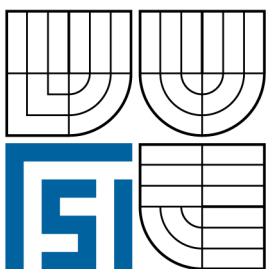


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA STROJNÍHO INŽENÝRSTVÍ
ÚSTAV AUTOMATIZACE A INFORMATIKY

FACULTY OF MECHANICAL ENGINEERING
INSTITUTE OF AUTOMATION AND COMPUTER SCIENCE

UKÁZKOVÉ ÚLOHY PRO VÝUKU AUTOMATIZACE SAMPLE TASKS FOR AUTOMATION EDUCATION

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

Aleš Dočekal

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. OLGA DAVIDOVÁ, Ph.D.

BRNO 2010

Zadání závěrečné práce

(na místo tohoto listu vložte originál a nebo kopii zadání Vaší práce)

Abstrakt

Cílem této bakalářské práce je vytvoření ukázkových úloh kombinačních a sekvenčních logických obvodů. Tyto ukázkové úlohy budou použity ve výuce předmětu automatizace. K návrhu úloh byl použit program LOGO!Soft Comfort. Ověření správné činnosti bylo provedeno na programovatelném automatu LOGO! 0BA5.

Abstract

The aim of this bachelor thesis puts mind to the problem of forming combinational and sequential logic circuit tasks. These demonstrational tasks will be used in the educational system for the subject automation. To design the tasks was used the program LOGO!Soft Comfort . Verification of proper running was done by the programmable logic controller LOGO! 0BA5.

Klíčová slova

Logické řízení, LOGO!Soft Comfort, programovatelný automat

KEYWORDS

Logic control, LOGO!Soft Comfort, programmable logic controller

Poděkování

Na tomto místě bych rád poděkoval své vedoucí práce Ing. Olze Davidové, Ph.D. za pomoc a rady, které mi poskytla při vypracování této práce.

Obsah

Zadání závěrečné práce	3
Abstrakt	7
Poděkování	9
Obsah	11
1 Úvod	13
2 Logické řízení	15
2.1 Formy popisu logické funkce	15
2.2 Kombinační logické obvody	16
2.3 Sekvenční logické obvody	16
2.3.1 Klopné obvody.....	16
2.3.1.1 Klopný obvod RS.....	16
3 Popis vývojového prostředí LOGO!Soft Comfort	18
3.1 Popis uživatelského rozhraní	18
3.1.1 Panel nástrojů pro simulaci.....	18
3.1.2 Stavová lišta.....	19
3.2 Popis tvorby programu	19
3.2.1 Vytvoření programu.....	19
3.2.2 Vytvoření objektu	20
3.2.3 Vlastnosti objektu	20
4 Popis programovatelného automatu LOGO!0BA5	21
5 Navržené úlohy	22
5.1 Ukázkové úlohy kombinačního logického obvodu	22
5.1.1 3-bitová sčítačka	22
5.1.2 Hlídaní úniku kapalin.....	26
5.1.3 Grayův kód	31
5.2 Ukázkové úlohy sekvenčního logického obvodu	34
5.2.1 Ovládání garážových vrat	34
5.2.2 Plnění kartónových krabic	39
5.2.3 Řízení pískování odlitku	43
6 Závěr	48
Seznam použité literatury	49

1 Úvod

V dnešní době jsou patrně nejrozšířenějšími řídicími systémy v automatizaci programovatelné automaty, neboli PLC. Původně byly programovatelné automaty navrženy k řešení úloh logického řízení, často jako přímá náhrada reléové logiky. V současných aplikacích se však zvyšuje podíl úloh regulačního typu, úloh monitorování řízeného procesu i úloh analogového měření.

Programovatelné automaty dělíme podle konstrukce na kompaktní a modulární. Kompaktní automaty jsou řešeny jako menší systémy a modulární jako větší. Princip činnosti kompaktních i modulárních programovatelných automatů je stejný, konstrukčním pojetím a uživatelskou koncepcí jsou však obě kategorie odlišné. Kompaktní automaty měly původně pevně danou konfiguraci integrovaných obvodů a byly uzavřeny v pouzdře. V poslední době je i u kompaktních programovatelných automatů snaha o určitý stupeň modularity, takže je možné přizpůsobit sestavu automatu potřebám konkrétní aplikace. [5]

Mezi kompaktní programovatelné automaty patří Siemens LOGO!, tento malý programovatelný automat je určený pro nejjednodušší aplikace. Lze s ním například řídit osvětlení výkladní skříně, řízení rolet, zavlažování rostlin ve skleníku atd. Existuje již 7 řad tohoto automatu (0BA0 - 0BA6). Základní modul má 8 vstupů a 4 výstupy. Lze jej rozšířit pomocí dalších modulů. Programy pro LOGO! můžeme vytvářet přímo na displeji nebo pomocí prostředí LOGO!Soft Comfort. Je to uživatelsky velice příjemné prostředí. Vytvoření programu je možné v jazyce funkční bloků nebo pomocí kontaktních schémat. Je možná simulace návrhu logického obvodu.

Automat LOGO! je součástí vybavení laboratoře v rámci Ústavu automatizace a informatiky Fakulty strojního inženýrství Vysokého učení technického v Brně. Tato bakalářská práce se zabývá návrhem úloh ve vývojovém programu LOGO!Soft Comfort.

2 Logické řízení

Využívá k řízení dvouhodnotových veličin. Jejich působení je takové, že jsou vždy jen dvě možnosti – ventil je otevřen / zavřen, vypínač je sepnut / vypnut, atd. Podobně i informace o stavu objektu jsou dvouhodnotové veličiny – hladina je nad / pod minimální hodnotou, teplota je nad / pod 18°C, atd. Dvouhodnotové veličiny jsou formálně vyjadřovány hodnotami 0 a 1. Jsou analogické s proměnnými výrokové logiky, a proto jsou vztahy mezi proměnnými nazývány logické funkce a řídicí obvody pracující na tomto principu jsou logické řídicí obvody. [1]

2.1 Formy popisu logické funkce

a) Pravdivostní tabulka

Je to nejběžnější forma zápisu logické funkce. Obsahuje všechny kombinace vstupních hodnot a jim odpovídajících výstupy. Obsahuje tedy 2^n řádků, kde n je počet vstupních proměnných.

a	b	c	y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Obr.2.1 Pravdivostní tabulka

b) Logický výraz

Je to algebraická forma vyjádření logické funkce. Každou logickou funkci můžeme vyjádřit ve dvou normálových formách a to v konjunktivní normálové formě (KNF) a v disjunktivní normálové formě (DNF).

Konjunktivní normálová forma se skládá z logických součtů (OR) proměnných, případně jejich negací a jednotlivé součty jsou spojeny logickým součinem (AND). Z pravdivostní tabulky vypisujeme řádky, kde nabývá výstupní hodnota 0. Negovaná proměnná je zastoupena ve výrazu přímo. Vstupní proměnná, která má hodnotu 1 je ve výrazu zastoupena svoji negací.

$$y = (a + b + c) + (a + b + \bar{c}) + (a + \bar{b} + c) + (\bar{a} + b + c)$$

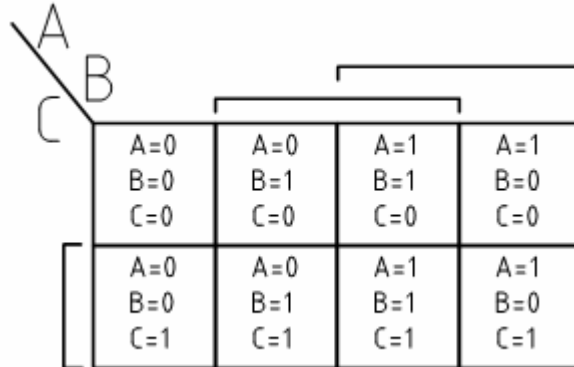
Disjunktivní normálová forma je tvořena logickým součinem (AND) proměnných, případně jejich negací, a jednotlivé klauzule jsou spojeny logickým součtem (OR). Z pravdivostní tabulky vypisujeme řádky, kde nabývá výstupní hodnota stavu 1.[9]

Negovaná proměnná je zastoupena ve výrazu negací. Vstupní proměnná, která má hodnotu 1 je zastoupena přímo.

$$y = \bar{a}\bar{b}c + \bar{a}b\bar{c} + a\bar{b}\bar{c} + abc$$

c) Karnaughova mapa

Slouží nejen pro zápis logické funkce, ale především pro její minimalizaci. Počet políček mapy je tolik, kolik je kombinací vstupních proměnných. Obsahuje tedy 2^n políček, kde n je počet vstupních proměnných.



Obr.2.2 Karnaughova mapa pro tři proměnné

2.2 Kombinační logické obvody

Kombinační logické obvody jsou takové logické obvody, ve kterých jsou výstupní stavy určeny jednoznačně vstupními stavy. Nezávisí tedy na předchozích hodnotách a neobsahuje žádné paměťové členy.

2.3 Sekvenční logické obvody

Sekvenční obvody se liší tím od kombinačních tím, že okamžitá hodnota vstupu závisí nejen na okamžitých hodnotách vstupů, ale i na jejich předcházející posloupnosti. Obsahují tedy paměťové prvky, které uchovávají předchozí stav obvodu a reagují na něj.

2.3.1 Klopné obvody

Klopné obvody jako paměťové členy jsou nejjednodušší sekvenční logické obvody. Úkolem klopných obvodů je zaznamenat přítomnost přechodné informace a uchovat tento stav i tehdy, když informace zmizí. Jejich výstupní stavy se mění skokem mezi dvěma hodnotami logických úrovní.

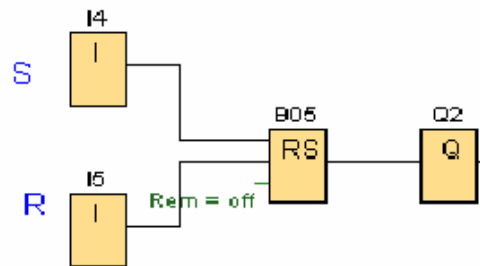
2.3.1.1 Klopný obvod RS

Nejjednodušším klopným je RS obvod. Má dva vstupy R a S, výstup Q a jeho negaci \bar{Q} . Princip činnosti:

- pokud přivedeme na vstupy R a S nulový signál, pak klopný obvod zůstává v předchozím stavu
- přivedeme-li na nastavovací vstup signál $S=1$ a na resetovací vstup signál $R=0$, přejde klopný obvod do stavu logická 1.
- přivedeme-li na resetovací vstup signál $R=1$ a na nastavovací $S=0$, přejde klopný obvod do stavu logická 0.
- přivedeme-li na vstupy $R=1$ a $S=1$, což znamená, že se mají do stavu logická jedna nastavit současně oba výstup, což by odporovali funkci klopného obvodu

S	R	Q^{t+1}	$\overline{Q^{t+1}}$
0	0	Q^t	$\overline{Q^t}$
0	1	0	1
1	0	1	0
1	1	Zakázaný stav	

Obr.2.1 Pravdivostní tabulka RS klopného obvodu

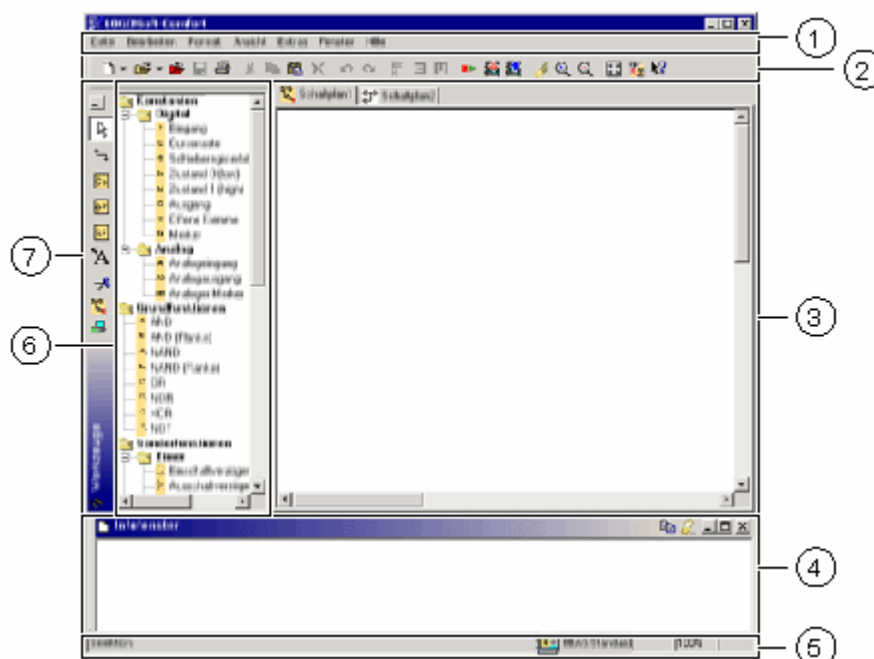


Obr.2.2 Zapojení klopného obvodu RS v LogoSoft!Comfort [4]

3 Popis vývojového prostředí LOGO!Soft Comfort

V této kapitole je popsáno uživatelské prostředí LOGO!Soft Comfort V6.0. Popis prostředí vychází z manuálu LOGO! Soft Comfort [3]. Toto vývojové prostředí slouží pro vytvoření logických obvodů na programovatelný automat Logo!.

