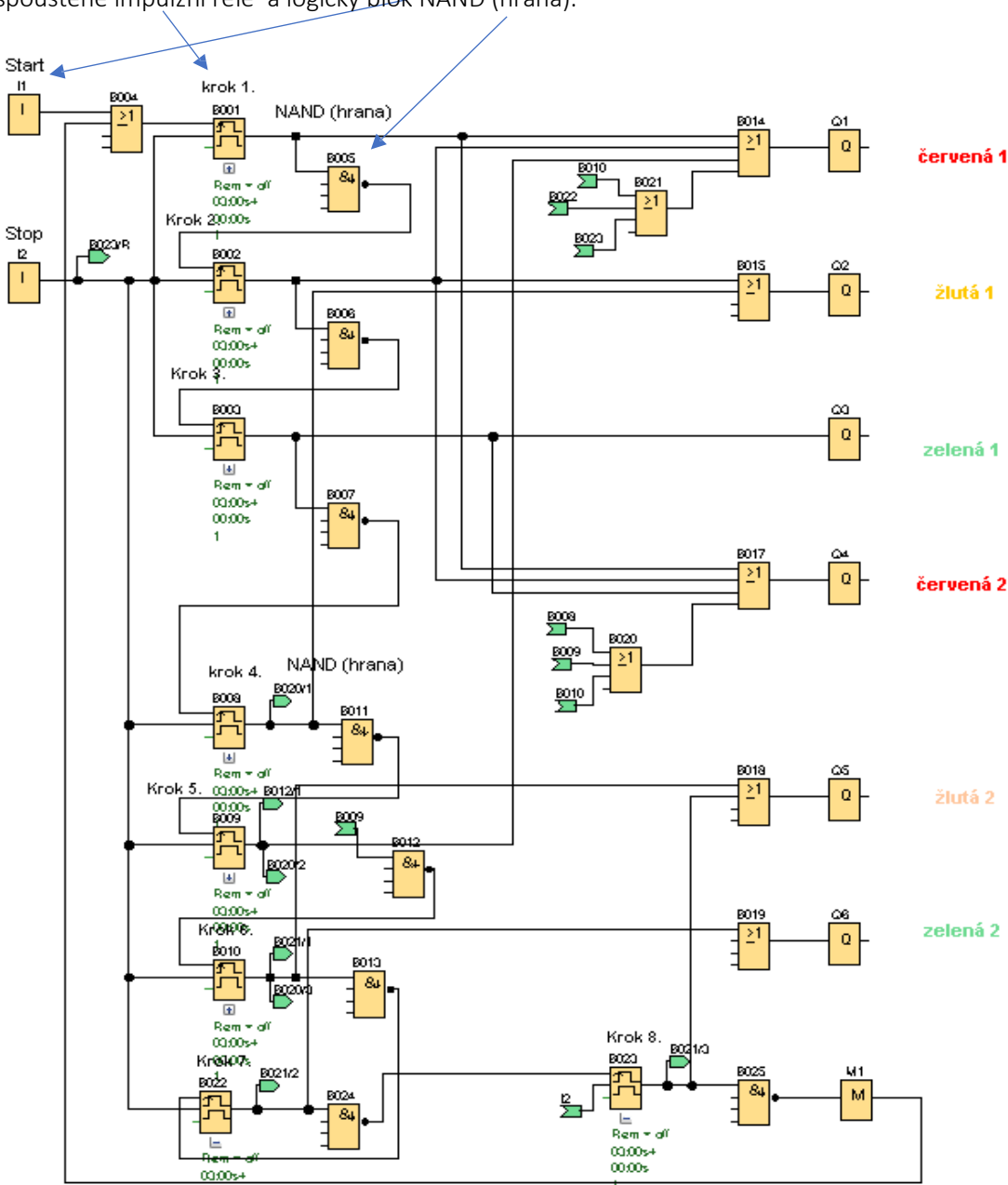
Na následujícím obrázku je vidět stav dvou semaforů v jednotlivých krocích vedle sebe. S tímto postupem spínání semaforů se setkáme např. při opravách silnice, kdy je zprovozněn střídavě jen jeden jízdní pruh, nebo na klasické křižovatce. Při spuštění semaforů svítí v obou směrech červená, ve druhém kroku svítí u prvního semaforu červená současně se žlutou. U druhého semaforu svítí červená. Další spínání světel je vidět na obrázku.



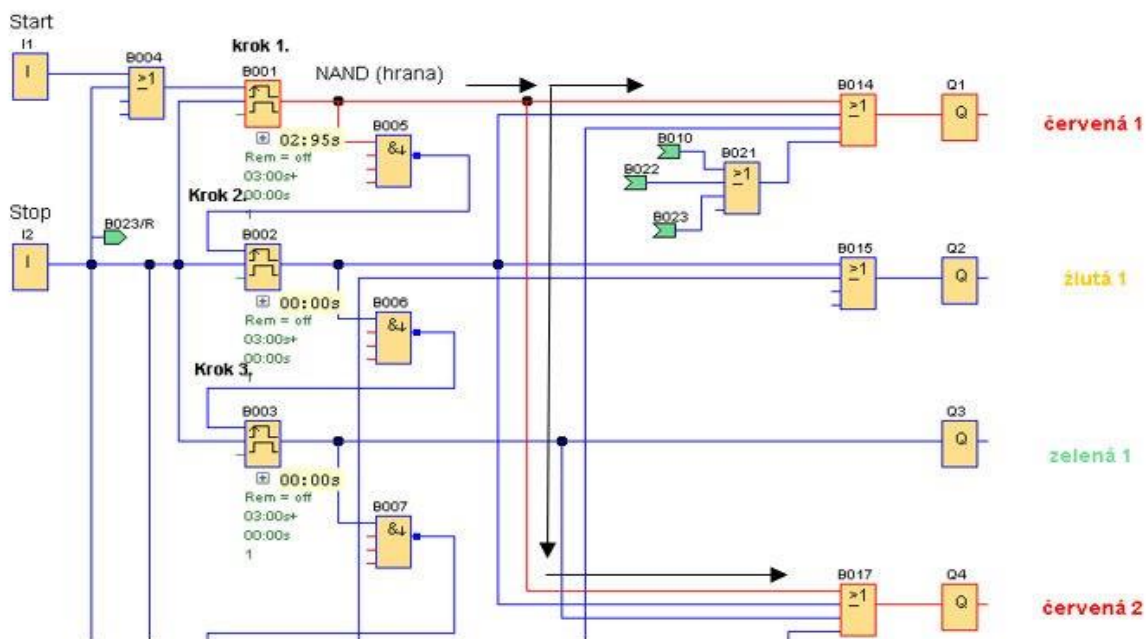
Stav semaforů v jednotlivých krocích je zapsán v tabulce .

barvy výstup	1. semafor			2. semafor			sepnuté výstupy
	červená Q1	žlutá Q2	zelená Q3	červená Q4	žlutá Q5	zelená Q6	
1. krok	1	0	0	1	0	0	Q1, Q4
2. krok	1	1	0	1	0	0	Q1, Q2, Q4
3. krok	0	0	1	1	0	0	Q3, Q4
4. krok	0	1	0	1	0	0	Q2, Q4
5. krok	1	0	0	1	0	0	Q1, Q4
6. krok	1	0	0	1	1	0	Q1, Q4, Q5
7. krok	1	0	0	0	0	1	Q1, Q6
8. krok	1	0	0	0	1	0	

Spouštění cyklování semaforů se provádí tlačítkem Start "I1". K posunu jednotlivých kroků slouží hranou spouštěné impulzní relé a logický blok NAND (hrana).

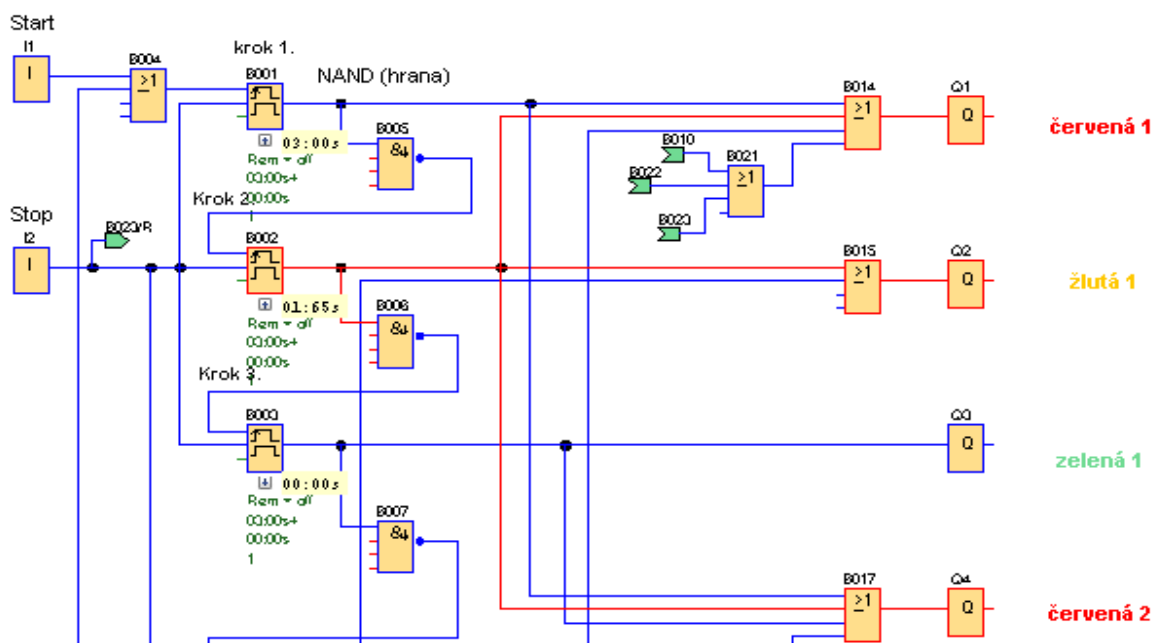


Po impulzu z tlačítka "I1", nebo z bloku "M1" na konci programu se spustí časování bloku "B001" a sepne se výstup "Q1" a "Q4" (černé šipky)



Po ukončení časování bloku "B001" se vypne jeho výstup, a protože blok NAND (hrana) vydá na výstupu impuls při přechodu na vstupu ze zapnuto na vypnuto spustí se časování bloku "B002".

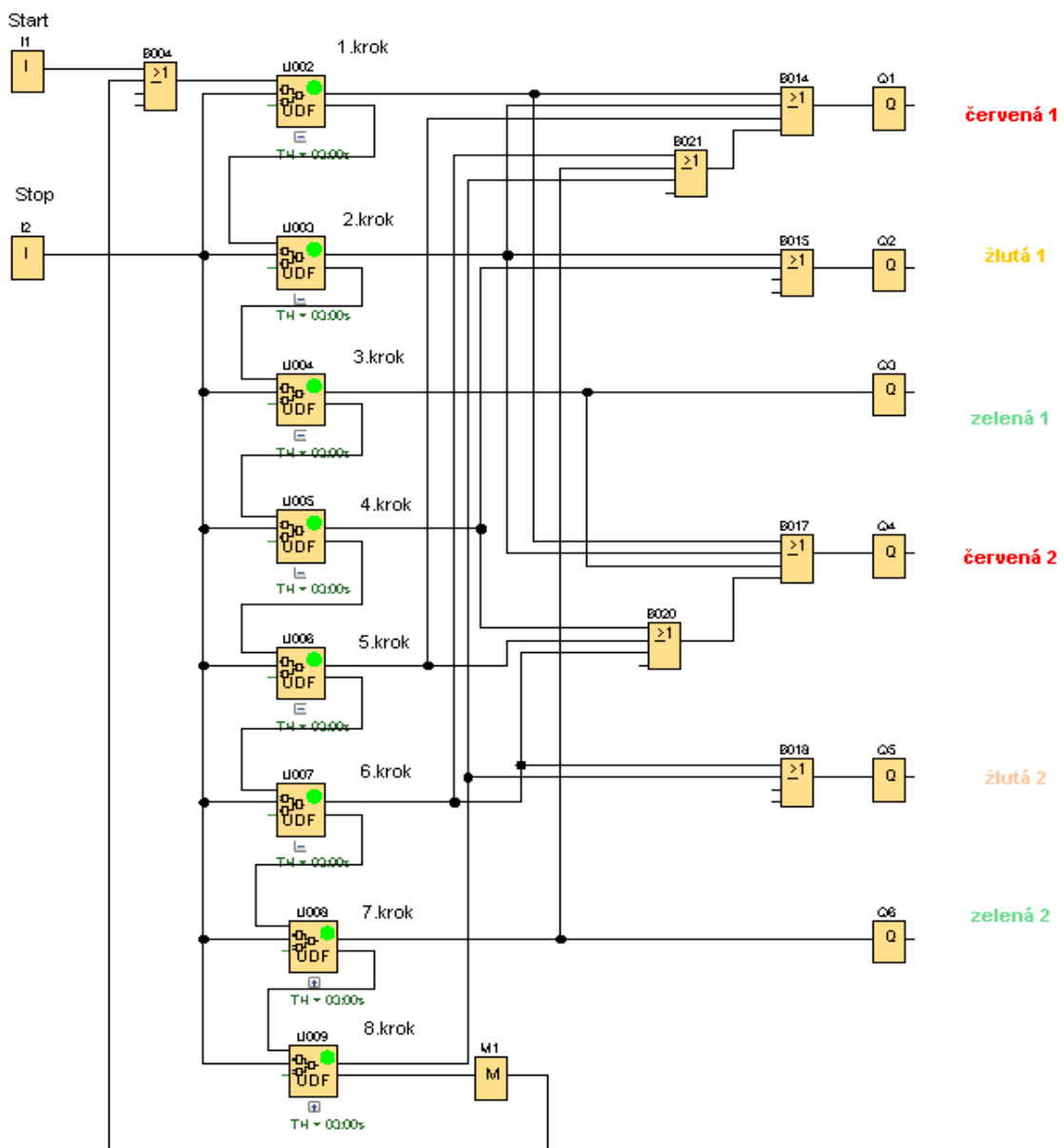
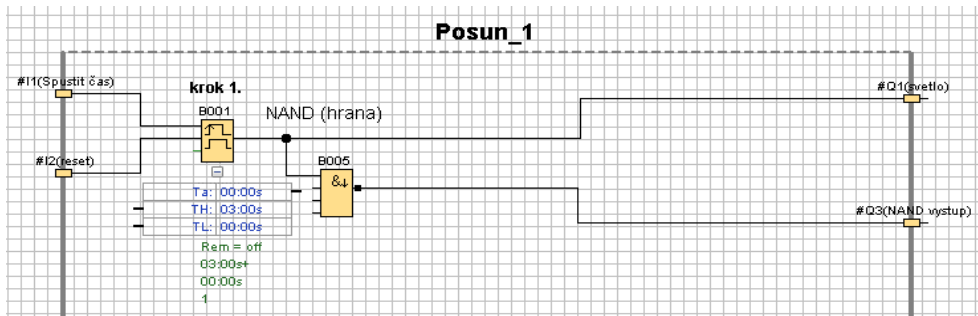
Tím se opět sepne výstup "Q1", dále výstup "Q2" a výstup "Q4". Sepnutí výstupů se děje propojením výstupu bloku hranou spínaného impulzního relé a bloků "OR" propojených se spínanými výstupy Q.



Stejným způsobem vždy po ukončení časování dojde k dalšímu kroku a propojí se výstupy zobrazené v tabulce.

Blok UDF

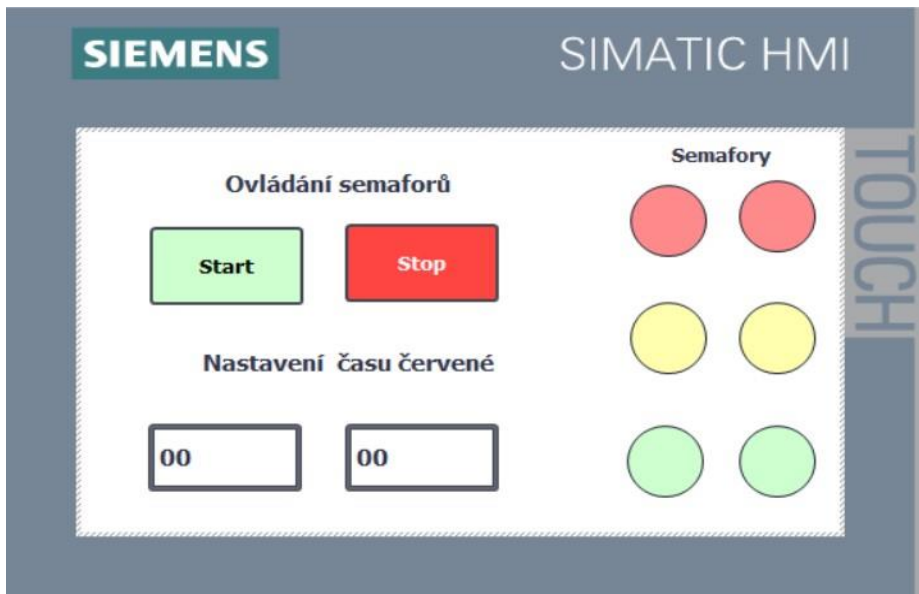
Protože se bloky hranou spouštěného impulzního relé a bloky NAND (hrana) opakují můžeme z nich vytvořit jeden blok UDF a tím zápis programu zřehlednit.



Panel HMI

Panel zobrazuje předchozí příklad s bloky UDF

Na panelu HMI můžeme sledovat spínání světel semaforů, měnit dobu rozsvícení červené, což je potřebné pro situaci, kdy je nutné zajistit zablokování průjezdu vozidel v obou směrech vzhledem k délce úseku omezeného úseku v jednom směru. Na uvedeném panelu je možné ovládat spuštění a zastavení semaforů, nastavení doby trvání červeného světla a sledování střídání barev na semaforech.

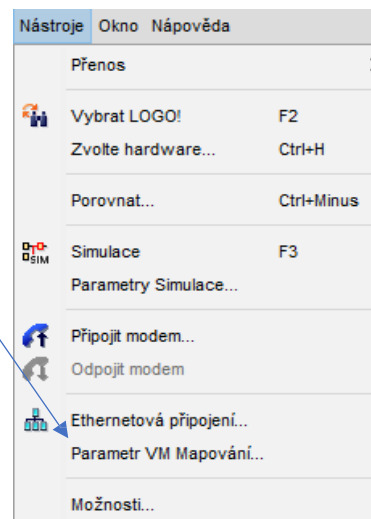


V LOGU v nastavíme adresy Bloků UDF

1. Nástroje

2. Parametr VM Mapování

3. Bloky pro červenou barvu



ID	Blok	Parametr	Typ	Adresa
1	U002 [UDF]	TH	Word	0
2	U004 [UDF]	TH	Word	2
3				

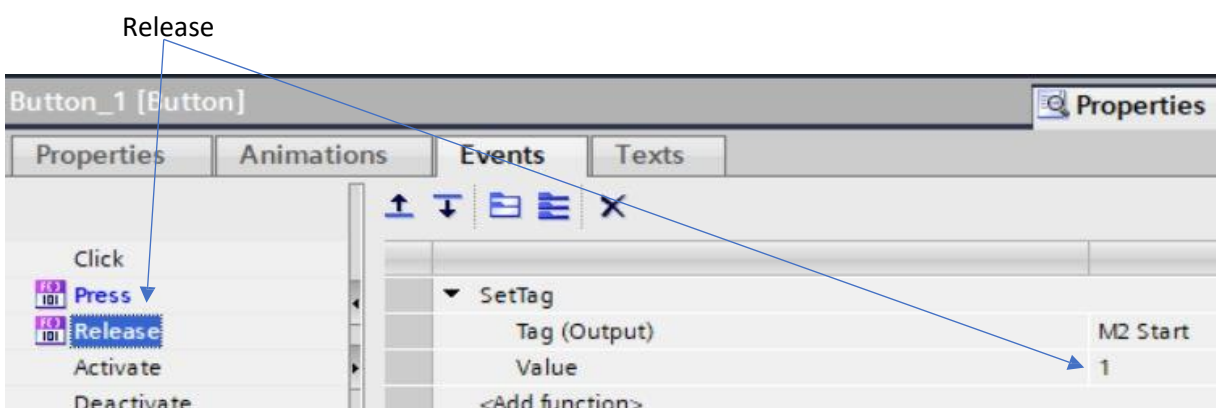
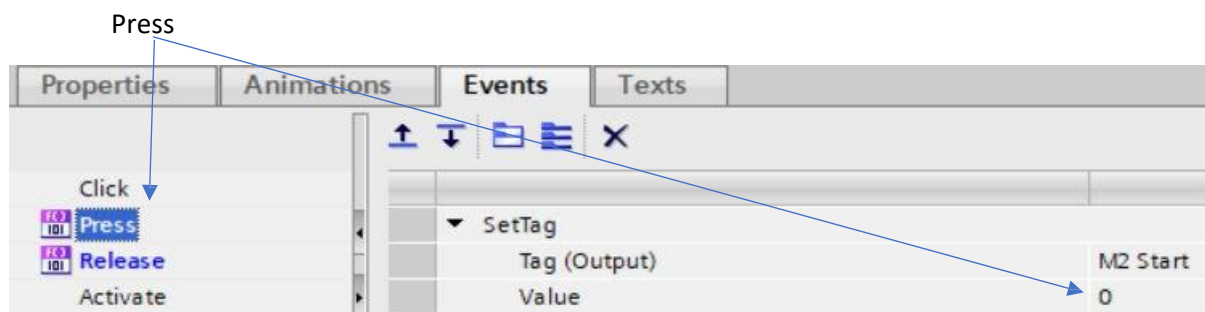
Podle postupu v II. dílu návodu naprogramujeme obrazovku

