

ՀՀ ԿՐԹՈՒԹՅԱՆ ԵՎ ԳԻՏՈՒԹՅԱՆ ՆԱԽԱՐԱՐՈՒԹՅՈՒՆ

ԵՐԵՎԱՆԻ ՊԵՏԱԿԱՆ ՀԱՄԱԼՍԱՐԱՆ

ՏԻԳՐԱՆ ՌՈՒԲԵՆԻ ԽԱԺԱԿՅԱՆ

**ՏԱՐԲԵՐ ՀԱՃԱԽՈՒԹՅՈՒՆՆԵՐՈՎ ՏԱԿՏԱՎՈՐՎՈՂ
ԹՎԱՅԻՆ ՍԽԵՄԱՆԵՐԻ ՀԱՄԱՁԱՅՆԵՑՄԱՆ ՄԻՋՈՑՆԵՐԻ
ՄՇԱԿՈՒՄԸ**

ԱՏԵՆԱԽՈՍՈՒԹՅՈՒՆ

**Ե.27.01 «Էլեկտրոնիկա, միկրո և նանոէլեկտրոնիկա»
մասնագիտությամբ տեխնիկական գիտությունների
թեկնածուի գիտական աստիճանի հայցման համար**

**Գիտական ղեկավար՝ ՀՀ ԳԱԱ թղթակից անդամ,
ՀՀ գիտության վաստակավոր գործիչ,
տ.գ.դ., պրոֆ. Վ. Շ. Մելիքյան**

ԵՐԵՎԱՆ 2017

ԲՈՎԱՆԴԱԿՈՒԹՅՈՒՆ

ՆԵՐԱԾՈՒԹՅՈՒՆ.....	4
ԳԼՈՒԽ 1. ՏԱՐԲԵՐ ՀԱՃԱԽՈՒԹՅՈՒՆՆԵՐՈՎ ՏԱԿՏԱՎՈՐՎՈՂ ԹՎԱՅԻՆ ՍԽԵՄԱՆԵՐԻ ՀԱՄԱՁԱՅՆԵՑՄԱՆ ՄԻՋՈՑՆԵՐԻ ՄՇԱԿՄԱՆ ԸՆԴՀԱՆՈՒՐ ՀԱՐՑԵՐԸ.....	9
1.1. Տարբեր հաճախություններով տակտավորվող թվային սխեմաների համաձայնեցման միջոցների մշակման անհրաժեշտությունը	9
1.2. Տարբեր հաճախություններով տակտավորվող թվային սխեմաների համաձայնեցման միջոցների մշակման արդի վիճակի վերլուծությունը.....	28
1.3. Տարբեր հաճախություններով տակտավորվող թվային սխեմաների համաձայնեցման միջոցներին առաջադրվող պահանջները և սկզբունքները	44
Եզրակացություններ.....	47
ԳԼՈՒԽ 2. ՏԱՐԲԵՐ ՀԱՃԱԽՈՒԹՅՈՒՆՆԵՐՈՎ ՏԱԿՏԱՎՈՐՎՈՂ ԹՎԱՅԻՆ ՍԽԵՄԱՆԵՐԻ ՀԱՄԱՁԱՅՆԵՑՄԱՆ ԱՌԱՋԱՐԿՎՈՂ ՄԻՋՈՑՆԵՐԸ	48
2.1. ԳԼՋ շեղումները հաշվի առնելու անհրաժեշտությունը տարբեր հաճախություններով տակտավորվող թվային սխեմաների համաձայնեցման առաջարկվող միջոցներում.....	49
2.2. Մետակայունության հայտնաբերմամբ և բնականոն վիճակի վերականգնմամբ համաձայնեցման մեթոդը.....	57
2.3. Սինքրոազդանշանի դինամիկ համապատասխանեցմամբ համաձայնեցման մեթոդը	65
2.4. Տարբեր հաճախություններով տակտավորվող թվային սխեմաների համաձայնեցման միջոցների կիրառմամբ նախագծման գործընթացը	73
2.5. Տարբեր հաճախություններով տակտավորվող թվային սխեմաների համաձայնեցման առաջարկվող միջոցների արդյունավետության գնահատումը ..	83
Եզրակացություններ.....	89
ԳԼՈՒԽ 3. ՏԱՐԲԵՐ ՀԱՃԱԽՈՒԹՅՈՒՆՆԵՐՈՎ ՏԱԿՏԱՎՈՐՎՈՂ ԹՎԱՅԻՆ ՍԽԵՄԱՆԵՐԻ ՀԱՄԱՁԱՅՆԵՑՄԱՆ ՄԻՋՈՑՆԵՐԻ ԻՐԱԿԱՆԱՑՈՒՄԸ	91

3.1. Տարբեր հաճախություններով տակտավորվող թվային սխեմաների համաձայնեցման ծրագրային միջոցի աշխատանքի սկզբունքը	94
3.3. Տարբեր հաճախություններով տակտավորվող թվային սխեմաների համաձայնեցման միջոցների արդյունավետության գնահատումը	111
Եզրակացություններ.....	116
ԵԶՐԱՀԱՆԳՈՒՄ	117
ՕԳՏԱԳՈՐԾՎԱԾ ԳՐԱԿԱՆՈՒԹՅԱՆ ՑԱՆԿ.....	119
ՀԱՎԵԼՎԱԾ 1. ՆԵՐԴՐՄԱՆ ԱԿՏ.....	131
ՀԱՎԵԼՎԱԾ 2. SMARTSYNC ԾՐԱԳՐՈՎ ՀԱՄԱՁԱՅՆԵՑՎԱԾ ORCA ՊՐՈՑԵՍՈՐԻ VERILOG ՆԿԱՐԱԳՐՈՒԹՅՈՒՆԻՑ ՀԱՏՎԱԾ	132
ՀԱՎԵԼՎԱԾ 3. SMARTSYNC ԾՐԱԳՐԱՅԻՆ ԳՈՐԾԻՔԻ PYTHON	143
ՆԿԱՐԱԳՐՈՒԹՅՈՒՆԻՑ ՀԱՏՎԱԾ.....	143
ՀԱՎԵԼՎԱԾ 4. ՕԳՏԱԳՈՐԾՎԱԾ ՆԿԱՐՆԵՐԻ, ԱՂՅՈՒՍԱԿՆԵՐԻ ԵՎ ՀԱՊԱՎՈՒՄՆԵՐԻ ՑԱՆԿԸ.....	148

ՆԵՐԱԾՈՒԹՅՈՒՆ

Թեմայի արդիականությունը: Վերջին տարիներին ինտեգրալ սխեմաներում (ԻՍ) կիսահաղորդչային սարքերի երկրաչափական չափերի շարունակական փոքրացումն ու այդ տարրերի տեղաբաշխման խտության էքսպոնենցիալ աճը հանգեցրել են ԻՍ-երում ֆունկցիոնալ հնարավորությունների էական մեծացման, ինչը այդ համակարգերի ցածր էներգասպառման ապահովման առումով նոր բարդություններ է առաջացրել: ԻՍ-երի էներգախնայողությունը բարձրացնելու համար տարբեր հեղինակների կողմից արդեն մշակվել են այդ բարդությունների հաղթահարմանն ուղղված մի շարք մեթոդներ: Դրանցից են բազմասնուցումային մեթոդը, մեկուսացնող փականների մեթոդը, մակարդակային տեղափոխիչների մեթոդը, դինամիկ լարման և հաճախականության մասշտաբավորումը, ԻՍ-ի՝ տարբեր հաճախություններով տակտավորվող ենթակառուցվածքների՝ սինքրոտիրույթների բաժանման մեթոդը և այլն:

ԻՍ-երում միմյանց հաղորդակցվող սինքրոտիրույթները ասինքրոն են, ինչի պատճառով այդ համակարգերը կոչվում են գլոբալ ասինքրոն տեղային սինքրոն (ԳԱՏՍ) ինտեգրալ սխեմաներ:

Սինքրոտիրույթների միջև փոխանցվող ազդանշանները, կատարելով սինքրոտիրույթի հատում, գործընթացի, լարման և ջերմաստիճանի (ԳԼՋ) փոփոխական լինելու պատճառով կարող են խախտել ընդունող սինքրոտիրույթի տրիգերների հաստատման և/կամ պահպանման ժամանակները՝ հանգեցնելով վերջիններիս ելքերի՝ տրամաբանական «0» և «1» վիճակներից տարբեր վիճակի՝ մետակայունության: Մետակայունությունը հավանականային երևույթ է, որի դեպքում ազդանշանը անորոշ ժամանակով կարող է տրամաբանական մակարդակներից տարբեր արժեքներ ընդունել, ինչը հանգեցնում է հաջորդող հանգույցների ֆունկցիոնալ խափանմանը: Սինքրոտիրույթի հատման ժամանակ մետակայունությունից խուսափելու նպատակով իրականացվում է սինքրոտիրույթների համաձայնեցում. վերջիններիս միջև ներդրվում են հատուկ սխեմաներ:

Համաձայնեցման ժամանակակից հայտնի միջոցները՝ տրիգերային համաձայնեցուցիչները, համաձայնեցման հերթահենք ինտերֆեյսը և այլն, ունեն նախագծման արդի պահանջները չբավարարող արագագործություն, առնվազն 2 տակտային պարբերության չափով հապաղում և իրագործման բարդություն:

Սինքրոտիզմների համաձայնեցման առկա մեթոդները չեն բավարարում ԻՍ-երի մակրոբնութագրերին՝ արագագործությանը, մակերեսին ու էներգասպառմանն առաջադրվող պահանջներին, քանի, որ դրանց իրականացման սկզբունքը հիմնված է հավելյալ տարրերի հաշվին լրացուցիչ հապաղում ներմուծելու և մետակայուն վիճակի առաջացման հավանականությունը նվազեցնելու վրա: Ժամանակակից ԻՍ-երին առաջադրվող պահանջների բավարարման համար առաջացել է տարբեր հաճախություններով տակտավորվող թվային սխեմաներում սինքրոտիզմների համաձայնեցման միջոցների մշակման խիստ անհրաժեշտություն, որոնք թույլ կտան նվազեցնել համաձայնեցման հապաղումը՝ սխեմայի մակերեսի ոչ էական ավելացման հաշվին:

Ատենախոսությունը նվիրված է համաձայնեցման մեթոդների և դրանց հիման վրա սինքրոնացման միջոցների մշակման հարցերի լուծմանը:

Հետազոտության առարկան: Տարբեր հաճախություններով տակտավորվող թվային սխեմաների հաղորդակցման ժամանակ առաջացող խնդիրները, սինքրոտիզմների համաձայնեցման եղանակները, դրանց իրականացման առանձնահատկությունները և համաձայնեցման միջոցների մշակման սկզբունքները:

Աշխատանքի նպատակը: Թվային ԻՍ-երի տարբեր հաճախություններով տակտավորվող ենթակառուցվածքների ճշգրիտ հաղորդակցման համար անհրաժեշտ հնարավորինս փոքր հապաղում ունեցող և մակերեսի աննշան մեծացմամբ համաձայնեցման միջոցների մշակումը և հետազոտումը:

Հետազոտության մեթոդները: Ատենախոսության կատարման ընթացքում օգտագործվել են թվային սխեմաների կառուցման տեսությունը, ասինքրոն սխեմաների հաղորդակցության առանձնահատկությունները, սարքավորումների նկարագրման լեզուներով դրանց նկարագրությունները, օբյեկտա-կողմնորոշված լեզվով ծրագրային գործիքային միջոցների իրականացման մեթոդները:

Գիտական նորույթը.

- Ձևակերպվել են տարբեր հաճախություններով տակտավորվող թվային սխեմաների սինքրոտիրույթների համաձայնեցման միջոցներին առաջադրվող պահանջները, որոնց բավարարումն ապահովում է ինտեգրալ սխեմաների անհրաժեշտ արագագործություն ու կիսահաղորդչային բյուրեղի վրա զբաղեցրած մակերես:
- Ստեղծվել է մետակայուն վիճակի հայտնաբերմամբ և բնականոն վիճակի վերականգնմամբ համաձայնեցման մեթոդ, որում, ազդանշանի վերաթողարկման շնորհիվ, ապահովվում է համաձայնեցման միջոցի մուտքից ելք հապաղման առնվազն 2 անգամ շահում՝ օգտագործված տրամաբանական փականների քանակի՝ ընդամենը 8%-ը չգերազանցող ավելացման հաշվին:
- Մշակվել է սինքրոազդանշանի դինամիկ համապատասխանեցմամբ սինքրոտիրույթների համաձայնեցման մեթոդը, որն օգտագործված տրամաբանական փականների քանակի մեծացման և իրականացման որոշակի բարդության հաշվին ապահովում է ընդունող սինքրոազդանշանի պարբերությունը չգերազանցող համաձայնեցման հապաղում:
- Առաջարկվել է համաձայնեցման միջոցների նախագծման գործընթաց, որն ապահովում է դրանց՝ գործընթացի, լարման և ջերմաստիճանի նկատմամբ զգայունության էական նվազեցում և համաձայնեցված համակարգի կայունություն:

Պաշտպանության են ներկայացվում հետևյալ դրույթները.

- Տարբեր հաճախություններով տակտավորվող թվային սխեմաների համաձայնեցման միջոցների մշակման սկզբունքները.
- Մետակայուն վիճակի հայտնաբերմամբ և բնականոն վիճակի վերականգնմամբ համաձայնեցման մեթոդը.
- Սինքրոազդանշանների դինամիկ համապատասխանեցմամբ մեթոդը.
- Սինքրոտիրույթների համաձայնեցման միջոցների նախագծման գործընթացը:

Աշխատանքի գործնական արժեքը: Ատենախոսությունում մշակված սկզբունքների և մեթոդների հիման վրա կառուցված համաձայնեցման միջոցների՝ SmartSync ծրագրային գործիքի միջոցով ներդնումը, վերջինիս պարզ և հարմարավետ ինտերֆեյսի շնորհիվ ապահովում է նախագծման ժամանակի շոշափելի կրճատում՝ այն հասցնելով մի քանի վայրկյանից մինչև մի քանի րոպե տևողության: OpenSPARC T1, OpenSPARC T2 և ORCA պրոցեսորներում սինքրոտիրությունների համաձայնեցումն իրականացվել է առաջարկված մետակայունության հայտնաբերմամբ ու բնականոն վիճակի վերականգնմամբ և սինքրոազդանշանների դինամիկ համապատասխանեցմամբ մեթոդներով և հայտնի հերթահենք միջոցով: SmartSync ծրագրային գործիքով առաջարկված առաջին մեթոդով համաձայնեցման կառուցվածքների ներդնումը, հերթահենք համաձայնեցման մեթոդի իրականացման համեմատ, օգտագործված փականների քանակի 7,1 - 7,74% ավելացման հաշվին ապահովում է 70 - 74% համաձայնեցման հապաղման նվազեցում, իսկ սինքրոազդանշանների համապատասխանեցման մեթոդով համաձայնեցման դեպքում, հերթահենք մեթոդի համեմատ ընդամենը 1,13 - 3,7% ավելի շատ փականներ օգտագործելու պարագայում ապահովվում է համաձայնեցման 80 - 81%-ով փոքր հապաղում:

Գիտական դրույթների հավաստիությունը հիմնավորված է մշակված համաձայնեցման միջոցներով կիրառական մեծ նշանակություն ունեցող պրոցեսորներում սինքրոտիրությունների համապատասխանեցմամբ, ինչի դեպքում այդ միջոցների արագագործության համար ստացված ցուցանիշները լիովին համընկնում են տեսական վելուծություններով ստացված ակնկալվող արժեքներին:

Ներդրումը: SmartSync ծրագրային միջոցը ներդրված է «Սինոփսիս Արմենիա» ՓԲԸ-ում: Այն օգտագործվում է բարձր արագագործություն պահանջող ԻՍ-երում հաղորդակցվող սինքրոտիրությունների համաձայնեցման համար, երբ անհրաժեշտ է ապահովել ԳԼԶ շեղումների նկատմամբ անտարբերություն, և երբ մակերեսի որոշակի մեծացումն ընդունելի է:

Աշխատանքի փորձարկումը: Ատենախոսության հիմնական գիտական և գործնական արդյունքները զեկուցվել են՝

- 9-րդ միջազգային «EWME: European Workshop on Microelectronics Education» գիտաժողովում (Գրենոբլ, Ֆրանսիա, 2012թ.),
- 9-րդ միջազգային «Micro- and Nanoelectronics» գիտաժողովում (Երևան, Հայաստան, 2013թ.),
- 57-րդ միջազգային «Electronics, Telecommunications, Computers, Automation and Nuclear Engineering» գիտաժողովում (Զլատիբոր, Սերբիա, 2013թ.),
- 13-րդ միջազգային «EWDTS: East-West Design & Test» սիմպոզիումում (Բաթում, Վրաստան, 2015թ.),
- 14-րդ միջազգային «EWDTS: East-West Design & Test» սիմպոզիումում (Երևան, Հայաստան, 2016թ.),
- «Սինոփսիս Արմենիա» ՓԲԸ ուսումնական դեպարտամենտի գիտական սեմինարներում (Երևան, Հայաստան, 2015-2017թթ.):

Հրապարակումներ: Ատենախոսության հիմնական դրույթները ներկայացված են 7 գիտական աշխատանքներում

Ատենախոսության կառուցվածքը և ծավալը: Ատենախոսությունը բաղկացած է 3 գլուխներից, եզրահանգումից, 155 անուն գրականության ցանկից և 4 հավելվածներից: 1-ին հավելվածում ներկայացված է ատենախոսության ներդրման ակտը, երկրորդում՝ համաձայնեցված սխեմաների նկարագրությունները: 3-րդ հավելվածում ներկայացված են մշակված ծրագրային միջոցի Python ծրագրավորման լեզվով նկարագրությունից հատվածներ, իսկ չորրորդում՝ ատենախոսությունում բերված նկարների, աղյուսակների և հապավումների ցանկերը: Հիմնական տեքստը կազմում է 124 էջ, որում ընդգրկված են 79 նկար և 13 աղյուսակ: Աշխատանքի ընդհանուր ծավալը, հավելվածների հետ միասին, կազմում է 148 էջ:

ԳԼՈՒԽ 1. ՏԱՐԲԵՐ ՀԱՃԱԽՈՒԹՅՈՒՆՆԵՐՈՎ ՏԱԿՏԱՎՈՐՎՈՂ ԹՎԱՅԻՆ ՍԽԵՄԱՆԵՐԻ ՀԱՄԱՁԱՅՆԵՑՄԱՆ ՄԻՋՈՑՆԵՐԻ ՄՇԱԿՄԱՆ ԸՆԴՀԱՆՈՒՐ ՀԱՐՑԵՐԸ

1.1. Տարբեր հաճախություններով տակտավորվող թվային սխեմաների համաձայնեցման միջոցների մշակման անհրաժեշտությունը

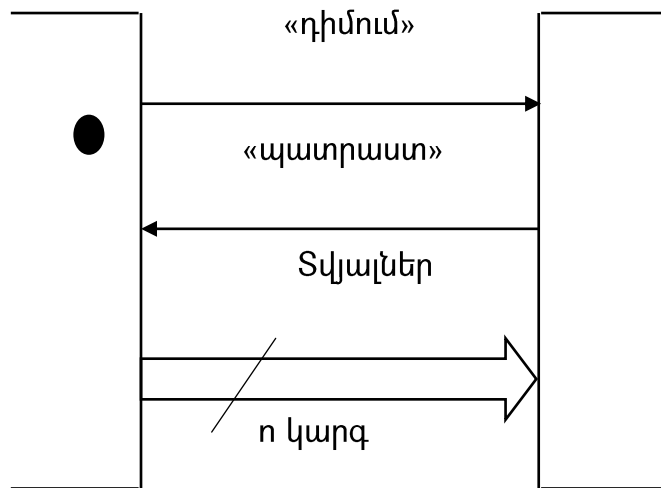
Վերջին մի քանի տասնամյակների ընթացքում ինտեգրալ սխեմաներում (ԻՍ) զրանցվել է տրանզիստորների քանակի էքսպոնենցիալ աճ [1]: ԻՍ-երի այդ զարգացումը պայմանավորված է տրանզիստորների չափերի անընդհատ նվազեցմամբ՝ մասշտաբավորմամբ [1-3], որի արդյունքում դրանք, 5 մկմ-ից փոքրանալով, դարձել են ընդհուպ մինչև 7 նմ [4,5]: Տեխնոլոգիական մասշտաբավորում իրականացնելու հիմնական նպատակներն են եղել՝

- ԻՍ-ի ֆունկցիոնալ հնարավորությունների ավելացումը [2, 5, 6]՝ ԻՍ-ի միավոր մակերեսում տրանզիստորների քանակի աճի միջոցով [6, 7] վերջիններիս տեղաբաշխման խտության մեծացման հաշվին,
- տրանզիստորի փոխանջատման ժամանակների և ԻՍ-ում միջմիացումների հապաղումների փոքրացումը, ինչը հանգեցնում է սխեմայի արագագործության բարելավմանը [2,6]:

Տրանզիստորների չափերի փոքրացումն առավել շոշափելի ազդեցություն է թողել մասնավորապես թվային ԻՍ-երի զարգացման վրա [8, 9], որոնց մի քանի քառակուսի միլիմետր մակերեսի վրա տեղաբաշխվում են տասնյակ միլիարդավոր տրամաբանական փականներ, տրիգերներ, մուլտիպլեքսորներ և թվային այլ սխեմաներ (ԹՍ) [7]: Կիսահաղորդչային մյուս տեխնոլոգիաների [10] համեմատ, իր հեշտ մասշտաբավորելիության շնորհիվ [1,9], կոմպլեմենտար մետաղ-օքսիդ-կիսահաղորդիչ (ԿՄՕԿ) տեխնոլոգիաները [9], որոնց հիմքում ընկած է ՄՕԿ տրանզիստորը [1,9,10], երեք տասնամյակ շարունակ գերիշխող դեր են ունեցել թվային ԻՍ-երում [9]:

Հայտնի է [2,8], որ թվային համակարգերի տարբեր ենթամասերում տրամաբանական գործողությունները կարող են կատարվել միմյանցից անկախ՝ միաժամանակ կամ խիստ որոշակի հերթականությամբ՝ ԹՄ-ը բաժանելով, համապատասխանաբար, ասինքրոն և սինքրոն տեսակների [2]: Նշված տեսակների տարբերությունը որոշակի հաճախականություն ունեցող ղեկավարող ազդանշանի՝ սինքրոնազդանշանի (ՍԱ) [2,11] առկայությունն է: Վերջինս առկա է սինքրոն սխեմաներում և հանդիսանում է տրամաբանական գործողությունների կատարման հաշվարկման սկզբնակետ՝ համակարգելով սինքրոն սխեմայի աշխատանքը [2,11]:

Մինչև 1970-ական թվականները էլեկտրոնային հաշվիչ սարքերը հիմնված են եղել [2,12,13] բացառապես ասինքրոն տեսակի սխեմաների վրա [12,13], որոնք ընդհանուր հաշվարկման սկզբնակետ չպահանջող ոչ համակցական [12,13] սխեմաներ են: Դրանցում տրամաբանական գործողությունները կատարվում են, երբ համապատասխան միջմիացումների վրա տեղի է ունենում ազդանշանի փոխանջատում՝ իրադարձություն [12-14]: Ընդհանուր սինքրոնազդանշան չունենալու պատճառով ասինքրոն սխեմաներում, գործողությունների ժամանակային հաջորդականությունը ղեկավարելու նպատակով, կիրառում են համաձայնեցման ձեռքսեղմման արձանագրություն (ՁԱ) [12,14,15]: Ըստ դրա՝ սխեմայի առանձին հաշվողական միավորները, իրենց գործառույթն իրականացնելուց հետո, գեներացնում են ազդանշաններ [15], որոնք սխեմայի մյուս միավորներին տեղեկացնում են գործողության ավարտի մասին: ՁԱ-ով համաձայնեցված ասինքրոն սխեմաներում տվյալների հոսքը ղեկավարվում է երկու՝ «դիմում» և «պատրաստ» ազդանշաններով (նկ. 1.1) [14, 16]:

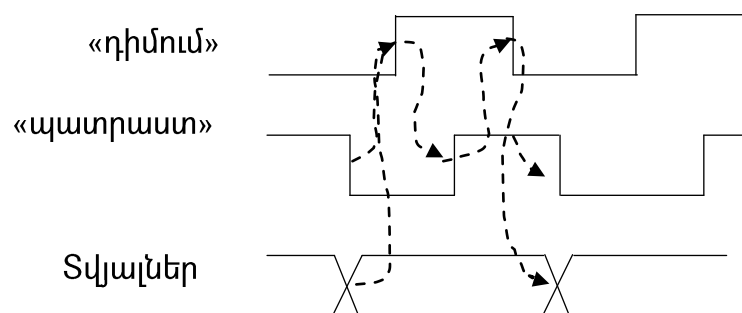


Նկ. 1.1. Ասինքրոն սխեմաների՝ ՁԱ-ով համաձայնացումը

Տարբերում են [15-17] ՁԱ-երի հետևյալ հիմնական տարատեսակները՝ քառափուլ-միագիծ, երկփուլ-միագիծ, քառափուլ-երկգծանի և երկփուլ-երկգծանի: Արձանագրության անվան մեջ փուլերի քանակը ցույց է տալիս ազդանշանների փոխանցումների թիվը [12], իսկ գծերի քանակը մատնանշում է տվյալների մեկ կարգի կոդավորման համար օգտագործված լարերը [12]: Քառափուլ-միագիծ ՁԱ-ում (նկ. 1.2) [12] ասինքրոն սխեմաները հաղորդակցվում են հետևյալ կերպ [12,13]՝

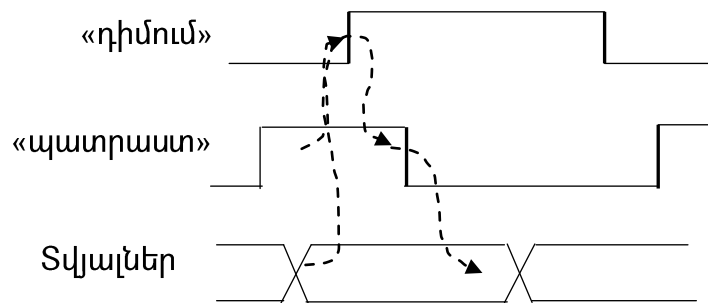
- ուղարկող տարրն (ՌԻՏ) ակտիվացնում է «դիմում» ազդանշանը՝ ընդունող տարրին (ԸՏ) տեղեկացնելով տվյալների հասանելիության մասին,
- ԸՏ-ն, ի պատասխան, ակտիվացնում է «պատրաստ» ազդանշանը՝ ՌԻՏ-ին տեղեկացնելով, որ տվյալներն ընդունվել են,
- ՌԻՏ-ն արձագանքում է՝ զրոյացնելով «դիմում» ազդանշանը,
- ԸՏ-ն զրոյացնում է «պատրաստ» ազդանշանը:

Հաղորդակցման այս 4 փուլերից հետո ՌԻՏ-ը կարող է ուղարկել տվյալների հաջորդ փաթեթը:



Նկ. 1.2. Քառափուլ-միագիծ ձեռքսեղմման արձանագրությունը

Քառափուլ միագիծ ՁԱ-ի հիմնական թերությունը ազդանշանների զրոյացումներն են [12, 15], որոնք հանդիսանում են լրացուցիչ ժամանակի ծախսի և էներգասպառման աղբյուր [12]: Այդ խնդրից զուրկ է երկփուլ միագիծ ՁԱ-ն (նկ. 1.3) [12]:



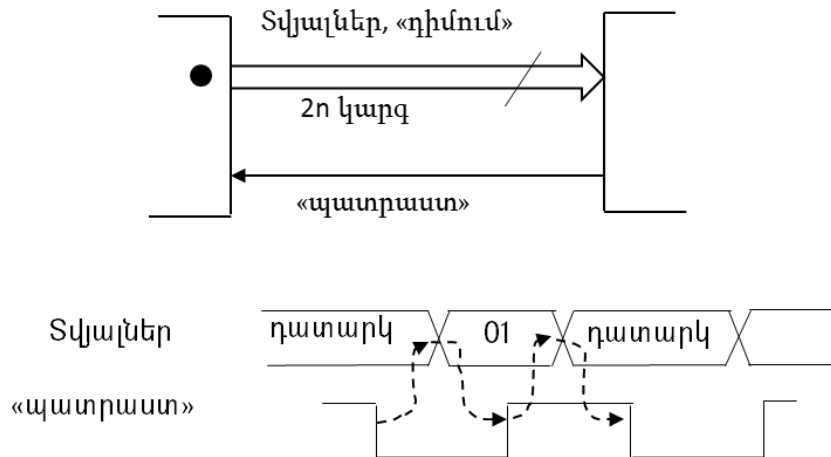
Նկ. 1.3. Երկփուլ միագիծ ձեռքսեղմման արձանագրությունը

