

2-ЗЕРТХАНАЛЫҚ ЖҰМЫС

TRACE MODE 6 ОРТАСЫНДА ЛОГИКАЛЫҚ ФУНКЦИЯЛАРДЫ ЖҮЗЕГЕ АСЫРУ

Жұмыстың мақсаты: Техно FBD бағдарламалау тілін қолдана отырып, Trace Mode 6 ортасында логикалық функцияларды орындау әдістемесін зерттеу және игеру.

Тапсырма: зертханалық сабақта оқытушы берген тапсырманы орындау. Тапсырмалар нұсқалары білім алушылардың сабаққа қатысу журналының реттік нөмірі бойынша беріледі.

1	$f = x_1x_2 \cup x_1\overline{x_3x_4} \cup \overline{x_1x_2x_3} \cup \overline{x_1x_4}$	13	$f = x_1x_2 \cup x_1x_2x_4 \cup \overline{x_1x_2x_3} \cup \overline{x_1x_4}$
2	$f = x_1\overline{x_4} \cup x_1\overline{x_2x_3} \cup \overline{x_1x_2} \cup \overline{x_1x_4}$	14	$f = x_1x_2 \cup x_1x_4 \cup x_1\overline{x_2x_3} \cup \overline{x_1x_2x_3x_4} \cup \overline{x_2x_3x_4}$
3	$f = \overline{x_1x_2x_3} \cup \overline{x_1x_2x_3} \cup x_4$	15	$f = \overline{x_1x_2} \cup x_1x_2x_4 \cup \overline{x_1x_2} \cup \overline{x_2x_4} \cup \overline{x_3x_4}$
4	$f = \overline{x_1x_2} \cup \overline{x_1x_4} \cup x_2\overline{x_3} \cup \overline{x_2x_4}$	16	$f = x_1\overline{x_2x_3} \cup \overline{x_1x_2} \cup \overline{x_1x_3} \cup \overline{x_1x_4} \cup x_2x_3x_4$
5	$f = x_1\overline{x_2} \cup \overline{x_1x_2} \cup \overline{x_1x_4} \cup \overline{x_2x_4}$	17	$f = x_1x_2 \cup x_1\overline{x_2x_4} \cup \overline{x_1x_2x_4} \cup x_2x_3 \cup x_3x_4$
6	$f = x_1x_4 \cup x_1x_2x_3 \cup \overline{x_1x_2x_4} \cup \overline{x_1x_4} \cup \overline{x_2x_3}$	18	$f = \overline{x_1x_2} \cup \overline{x_1x_2} \cup \overline{x_1x_4} \cup \overline{x_2x_4} \cup \overline{x_2x_3x_4}$
7	$f = x_1x_2 \cup x_1\overline{x_3} \cup \overline{x_1x_2x_3} \cup x_2\overline{x_4} \cup x_3\overline{x_4}$	19	$f = x_1x_2 \cup \overline{x_1x_2x_3} \cup \overline{x_2x_3x_4}$
8	$f = x_1x_2x_4 \cup x_1\overline{x_2x_3} \cup \overline{x_1x_3} \cup x_2x_3 \cup \overline{x_1x_2x_4} \cup x_3x_4$	20	$f = x_1x_2x_4 \cup x_1x_2x_4 \cup \overline{x_1x_4} \cup \overline{x_1x_2x_3} \cup \overline{x_2x_3x_4}$
9	$f = x_1x_2x_3 \cup x_1x_2\overline{x_4} \cup \overline{x_1x_2x_4} \cup \overline{x_2x_3x_4} \cup \overline{x_1x_4} \cup \overline{x_1x_2x_3}$	21	$f = x_1x_2 \cup \overline{x_1x_2x_3} \cup \overline{x_1x_2x_4} \cup \overline{x_2x_4}$
10	$f = x_1x_2x_4 \cup x_1x_2x_4 \cup \overline{x_1x_2x_3} \cup \overline{x_1x_2x_4} \cup \overline{x_2x_3x_4}$	22	$f = \overline{x_1x_2} \cup \overline{x_1x_4} \cup x_2x_3 \cup \overline{x_2x_3x_4}$
11	$f = x_1x_2x_4 \cup x_1x_2x_4 \cup \overline{x_1x_2x_3} \cup \overline{x_1x_2x_4} \cup \overline{x_2x_3x_4} \cup \overline{x_2x_3x_4}$	23	$f = x_1x_2x_3 \cup \overline{x_1x_2} \cup \overline{x_1x_4} \cup \overline{x_2x_3x_4} \cup x_3\overline{x_4}$
12	$f = x_1\overline{x_2x_3} \cup \overline{x_1x_2x_3} \cup x_1\overline{x_2x_3} \cup \overline{x_1x_2x_3} \cup x_3x_4 \cup \overline{x_3x_4}$	24	$f = x_1\overline{x_2x_3} \cup x_1x_2\overline{x_4} \cup \overline{x_1x_4} \cup \overline{x_2x_3}$
		25	$f = x_1x_2\overline{x_4} \cup \overline{x_1x_2} \cup \overline{x_1x_3} \cup \overline{x_2x_4}$

1 ҚЫСҚАША ТЕОРИЯЛЫҚ БӨЛІМ

1.1 логика алгебрасы

Аналогтық электронды құрылғылардан айырмашылығы, сандық құрылғыларда (DS) кіріс және шығыс сигналдары шектеулі күйлерді қабылдай алады. Логикалық келісімге (MEMCT 2.743-82) сәйкес Зу элементтерінің нақты физикалық іске асырылуына байланысты физикалық шаманың

неғұрлым оң мәніне, "Н" - деңгей "логикалық 1" күйіне, ал "L - деңгей" - "логикалық 0" оң мәніне азырақ сәйкес келеді. Мұндай келісім оң логика деп аталады.

Кері қатынас теріс логика деп аталады. МЕМСТ ' е 19480-89-да негізгі параметрлердің атаулары, анықтамалары мен шартты белгілері және цифрлық микросхемалардың сипаты берілген.

ЦУ дизайнының теориялық негізі логикалық айнымалылармен жұмыс істейтін алгебра логикасы немесе логикалық Аль-гебра болып табылады. Тек екі мәнді қабылдайтын логикалық айнымалылар үшін.

4 негізгі операция бар (кесте 2.1):

- логикалық "және" (AND) конъюнкция немесе логикалық көбейту операциясы * немесе \wedge ;

- логикалық "немесе" (OR) операциясы, ажырату немесе логикалық қосу, + немесе \vee ;

- логикалық "емес" (NOT) операциясы, мәнді өзгерту, инверсия немесе теріске шығару логикалық өрнектің үстіндегі сызықпен белгіленеді. Инверсия кейде мәтінде " ~ " белгісімен көрсетіледі;

- эквиваленттік операция " = " деп белгіленеді . Келесі қатынастар аксиомалар болып табылады.