Nastavení v HMI tags

HMI tags						
	Name ▲	Data type	Connection	PLC n...	PLC tag	Address
🔍	cervena1	Word	Logo8		<Undefined>	VW 0
🔍	cervena2	Word	Logo8		<Undefined>	VW 2
🔍	M2 Start	Bool	Logo8		<Undefined>	M 0.1
🔍	M3 Stop	Bool	Logo8		<Undefined>	M 0.2
🔍	Q1 červená 1	Bool	Logo8		<Undefined>	Q 0.0
🔍	Q2 žlutá 1	Bool	Logo8		<Undefined>	Q 0.1
🔍	Q3 zelená 1	Bool	Logo8		<Undefined>	Q 0.2
🔍	Q4 červená 2	Bool	Logo8		<Undefined>	Q 0.3
🔍	Q5 žlutá 2	Bool	Logo8		<Undefined>	Q 0.4
🔍	Q6 zelená 2	Bool	Logo8		<Undefined>	Q 0.5

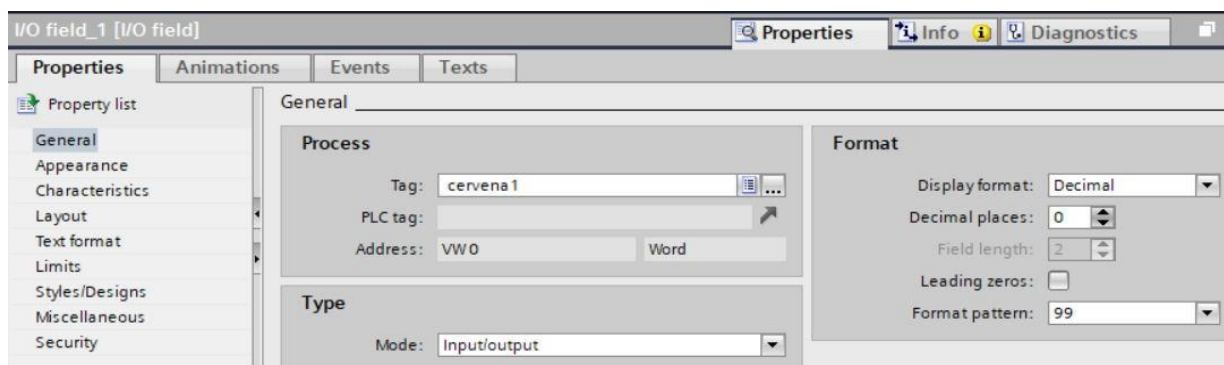
Nastavení pro tlačítko "Start"

1. Events
2. Klik na Press
3. Klik LT na rozbalovací šipku
4. Klik na Calculation
5. Klik LT na SetTag
6. Nastaví se:



Stejně se nastaví i tlačítko "Stop"

Nastavení zobrazovače pro čas červená1

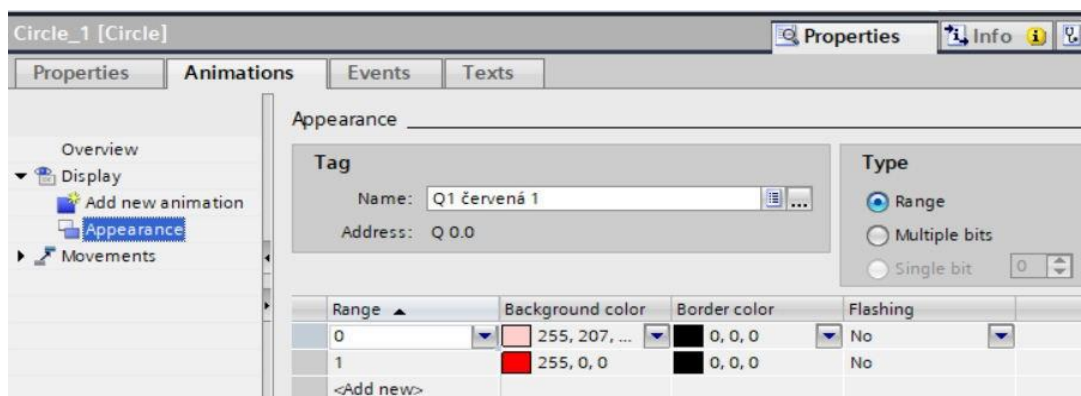


Čas na panelu změníme stiskem pole zobrazení času. Na klávesnici přepíšeme čas a potvrdím Enter



Stejně nastavíme i druhý zobrazovač pro červenou na druhém semaforu.

Nastavení změny barev semaforu: Při sepnutí výstupu se změní růžová na červenou.

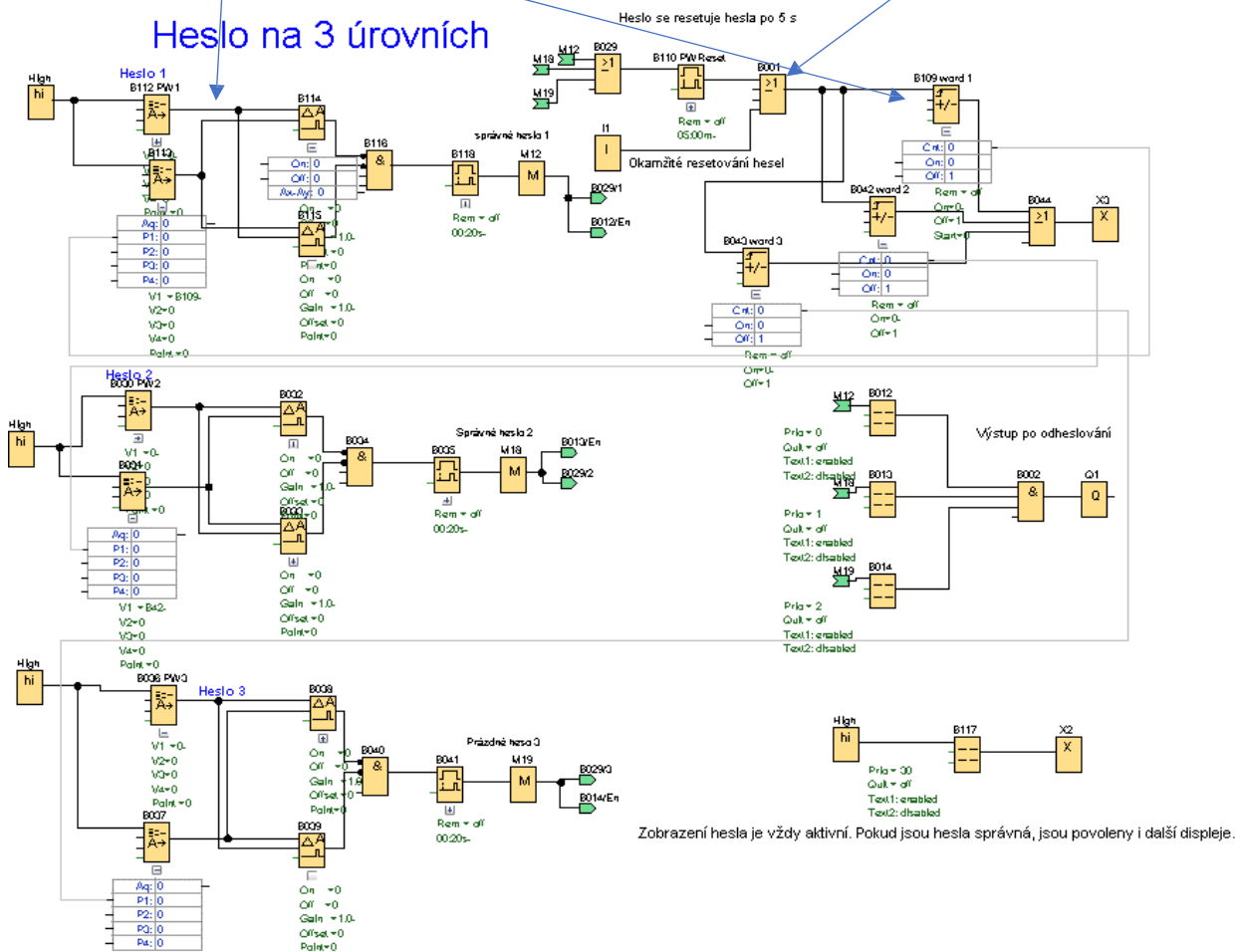


HESLO

Příklad je převzatý ze stránek Siemens. Jeho účelem je spuštění výstupu "Q1" po zadání tří číselných hesel. Například tři stupňové zabezpečení spuštění technologie, vstup do bezpečnostní oblasti apod. Jsou-li všechna tři hesla správně zadána sepne se "Q1". Hesla se mohou zadávat přes displej LOGA za pomoci kurzorů. V běhu LOGA se najde obrazovka, kam se mají zadávat hesla, podrží se ESC alespoň 1-2s, rozblíknají se vnitřní parametry, přiřazené na displeji. Po rozblíknání se může měnit jejich hodnota šipkami nahoru/dolů a poté se potvrdí tlačítkem OK. Šipkami doprava/doleva se dostaneme na další parametry na obrazovce. Další možností nastavení hesel je panel HMI.

V levé části je soubor bloků, které porovnávají zadané číslo hesla s nastavenou konstantou, jsou-li hodnoty stejné sepne se výstup z bloku AND

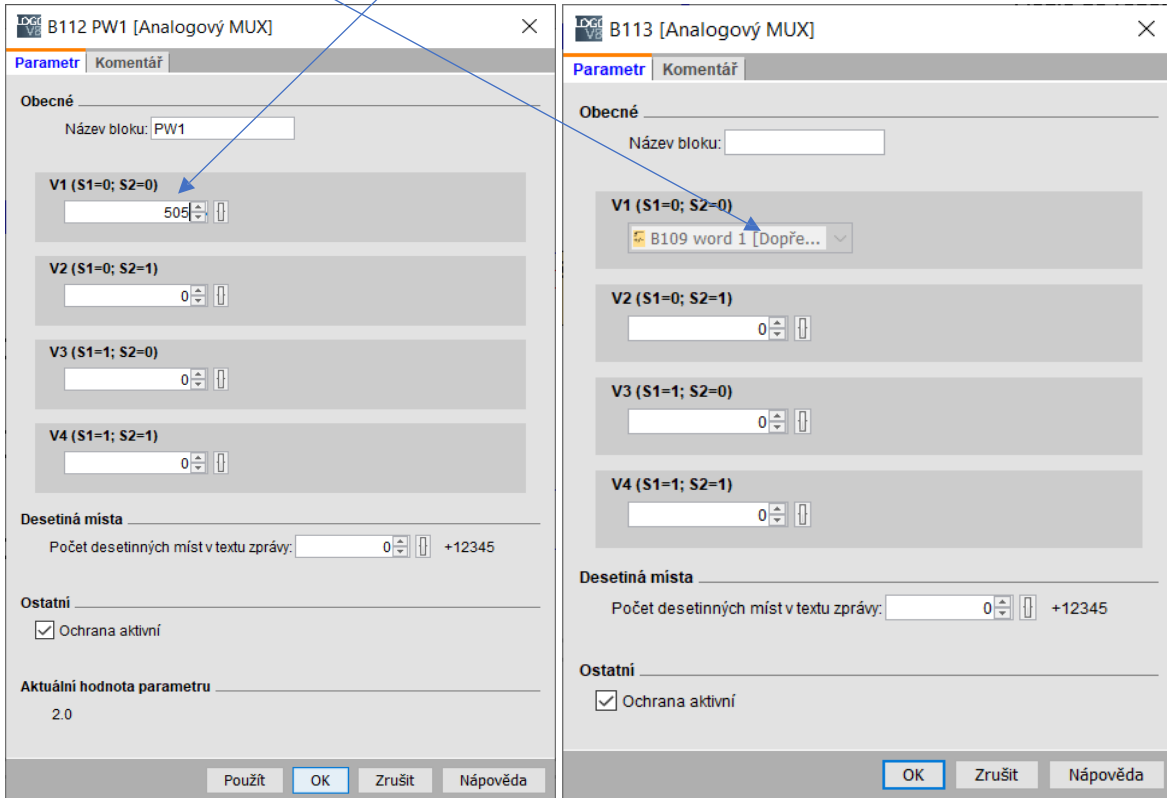
V pravé části jsou čítače, kde se nastavuje číselná hodnota hesla. Výstupem z "OR" se resetují čítače.



Resetování čítačů se provádí tlačítkem "I1", nebo automaticky po nastavené době.

Princip funkce:

V programu jsou použity bloky "Analogový MUX". Je jich celkem 6. Vždy dva pro jedno heslo. Vrchní blok slouží pro zapsání hodnoty hesla. Je uloženo trvale. Spodní blok má parametrové propojení s čítačem, do kterého zapisujeme hodnotu hesla pomocí displeje LOGA, nebo panelu HMI

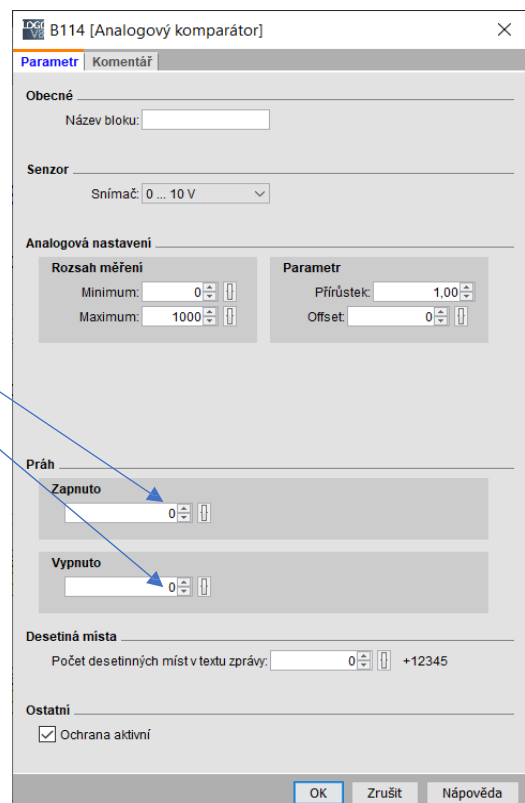
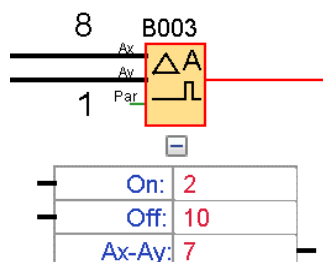


Další použité bloky jsou "Analogové komparátory". Ty porovnávají výstupní hodnoty z bloků "Analogový MUX"

Stav analogového komparátoru zapnutí a vypnutí je nastaven na hodnotu "0"

Princip komparátoru (hodnoty nesouvisí s příkladem):

Komparátor se sepne, když rozdíl vstupů $A_x - A_y$ dosáhne hodnoty ON a vypne po dosažení rozdílu hodnoty QFF. V příkladu je $A_x = 8$, $A_y = 1$, Rozdíl je 7. Výstup je sepnutý, protože " $A_x - A_y$ " leží v intervalu 2 až 10.

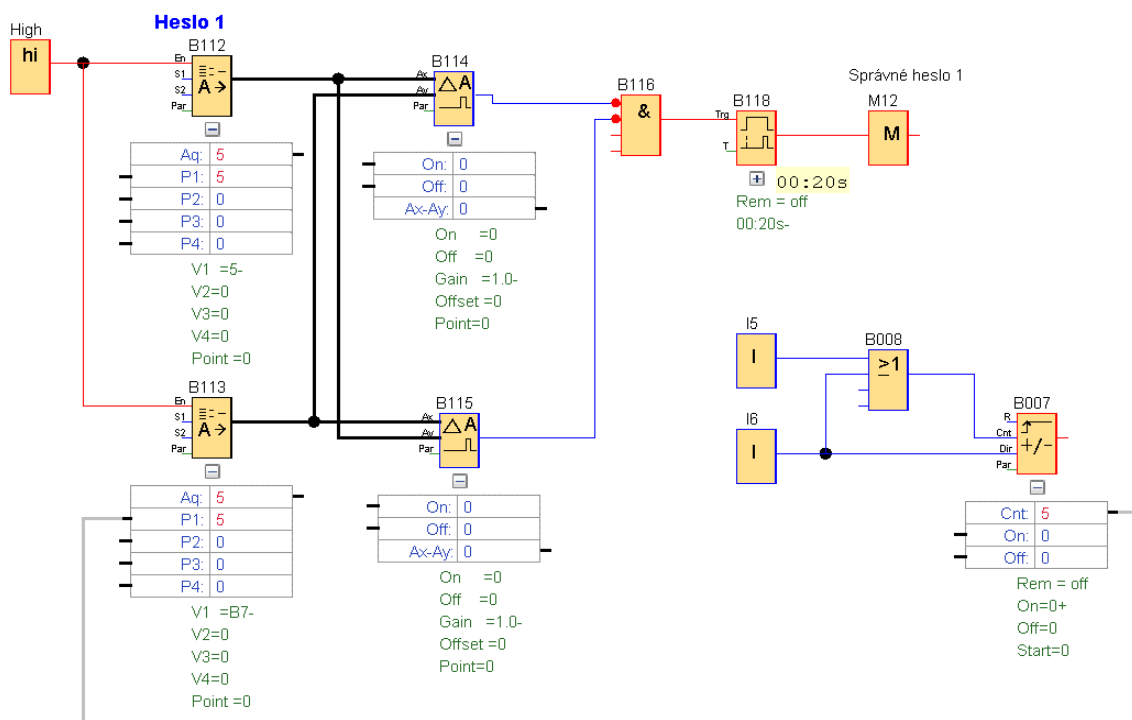


Podle zapojení použitým v příkladu Heslo má komparátor při nastavení (ON = 0, OFF = 0) jediný **stav vypnutí** při dosažení stejných hodnot na vstupech, a tedy rozdílu $A_x - A_x = 0$.

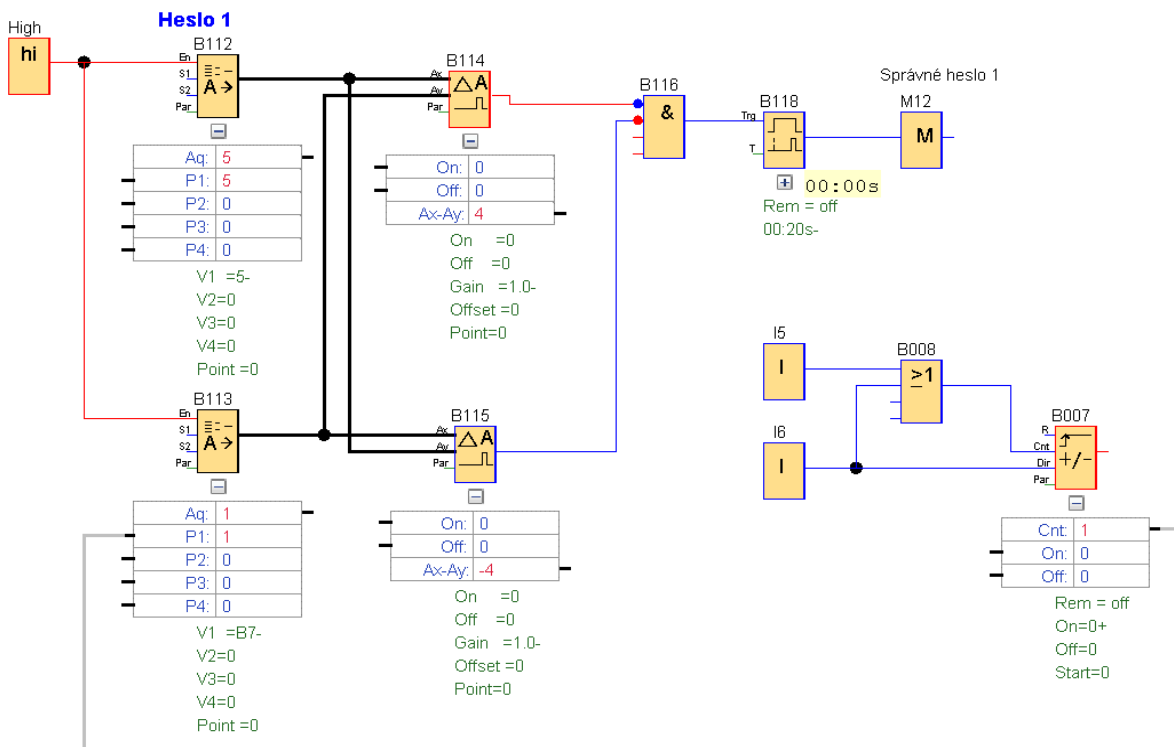
V tabulce jsou tři stavy, které lze v daném zapojení docílit.

var.	B112	B113	B114				B115				neg. výst.	neg. výst.	B016
			Ax	Ay	Ax-Ay	výstup	Ax	Ay	Ax-Ay	výstup	B114	B115	
1.	5	5	5	5	0	0	5	5	0	0	1	1	1
2.	5	1	5	1	4	1	1	5	-4	0	0	1	0
3.	5	6	5	7	-1	0	6	5	1	1	1	0	0