Այս դեպքում փոխանցվող տվյալները «դիմում» և «պատրաստ» ազդանշանների միջոցով կոդավորվում են որպես $0 \rightarrow 1$ կամ $1 \rightarrow 0$ փոխանջատումներ [15], որոնք միմյանց համարժեք իրադարձություններ են: Դա նշանակում է, որ համապատասխան ազդանշանն ակտիվացնելու կամ զրոյացնելու փոխարեն՝ տեղի է ունենում տվյալ ազդանշանի կամայական փոխանջատում [13]: Նկարագրված երկու ՁԱ-երում տվյալների փոխանակությունը նախաձեռնվում էր ՈՒՏ-ի կողմից, սակայն հնարավոր է արձանագրություն, որում ԸՏ-ն է դիմում ՈՒՏ-ին՝ տվյալներ ստանալու համար [12]: Այս դեպքում «դիմում» և «պատրաստ» ազդանշանները փոխատեղվում են: Համաձայնեցման արձանագրության մեկ այլ տեսակը քառափուլ-երկգծանի ՁԱ-ն է (նկ. 1.4) [12], որի դեպքում «դիմում» ազդանշանը ներդրված է տվյալների փաթեթի մեջ, և փոխանցվող տվյալի յուրաքանչյուր կարգ կոդավորվում է երկու կարգով՝ մեկը տվյալի տրամաբանական բարձր, մյուսը՝ ցածր արժեքի համար: Փոխանցվող յուրաքանչյուր d տվյալ կոդավորվում է աղ. 1.1-ում [12] նշված արժեքներից մեկով:

Աղյուսակ 1.1

Մուտքային կարգի երկկարգանի ներկայացումը

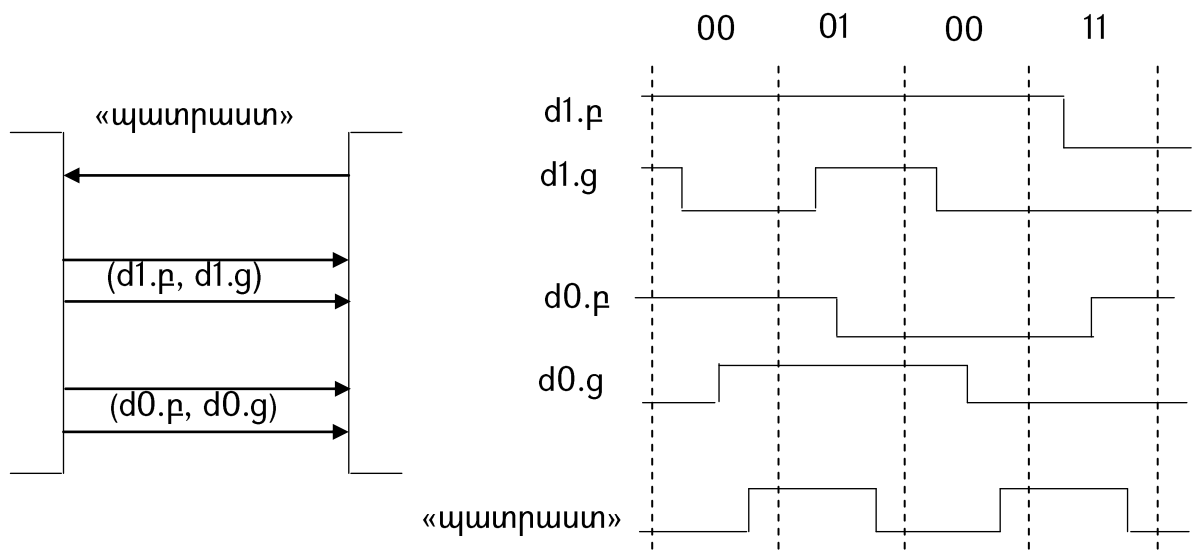
Արժեքը	Տրամաբանական արժեքը
{Բարձր, ցածր}	
{0, 0}	Դատարկ ազդանշան
{0, 1}	Տրամաբանական 0
{1, 0}	Տրամաբանական 1



Նկ. 1.4. Քառափուլ-երկգծանի ձեռքսեղմման արձանագրությունը

Հարկ է նշել, որ յուրաքանչյուր տրամաբանական արժեք փոխանցելուց հետո, ՈԻՏ-ն ուղարկում է դատարկ ազդանշան [13]: Համաձայնեցման այս արձանագրությունը բավարար հուսալի է և կախված չէ հաղորդակցվող սխեմաների միջմիացումների հապաղումներից. հապաղման նկատմամբ անտարբեր է [12, 18]:

Երկգծանի մյուս ՁԱ-ն երկփուլն է (նկ. 1.5) [12]: Փոխանցվող մեկ կարգը կրկին կոդավորվում է 2 կարգով, որոնք այս դեպքում ներկայացնում են փոխանջատումներ [18]. Վ.բ կարգի փոխանջատումը մատնանշում է տվյալի տրամաբանական բարձր արժեք ունենալը, իսկ վ.գ-ինը՝ ցածր: Բազմակարգ ազդանշանների փոխանցման դեպքում «պատրաստ» ազդանշանն ակտիվացվում է միայն այն ժամանակ, երբ յուրաքանչյուր կարգը կոդավորող զույգի ուղիղ մեկ անդամ փոխում է արժեքը [12]: Այս դեպքում դատարկ ազդանշան այլևս չի փոխանցվում:



Նկ. 1.5. Երկփուլ-երկգծանի ձեռքսեղմման արձանագրությունը

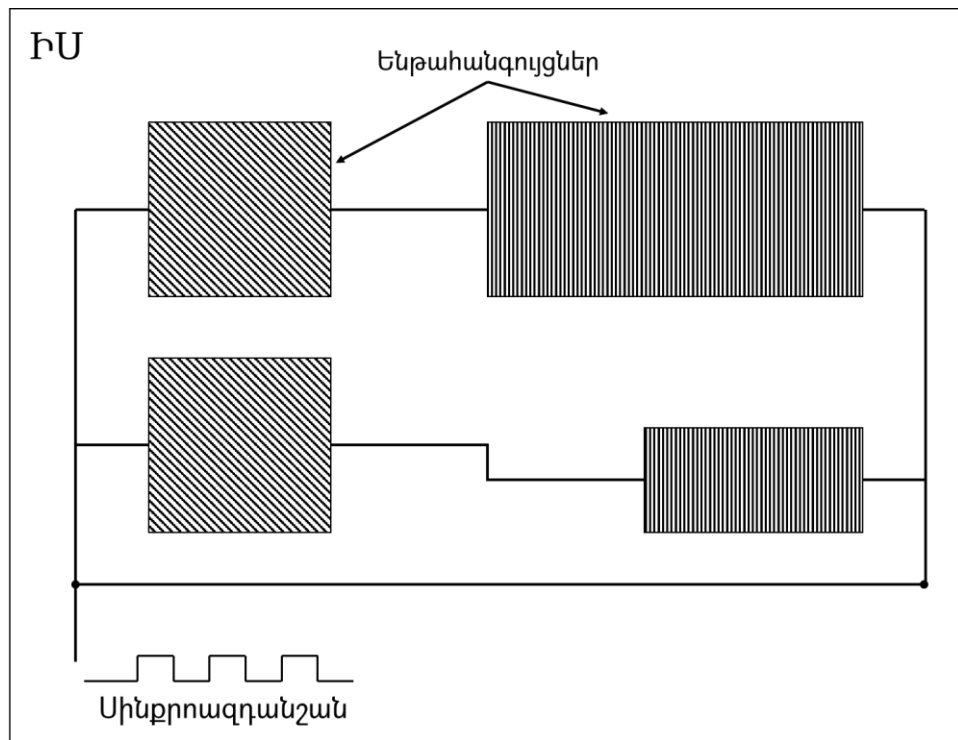
Ասինքրոն սխեմաներն ունեն մի շարք առավելություններ [19,20].

- Ցածր էներգասպառում: Տրամաբանական գործողությունների իրադարձային բնույթի շնորհիվ՝ էներգիա է սպառվում միայն այն ժամանակ, երբ փականի մուտքային արժեքը փոխվում է [20,21]: Դա էապես նվազեցնում է էներգասպառումը [20]:
- Բարձր արագագործություն: Ասինքրոն սխեմաներում կատարվող գործողությունների արագությունները սահմանափակվում են միայն փականների սեփական հապաղումներով:
- Տեխնոլոգիական փոփոխականությունների, սնուցման լարման և ջերմաստիճանի տատանումների նկատմամբ կայունություն [22]:

Ասինքրոն սխեմաների հուսալի աշխատանքն ապահովելու համար հնարավոր չէ խուսափել համաձայնեցման արձանագրությունների իրականացումից [15,17], սակայն դա անխուսափելիորեն հանգեցնում է սխեմայի մակերեսի, արագագործության և էներգասպառման վատթարացմանը:

1970-ականներին ԻՍ-ի բարդության սրընթաց աճի [7] պատճառով դժվարացել էր ասինքրոն համակարգերի անխափան աշխատանքի կազմակերպումը [23], ինչի պատճառով վերջիններն իրենց տեղը զիջեցին սինքրոն՝ համաձայնեցված համակարգերին [24]:

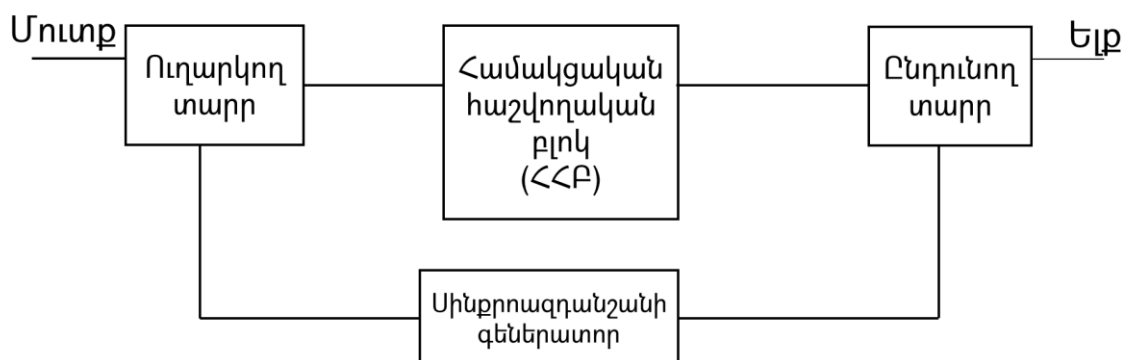
Սինքրոն համակարգերը [2] ԹՄ-եր են, որոնց ենթակառուցվածքները ղեկավարվում՝ տակտավորվում են մեկ ընդհանուր պարբերական ազդանշանով՝ սինքրոնազդանշանով [8,9]: Նույն սինքրոնազդանշանն ունեցող սխեմաների համախմբի մեջ՝ սինքրոտիրույթում (նկ. 1.6), տրամաբանական գործողությունները կատարվում են այդ տակտավորող ազդանշանով թելադրված ժամանակի դիսկրետ պահերին [25]: Դրանք սինքրոնազդանշանի դրական և բացասական ճակատներին համապատասխանող պահերն են [25]:



Նկ. 1.6. Նույն սինքրոտիզացիայի գտնվող հանգույցները

Գործողությունների կատարման նման հաջորդականացումը հայտնի է որպես համաձայնեցում [26]: Թվային սինքրոն համակարգերը, համաձայնեցման տեսակից կախված, լինում են լրիվ համաձայնեցված [2] և գլոբալ ասինքրոն – տեղային սինքրոն (ԳԱՏՍ) [2,16,25,26]:

Լրիվ համաձայնեցում ունեցող սխեմաներում տրամաբանական գործողությունների՝ անհրաժեշտ հերթականությամբ կատարումն ապահովվում է այդ համակարգի տարբեր ենթակառուցվածքների [25]՝ միևնույն սինքրոազդանշանով ղեկավարվելու շնորհիվ: Տարրական համաձայնեցված սխեման [2] բաղկացած է ՈՒՏ-ից և ԸՏ-ից, դրանց միջև գտնվող համակցական հաշվողական բլոկից ($\llcorner F$) և սինքրոազդանշանի գեներատորից (նկ. 1.7) [2]:



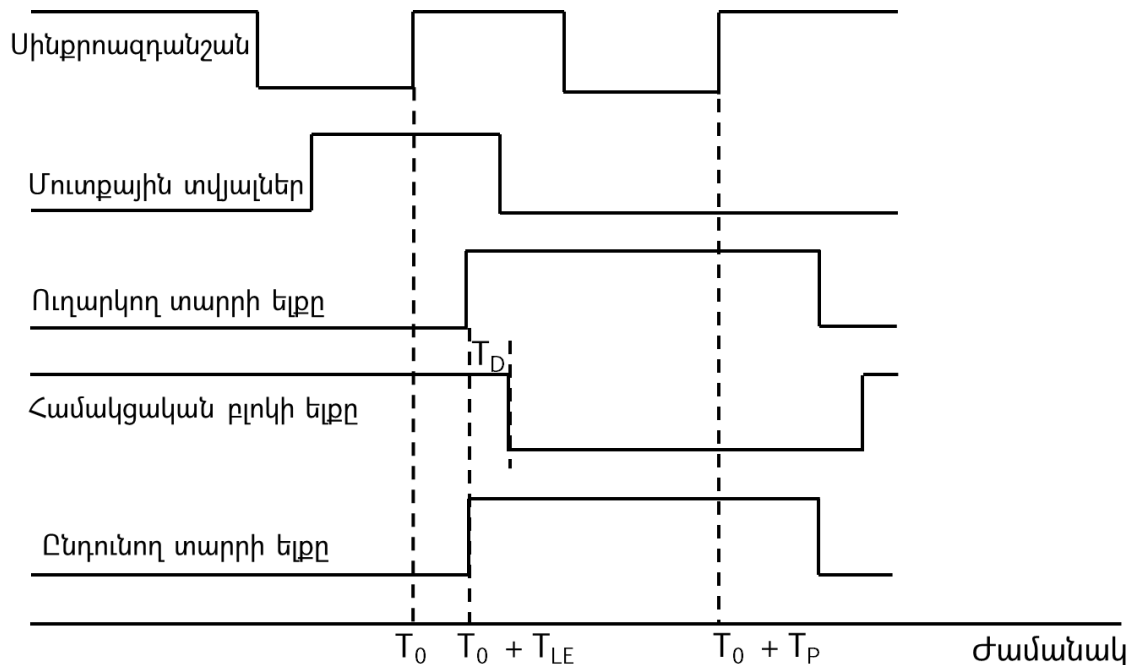
Նկ. 1.7. Տարրական համաձայնեցված սխեմայի կառուցվածքը

Տակտավորող ազդանշանի ղեկավարող ճակատից որոշ ժամանակ անց ՌԻՏ-ի ելքը փոխանջատվում է և փոխանցվում տրամաբանական փականներից բաղկացած ՀՀԲ-ի մուտքին [26]: ՀՀԲ-ի ելքն ընդունում է համապատասխան արժեքը, որը սինքրոազդանշանի հաջորդ ճակատից ԸՏ-ի հապաղում հետո հասանելի է դառնում վերջինիս ելքում (նկ. 1.8):

Մեկ ընդհանուր սինքրոազդանշան ունեցող համակարգում տեղադրվելու դեպքում նկ. 1.7-ում պատկերված սինքրոն սխեման համաձայնեցված կլինի համակարգի մյուս ենթահանգույցների հետ [26]:

Սինքրոն սխեմաների կարևորագույն ժամանակային բնութագրերն են.

- Արտադրողականությունը, որը հավասար է համակարգի սինքրոազդանշանի հաճախականությանը՝ $\frac{1}{T_{su}}$ [2,26]:
- Մուտք-ելք հապաղումը (ՄԵՀ), որը մուտքային նոր ազդանշանի ընդունման պահից մինչև ելքում դրան համապատասխան ազդանշանի ձևավորումը ընկած ժամանակահատվածն է [2]:



Նկ. 1.8. ՀՀԲ-ի՝ շրջիչ լինելու դեպքում տարրական համաձայնեցված սխեմայի ժամանակային դիագրամները

Լրիվ համաձայնեցված համակարգերի հիմնական առավելություններն են դրանցում իրականացվող հաշվողական գործողությունների կառավարելիությունը [26,27] և վիճակների համաձայնեցված փոփոխությամբ պայմանավորված կանխատեսելիությունը [26]: Նշված առավելությունների շնորհիվ՝ մշակվել են լրիվ համաձայնեցված համակարգերի ավտոմատացված նախագծման [28] և թեստավորման մի շարք մեթոդներ [29] և ծրագրային գործիքային միջոցներ [28]: Համաձայնեցման այս տեսակը, սակայն, ունի էական թերություններ [2,26], որոնք համատեղելի չեն ժամանակակից ԻՍ-երի նախագծերին ներկայացվող պահանջներին: Այդ թերություններն են.

- Համակարգի ցածր արագագործությունը [30]. լրիվ սինքրոն համակարգի արտադրողականությունը ուղղակիորեն կապված է սինքրոազդանշանի պարբերության հետ: Այդ պատճառով համակարգի մյուս՝ ավելի արագագործ հատվածները, տակտավորող ազդանշանով դրվում են սպասման մեջ, ինչը էապես նվազեցնում է ընդհանուր համակարգի արագագործությունը [2,30]:
- Կորստի էներգասպառման մեծացումը [31,32]. ԻՍ-ում եղած տրիգերները իրենց վիճակը չեն փոխում սինքրոազդանշանի յուրաքանչյուր պարբերության ժամանակ, սակայն վերջինս պարբերաբար փոխանջատվում է՝ զգալիորեն ավելացնելով ԻՍ-ի սպառած փոխանջատման էներգիան [33]:
- Ներհամակարգային հապաղումների փոփոխականությունը [34]. արտադրական գործընթացի ոչ միօրինակ լինելու, սնուցման լարման տատանումների և միջավայրի ջերմաստիճանի փոփոխականության [35] պատճառով լրիվ սինքրոն սխեմաներում, տարրերի սեփական հապաղումների փոփոխությունների պատճառով, սինքրոազդանշանի բաշխման ցանցի տարբեր ճյուղերում [36] սինքրոազդանշանը տարածվում է տարբեր հապաղումներով [37], և հետևաբար՝ խախտվում է համակարգի սինքրոնությունը:

Հայտնի է [6,7], որ շնորհիվ ԻՍ-երի ֆունկցիոնալ առաջընթացի՝ կատարելագործվել են շարժական էլեկտրոնային սարքավորումները՝ դյուրակիր համակարգիչները և խելացի հեռախոսները, որոնց համար մեծ կարևորություն է

ներկայացնում մարտկոցի՝ առանց լիցքավորման աշխատանքի տևողությունը: Ուստի, ժամանակակից ԻՍ-երի կարևորագույն բնութագիր է ցածր էներգասպառումը [38,39]: ԻՍ-ի աշխատանքի ժամանակ դրա տարբեր ենթակառուցվածքներն աշխատում են տարբեր ծանրաբեռնվածություններով [40], և դրանցից որոշ մասը ակտիվ վիճակում է գտնվում միայն որոշակի ժամանակահատվածներում [40]: Այս հանգամանքը հաշվի առնելով՝ ԻՍ-ի էներգասպառումը կարելի է նվազեցնել՝ փոքրացնելով առանձին ֆունկցիոնալ բլոկների սնուցման լարումները և տակտային հաճախականությունները [41]: Սպառված հզորությունը բաղկացած է ստատիկ և դինամիկ բաղադրիչներից [41]: Դինամիկ էներգիայի հիմնական մասը ելքային ունակության լիցքավորման և լիցքաթափման վրա ծախսվող փոխանջատման էներգիան է: Մեկ փոխանջատման ժամանակ սպառված էներգիան է [42]՝

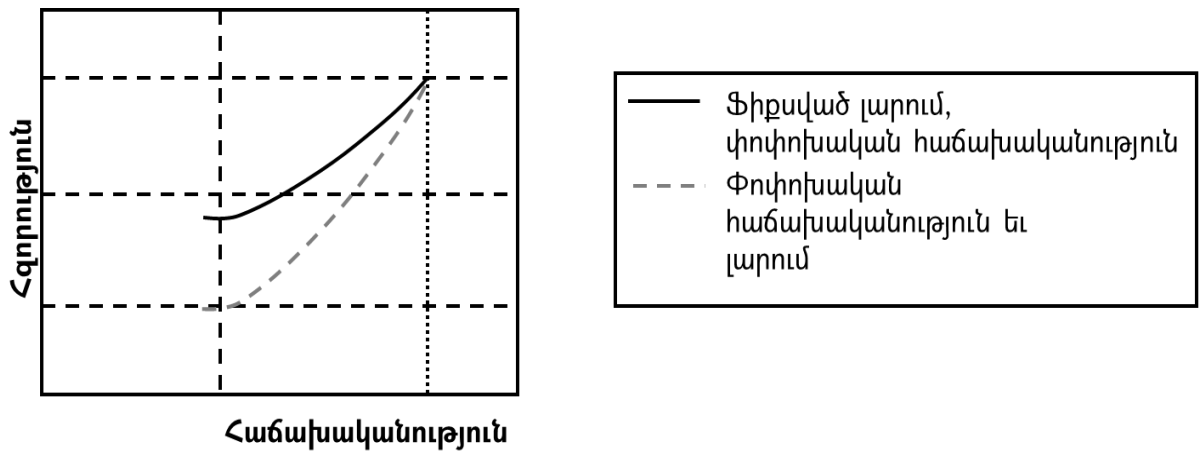
$$E_{\varphi\eta\mu} = C_{\rho\epsilon\eta} * V_{\mu\epsilon}^2, \quad (1.1)$$

որտեղ $C_{\rho\epsilon\eta}$ -ը ելքային բեռի ունակությունն է, իսկ $V_{\mu\epsilon}$ -ը՝ սնուցման լարումը: (1.1)-ից դինամիկ էներգիայի համար ստացվում է [42]՝

$$P_{\eta\mu} = E_{\varphi\eta\mu} * f = C_{\rho\epsilon\eta} * V_{\mu\epsilon}^2 * Q_{\varphi\eta\mu} * f_{\mu\mu}, \quad (1.2)$$

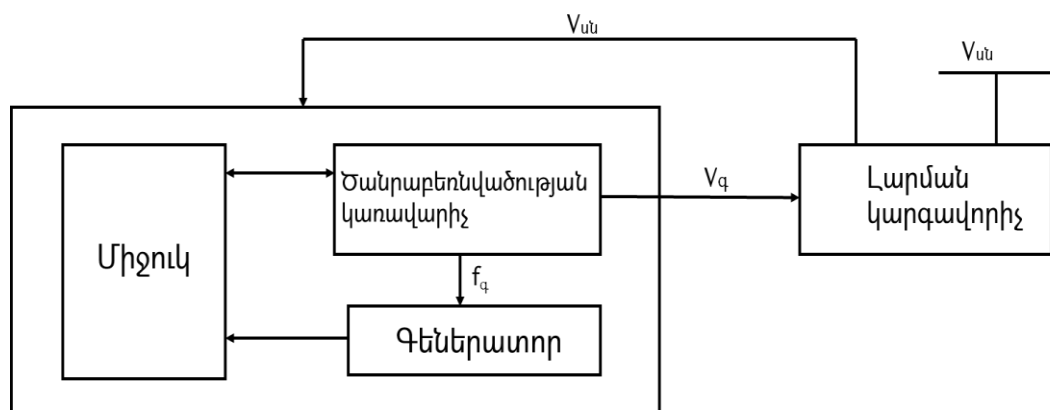
որտեղ f -ը փոխանջատումների հաճախականությունն է, $Q_{\varphi\eta\mu}$ -ը՝ ելքային ազդանշանի փոխանջատվելու հավանականությունը, իսկ $f_{\mu\mu}$ -ը՝ համակարգի սինքրոազդանշանի հաճախականությունը:

(1.2)-ից երևում է, որ սխեմայի էներգասպառումը ուղիղ համեմատական է դրա սինքրոազդանշանի հաճախականությանը (նկ. 1.9) [43], հետևաբար՝ վերջինիս նվազեցումը կհանգեցնի էներգասպառման նվազման: ԻՍ-ի սպառած ամբողջ դինամիկ հզորության մինչև 50%-ը բաժին է հասնում տակտային ազդանշանի բաշխման ցանցին [42], ուստի ԻՍ-ի առանձին ենթահամակարգերի տակտային հաճախականությունների հնարավոր նվազեցումը կբարելավի վերջիններիս և, հետևաբար, ամբողջ ԻՍ-ի էներգախնայողությունը [43]:



Նկ. 1.9. Էներգասպառման՝ հաճախականությունից ունեցած կախվածությունը

ԻՍ-երի էներգախնայողության բարձրացման նպատակով մշակվել են մի շարք եղանակներ, որոնցից է սնուցման լարման և հաճախականության դինամիկ մասշտաբավորման (ԼՀԴՄ) մեթոդը [41-43]: Այն, ելնելով ԻՍ-ի առանձին հանգույցների տվյալ պահի ծանրաբեռնվածությունից, դրանց համար կատարում է լարման և սինքրոազդանշանի հաճախականության մասշտաբավորում [41]՝ ըստ անհրաժեշտության դրանք մեծացնելով կամ փոքրացնելով: ԼՀԴՄ-ի իրականացման դեպքում ԻՍ-ի՝ դինամիկ մասշտաբավորմանը ենթակա ենթակառուցվածքներում տեղադրվում են դրանց աշխատանքային բեռնվածությունը գնահատող սխեմաներ (նկ. 1.10)[40]:



Նկ. 1.10. ԻՍ-ում ծանրաբեռնվածությունը գնահատող սխեման

Ելնելով կատարված գնահատումներից՝ նշված սխեմաները համապատասխան կարգավորող ազդանշանների միջոցով սինքրոազդանշանի գեներատորին հրահանգում են մասշտաբավորել տակտավորման հաճախականությունը, իսկ լարման կարգավորիչին՝ սնուցման լարումը [44]: Որպես

կանոն ԼՀԴՄ համակարգերն օգտագործում են լարում/հաճախականություն դիսկրետ արժեքներ [42,43]: Այդ արժեքները որոշելը ներկայացնում է նախագծման բարդություն և խիստ կախված է սխեմայի կիրառությունից: Հարկ է նշել, որ չափազանց փոքր քանակով արժեքներ ունեցող ԼՀԴՄ-երի դեպքում սխեման կարող է ծախսել մեծ ժամանակ այդ արժեքների միջև փոխանջատվելիս [41,43,44], իսկ մեծ քանակով արժեքների դեպքում սխեման ժամանակի հիմնական մասը կծախսի այդ արժեքների միջև փոխանջատվելով [42]: ԼՀԴՄ մեթոդը օպտիմալ կիրառելու նպատակով անհրաժեշտ է որոշել տվյալ սխեմայի տարբեր բեռնվածություններին համապատասխանող աշխատանքային հաճախականությունները և դրանց համար անհրաժեշտ սնուցման լարումները [45]: Հայտնի է [42], որ լարման կարգավորիչներն ու հաճախականային գեներատորները աշխատանքային մի մակարդակից մյուսին անցնում են որոշակի ժամանակի ընթացքում: Լարումների փոխանջատումները կատարվում են տասնյակ միկրովայրկյաններից մինչև միլիվայրկյանների ընթացքում [42,43], մինչդեռ հաճախականության արժեքների միջև անցումները մի քանի անգամ ավելի արագ են [42]:

Լրիվ համաձայնեցված և ասինքրոն համակարգերի առավելություններն օգտագործելու ու թերությունները հնարավորինս շրջանցելու խնդիրը լուծվում է համակարգերի համաձայնեցման ԳԱՏՍ եղանակով [46]:

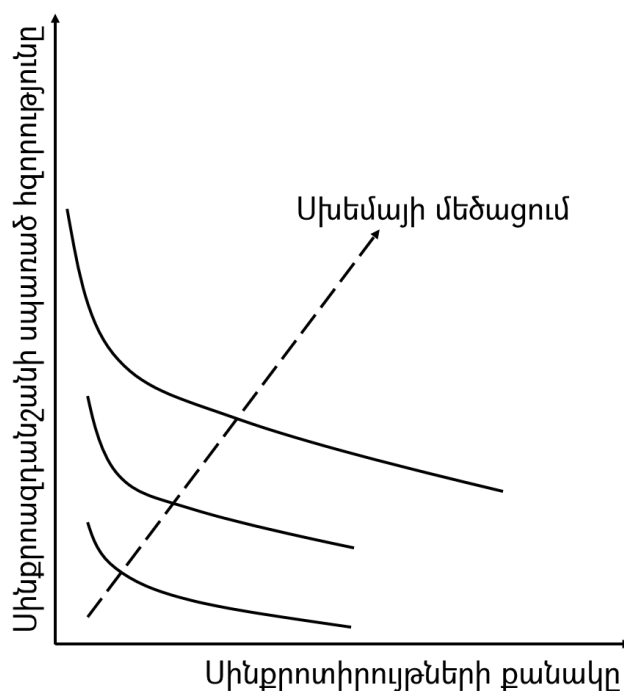
Դրա էությունը հետևյալն է. ԻՍ-ը բաժանվում է հատվածների, որոնցից յուրաքանչյուրում գտնվող սխեմաները տակտավորվում են միևնույն սինքրոազդանշանով [47], սակայն առանձին հատվածները մեկը մյուսի նկատմամբ ասինքրոն են [46,47]: ԻՍ-երի տարբեր ենթակառուցվածքներին կարող են ներկայացվել արագագործության տարբեր պահանջներ [40], ինչը նպատակահարմար է դարձնում դրանց՝ տարբեր սինքրոազդանշաններով տակտավորելը [48]: Նշված հանգամանքների շնորհիվ՝ ԳԱՏՍ մոտեցման կիրառումը հնարավորություն է տալիս լրացուցիչ նվազեցնել համակարգի էներգասպառումը [42,43,45,47,48]:

Տեղային սինքրոն բլոկերի բավարար քանակը նվազեցնում է համակարգի՝ սինքրոազդանշանի փոխանջատումներով պայմանավորված էներգասպառումը (նկ. 1.11) [2, 47]: Այդ պատճառով համակարգի օպտիմալ բաժանումը տարբեր

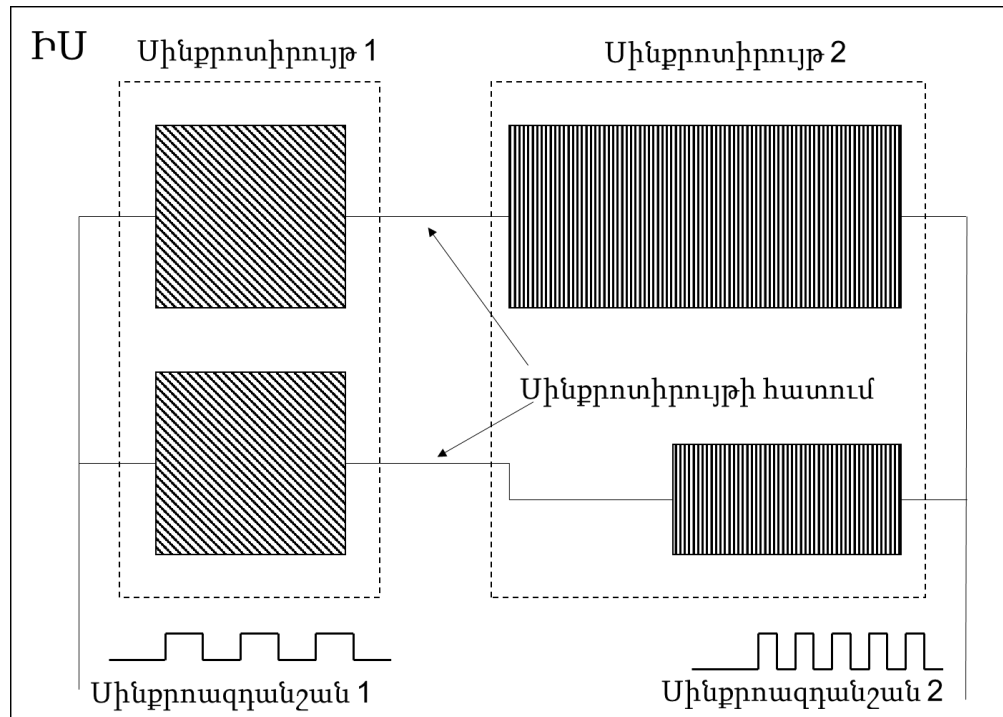
հաճախականային տիրույթների ներկայացնում է ԳԱՏՍ մոտեցմամբ նախագծման հիմնական բարդությունը:

ԳԱՏՍ համակարգերում (նկ. 1.12) սինքրոտիրույթների միջև տեղի է ունենում տվյալների ասինքրոն փոխանակություն, այսինքն՝ փոխանցվող ազդանշանը կատարում է սինքրոտիրույթի հատում: Այդ ԻՍ-երը նախագծվում են այնպես, որ ուղարկված տվյալները ենթարկվեն ընդունող սինքրոտիրույթի տարրերի ժամանակային սահմանափակումներին (հաստատման ժամանակ, պահմապանման ժամանակ և այլն) [49]:

Հայտնի է [50], որ արտադրական գործընթացի անկատարությունները, սնուցման լարման և միջավայրի ջերմաստիճանի տատանումները կարող են պատճառ հանդիսանալ հաղորդակցվող սինքրոտիրույթների տակտավորող ազդանշանների թրթռոցների [51], որոնց պատճառով տվյալների ընդունումը տեղի կունենա սահմանված սահմանափակումների խախտմամբ, իսկ ընդունված ազդանշանը կհայտնվի մետակայուն վիճակում [52-54]: Մետակայունությունը ԹՍ-երին բնորոշ հավանականային երևույթ է [52], որը կարող է հանգեցնել համակարգի ֆունկցիոնալ խափանման: Այն առաջանում է տրիգերի ելքին, երբ վերջինիս ասինքրոն մուտքը փոխանջատվում է տրիգերի հաստատման և/կամ պահպանման ժամանակներից ուշ կամ շուտ [53]:

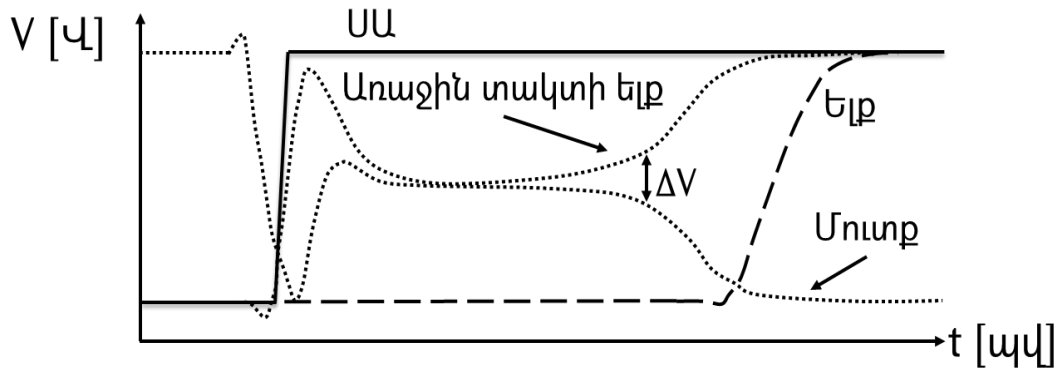


Նկ. 1.11. Սինքրոազդանշանի էներգասպառման՝ սինքրոտիրույթների քանակից ունեցած կախվածությունը



Նկ. 1.12. Երկու սինքրոտիրույթ ունեցող ԳԱՏՍ համակարգը

Ժամանակային այդ հատվածը հայտնի է «արգելված միջակայք» անվամբ [52,54]: Տրիգերի մուտքի՝ վերջինիս արգելված միջակայքում փոխանջատվելու դեպքում սարքի ելքն ընդունում է տրամաբանական մեկ և զրո արժեքներից տարբերվող արժեք: Դա հայտնի է մետակայուն վիճակ անվամբ [52]: Այդ արժեքի մեծությունը կախված է համակարգի սնուցման լարումից: Ազդանշանը մետակայուն վիճակում կարող է գտնվել անորոշ ժամանակ, որից հետո, կախված տրիգերի մուտքի՝ արգելված միջակայքում փոխանջատման պահից, կարող է ընդունել նոր արժեքը կամ վերադառնալ նախորդ վիճակ (նկ. 1.13) [55]:



Նկ. 1.13. Երկտակտ տրիգերում մետակայուն վիճակի առաջացումը

Ազդանշանի՝ մետակայուն վիճակում գտնվելու $Q_{մետ}$ հավանականությունը որոշվում է [56]՝

$$Q_{մետ} = Q_{մետ.առ} * Q_{մետ.0} , \quad (1.3)$$

որտեղ $Q_{մետ.առ}$ -ը ազդանշանի մետակայուն վիճակում հայտնվելու, $Q_{մետ.0}$ -ը սինքրոազդանշանի պասիվ դառնալու պահից t_y ժամանակ հետո մետակայուն վիճակի պահպանման հավանականություններն են: $Q_{մետ}$ -ը հայտնի է նաև որպես խափանման հավանականություն:

Մետակայունության առաջացման $Q_{մետ.առ}$ հավանականությունը որոշվում է [56]՝

$$Q_{մետ.առ} = \frac{T_0}{t_{\Sigma}} , \quad (1.4)$$

որտեղ T_0 -ը սինքրոազդանշանի ճակատի շուրջ արգելված միջակայքն է, իսկ t_{Σ} -ը՝ սինքրոազդանշանի պարբերությունը: (1.4)-ը կարելի է ներկայացնել [56]՝

$$Q_{մետ.առ} = f_{\Sigma} * T_0 : \quad (1.5)$$

Տրիգերի ելքի՝ մետակայուն վիճակ ընկնելու պահից t_y ժամանակ անց մետակայուն վիճակի պահպանման հավանականությունը որոշվում է [56]՝

$$Q_{մետ.0} = e^{-\frac{t_y}{\tau}} : \quad (1.6)$$

Այսպիսով, (1.3)-(1.6)-ից խափանման հավանականության համար ստացվում է [56]՝

$$Q_{մետ} = f_{\Sigma} T_0 e^{-\frac{t_y}{\tau}} : \quad (1.7)$$

τ – ն բնութագրում է տրիգերի ելքի՝ մետակայուն վիճակից դուրս գալու արագությունը [55,56]:

$$\tau = \frac{t_2 - t_1}{\ln\left(\frac{\Delta V_2}{\Delta V_1}\right)}, \quad (1.8)$$

որտեղ ΔV_1 – ը և ΔV_2 – ը, համապատասխանաբար, t_1 և t_2 պահերին մետակայուն ազդանշանի և դրա շրջված արժեքի տարբերություններն են: Այդ տարբերությունները մետակայուն վիճակում մոտ են զրոյին, իսկ բնականոն վիճակում՝ սնուցման լարման արժեքին [54-56]:

(1.7) – ում $T_0 e^{-\frac{t_y}{\tau}}$ բաղադրիչը կոչվում է մետակայունության պատուհան և նշանակվում է δ տառով [55]՝

$$\delta = T_0 e^{-\frac{t_y}{\tau}}: \quad (1.9)$$

Տրիգերի՝ ասինքրոն մուտքի $f_{մուտք}$ հաճախականությամբ փոխանջատման դեպքում դրա λ խափանման հաճախության համար կստացվի [55]՝

$$\lambda = f_{մուտք} * f_{սս} * \delta: \quad (1.10)$$

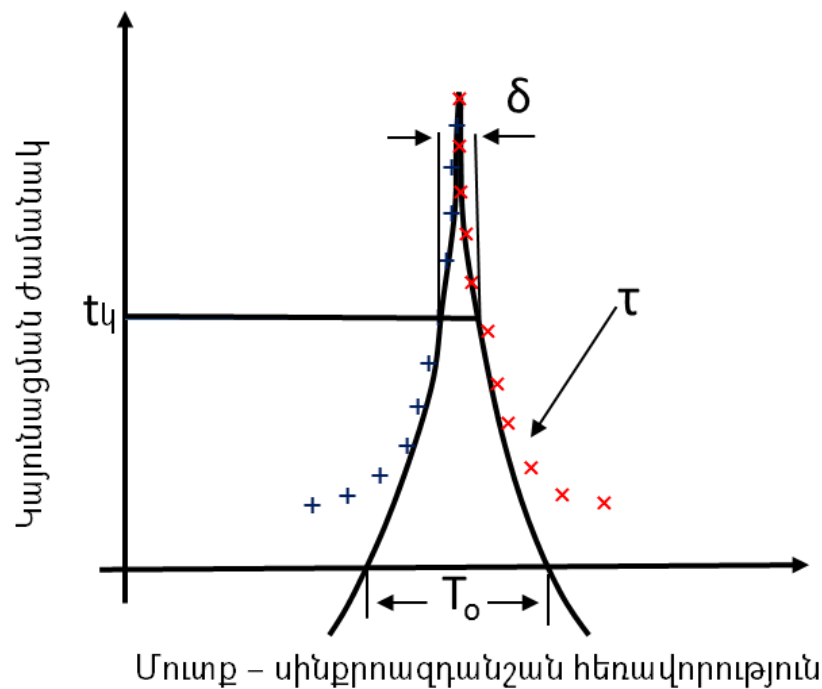
(1.10)-ից տրիգերի խափանման պարբերականության համար կստացվի [55]՝

$$T_{խափ} = \frac{1}{\lambda} = \frac{e^{\frac{t_y}{\tau}}}{f_{մուտք} * f_{սս} * T_0}: \quad (1.11)$$

Մետակայունության հավանականային բնույթի պատճառով $T_{խափ}$ -ը հավանական խափանման պարբերությունն է: Որքան տրիգերի մուտքը սինքրոազդանշանի ճակատին մոտ է փոխանջատվում, այնքան ավելի երկար է մետակայուն վիճակի կայունացման տևողությունը (նկ. 1.14) [55,56]:

Մետակայունության առաջացրած հետևանքների վտանգավորության պատճառով՝ ԳԱՏՍ համակարգերում սինքրոտիզմների միջև տվյալների հուսալի փոխանակություն ապահովելը նախագծման կարևորագույն և բարդ խնդիրներից մեկն է:

Ի սկզբանե ԳԱՏՍ համակարգերը մշակվել են սինքրոն և ասինքրոն սխեմաների առավելությունները միավորելու նպատակով [2,47,48]՝ զուգահեռաբար հաղթահարելով դրանց թերությունները [2]:



Նկ. 1.14. Կայունացման ժամանակի՝ մուտքի և սինքրոազդանշանի ճակատների հեռավորությունից կախումը

Գլոբալ սինքրոազդանշանի բացակայությունը ԳԱՏՍ համակարգերին տալիս է մի շարք առավելություններ՝

- հեշտանում է տեղային մակարդակում սինքրոազդանշանի բաշխման ցանցի նախագծումը [57],
- առանձին սինքրոազդանշանների ուղիների համեմատաբար կարճ լինելու շնորհիվ՝ զգալիորեն նվազեցվում են տակտավորող ազդանշանի ուշացումները և դրանց ունեցած բացասական ազդեցությունները [48,53],
- բացակայում է մեկ գլոբալ սինքրոազդանշանի՝ չափազանց բարդ կառուցվածք ունեցող բաշխման ցանցի պատճառով ցրված էներգիան [58],
- բազմաթիվ տեղային սինքրոն միավորների գոյությունը կարող է էապես կրճատել համակարգի էներգասպառումը, քանի որ յուրաքանչյուր միավորի համար կարելի է որոշել նրա համար օպտիմալ համաձայնեցման հաճախականությունը [42,43],

- Էներգասպառման նվազեցման՝ ԼՀԴՄ մեթոդի կիրառումը պարզեցվում է [44]:

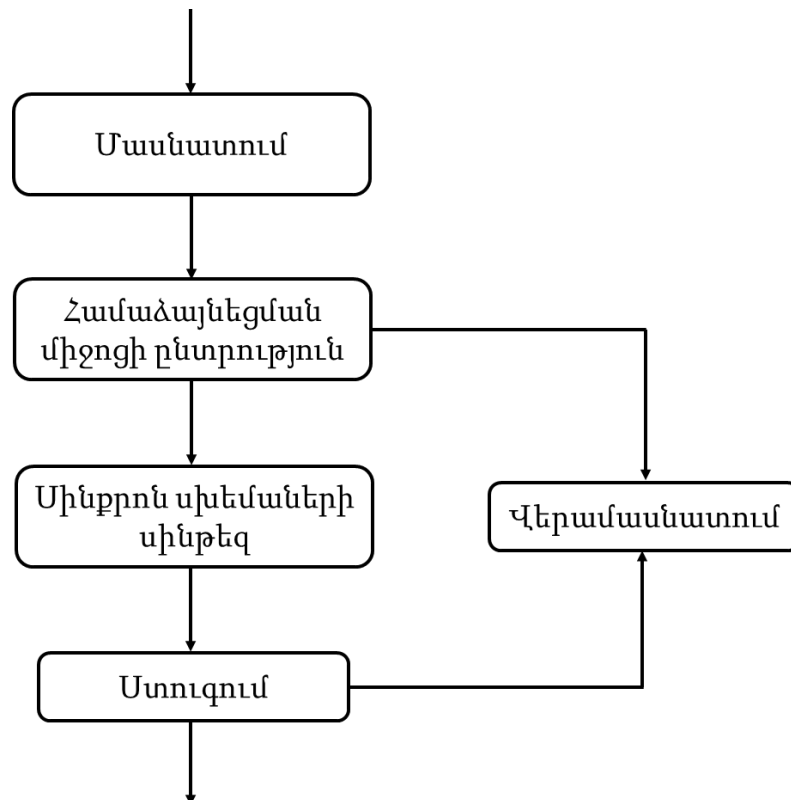
Ցույց է տրվել, որ այս մեթոդի կիրառումը ԳԱՏՍ համակարգում հանգեցնում է սպառված էներգիայի 33% նվազման՝ 10% արագագործության վատթարացման հաշվին [44]:

ԳԱՏՍ համակարգերի այլ առավելություններից է փոխանջատման աղմուկների և էլեկտրամագնիսական ինտերֆերենցիայի նվազեցումը [59,60]:

Գլոբալ սինքրոազդանշան չունեցող այս համակարգերը, սակայն, ունեն մի շարք կարևոր սահմանափակումներ [2,46-48]: Ասինքրոն ենթահանգույցների համաձայնեցման ՁԱ-երից պահանջվում է մետակայունության նկատմամբ բարձր հուսալիություն [16,17], բարձր արտադրողականություն և հնարավորինս փոքր ՄԵՀ [61]: Բոլոր երեք պահանջներին միաժամանակ բավարարելը չափազանց բարդ խնդիր է [46]: Այդ իսկ պատճառով ՁԱ-ի արդունավետությունը որոշվում է տվյալ սխեմայի առաջնային պահանջները բավարարելու ունակությամբ [12,13], ինչը խոչընդոտում է համաձայնեցված սխեմաների համապիտանի նախագծման երթուղու մշակմանը: Համապատասխան ՁԱ մշակելու խնդիրը սինքրոտիրոյթների քանակի աճի հետ ավելի է բարդանում [2]: Միևնույն ժամանակ, լրիվ ասինքրոն սխեմաների համանմանությամբ ՁԱ-ի կիրառումը մեծացնում է համակարգի ընդհանուր մակերեսն ու էներգասպառումը և նվազեցնում արագագործությունը [59]: ԳԱՏՍ համակարգի արագագործությունը խիստ կախված է դրանում ասինքրոն բլոկների համաձայնեցման ընտրված եղանակից [14], ինչը պահանջում է նախագծի մանրամասն ուսումնասիրություն և պլանավորում: ԳԱՏՍ ԻՍ-երի նախագծման համար մշակված է երթուղի (նկ. 1.15) [2]: Նախագծման առաջին փուլում իրականացվում է համակարգի բաժանում ավելի փոքր միավորների՝ ելնելով վերջինիս հիերարխիկական կառուցվածքից [2]: Այնուհետև, հիմնվելով սխեմայի սինքրոազդանշանի պարբերության, մուտքային և ելքային ազդանշանների ճակատների տևողության վրա՝ իրականացվում են մի քանի ՁԱ-եր [2,14]: Այս փուլում հաղորդակցվող սինքրոտիրոյթների յուրաքանչյուր զույգի համար ընտրվում է համաձայնեցման համապատասխան միջոցը [2,13]: Այնուհետև կատարվում է սխեմաների սինթեզ, և

ստուգվում է նախագծված համակարգի՝ հզորության և արագագործության պահանջներին բավարարելը: Տեխնիկական առաջադրանքին չբավարարելու դեպքում նախագծման փուլերը կրկնվում են:

Բացի ՁԱ-ի օպտիմալ ընտրությունից, ԳԱՏՍ համակարգերը պետք է ունենան էներգախնայող և հուսալի սինքրոազդանշանների գեներատորներ [62,63]: Որպես սինքրոազդանշանի գեներատոր օգտագործվող օղակաձև գեներատորները [62] բավականին զգայուն են միջավայրի փոփոխությունների նկատմամբ, և դրանց փոխարեն նախընտրելի է կիրառել փուլային համալարման համակարգերը [63]:



Նկ. 1.15. ԳԱՏՍ համակարգերի նախագծման երթուղին

Սինքրոտիրոյթների քանակի աճին զուգընթաց՝ էական մեծությունների է հասնում սինքրոազդանշանների գեներատորներին բաժին ընկնող էներգասպառումը [42, 64]:

Այսպիսով, ԳԱՏՍ ԻՍ-երի ճշգրիտ և անխափան գործունեությունն ապահովելու համար անհրաժեշտ է ասինքրոն բլոկների համաձայնեցում, ինչը իրականացվում է համաձայնեցուցիչների [65] և համաձայնեցման ինտերֆեյսների միջոցով [66]:

1.2. Տարբեր հաճախություններով տակտավորվող թվային սխեմաների համաձայնեցման միջոցների մշակման արդի վիճակի վերլուծությունը

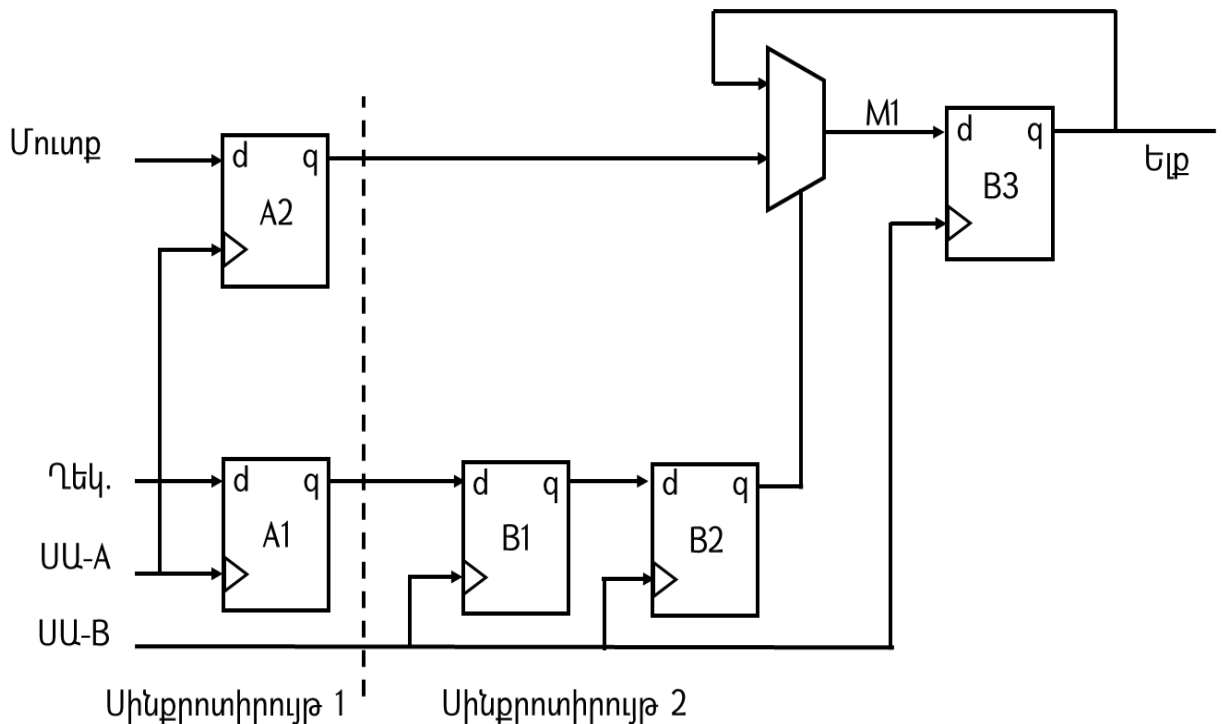
Նախորդ շարադրանքից հետևում է, որ ներկայումս առկա է տարբեր հաճախություններով տակտավորվող ԹՍ-երի համաձայնեցման միջոցների մշակման խիստ անհրաժեշտություն:

Համաձայնեցման հայտնի մոտեցումների վերլուծությունը [17,65-59] ցույց է տալիս, որ դրանք մետակայունության առաջացման հավանականությունը նվազեցնում են սինքրոտիրոյթների միջև լրացուցիչ հապաղումների ներմուծման և արտադրողականության նվազեցման հաշվին [65,66]: Գոյություն ունեցող համաձայնեցման միջոցները դասակարգվում են հետևյալ խմբերի՝ տրիգերային համաձայնեցուցիչներ [65,69] և համաձայնեցման ինտերֆեյսներ [66,69,70]:

Տրիգերային համաձայնեցուցիչներն ու դրանց տարատեսակները ուսումնասիրվել են բազմաթիվ աշխատանքներում [17,65,66,69,71,72]: Մասնավորապես, [72]-ում քննարկվում է հետադարձ կապով մուլտիպլեքսորով տրիգերային համաձայնեցուցիչը (նկ. 1.16): ՈՒՍ-ը, ղեկավարող ազդանշան գներացնելով, տեղեկացնում է ԸՍ-ին, որ վերջինիս մուտքային տվյալները հասանելի են: Ղեկավարող ազդանշանը, երկտրիգերանի համաձայնեցուցչի միջոցով ԸՍ-ում համաձայնեցվելով, ակտիվացնում է մուլտիպլեքսորի՝ ՈՒՍ-ից տվյալ ընդունող մուտքը: Երբ ղեկավարող ազդանշանը պասիվ է, ակտիվ է մուլտիպլեքսորի՝ B3 տրիգերի ելքին միացված մուտքը: Այս համաձայնեցուցչի էական թերությունը 4 տակտային պարբերության հապաղումն է, որը նվազեցնում է համակարգի արտադրողականությունը:

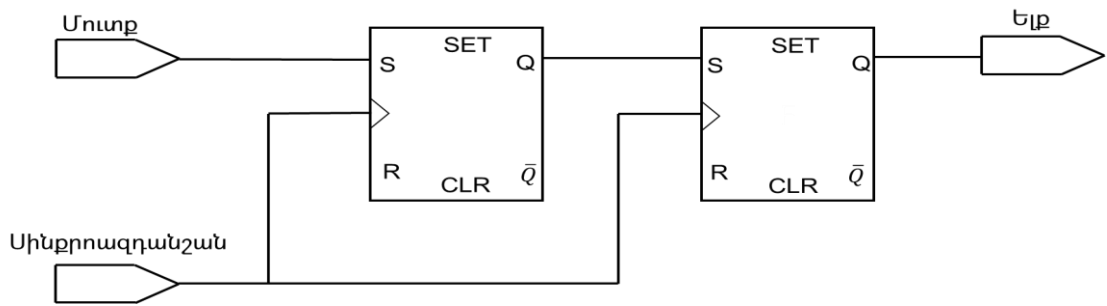
[17]-ում ուսումնասիրված են 3 ամենատարածված տրիգերային համաձայնեցուցիչները՝ մակարդակային (ՄՀ), ճակատ-գրանցող (ՃԳ) և իմպուլս գներացնող (ԻԳ): Աշխատանքում կատարվել է դրանց էներգասպառման և հապաղման համեմատություն՝ առանց ձեռքսեղմման արձանագրությունների և դրանց իրականացմամբ դեպքերում: ՄՀ-ի (նկ. 1.17) [17,65] աշխատանքի սկզբունքը հիմնված է ընդունող սինքրոտիրոյթի մուտքին լրացուցիչ հապաղում ավելացնելու վրա: Այս համաձայնեցուցչի առաջին տրիգերի ելքային ազդանշանի՝ մետակայուն վիճակում հայտնվելու դեպքում տրվում է