(1)	$0 + 0 = 0$	$1 * 1 = 1$	(1')
(2)	$1 + 1 = 1$	$0 * 0 = 0$	(2')
(3)	$1 + 0 = 0 + 1 = 1$	$0 * 1 = 1 * 0 = 0$	(3')
(4)	$\sim 1 = 0$	$\sim 0 = 1$	(4')

(1, 2) және (1', 2') мыналардан тұрады: $x + x = x$ және $x * x = x$. (5)

(1, 3) және (2', 3') мыналардан тұрады: $x + 0 = x$ және $0 * x = 0$. (6)

(2, 3) және (1', 3') мыналардан тұрады: $1 + x = 1$ және $x * 1 = x$. (7)

(3) және (3') мыналардан тұрады: $x + \sim x = 1$ және $\sim x * x = 0$. (8)

(4) және (4') мынадай: $\sim(\sim x) = x$. (9)

Және соңында (1,1'), (2,2'), (3,3') және (4,4') :

$$\sim (x_0 + x_1) = \sim x_0 * \sim x_1 \text{ және } \sim (x_0 * x_1) = \sim x_0 + \sim x_1. (10)$$

Соңғы өрнектер (10) қосарлылық принципі немесе Де Морган теоремасы деп аталады (логикалық қосындының инверсиясы инверсияның логикалық көбейтіндісіне тең және керісінше). Р айнымалылар үшін қосарлылық қатынасы, көбінесе келесі түрде жазылады:

$$\sim (x_1 + .. + x_n) = \sim x_1 * .. * \sim x_n \text{ және } \sim (x_1 * .. * x_n) = \sim x_1 + .. + \sim x_n. (11)$$

1.2 ауыстыру функциялары

N-логикалық айнымалылар (дәлелдер) үшін олардың 2^n комбинациясы немесе екілік жиынтығы бар. Әрбір осындай жиынтықта 0 немесе 1 функциясының мәні анықталуы мүмкін. Егер функцияның мәндері кем дегенде бір жиынтықта өзгеше болса, функциялар әртүрлі болады.

N аргументтерінің ауысу функциясының жалпы саны (PF) $N = 2^{2^n}$. үшін $N = 2$, $N = 16$. $N = 3$, $N = 256$ және одан кейін өте тез өседі. 2 айнымалыдан 16 функция практикалық мәнге ие, өйткені кез-келген күрделі өрнекті протозоидтардың құрамы ретінде қарастыруға болады. 2.2-кестеде $n = 2$ үшін кейбір ҚҚ келтірілген. Мұнда i -X1 және x_0 кіріс айнымалыларының жиынтығы.

Кесте 2.2. $N = 2$ үшін ауысу функциялары

i	0	1	2	3	ОБОЗНАЧЕНИЕ ФУНКЦИИ
x_0 x_1	0 0	1 0	0 1	1 1	
F0	0	0	0	0	F0 = 0, константа "0"
F1	0	0	0	1	F1 = $x_1 * x_0$, "И"
F6	0	1	1	0	F6 = $x_1 (+) x_0$, "ИСКЛЮЧАЮЩЕЕ ИЛИ"
F7	0	1	1	1	F7 = $x_1 + x_0$, "ИЛИ"
F8	1	0	0	0	F8 = $\sim(x_1 + x_0)$, "ИЛИ - НЕ"
F9	1	0	0	1	F9 = $\sim(x_1 (+) x_0)$, "ИСКЛ. ИЛИ - НЕ"
F12	1	1	0	0	F12 = $\sim x_1$, "НЕ"
F14	1	1	1	0	F14 = $\sim(x_1 * x_0)$, "И - НЕ"
F15	1	1	1	1	F15 = 1, константа "1"

Маңызды анықтамалар:

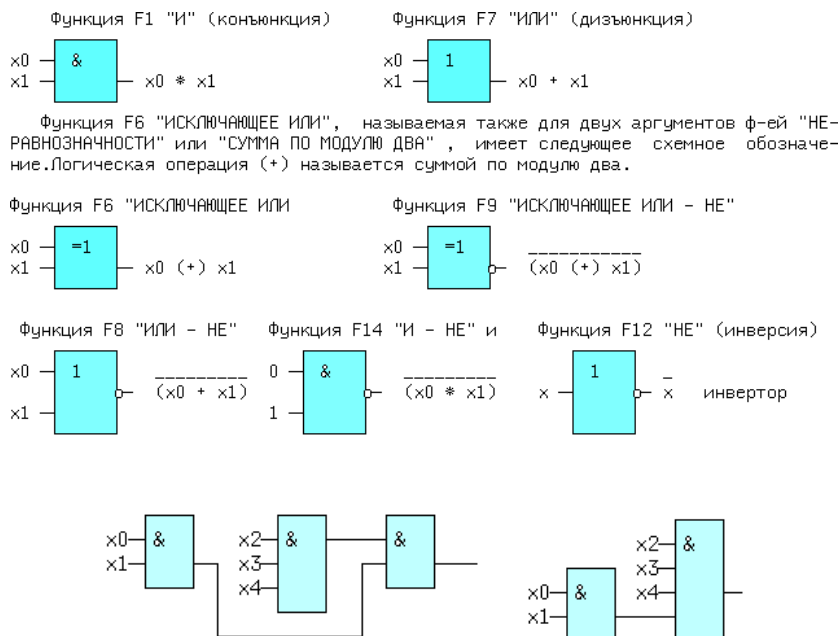
- 1 "және" функциясы, егер оның барлық дәлелдері бірлікке тең болса, бірлікке тең болады.
- 2 "немесе" функциясы, егер ол кем дегенде бір дәлелге тең болса, бірлікке тең.
- 3 "эсклюзивті НЕМЕСЕ" (XOR) функциясы, егер ол тек бір аргументке тең болса, бірлікке тең.

1.3 схемалардағы логикалық функциялардың шартты белгілері

Логикалық функцияны қалыптастыруға қатысатын логикалық элементтің кірістерінің саны біріктіру коэффициенті деп аталады-Коб (бөлу коэффициентімен шатастырмау керек). 2.1-суретте келтірілген барлық тізбектерде Инверторды қоспағанда, біріктіру коэффициенті екіге тең.

Өнеркәсіп $cb = 2, 3, 4, 8$ схемаларын шығарады. Кірістердің басқа саны бар тізбектерді алу үшін негізгі элементтерді біріктіруге болады.

Мысалы, егер 5 кіріс схемасы қажет болса, онда оны келесі жолмен алуға болады: $x_0 * x_1 * x_2 * x_3 * x_4 = (x_0 * x_1) * (x_2 * x_3 * x_4) = (x_0 * x_1) * x_2 * x_3 * x_4$, яғни екі 2 кіріс және бір 3 кіріс тізбегі қажет, ал бірінші нұсқа үшін немесе біреуі 2 кіріс және біреуі 4 кіріс-екінші (2.2-сурет).