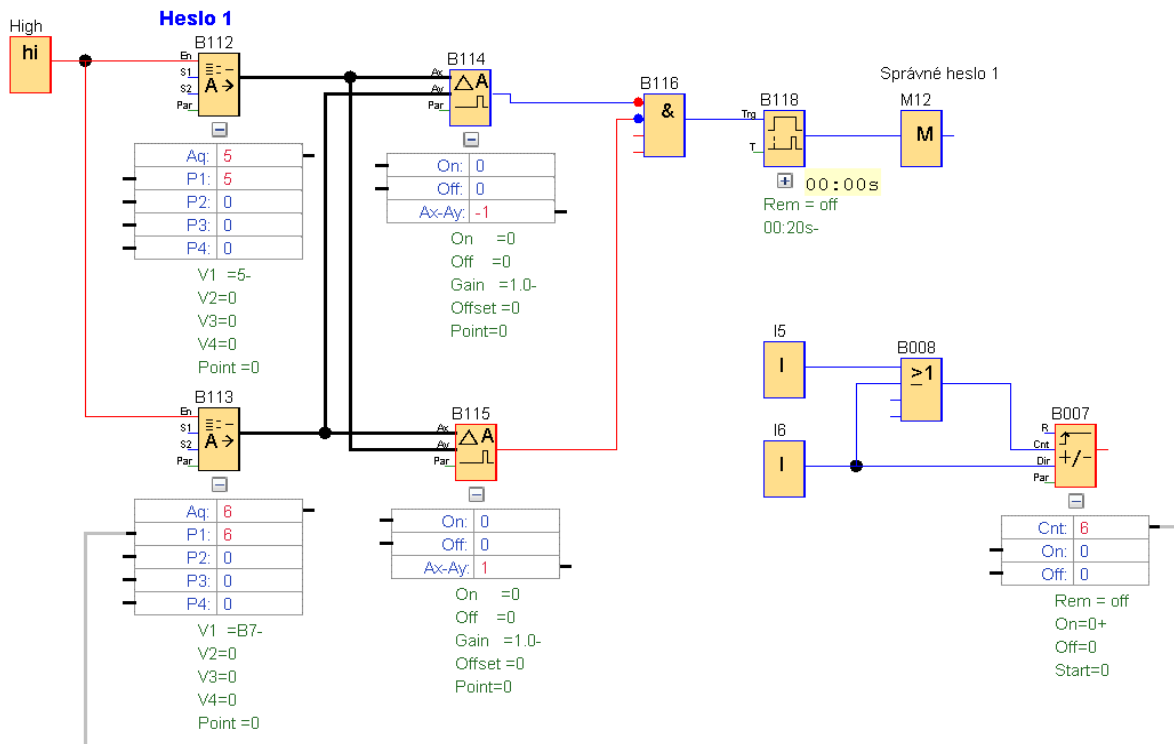
1. stav - hodnoty v blocích Analogový MUX jsou shodné (je zapsáno správné heslo)



2.stav - hodnota bloku "B012" je vyšší než hodnota bloku "B113"

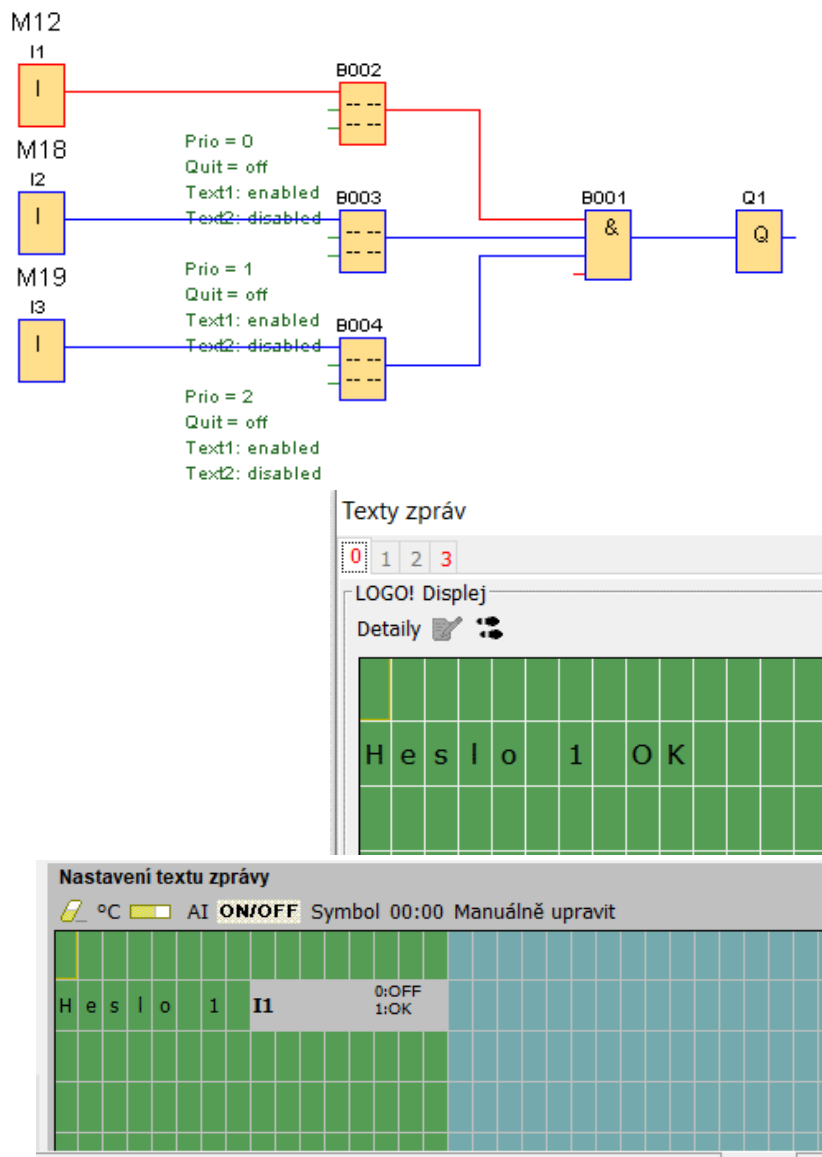


3.stav - hodnota bloku "B012" je nižší než hodnota bloku "B113"

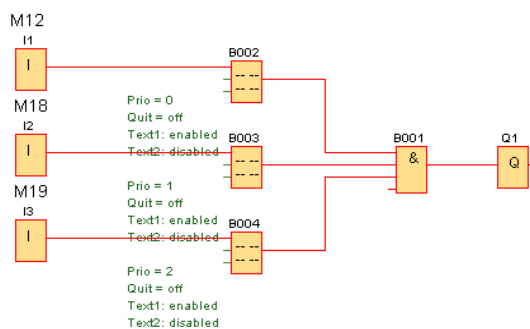


Pro zpřehlednění výkladu byly výstupy M 12, M18, M19 jednotlivých hesel nahrazeny vstupy "I1, I2, I3".

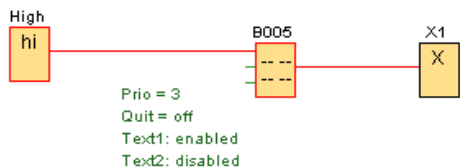
Pokud je heslo správně zadáno sepne se výstup z "M12", tím je aktivován displej a jeden vstup bloku "AND".



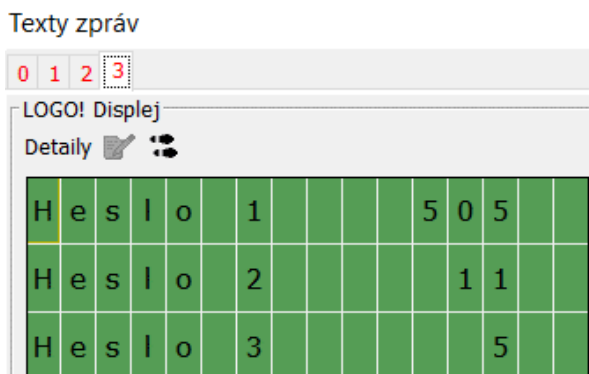
Zadají-li se správně všechna tři hesla sepne se výstup "Q1", který odblokuje chráněné části.



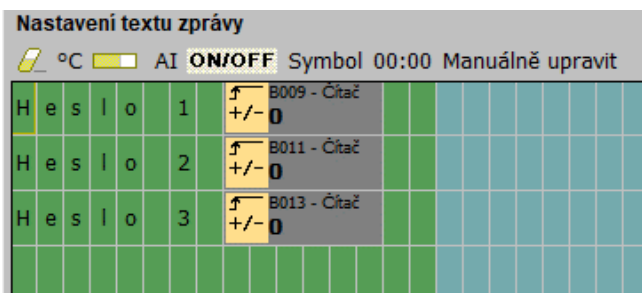
Textová zpráva v tomto zapojení zobrazuje hodnoty čítačů



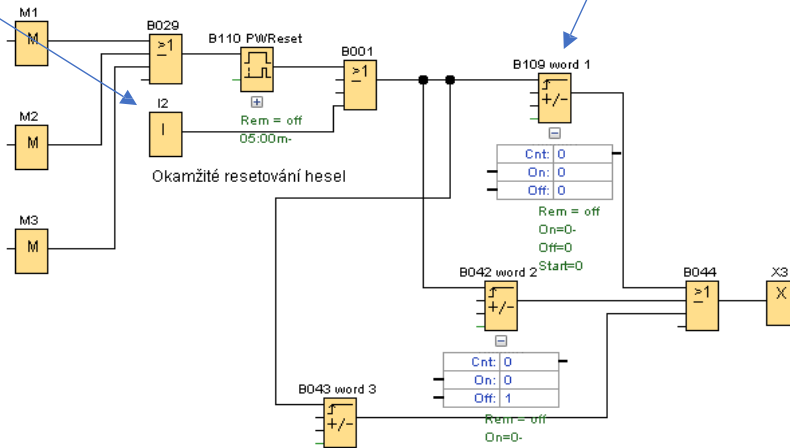
Zobrazení na displeji



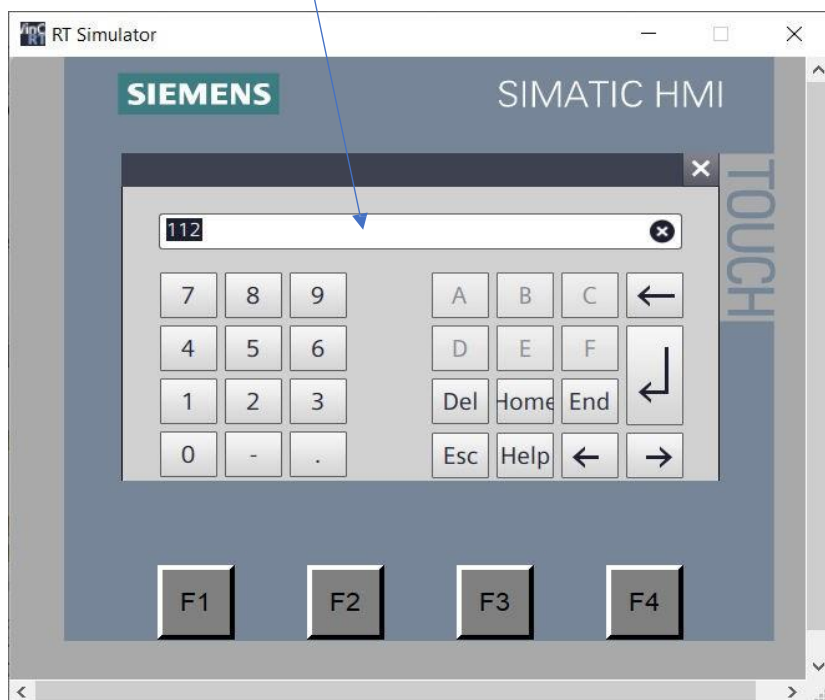
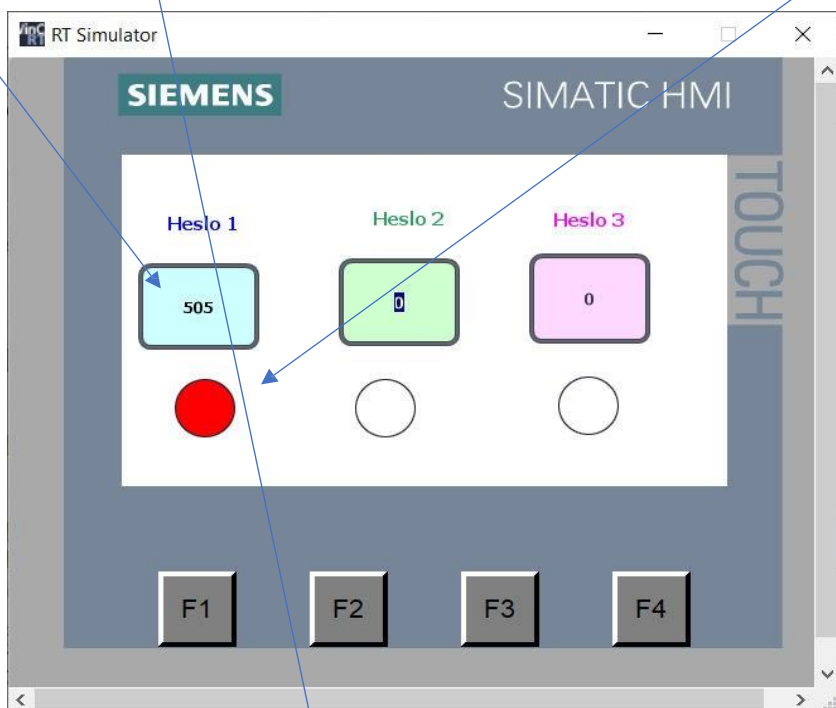
Nastavení textu a čítačů



Poslední částí programu je resetování čítačů, nebo-li hesla. Heslo je možné resetovat tlačítkem "I2", nebo se čítač vynuluje automaticky po nastavené době blokem "B109" (zpožděné zapnutí).



Zadávání hesel můžeme provést i pomocí panelu HMI. V uvedeném příkladu obrazovky se stiskem obdélníku zobrazí klávesnice, na níž zadáme heslo. Je-li heslo správné rozsvítí se červená kontrolka na obrazovce.



Zadání adres použitých na obrazovce

Name	Data type	Connection	PLC na...	PLC tag	Address
Heslo 1_čítač_B109	DWord	LOGO 8		<Un defi...	VD 0
Heslo2_čítač_B042	DWord	LOGO 8		<Un defi...	VD 4
Heslo3_čítač_B043	DWord	LOGO 8		<Un defi...	VD 8
M12_heslo1	Bool	LOGO 8		<Un defi...	M 1.3
M18_heslo2	Bool	LOGO 8		<Un...	M 2.1
M19_heslo3	Bool	LOGO 8		<Un defi...	M 2.2

Podrobný popis programování panelu HMI je v druhém dílu LOGO! Krok za krokem II.

VYTÁPĚNÁ BOUDA PRO KOCOURA

autor programu Ing. Jan Tlučoř

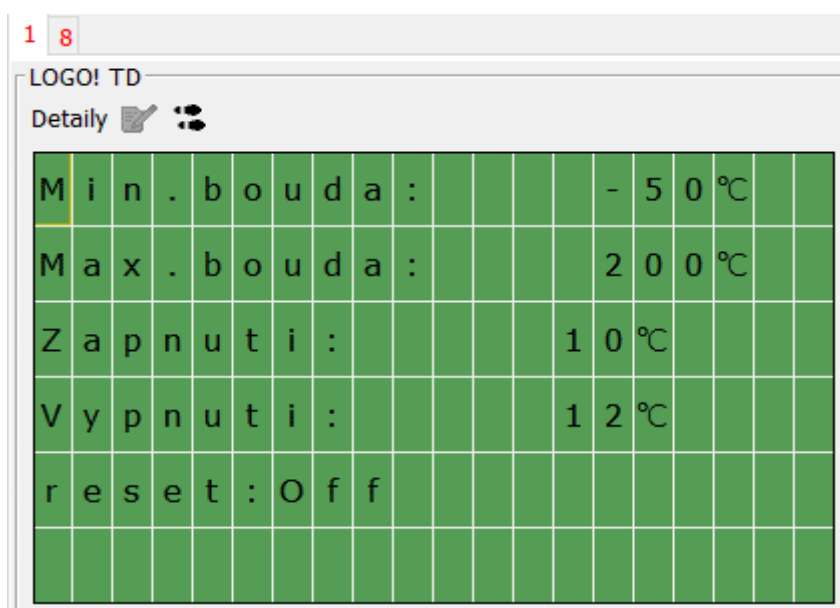
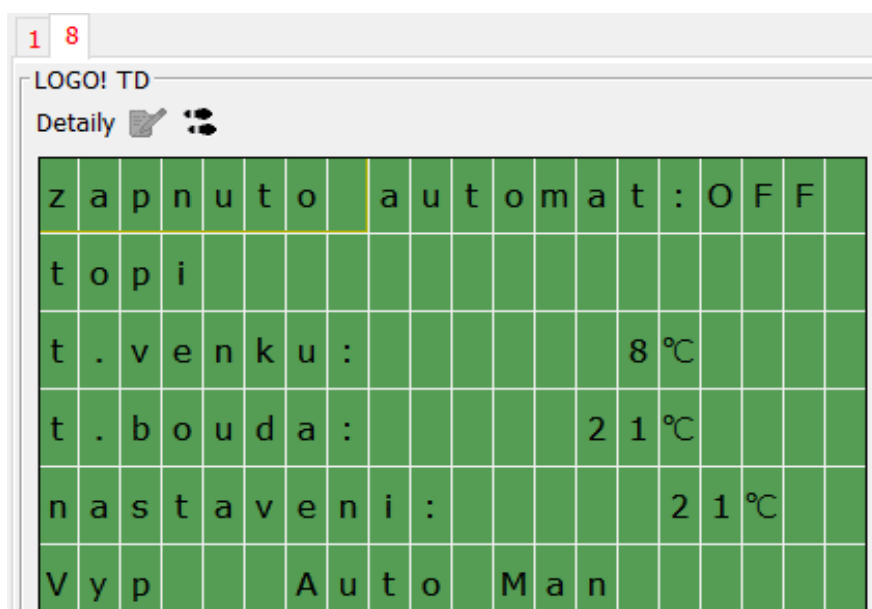
Místo pro ležení kocoura je vytápěno elektrickou topnou rohoží umístěnou pod dlaždicí. Tam je také teplotní čidlo Pt 100 "AI1". Druhé teplotní čidlo "AI2" je umístěné venku. Vytápění spíná při venkovní teplotě nižší 10°C a vypíná při 12°C. Teplota vytápěné dlaždice je nastavena na 21 °C. Ovládání je provedeno na venkovním panelu LOGO! TD.

Klávesou F1 se vytápění vypne

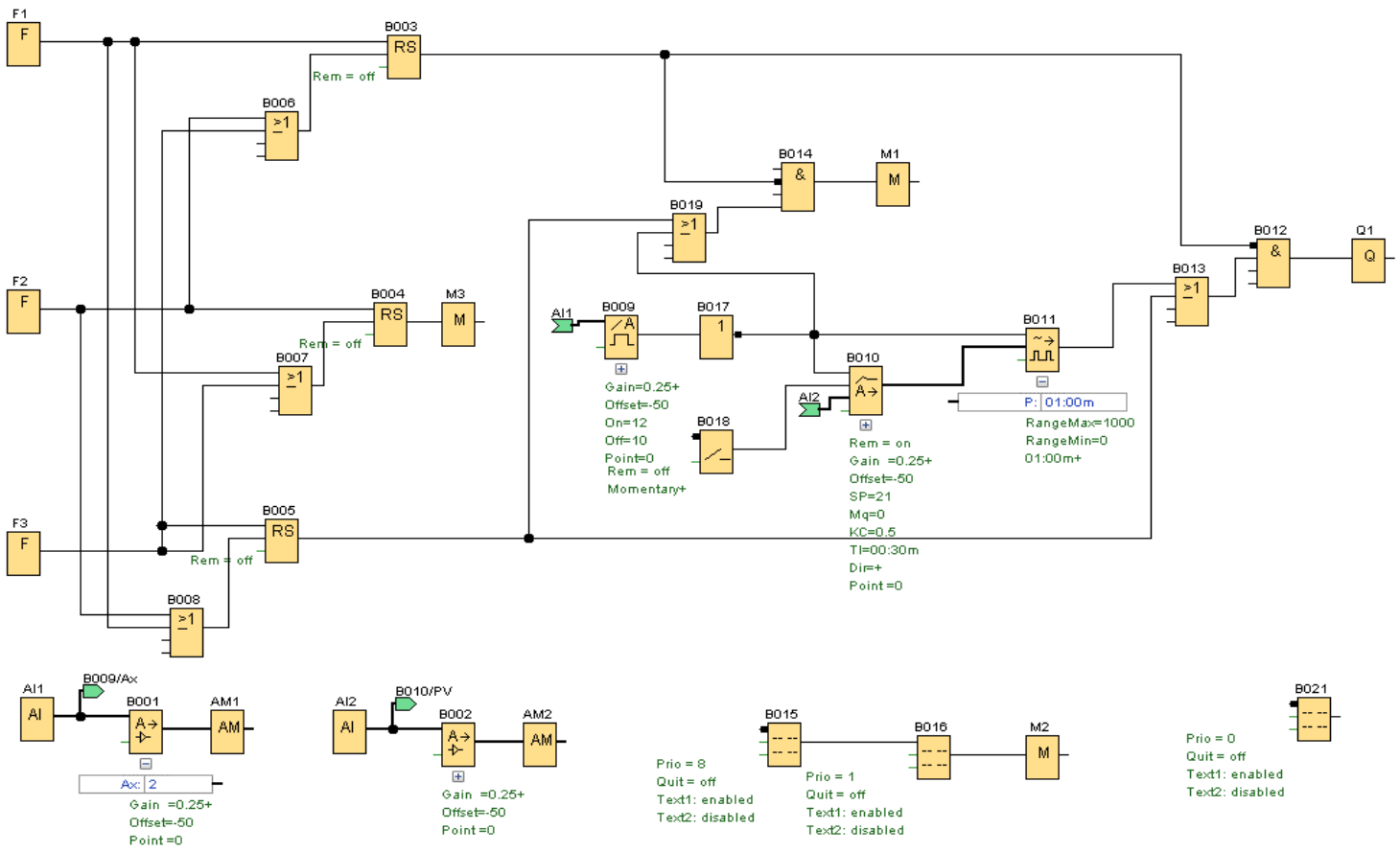
Klávesou F2 se zapne automatická regulace v závislosti na snímaných teplotách

Klávesou F3 se zapne trvalé vytápění, nezávislé na teplotách

Stav vytápění je možné sledovat na přepínatelných obrazovkách LOGA! TD.