ևս մի պարբերություն, որի ընթացքում այն կվերականգնի իր բնականոն վիճակը, և ՄՀ-ի ելքում կձևավորվի ճիշտ տրամաբանական արժեք [65]: ՄՀ-ը հիմնականում կիրառվում է միակարգ ղեկավարող ազդանշանների համաձայնեցման համար [66]:



Նկ. 1.16. Հետադարձ կապով մուլտիպլեքսորով տրիգերային համաձայնեցուցիչը

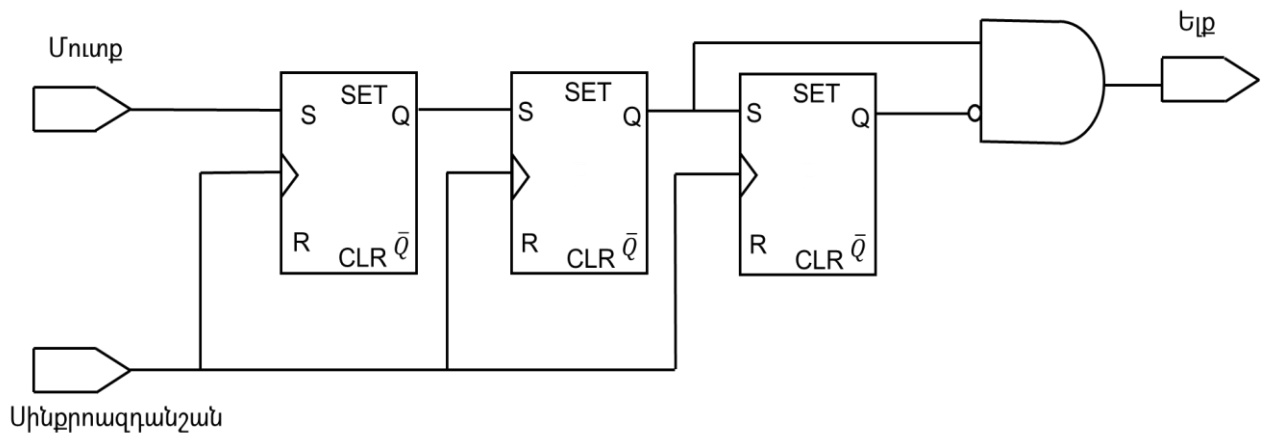
Այն ունի էական թերություն, որը կտրուկ նվազեցնում է դրա հուսալիությունը [65,66]: Արվում է ենթադրություն, որ առաջին տրիգերի ելքում առաջացած մետակայունությունը վերականգնվում է առավելագույնը մեկ պարբերության ընթացքում: Նշված ենթադրությունը գործնական կարևորություն ներկայացնող մի շարք դեպքերում կարող է տեղի չունենալ, ինչը կհանգեցնի համակարգի խափանման: ՄՀ-ի միջոցով բազմակարգ ազդանշաններ համաձայնեցնելու դեպքում այդ ազդանշանի յուրաքանչյուր կարգ համաձայնեցվելու է առանձին ՄՀ-ով [65]: Փոխանջատվող կարգերը, որոնք կհայտնվեն մետակայուն վիճակում, իրենց ընդունող տրիգերներին կհասնեն սինքրոնազդանշանի տարբեր ճակատների ժամանակ՝ առաջացնելով ազդանշանի աղավաղում [66]: Աղավաղումներ չեն լինի, եթե ապահովվի միաժամանակ միայն մեկ կարգի փոխանջատումը, ինչը հնարավոր է իրականացնել բազմակարգ ազդանշանը Գրեյի կոդավորման ենթարկելով [55,56]: Այնուամենայնիվ, համաձայնեցման նմանատիպ եղանակը բազմաթիվ կիրառություններում նպատակահարմար չէ:



Նկ. 1.17. Մակարդակային համաձայնեցուցիչը

[65]-ում նկարագրվում է մուտքային ազդանշանի փոխանջատումը գրանցող համաձայնեցուցիչ, որը հայտնի է ճակատ գրանցող համաձայնեցուցիչ (ՃԳՀ) անվամբ (նկ. 1.18) [65]:

ՃԳՀ-ը, սխեմայում օգտագործված համակցական տարրից կախված, կարող է գրանցել կամ դրական, կամ բացասական ճակատները և ելքում գեներացնել ակտիվ կամ պասիվ ազդանշան [65]: Այս համաձայնեցուցիչն ունի նույն թերություններն, ինչ ՄՀ-ն. այն ունի երկու տակտային պարբերությանը հավասար մուտք-ելք հապաղում և արվում է ենթադրություն, որ մետակայուն ազդանշանն ունակ է վերականգնվելու սինքրոազդանշանի մեկ պարբերության ընթացքում: ՃԳՀ-ի ճշգրիտ աշխատանքի համար դրվում է պահանջ, որ մուտքային իմպուլսի տևողությունը պետք է մեծ լինի դրա սինքրոազդանշանի պարբերությունից:

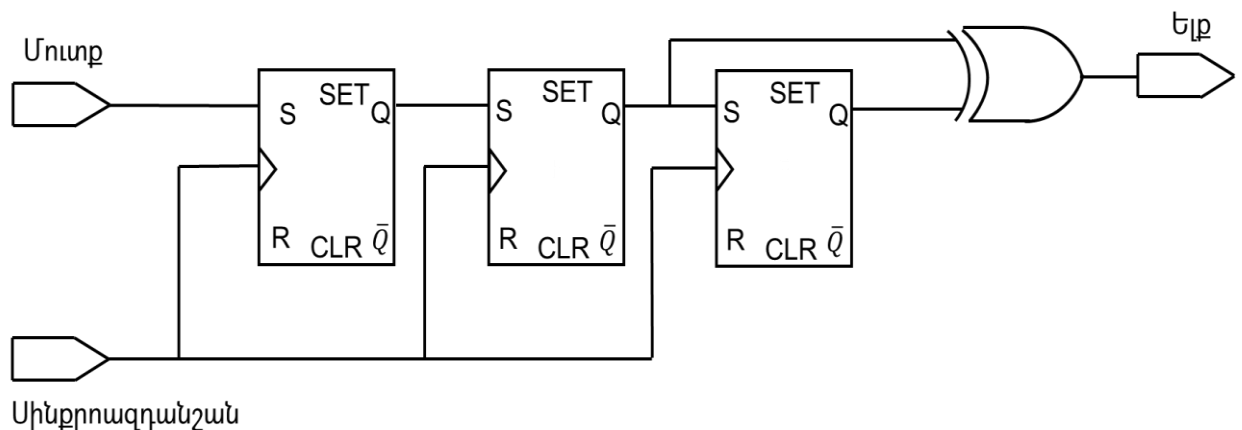


Նկ. 1.18. Ճակատ գրանցող համաձայնեցուցիչը

Համաձայնեցուցի մյուս տեսակը որը քննարկվել է [65]-ում, ԻԳ համաձայնեցուցիչն է (նկ. 1.19) [65], որը մուտքային ազդանշանի փոխանջատումը գրանցելու դեպքում ելքում գեներացնում է տակտային ազդանշանի պարբերության չափով իմպուլս և կրկին հիմնված է վերոհիշյալ ենթադրության վրա:

Համաձայնեցուցիչների այս տեսակների վերաբերյալ [65]-ում արվել և ընդհանրացվել են հետևյալ դիտարկումները.

- Դրանք ունեն տակտային պարբերության կրկնապատիկի չափով հապաղում:
- Նախապես արվում է ենթադրություն, որ առաջին տրիգերի ելքը մետակայուն վիճակում հայտնվելու դեպքում պետք է վերականգնվի մակարդակային համաձայնեցուցչի դեպքում, մեկ տակտային պարբերության ընթացքում, այլապես համաձայնեցուցչի ելքը ևս կհայտնվի մետակայուն վիճակում: Բարձր հաճախականային և բազմաթիվ այլ կիրառություններում մետակայուն ազդանշանից վերականգնվելու համար կարող է ավելի շատ ժամանակ պահանջվել
- Դրանք կիրառական են միակարգ ղեկավարող ազդանշաններ համաձայնեցնելիս: Բազմակարգ ազդանշանների համաձայնեցման դեպքում վերջինիս յուրաքանչյուր կարգը կհամաձայնեցվի մեկ առանձին համաձայնեցուցչով: Այդ դեպքում, տարրերի ներքին հապաղումների շեղումների պատճառով, կա հավանականություն, որ սինքրոնացված կարգերը ընդունող սինքրոտիրույթի մուտքին կհասնեն մեկը մյուսի նկատմամբ որոշակի ուշացումներով, ինչը կհանգեցնի ազդանշանի աղավաղման:



Նկ. 1.19. Իմպուլս գեներացնող համաձայնեցուցիչը

[65]-ում նշվում է, որ համաձայնեցուցիչների այս տեսակները կիրառվում են նաև ձեռքսեղմման արձանագրություն իրականացնող համաձայնեցման մոտեցումներում: Մետակայունությունը բացառելու տեսանկյունից այդ մոտեցումներն ավելի հուսալի են, սակայն ունեն ավելի մեծ հապաղում, էներգասպառում և զբաղեցնում են

համեմատաբար մեծ մակերես: [65]-ում ստացված արդյունքների համաձայն՝ ՄՀ-երի ՄԵՀ-ն ամենափոքրն է, իսկ ԻԳՀ-երինը՝ ամենամեծը: Էներգասպառման տեսանկյունից ՃԳՀ-ն և ԻԳ-ն ունեն համարժեք արդյունքներ: Չնայած ԻԳՀ-ն ունի ամենամեծ հապաղումն ու էներգասպառումը, այնուամենայնիվ, դրա միջոցով լուծվում է ՃԳՀ-ում առաջացող խնդիրը՝ ազդանշանի՝ ցածր հաճախականությամբ սինքրոտիզմի բարձրին ճշգրիտ անցումը: Սակայն, երբ մուտքային ազդանշանի պարբերությունը կրկնակի մեծ է ԸՍ-ի սինքրոազդանշանի պարբերությունից, տեղի է ունենում տվյալների կորուստ: Ստացված արդյունքների համաձայն ամբողջ ընթացքը կազմում է 6-7 տակտային պարբերություն:

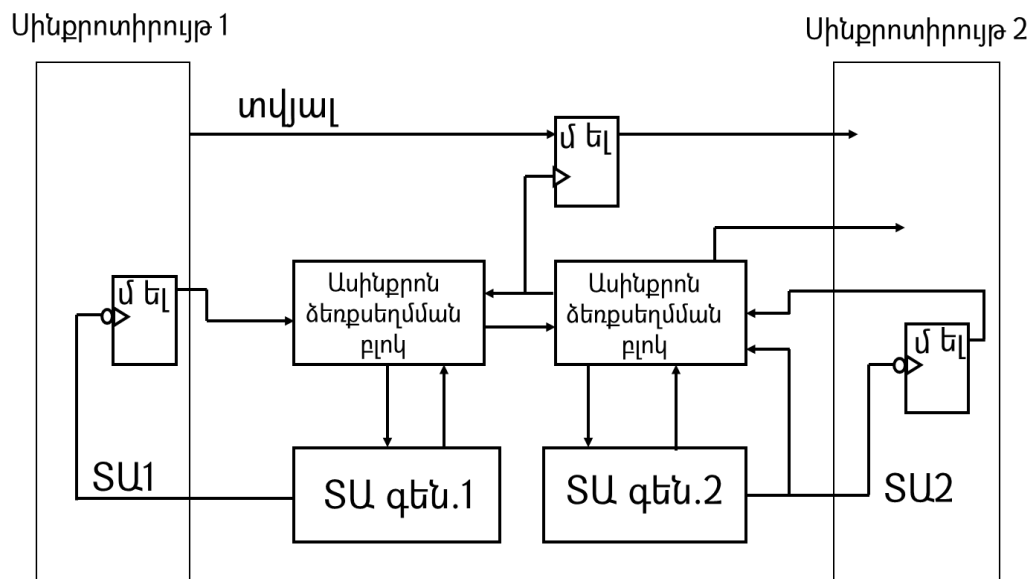
Միլիարդավոր տրանզիստորներ պարունակող բազմաֆունկցիոնալ ԻՍ-երը նախագծվում են կիրառության լայն սահմաններ ունեցող և վերաօգտագործելի կառուցվածքների միջոցով, որոնք հայտնի են «մտավոր սեփականություն» (ՄՍ) անվամբ: Ժամանակակից մի շարք պրոցեսորներ, հիշող սխեմաներ, մուտք/ելքի հանգույցներ և այլ սխեմաներ ՄՍ-եր են: Միևնույն ԻՍ-ում միմյանց հետ հաղորդակցվող տարբեր ՄՍ-երի օգտագործումը առաջացնում է դրանք համաձայնեցնելու կարիք, քանի որ դրանք տակտավորվում են տարբեր հաճախականություններով: Այդ բլոկների համաձայնեցման համար մշակվել են մի շարք ինտերֆեյսներ:

Հայտնի է [66], որ սինքրոազդանշանի հատում կատարող ազդանշանները հիմնականում տվյալների բազմակարգ ազդանշաններ են, որոնց համաձայնեցման համար մշակվել են համաձայնեցման արձանագրություն իրականացնող բազմաթիվ ինտերֆեյսներ: [66]-ում առաջարկվել են համաձայնեցման ինտերֆեյսների տարբեր նախագծեր, որոնց նպատակն է ազդանշանի համաձայնեցման բարձր հուսալիության, հնարավորինս ցածր էներգասպառման, փոքր հապաղման և հնարավորինս մեծ արտադրողականության ապահովումը: ԳԱՏՍ համակարգերում կիրառվող ինտերֆեյսների տեսակներից են՝

- սինքրոազդանշանի ընդհատմամբ (ՍԸԻ),
- սահմանային համաձայնեցմամբ (ՍՀԻ),
- համաձայնեցման հերթահենք (ՀՀԻ):

[66]-ում նկարագրվում է սինքրոազդանշանի ընդհատմամբ ինտերֆեյսը (նկ. 1.20) [66]:

Երբ ուղարկող սինքրոտիրույթը պատրաստ է ուղարկելու տվյալների կարգերը, այն դիմում է ասինքրոն ձեռքսեղմման բլոկին, որն ընդհատում է երկու սինքրոտիրույթների տակտավորող ազդանշանների գեներացիան [66]: Այնուհետև մակարդակային տրիգերի միջոցով կարգերը տեղափոխվում են ընդունող սինքրոտիրույթ, ինչից հետո վերջինս դիմում է ասինքրոն ձեռքսեղմման բլոկին, որն էլ վերականգնում է սինքրոազդանշանների գեներացիան և համակարգի բնականոն աշխատանքը [66]:



Նկ. 1.20. Սինքրոազդանշանի ընդհատմամբ համաձայնեցման ինտերֆեյսը [66]-ում նշվում են ՍԸԻ-ի հետևյալ առավելությունները.

- Սինքրոազդանշանի ընդհատում իրականացնելու շնորհիվ՝ իսպառ վերանում են մետակայունության առաջացման հավանականությունը և համաձայնեցուցիչների կիրառման անհրաժեշտությունը [66]:
- Հաղորդակցվող սինքրոտիրույթների սինքրոազդանշանները կարող են ունենալ կամայական հաճախականություններ և միմյանց նկատմամբ փուլային շեղումներ [66]:

Որպես այս ինտերֆեյսի հիմնական թերություններ [66]-ում նշվում են հետևյալները.

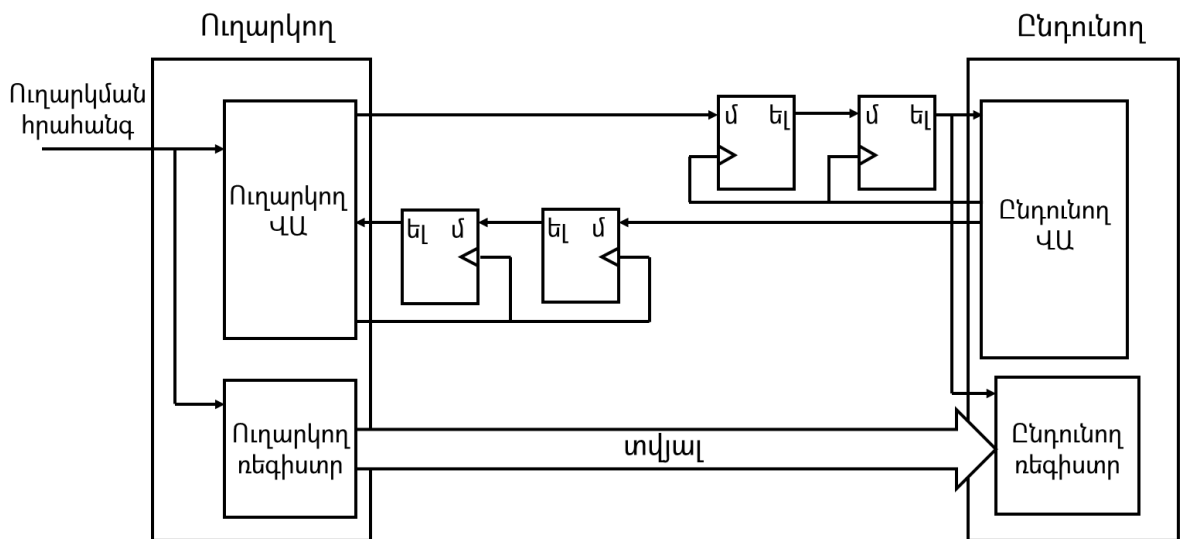
- Որոշ դեպքերում այս ինտերֆեյսում որպես սինքրոազդանշանի գեներատորներ օգտագործվում են օղակաձև գեներատորները, որոնք արտադրության տեխնոլոգիայի, լարման և ջերմաստիճանների տատանումների նկատմամբ

ունեցած զգայունության պատճառով գործնական կիրառության համար նպատակահարմար չեն [66]:

- Սինքրոազդանշանը ընդհատելիս՝ անհնար է ապահովել համակարգի բոլոր տրիգերների միաժամանակյա և ակնթարթային ընդհատումը: Այս հանգամանքը կարող է հանգեցնել սխեմայում ազդանշանների մրցումների և համակարգի ֆունկցիոնալ խափանման [66]:
- Տակտավորող ազդանշանների վերագործարկման ժամանակ հնարավոր են դրանց թրթռոցներ, որոնք կարող են հանգեցնել սինքրոազդանշանի իմպուլսի տևողության տարբեր արժեքների: Դա կարող է հանգեցնել ԻՍ-ի տարբեր հանգույցների մուտքերին սինքրոազդանշանի հապաղման տարբերությունների, ինչն առաջացնում է նախատեսված տրամաբանական գործողությունների խափանման վտանգ [51,66]:

Բարձր արտադրողականություն չպահանջող կիրառություններում ՍՀԻ-ը հաճախ կիրառվող համաձայնեցման միջոց է: Այն բաղկացած է երկու մակարդակային համաձայնեցուցիչներից, որոնք համաձայնեցնում են ղեկավարող ազդանշանները: Այս ինտերֆեյսը (նկ. 1.21)[66] իրականացնում է քառափուլ ՁԱ, որի էությունը հետևյալն է. ազդանշանի՝ սինքրոտիրույթի հատում կատարելուց առաջ ուղարկող սինքրոտիրույթը (ՈԻՍ) ընդունողին (ԸՍ) է ուղարկում ՄՀ-ով համաձայնեցված «դիմում» ազդանշան, որով ակտիվացվում է ընդունող ռեգիստրը, և հաջորդող ճակատի ժամանակ տվյալային կարգերը փոխանցվում են դրան: Կարգերի փոխանցման ավարտից հետո ԸՍ-ն ՈԻՍ-ին է ուղարկում դրանում համաձայնեցվող «պատրաստ» ազդանշանը, որով թույլատրում է կարգերի հաջորդ փաթեթի փոխանցումը [66]: Նկարագրված մոտեցումն ունի առավելագույնը 3 տակտային պարբերության չափով ուշացում, ինչը նվազեցնում է դրա արագագործությունը, սակայն այս մոտեցման հիմնական և կարևորագույն թերությունը դրա ցածր արտադրողականությունն է. տվյալների յուրաքանչյուր հաջորդ փոխանակությունը հնարավոր է ընդունող սինքրոտիրույթի տակտավորող ազդանշանի 12 պարբերությունը մեկ [66]:

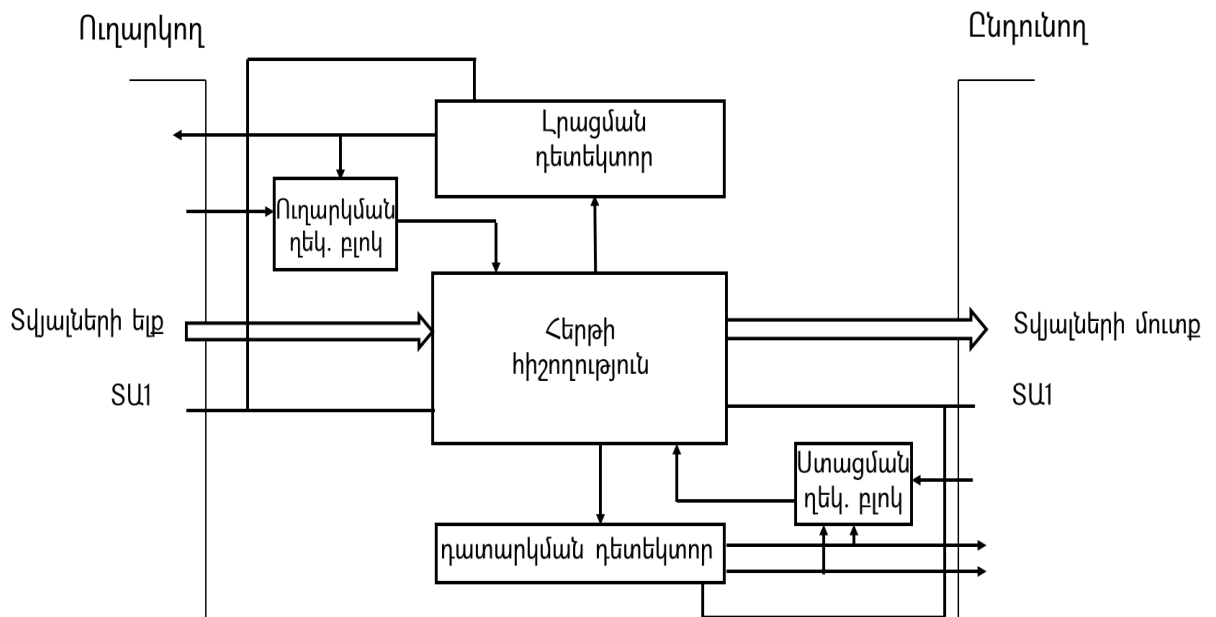
Նշված թերությունները աննպատակահարմար են դարձնում ՍՀԻ-ի կիրառությունը բարձր հաճախականություններով աշխատող ԻՍ-երում:



Նկ. 1.21. Սահմանային համաձայնեցմամբ ինտերֆեյսը

ՀՀ-երը կիրառելի են համեմատաբար բարձր արագագործություն պահանջող տվյալների փոխանցման դեպքում: [66] – ում նկարագրված ՀՀ-ն (նկ. 1.22) [66] ունի ցիկլիկ տիպի հերթ և աշխատում է երկու սինքրոազդանշանով, որոնցից մեկը սինքրոնացնում է ՌԻՍ-ն, մյուսը՝ ԸՍ-ն: ՌԻՍ-ը տվյալները տեղադրում է հերթի վերջում, իսկ ընդունողը վերցնում է այն հերթի սկզբից: Այսինքն, ՀՀ-ն երկինտերֆեյս համաձայնեցման միջոց է, որն ունի ուղարկման և ստացման առանձնացված բլոկներ: Այս մեխանիզմը [66] թույլ է տալիս իրարից մեկուսացնել հերթի մեջ գրելու և հերթից կարդալու գործողությունները և դրանք ղեկավարել համապատասխան սինքրոազդանշաններով: Հերթի մեջ տվյալների կարգերի մուտքագրման և ընթերցման գործողությունների՝ տարբեր տակտավորող ազդանշաններով ղեկավարվելու պատճառով ՌԻՍ-ի և ԸՍ-ի նկատմամբ ասինքրոն են համապատասխանաբար հերթի դատարկ և լրացված լինելն ազդարարող ազդանշանները [66]: Այդ իսկ պատճառով նշված ազդանշանները սինքրոտիրույթների հետ համաձայնեցվում են ՄՀ-երի միջոցով: Սակայն ՄՀ-երի առաջացրած ուշացումները հանգեցնում են ՌԻՍ-ի և ԸՍ-ի միջոցով լրացման և դատարկ վիճակի ազդանշանների հապաղումների՝ առաջացնելով հնարավոր որոշ խնդիրներ:

Դրանք կարող են լինել՝ լրացված հերթի մեջ մուտքագրելը [66] և դատարկ հերթից ընթերցելը [66]: ՀՀ-ի հիմնական առավելությունը մեծ արտադրողականությունն է, որը դանդաղ սինքրոտիրույթի յուրաքանչյուր տակտային պարբերության ընթացքում 1 կարգային փաթեթի փոխանակումն է:



Նկ. 1.22. Համաձայնեցման հերթահենք ինտերֆեյս

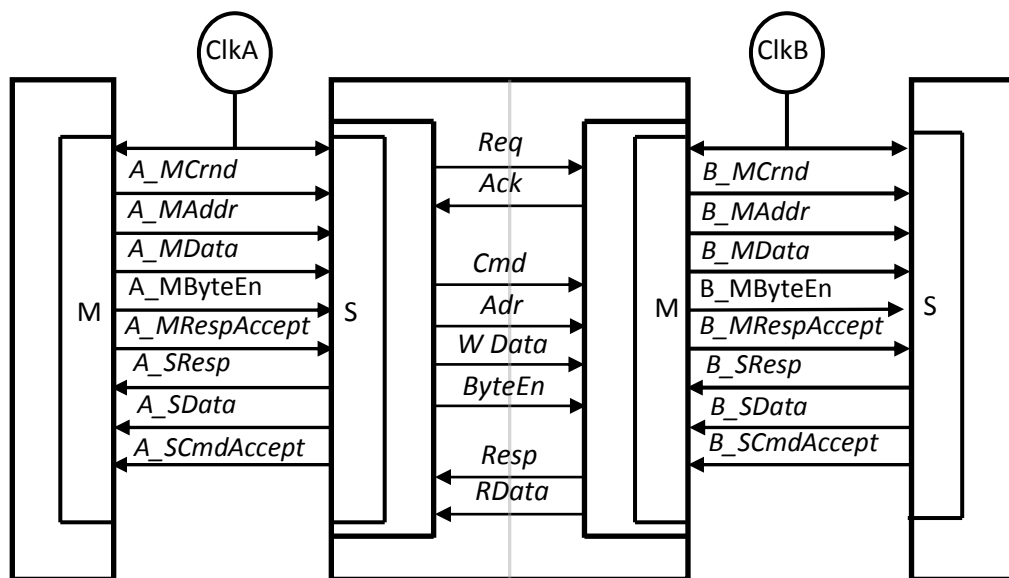
ՀՀԻ-ի կիրառության առաջին սահմանափակումն այն է, որ տվյալների փաթեթների փոխանցումը հնարավոր է միայն եթե հերթը երբեք չի դառնում դատարկ կամ լրացված: ՀՀԻ-ի հիմնական թերություններն են [66].