1.4 логикалық функцияларды ұсыну әдістері

Сандық құрылғыны жобалаудың негізгі мақсаты оның логикалық функциясын (LF) және оған сәйкес схемалық іске асыруды алу болып табылады. LV ұсынудың әртүрлі формалары болуы мүмкін: 1) ауызша, 2) графикалық, 3) кестелік, 4) алгебралық, 5) алгоритмдік тілде (мысалы, VHDL) және 6) схемалық. Мысал ретінде, ауызша сипаттамамен берілген x_1 және x_0 екі айнымалыдан Y функциясын қарастырыңыз: $Y = 1$, Егер айнымалылар тең болмаса және $Y = 0$, егер $x_1 = x_0$ болса. Мұндай жаттығу терапиясы тең емес функция деп атауға ыңғайлы. Y кестелік көрінісіне өтіңіз (2.3-кесте):

$$Y = \sum_{i=0}^{2^n-1} (f_i * m_i). \tag{12}$$

i	0	1	2	3
x_0	0	1	0	1
x_1	0	0	1	1
Y	f_0	f_1	f_2	f_3
	0	1	1	0

Мысал ретінде екі айнымалыдан $F = f(x_1, x_0)$ функциясын қарастырыңыз. Бұл функцияның X_1 айнымалысы бойынша ыдырауы: $F = \sim x_1 * f(0, x_0) + x_1 * f(1, x_0)$. X_0 айнымалысы үшін осы әрекетті жалғастыра отырып, біз аламыз:

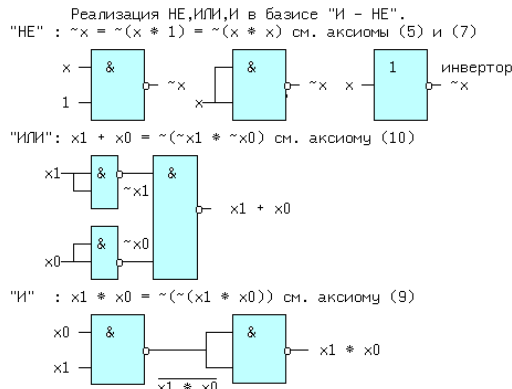
$$F = \sim x_1 * (\sim x_0 * (f(0, 0) + x_0 * (f(0, 1))) + x_1 * (\sim x_0 * (f(1, 0) + x_0 * (f(1, 1))))$$

$$= \sim x_1 * \sim x_0 * f(0, 0) + \sim x_1 * x_0 * f(0, 1) + x_1 * \sim x_0 * f(1, 0) + x_1 * x_0 * f(1, 1). \quad (13)$$

Өрнек (13) барлық ауысу функцияларын тек үш негізгі логикалық амалдарды қолдана отырып, екі айнымалыдан жазуға мүмкіндік береді.

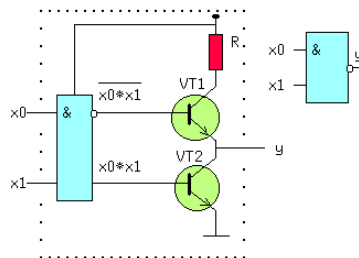
1.5 логикалық негіз

Кез-келген басқа функцияны жүзеге асыруға мүмкіндік беретін қарапайым жаттығу терапиясының жиынтығы логикалық негіз (lb) деп аталады. Функциялар және, немесе, ең аз маңдай болып табылмайды, өйткені оларды басқа функциялар арқылы ұсынуға болады (2.3-сурет), мысалы, F8(немесе - емес) немесе F14(және-ЕМЕС) арқылы.



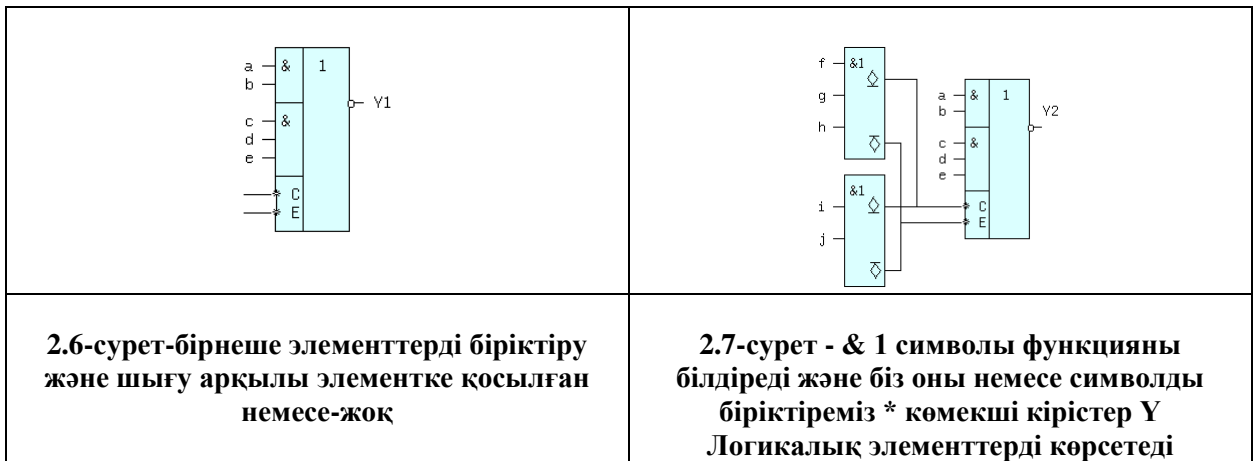
1.6 логикалық элементтердің схемалық ерекшеліктері

Жоғарыда келтірілген логикалық элементтер және, немесе, емес және басқалары кейбір схемалық сипаттамаларға ие болуы мүмкін. Негізгі логикалық элементті қарастырыңыз. 2.4-суретте жеңілдетілген схема және оның белгісі көрсетілген.



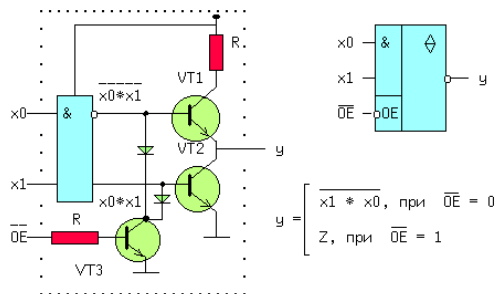
"Және – немесе - емес" элементтерінің схемалары және кеңейткіштер бірнеше элементтерді біріктіреді және элементке немесе-емес шығулармен қосылады (2.6-сурет). Егер A, b, ..., e айнымалыларының саны жеткіліксіз болса, C және E кеңейту кірістеріне қосылатын кеңейту элементтері қолданылады (ашық коллектор мен эмиттер үшін кірістер). & 1 символы функцияны немесе (2.7-сурет) арқылы біріктірілген функцияны білдіреді. Мұнда және одан әрі символ * логикалық элементтердің көмекші кірістерін көрсетеді.

Бұл схемаларда, элементтердегі сияқты, немесе пайдаланылмаған кіріс немесе 0-ге қосылуы керек. Сондықтан, егер бөлімдердің бірі қатыспаса, оның біреуі үшін 0 беру керек. Әйтпесе, Y әрқашан 0 болады. Бұл TTL(W) технологиясы бойынша жасалған схемалардың ерекшелігі, өйткені бұл схемалардың қосылмаған логикалық кірісі логикалық 1-ге тең.



Егер сигнал $\sim OE = 0$ болса, $vt3$ Транзисторы жабық және қарсы диодтар I элементінің логикалық шығуларына әсер етпейді. $Vt1$ және $VT2$ транзисторларының базаларындағы кернеулер антифазада болады, егер $x0 * x1 = 1$ болса, онда жоғарғы транзистор жабық, ал төменгі транзистор ашық. $Vt2$ коллекторының потенциалы шамамен нөлге тең, сондықтан $y = 0$.