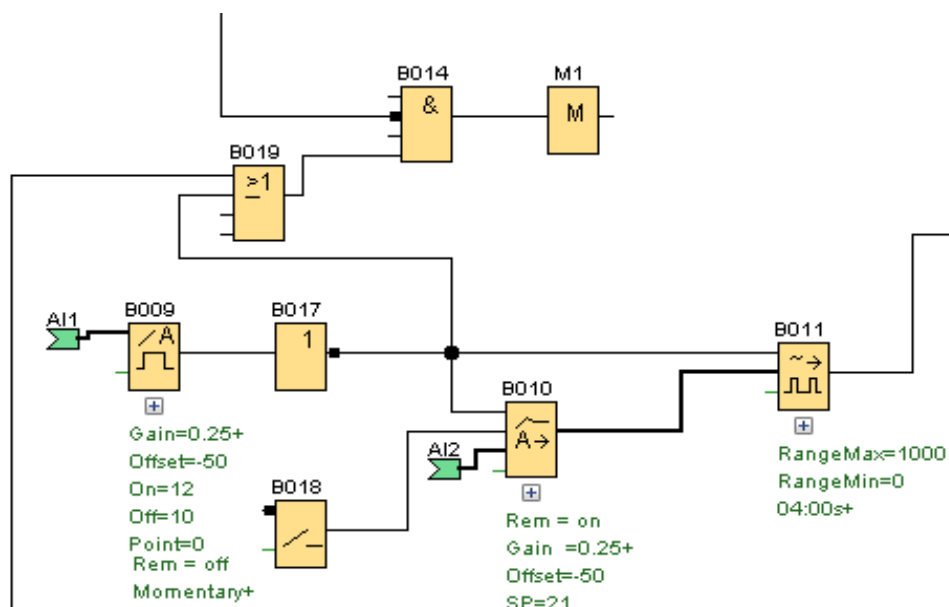


Program



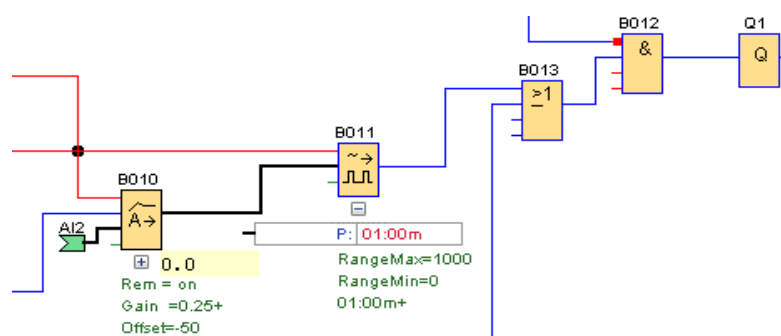
Automatická regulace

Omezení vytápění nad 12°C je dáno Analogovým spínačem "B009" a negací jeho výstupu. Při teplotě vyšší než 12°C se sepne výstup z Analogového spínače neguje se blokem NOT "B017" a tím se vypnou vstupy do bloku PI regulátor "B010" a PWM "B011", tím následně nedojde k sepnutí Výstupu Q1.

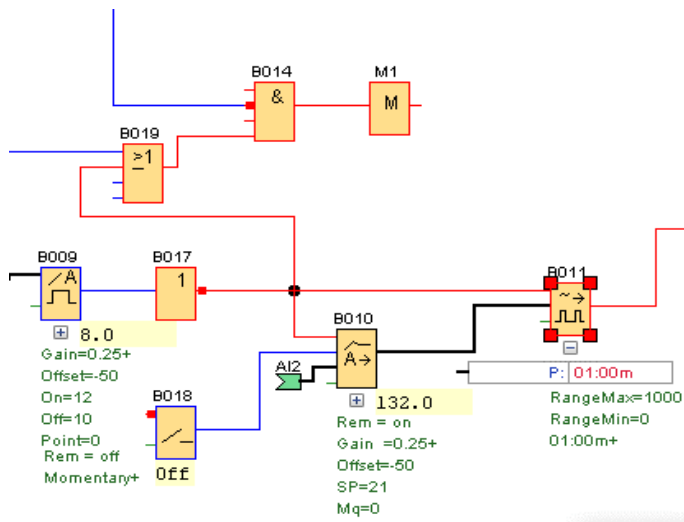


PI regulátor "B010" je spojitý regulátor. Jeho výstup je analogový, protože výstup Q1 může být sepnutý nebo vypnutý, převede se analogová hodnota na digitální pomocí bloku Pulzně šířkový modulátor PWM "B011". Na něm se nastaví doba pulzu. Zde je 1 minuta. Na poměru vstupní analogové hodnoty a nastaveného rozsahu měření se rozdělí doba trvání sepnutí a vypnutí.

Př.: Rozsah měření je 0 až 1000. Doba pulzu je 60s. Vstupní hodnota je 250. Výpočet doby sepnutí : $250/1000 = x/60 \Rightarrow x = 15s$. Doba sepnutí bude 15s a doba vypnutí 45 s. Čím blíže k nastavené hodnotě se bude teplota blížit, tím kratší bude doba sepnutí.



Blok M1 je zde z důvodu zobrazení na displeji, kde se zobrazí topí nebo netopí. Jeho sepnutí je závislé na stavu z výstupu "B017".



Nastavení parametrů v uvedeném příkladu programu

B009 [Analogový spínač]

Parametr | Komentář

Obecné
Název bloku: _____

Senzor
Snímač: PT100/PT1000

Analogová nastavení

Rozsah měření	Parametr
Minimum: -50	Přírůstek: 0,25
Maximum: 200	Offset: -50

Jednotka
 Celsia
 Fahrenheita

Rozlišení
 x 1
 x 0.1

Prahové hodnoty

Zapnuto: 12

Vypnuto: 10

Desetinná místa
Počet desetinných míst v textu zprávy: 0 +12345

Ostatní

OK Zrušit Nápověda

B010 [PI regulátor]

Parametr | Komentář

Obecné
Název bloku: _____

Senzor
Snímač: PT100/PT1000

Analogová nastavení

Rozsah měření	Parametr
Minimum: -50	Přírůstek: 0,25
Maximum: 200	Offset: -50

Jednotka
 Celsia
 Fahrenheita

Rozlišení
 x 1
 x 0.1

Výstup

Nastavené hodnota (SP): 21

Manuální výstup (Mq): 0

Parametr:

Sada parametrů: Teplota rychle

Zesílení regulátoru (KC): 0,50

Integrovaná konstanta (TI): 0 : 30 Minuty (m:s)

Směr (Dir): Nahoru (+)
 Dolů (-)

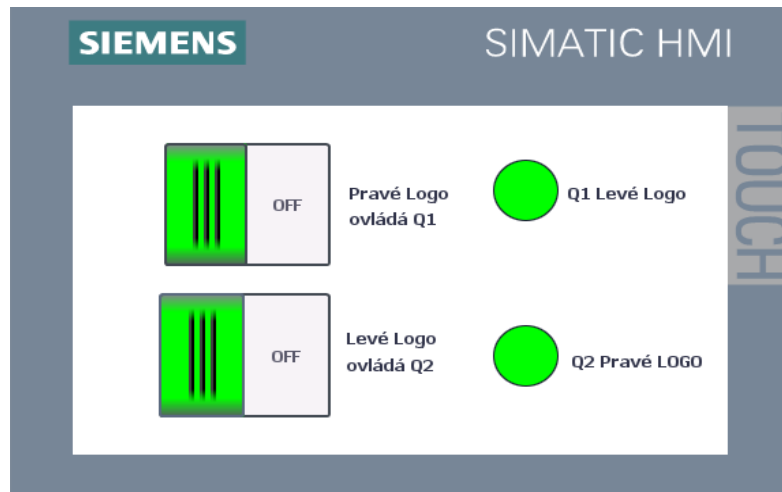
Desetinná místa
Počet desetinných míst v textu zprávy: 0 +12345

Ostatní
 Paměť
 Ochrana aktivní

OK Zrušit Nápověda

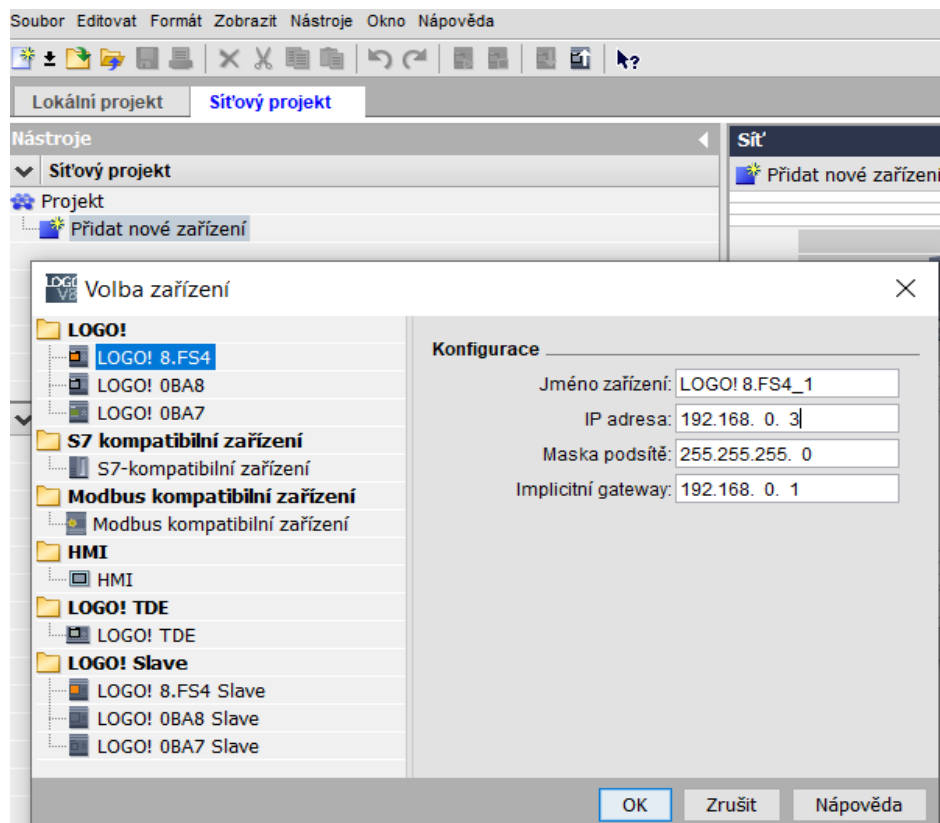
OVLÁDÁNÍ DVOU PŘÍSTROJŮ LOGO JEDNÍM PANELEM HMI

Na jednoduchém příkladu vysvětlíme funkci ovládání. Máme dvě LOGA a jeden panel HMI. Stiskem ovládače "Pravé LOGO ovládá Q1" na panelu LOGO sepneme výstup Q1 levého LOGA s adresou 292.168.0.3 a stiskem ovládače "Levé LOGO ovládá Q2" sepneme výstup Q2 pravého LOGA s adresou 292.168.0.4



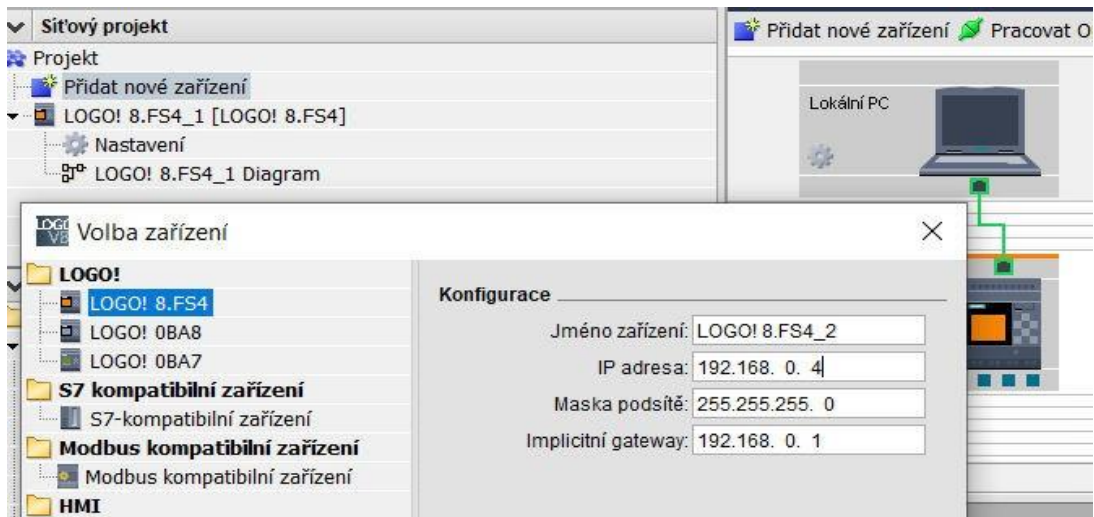
Vytvoření programu v LOGU

1. Klik LT na "Síťový projekt"
2. Klik LT na "Přidat nové zařízení"

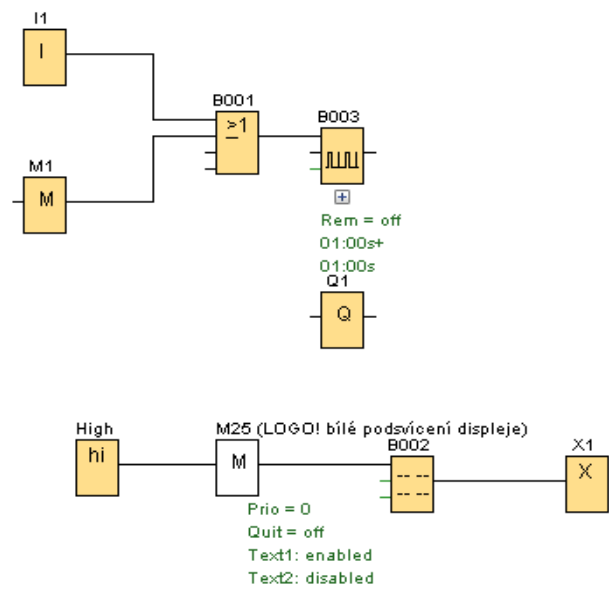
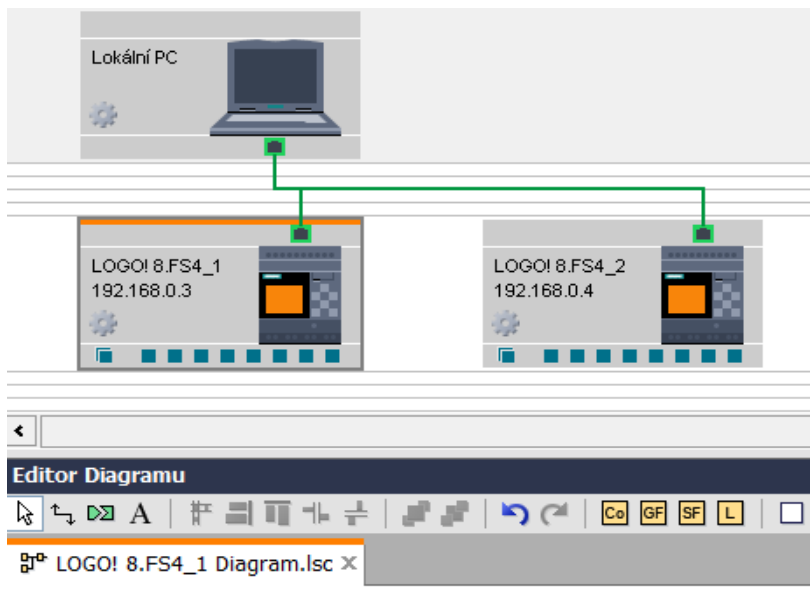


3. Zapiše se "IP adresa" (v každém LOGU musí být jiná)
4. Klik na "OK"

5. Stejným způsobem přidáme další LOGO.

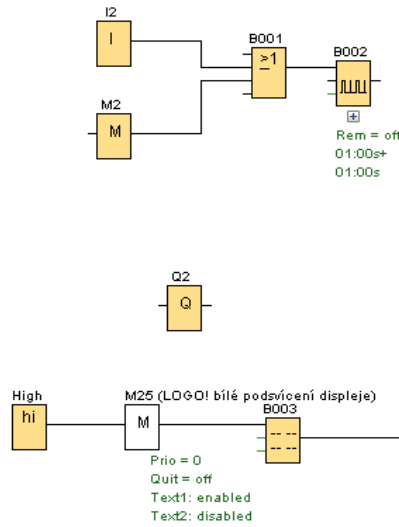


6. Uděláme program pro první LOGO



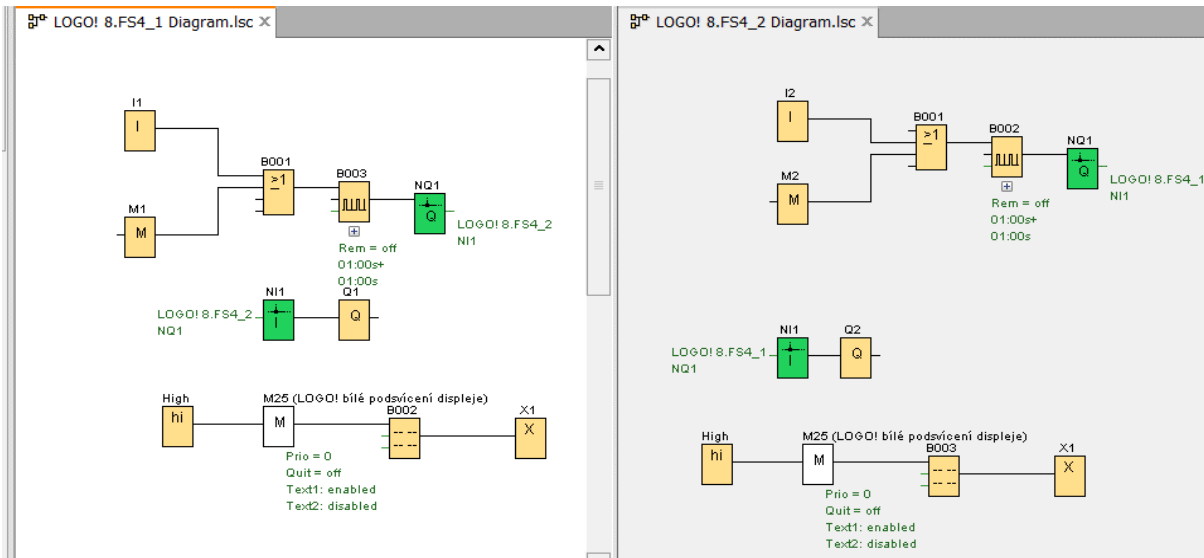
7. Uděláme program pro druhé LOGO (v tomto příkladu uděláme v obou Logách stejný)

LOGO! 8.FS4_2 Diagram.lsc X

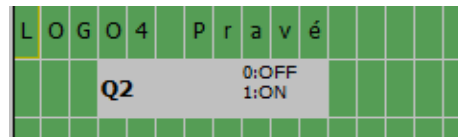
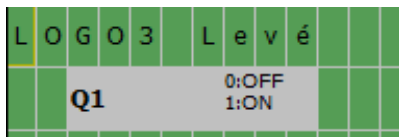


8. Kliknutím LT na rozdělovací ikonu  se zobrazí oba programy odděleně

9. Propojíme vstupy s výstupy na obou stranách



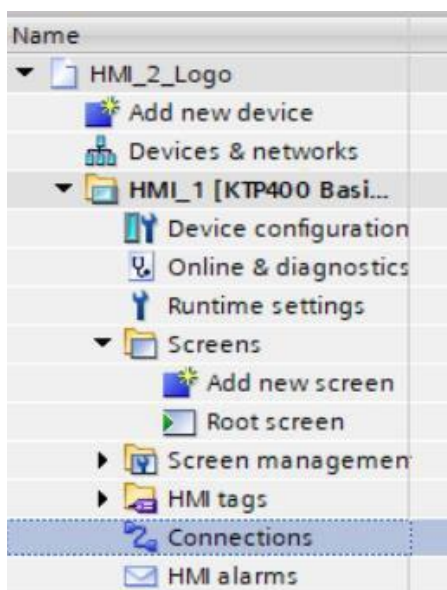
Texty zpráv na displeji



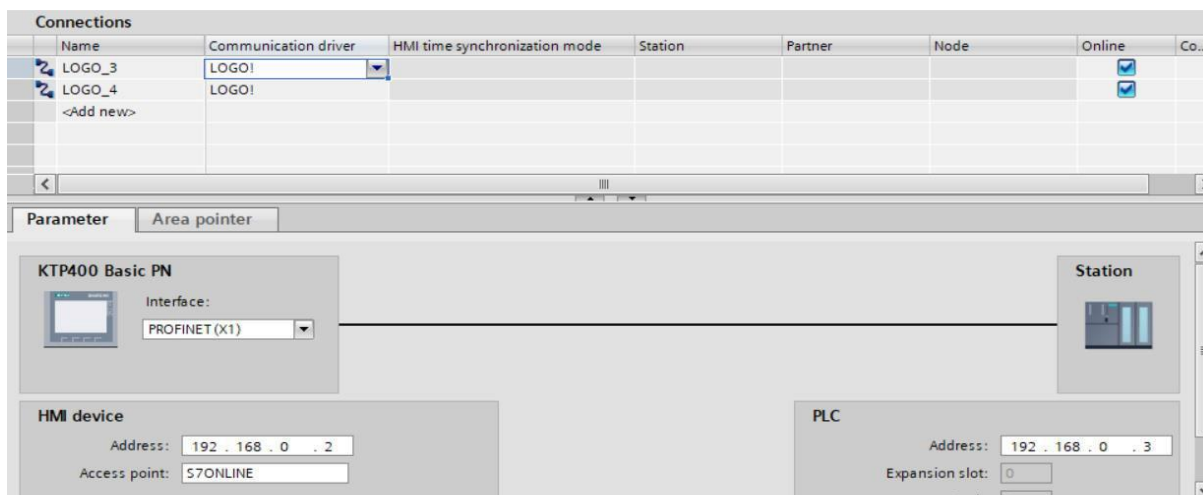
Nastavení panelu HMI

Po založení nového programu provedeme nastavení.