- 3-4 տակտային պարբերությանը հավասար մուտք-ելք հապաղումը: Արագագործ կիրառություններում այս թերությունը դառնում է էական, իսկ դրանցում ՀՀԻ-ի կիրառությունը՝ աննպատակահարմար:
- Սինքրոտիզմի տակտավորման հաճախականությունների առավելագույն հարաբերությունը 3 է: Դրանից մեծ հարաբերությունները հանգեցնում են ՀՀԻ-ի խափանման:
- Մեծ մակերեսը, որը, հերթի հիշողության մեծացմանը զուգահեռ, աճում է:
- Ի տարբերություն սինքրոազդանշանի ընդհատման մոտեցման՝ ՀՀԻ-ի դեպքում կա մետակայուն ազդանշանի՝ ընդունող սինքրոտիզմի ներթափանցելու հավանականություն:
- ՀՀԻ-ի առավելությունը՝ բարձր արտադրողականությունը, դառնում է թերություն երբ սինքրոտիզմի տակտավորման հաճախականությունների հարաբերությունը մեծ է այս մոտեցմանը բնորոշ չափից:

[73]-ում ներկայացված է բաց միջուկի արձանագրություն (ԲՄԱ)՝ արձանագրությամբ հաղորդակցվող բլոկների համաձայնեցման ինտերֆեյս (ՀԻ) (նկ.

1.23) [73]: ԲՄԱ-ը ստանդարտ է [73], որն ուղղված է համակարգային ԻՍ-ում (<ԻՍ> օգտագործված տարբեր ՄՍ-երի ճշգրիտ հաղորդակցությունն ապահովելուն՝ թույլ տալով նվազեցնել նախագծման ժամանակը, ռիսկերը և հետևաբար՝ <ԻՍ>-ի գինը [74]:

Այն հիմնված է աշխատանքի «տեր-ծառա» սկզբունքի վրա [73] և ապահովում է հանգույցից հանգույց հաղորդակցություն, որի դեպքում գրելու կամ ընթերցման դիմում կարող է ուղարկվել միայն «տեր» հանգույցից «ծառա» հանգույց [73]: Ենթադրվում է նաև [73], որ նշված երկու հանգույցները տակտավորվում են միևնույն սինքրոնիզացիայով: Տարբեր սինքրոնիզացիաներում գործող «տեր-ծառա» հաղորդակցվող հանգույցները անհրաժեշտ է լրացուցիչ համաձայնեցնել (նկ. 1.23) [73]:

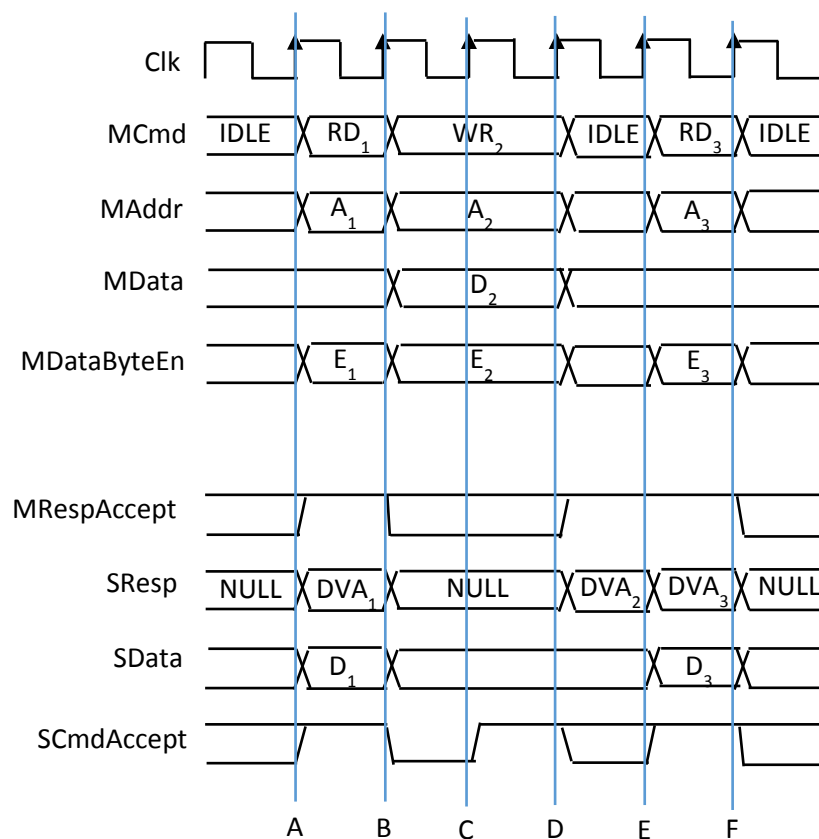


Նկ. 1.23. ԲՄԱ արձանագրությամբ հաղորդակցվող բլոկերի համաձայնեցման ինտերֆեյսը

Համակարգի արագագործության վրա համաձայնեցման ինտերֆեյսի ունեցած ազդեցությունը նվազագույնին հասցնելու համար [73]-ում հաղորդակցման երկու ուղղություններից յուրաքանչյուրում համաձայնեցվել է մեկական ազդանշան: Համաձայնեցման ինտերֆեյսը մշակվել է ԲՄԱ-ի ԲՄԱio ենթատեսակի հիման վրա [73], որը հնարավորություն է տալիս միաժամանակ կարդալ կամ գրել միայն մեկ տվյալային բառ: Այն, հիմնականում, օգտագործվում է <ԻՍ>-երում՝ պրոցեսորների հաղորդակցությունը հիշող սարքերի և մուտքի-ելքի հանգույցների հետ իրականացնելու համար [73]: ԲՄԱio ինտերֆեյսում MCrnd-ը և SCmdAccept-ը ղեկավարող ազդանշաններ են, որոնք վերահսկում են հրամանային բառի և

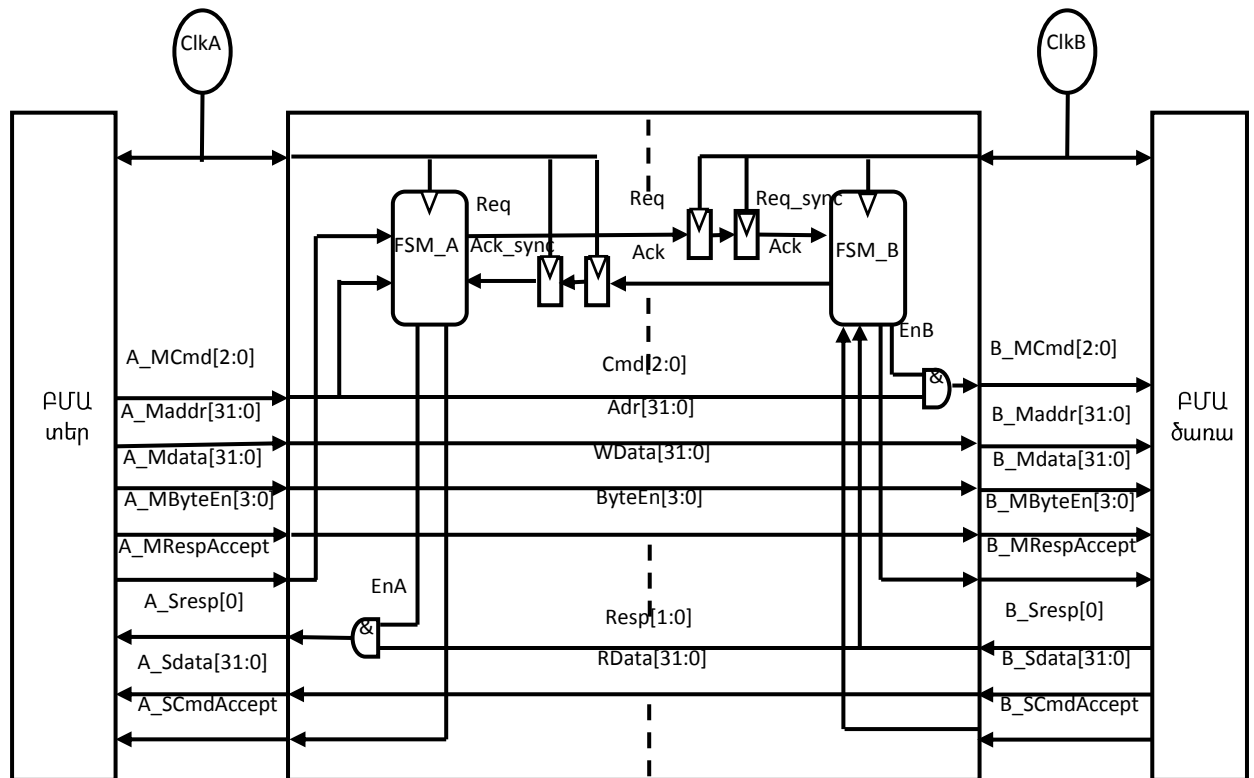
գրանցման համար նախատեսված տվյալների՝ MData ազդանշանի փոխանցումը, իսկ SResp-ն ու MRespAccept-ը՝ ընթերցվող SData-ինը (նկ. 1.24) [73]: Նկ. 1.24-ում MAddr-ը գրանցման կամ ընթերցման բջջի հասցեն է:

Մշակվել է ԲՄԱio արձանագրության վրա հիմնված համաձայնեցման ինտերֆեյսի երկու տեսակ: Դրանցից առաջինը (նկ. 1.25) [73] չի իրականացնում փոխանցվող տվյալների բուֆերացում և դրա հաշվին զբաղեցնում է ավելի փոքր մակերես: Ինտերֆեյսի մյուս տեսակը (նկ. 1.26) [73] բուֆերացնում է տվյալները՝ ցուցաբերելով նվազագույն ՄԵՀ [73]: Req և Ack ազդանշանները (նկ. 1.25) համաձայնեցվում են ՄՀ-ի միջոցով՝ քառափուլ համաձայնեցման եղանակով [73]:



Նկ. 1.24. ԲՄԱio ինտերֆեյսի աշխատանքը

Նկ. 1.25-ում պատկերված է նաև, որ հասցեի և տվյալային ազդանշանները մի սինքրոտիզացիոն մյուսը փոխանցվում են առանց բուֆերացման [73], իսկ ղեկավարող ազդանշանները գեներացվում կամ մասնակիորեն կառավարվում են վերջավոր ավտոմատներով (ՎԱ) [73]: ԲՄԱ-տեր հանգույցի կողմից տվյալների փոխանակման նախաձեռնումը նկ. 1.25-ում պատկերված համակարգի բոլոր ղեկավարող ազդանշանները և ՎԱ-երը գտնվում են սպասման ռեժիմում:



Նկ. 1.25. Առանց տվյալների բուֆերացման ԲՄԱio համաձայնեցման ինտերֆեյսը

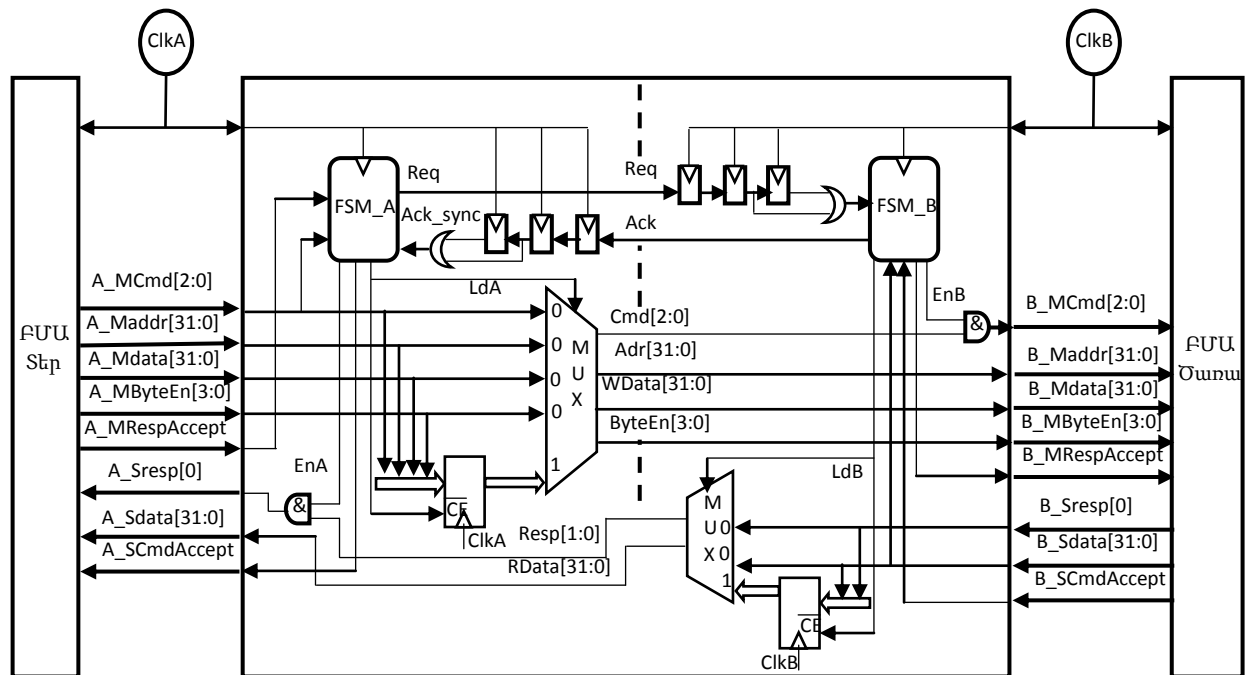
Տվյալների փոխանակման համար ԲՄԱ-տեր հանցույցը A_Mcmd ազդանշանին վերագրում է գրելու կամ ընթերցման գործողություններին համապատասխանող արժեք, որից հետո FSM_A-ն ակտիվացնում է Req ազդանշանը [73]: Req_sync ազդանշանն ընդունելուց հետո FSM_B ՎԱ-ն, ակտիվացնելով EnB ազդանշանը, թույլատրում է տվյալների գրանցումը կամ ընթերցումը [73]:

Տվյալների բուֆերացմամբ ԲՄԱio համաձայնեցման ինտերֆեյսի (նկ.1.26) աշխատանքի սկզբունքը [73] շատ նման է բուֆերացում չկատարող ինտերֆեյսի աշխատանքին, սակայն այս դեպքում ԻԳ համաձայնեցուցիչների միջոցով իրագործված է Req և Ack ազդանշանների երկփուլ համաձայնեցում [73]:

Նկարագրված ինտերֆեյսները ՀԻՍ-ում զբաղեցրած մակերեսներով էականորեն չեն տարբերվում, դրանց հիմնական տարբերությունը արագագործությունն է [73]: Տվյալների՝ անմիջապես իրար հաջորդող փոխանցումների դեպքում [73]-ում կատարված բոլոր չափումները ցույց են տվել, որ բուֆերացնող ինտերֆեյսը մոտ երկու անգամ ավելի փոքր ՄԵՀ ունի քան բուֆերացում չիրականացնողը:

[73]-ում առաջարկված երկու ինտերֆեյսները ՀՀԻ-ների համեմատ ունեն մակերեսի և արտադրողականության առավելություն: Դրանց թերությունն է առնվազն

3 տակտային պարբերության չափով ՄԵՀ-ը և այն, որ չի քննարկվում Req և Ack ազդանշանների համաձայնեցուցիչների ելքերում մետակայունության առաջացման հավանականությունը:

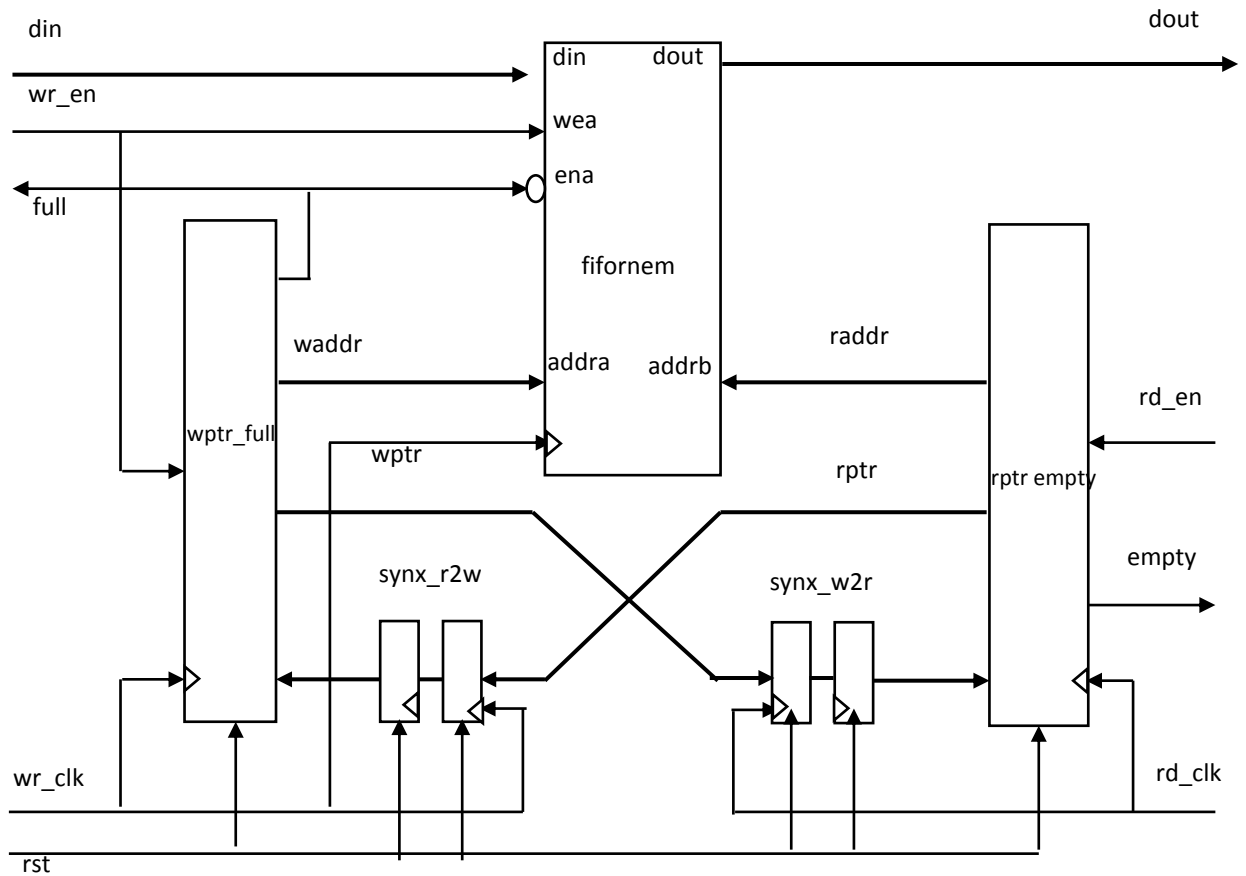


Նկ. 1.26. Տվյալների բուֆերացմամբ ԲՄԱio համաձայնեցման ինտերֆեյսը

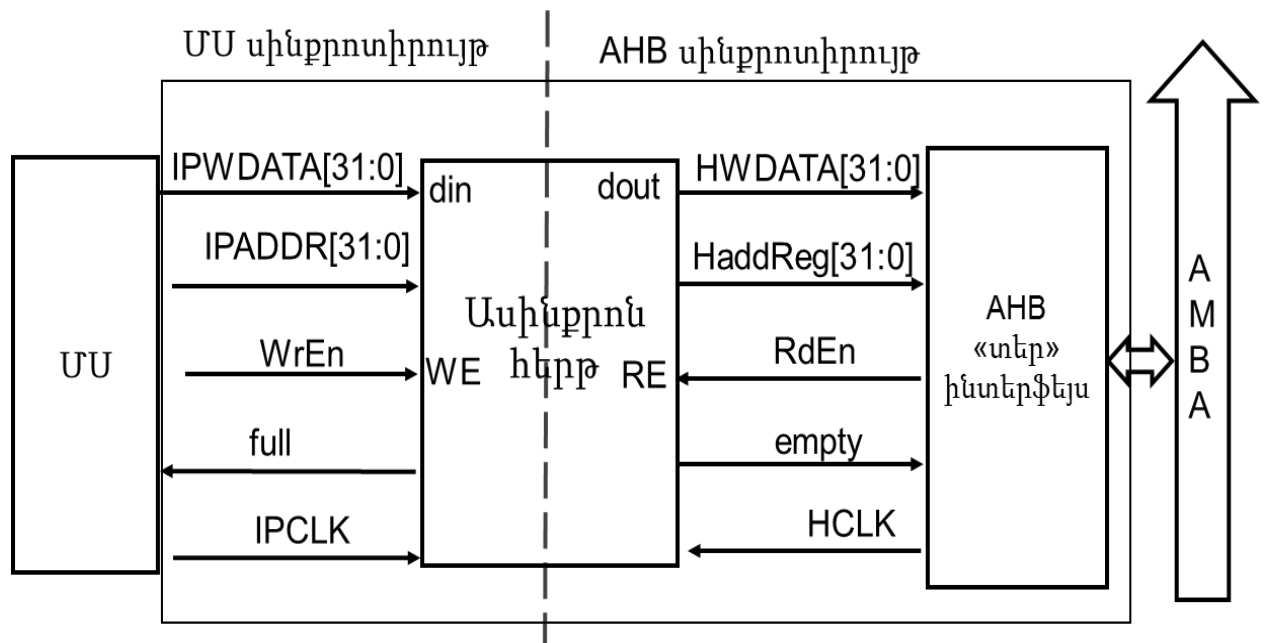
Վերջինիս տեղի ունենալու դեպքում համապատասխան ՎԱ-ն ևս կհայտնվի մետակայուն վիճակում խափանելով ամբողջ համակարգի ֆունկցիոնալությունը:

Մեկ այլ աշխատանքում [74] մշակվել են AHB և ԲՄԱ արձանագրությունների վրա հիմնված համաձայնեցման ինտերֆեյսներ, և կատարվել է դրանց արտադրողականության գնահատում և համեմատություն: AHB ինտերֆեյսի նախագծումը բաժանված է երկու մասի [74]՝ գրելու և կարդալու գործողությունների համաձայնեցման համար: Դրա պատճառն այն է, որ ՄՍ բլոկը, կախված գործողությունից, հանդես է գալիս կամ որպես «տեր», կամ որպես «ծառա»: Համաձայնեցման համար օգտագործվում է ասինքրոն հերթ (նկ. 1.27) [74], որի գրելու և ընթերցման ցուցիչները (wptr և rptr) սինքրոտիզացիայի հատում կատարելու պատճառով համաձայնեցվում են ՄՀ-երով:

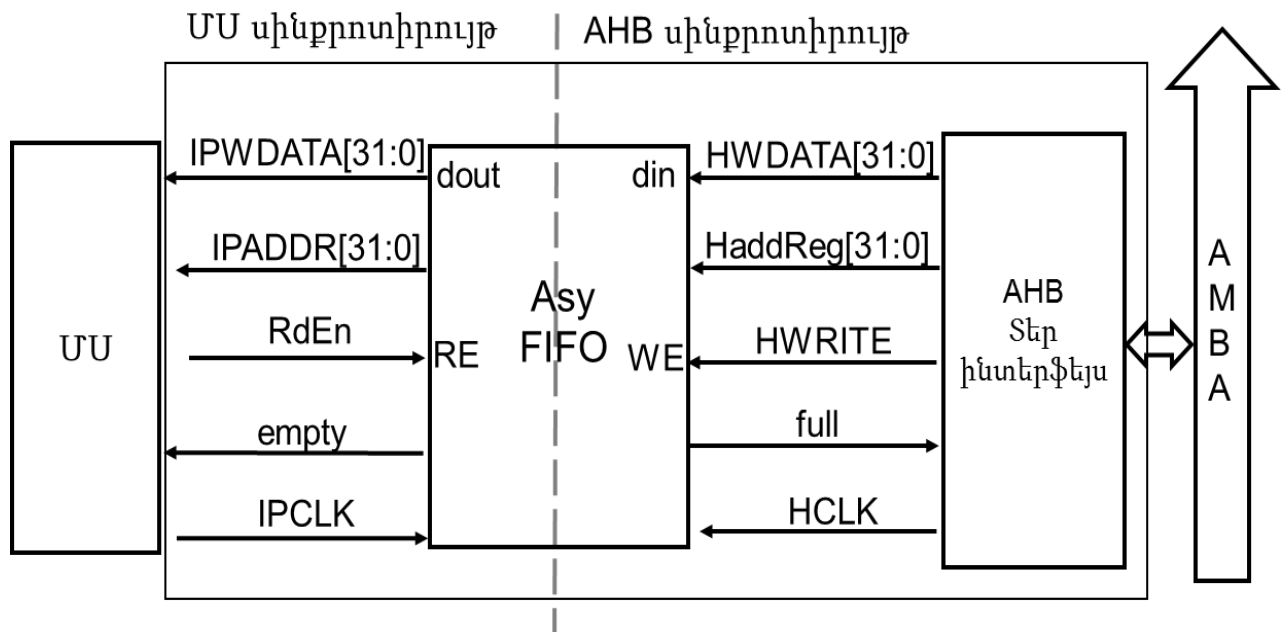
Գրելու գործողության ժամանակ ՄՍ-ն հանդես է գալիս որպես «տեր» հանգույց (նկ. 1.28) [74], որը տվյալներն ուղարկում է AHB դոդին, իսկ ընթերցման գործողության ժամանակ AHB դոդը տվյալները փոխանցում է «ծառա» հանգույց հանդիսացող ՄՍ-ին (նկ. 1.29) [74]:



Նկ. 1.27. Ասինքրոն հերթի սխեման

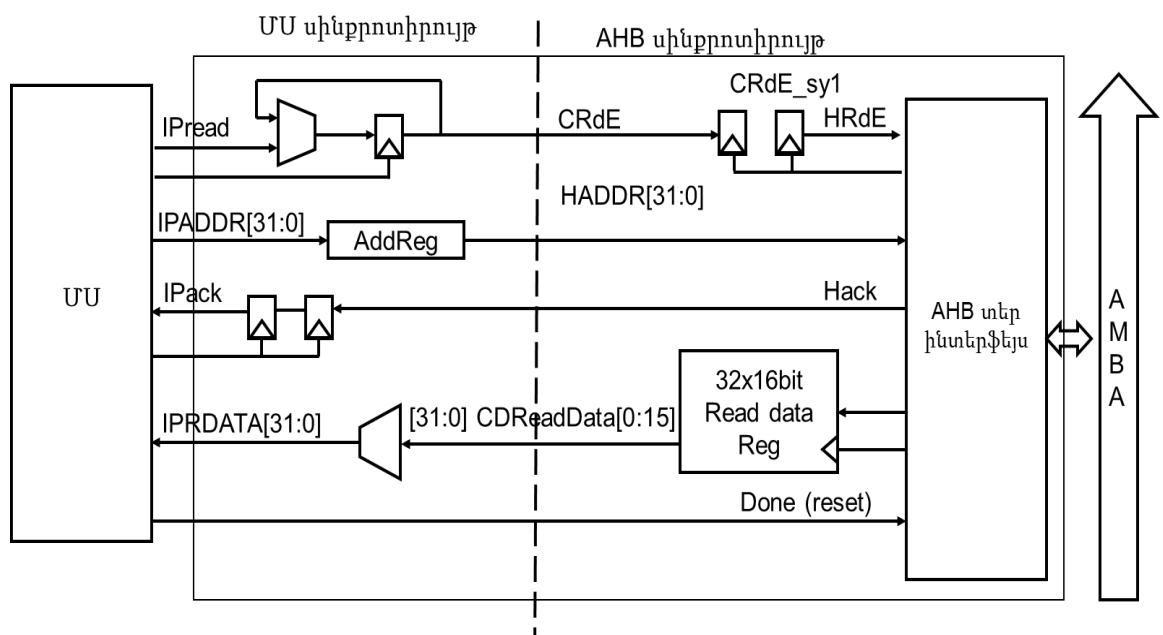


Նկ. 1.28. ՄՄ բլոկի «տեր» հանգույց լինելու դեպքում միջասինքրոտիրույթային գրանցման գործողության համաձայնեցումը