3.1 Popis uživatelského rozhraní



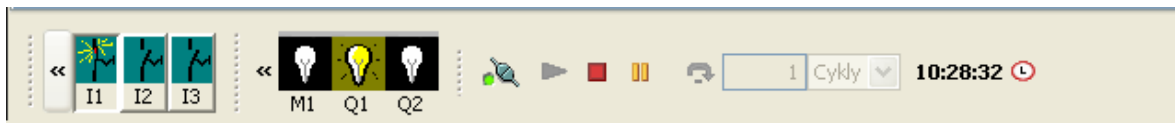
Obr.3.1 Okno programu

- 1 Lišta menu
- 2 Standardní nástrojová lišta
- 3 Programovací rozhraní
- 4 Informační pole
- 5 Stavová lišta
- 6 Nabídka programových funkcí
 - konstanty a konektory
 - základní funkce
 - speciální funkce
- 7 Lišta nástrojů pro programování

3.1.1 Panel nástrojů pro simulaci

Po spuštění simulace klávesou F3 se zobrazí panel nástrojů Obr.2. Tento panel obsahuje:

- ikony (spínače) pro kontrolu hodnot vstupních proměnných.
- ikony (žárovky) pro zobrazování výstupních hodnot.
- ikonu pro simulaci výpadku napájení
- ikony pro řízení simulace
- ikony pro časovou kontrolu



Obr.3.2 Panel nástrojů pro simulaci

3.1.2 Stavová lišta




Obr.3.3 Stavová lišta

- 1- informační pole
- 2- zobrazuje vybraný typ LOGO!
- 3- Zobrazuje aktuální stav měřítka lupy
- 4- Zobrazuje aktuální stav schématu

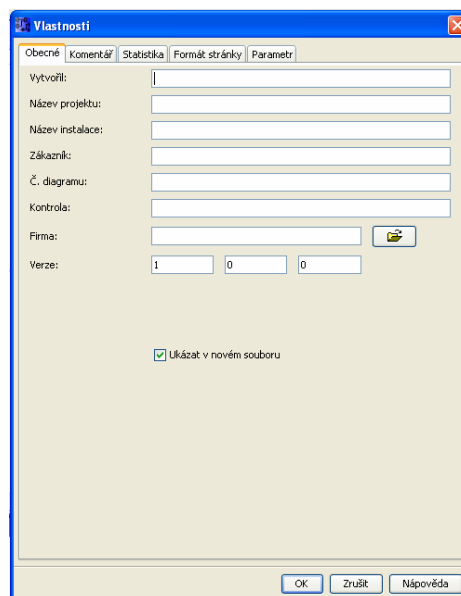
3.2 Popis tvorby programu

V této kapitole si popíšeme tvorbu programu. Budeme používat standardní nástrojovou lištu a nabídku programových funkcí. Bloky budeme vkládat do programovacího rozhraní.

3.2.1 Vytvoření programu

Po spuštění programu LOGO!Soft Comfort vytvoříme nový program kliknutím na ikonu na standardní nástrojové liště . Dojde ke spuštění nabídky vlastnosti projektu (Obr.4) a po vyplněním dojde ke spuštění projektu.

V nabídce Nástroje/Možnosti můžeme nastavit v jakém jazyce se bude projekt spouštět. Můžeme zvolit mezi jazykem funkčních bloků (FBD) a nebo jazykem kontaktních schémat (LAD).



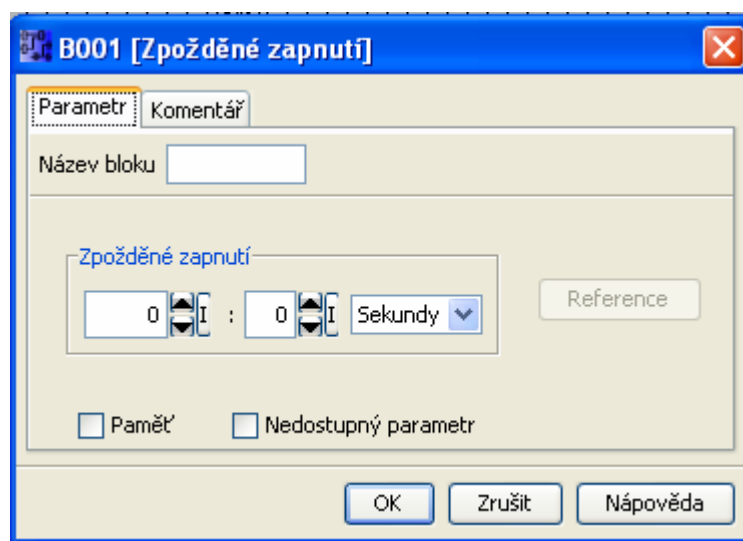
Obr.3.4 Vlastnosti programu

3.2.2 Vytvoření objektu

Objekty vkládáme z nabídky nástrojů (Obr.3.1) Vybereme blok z nabídky konstant speciálních funkcí a nebo základních funkcí. Klikneme nepožadovaný blok a pak klikneme do programovacího rozhraní, blok se v něm objeví.

3.2.3 Vlastnosti objektu

Kliknutím pravým tlačítkem na blok a následným výběrem vlastnosti bloku se nám otevře nabídka viz. obr.5. Obsah vlastností bloku závisí na vybraném objektu. Za objekty se nepovažují pouze bloky, ale i spojovací čáry, programovací plocha a nástrojové lišty. Ke každému objektu můžeme napsat komentář.

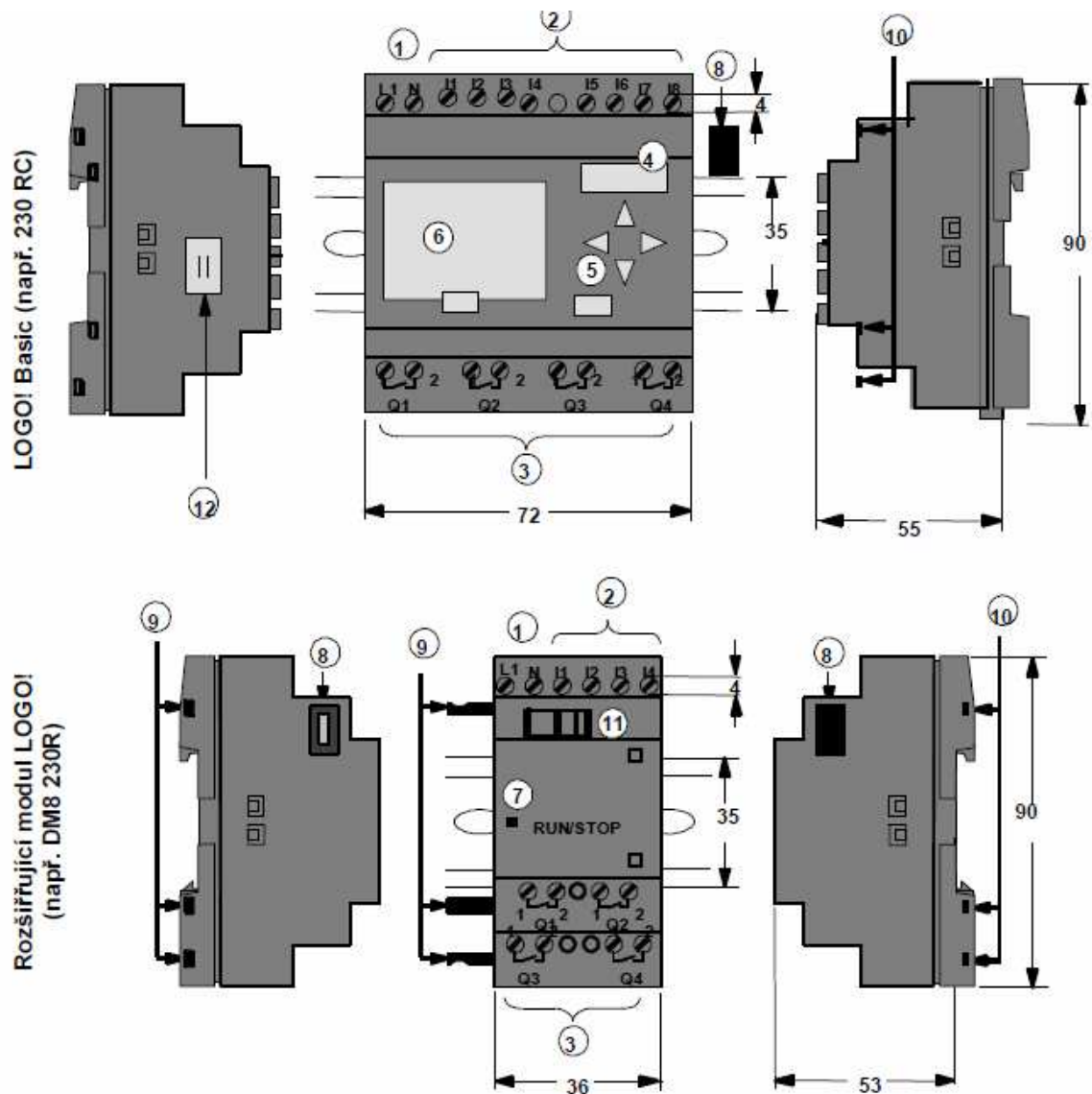


Obr.3.5 Vlastnosti objektu

4 Popis programovatelného automatu LOGO!0BA5

V této kapitole je popsán programovatelný automat LOGO!0BA5. Vychází z manuálu firmy Siemens. [2]

Uspořádání automatu Logo!



Obr.4.1 Programovatelný automat LOGO! a přídatný modul LOGO!

1 -připojení zdroje
2-vstupy
3-výstupy
4-slot modulu s krytem
5-Řídící panel
6-LCD

7-Stavový displej RUN/STOP
8-Rozšiřující rozhraní
9-Mechanické kódování piny
10-Mechanické kódování - zásuvky
11-Šoupátko
12-Zemní svorka

5 Navržené úlohy

V této kapitole budou popsány navržené úlohy na kombinační logický obvod a na sekvenční logický obvod. U všech úloh bude popis zadání, vysvětlení postupu řešení a realizace v LOGO!Soft Comfort.

5.1 Ukázkové úlohy kombinačního logického obvodu

5.1.1 3-bitová sčítačka

Zadání

- navrhnete a realizujete polosčítačku
- navrhnete a realizujete úplnou sčítačku
- navrhnete logický obvod pro 3-bitovou sčítačku
- k realizaci obvodu použijte program LOGO!Soft Comfort

Navržení a realizace polosčítačky

Realizuje sčítání dvou jednomístných binárných čísel. Vstupem jsou dvě hodnoty A,B. Výstupem je součet S a C je přenos do vyššího řádu. Poloviční sčítačka neumí ke sčítancům přičíst přenos z nižšího řádu. Proto bychom z ní nemohli sestavit vícebitovou sčítačku.