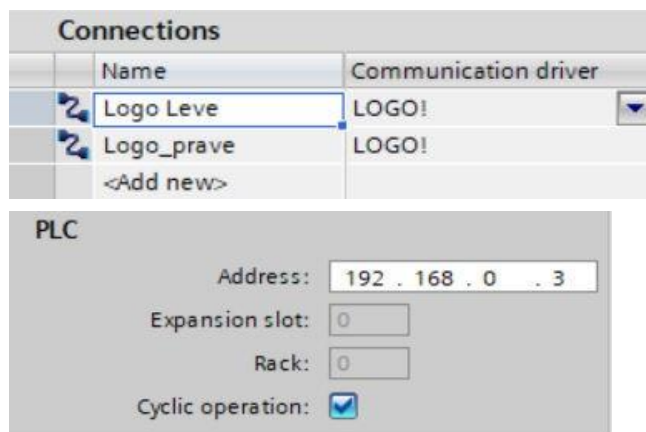
1. Klik na Connections



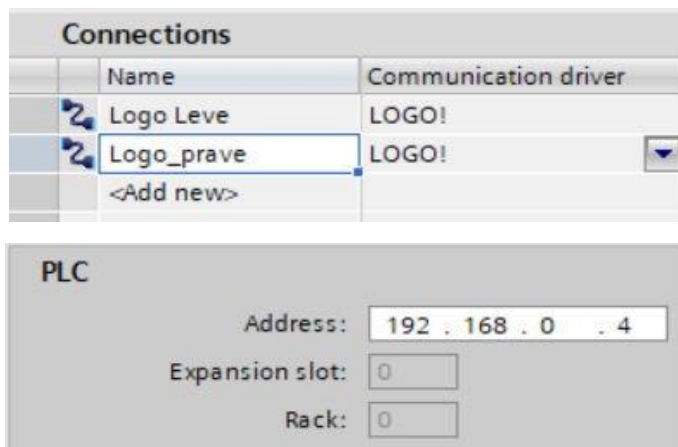
2. Přidáme 1. LOGO. Do sloupce "Name" zapíšeme zvolený název a ve sloupci "Communication driver" vybereme LOGO!.
3. Ve spodní tabulce zapíšeme adresu příslušného LOGA.
4. Přidáme další řádek "Add new" a zapíšeme druhé LOGO



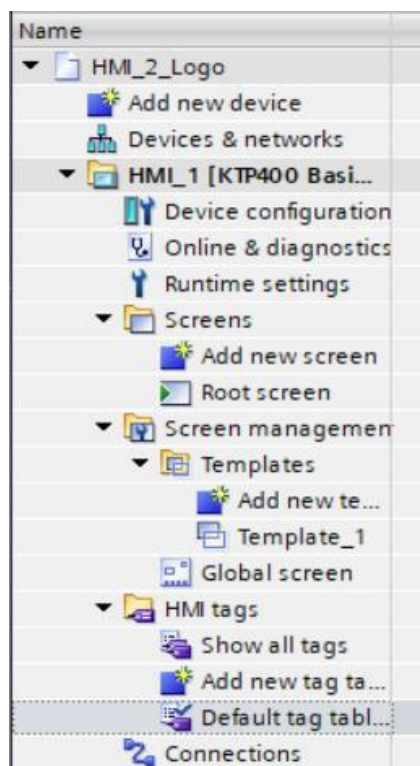
Detail zápisu levého LOGA



Detail zápisu pravého LOGA



5. Nastavíme "Default tag table" (Nastavení výchozí tabulky proměnných)

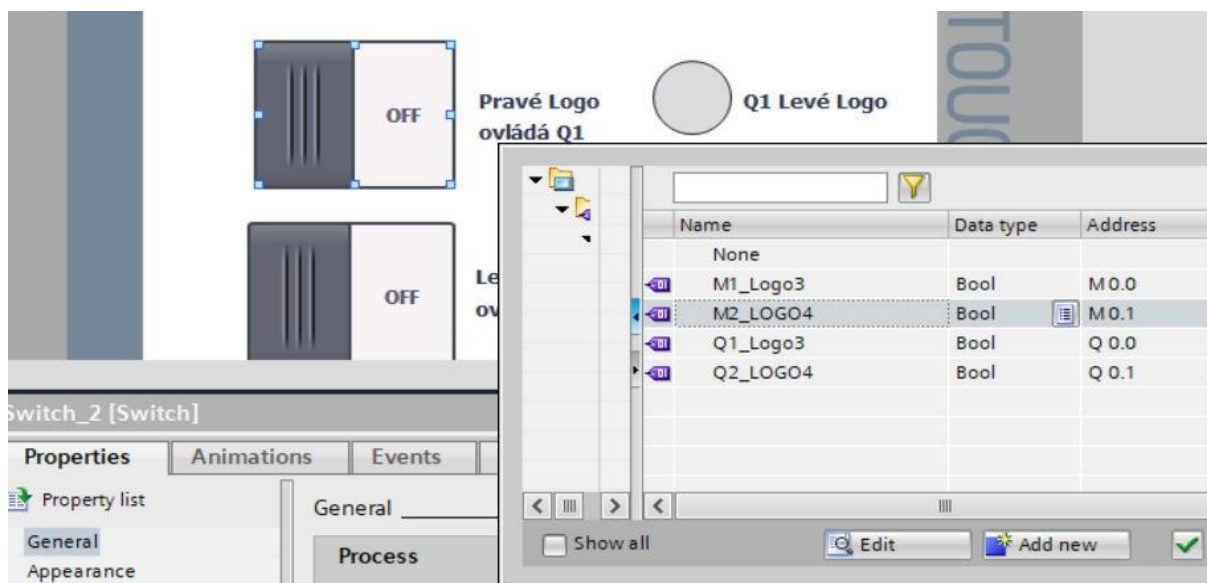


6. Do tabulky vložíme názvy, typy dat, connection, adresy. Důležitý je výběr "Connection", tím rozlišíme LOGO, ze kterého ovládáme další LOGO . Pokud použijeme jiné bloky, než jsou vstupy a výstupy musíme provést nastavení v jednotlivých LOGách v "Parametr VM Mapování". V tomto jednoduchém příkladu máme jen vstup a výstup v každém LOGU.

Default tag table						
Name ▲	Data type	Connection	PLC name	PLC tag	Address	
M1_Logo3	Bool	LOGO_3		<Un...>	M 0.0	
M2_LOGO4	Bool	LOGO_4		<Undef...>	M 0.1	
Q1_Logo3	Bool	LOGO_3		<Undef...>	Q 0.0	
Q2_LOGO4	Bool	LOGO_4		<Undef...>	Q 0.1	
<Add new>						

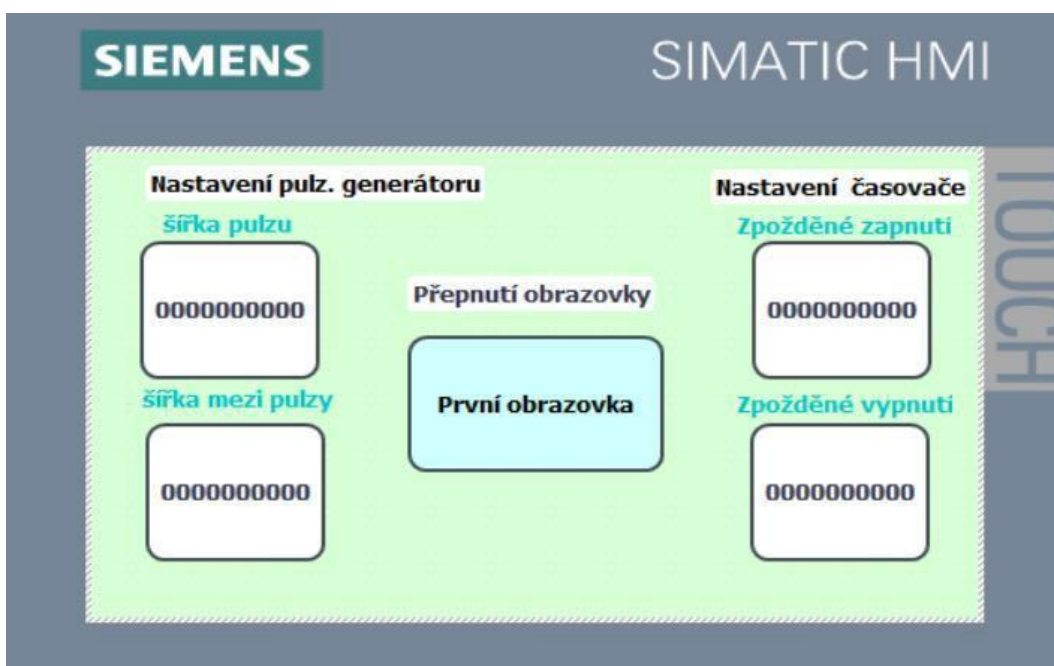
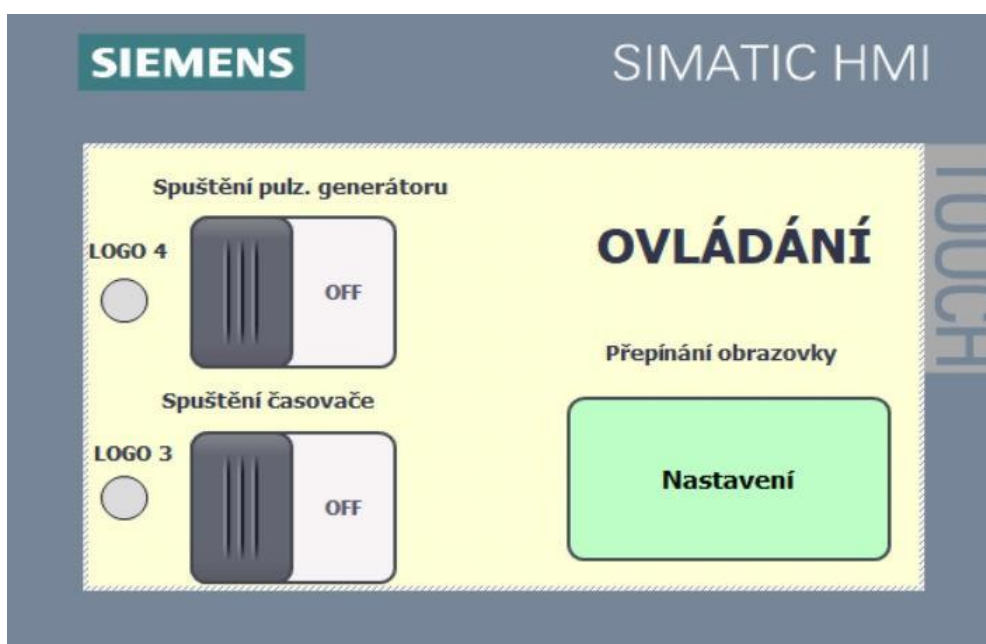
Naprogramování panelu HMI

Na programovací plochu vložíme dvě ovládací tlačítka (M1, M2) a nastavíme jejich adresy. To samé provedeme pro signalizační grafické prvky (Q1, Q2).



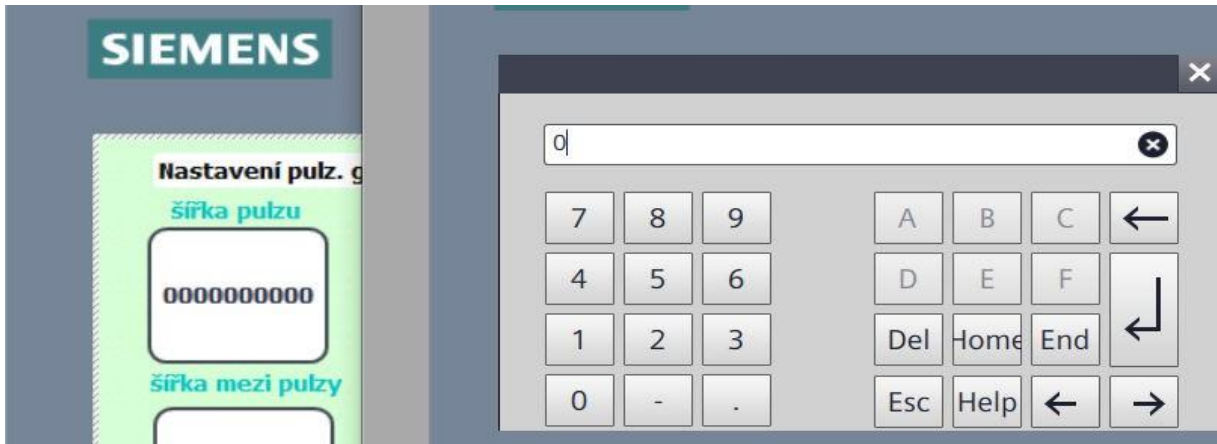
Změna parametrů na panelu HMI při ovládání dvou přístrojů LOGO

Zde jsou dvě loga navzájem propojena datovým kabelem, obdobně jako u předchozího příkladu LOGEM s adresou 292.168.0.3 (levé) se spouští výstup "Q1" v LOGU s adresou 292.168.0.4 (pravé) a naopak LOGEM s adresou 292.168.0.4 se spouští výstup "Q1" LOGA s adr. 292.168.0.3 (levé). Před výstupem z LOGA 3 je v programu zařazen blok "B001" asynchronní generátor pulzů u něho je možné na panelu HMI měnit šířku pulzů a šířku mezi pulzy. V programu LOGO 4 je na výstupu zařazen blok "B005" Zpožděné zap/vyp. U něho lze měnit dobu zapnutí a vypnutí. Stiskem spínače "Spuštění pulz. generátoru" na panelu HMI se začne spínat a rozpínat výstup "Q1" na LOGU 4 (pravé). Stiskem spínače "Spuštění časovače" na panelu HMI se po časové prodlevě sepne výstup "Q1" na LOGU 3 (levé). Opětovným stiskem se po nastavené prodlevě výstup vypne. Doba zpožděného zapnutí a vypnutí je možné nastavit na panelu HMI.



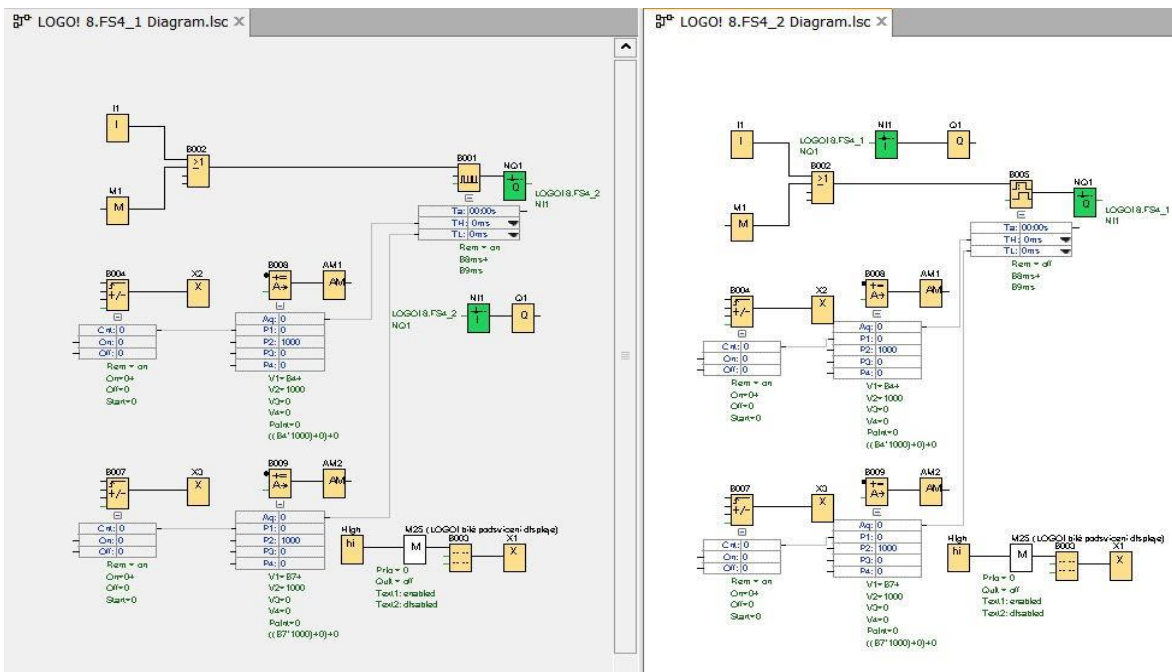
Spouštění výstupů se provádí na první obrazovce, změna parametrů na druhé obrazovce. Přepínání z první na druhou se děje stiskem tlačítka "Nastavení" a přepnutí s druhé obrazovky na první se děje stiskem tlačítka "První obrazovka".

Kliknutím do okna "šířka pulzu" se otevře klávesnice a stiskem číslic zvolíme požadovanou dobu šířky pulzu. Stejný postup zvolíme pro nastavení ostatních parametrů.



Program LOGO

Postupem uvedeným v předchozím příkladu vytvoříme program v jednotlivých LOGách a jejich vzájemné propojení.



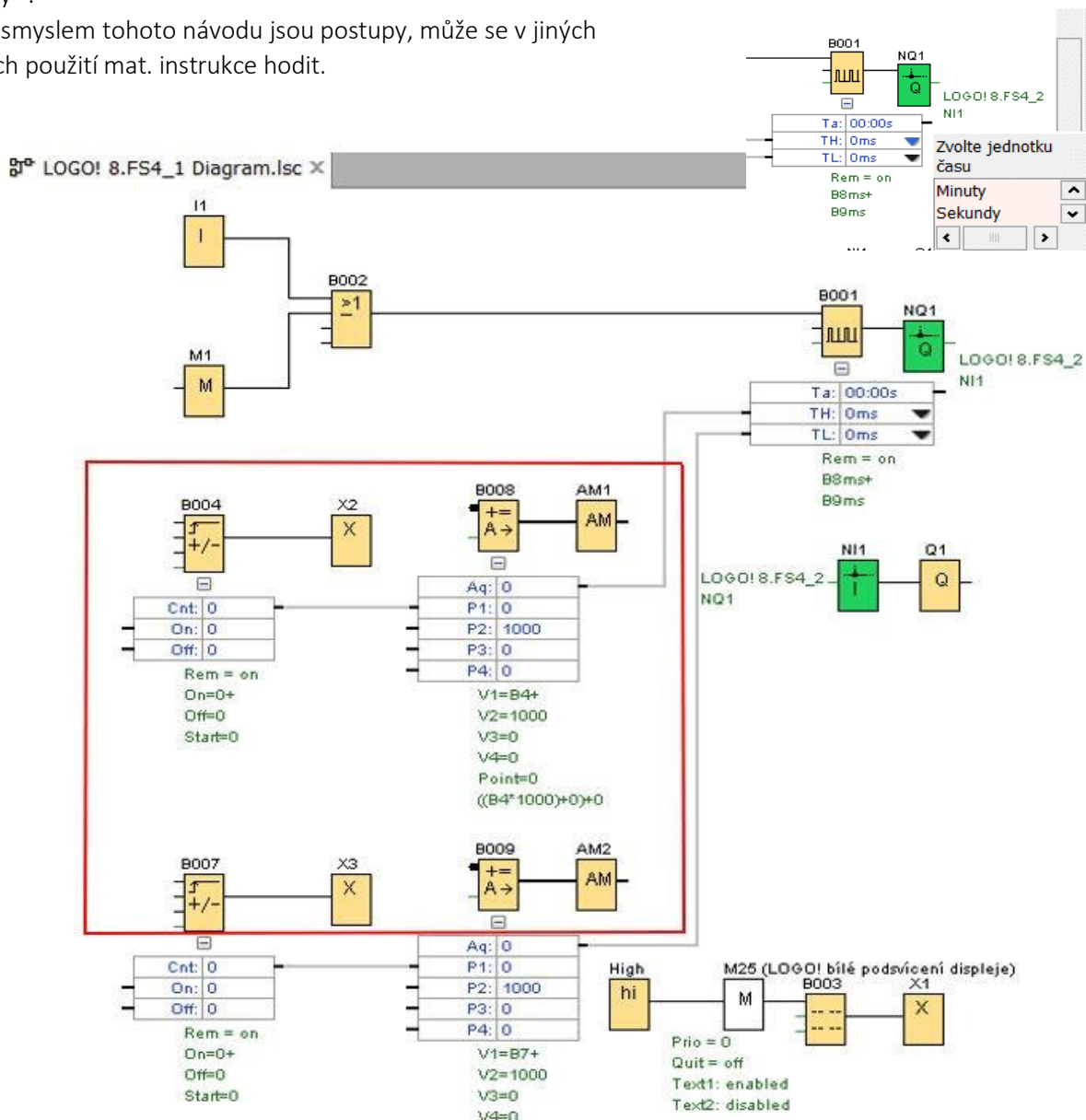
Program v LOGU

Program zapsaný v LOGU s adresou 292.168.0.3 (levé)

I1 je vstup, ovládaný mechanickým spínačem. M1 je příznak, pomocí něhož se spíná vstup z panelu HMI. Sepnutím jednoho ze vstupů se spustí pulzní generátor „B001“ a střídavě bude spínat a rozepínat síťový výstup NQ1 a tím i výstup "Q1" v LOGU s adresou 292.168.0.4.