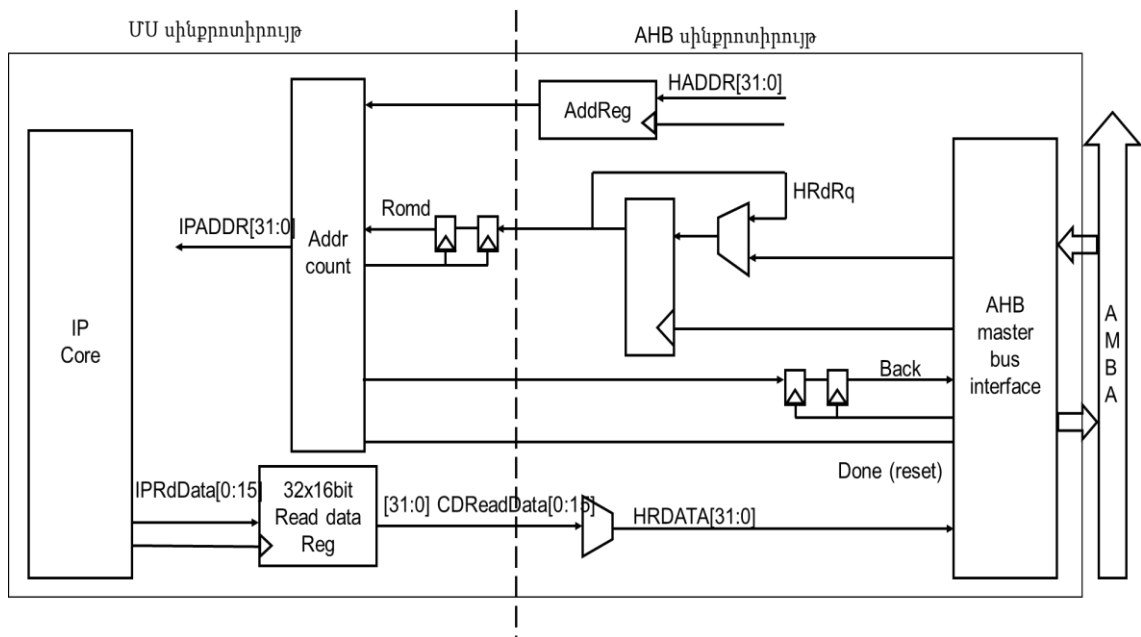


Նկ. 1.29. ՄՄ բլոկի «ծառա» հանգույց լինելու դեպքում միջսինքրոտիզացիային գրանցման գործողության համաձայնեցումը

Ընթերցման գործողությունը ավելի բարդ իրականացվող է, քանի որ այն երկուղված է [74]: Այս դեպքում օգտագործվում են ՄՀ-ով սինքրոնացվող Req և Ack դեկլարող ազդանշաններ: Այս գործողության ժամանակ AHB-ից տվյալներ ստացող ՄՄ-ն հանդիսանում է «տեր» հանգույց (նկ. 1.30) [74], եթե տվյալները ստանալու հարցումը դա է նախաձեռնել, և «ծառա» հանգույց (նկ. 1.31) [74]՝ AHB-ի հարցման դեպքում:

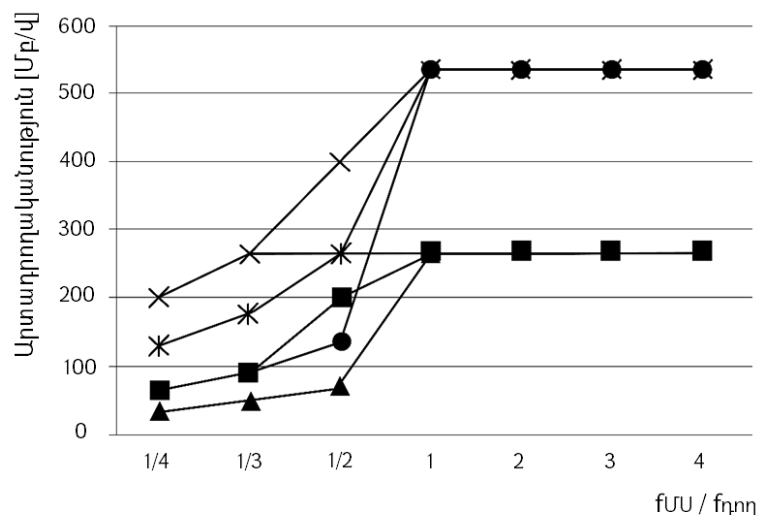


Նկ. 1.30. ՄՄ բլոկի՝ «տեր» հանգույց լինելու դեպքում ընթերցման գործողության համաձայնեցումը

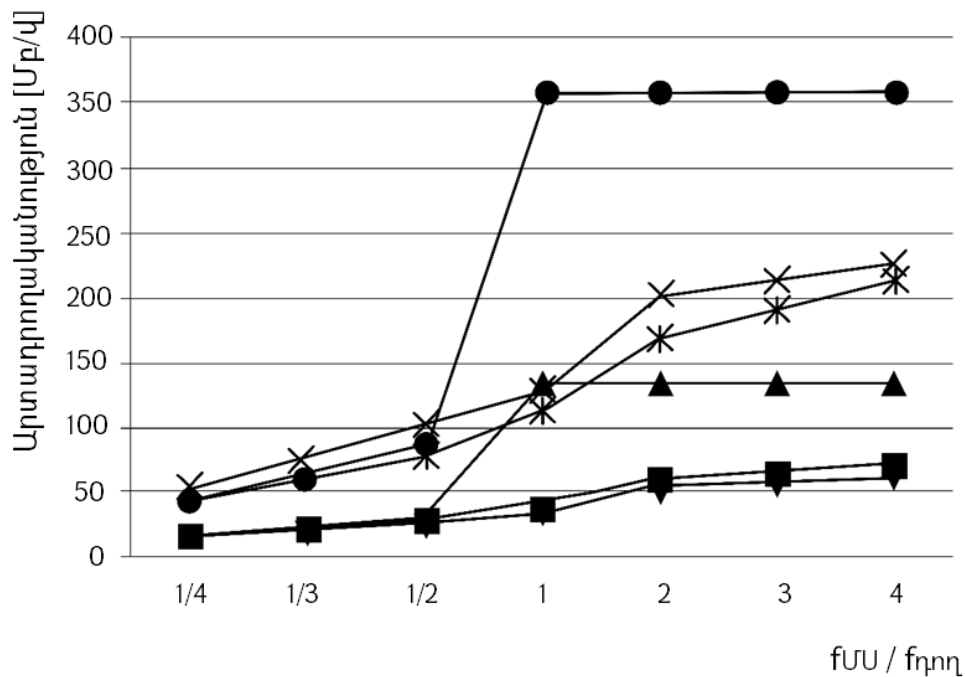


Նկ. 1.31. ՄՍ բլոկի՝ «ճառա» հանգույց լինելու դեպքում ընթերցման գործողության համաձայնեցումը

Մշակվել է նաև ԲՄԱ արձանագրության հիման վրա AHB ինտերֆեյս [74]: Կատարվել է վերջինիս և AHB-ի վրա հիմնված ինտերֆեյսների գրանցման (նկ. 1.32) [74] և ընթերցման (նկ. 1.33) [74] գործողությունների արտադրողականությունների համեմատություն: Զբաղեցրած մակերեսի տեսանկյունից ԲՄԱ հիմքով ինտերֆեյսը 40000-ից ավելի փականներով [74] ավելի մեծ մակերես է զբաղեցնում, քան 28000 փական ունեցող AHB հիմքով [74]:



Նկ. 1.32. Գրանցման գործողության դեպքում AHB և ԲՄԱ հիմքով համաձայնեցման ինտերֆեյսների արտադրողականությունների՝ ՄՍ բլոկի և դրդի հաճախականային հարաբերությունից կախումը



Նկ. 1.33. Ընթերցման գործողության դեպքում AHB և ԲՄԱ հիմքով համաձայնեցման ինտերֆեյսների արտադրողականությունների՝ ՄՍ բլոկի և դողի հաճախականային հարաբերությունից կախումը

Այսպիսով, համաձայնեցման առկա տարբեր միջոցների մի մասը տրիգերային համաձայնեցուցիչներ էին, մյուսները՝ ավելի բարդ կառուցվածք ունեցող և որոշակի արձանագրություն իրականացնող համաձայնեցման ինտերֆեյսներ: Դրանց վերլուծությունը ցույց է տալիս, որ վերջիններս չեն ապահովում ԳԱՏՍ ԻՍ-երին առաջադրվող արագագործության, էներգասպառման, արտադրողականության և մակերեսի պահանջները: Դա նշանակում է, որ առաջացել է նոր սկզբունքներով գործող համաձայնեցման միջոցների մշակման ծայրահեղ անհրաժեշտություն:

1.3. Տարբեր հաճախություններով տակտավորվող թվային սխեմաների համաձայնեցման միջոցներին առաջադրվող պահանջները և սկզբունքները

ԳԱՏՍ ԻՍ-երին առաջադրված արագագործության, էներգասպառման, արտադրողականության և մակերեսի պահանջվող ցուցանիշների ստացման համար անհրաժեշտ է կառուցել նոր տիպի համաձայնեցման սխեմաներ, որոնց առաջադրվում են ստորև շարադրված պահանջները: Ասինքրոն ԹՍ-երում մետակայունության

առաջացման հավանականությունը միշտ առկա է եղել: Մինչև 90 նմ տեխնոլոգիան [1,4], ԻՍ-երում տրամաբանական փականների հապաղմները և տրանզիստորների բնութագրերը գործընթացի, սնուցման լարման և ջերմաստիճանի (ԳԼՋ) փոփոխություններից ունեին համեմատաբար թույլ կախում [1]: Դրա շնորհիվ հնարավոր էր սխեմատեխնիկական որոշակի լուծումների միջոցով խուսափել մետակայուն վիճակից [69]: Տրանզիստորների չափերի փոքրացման, սնուցման լարումների նվազեցման և ԻՍ-երի աշխատանքային հաճախականությունների աճման հետևանքով սարքերի բնութագրերի կախումը ԳԼՋ-ից դարձել է շոշափելի [1,3,4]: Վերջինս հանգեցնում է ԹՍ-ում տրամաբանական փականների հապաղումների՝ ԳԼՋ-ից կախված փոփոխությունների [9,10]: Փոփոխական հապաղումները պատճառով ԳԱՏՍ համակարգում հաղորդակցվող սինքրոտիզմների ընդունող տրիգերների ժամանակային սահմանափակումները կարող են խախտվել [69]՝ առաջացնելով մետակայուն վիճակ: Մետակայունության առաջացման հավանականության աճին են նպաստում ցածր էներգասպառմամբ նախագծման մեթոդները, մասնավորապես, սնուցման լարման և հաճախականության դինամիկ մասշտաբավորումը [41,44,45]: ԳԱՏՍ համակարգերում սինքրոտիզմների մետակայունությունից խուսափելու համար օգտագործում են համաձայնեցնող սխեմաներ (ՀՍ): 1.2-ում քննարկված ՀՍ-երից յուրաքանչյուրը մյուսների նկատմամբ ուներ առավելություններ և թերություններ, որոնք ազդում էին ընդհանուր համակարգի մակրոբնութագրերի վրա: Ժամանակակից ԻՍ-երին առաջադրվող արագագործության, ցածր էներգասպառման, բարձր աշխատանքային հաճախականությունների պահանջները և կիրառվող նախագծման մեթոդները բավարարող ՀՍ-երի կարևորագույն բնութագրերն են [2].

- Մուտք-ելք հապաղումը. ՀՍ-երը սինքրոտիզմների միջև ավելացնում են լրացուցիչ հապաղում, որն ազդում է համակարգի արագագործության և արտադրողականության վրա:
- Առավելագույն արտադրողականությունը. մեծ արտադրողականություն ունեցող ՀՍ-երը կարող են կիրառվել տարբեր արագագործություն պահանջող ԻՍ-երում՝ դառնալով որոշակիորեն համապիտանի համաձայնեցման միջոց:

- Զբաղեցրած մակերեսը:
- Սինքրոտրոնային հաճախականային հարաբերությունը. այս մեծությունը ցույց է տալիս, թե տվյալ ՀՄ-ն ինչ հաճախականային հարաբերությամբ սինքրոտրոնային կարող է համաձայնեցնել,
- Կիրառելիության հաճախականային միջակայք. տվյալ ՀՄ-ով համաձայնեցվող հնարավոր հաճախականություններն են:
- Համապիտանիությունը. այս մեծությունը որոշում է ՀՄ-ի կիրառելիության տիրույթը:

Համաձայնեցման միջոցների վերոհիշյալ բնութագրերը հաշվի առնելով՝ դրանց առաջ դրվում են հետևյալ պահանջները.

1. Հնարավորինս փոքր, ազդանշանի վիճակից անկախ համաձայնեցման հապաղում: Համաձայնեցման առկա լուծումներում նվազագույն հապաղումը կազմում է սինքրոազդանշանի 2 պարբերությանը հավասար մեծություն, և ազդանշանի բնականոն ու մետակայուն վիճակների հապաղումները տարբեր արժեքներ ունեն:
2. Փոքր սեփական մակերես: Համաձայնեցման միջոցի սեփական մակերեսի մեծությունը պետք է լինի այնպիսին, որ էապես չմեծացնի ԻՄ-ի ընդհանուր մակերեսը:
3. Հնարավորինս մեծ հաճախականային և կարգային համապիտանիություն: Անհաժեշտ է, որ համաձայնեցման միջոցը կարողանա համաձայնեցնել հնարավորինս մեծ քանակով հաճախականություններ և ունենա միակարգ և բազմակարգ ազդանշանների անխափան համաձայնեցման ունակություն, ինչը կբարձրացնի դրա վերաօգտագործելիությունը և կնվազեցնի գինը:

Ատենախոսությունում առաջարկվում են տարբեր հաճախություններով տակտավորվող թվային սխեմաների համաձայնեցման նոր սկզբունքներ, որոնք կապահովեն գործնական պահանջները բավարարող հապաղում, կիսահաղորդչային բյուրեղի վրա զբաղեցրած փոքր մակերես և հաճախականային ու կարգային համապիտանիություն:

Մշակված է տարբեր հաճախությունների համաձայնեցման երկու նոր մեթոդ: Դրանցից առաջինը հայտնաբերում է ազդանշանի մետակայությունը և ԸՍ-ին փոխանցում է վերականգնված ճիշտ տվյալային կարգը՝ ապահովելով բնականոն և մետակայուն ազդանշանների միևնույն՝ մեկ տակտային պարբերության հավասար հապաղում: Երկրորդ մեթոդն իրականացնում է սինքրոազդանշանի համաձայնեցում՝ գեներացնելով ԸՍ ՍԱ-ի տարբեր փուլային շեղումներով ազդանշաններ և դրանցից ընտրելով ՌԻՍ ՍԱ-ին ամենահամապատասխան ազդանշանը:

Եզրակացություններ

1. Ժամանակակից ԻՍ-երի արագագործության, զբաղեցրած փոքր մակերեսի, ցածր էներգասպառման և դրանց անխափան գործունեության ապահովման պահանջները թելադրում են տարբեր հաճախություններով տակտավորվող թվային սխեմաների համաձայնեցման միջոցների մշակման անհրաժեշտությունը:
2. Համաձայնեցման առկա միջոցների վերլուծությունը ցույց է տալիս, որ դրանք չեն բավարարում գլոբալ ասինքրոն տեղային սինքրոն ինտեգրալ սխեմաների արագագործությանը և զբաղեցրած մակերեսին առաջադրվող պահանջները, ինչով պայմանավորված է նոր սկզբունքներով գործող համաձայնեցման միջոցների մշակման ծայրահեղ կարևորությունը:
3. Ձևակերպվել են տարբեր հաճախություններով տակտավորվող թվային սխեմաների համաձայնեցման միջոցներին առաջադրվող պահանջները, որոնց ապահովումը կապահովի ժամանակակից ինտեգրալ սխեմաներին առաջադրվող պահանջների բավարարումը:

ԳԼՈՒԽ 2. ՏԱՐԲԵՐ ՀԱՃԱԽՈՒԹՅՈՒՆՆԵՐՈՎ ՏԱԿՏԱՎՈՐՎՈՂ ԹՎԱՅԻՆ ՍԽԵՄԱՆԵՐԻ ՀԱՄԱՁԱՅՆԵՑՄԱՆ ԱՌԱՋԱՐԿՎՈՂ ՄԻՋՈՑՆԵՐԸ

Ստորև նկարագրված են մշակված համաձայնեցման մեթոդներն ու միջոցները [75, 76], որոնց կիրառումը ԳԱՏՍ ԻՍ-երում թույլ կտա, առանց մակերեսի և էներգասպառման էական փոփոխությունների, համաձայնեցնել տարբեր հաճախություններով տակտավորվող թվային սխեմաներ՝ ապահովելով սինքրոտիզմի միջև հնարավորինս փոքր հապաղում և բացառելով մետակայուն վիճակի առաջացումը: Առաջարկվող մեթոդները արտադրման տեխնոլոգիական գործընթացից անկախ են և կիրառելի ուղարկող ու ընդունող սինքրոտիզմի տակտավորման հաճախականությունների մեծ քանակով արժեքների դեպքում:

Համաձայնեցման առկա միջոցների թերությունները հաշվի առնելով՝ մշակվել են խնդրի լուծման նոր մեթոդներ, որոնք ասինքրոն սխեմաները համաձայնեցնում են հայտնի միջոցներից [65,66] սկզբունքային տարբերություններ ունեցող եղանակներով:

Սինքրոտիզմի մեթոդները համաձայնեցնելու նպատակով առաջարկվել է երկու մեթոդ՝

- ազդանշանի մետակայուն վիճակի հայտնաբերմամբ և դրա հետևանքով ազդանշանի կորցված արժեքի վերաթողարկմամբ,
- սինքրոազդանշանների դինամիկ համաձայնեցմամբ:

Վերոհիշյալ մեթոդների հիման վրա մշակվել են թվային համակարգեր [75, 76], որոնք, բացառելով մետակայուն վիճակի առաջացումը, ապահովում են ԻՍ-ում հաղորդակցվող սինքրոտիզմի անխափան աշխատանքը:

Առաջարկված մեթոդների գործնական կիրառմամբ համաձայնեցվել են OpenSPARC T1 [77], OpenSPARC T2 [77] և ORCA [78] պրոցեսորների հաղորդակցվող սինքրոտիզմի մեթոդները: Մշակված համաձայնեցման մեթոդներով և հերթահենք մեթոդով [66] նշված համակարգերի սինքրոնացումից հետո համեմատվել է պրոցեսորների փականների ընդհանուր քանակը, որից կախված են համակարգի մակերեսը, նաև համաձայնեցման հապաղումը: Ստացված արդյունքների հիման գնահատվել է համաձայնեցման իրականացված մեթոդների արդյունավետությունը:

2.1. ԳԼՋ շեղումները հաշվի առնելու անհրաժեշտությունը տարբեր հաճախություններով տակտավորվող թվային սխեմաների համաձայնեցման առաջարկվող միջոցներում

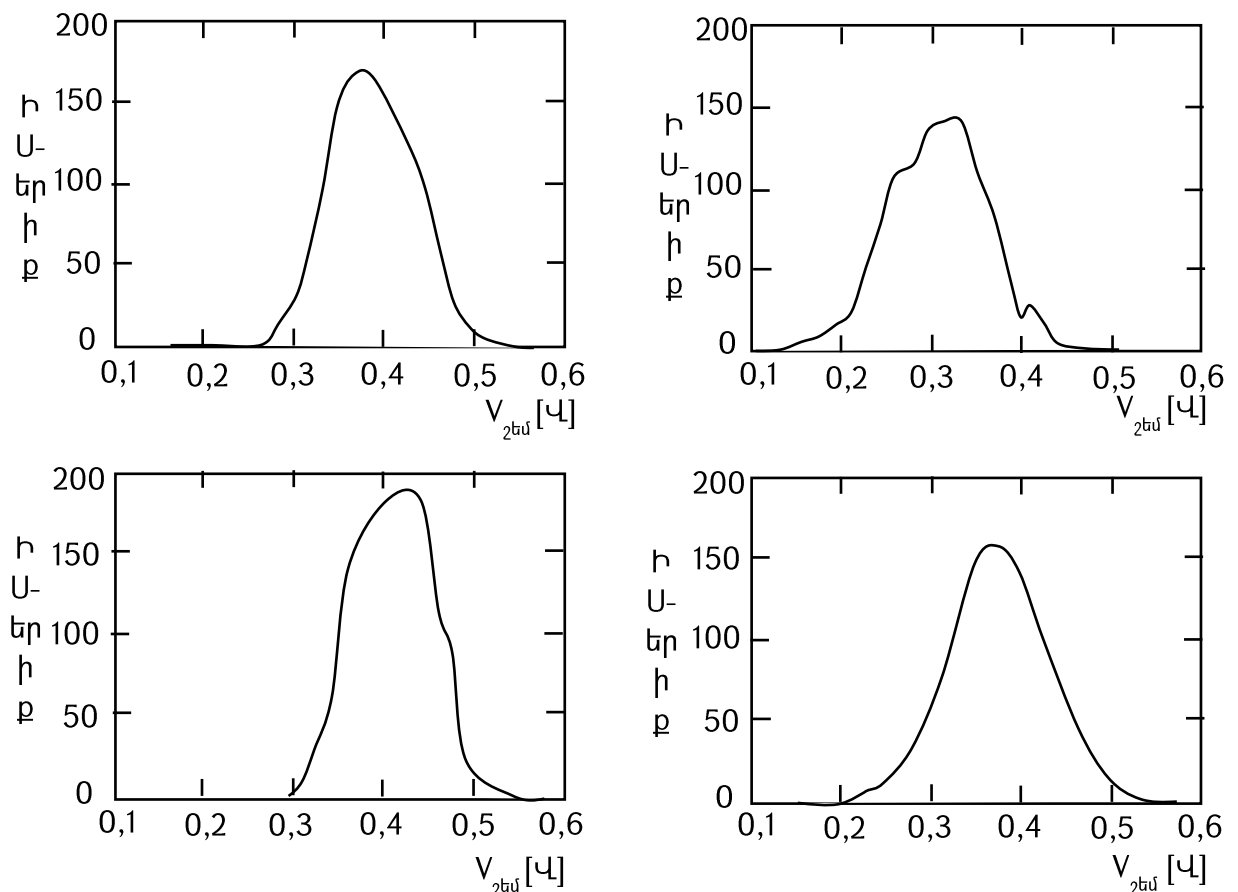
Ինչպես նկարագրված էր 1.1-ում, ժամանակակից ԻՍ-երում էներգասպառումը նվազեցնելու նպատակով տարբեր արագագործություն պահանջող և տարբեր բեռնվածություններով աշխատող ենթակառուցվածքները տակտավորվում են տարբեր հաճախականություններով սինքրոնազդանշանների միջոցով [48], ինչի պատճառով ԻՍ-ը բաժանվում է սինքրոտիրոյթների: Յուրաքանչյուր սինքրոտիրոյթի սահմանային տարրերը ճակատով ղեկավարվող տրիգերներ են: ԳԱՏՍ համակարգերում հաղորդակցվող սինքրոտիրոյթների միջև տվյալների ճշգրիտ և այն անխափան փոխանակումը տեղի է ունենում, երբ ուղարկված ազդանշանն ընդունող տրիգերն ընդունում է իր իսկ հաստատման ժամանակից շուտ և անփոփոխ է մնում դրա սինքրոնազդանշանի ղեկավարող ճակատից հետո պահպանման ժամանակի ընթացքում [49]: Տրիգերի ժամանակային սահմանափակումները բավարարելու նպատակով իրականացվում է տվյալների փոխանցման համապատասխանեցում [54, 55]: Փականային մակարդակի մոդելավորման միջոցով հավաստվում է սինթեզված սխեմայի աշխատանքի ճշտությունը: Սակայն իրական սխեմայում առաջացող անցանկալի երևույթները, որոնք հաշվի չէին առնվել վերջինիս մոդելավորման ժամանակ, վտանգ են ներկայացնում ԻՍ-ի անխափան աշխատանքի համար:

Հայտնի է [35], որ արտադրված ԻՍ-երում ի հայտ են գալիս երևույթներ, որոնց ազդեցությունը սխեմայի աշխատանքի վրա կամ դժվար է կանխատեսել, կամ առհասարակ հնարավոր չէ: Արտադրական գործընթացի անկատարությունները, լարման կամ ջերմաստիճանի տատանումները բացասական ազդեցության հիմնական գործոններն են [35]:

Արտադրական գործընթացի շեղումները (ԱԳՇ) [79] դրսևորվում են կիսահաղորդչային սարքերի իրական և իդեալական ֆիզիկական տեսքերի կամ չափերի տարբերություններով և առաջանում են սարքավորումների, կիսահաղորդչային նյութերի և գործընթացի իրականացման ոչ միօրինակությունների պատճառով [78]: ԱԳՇ-երը կարող են հանգեցնել սարքերի էլեկտրական բնութագրերի էական

փոփոխությունների [79]՝ նվազեցնելով ԻՍ-ի արտադրողականությունը և նույնիսկ առաջացնելով ֆունկցիոնալ խափանումներ: Նշված շեղումները կարող են լինել պարբերական կամ պատահական [80]: Պարբերական բնույթի շեղումները հիմնականում պայմանավորված են արտադրական գործընթացում կիրառվող սարքավորումների իրականացրած գործողությունների՝ ժամանակի ընթացքում միատեսակ շեղումներով: Այդ շեղումների ազդեցությունները ԻՍ-ի վրա կարելի է կանխատեսել և հաշվի առնել սխեմայի մոդելավորման ժամանակ [81]՝ դրանք բնութագրող նմանակման մոդելների միջոցով [82]: ԱԳՇ-երի ազդեցությունները հնարավոր է նվազեցնել արտադրական գործընթացը որոշակի փոփոխությունների ենթարկելով [81]:

ԱԳ պատահական շեղումները սահմանվում են [83] որպես ԻՍ-երի արտադրողականության վրա ազդող և արտադրության մեջ կիրառվող սարքավորումների գործառույթներում առկա վիճակագրական տատանումներ: Դրանք անկանխատեսելի են, սակայն վիճակագրական բնույթ ունենալու շնորհիվ հնարավոր է հաշվարկել դրանց առավելագույն և նվազագույն ազդեցությունները (նկ. 2.1) [83]:



Նկ. 2.1. ԱԳՇ-երից կախված՝ շեմային լարման տատանումները

Պատահական ԱԳՇ-երի վատագույն ազդեցությունները հաշվի առնող սխեմաները նախագծվում և մոդելավորվում են վերոհիշյալ վիճակագրական տվյալներին համապատասխանող պայմանների համար [83]: Անհրաժեշտ է նաև հաշվի առնել, որ ԱԳՇ-երի նկատմամբ կայունության ապահովումը նվազեցնում է ԻՍ-ի արագագործությունը [84]:

Իրական սխեմաներում առաջացող աղմուկների, գծերի դիմադրությունների տատանումների պատճառով դրանց սնուցման լարումները անընդհատ ենթարկվում են որոշակի շեղումների [85]: Ըստ նախագծման սահմանված չափանիշների [85]՝ ԻՍ-երի նախագծումը պետք է կատարվի՝ հաշվի առնելով լարման տատանումների մինչև 12% շեղումը: Սնուցման լարումը ուղղակիորեն ազդում է ԻՍ-ի երկու ամենակարևոր մակրոբնութագրերի՝ արագագործության և էներգասպառման վրա [86]: Ուստի, դրա կամայական շեղումները [87] կարող են այդ բնութագրերի վրա ունենալ վճռորոշ ազդեցություն:

Ջերմաստիճանային տատանումները [88] համակարգի արտադրողականության [89] վրա ազդող մյուս գործոններն են, որոնց ազդեցությունները շոշափելի են 100 նմ-ից փոքր հոսքուղու երկարությամբ տրանզիստորների վրա [90-92]:

Թվային սխեմաներում ԳԼՋ շեղումներն առաջացնում են, մասնավորապես, հաճախականությունների և էներգասպառման շեղումներ [93-95], որոնք կարող են ազդել սխեմայի աշխատունակության վրա՝ հանգեցնելով նախագծման՝ տեխնիկական առաջադրանքով սահմանված պահանջների խախտմանը:

Խնդիրը լուծելու համար ԻՍ-երը նախագծում են՝ հաշվի առնելով աշխատանքի ընթացքում որոշակի շեղումների առաջացումը [96]: Լարման և ջերմաստիճանի տատանումներն ազդում են տրանզիստորների արագագործության վրա [97]: Տեխնոլոգիական առաջընթացին զուգընթաց՝ դրանց ազդեցությունը կիսահաղորդչային սարքերի էլեկտրական բնութագրերի վրա դառնում են ավելի շոշափելի [98,99]: Քանի որ ԳԼՋ շեղումների ազդեցությունը էական է հատկապես ԻՍ-ի տրանզիստորների ժամանակային բնութագրերի վրա [100], վերջիններս ստուգվում են ԳԼՋ տարբեր կոմբինացիաների առկայության դեպքում: ԳԼՋ շեղումների

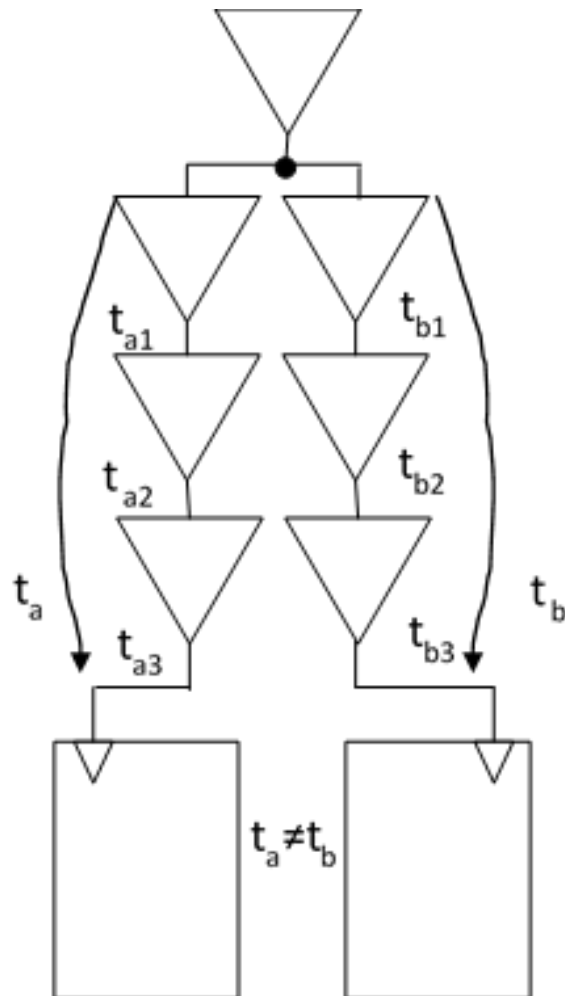
ազդեցությունը սինքրոն սխեմաներում զգալի է սինքրոազդանշանի բաշխման ցանցում [101-106]. տեղի է ունենում տակտավորող ազդանշանի իմպուլսի տևողության փոփոխություն և/կամ տարրերի մուտքերին վերջինիս հապաղման շեղումներ: Իմպուլսի տևողության փոփոխությունները [101] պայմանավորված են այն հանգամանքով, որ ազդանշանի դրական և բացասական ճակատները ձևավորվում են համապատասխանաբար ԲՄՕԿ և ՈՄՕԿ տրանզիստորների միջոցով, որոնց վրա ԳԼՋ շեղումների ազդեցությունը համաչափ չէ: Այդ պատճառով, մինչև տրիգերների մուտքերին հասնելը, հակադիր ճակատները ձևավորվում են տարբեր հապաղումներով և հանգեցնում են սինքրոազդանշանի իմպուլսի տևողության փոփոխականության [101]: Ժամանակակից ԻՍ-երում օգտագործվող տրիգերները հիմնականում երկտակտ են [107], և կասկադները տակտավորվում են սինքրոազդանշանի հակադիր ճակատներով: Վերոհիշյալից հետևում է, որ տրիգերի բաղկացուցիչները կարող են տակտավորվել իմպուլսների փոփոխական տևողությամբ ազդանշանով [102], ինչը հնարավոր է՝ հանգեցնի վերջինիս ֆունկցիոնալ խափանման:

ԳԼՋ շեղումների պատճառով դեպի տրիգերների մուտք տարբեր ուղիներով սինքրոազդանշանի տարածումը ևս կարող է ունենալ տարբեր մեծության հապաղումներ [108-111]: Դա պայմանավորված է ուղիներում տարբեր տարրերի առկայությամբ, որոնց վրա միջավայրի շեղումներն ունենում են անհավասար ազդեցություններ (նկ. 2.2) [112]:

Սինքրոազդանշանի հապաղումների արժեքները թույլատրելի սահմաններում պահելու համար կատարում են նախագծի լավարկում [109]. նվազեցնում կամ ավելացնում են ուղում տրամաբանական փականների քանակը, մեծացնում են իմպուլսի տևողությունը [110] և այլն:

Սինքրոազդանշանի՝ ԳԼՋ շեղումներով պայմանավորված հապաղումների փոփոխությունները հանգեցնում են ազդանշանի փուլային շեղումների, ինչը դրսևորվում է ղեկավարող ճակատի թրթռոցներով [111-115]: Դրանց հետևանքն է, որ տրիգերն ընդունում է դեռ չկայունացած ազդանշանը [111]: Այդ դեպքում առաջանում է մետակայունության հավանականություն, ինչը կարող է հանգեցնել ընդհանուր համակարգի խափանման:

Մետակայունության առաջացման խնդիրը լուծելու համար 1.2-ում նկարագրված եղանակներով իրականացվում է հաղորդակցվող սինքրոտիզմի համաձայնեցում: Ակնհայտ է, որ համակարգի կայունությունն ապահովելու նպատակով համաձայնեցման մեթոդներն այս կամ այն կերպ պետք է հաշվի առնեն ԳԼՋ շեղումների առկայությունը և ձգտեն հնարավորինս մեղմել դրանց բացասական ազդեցությունները ժամանակային բնութագրերի վրա: Համաձայնեցման գոյություն ունեցող միջոցները, սակայն, իրենց իրականացման եղանակի առանձնահատկություններից ելնելով, ունեն որոշակի զգայունություն ԳԼՋ շեղումների նկատմամբ և էապես նվազեցնում են ԻՍ-երի արագագործությունը, մեծացնում ընդհանուր մակերեսը, էներգասպառումն ու կիրառելի են միայն խիստ սահմանափակ հաճախականությունների համաձայնեցման համար:

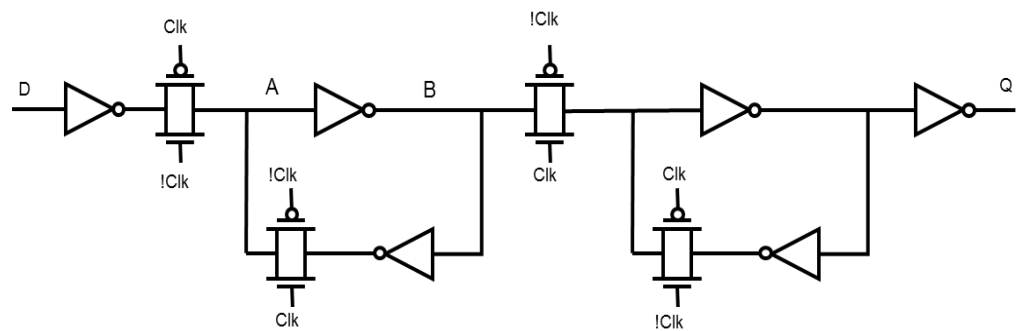


Նկ. 2.2. Տարբեր ուղիներով սինքրոազդանշանի տարածման հապաղումները ԳԱՏՍ ԻՍ-երում սինքրոտիզմի համաձայնեցման և նշված խնդիրների լուծման համար առաջարկված է ԳԼՋ շեղումների նկատմամբ անտարբեր,

մետակայունության առաջացումը բացառող, բարձր արագագործություն և փոքր մակերես ապահովող երկու մեթոդ: Դրանց հիման վրա մշակվել են համապատասխան համաձայնեցնող համակարգերը:

2.2. Մետակայունության՝ համակարգի բնութագրերի վրա ունեցած ազդեցությունների գնահատումը

Սինթրոտիրոյթների համաձայնայնեցման արդյունավետ մեթոդներ մշակելու նպատակով իրականացվել է [116] մետակայունության ազդեցությունների ուսումնասիրություն և սխեմայի բնութագրերի շեղումների գնահատում: Հետազոտություններն իրականացվել են երկտակտ տրիգերի օրինակով (նկ. 2.3) [107],

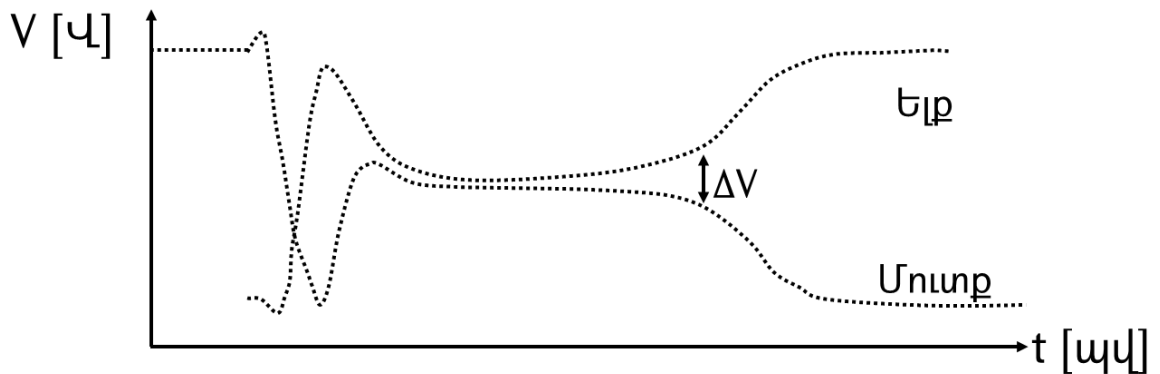


Նկ. 2.3. Երկտակտ տրիգերի կառուցվածքը

Տրիգերի ժամանակային սահմանափակումների՝ հաստատման և/կամ պահպանման ժամանակների խախտման դեպքում առաջին կասկադի ելքում A և B ազդանշանները հայտնվում են մետակայուն վիճակում, որի արդյունքում շրջիչի մուտքն ու ելքը դառնում են իրար գրեթե հավասար՝ խախտելով դրա տրամաբանական ֆունկցիան (նկ. 2.4): Գործնականում, կասկադներից յուրաքանչյուրի ելքը կարող է հայտնվել մետակայուն վիճակում, սակայն կարևոր է բացառել երկրորդ կասկադի մետակայունությունը, քանի որ սարքի ելքը ձևավորվում է հենց դրանից: B մետակայուն հանգույցի՝ տակտային ազդանշանի բացասական ճակատին մոտ վերականգնվելու դեպքում, երկրորդ կասկադի ելքը ևս հայտնվում է մետակայուն վիճակում:

Սինթրոազդանշանի բացասական ճակատից բավարար ժամանակ առաջ վերականգնվելու դեպքում երկրորդ կասկադը ընդունում է կայուն տրամաբանական երկու արժեքներից մեկը [107]: Առաջին կասկադի ելքին մետակայունության առաջացումը, տրիգերի ժամանակային սահմանափակումների խախտման չափից կախված, կարող է հանգեցնել հետևյալ ելքային արդյունքներից մեկին [69].

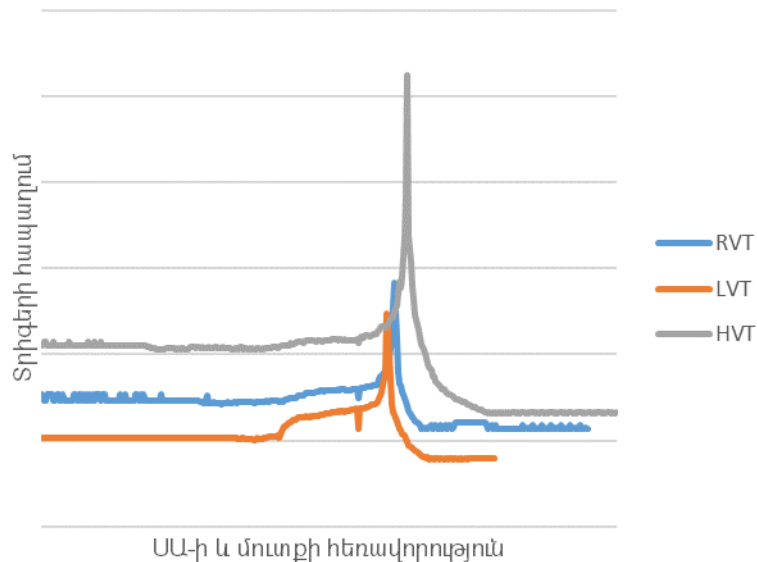
- Ազդանշանի՝ անվանական հապաղումից մեծ հապաղում: Այս դեպքը տեղի է ունենում, երբ A հանգույցում ազդանշանը վերականգնվում է տակտավորման պարբերությունից փոքր ժամանակում:
- Ազդանշանի կորուստ, երբ ելքային ազդանշանն ընդհանրապես չի փոխանջատվում:



Նկ. 2.4. Երկտակտի տրիգերում մետակայունության առաջացումը

Ուսումնասիրվել են [116] տրիգերի մետակայունության բնութագրերը, դրա ժամանակային սահմանափակումների՝ տարբեր չափերի խախտումների դեպքում: Նմանակումներն իրականացվել են շեմային լարման երեք՝ բարձր, ցածր և նոմինալ արժեքների դեպքում, հետևյալ երեք պայմաններում՝ TT, FF, SS:

Մոդելավորման արդյունքները (նկ. 2.5) ցույց են տվել, որ մետակայուն վիճակում, տրիգերի մուտքի՝ դրա ժամանակային սահմանափակումները խախտելու չափից կախված, սարքի հապաղումն աճում է էքսպոնենցիալ օրենքով՝ հասնելով նոմինալ հապաղման կրկնակի արժեքին (աղ. 2.1):



Նկ. 2.5. Տրիգերի հապաղման՝ դրա ժամանակային սահմանափակումների խախտման
չափից կախվածությունը

Աղյուսակ 2.1

Տարբեր շեմային լարումների և ջերմաստիճանների դեպքում բնականոն և մեղակայուն
վիճակներում տրիգերի հապաղման արժեքները (պվ)

	Անվանական $V_{2\text{եմ}}$		Ցածր $V_{2\text{եմ}}$		Բարձր $V_{2\text{եմ}}$	
	Բնականոն	Մետ.	Բնականոն	Մետ.	Բնականոն	Մետ.
TT	76,8	142	51,5	124	105	263
FF	46,7	100	20,8	55	45	100
SS	107	221	84,5	207	164	350

Հապաղման մեծ արժեքները հանգեցնում են կայունացման ժամանակի փոքր արժեքների (աղ. 2.2), ինչը, ըստ (1.11)-ի, նվազեցնում է համակարգի խափանման պարբերությունը և, հետևաբար, սարքի հուսալիությունը:

Աղյուսակ 2.2

Տարբեր շեմային լարումների և ջերմաստիճանների դեպքում տրիգերի մեղակայուն
ելքի կայունացման ժամանակի արժեքները (պվ)

	Անվանական $V_{2\text{եմ}}$		Ցածր $V_{2\text{եմ}}$		Բարձր $V_{2\text{եմ}}$	
	Բնականոն	Մետ.	Բնականոն	Մետ.	Բնականոն	Մետ.
TT	873,2	808	898,5	826	845	687

FF	903,3	850	929,2	895	905	850
SS	843	729	865,5	743	786	600

Մետակայունության ազդեցության ուսումնասիրությունը ցույց է տալիս, որ մետակայուն ազդանշանի վերականգնման դեպքում երկտակտ տրիգերի հապաղումը կարող է մեծանալ մինչև դրա անվանական արժեքի կրկնակի արժեքը, ինչը անթույլատրելի է ԻՍ-ի արագագործության ապահովման տեսանկյունից:

Համանման արժեքներ են ստացվել բնականոն և մետակայուն վիճակների դեպքում մակարդակային, ճակատ գրանցող և իմպուլս գեներացնող տրիգերային համաձայնեցուցիչների հապաղումների համար (աղ. 2.3): Վերջիններս մոդելավորվել են 1 ԳՀց սինքրոնազդանշանով տակտավորվելու, անվանական շեմային լարման և ջերմաստիճանի պայմաններում:

Աղյուսակ 2.3

Բնականոն և մետակայուն վիճակների առաջացման դեպքում տրիգերային համաձայնեցուցիչների հապաղումները

Համաձայնեցուցիչ	Բնականոն	Մետակայուն
Մակարդակային	1,119	1,985
Ճակատ գրանցող	1,142	2,01
Իմպուլս գեներացնող	1,75	2,567

2.2. Մետակայունության հայտնաբերմամբ և բնականոն վիճակի վերականգնմամբ համաձայնեցման մեթոդը

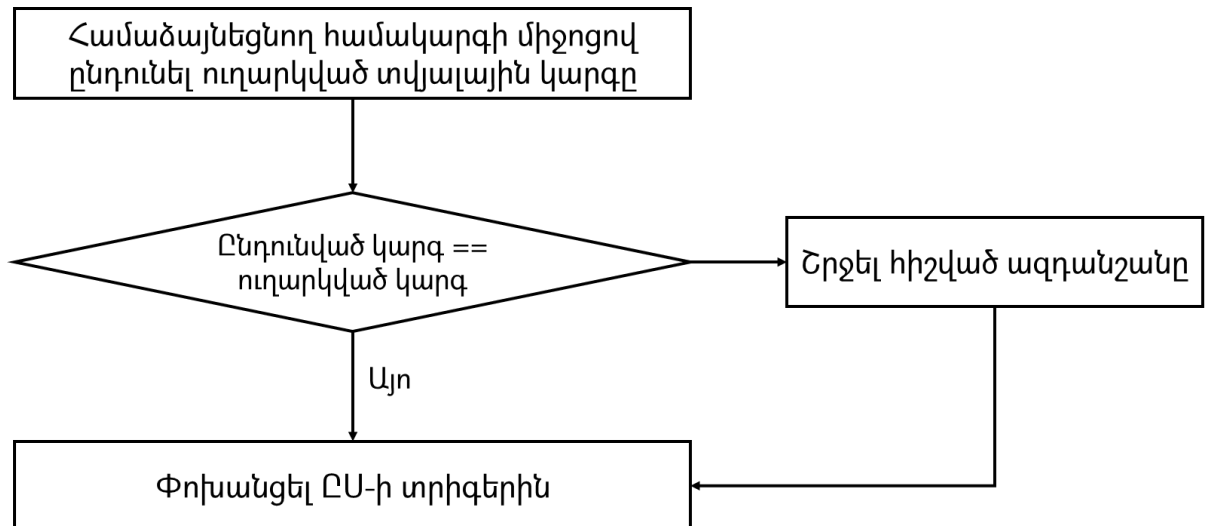
Առաջարկված առաջին մեթոդը [75] հիմնված է մետակայունություն երևույթի հայտնաբերման և ազդանշանի բնականոն վիճակի վերականգնման վրա: Գործնական կիրառություններում մետակայունության առաջացման դեպքում ազդանշանը կարող է որոշ անհայտ ժամանակից հետո անցում կատարել ակնկալվող կայուն վիճակի կամ վերադառնալ նախորդին՝ հանգեցնելով ազդանշանի տվյալային կարգի կորստի: Այսինքն, եթե ՈՒՍ-ի ելքային ազդանշանը փոխանջատվել է, իսկ ԸՍ ՍԱ-ի ղեկավարող ճակատից հետո այդ փոխանջատված արժեքը չի հաստատվել ընդունող տրիգերի ելքում, ապա վերջինս հայտնվել է մետակայուն վիճակում: Ակնհայտ է, որ ազդանշանի

նոր փոխանջատված արժեքը պետք է լինի նախորդի շրջված տարբերակը: Բոլոր այդ հանգամանքները հաշվի առնելով և ԸՍ-ում մետակայունության պատճառով ֆունկցիոնալ խափանումը բացառելու նպատակով հնարավոր է ՈԻՍ-ի ուղարկած ազդանշանը ընդունել այդ երկու սինքրոտիզացիայի միջև դրված համաձայնեցման համակարգի միջոցով, որը կստուգի ստացված արժեքի ստույգությունը և, մետակայուն վիճակի առաջացման դեպքում, ԸՍ-ի մուտքին կփոխանցի վերաթողարկված ճիշտ տվյալային կարգը:

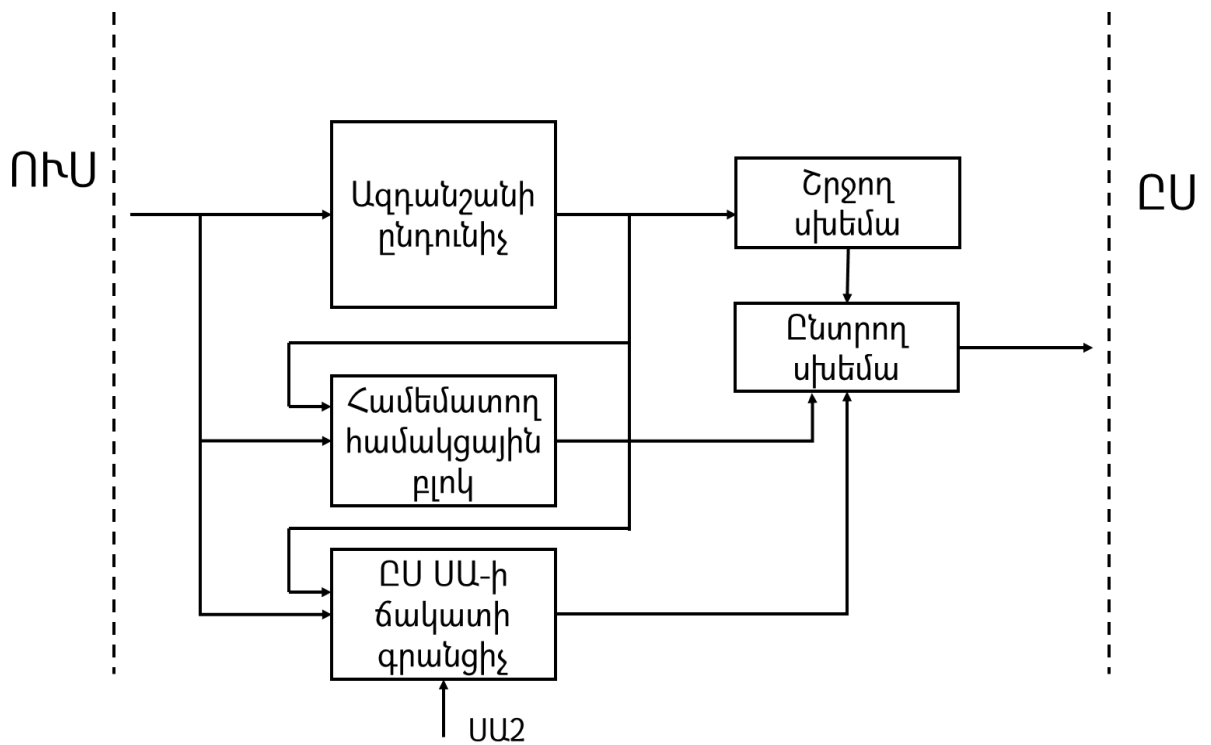
Նկարագրված համաձայնեցման մեթոդի (նկ. 2.6) հիման վրա մշակվել է թվային համակարգ (նկ. 2.7), որն ունի հետևյալ կառուցվածքը՝

- ՈԻՍ-ի ուղարկած ազդանշանն ընդունող համակարգ,
- ՈԻՍ-ի և ԸՍ-ի ելքը համեմատող համակցային բլոկ,
- ԸՍ-ի սինքրոազդանշանի ղեկավարող ճակատի գրանցիչ,
- ազդանշանի շրջիչ,
- ընդունված կամ վերաթողարկված ազդանշանի ընտրիչ:

ՈԻՍ-ի ելքը ընդունող համակարգը տակտավորվում է ընդունող սինքրոտիզացիայի սինքրոազդանշանով: Այն հիշում է ազդանշանի ստացված արժեքը, ինչի հիման վրա տեղի է ունենում մետակայուն վիճակի հայտնաբերումը:



Նկ. 2.6. Մետակայունության հայտնաբերման մեթոդի բլոկ-դիագրամը



Նկ. 2.7. Մետակայունության հայտնաբերմամբ համաձայնեցման մեթոդն
իրականացնող համակարգի կառուցվածքը

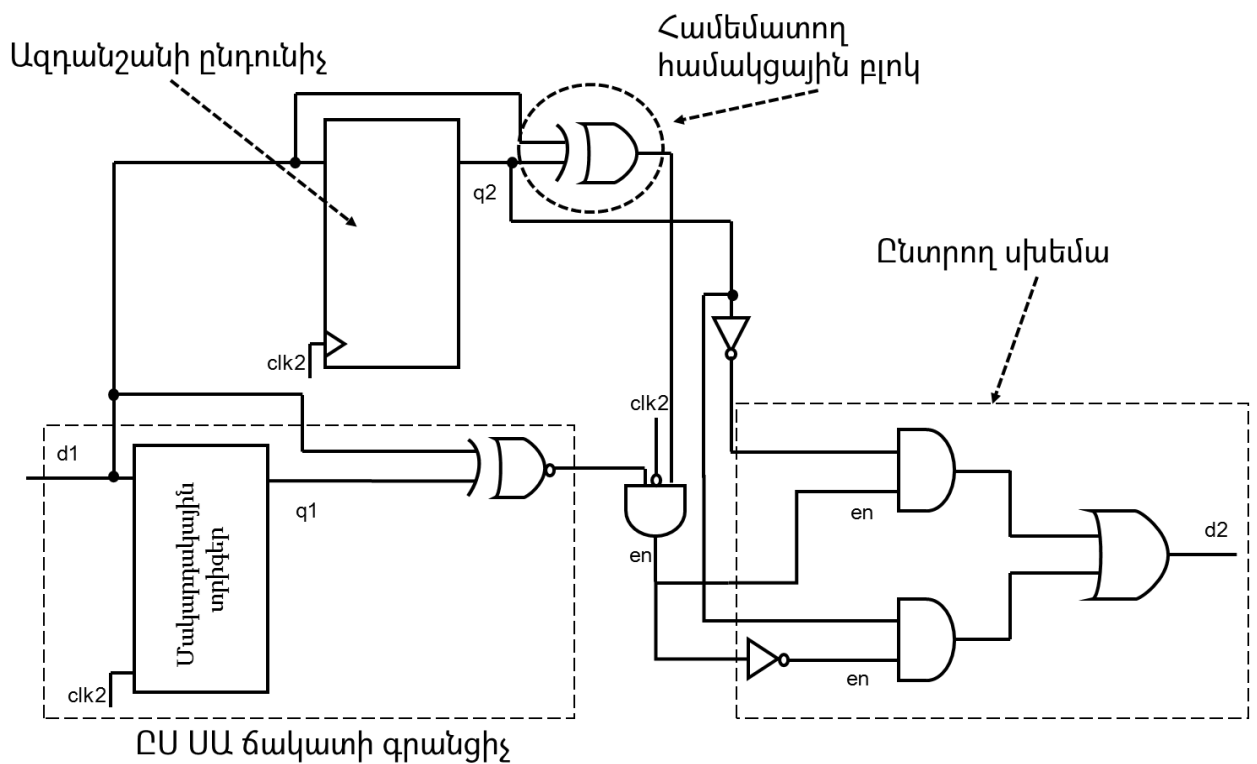
Նկ. 2.7-ում պատկերված համակարգի աշխատանքի սկզբունքը հետևյալն է.