A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

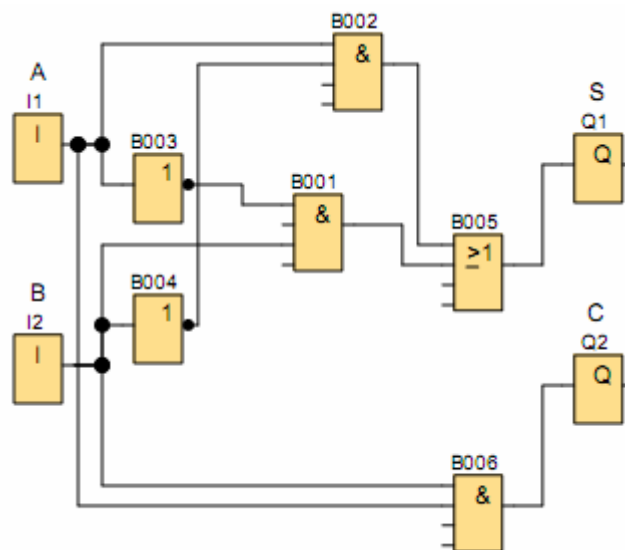
Obr. 5.1 Pravdivostní tabulka polosčítačky

Algebraické rovnice pro polosčítačku:

$$S = \bar{A}B + A\bar{B}$$

$$C = AB$$

Realizace v LOGO!Soft Comfort:



Obr.5.2 Realizace polosčítačky

Navržení a realizace úplné sčítačky

Úplná sčítačka realizuje sčítání dvou jedno-bitových čísel, vrátit jednobitový výsledek a přenos. Umí navíc aplikovat přenos z nižšího řádu.

Má 3 vstupy – A, B a přenos z nižšího řádu C_{in}

2 výstupy – S a C_{out} přenos do vyššího řádu

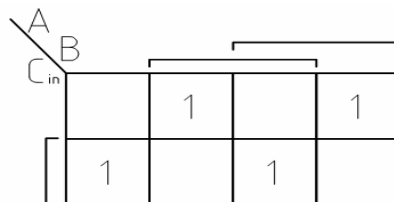
A	B	C_{in}	S	C_{out}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Obr.5.3 Pravdivostní tabulka úplné sčítačky

Algebraické rovnice pro úplnou sčítačku:

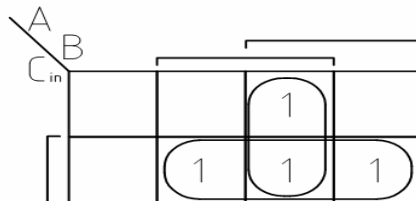
$$S = \overline{A}\overline{B}C_{in} + \overline{A}B\overline{C_{in}} + A\overline{B}\overline{C_{in}} + ABC_{in}$$

$$C_{out} = \overline{A}BC_{in} + A\overline{B}C_{in} + ABC_{in} + ABC_{in}$$



Obr.5.4 Karnaughova mapa pro součet S

Z Karnaughovy mapy vyplývá, že součet S je v minimalizovaném tvaru.

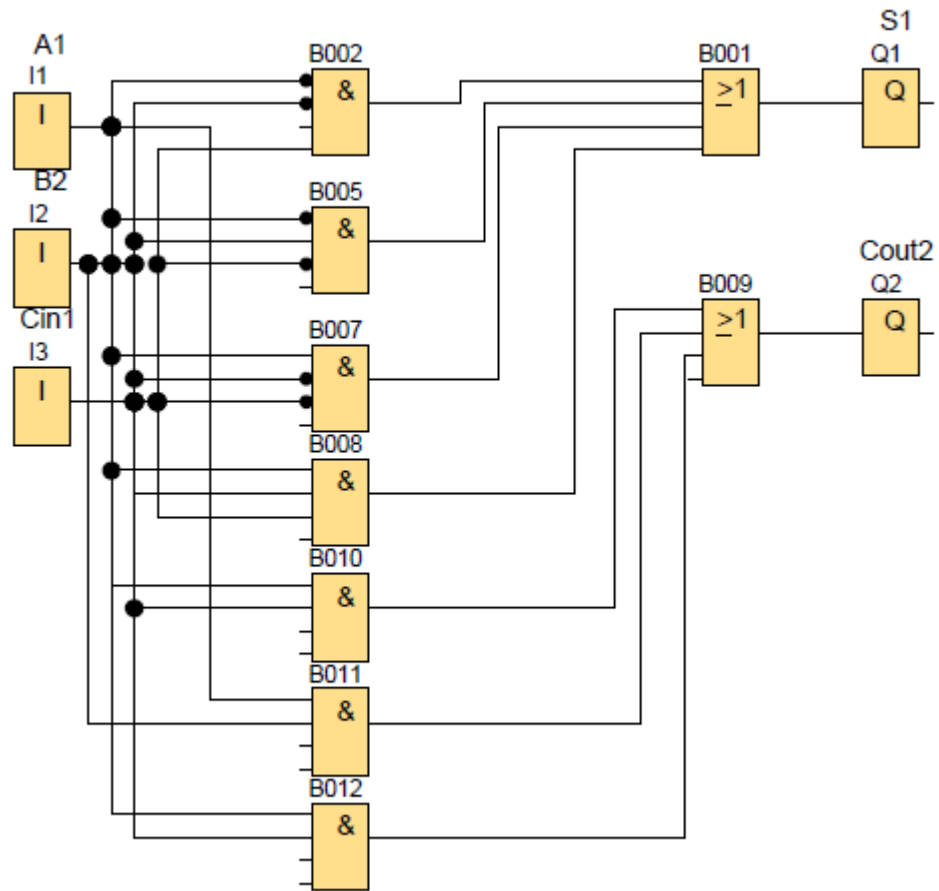


Obr.5.5 Karnaughova mapa pro C_{out}

Z Karnaughovy mapy získáváme pro přenos do vyššího řádu tento minimalizovaný tvar:

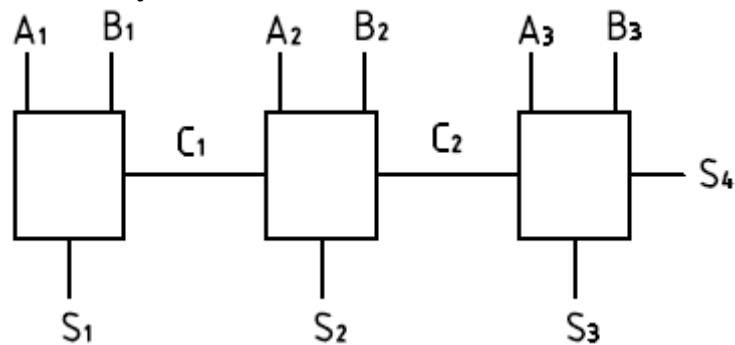
$$C_{out} = AC_{in} + AB + BC_{in}$$

Realizace v LOGO!Soft Comfort:



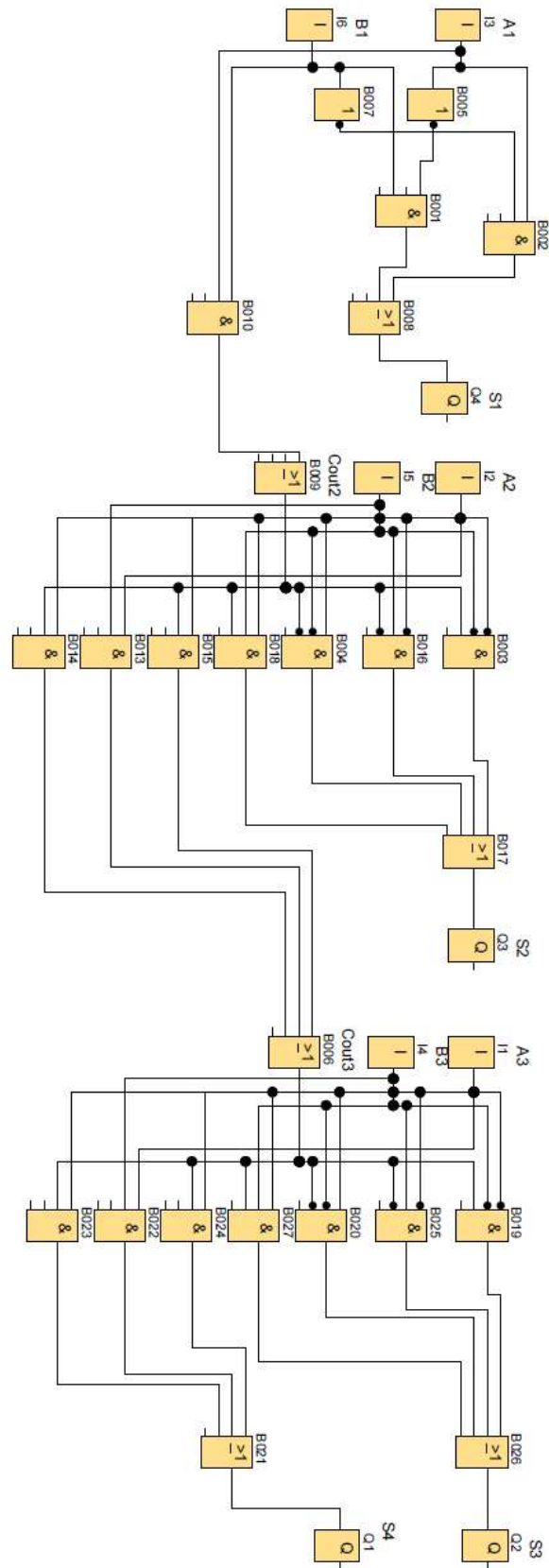
Obr.5.6 Realizace úplné sčítačky

Realizace 3-bitové sčítačky



Obr.5.7 Blokové schéma 3-bitové sčítačky

Z blokového schématu 3-bitové sčítačky vyplývá, že se jedná o kombinaci polosčítačky a dvou úplných sčítaček.



Obr.5.8 Realizace 3-bitové sčítačky

5.1.2 Hlídaní úniku kapalin

Zadání

V podniku jsou tři nádrže, které mají společnou jímku pro nežádoucí únik kapalin. Tři senzory detekují únik kapalin do jímky. Při úniku jedné kapaliny bude svítit oranžová kontrolka, při úniku dvou kapalin červená kontrolka a při úniku tří kapalin červená + spuštění alarmu.

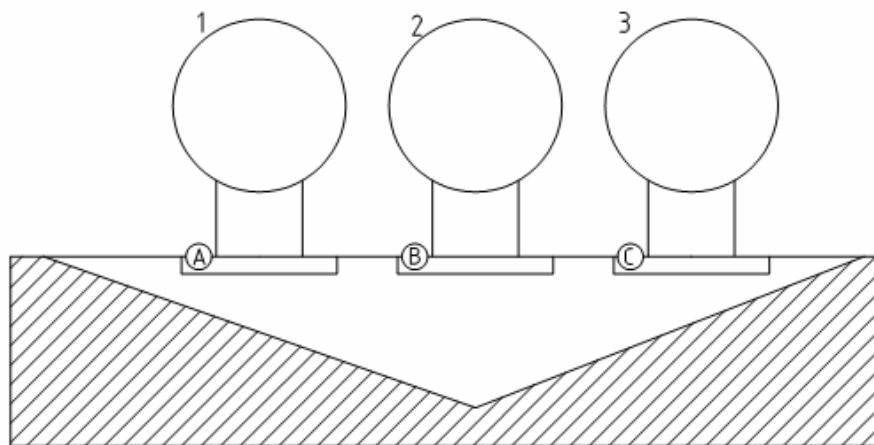
Vstupy:

A=1 senzor A detekuje únik z nádrže 1

B=1 senzor B detekuje únik z nádrže 2

C=1 senzor C detekuje únik z nádrže 3

Úkol: Proved'te realizaci prvky Booleovy algebry, prvky NOR a NAND. Obvody realizujte v programu LOGO!Soft Comfort a správnou funkčnost ověřte na PA.