V červeném rámečku jsou bloky pro změnu parametrů generátoru pulzů. V panelu HMI je možné měnit parametry u bloků s datovým typem Dword a ten mají čítače. Blokem čítače "B004" měníme hodnotu pulzního generátoru TH - šířka pulzu. Blokem "B007" se mění hodnota pulz. generátoru TL - šířka mez i pulzy. Protože se hodnota v pulzním generátoru zobrazuje v milisekundách, a my ji chceme v sekundách, tak hodnotu z čítače vynásobíme 1000 krát. To se provede blokem "B008" matematické instrukce pro parametr TH a blokem "B009" pro parametr TL asynchronního generátoru. V tomto případě by bylo jednodušší vynechat bloky matematických instrukcí kliknutím na rozbalovací šipku generátoru pulzů zvolit "Sekundy".

Protože smyslem tohoto návodu jsou postupy, může se v jiných případech použití mat. instrukce hodit.



Program v LOGU 292.168.0.4 je shodný, jenom místo pulzního generátoru je tam zpožděně zap/vyp.

V LOGU 292.168.0.3 nastavíme v okně "Nástroje" -"Parametr VM Mapování " adresy použitých bloků.

ID	Blok	Parametr	Typ	Adresa
1	B004 [Dopředný a zpětný čítač]	Čítač	DWord	0
2	B007 [Dopředný a zpětný čítač]	Čítač	DWord	4
3				

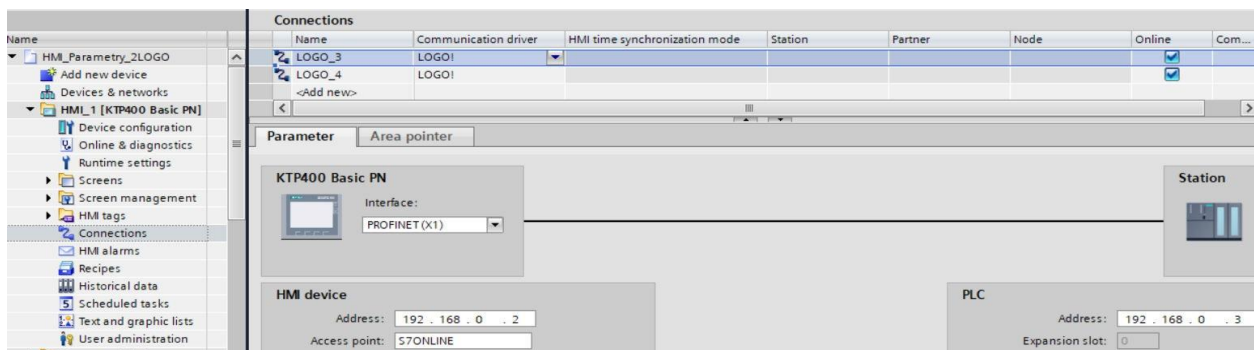
To samé uděláme i pro druhé LOGO s adresou 292.168.0.4

ID	Blok	Parametr	Typ	Adresa
1	B004 [Dopředný a zpětný čítač]	Čítač	DWord	0
2	B007 [Dopředný a zpětný čítač]	Čítač	DWord	4
3				

Protože se jedná o dva různé přístroje adresy jsou shodné.

Nastavení v panelu HMI

- Nastaví se v "Connection" adresy pro obě LOGA. Při zadání prvního LOGA nesmíme zapomenout přepsat adresu, na tu v LOGU, zde je to 192.168.0.3. Kliknutím a "Add" se přidá druhé LOGO s adresou 192.168.0.4.

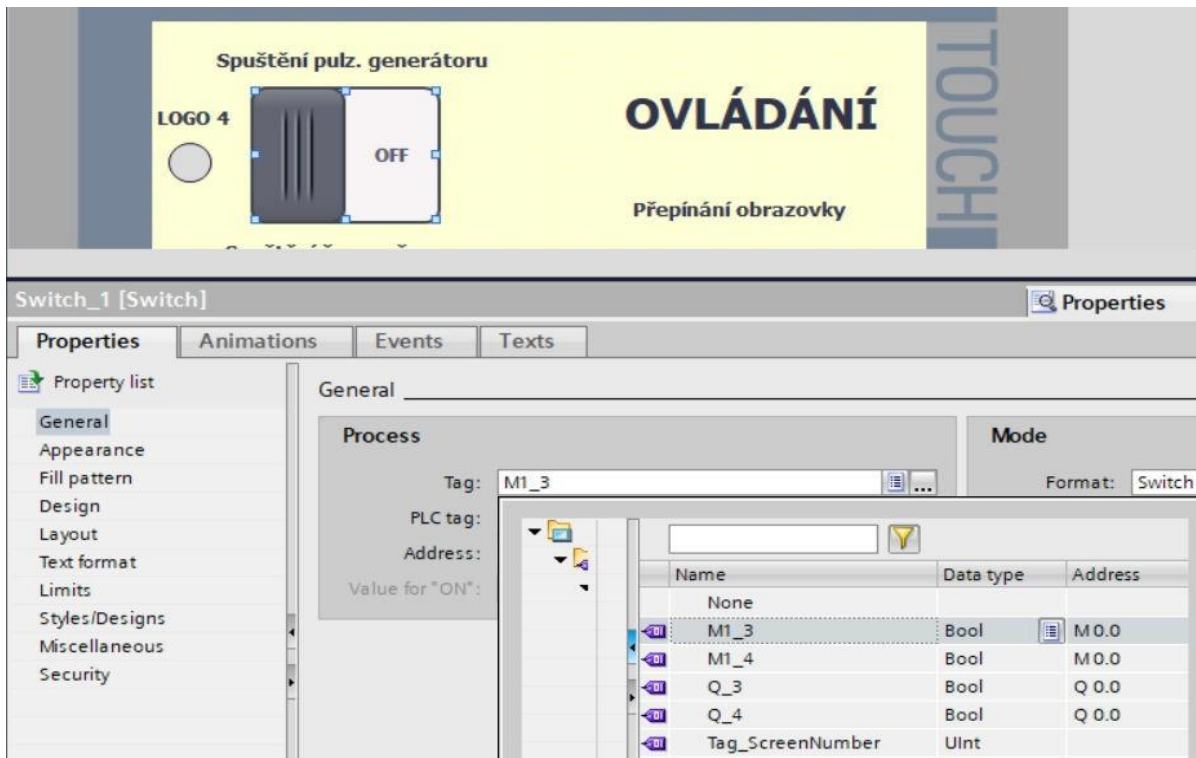


- Vyplní se tabulka "Default tag table"

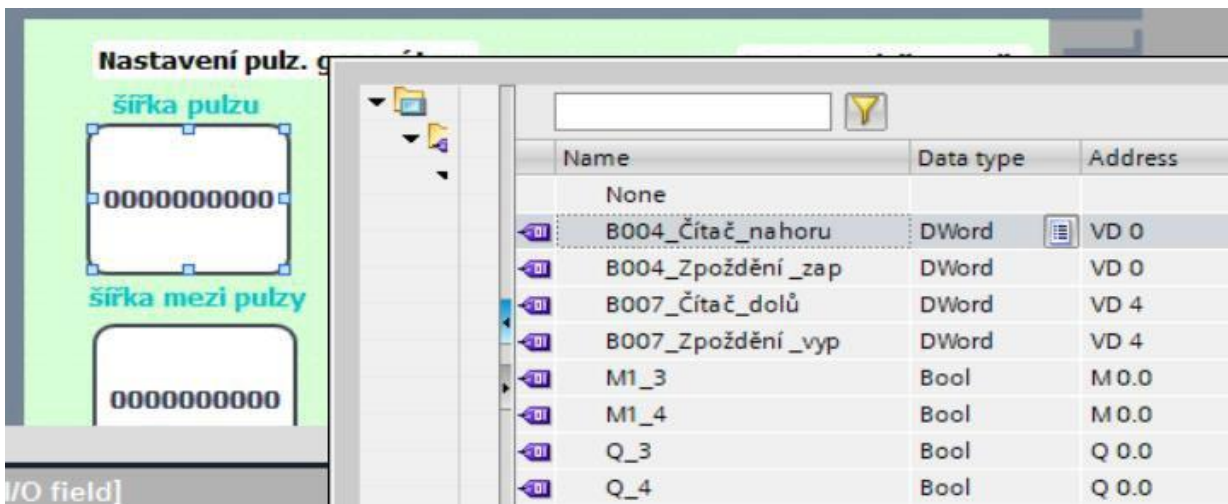
Pozor se musí dávat na správný výběr "Connection" . (Logo_3, Logo4)

Name	Name	Data type	Connection	PLC name	PLC tag	Address
	B004_Čítač_nahoru	DWord	LOGO_3		<Undefined>	VD 0
	B004_Zpoždění_zap	DWord	LOGO_4		<Undefined>	VD 0
	B007_Čítač_dolů	DWord	LOGO_3		<Undefined>	VD 4
	B007_Zpoždění_vyp	DWord	LOGO_4		<Undefined>	VD 4
	M1_3	Bool	LOGO_3		<Undefined>	M 0.0
	M1_4	Bool	LOGO_4		<Undefined>	M 0.0
	Q_3	Bool	LOGO_3		<Undefined>	Q 0.0
	Q_4	Bool	LOGO_4		<Undefined>	Q 0.0
	Tag_ScreenNumber	UInt	<Internal tag>		<Undefined>	
	<Add new>					

Při adresování jednotlivých bloků na programovací ploše parametru HMI musíme dbát na správné přiřazení adresy.



Přiřazení adresy na druhé obrazovce.



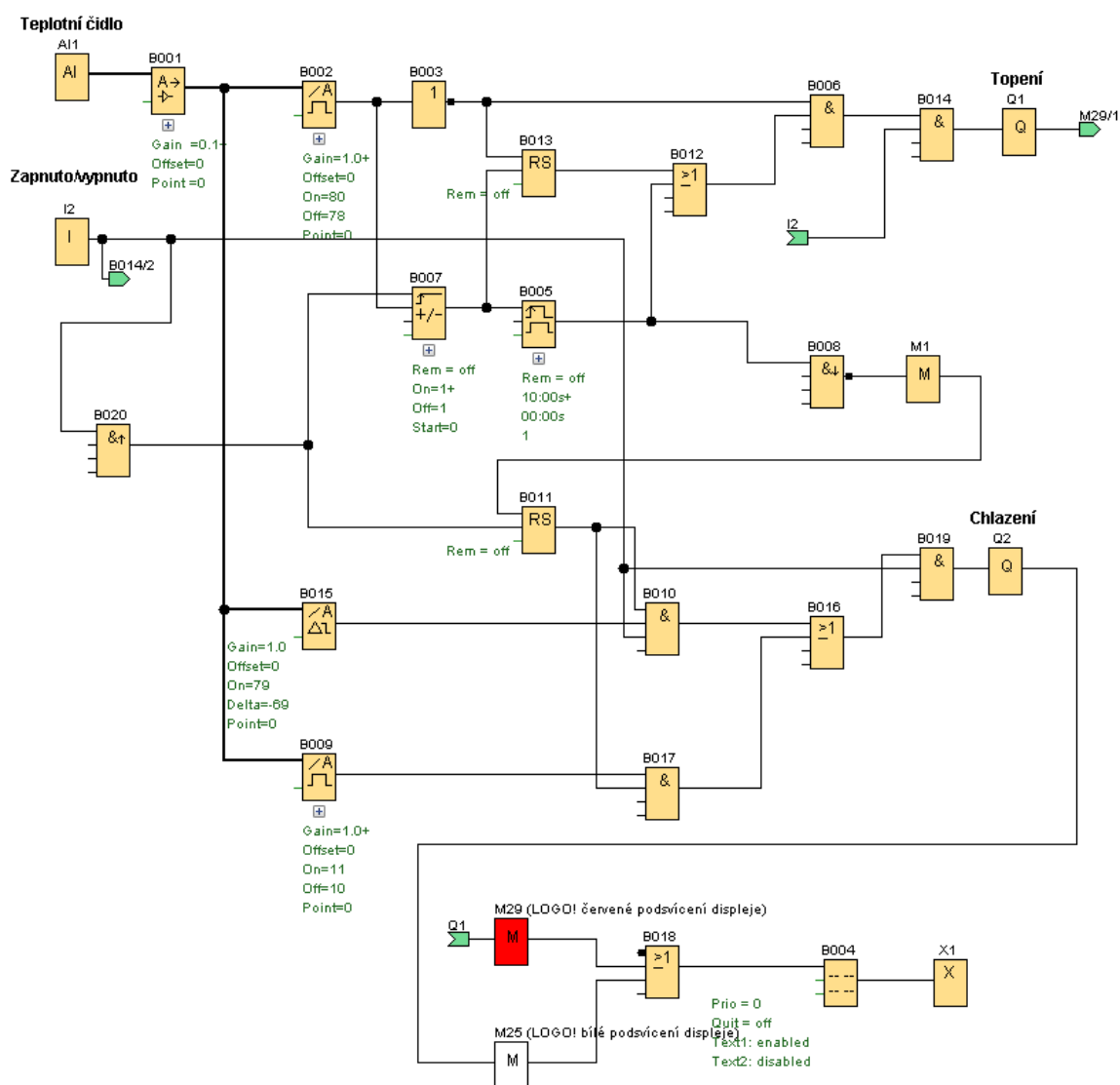
Pasterizace

Podstatou pasterizace je krátkodobé zvýšení teploty, které způsobí zničení nesporeulujících patogenních mikroorganismů. Používá např. pro pasteraci mléka, piva, moštu.

Zde je zvolen ohřev v nádobě pasteru na teplotu v rozmezí 78–80 °C, ponechání na této teplotě po dobu 10 s a po uběhnutí 10 s se sepne chlazení. Z technického pohledu se jedná o kombinaci topení a chlazení. Chlazení sníží teplotu produktu 10 °C a udržuje ji mezi 10–12°C. Na jakou teplotu a po jakou dobu provádět ohřev musí zadat technolog výroby.

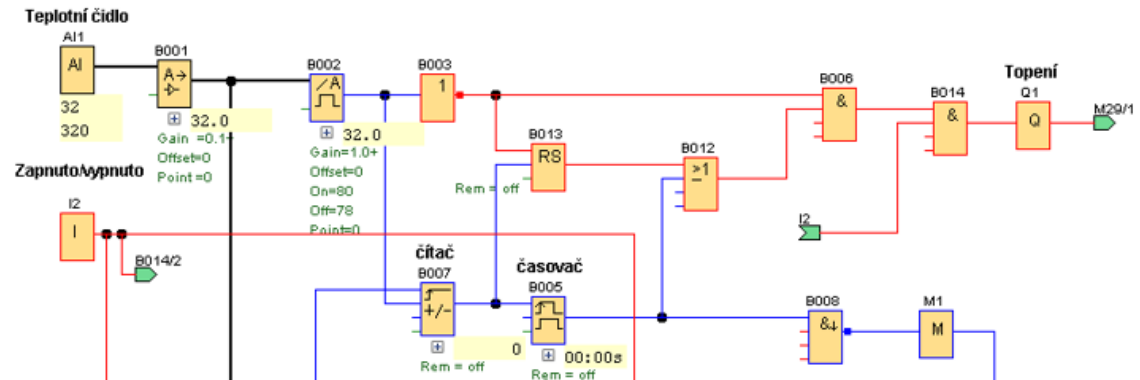
Zapnutí se provede spínačem „I1“ Zapnuto/vypnuto. Zapnutím se uvede do činnosti topení – výstup „Q1“. Po překročení teploty 80 °C se sepne časování. To trvá nastavenou dobu 10 s. Teplota se po tuto dobu udržuje v rozmezí 78 až 80°C. Po ukončení časování se zapne chlazení „Q2“. Chlazení zůstává v chodu do poklesu teploty 10 °C, zvýší-li se teplota nad 12 °C chlazení se opět zapne.

Program pasterizace

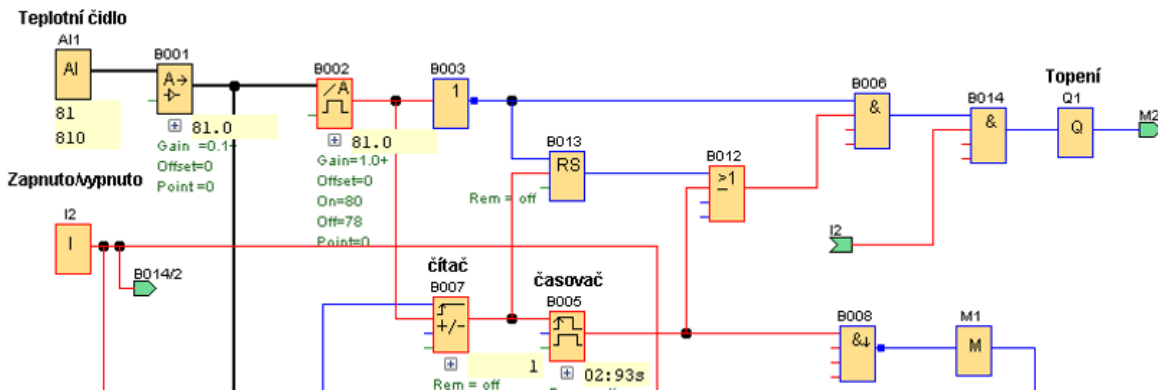


Stav teploty, topení, chlazení je zobrazován na displeji LOGA. Pokud dochází k ohřevu je displej podbarvený červeně, v případě chlazení je podbarvený bíle a pokud je topení nebo chlazení vypnuto po dosažení nastavené hodnoty je podbarven zeleně.

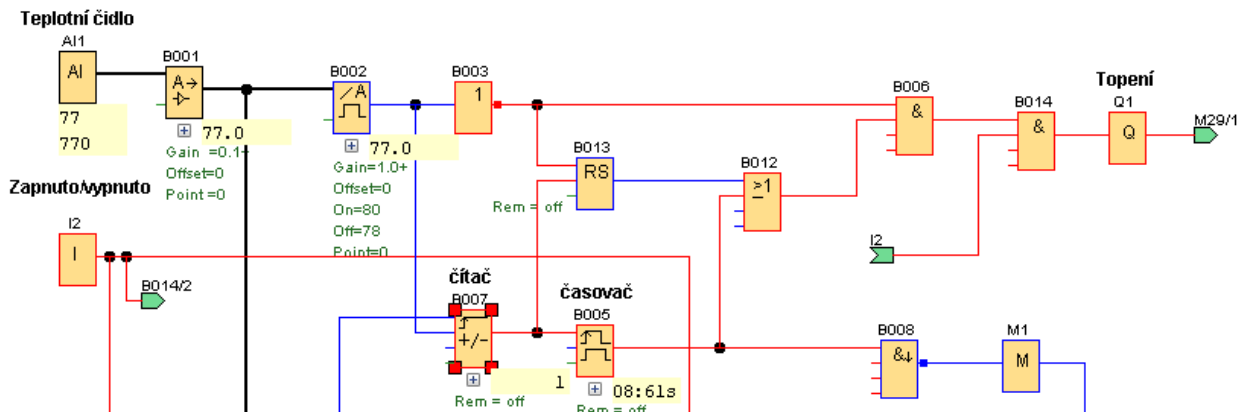
Stav po zapnutí, kdy teplota je nižší než 80°C. Je sepnutý blok AND „B014“, a to spínačem „I2“ a sepnutým výstupem bloku AND „B006“. Blok AND „B006“ je sepnutý blokem negace „B003“ a blokem RS „B013“, jeho výstup se spíná společně se sepnutím výstupu bloku negace. Čítač „B007“ má hodnotu „0“.



Po dosažení teploty 80 °C se sepne výstup analogového spínače „B002“, dojde k navýšení hodnoty čítače na „1“ a sepne se jeho výstup. Tím dojde k vypnutí bloku RS „B013“ a sepne se výstup časovače „B005“.

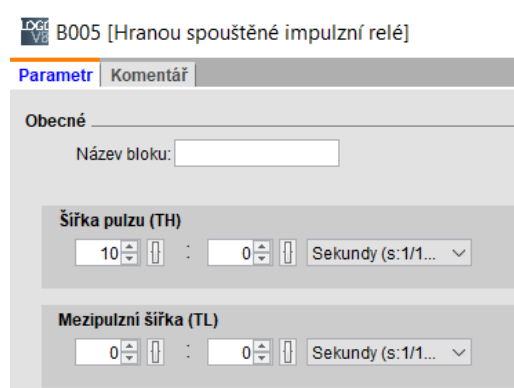
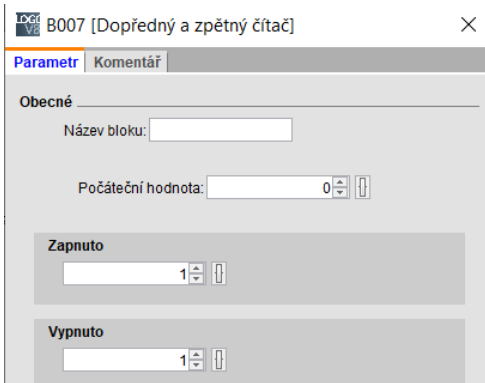


Při snížení teploty na 78 °C se výstup bloku negace „B003“ opět sepne a tím se sepne i topení „Q1“. Střídavé zapínání a vypínání topení „Q1“ platí podobu časování bloku „B005“.