Басқа $x0$ және $x1$ мәндерінде төменгі транзистор құлыпталады, ал жоғарғы жағы ашық және тізбектің шығуында - жоғары деңгей, яғни $\sim oe = 0$ кезінде тізбек қарапайым элемент ретінде жұмыс істейді және болмайды. Сурет $\sim OE = 1$ кезінде айтарлықтай өзгереді. $Vt3$ Транзисторы қаныққанға дейін ашылады және $Vt1$ және $VT2$ транзисторларының базасында потенциал оларды құлыптау арқылы нөлге дейін төмендейді. "Y" шығысы ішкі логикалық тізбектен ажыратылады. Схемаларда тристабильді элементтер көлденең сызығы бар ромбпен немесе Z әрпімен белгіленеді.



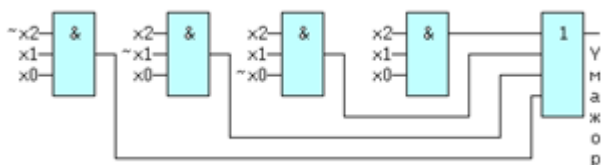
1.7 логикалық функцияларды азайту

(12) формула бойынша алынған өрнек түпнұсқамен салыстырғанда айнымалылар мен операциялардың аз саны бар түрге түрлендірілуі мүмкін (әрдайым емес). Мұндай түрлендіру минимизация деп аталады.

Мысалды қарастырайық. Үш екілік XI сенсоры бар. Екі немесе одан да көп сенсорлардың 1 мәні тең болған кезде LF Y major 1 мәнін енгізу керек. Бұл функция көпшілік деп аталады (сурет 2.9).

Номер i входного набора	Сигналы от датчиков			ЛФ Ymajor
	x2	x1	x0	
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1

(12) формуласына сәйкес жазамыз: $Y_{\text{major}} = \sim x_2 * x_1 * x_0 + x_2 * \sim x_1 * x_0 + x_2 * x_1 * \sim x_0 + x_2 * x_1 * x_0$ (3, 5, 6, 7 - кестенің 2.9 жолдары). Алынған өрнек 2.10-суреттегі схемаға сәйкес келеді.



Схемада "және" 4 үш жақты және "немесе" 1 төрт жақты элемент бар. ЕДШ-нің минималды формасын табу алгебралық түрлендіру әдісімен, Карнот кестелерін немесе үлкен жобаларға арналған машиналық әдістерді қолдана отырып жасалады.

1.8 Карнот Кестесі

Карнот кестесі (ТК) шындық кестесінің өзгертілген жазбасы болып табылады. Мажоритарлық функция үшін соңғы мысалдан (мысалы) 2.11-суретте көрсетілгендей көрінеді.

	$x_1 x_0$	"Үмажор"		
x_2	00	01	11	10
0	0	0	1 В	0
1	0	1 А	1	1 С

Карнот кестесін құру ережелерін ұсынамыз:

1 ТК жасушаларының саны ақиқат кестесінің жолдарының санына тең.

2 сөздер және одан тыс аргументтердің мағыналары қабылданады. Аргументтерді бөлу тәртібі мынада: көлденең және тігінен көршілес екі ұяшықта тек бір Аргументтің мәні ерекшеленеді (сондықтан кестенің оң жақ жиектерінде орналасқан жасушалар көрші болып саналады).

3 тиісті LF мәндері жасушаларда орналасады.

4 бірлік жасушалар 2 және жасушалардың тіктөртбұрыштарына (имплантаттарға) біріктіріледі.

5 кез-келген тіктөртбұрыш үшін көрші ұяшықтарда олардың мағынасын өзгертпейтін мәтіндік дәлелдердің көбейтіндісі жазылады.

6 өзгеріс жұмыста тікелей түрде жүреді, егер IX көрші жасушалардағы мән 1 болса, керісінше жағдайда керісінше.

7 алынған жұмыстар қажетті жаттығу терапиясы бойынша немесе жинақталған.

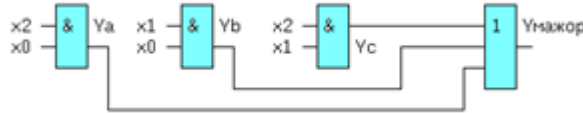
2.12 сурет үлгісінде 3 тіктөртбұрыш бар-А, Б, В, және:

- I = $x_2 * x_0$ (өйткені көрші ұяшықтағы X1 оның мәнін өзгертеді, сондықтан топқа кіруге болмайды);

- YP = $x_1 * x_0$;

- Yc = $x_2 * x_1$.


- $Y_{масар} = Y + YP + Yc = x_2 * x_0 + x_1 * x_0 + x_2 * x_1. (14)$

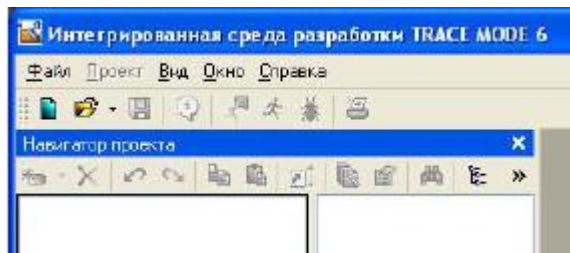


2 ЖҰМЫСТЫ ОРЫНДАУ ТӘРТІБІ

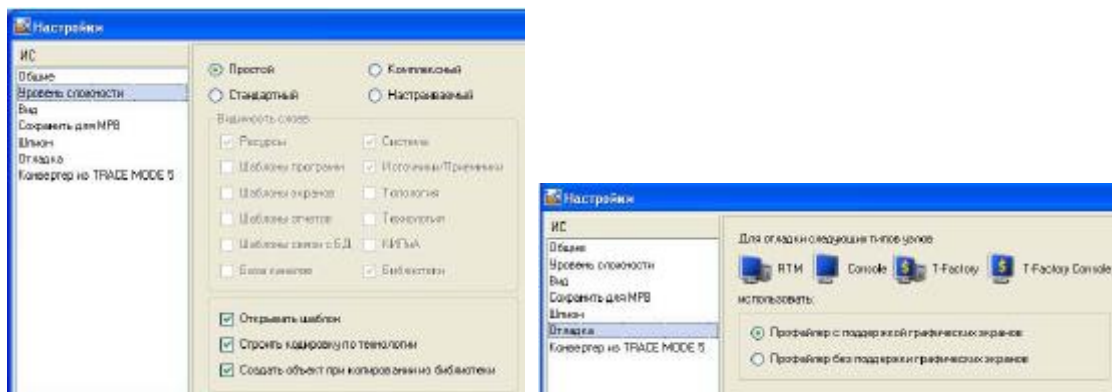
2.1 логикалық функцияны іске асыру әдістемесі

2.1.1 $f = x_1x_2x_4 \cup x_1x_2 \cup x_1x_3 \cup x_2x_4$ логикалық функцияны іске асыру әдістемесін сипаттаймыз: Trace Mode 6 ортасында техно FBD бағдарламалау тілін қолдана отырып.