Obr.5.9 Nádrže v podniku

Ze zadání sestavíme tuto pravdivostní tabulku:

A	B	C	ORA	CER	ALARM
0	0	0	0	0	0
0	0	1	1	0	0
0	1	0	1	0	0
0	1	1	0	1	0
1	0	0	1	0	0
1	0	1	0	1	0
1	1	0	0	1	0
1	1	1	0	1	1

Obr.5.10 Pravdivostní tabulka-hlídaní úniku kapalin

Z pravdivostní tabulky vyjádříme funkci v disjunktivní normální formě.

Algebraická rovnice pro oranžové světlo:

$$ORA = \overline{A}BC + A\overline{B}C + A\overline{B}\overline{C}$$

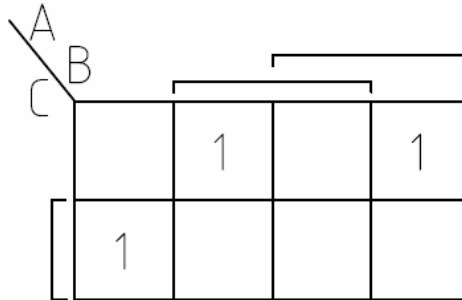
Algebraická rovnice pro červené světlo:

$$CER = \overline{A}BC + A\overline{B}C + A\overline{B}\overline{C} + ABC$$

Algebraická rovnice pro spuštění alarmu:

$$ALARM = ABC$$

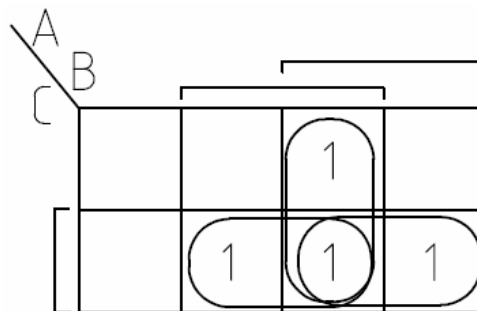
Karnaughova mapa pro oranžové světlo:



Obr.5.11 mapa pro oranžové světlo

Z Karnaughovy mapy je patrné, že se jedná o minimalizovaný výraz.

Karnaughova mapa pro červené světlo:



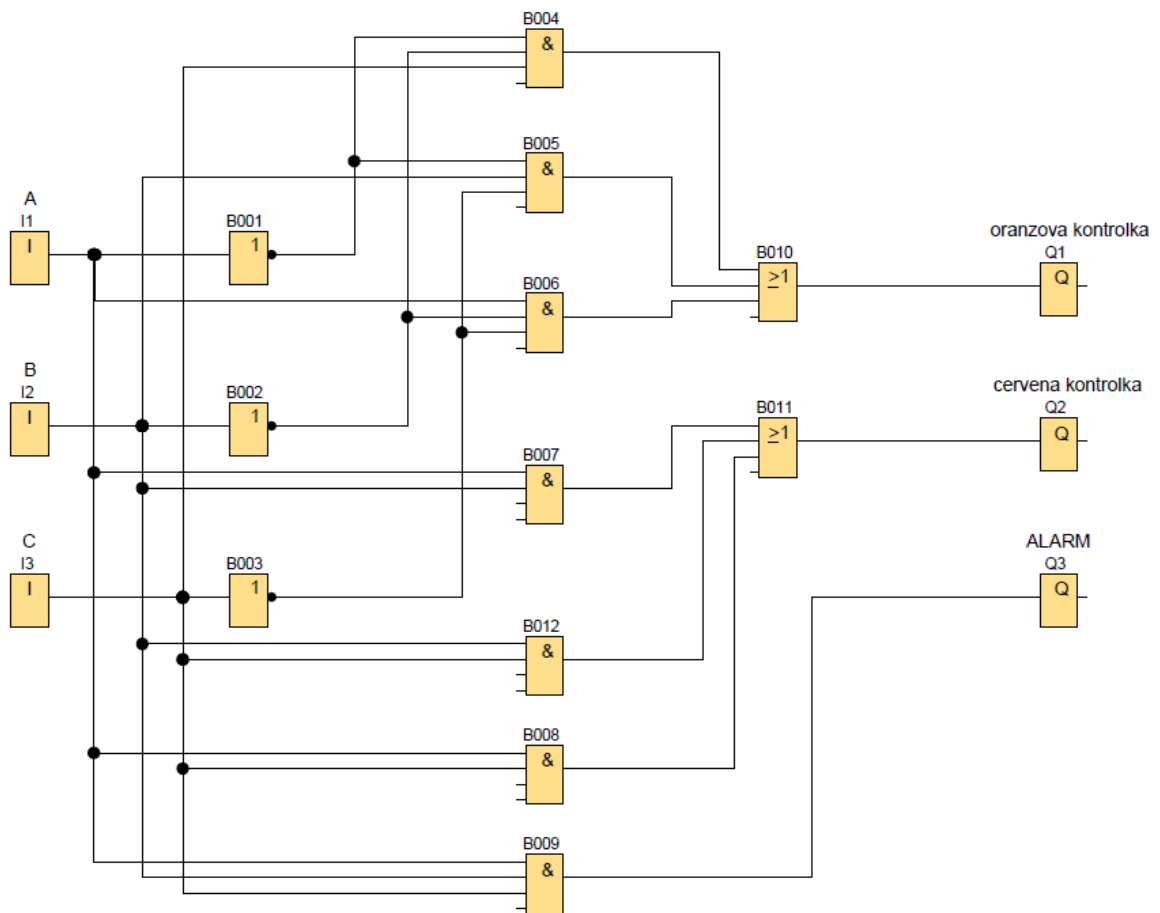
Obr.5.12 mapa pro červené světlo

Z Karnaughovy mapy získáme tento minimalizovaný tvar:

$$CER = AB + BC + AC$$

Realizace prvky NOT, OR, AND

Při realizaci prvky Booleovy algebry bylo použito 3x hradlo NOT, 7x hradlo NAND a 3x hradlo OR.



Obr.5.13 Realizace v LOGO!Soft Comfort prvky NOT,OR a AND

Realizace prvky NAND

Převod na NAND funkci rovnice pro oranžové světlo:

$$ORA = \overline{ABC} + \overline{ABC} + \overline{ABC} = \overline{\overline{ABC}} + \overline{\overline{ABC}} + \overline{\overline{ABC}} = \overline{\overline{\overline{ABCABCABC}}}$$

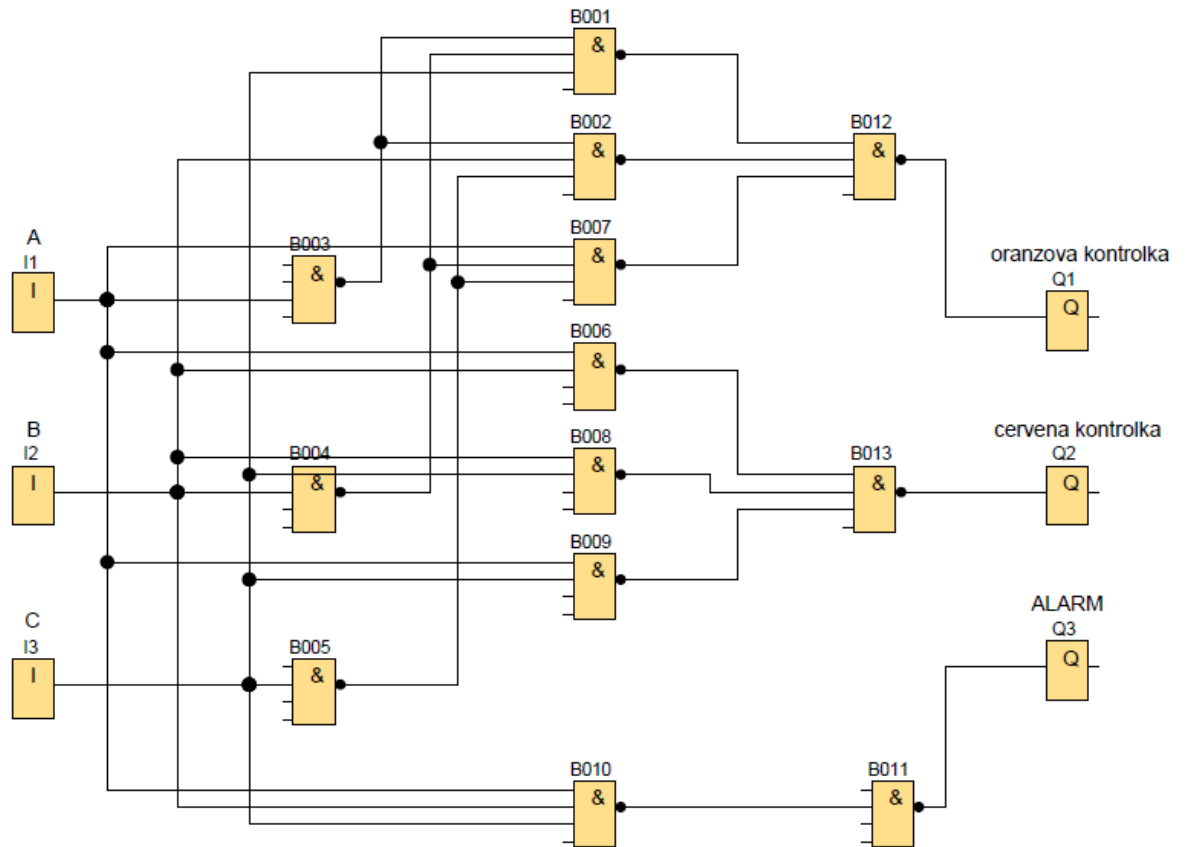
Převod na NAND funkci rovnice pro červené světlo:

$$CER = AB + BC + AC = \overline{\overline{AB}} + \overline{\overline{BC}} + \overline{\overline{AC}} = \overline{\overline{\overline{ABBCAC}}}$$

Převod na NAND funkci rovnice pro spuštění alarmu:

$$ALARM = \overline{\overline{ABC}}$$

Při této realizaci bylo použito 13x hradlo NAND.