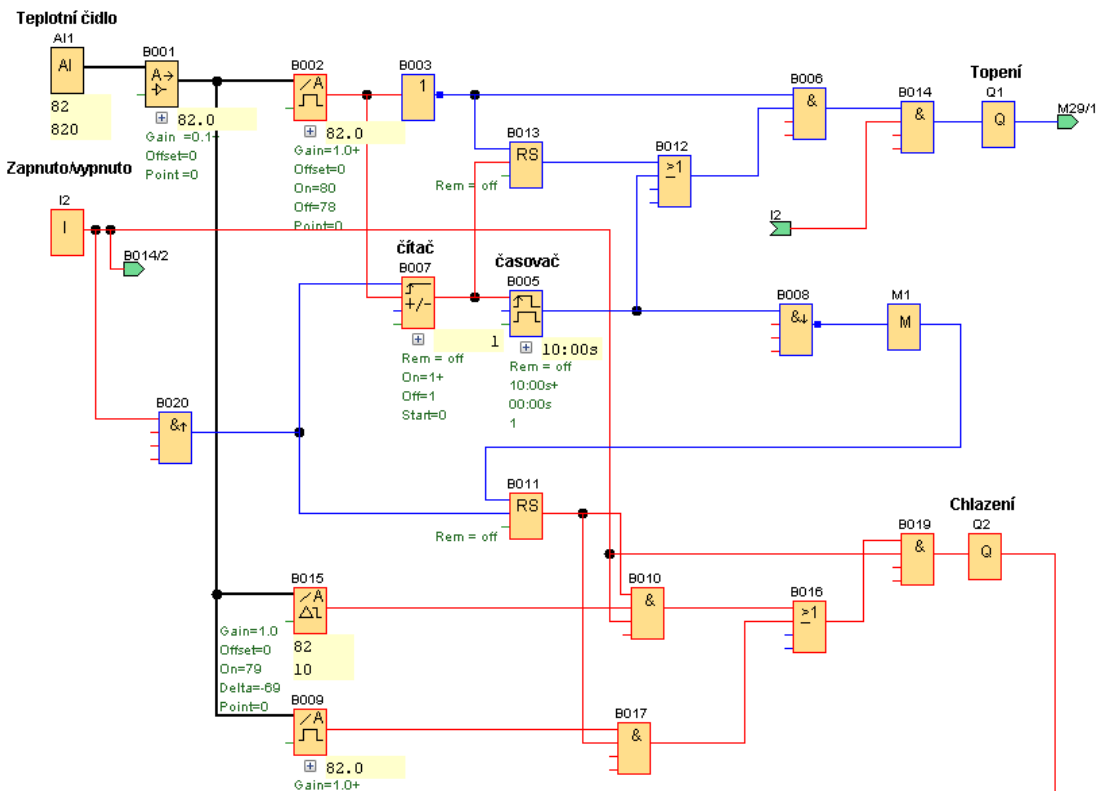


Nastavení čítače

Nastavení časovače



Po ukončení časování dojde trvale k vypnutí Topení Q1 (výstupy bloků AND „B006“ a „B014“ jsou vypnuty). Vypnutím časovače „B005“ dojde impulzem NAND (hrana „B008“ k sepnutí relé RS „B011“ a tím i sepnutí výstupu „Q2“ Chlazení.



Regulace chlazení je provedena ve dvou krocích. Analogový spínač „B015“ sepne chlazení při 79 °C a vypne ho při 10°C. Protože chceme udržovat teplotu v rozmezí 10 až 12 °C je zde druhý Analogový spínač „B009“, a ten sepne chlazení při překročení teploty 12 °C a udržuje teplotu v rozmezí 10–12°C. Pokud by byl jen jeden analogový spínač „B015“, tak by sepnul až při oteplení až na 79°C.

Zapnuto

79

Vypnuto

10

Zapnuto

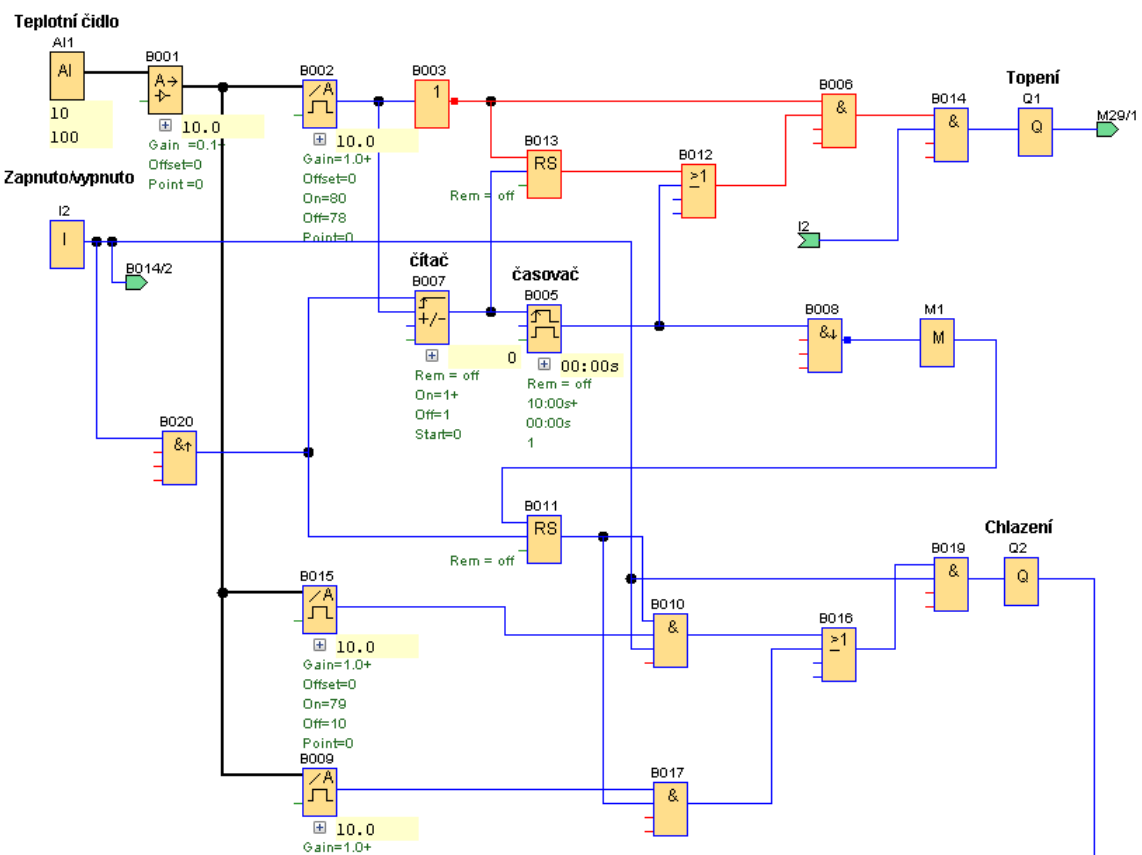
12

Vypnuto

10

Vypnutí pasterizace

Vypnutím spínače se vypne topení i chlazení a vynuluje se čítač.



Vypnutím spínače „I2“ se vypne relé RS B011, vstup bloku AND „B019“ a AND „B010“ a tím chlazení a dále dojde k vypnutí vstupu bloku AND „B014“, tím se vypne topení.

Čítač „B007“ spíná při hodnotě 1, jeho sepnutím při dosažení teploty 81 °C se vypne výstup bloku RS „B0013“ a současně se sepne časovač „B005“. Pokud by v tomto zapojení čítač nebyl nedošlo by k omezení doby ohřevu časovačem „B005“.

Ovládání jímky a zavlažování pomocí LOGO! a Node-Red

Na zahradě je vybudována jímka pro shromažďování vody. Zde je nainstalován analogový senzor hladiny vody a motor/pumpa. Z jímky se bude přečerpávat voda do druhé shromažďovací jámy, kde bude umístěno další čerpadlo, které nám vody přímo přečerpá na zahradu, a budeme s ní moci zavlažovat. Vše to bude moci být ovládáno fyzicky přes tlačítka, nebo přes webový server Node-Red, který budeme mít nainstalovaný na lokálním Raspberry Pi. Pumpa na zavlažování se bude automaticky zapínat při východu a západu Slunce na 20 minut. Také se automaticky bude hlídat hladina vody, aby nedošlo ke zničení pump. Tzn. jakmile klesne voda v jímce pod 5%, zablokuje se pumpa a jakmile bude nad 95% začne se voda automaticky přepouštět do shromažďovací jámy. Toto uživatel může v mezních hodnotách zapnout i vypnout sám. Jamile bude shromažďovací jáma na 5% kapacity vody, pumpa se zablokuje, a pokud hladina vody vystoupá nad 95% spustí se zavlažování, aby nedošlo k přetokům a případnému zničení zařízení.

Nastavení Raspberry a Node-Red

Budeme potřebovat Raspberry Pi, které je schopné vytvořit server, naše LOGO! Schopné S7 komunikace (v příkladu je použito LOGO8!) a možnost připojení k domácí síti. Jestliže toto seženeme, tak můžeme stáhnout Raspberry Pi OS. Přes tento program si na mikro SD kartu nahrajeme operační software našeho Raspberry zařízení. <https://www.raspberrypi.com/software/>

Po nainstalování soubor otevřeme a pomocí průvodce OS na mikro SD kartu nainstalujeme.

Install Raspberry Pi OS using Raspberry Pi Imager


Raspberry Pi Imager is the quick and easy way to install Raspberry Pi OS and other operating systems to a microSD card, ready to use with your Raspberry Pi. [Watch our 45-second video](#) to learn how to install an operating system using Raspberry Pi Imager.

Download and install Raspberry Pi Imager to a computer with an SD card reader. Put the SD card you'll use with your Raspberry Pi into the reader and run Raspberry Pi Imager.

Download for Windows

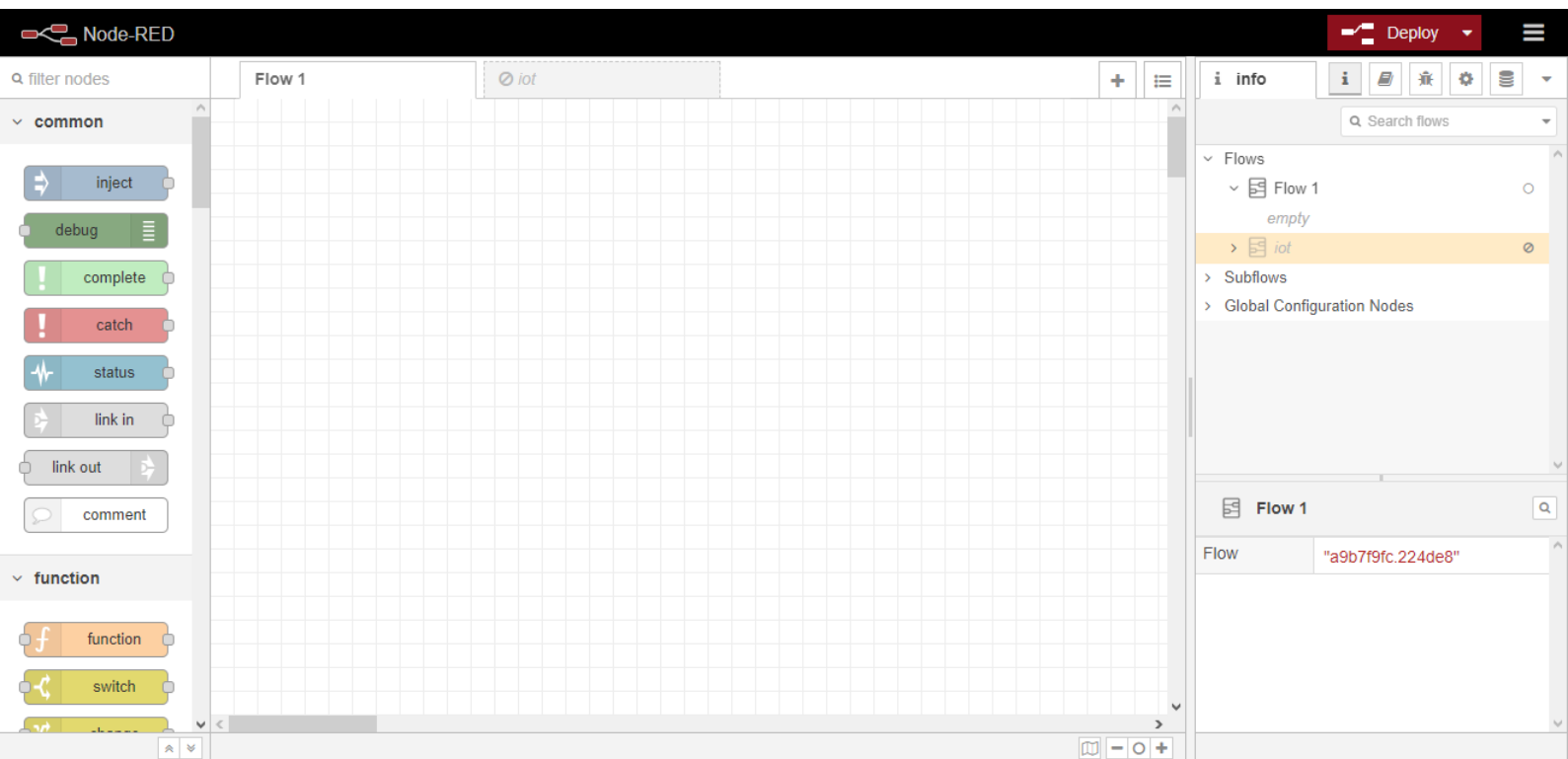
[Download for macOS](#)

[Download for Ubuntu for x86](#)



Jakmile zapojíme Raspberry do napájení a vše se nám nabojuje, tak si otevřeme prohlížeč na Raspberry. POZOR, pokud prohlížeč nebude fungovat je možné, že nejste připojeni k internetu, proto zkontrolujte připojení k Wi-Fi síti. Jakmile se nám prohlížeč otevře, můžeme tam zadat tuto adresu k nainstalování Node-Red: <https://nodered.org/docs/getting-started/raspberrypi>. Zde je průvodce a vše co je potřeba k instalaci.

Poté co nainstalujeme a spustíme Node-Red to bude vypadat asi takto. (bez IoT)



Abychom museli pokračovat dál, tak musíme stáhnout doplňky do Node-Redu. Musíme stáhnout:

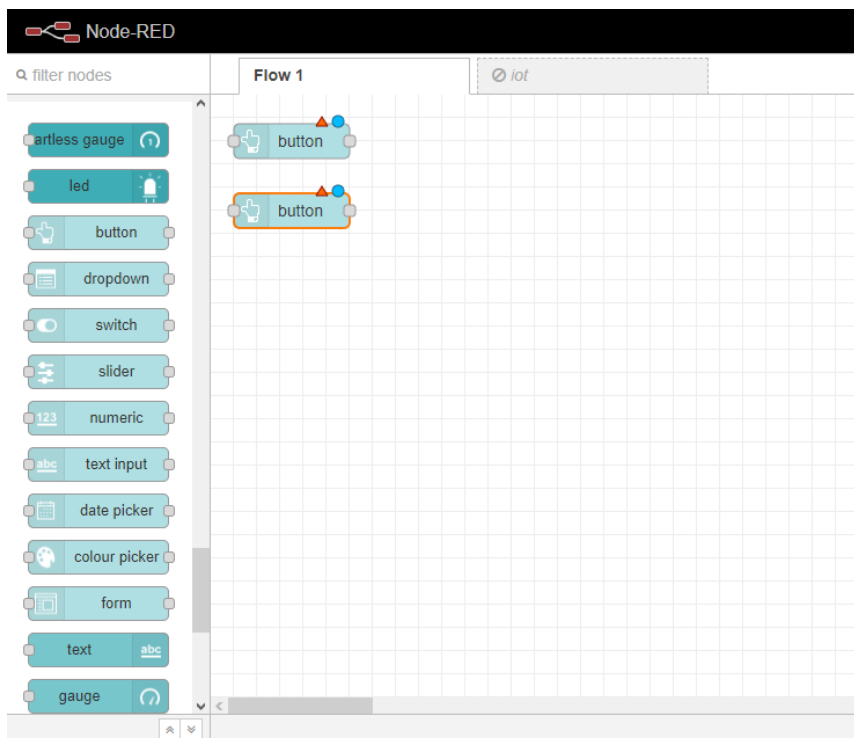
Node-red-dashboard <https://flows.nodered.org/node/node-red-dashboard>

Node-red-contrib-s7 <https://flows.nodered.org/node/node-red-contrib-s7>

Cokoliv dalšího, co budete potřebovat pro vylepšení dashboard.

Jakmile si rozkliknete ten odkaz u node red contrib s7, tak je tam kompletní návod, jak to funguje a jak co instalovat.

Nejdříve si přidáme tlačítka na zapnutí a vypnutí jímky. Ty najdeme po záložkou dashboard vlevo



Následně tyto tlačítka nastavíme. Group nemusíme měnit, přejmenujeme si název tlačítka na vypnuto nebo zapnuto a msg.payload přejmenujeme na false, když je to stav vypnuto a true, když je to stav zapnuto.

Poté musíme za to dát nod change. Ten nastavíme, aby nám po stisku tlačítka odešla zpráva do zařízení typ Boolean True. Zase true, pokud je to zapnout nebo false pokud je to vypnout.

The image shows two screenshots of the Node-RED interface. The top screenshot is the 'Edit change node' dialog box. It features a title bar with 'Edit change node' and three buttons: 'Delete', 'Cancel', and 'Done'. Below the title bar is a 'Properties' section with a gear icon and three sub-panels: 'Name' (containing 'Name'), 'Rules' (containing a 'Set' node with 'msg.payload' and 'true'), and another 'Properties' section (containing various node settings like Group, Size, Icon, Label, Tooltip, Color, Background, and Payload set to 'false').

The bottom screenshot is a detailed view of the 'Properties' section for a node. It includes a 'Group' dropdown set to '[ledky] leds', 'Size' set to 'auto', 'Icon' set to 'optional icon', 'Label' set to 'vypnuto', 'Tooltip' set to 'optional tooltip', 'Color' set to 'optional text/icon color', and 'Background' set to 'optional background color'. There is a checked checkbox for 'When clicked, send:' with 'Payload' set to 'false' and 'Topic' set to 'msg.topic'. At the bottom, there is an 'Enabled' checkbox.

Nyní musíme nastavit node s7 endpoint. Vybereme tento nod a nastavíme jej takto:

Dialog: Edit s7 out node > Add new s7 endpoint config node

Buttons: Cancel, Add

Properties

Connection Variables

Transport: Ethernet (ISO-on-TCP)

Address: 192.168.0.3 Port: 102

Mode: TSAP

Local TSAP: 02.00 Remote TSAP: 01.00

Cycle time: 1000 ms

Timeout: 2000 ms

Name: LOGO18

Enabled 0 nodes use this config On all flows

Address je IP adresa Vašeho LOGO! a Name je, jak si tento nod chcete pojmenovat. V záložce Variables si nastavíme tyto proměnné. Budeme je pak využívat dále v programu.

Dialog: Edit s7 out node > Edit s7 endpoint node

Buttons: Delete, Cancel, Update

Properties

Connection Variables

Variable list

DB1,X4.0	Q01	X
DB1,X4.1	Q02	X
DB1,X4.2	Q03	X

Q01 je pumpa v jímce a Q02 je pumpa na zahradu a Q03 je tlačítko STOP.

Nakonec si jen zvolíme jméno pro nod a jakou proměnnou chceme měnit.

Edit s7 out node

Delete Cancel Done

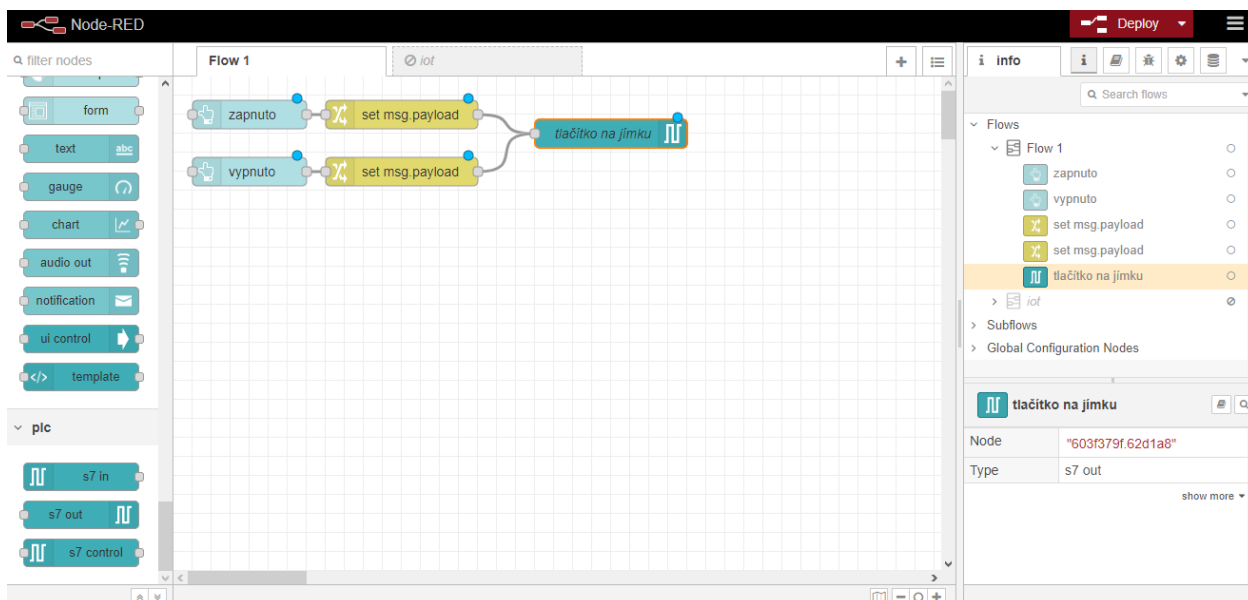
Properties

⚡ PLC LOGO!8

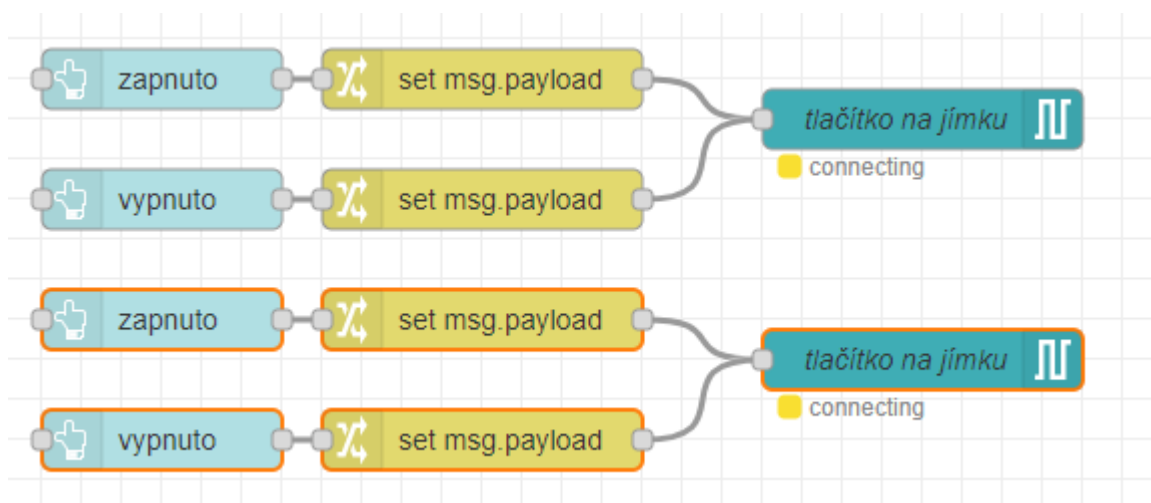
🔗 Variable Q01 DB1,X4.0

📌 Name tlačítko na jímku

Konečný výsledek v této fázi by měl vypadat zhruba takto.