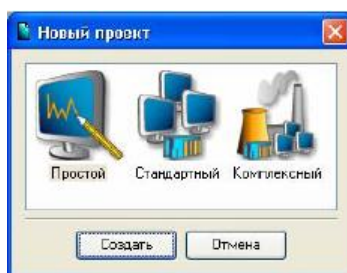
Trace Mode 6 ортасында, әдетте, жаңа автоматтандыру жобасын жасау әрқашан Trace Mode IDE 6 (base) пәрменін орындаудан басталады. Бұл пәрмен ОЖ-нің "бағдарламалар" мәзіріндегі аспаптық жүйені орнату тобында немесе ОЖ Жұмыс үстеліндегі белгішені  тінтуірдің сол жақ батырмасымен екі рет басу арқылы орналасқан (2.13-сурет).



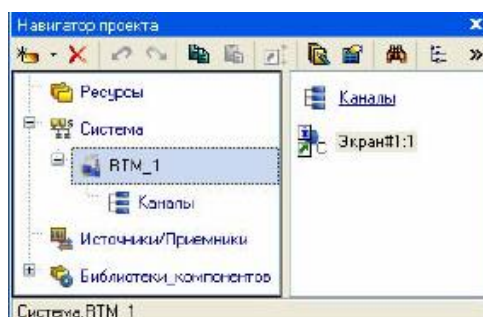
2.1.2 Файл мәзірінде ортаны іске қосқаннан кейін "IP параметрлері..." пәрменін таңдаңыз. Пайда болған терезеде "қиындық деңгейін" таңдап, оны 2.2-суретте көрсетілгендей сол жақта реттеңіз, содан кейін "күйін келтіруді" таңдаңыз және конфигурациялаңыз (2.14-сурет, оң жақта).



2.1.3 Trace Mode 6 ортасының параметрлерін орнатқаннан кейін Аяқтау түймесін басыңыз. Trace Mode 6 панелінің белгішесін пайдаланып, экранда ашылған диалогта жаңа жоба жасаңыз (2.15-сурет).



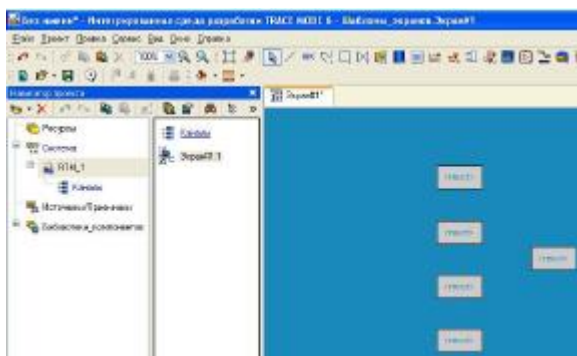
2.1.4 енді біз "қарапайым даму стилін" таңдаймыз. Экранның "жасау" батырмасында тінтуірдің сол жақ батырмасын (LK) басқаннан кейін, "жоба навигаторының" сол жақ терезесінде "ARM RTM_1" түйіні бар жоба ағашы пайда болады. "RTM_1" түйінін LK екі рет басу арқылы ашамыз, түйінінің мазмұны "жоба навигаторының" оң жақ терезесінде көрсетіледі - бос арналар тобы және графикалық экранның "ARM" түйінінде көрсетуге арналған "#1 экран шақыру" класының бір арнасы (2.16-сурет).



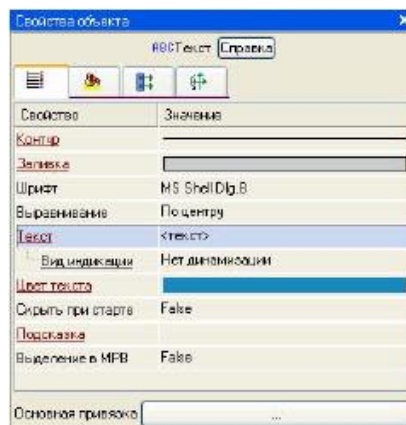
2.2 графикалық экран құру

2.2.1 "Экран#1" компонентіне екі рет басу арқылы графикалық редактор терезесін ашыңыз. Экранда X1, X2, X3, X4 кіріс айнымалыларының сандық мәнін және Y шығыс мәнін графикалық элементтің төлсіпатын (GE) динамизациялауды көрсету үшін динамикалық мәтін шығысын дайындаймыз.

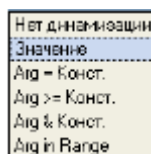
2.2.2 экран үлгісінің аргументінің мақсатын анықтаңыз. 2.17-суретте көрсетілгендей төрт ГЭ жасап, орналастырамыз.



2.2.3 "мәтін" жолында тышқанның сол жақ батырмасын екі рет басу арқылы "ақпарат түрі" мәзірін шақыру (2.18-сурет).



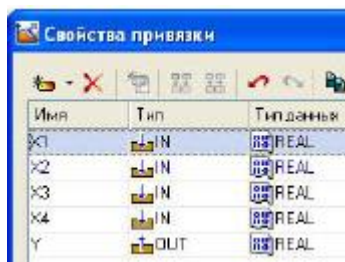
2.2.4 оң жақ жол өрісінде LK түймесін басып, қол жетімді типтердің тізімін шақырыңыз. "Мән" түрін таңдаңыз (2.19-сурет).



2.2.5 ашылған мәзірде динамизация параметрлерін баптау (2.20-сурет).



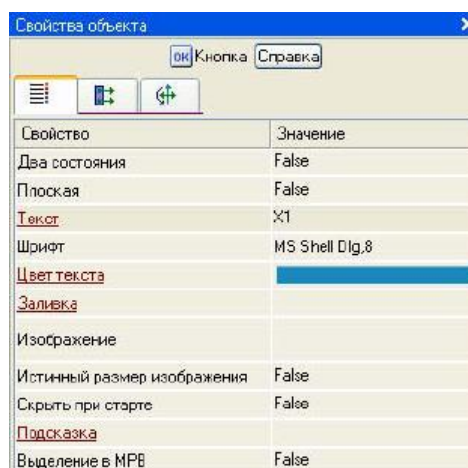
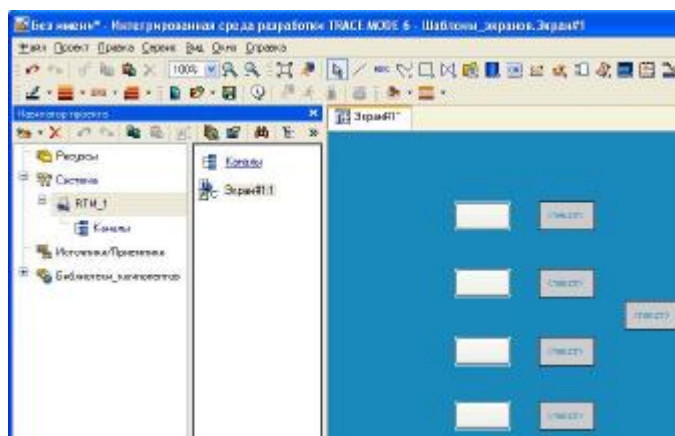
2.2.6 "байланыстыру" сипатын таңдаңыз. Ашылған "байланыстыру қасиеті" терезесінде оның құралдар тақтасындағы түймені басып, 4 кіріс және бір шығыс экранының бес дәлелін жасаңыз (2.21-сурет).