Obr.5.14 Realizace v LOGO!Soft Comfort prvky NAND

Realizace prvky NOR

Převod na NOR funkci rovnice pro oranžové světlo:

$$\begin{aligned} ORA &= \overline{\overline{A}BC} + \overline{A\overline{B}C} + \overline{AB\overline{C}} = \overline{\overline{\overline{A}BC}} + \overline{\overline{A\overline{B}C}} + \overline{\overline{AB\overline{C}}} = \overline{\overline{\overline{A} + \overline{B} + \overline{C}}} + \overline{\overline{\overline{A} + \overline{B} + \overline{C}}} + \overline{\overline{\overline{A} + \overline{B} + \overline{C}}} = \\ &= \overline{\overline{\overline{A} + \overline{B} + \overline{C}}} + \overline{\overline{\overline{A} + \overline{B} + \overline{C}}} + \overline{\overline{\overline{A} + \overline{B} + \overline{C}}} = \overline{\overline{\overline{A} + \overline{B} + \overline{C}}} + \overline{\overline{\overline{A} + \overline{B} + \overline{C}}} + \overline{\overline{\overline{A} + \overline{B} + \overline{C}}} \end{aligned}$$

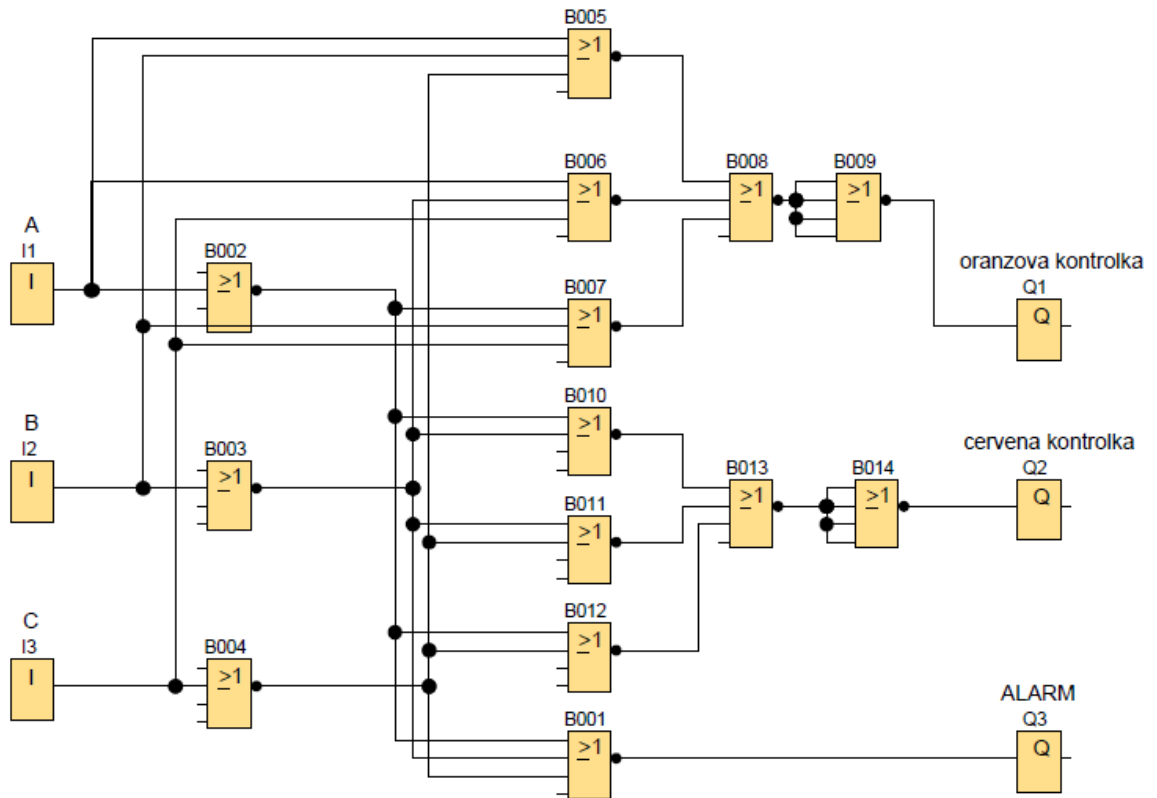
Převod na NOR funkci rovnice pro červené světlo:

$$CER = \overline{AB} + \overline{BC} + \overline{AC} = \overline{\overline{\overline{AB}}} + \overline{\overline{\overline{BC}}} + \overline{\overline{\overline{AC}}} = \overline{\overline{\overline{A} + \overline{B}}} + \overline{\overline{\overline{B} + \overline{C}}} + \overline{\overline{\overline{A} + \overline{C}}} = \overline{\overline{\overline{A} + \overline{B} + \overline{B} + \overline{C}}} + \overline{\overline{\overline{A} + \overline{C}}}$$

Převod na NOR funkci rovnice pro spuštění alarmu:

$$ALARM = \overline{\overline{\overline{ABC}}} = \overline{\overline{\overline{A} + \overline{B} + \overline{C}}}$$

Při realizaci prvky NOR bylo použito 14x hradlo NOR.



Obr.5.15 Realizace v LOGO!Soft Comfort prvky NOR

5.1.3 Grayův kód

Zadání

Navrhňte logický obvod, který bude převádět čtyř místný binární kód na Grayův kód. Obvod realizujte v LOGO!Soft Comfort . Činnost obvodu ověřte na PA.

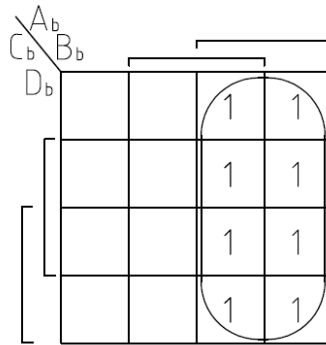
A _b	B _b	C _b	D _b	A _g	B _g	C _g	D _g
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	1
0	0	1	1	0	0	1	0
0	1	0	0	0	1	1	0
0	1	0	1	0	1	1	1
0	1	1	0	0	1	0	1
0	1	1	1	0	1	0	0
1	0	0	0	1	1	0	0
1	0	0	1	1	1	0	1
1	0	1	0	1	1	1	1
1	0	1	1	1	1	1	0
1	1	0	0	1	0	1	0
1	1	0	1	1	0	1	1
1	1	1	0	1	0	0	1
1	1	1	1	1	0	0	0

Obr.5.16 Pravdivostní tabulka – Grayův kód

Algebraické rovnice:

$$\begin{aligned}
 A_g &= \overline{A_b} \overline{B_b} \overline{C_b} \overline{D_b} + \overline{A_b} \overline{B_b} \overline{C_b} D_b + \overline{A_b} \overline{B_b} C_b \overline{D_b} + \overline{A_b} \overline{B_b} C_b D_b + \overline{A_b} B_b \overline{C_b} \overline{D_b} + \overline{A_b} B_b \overline{C_b} D_b + \\
 &+ \overline{A_b} B_b C_b \overline{D_b} + \overline{A_b} B_b C_b D_b \\
 B_g &= \overline{A_b} \overline{B_b} \overline{C_b} \overline{D_b} + \overline{A_b} \overline{B_b} \overline{C_b} D_b + \overline{A_b} \overline{B_b} C_b \overline{D_b} + \overline{A_b} \overline{B_b} C_b D_b + \overline{A_b} B_b \overline{C_b} \overline{D_b} + \overline{A_b} B_b \overline{C_b} D_b + \\
 &+ \overline{A_b} B_b C_b \overline{D_b} + \overline{A_b} B_b C_b D_b \\
 C_g &= \overline{A_b} \overline{B_b} \overline{C_b} \overline{D_b} + \overline{A_b} \overline{B_b} \overline{C_b} D_b + \overline{A_b} \overline{B_b} C_b \overline{D_b} + \overline{A_b} \overline{B_b} C_b D_b + \overline{A_b} B_b \overline{C_b} \overline{D_b} + \overline{A_b} B_b \overline{C_b} D_b + \\
 &+ \overline{A_b} B_b C_b \overline{D_b} + \overline{A_b} B_b C_b D_b \\
 D_g &= \overline{A_b} \overline{B_b} \overline{C_b} \overline{D_b} + \overline{A_b} \overline{B_b} \overline{C_b} D_b + \overline{A_b} \overline{B_b} C_b \overline{D_b} + \overline{A_b} \overline{B_b} C_b D_b + \overline{A_b} B_b \overline{C_b} \overline{D_b} + \overline{A_b} B_b \overline{C_b} D_b + \\
 &+ \overline{A_b} B_b C_b \overline{D_b} + \overline{A_b} B_b C_b D_b
 \end{aligned}$$

Karnaughova mapa pro první bit A_g :

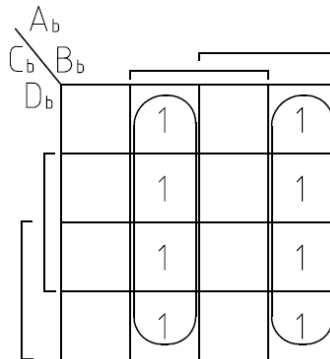


Obr.5.17 Karnaughova mapa pro A_g

Z Karnaughovy mapy získáváme tento minimalizovaný tvar:

$$A_g = \overline{A_b}$$

Karnaughova mapa pro druhý bit B_g :

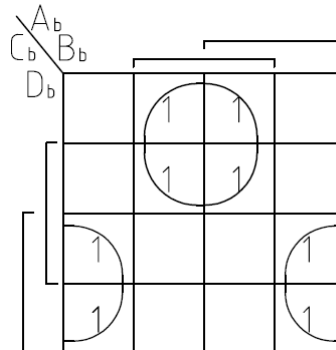


Obr.5.18 Karnaughova mapa pro B_g

Z Karnaughovy mapy získáváme tento minimalizovaný tvar:

$$B_g = \overline{A_b}B_b + A_b\overline{B_b}$$

Karnaughova mapa pro třetí bit C_g :

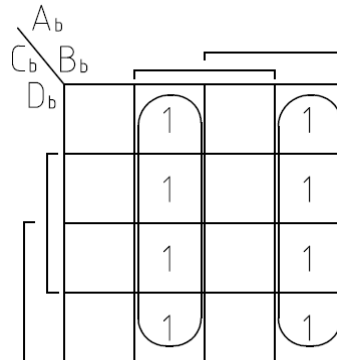


Obr.5.19 Karnaughova mapa pro C_g

Z Karnaughovy mapy získáváme tento minimalizovaný tvar:

$$C_g = B_b\overline{C_b} + \overline{B_b}C_b$$

Karnaughova mapa pro čtvrtý bit D_g :

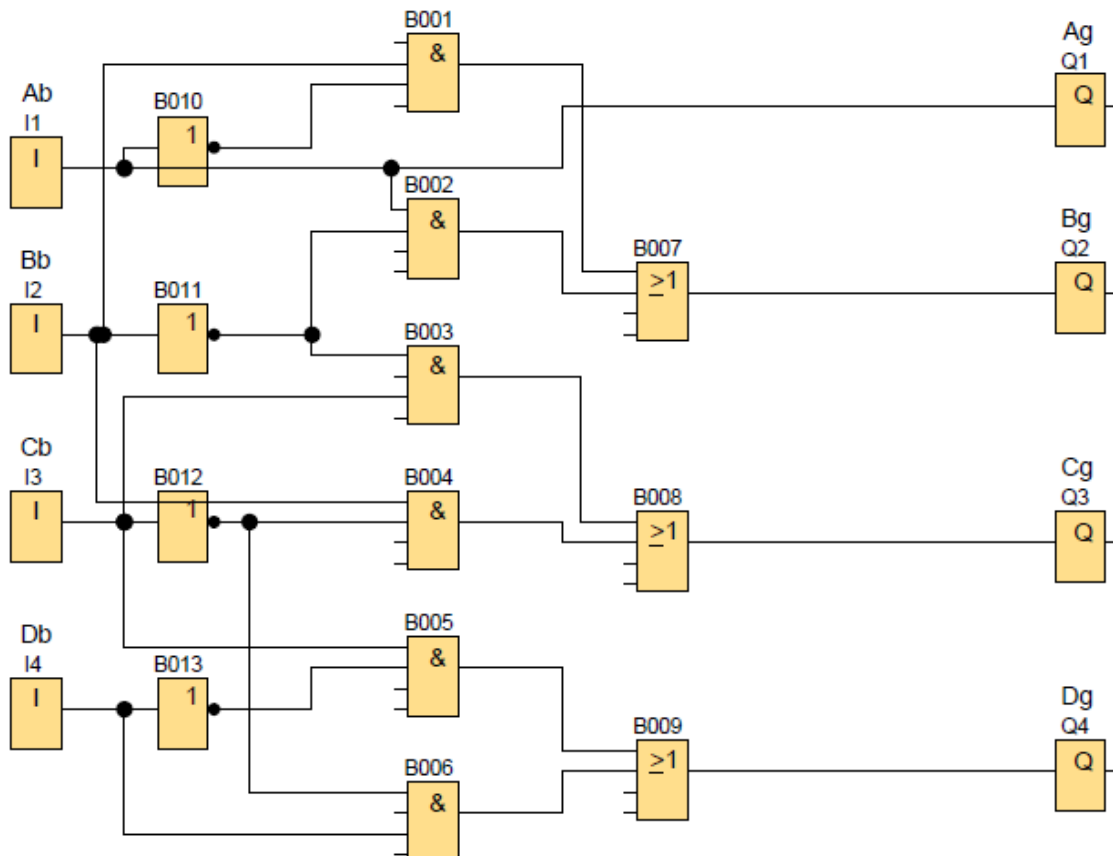


Obr.5.20 Karnaughova mapa pro D_g

Z Karnaughovy mapy získáváme tento minimalizovaný tvar:

$$D_g = \overline{C_b}D_b + C_b\overline{D_b}$$

Realizace v LogoSoft!Comfort



Obr.5.21 Realizace Převodu binárního kódu na Grayův kód

5.2 Ukázkové úlohy sekvenčního logického obvodu

5.2.1 Ovládání garážových vrat

Zadání

Proveďte návrh sekvenčního logického obvodu pro řízení ovládání garážových vrat.