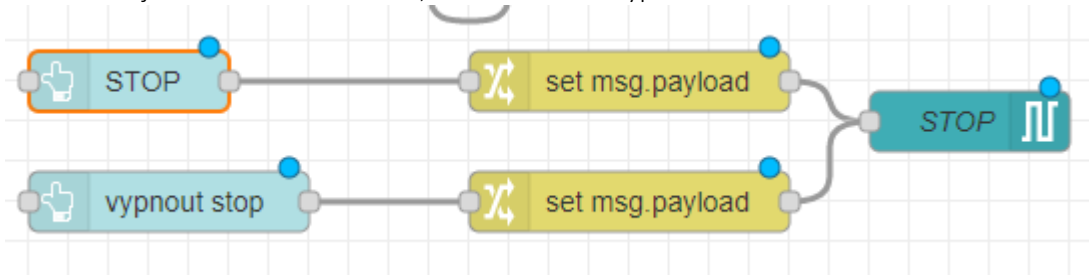
Nyní to zkopírujeme a změníme proměnnou na Q02 a budeme moct měnit obě pumpy.



A když do vyhledávacího řádku zadáme adresu localhost:1880/ui/ tak odsud budeme moci měnit hodnoty zapnuto a vypnuto. Vypadá to nějak takto po lehkých úpravách.



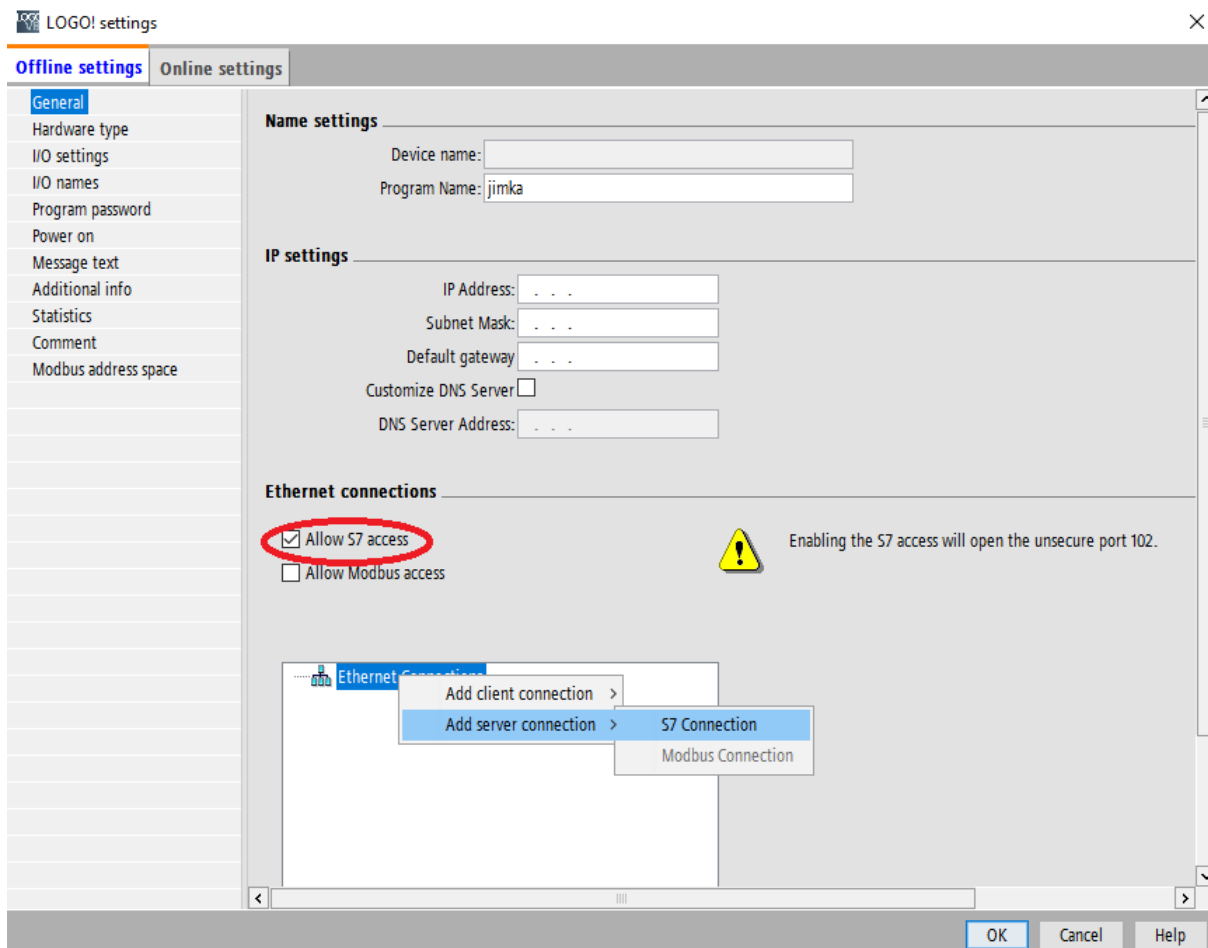
Nyní ještě přidáme funkci STOP. Znovu vezmeme nody, zkopírujeme a v proměnných jim přidáme hodnotu Q03. Takže jakmile stiskneme STOP, tak se nám vše vypne.



Node-Red má spousty možností a je to open source, takže lidé tam přidávají své výtvořky neustále. Stačí tomu věnovat chvíli a budete mít profesionální backend stránku.

Nastavení LOGO! A Logo Soft

Vytvoříme nový program pro ovládání jímky. Zde musíme zaškrtnout povolení S7 připojení. Poté kliknutím pravého tlačítka myši na Ethernet Connections se nám zobrazí možnost připojit server a zde možnost server s připojením S7. Vybereme toto.

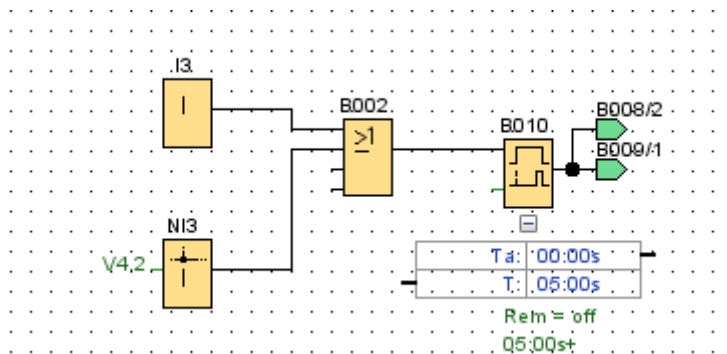


Po předchozím kroku se nám zobrazí něco jako connection1. Na to dvojklikneme a zobrazí se nám možnosti. Zde zaškrtneme obě zaškrťovací pole a do Remote Properties (Client) napíšeme 01:00. Mělo by to vypadat takto:

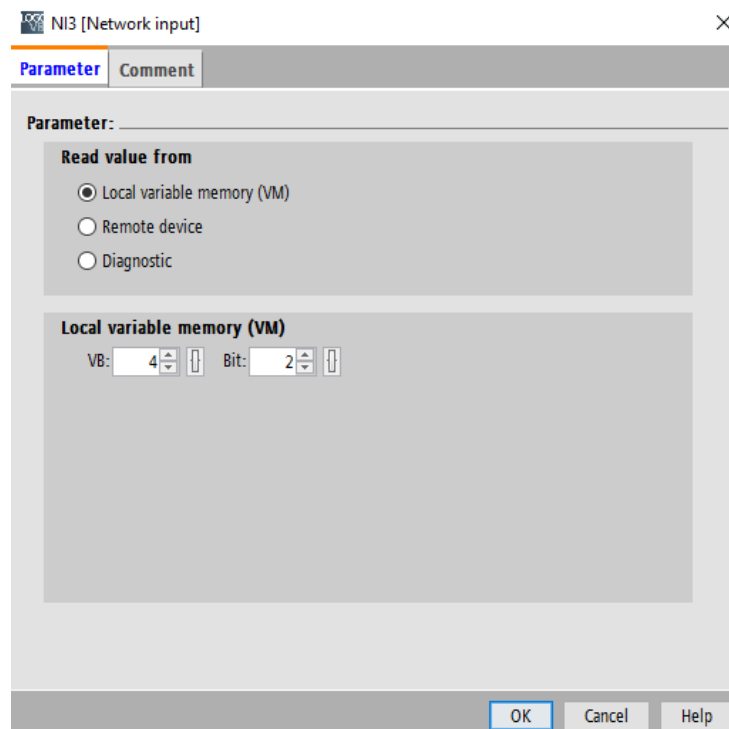
v jímce. Tudíž se to nespustí pokud v jímce nebude dostatečně velké množství vody. Z AND nám to jde ještě do jednoho AND. To nám kontroluje zda není ve shromažďovací jámě náhodou příliš moc vody, aby nám nedošlo k přetečení zase. Pokud jsou všechny parametry splněny a nikde není chyba, tak se nám spustí pumpa na vodu z jímky

Tlačítko STOP.

Toto je fyzické tlačítko, které jakmile se stiskne nám po 5 sekundách zablokuje jakékoliv probíhající funkce. Toto tlačítko také budeme mít na serveru pro jistotu. Výstup z tlačítek jde do funkce OR aby mohly fungovat nezávisle na sobě.

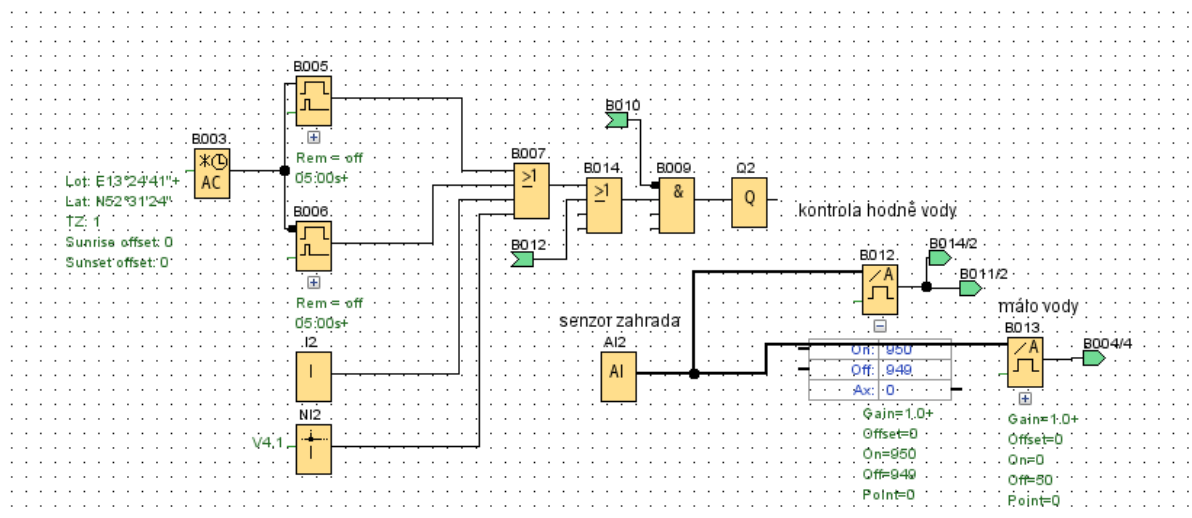


Nastavení pro kontrolu ze serveru proběhne úplně stejně jako u vstupu tlačítka pro ovládání pumpy v jímce. Pouze místo bitové hodnoty 0 zde zadáme 2.



Ovládání shromažďovací jámy

Program bude vypadat takto:



Je to vlastně obdoba programu pro ovládání jímky. Avšak zde máme nějaké prvky navíc. Protože chceme, aby se nám zahrada začala zalévat při východu a západu Slunce vždy na 20 minut, tak sem musíme přidat blok Astronomical clock. Tento blok nám vysílá ON, jakmile je světlo a OFF, jakmile je tma. Jediné nastavení je zde časového pásma, proto si nastavíme Berlín v Německu.

B003 [Astronomical clock] ✕

Parameter | Comment

Parameter

Block name:

Location Info

Location:

Longitude: E ° ' "

Latitude: N ° ' "

Time Zone:

Time offset

Sunrise offset: Minutes

Sunset offset: Minutes

Others

Protection Active

Za tím je blok Wiping Relay. Ten nám hodnotu ON propustí na námi určený čas, v tomto případě budeme chtít na 20 minut.

B005 [Wiping relay (pulse output)] ✕

Parameter | Comment

Parameter

Block name:

Off Time (TL)

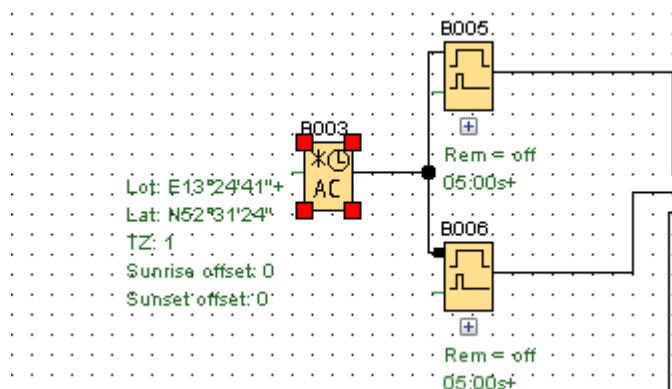
: Minutes (m:s)

Others

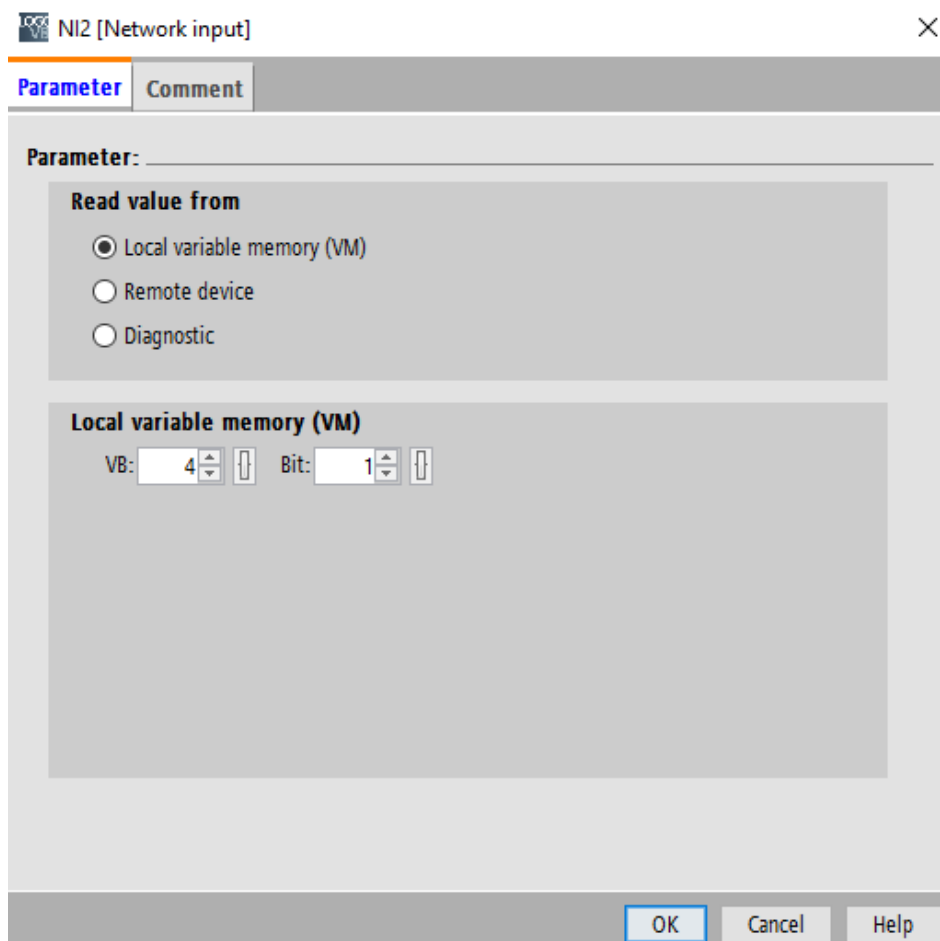
Retentivity

Protection Active

Tento blok zde musíme mít dvakrát, pro východ a západ Slunce. Protože nám astronomické hodiny vysílají ON signál, tak musíme druhý vstup znegovat, tak aby nám to reagovalo na západ.

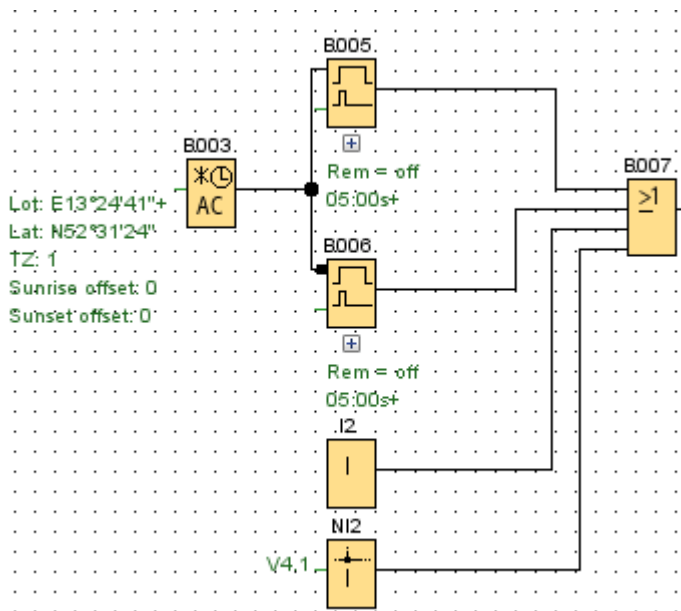


Pak zde máme jedno fyzické tlačítko a jedno tlačítko ovládané přes Node-Red server. Network



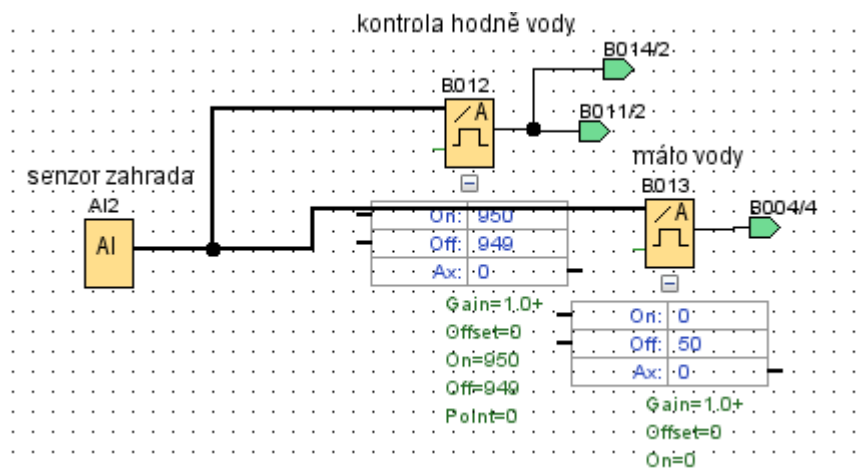
Input tlačítko musíme nastavit na VB je 4 a bit je 1.

Všechny tyto výstupy půjdou do OR bloku, aby bylo možné je zapínat nezávisle na sobě.



Z OR bloku nám výstup jde do dalšího OR bloku. To je výstup z funkce, která nám kontroluje hladinu vody v shromažďovací jámě. Do OR bloku to jde, protože pokud by hladina vody byla moc vysoká, tak nám to automaticky zapne pumpu na zahradě, aby bylo možné potřebnou vodu odčerpat a nedošlo k jejímu přetečení a následnému zničení nějakého zařízení.

Je zde vstup ze senzoru, který bude na dně nádržky a dva porovnávací bloky. Pokud hladina přesáhne nějakou hodnotu, tak se nám spustí daný signál ON nebo OFF.



Nakonec je zde blok AND, do kterého jde výstup z funkce STOP. Blok AND nám zase zajistí, že pokud se stiskne tlačítko STOP, tak se vše přeruší a nemůže se nic dále zničit.