2.2.7 екі рет нұқу арқылы Аргументтің атын таңдап, оны "X1" пернесінен енгізу арқылы өзгертіңіз (Enter пернесін басу арқылы енгізуді аяқтаңыз). Осы дәлелмен байланысты растау-"Дайын" түймесін басу арқылы. Сол сияқты, барлық кіріс және шығыс дәлелдерін реттеңіз.

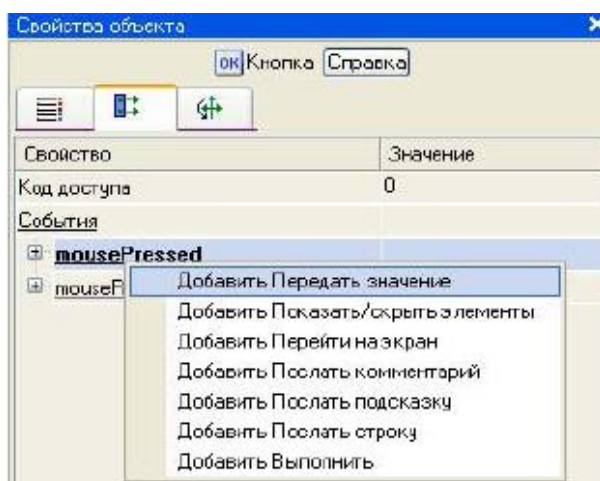
2.2.8 экран құрамына пернетақтадан сандық мәндерді енгізуге мүмкіндік беретін құрал енгізілді. Оларды қабылдау үшін жаңа экран үлгісінің дәлелін жасаңыз.

2.2.9 графикалық редактордың құралдар тақтасында GE батырмасы белгішесін таңдаңыз. Тінтуірдің көмегімен оны 2.22-суретте көрсетілгендей экран өрісіне орналастырыңыз.

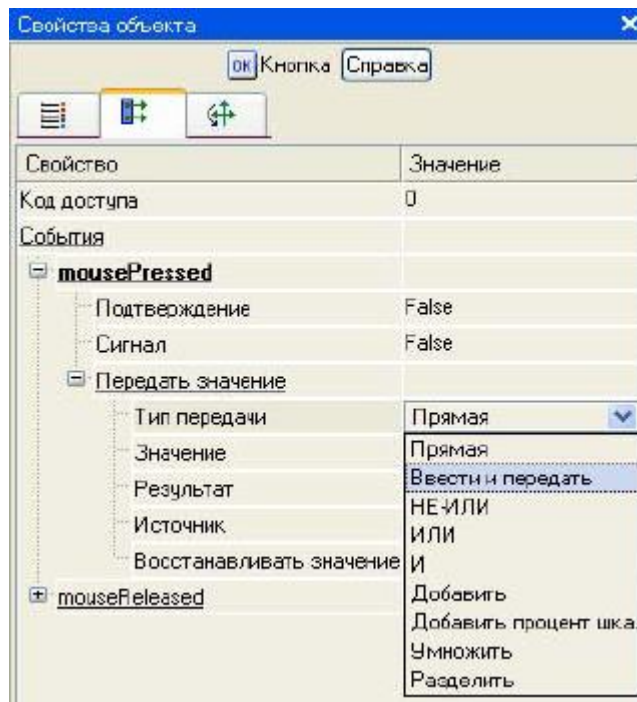


2.2.10 бірінші батырмамен өңдеу режиміне өтіп, графикалық элементтің қасиеттер терезесін шақырыңыз (2.23-сурет).

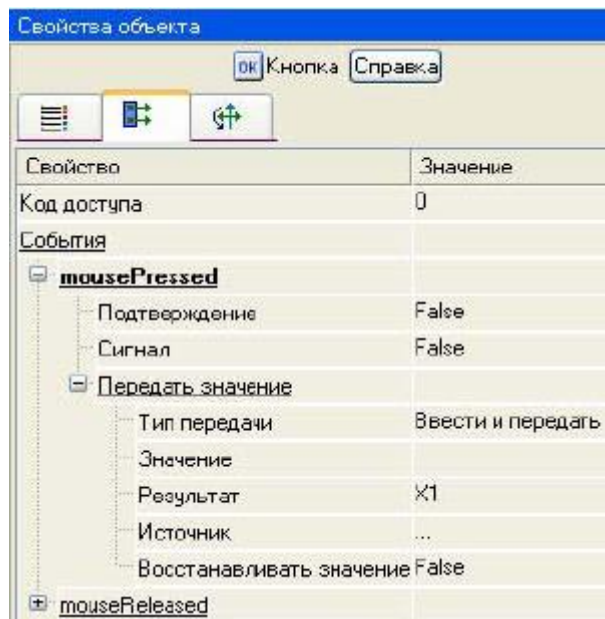
2.2.11 мәтін жолында "X1" енгізіңіз. "Оқиғалар" бланкісін ашып, "басу арқылы (pressed)" мәзірін ашыңыз. Тізімнен "мән беруді қосу" пәрменін таңдаңыз және таңдалған команданың Параметрлер мәзірін жабыңыз (2.24-сурет).



2.2.12 "беру түрі (Send Type)" өрісінде "енгізу және беру" тізімінен таңдау (Enter & Send)" (2.25-сурет).



2.2.13 LK "нәтиже" өрісінде аргументтердің кестелік редакторын шақырып, X1 аргументіне байланыстырыңыз. Байланыстырғаннан кейін "объектінің қасиеттері" терезесі 2.26-суретте көрсетілгендей көрінеді. Ұқсас түрде теңшеу қалған түймелер X2, X3, X4.



2.3 экран дәлелін арнаға байланыстыру

2.3.1 Жасаймыз бойынша аргументам X1, X2, X3, X4 және шығу Y үлгісін экранның жаңа арналарды және отредактируем оларды байлау. "Жүйе" қабатында "RTM_1" түйінін ашыңыз. ДК көмегімен мәтінмәндік мәзір арқылы "Экран#1" компонентінің қасиеттерін шақыру (2.27-сурет).

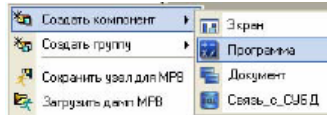
Имя	Тип	Тип данных	Значение по умолчанию	Примечание
X1	IN	REAL		X1:Реальное значение [Система.RTM_1]
X2	IN	REAL		X2:Реальное значение [Система.RTM_1]
X3	IN	REAL		X3:Реальное значение [Система.RTM_1]
X4	IN	REAL		X4:Реальное значение [Система.RTM_1]
Y	OUT	REAL		Y:Выходное значение [Система.RTM_1]

2.3.2 аргументтер койындысын таңдаңыз, LC X1, X2, X3, X4 және шығу Y аргументтерін таңдаңыз және белгішені пайдаланып жаңа арна жасаңыз. Нәтижесінде "RTM_1" торабында автоматты түрде арналар салынады (2.28-сурет).