Vstupní proměnné:

$O=1$... požadavek pro otevření vrat

$Z=1$... Požadavek pro zavření vrat

$S_O=1$... spínač pro hlášení maximálního otevření vrat

$S_Z=1$... spínač pro hlášení zavřených vrat

Výstupní proměnné:

$P_O=1$...zapnutí pohonu pro otevření vrat

$P_Z=1$...zapnutí pohonu pro zavření vrat

Předpokládá se bez poruchovost spínačů – nenastane $S_O=S_Z=1$ a nemůže nastat požadavek pro otevření vrat a zavření vrat - $O=Z=1$. Požadavek pro otevření nebo zavření vrat může přijít pouze při $S_O=1$ nebo $S_Z=1$

Realizujte jako klopný obvod RS v LOGO!Soft Comfort. Na PA ověřte správnou činnost logického obvodu.

Pravdivostní tabulku sestavíme z možných časových sledů vstupů a výstupů.

Stav	O	Z	S_O	S_Z	P_O	P_Z	
1	0	0	0	1	0	0	Vrata jsou zavřena
2	1	0	0	1	1	0	Požadavek pro otevření vrat, spuštění pohonu pro otevření
3	0	0	0	0	1	0	Běží motor pro otvírání
4	0	0	1	0	0	0	Dosažení spínače otevřených dveří
5	0	1	1	0	0	1	Požadavek pro zavření vrat, spuštění pohonu pro zavření
6	0	0	0	0	0	1	Běží motor pro zavírání
1	0	0	0	1	0	0	opakování

Obr.5.22 Pravdivostní tabulka – ovládání garážových dveří

U stavů 3 a 6 jsou stejné vstupní proměnné, ale jiné výstupní, proto se jedná o sekvenční úlohu.

Řešení ovládání pro pohon P_o :

Stav č.	O	Z	S_o	S_z	P_o	R	S
1	0	0	0	1	0	0	0
2	1	0	0	1	1	0	1
3	0	0	0	0	1	0	0
4	0	0	1	0	0	1	0
5	0	1	1	0	0	0	0
6	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0
7	1	0	1	0	0	1	0

Obr.5.23 Pravdivostní tabulka – pro ovládání pohonu P_o

Algebraický výraz pro Set S:

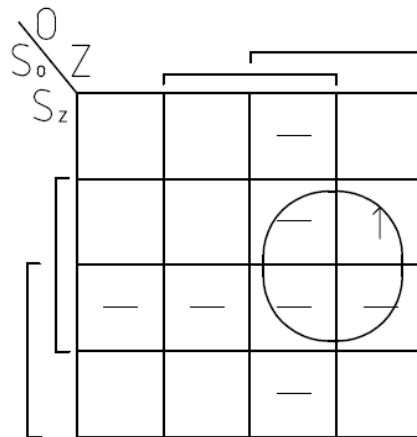
$$S = O\overline{Z}S_oS_z$$

Algebraický výraz pro Reset R:

$$R = O\overline{Z}S_o\overline{S_z} + O\overline{Z}S_oS_z$$

Využijeme stavy, které nemohou nastat a doplníme je logickými jedničkami. Tím zjednodušíme funkci.

Karnaughova mapa pro Set S:

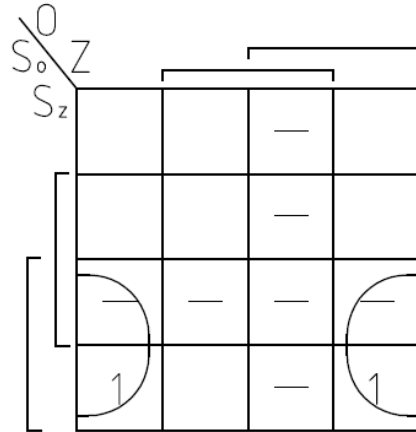


Obr.5.24 Karnaughova mapa pro Set

Minimalizovaný výraz:

$$S = OS_z$$

Karnaughova mapa pro reset R:



Obr.5.25 Karnaughova mapa pro reset

Minimalizovaný výraz:

$$R = \bar{Z}S_0$$

Pravdivostní tabulka pro pohon Pz:

Stav č.	O	Z	S ₀	S _z	P _z	R	S
1	0	0	0	1	0	0	0
2	1	0	0	1	0	0	0
3	0	0	0	0	0	0	0
4	0	0	1	0	0	0	0
5	0	1	1	0	1	0	1
6	0	0	0	0	1	0	0
1	0	0	0	1	0	1	0
7	0	1	0	1	0	1	0

Obr.5.26 pravdivostní tabulka pro pohon Pz

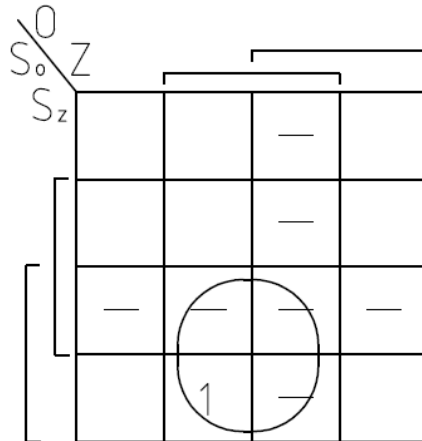
Algebraický výraz pro Set S:

$$S = \bar{O}ZS_0\bar{S}_z$$

Algebraický výraz pro Reset R:

$$R = \bar{O}ZS_0\bar{S}_z + \bar{O}ZS_0S_z$$

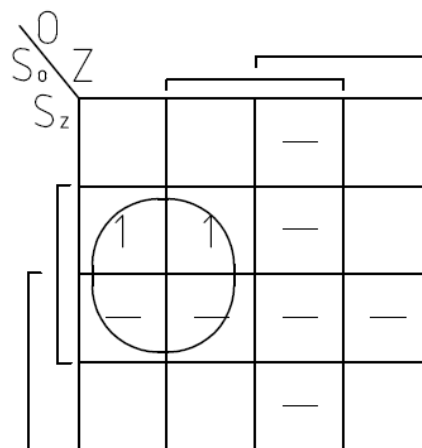
Karnaughova mapa pro set S:



Obr.5.27 Karnaughova mapa pro set

Minimalizovaný výraz:

$$S = ZS_0$$

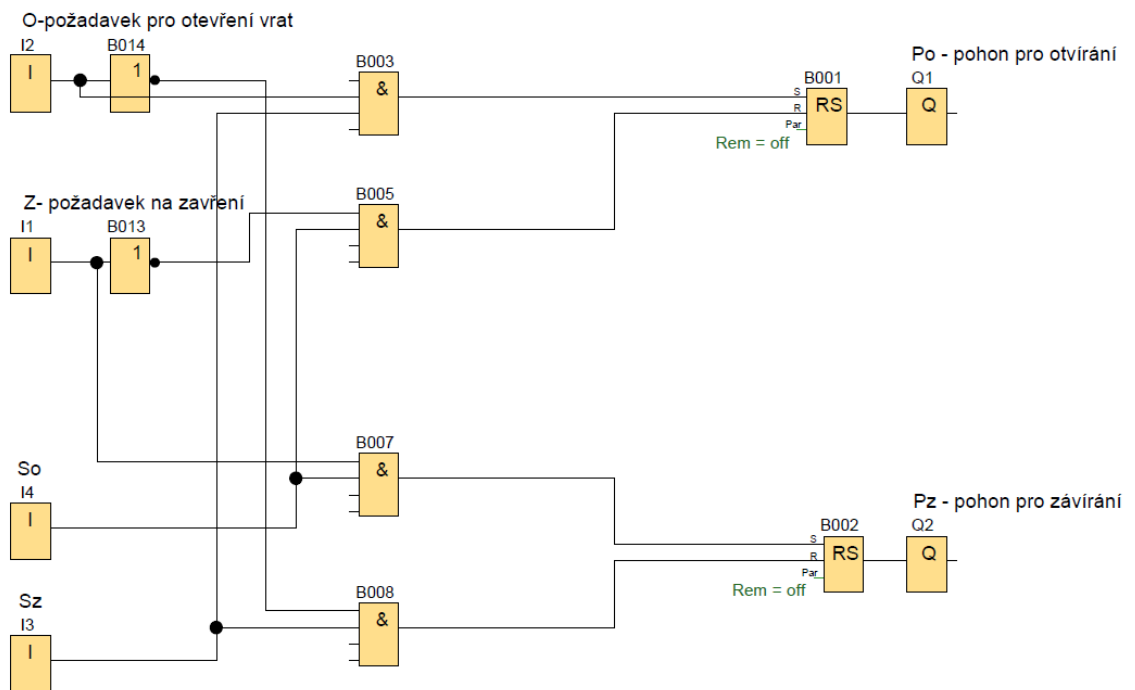


Obr.5.28 Karnaughova mapa pro reset

Minimalizovaný výraz:

$$R = \overline{O}S_z$$

Realizace v LOGO!Soft Comfort pomocí KO RS:



Obr.5.29 Realizace řízení otvírání/zavírání garážových vrat

5.2.2 Plnění kartónových krabic

Zadání

Proces plnění kartónových krabic lahvemi je řízen dvěma binárními signály Y_p , Y_z . O stavu procesu informují signály A,B,C,D. Signály A, B o umístění krabice na pásu. Signály C,D informují o stavu zaplnění krabic. V případě, že není žádná krabice připravena k plnění robot se vypne vypnutý.

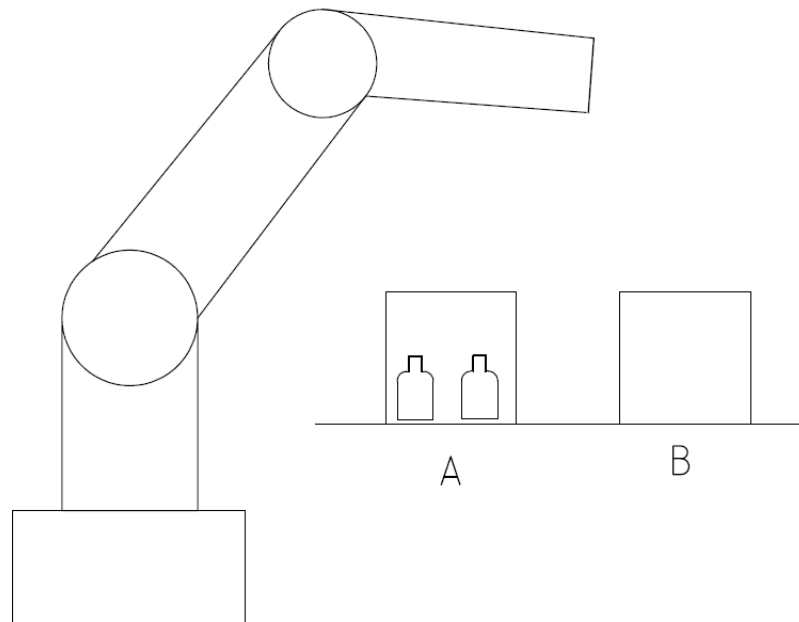
Vstupy:

A,B =1... krabice připravena
=0 ... krabice nepřipravena

C,D=1 ...kartónová krabice je plná
=0není plná

Výstupy: $Y_p = 1$ plnění krabice B
= 0 plnění krabice A
 $Y_z = 1$ zapnutý robot
= 0 vypnutý robot

Úkol: Realizujte jako klopný obvod RS v LOGO! Soft Comfort a správnou činnost ověřte na PA.