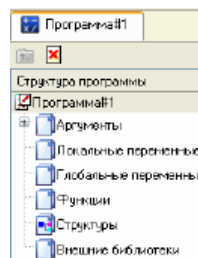


2.4 Fbd Техно тілінде бағдарлама құру

2.4.1 берілген логикалық функцияны жүзеге асыратын бағдарламаны жасаңыз. LK екі рет басу арқылы "RTM_1" түйінін ашып, ондағы "грамм" компонентін жасаңыз (2.29-сурет).



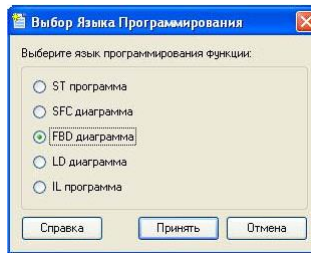
2.4.2 "№ 1 Бағдарлама" компонентін таңдаңыз және ДК "үлгіні өңдеу" тармағын таңдап, контекстік мәзірге қоңырау шалыңыз және бағдарламаларды өңдеу режиміне өтіңіз (2.30-сурет).



2.4.3 "№ 1 Бағдарлама" шаблон ағашында "аргументтер кестелік аргумент редакторын X1, X2, X3, X4 және Y шығу 4 аргументін құру белгішесімен шақыру.

Имя	Тип	Тип данных
X1	IN	REAL
X2	IN	REAL
X3	IN	REAL
X4	IN	REAL
Y	OUT	REAL

2.4.4 шаблон ағашында "№ 1 Бағдарлама" жолын және ашылған "тіл таңдау" диалогында "FBD" тілін таңдаңыз (2.32-сурет).



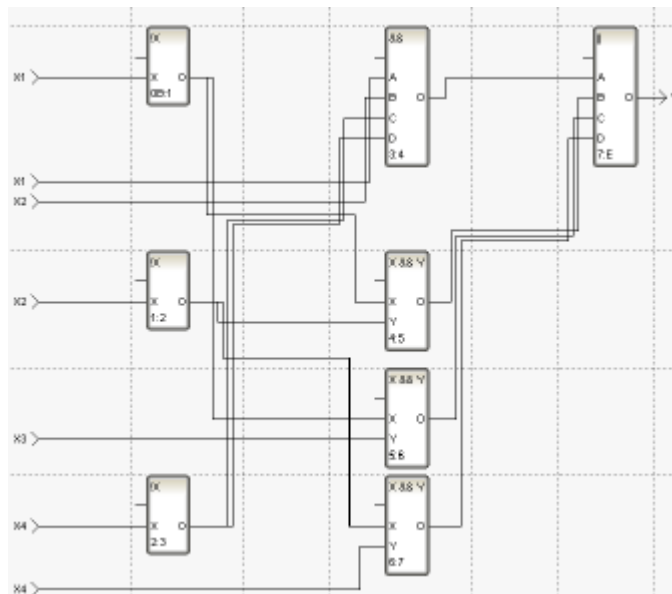
2.4.5 жарияланған айнымалысы бар бағдарлама редакторының ашылған терезесінде "қабылдау" экранының батырмасын басу арқылы тапсырмаға сәйкес бағдарлама жасаңыз.

Fbd блоктарының палитрасын тандау үшін батырманы басу керек, содан кейін FBD блоктарын таңдау терезесі пайда болады (2.33-сурет). Бағдарламаны әзірлеу кезінде fbd блоктарының жоғарғы кірістері пайдаланылмайды, өйткені олар блоктарды қайта есептеу тәртібін өзгертуге арналған, ал ақпараттық кірістер екіншіден бастап кірістер болып табылады.

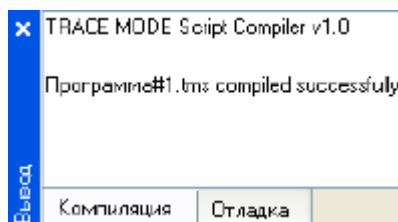


2.4.6 логикалық функцияны жүзеге асыру үшін келесі блоктарды таңдаймыз: "Io функциялары" бөлімінен "fbd блоктары" инверсия (!X), логикалық көбейту ($X \& \& Y$ және $\&\&$), логикалық қосу (\parallel).

Осы мысал үшін барлық блоктарды орналастырғаннан кейін бағдарлама 2.34-суретте көрсетілгендей болады.



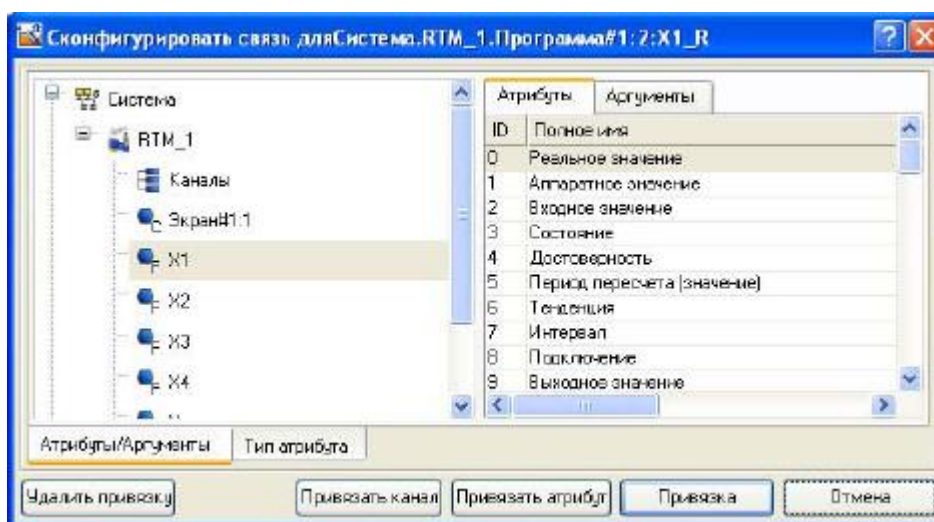
2.4.7 редактордың құралдар тақтасындағы белгішені немесе F7 "ыстық пернені" пайдаланып, бағдарламаны құрастырыңыз және белгішені пайдаланып құралдар тақтасынан шақырылған Шығыс терезесінде (шығу) сәтті құрастырылғанына көз жеткізіңіз.



2.5 бағдарламаның дәлелдерін байланыстыру

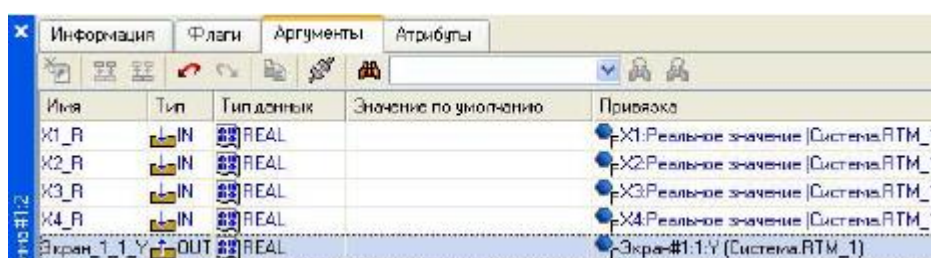
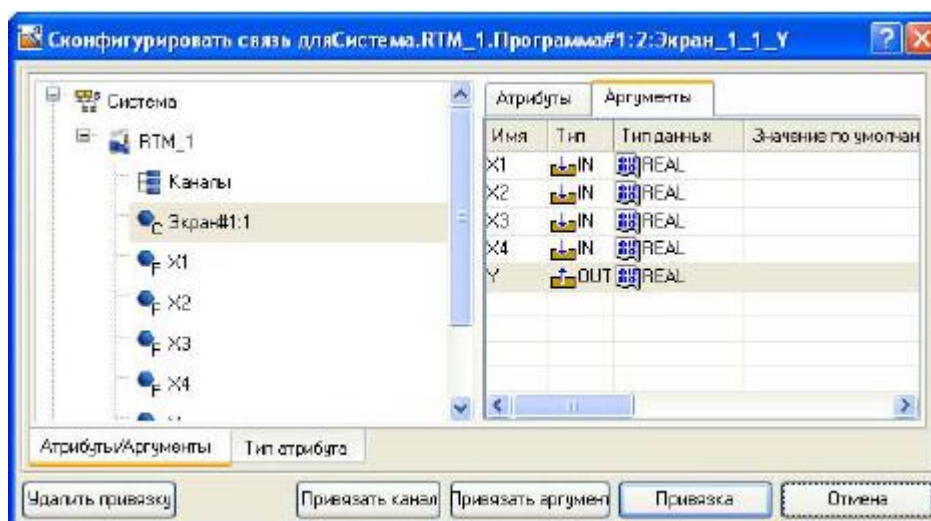
2.5.1 бағдарлама дәлелдерін арна атрибуттарына байланыстырамыз.

Мәтінмәндік мәзір арқылы "№ 1 Бағдарлама" компонентінің қасиеттерін шақырыңыз. "Аргументтер" қойындысын таңдаңыз. "Байланыстыру" өрісіне екі рет басу арқылы бағдарлама дәлелдерін арналардың атрибуттарына-X1, X2, X3, X4 дәлелдерін X1, X2, X3, X4 арналарының нақты мәніне байлаңыз (2.36 - сурет).




2.5.2 Y бағдарламасының "аргументті байланыстыру" өрісіне екі рет басу арқылы байланыс параметрлері терезесін шақырыңыз, сол жақ терезеде "қоңырау шалу" класының арнасын "Экран#1" таңдаңыз, ал оң жақ терезеде "аргументтер" қойындысын таңдап, Y дәлелін және "байланыстыру" түймесін басыңыз (2.37-сурет).

Нәтижесінде бізде 2.38-сурет болады. Осыдан кейін "№ 1 Бағдарлама" коммуналдық меншік терезесін жабыңыз.

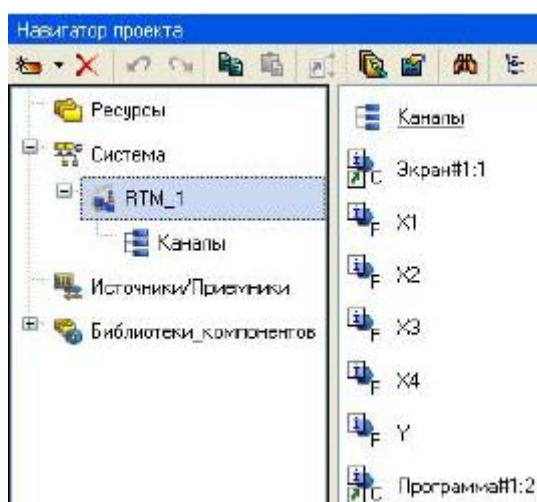


2.6 жобаны іске қосу

2.6.1  белгішесі арқылы жобаны сақтау .

Құралдар тақтасында пәрменді таңдап, жобаны нақты уақытта іске қосу үшін құрастырыңыз.

Жоба навигаторында "RTM_1" торабын таңдаңыз (2.39-сурет).



2.6.2 құралдар тақтасындағы белгішені таңдап, орындау режимін іске қосыңыз.

Есепті жасау үшін "Файл" мәзірінде "жобаны құжаттау" және алынған *пәрменін орындау қажет.html файлын басып шығару немесе скриншотты алып тастау керек.

3 ЕСЕПТЕ МЫНАЛАР БОЛУЫ КЕРЕК

3.1 жұмыстың атауы мен мақсаты.

3.2 тапсырма шарты (тапсырмалардың толық мәтіні).

3.3 орындалған жұмыстардың нәтижелері және құрастырылған бағдарламалардың бастапқы мәтіндері.

3.4 қорытындылар мен ұсыныстар.

4 БАҚЫЛАУ СҰРАҚТАРЫ

4.1 "SCADA жүйесі"терминін қолданудың мәнін түсіндіріңіз.

4.2 Trace Mode 6 арна монитормының жұмыс принципі.

4.3 таратылған ТП АБЖ жұмысын модельдеу әдістемесін сипаттаңыз.

4.4 "резервтеу" дегеніміз не?

4.5 "Автоқұрылыс" дегеніміз не?

4.6 Trace Mode 6-да түйін арналарын мұрағаттау қалай жүзеге асырылады?

4.7 жоба арналарын мұрағаттау қалай жүзеге асырылады?

4.8 SCADA жүйесінің дабыл есебі және хабарлама генерациясы не үшін жасалады?

4.9 Trace Mode 6-да оператордың графикалық интерфейсінің жұмысы мен қолданылуын түсіндіріңіз.

4.10 компоненттердің, қабаттардың және түйіндердің жіктелуін келтіріңіз.

4.11 техно-FBD бағдарламалау тілінің негізгі түсініктерін атаңыз және түсіндіріңіз.

4.12 FBD блоктарының кіру/шығу түрлерін білесіз.

4.13 FBD бағдарламаларын жазу тәртібін келтіріңіз.

4.15 жұмыста пайдаланылған fbd-блоктардың кіру/шығу мақсатын түсіндіріңіз.

Әдебиеттер

1 TRACE MODE 6 & T-FACTORY Пайдаланушы нұсқаулығы. Жылдам бастау. М.: Қаржы және статистика, 2016. AdAstrA. Research Group, Ltd. – 163 Б.

2 Пайдаланушы нұсқаулығы. TRACE MODE 6. 7 басылым. М.: Қаржы және статистика, 2016. AdAstrA. Research Group, Ltd., 1 том-554 с. 2 том-598 с.

3. Деменков н. п. SCADA-жүйелер ТП АБЖ жобалау құралы ретінде. М.: ММТУ баспасы. Н. Э. Бауман, 2015. – 328 б.

4 М. Ю. шаян тәрізділер автоматтандырудың техникалық құралдары. МГИУ, 2006 – Б.185.

5 Лапшенков г. И., Полоцкий л. м. химия өнеркәсібіндегі өндірістік процестерді автоматтандыру. М.: Химия, 2018 – Б. 296.

6 Сидельников С. и. комбинациялық тізбектерді синтездеу. Новомосковск: НФ РХТУ им. Менделеев, 2019-37 б.