Obr.5.30 Plnění kartónových krabic

Řešení

Pravdivostní tabulka

	A	B	C	D	Y_P	Y_Z	
1	0	0	0	0	0	0	Krabice A ani B není připravena, robot je vypnutý
2	1	0	0	0	0	1	Krabice A se plní
3	1	1	0	0	0	1	Krabice A se plní, Krabice B je připravena
4	1	1	1	0	1	1	Krabice A naplněna, plní se krabice B
5	0	1	0	0	1	1	Odebrána krabice A, krabice B se plní
6	1	1	0	0	1	1	Krabice A připravena, Krabice B se plní
7	1	1	0	1	0	1	Krabice A se plní, krabice B naplněna
8	1	0	0	0	0	1	Krabice A se plní, krabice B odebrána
9	1	0	1	0	0	0	Krabice A naplněna, B odebrána, robot je vypnutý
10	1	1	1	0	1	1	Krabice A naplněna, plní se B
11	1	1	1	1	1	0	Krabice A i B naplněna, robot je vypnutý
12	0	1	0	1	1	0	Krabice A odstraněna, B krabice plná, robot vypnutý
13	1	1	0	1	0	1	Plní se krabice A, B naplněna
14	1	1	1	1	0	0	Krabice A i B naplněna

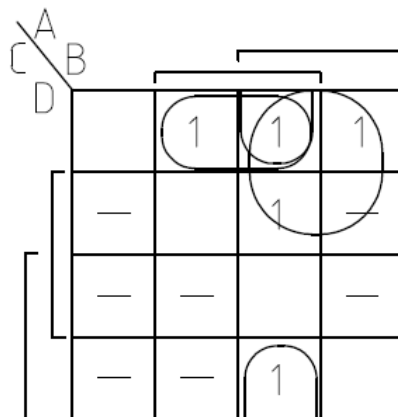
Obr.5.31 Pravdivostní tabulka – Plnění kartónových krabic

Z pravdivostní tabulky plyne, že Y_P je sekvenční logická funkce a Y_Z kombinační logická funkce.

Algebraické rovnice pro Y_Z :

$$Y_z = \overline{A}\overline{B}\overline{C}\overline{D} + \overline{A}B\overline{C}\overline{D} + \overline{A}B\overline{C}D + \overline{A}BC\overline{D} + \overline{A}BCD$$

Pro minimalizaci využijeme stavy, které nemohou nastat.

Obr.5.32 Karnaughova mapa pro Y_z

Získaný minimalizovaný tvar z mapy:

$$Y_z = \overline{B}\overline{C}\overline{D} + \overline{A}\overline{C} + \overline{A}B\overline{D}$$

Řešení pro ovládání plnění Y_p

A	B	C	D	Y_p	S	R
0	0	0	0	0	0	0
1	0	0	0	0	0	0
1	1	0	0	0	0	0
1	1	1	0	1	1	0
0	1	0	0	1	0	0
1	1	0	0	1	0	0
1	1	0	1	0	0	1
1	0	0	0	0	0	0
1	0	1	0	0	0	0
1	1	1	0	1	1	0
1	1	1	1	1	0	0
0	1	0	1	1	0	0
1	1	0	1	0	0	1
1	1	1	1	0	0	0

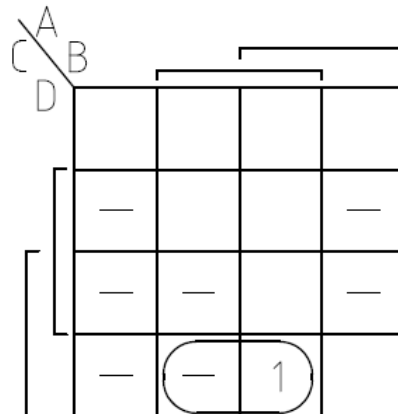
Obr.5.33 Pravdivostní tabulka pro plnění

Algebraické rovnice:

$$S = ABC\bar{D}$$

$$R = ABCD$$

Minimalizace set S:

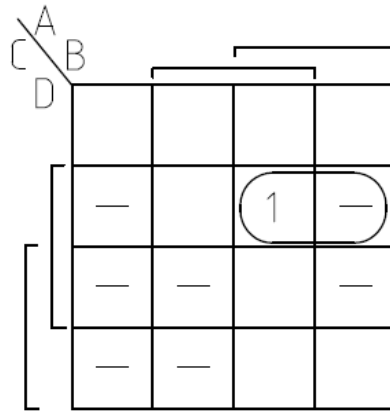


Obr.5.34 Karnaughova mapa pro Set

Získaný minimalizovaný tvar:

$$S = BC\bar{D}$$

Minimalizace Reset:

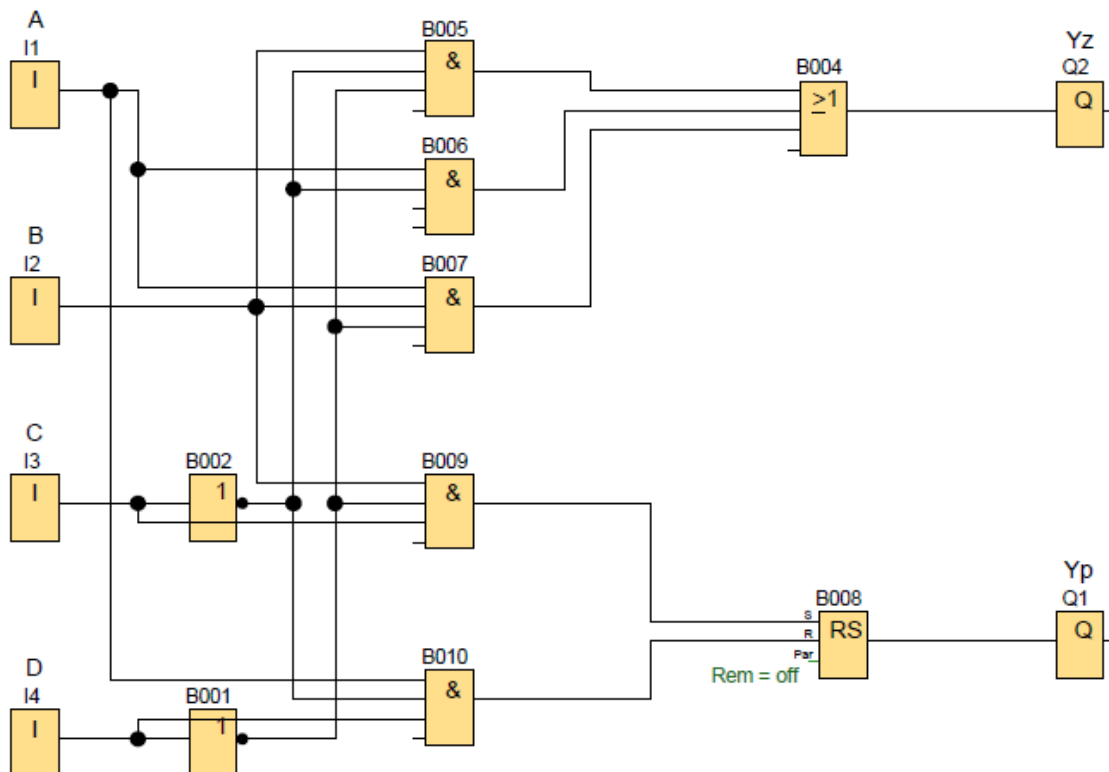


Obr.5.35 Karnaughova mapa pro Reset

Získaný minimalizovaný tvar:

$$R = \overline{ACD}$$

Realizace v Logo!Soft Comfort



Obr.5.36 Realizace plnění kartónových krabic

5.2.3 Řízení pískování odlitku

Zadání

Na odlitku je potřeba provést pískování dvou ramen. Navrhněte řízení pro pohyb stolu a pro řízení pískovací trysky. V poloze levého spínače S_L dojde k pískování levého ramena odlitku a v poloze spínače S_P dojde k zastavení posuvu a k pískování pravého ramena. Při pískování levého nebo pravého ramene odlitku nesmí být spuštěný posuv stolu.

Vstupy:

$S_L, S_P = 1$ stůl je v poloze spínače S_L resp. S_P

$Z=1$ – spuštění posuvu stolu

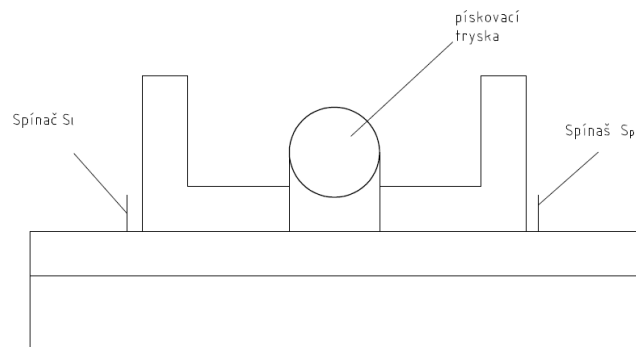
$Z=0$ vypnutí posuvu stolu

$P=1$ spuštění pískování

Výstupy:

$M_L, M_P = 1$ posuv stolu doleva nebo doprava

$T=1$ pískování pravého nebo levého ramena odlitku



Obr.5.37 Pískování odlitku

Pravdivostní tabulka sestavená podle sledu událostí:

S_L	S_P	Z	P	M_L	M_P	T	
1	0	0	0	0	0	0	Tryska je v poloze spínače S_L
1	0	0	1	0	0	1	Spuštění pískování levého ramena odlitku
1	0	1	0	0	1	0	Ukončení pískování a spuštění posuvu doprava
0	0	1	0	0	1	0	Pohyb posuvu doprava
0	1	0	0	0	0	0	Dosažení spínače S_P a vypnutí posuvu
0	1	0	1	0	0	1	Spuštění pískování pravého ramena
0	1	1	0	1	0	0	Ukončení pískování a spuštění posuvu doleva
0	0	1	0	1	0	0	Pohyb posuvu doleva
1	0	0	0	0	0	0	Cyklus se bude opakovat po výměně odlitku

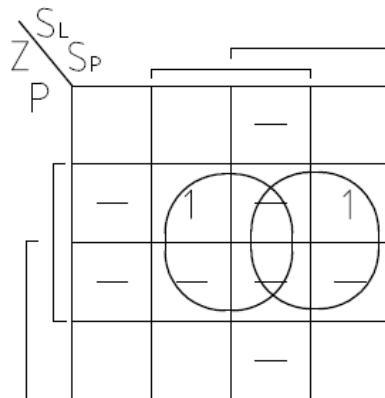
Obr.5.38 Pravdivostní tabulka – pískování odlitku

Řešení ovládání pískovací trysky

Algebraická rovnice logické funkce pro ovládání trysky:

$$T = S_L \overline{S_P} \overline{Z} P + \overline{S_L} S_P \overline{Z} P$$

Minimalizace pomocí Karnaughovy mapy



Obr.5.39 Karnaughova mapa

Minimalizací získáváme:

$$T = S_L P + S_P P$$

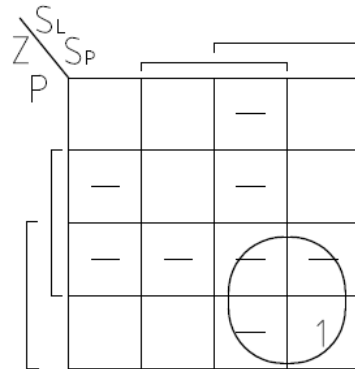
Řešení ovládání posuvu doprava:

S_L	S_P	Z	P	M_P	S	R
1	0	0	0	0	0	0
1	0	0	1	0	0	0
1	0	1	0	1	1	0
0	0	1	0	1	0	0
0	1	0	0	0	0	1
0	1	0	1	0	0	0
0	1	1	0	0	0	0
0	0	1	0	0	0	0
1	0	0	0	0	0	0

Obr.5.40 Pravdivostní tabulka pro ovládání posuvu doprava

Algebraická rovnice pro set S:

$$S = S_L \overline{S_P} \overline{Z} P$$



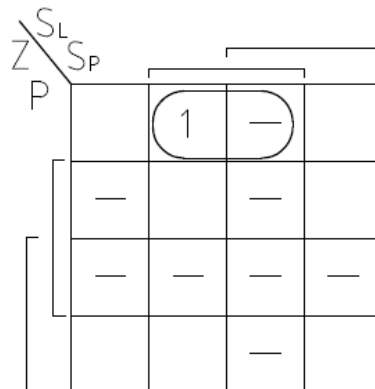
Obr.5.41 Karnaughova mapa pro Set

Z Karnaughovy mapy získáváme:

$$S = S_L Z$$

Algebraická rovnice pro reset:

$$R = \overline{S_L} S_P \overline{PZ}$$



Obr.5.42 Karnaughova mapa pro Reset

Z Karnaughovy mapy získáváme:

$$R = S_P \overline{PZ}$$

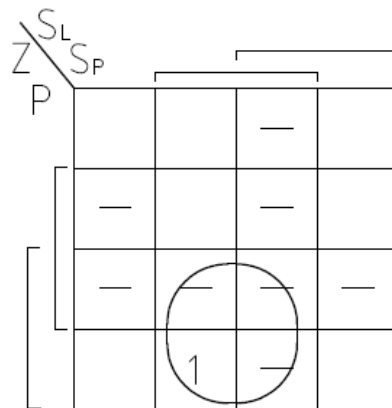
Řešení ovládání posuvu doleva

S_L	S_P	Z	P	M_L	S	R
1	0	0	0	0	0	0
1	0	0	1	0	0	0
1	0	1	0	1	0	0
0	0	1	0	1	0	0
0	1	0	0	0	0	0
0	1	0	1	0	0	0
0	1	1	0	0	1	0
0	0	1	0	0	0	0
1	0	0	0	0	0	1

Obr.5.43 Pravdivostní tabulka pro ovládání posuvu doleva

Algebraická rovnice pro set S:

$$S = \overline{S_L} S_P Z \overline{P}$$



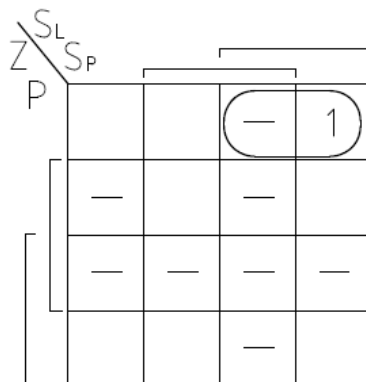
Obr.5.44 Karnaughova mapa pro Set

Z Karnaughovy mapy získáváme:

$$S = S_P Z$$

Algebraická rovnice pro reset:

$$R = S_L \overline{S_P} \overline{P Z}$$

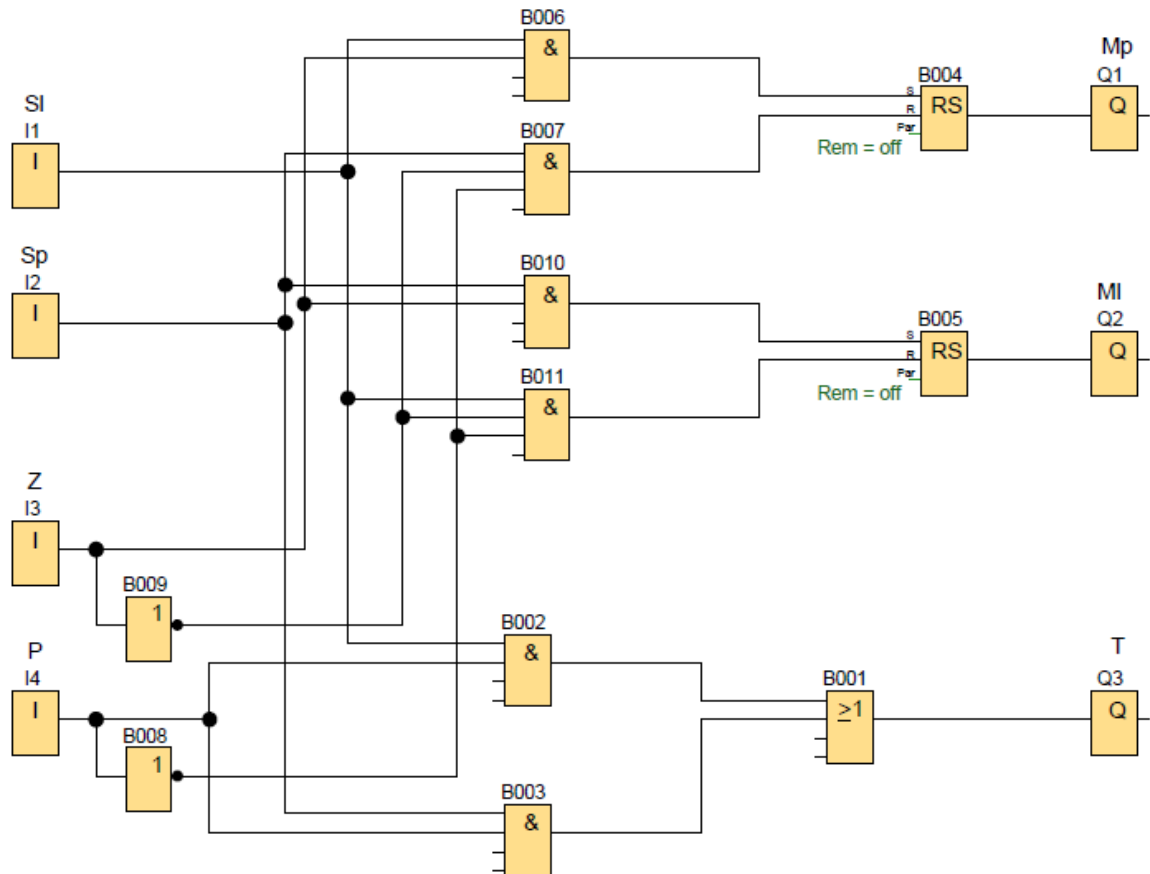


Obr.5.45 Karnaughova mapa pro Reset

Z Karnaughovy mapy získáváme:

$$R = S_L \overline{PZ}$$

Realizace v LOGO!SoftComfort



Obr.5.46 Realizace pískování odlitku

6 Závěr

Hlavním cílem této bakalářské práce bylo seznámit se s vývojovým prostředím LogoSoft! Comfort. Provést krátký popis tohoto vývojového prostředí a následně navrhnout ukázkové úlohy, které budou využity ve výuce automatizace.

Ve druhé kapitole byly popsány formy popisu logické funkce, bylo ukázáno zapojení klopného obvodu RS v LogoSoft! Comfort, tohle zapojení bude použito v navržených sekvenčních úlohách.

Třetí kapitola se věnuje popisu vývojového prostředí již zmíněného LogoSoft! Comfort. Tahle kapitola zefektivní práci ve vývojovém prostředí. Najdeme zde popis důležitých panelů, popis tvorby programu a vytvoření objektu.

Ve čtvrté kapitole je popsán programovatelný automat LOGO! 0BA5, na kterém lze realizovat logické obvody vytvořené ve výše zmíněném prostředí. Je uveden popis PA, kde se nachází jednotlivé ovládací prvky.

Předposlední kapitola se věnuje návrhu ukázkových úloh. Byly navrženy tři ukázkové úlohy na kombinační logický obvod a tři úlohy na sekvenční logický obvod. V úlohách je vždy popsáno zadání, řešení úlohy a realizace logického obvodu v LogoSoft! Comfort.

Cílem první úlohy je vytvoření tří bitové binární sčítačky. Prvním krokem je návrh polosčítačky a druhým úplné sčítačky. Posledním krokem je realizace samotné tří bitové sčítačky pomocí jedné polosčítačky a dvou úplných sčítaček. Funkce sčítačky byla ověřena na programovatelném automatu.

V další úloze je cílem návrh signalizace pro informaci o stavu úniku kapalin. Při úniku jedné kapaliny má svítit oranžové světlo, při úniku dvou kapalin červené a při úniku tří kapalin červené + spuštění algebr. Tahle úloha byla realizovaná prvky Booleovy algebry, prvky NOR a NAND. Pomocí De Morganových zákonů byla základní logické funkce převedena na NOR a NAND algebru.

U poslední navržené úlohy na kombinační logický obvod je cílem převést 4-místný binární kód na Grayův kód. Z pravdivostní tabulky byly vytvořeny logické funkce pro jednotlivé bity a následně jejich minimalizace. Byla ověřena správná činnost logického obvodu.

Úkolem první úlohy na sekvenční logický obvod je navrhnout ovládání otvírání a zavírání dveří. V pravdivostní tabulce, sestavené dle sledu událostí, byly pro stejné vstupy různé výstupní stavy, proto se jedná o sekvenční úlohu. Řešení bylo provedeno pomocí dvou klopných obvodů RS, jeden obvod realizuje otvírání a druhý zavírání vrat.

Cílem další úlohy je navrhnout řešení pro proces plnění kartónových krabic. Proces je řízen dvěma výstupními binárními signály a o stavu procesu informují čtyři signály. Jeden výstupní signál byl řešený jako kombinační logický obvod a druhý pomocí klopného obvodu RS.

Úkolem poslední úlohy je navrhnout řízení pískování dvou ramen odlitku. Bylo provedeno návrh řízení posuvu stolu a řízení pískování odlitku. Pro řízení posuvu byl použit RS klopný obvod, řízení pískování bylo vyřešeno jako kombinační logický obvod.

Funkce všech navržených úloh byla ověřena na programovatelném automatu.

Seznam použité literatury

- [1] Švarc, Ivan. Automatizace : automatické řízení. 2. vyd. Brno : Akademické nakladatelství CERM 2005. 262 s. SBN 80-214-2943-7
- [2] Manuál LOGO! 0BA6[online]. [cit. 12.únor 2010]. Dostupné z WWW: http://www1.siemens.cz/ad/current/index.php?ctxnh=7ef3b879ac&ctxp=doc_manualy.
- [3] Manuál LOGO! Soft Comfort V6.0 [online]. [cit. 12.únor 2010]. Dostupné z WWW: <http://www1.siemens.cz/ad/current/file.php?fh=d6aa88fd13&aid=1912875>.
- [4] Vybrané prvky a náhrady standardních prvků pro návrh sekvenčních obvodů pomocí LOGO[online]. [cit. 30.duben 2010]. Dostupné z WWW: <http://support.dce.felk.cvut.cz/lor/cviceni/LOGO/prvky.pdf>
- [5] Martinásková, Marie; Šmejkal, Ladislav. Řízení programovatelnými automaty. Praha: Vydavatelství ČVUT, 1998. 165 s. ISBN: 80-01-017664.
- [6] Švarc, Ivan; Šeda, Miloš; Vítěčková, Miluše. Automatické řízení. Brno: Akademické nakladatelství CERM, 2007. 324 s. ISBN: 978-80-214-3491-2.
- [7] Hofreiter, Milan. Příklady a návody z automatického řízení. Praha : Vydavatelství ČVUT, 2002. 139 s. ISBN: 80-01-02447-4.
- [8] Zítek, Pavel; Hofreiter, Milan. Automatické řízení. Praha : Vydavatelství ČVUT, 2004. 148 s. ISBN: 80-01-03020-2.
- [9] Loskot, Roman; Valášek, Pavel. Logické obvody a kódy. Hradec Králové : Gaudeamus, 1997. 139 s. ISBN: 80-7041-961-X.

Seznam příloh

Příloha č.1 – Příložené medium CD-R obsahující:

- Tento dokument v elektronické podobě
- Zdrojové kódy řešených úloh