1. Ազդանշանի ընդունիչը (ԱԸ) ընդունում է ՌԻՍ-ի ուղարկած ազդանշանը՝ մեկուսացնելով հավանական մետակայուն վիճակը ԸՍ-ի մուտքից և բացառելով դրա առաջխաղացումը:
2. Համեմատող համակցային բլոկը (ՀաԲ) գրանցում է ՌԻՍ-ի և ԱԸ-ի ելքերի միջև տարբերության առկայությունը, որը կարող է լինել նշված երկու դեպքերից մեկում.
 - ա) ՌԻՍ-ի ելքում ազդանշանը փոխանջատվել է, սակայն ընդունող սինքրոազդանշանի ճակատը դեռ չի գրանցվել: Այս դեպքում ԸՍ-ի սինքրոազդանշանի ճակատը ըստ ժամանակի բավարար հեռացված է ընդունվող ազդանշանի ճակատից՝ բացառելով ԱԸ-ի ելքում մետակայունության առաջացումը:
 - բ) ՌԻՍ-ի ելքում ազդանշանը փոխանջատվել է, ընդունող սինքրոազդանշանի ճակատը արդեն տեղի է ունեցել, սակայն ԱԸ-ն չի գրանցել նոր՝ փոփոխված արժեքը: Սա կարող է տեղի ունենալ միայն այն դեպքում, երբ ԱԸ-ի ընդունող

տրիգերի ելքը հայտնվել է մետակայուն վիճակում: ՀաՔ-ի ելքը հանդիսանում է ընտրող սխեմայի մուտքերից մեկը:

3. ԸՍ ՍԱ-ի ճակատի գրանցիչը անհրաժեշտ է վերոհիշյալ երկու դեպքերից մետակայուն վիճակի առաջացումը տարանջատելու համար: Դրա ելքը ևս տրվում է ընտրող սխեմայի մուտքին և ՀաՔ-ի ելքի հետ միասին արձանագրում է մետակայուն վիճակի առաջացման փաստը
4. Շրջող սխեման համակցային սխեմա է, որի ելքում գեներացվում է ԱԸ-ի ելքային արժեքի շրջված տարբերակը: Մետակայուն ազդանշանի հայտնաբերման դեպքում ԸՍ-ի մուտքին է փոխանցվում այս սխեմայի ելքը
5. Ընտրող սխեման, կախված նրանից՝ ԱԸ-ի ելքը մետակայուն վիճակում է թե ոչ, ղեկավարող ազդանշանների միջոցով ԸՍ-ի մուտքին է փոխանցում համապատասխանաբար շրջող սխեմայի կամ ԱԸ-ի ելքային կարգերը

Նկարագրված համակարգի համար մշակվել է փականային մակարդակի նկարագրություն (նկ. 2.8): Նկ. 2.8-ում ԱԸ-ն ներկայացնում է ԸՍ-ի սինքրոնազդանշանի ճակատով տակտավորվող երկտակտ տրիգեր (նկ. 2.3), որն ԸՍ-ի համար իրականացնում է հավանական մետակայունության «էկրանավորում»: Ճակատով ղեկավարվող տրիգերների երկտակտ բնույթի շնորհիվ հավանական մետակայուն վիճակի առաջխաղացումը դեպի տրիգերի Q ելք կանխվում է դրա առաջին կասկադի միջոցով:

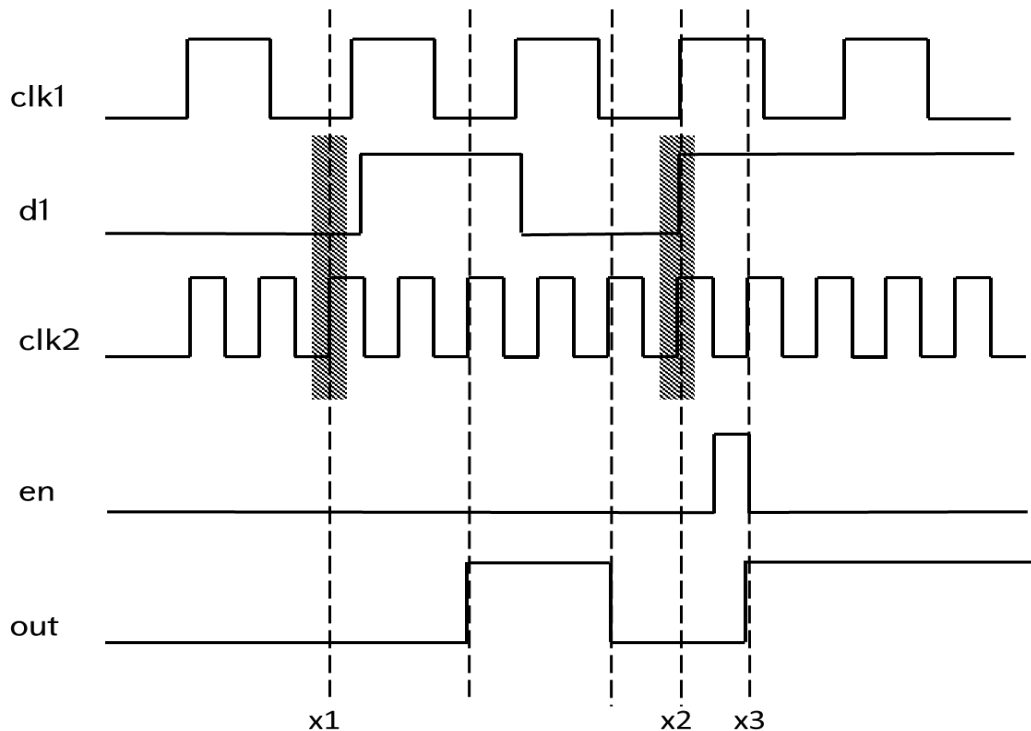


Նկ. 2.8. Մետակայունության հայտնաբերմամբ համաձայնեցման միջոցի փականային մակարդակի սխեման

Համեմատող սխեման իրականացված է ԲԱՑԱՌՈՂ-ԿԱՄ փականի միջոցով, որի ելքն ընդունում է տրամաբանական «1» արժեք, եթե հայտնաբերվել է ԱԸ տրիգերի մուտքի և ելքի անհամապատասխանություն: Շրջող սխեման իրականացված է շրջիչի միջոցով: ԸՍ ՍԱ-ի ճակատի գրանցումն արվում է անուղղակիորեն. ԸՍ սինքրոազդանշանի տրամաբանական «բարձր» մակարդակով ղեկավարվող տրիգերն ընդունում է ՈԻՍ-ից ուղարկված տվյալային կարգը, իսկ ԲԱՑԱՌՈՂ-ԿԱՄ-ՈՉ փականը համեմատում է այդ տրիգերի մուտքային և ելքային ազդանշանների հավասարությունը: Վերջինս տեղի է ունենում, երբ հաստատվում է ՍԱ-ի տրամաբանական «1» մակարդակը, այսինքն տեղի է ունենում դրական ճակատի իրադարձություն: ԲԱՑԱՌՈՂ-ԿԱՄ և ԲԱՑԱՌՈՂ-ԿԱՄ-ՈՉ փականների ելքերի՝ միաժամանակ «բարձր» արժեք ընդունելու դեպքը արձանագրվում է ԵՎ փականի միջոցով և կարող է կայանալ միայն այն դեպքում, երբ տեղի է ունեցել փոխանջատված ազդանշանի կորուստ, որը, միարժեքորեն, մետակայունության հետևանք է: ԵՎ փականի ելքը ընտրող սխեմայի՝ 2:1 մուլտիպլեքսորի ղեկավարող մուտքն է: Այդ

մուլտիպլեքսորի ելքը միացված է ԸՍ-ի ընդունող տրիգերի մուտքին, իսկ տվյալային մուտքերը ԱԸ տրիգերի և շրջիչի ելքերն են:

Նկարագրված համաձայնեցման միջոցի աշխատանքի ժամանակային դիագրամն է՝

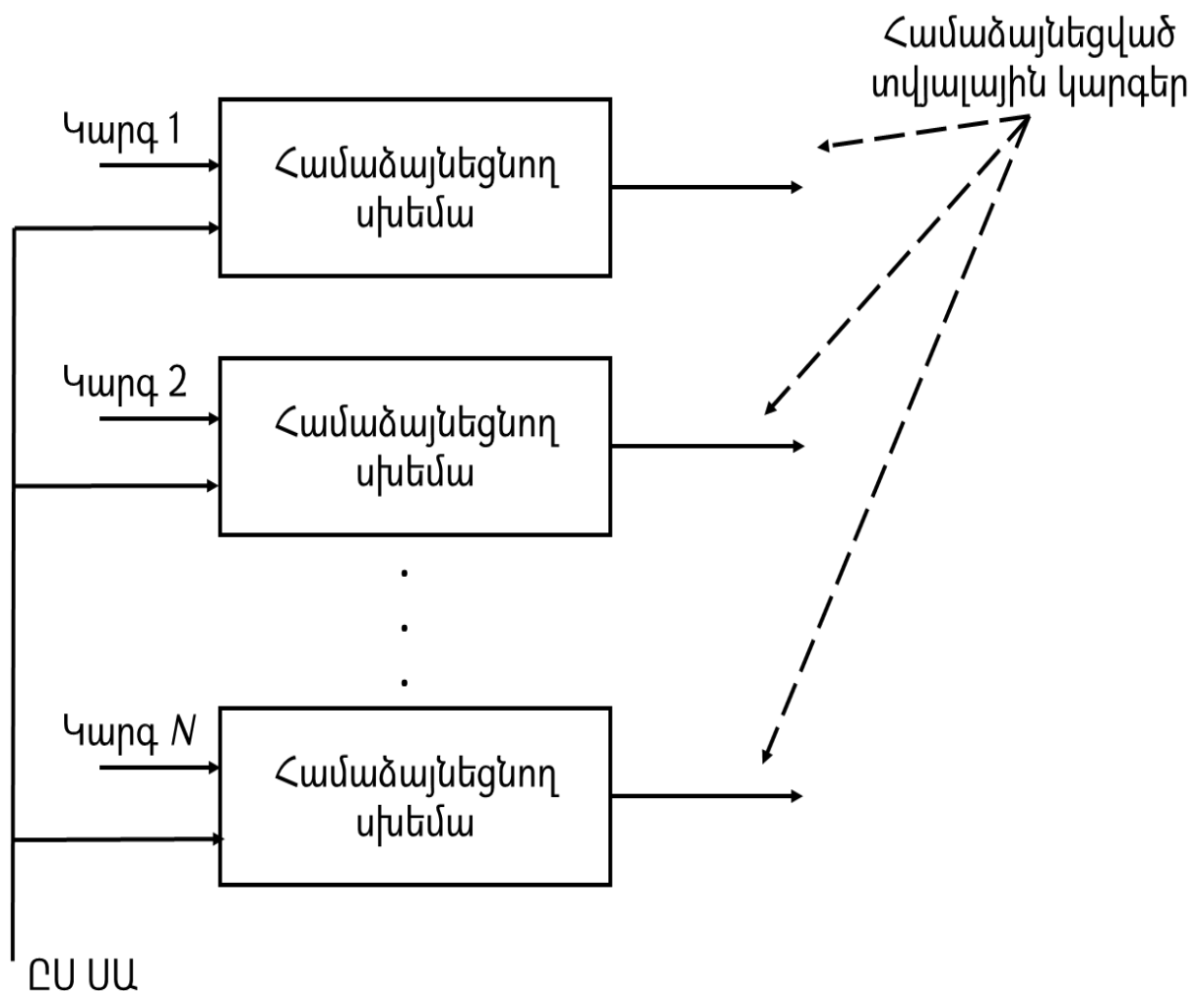


Նկ. 2.9. Առաջարկված համաձայնեցման միջոցի աշխատանքի ժամանակային դիագրամը

Նկ. 2.9-ում պատկերված ազդանշաններից clk1-ը ՌԽՍ-ի սինքրոազդանշանն է, իսկ clk2-ը՝ ԸՍ-ինը: d1-ով է նշանակված ՌԽՍ-ից դեպի ԸՍ ուղարկված ազդանշանը, en-ով՝ ընտրող սխեմայի ղեկավարող մուտքը, իսկ out-ը համաձայնեցման միջոցի ելքն է, որը միացված է ԸՍ-ի ընդունող տրիգերի մուտքին: Պատկերված դիագրամում ընդգծված հատվածները ներկայացնում են ԸՍ ՍԱ-ի ճակատի շուրջն արգելված տիրույթը: Մուտքային ազդանշանի փոխանջատումները x1 պահին բավարարում են ԱԸ տրիգերի ժամանակային պահանջները, և մեկ պարբերություն անց համաձայնեցման միջոցի ելքը փոխանջատվում է: Ժամանակի x2 պահին տեղի ունեցած փոխանջատումը կատարվել է clk2-ի արգելված տիրույթում: Այդ դեպքում ՀաԲ-ը արձանագրում է տվյալային կարգի կորուստ և clk2-ի բացասական ճակատից հետո մակարդակային տրիգերի հետ ակտիվացնում է en ազդանշանը, և x3 պահին՝ մետակայունության առաջացումից մեկ պարբերություն անց, տեղի է ունենում համաձայնեցման միջոցի

ելքի փոխանջատումը: Ակնհայտ է, որ բնականոն և մետակայուն ազդանշանների դեպքում համաձայնեցման սխեմայով տարածման հապաղումը մոտավորապես հավասար է ԸՍ ՍԱ-ի մեկ պարբերությանը: Դրա շնորհիվ համակարգը դառնում է ավելի հուսալի և կանխատեսելի:

Մետակայունության հայտնաբերմամբ և բնականոն վիճակի վերականգնմամբ համաձայնեցման մեթոդը սինքրոնացնում է մեկ տվյալային կարգ: Ժամանակակից ԻՍ-երում սինքրոտիզացիայի միջև հիմնականում փոխանցվում են բազմակարգ ազդանշաններ: Նկարագրված մեթոդը կիրառելի է նաև վերջիններիս համաձայնեցման դեպքում՝ յուրաքանչյուր կարգի համար օգտագործելով մեկ համաձայնեցման սխեմա (նկ.2.10):



Նկ. 2.10. N-կարգանի ազդանշանի համաձայնեցումը

1.1-ում նշվել էր, որ բազմակարգ ազդանշանի կարգերն առանձին համաձայնեցնելու դեպքում կարգերից մեկի՝ մետակայուն վիճակում հայտնվելու

պատճառով այն իր ընդունող տրիգերին կհասնի ավելի ուշ, քան մյուս կարգերը՝ դրանով իսկ հանգեցնելով ազդանշանի աղավաղման: Նկ. 2.10-ում իրականացված համաձայնեցման դեպքում, անկախ նրանից, թե ազդանշանները մետակայուն վիճակում հայտնվել են թե ոչ, և անկախ այդ վիճակում հայտնված տվյալային կարգերի քանակից, դրանք բոլորը ԸՍ-ի մուտքին հասնելու են միաժամանակ՝ ԸՍ ՍԱ-ի մեկ պարբերության ընթացքում:

Առաջարկվող համաձայնեցման միջոցն առկա մյուս միջոցների համեմատ ունի էական առավելություններ: Տրիգերային համաձայնեցուցիչների նկատմամբ ունեցած առավելություններն են՝

- Ավելի բարձր արագագործություն, որով պայմանավորված է փոքր հապաղումը
- ԳԼՋ շեղումների նկատմամբ անտարբերություն: Մեթոդը հիմնված չէ մետակայուն ազդանշանի՝ առավելագույնը մեկ պարբերության ընթացքում վերականգնվելու ենթադրության վրա, որը ԳԼՋ որոշակի շեղումների դեպքում կարող է տեղի չունենալ:
- Բազմակարգ ազդանշանների համաձայնեցումը կատարվում է՝ բացառելով կարգերի ուշացման պատճառով ազդանշանի հավանական աղավաղումը:

Թերություններն են՝

- տարրերի համեմատաբար մեծ քանակի պատճառով զբաղեցրած մեծ մակերեսը
- միայն բարձր հաճախականությունից ցածրին ազդանշանի փոխանցման համաձայնեցումը:

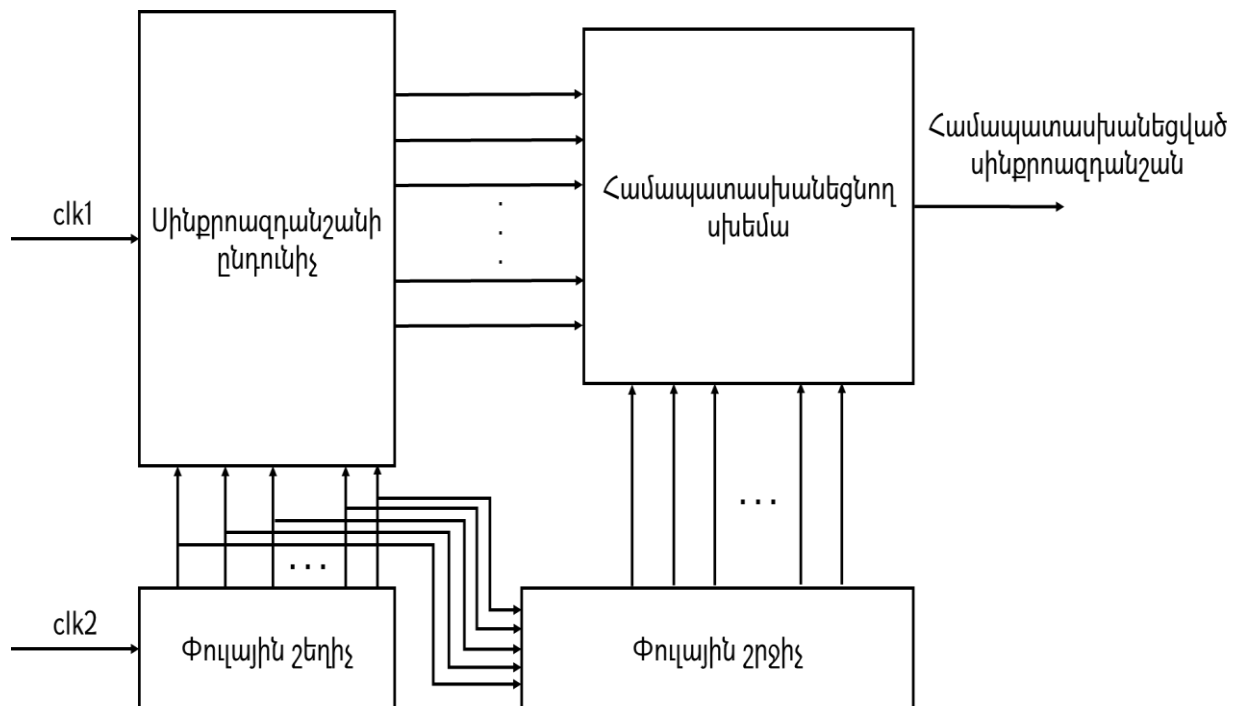
Գլխավորապես բազմակարգ ազդանշանների համաձայնեցման համար կիրառվող ինտերֆեյսների համեմատ այս միջոցը ունի հետևյալ առավելությունները՝

- բարձր արագագործություն և արտադրողականություն,
- զբաղեցրած փոքր մակերես,
- իրականացման պարզություն, ինչը պայմանավորված է նրանով, որ այն չի իրականացնում համաձայնեցման որևէ արձանագրություն:

Թերությունն այն է, որ առաջարկված միջոցը համաձայնեցնում է միայն բարձր հաճախականությունից ցածրին ազդանշանի փոխանցումը:

2.3. Սինքրոազդանշանի դինամիկ համապատասխանեցմամբ համաձայնեցման մեթոդը

Առաջարկվող համաձայնեցման երկրորդ մեթոդը [75] հիմնված է հաղորդակցվող սինքրոտիմայթների սինքրոազդանշանների դինամիկ համապատասխանեցման վրա: Մեթոդը, ԸՍ-ի սինքրոազդանշանի փուլը տարբեր չափերով շեղելով, վերջիններից ընտրում է ՈԻՍ-ի սինքրոազդանշանին փուլով ամենահամապատասխանը և հենց այդ ազդանշանով տակտավորում է ԸՍ-ի սահմանային տրիգերները: Այս մեթոդի հիման վրա մշակվել է համաձայնեցման համակարգ (նկ. 2.11):



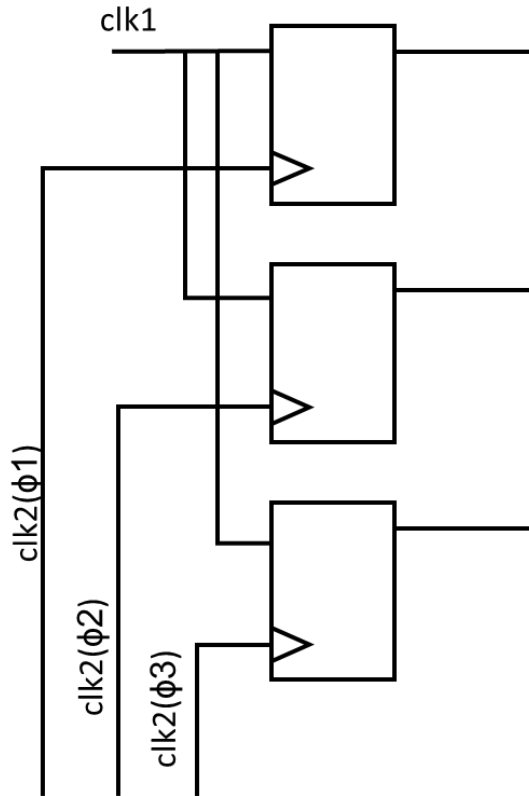
Նկ. 2.11. Սինքրոազդանշանների համապատասխանեցմամբ համաձայնեցման համակարգի սկզբունքային կառուցվածքը

Նկ. 2.11-ում պատկերված համակարգի աշխատանքի սկզբունքը հետևյալն է՝

- Փուլային շեղիչը (ՓՇ) գեներացնում է ԸՍ ՍԱ-ից փուլով շեղված N հատ ազդանշան: Այդ ազդանշանները տակտավորում են սինքրոազդանշանի ընդունիչի N համապատասխան հատվածները: Ազդանշանների փուլերը միմյանցից շեղված են հավասար չափով:
- Սինքրոազդանշանի ընդունիչը (ՍԸ) մուտքին ընդունում է ՈԻՍ ՍԱ-ն: Այստեղ փուլերով շեղված N հատ տակտավորող ազդանշանները կատարում են $clk1$ -ի ընտրումներ և ձևավորում համապատասխան ելքեր:

- ՍԸ-ի ելքերի հիման վրա համապատասխանեցնող սխեման ($\angle U$) հայտնաբերում է $clk2$ -ի այն փուլային շեղումը, որը համափուլ է $clk1$ -ին: Այդ ազդանշանը տակտավորում է ԸՍ-ի սահմանային տրիգերները:

ՍԸ-ն իրականացվել է $clk2$ ազդանշանից որոշակի $\Delta\phi$ փուլով շեղված ազդանշաններով տակտավորվող տրիգերներով (նկ. 2.12):

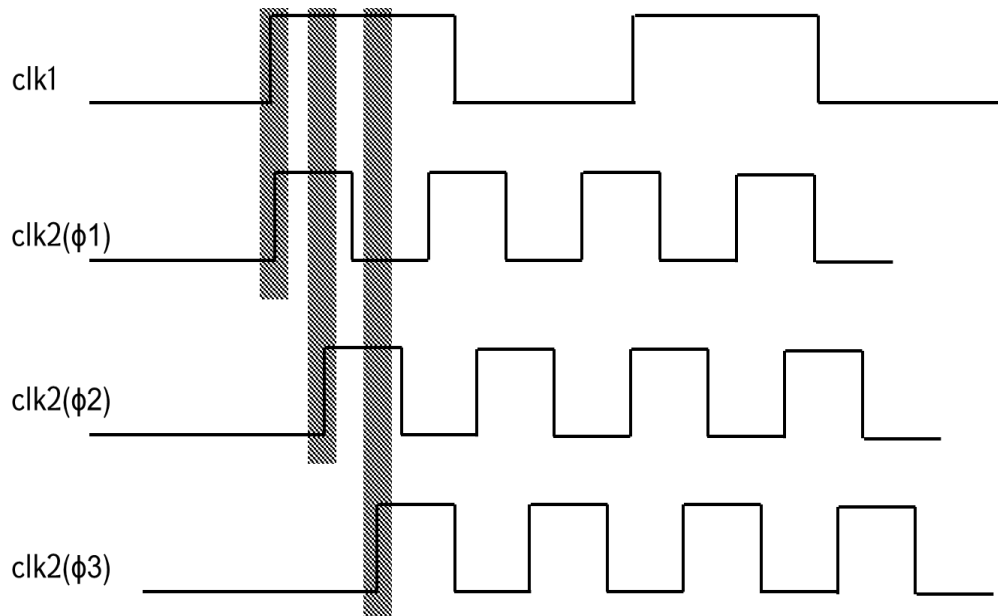


Նկ. 2.12. Սինքրոազդանշանի ընդունիչը

Նկ. 2.12-ում պատկերված տրիգերների մուտքը ուղարկող սինքրոտիրույթի $clk1$ սինքրոազդանշանն է: Քանի որ $clk1$ -ը և $clk2$ -ը միմյանց նկատմամբ ասինքրոն են, հնարավոր է $clk2$ -ի այնպիսի փուլ, որի ճակատը համընկնում է $clk1$ -ի ճակատի հետ: Նմանատիպ փուլի առկայությունը հանգեցնում է համապատասխան տրիգերի ելքի մետակայունության (նկ. 2.13): $clk2$ ազդանշանի փուլային շեղումները ստանալիս անհրաժեշտ է դրանք ընտրել այնպես, որ միայն մեկ ընտրման դեպքում տրիգերի ելքը հայտնվի մետակայուն վիճակում: Եթե $\phi1$ փուլով $clk2$ սինքրոազդանշանի միջոցով $clk1$ -ի ընտրման ժամանակ վերջինիս ճակատն ընկել է $clk2$ -ի արգելված տիրույթի մեջ, ապա մյուս փուլերը պետք է $\phi1$ -ից և միմյանցից շեղված լինեն առնվազն արգելված տիրույթի լայնության չափով: Ապահովության համար բավարարվում է հետևյալ պայմանը՝

$$\Delta t > 1.2t_{\text{արգ}} : \quad (2.1)$$

N հատ փուլային շեղումների դեպքում N-րդ սինքրոազդանշանի ճակատը կարող է համընկնել տվյալային կարգի կիսապարբերության բացասական ճակատի հետ և առաջացնել ևս մի մետակայուն ազդանշան: Առաջարկված մեթոդում նախ դիտարկվում է clk1-ի կիսապարբերության ընթացքում տեղի ունեցող 3 ընտրում:



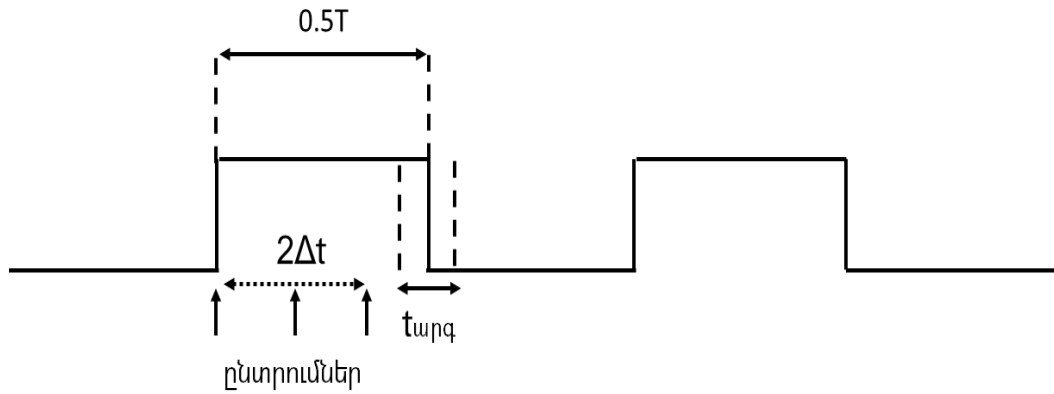
Նկ. 2.13. Տարբեր փուլային շեղումներով clk1-ի ընդունումը

Եթե առաջին ընտրումը համընկնում է ՌԻՍ ՍԱ-ի դրական ճակատի հետ, ապա կա որոշակի հավանականություն, որ $2\Delta t$ շեղմամբ 3-րդ ընտրումը կհամընկնի դրա բացասական ճակատի հետ: Միայն մեկ մետակայուն ազդանշան ապահովելու համար (նկ. 2.14) անհրաժեշտ է ազդանշանի համապատասխան հապաղումն ընտրել այնպես, որ՝

$$2\Delta t < \frac{T}{2} - t_{\text{արգ}} : \quad (2.2)$$

Ակնհայտ է, որ, ընտրումների քանակից կախված, clk1-ի և clk2-ի միջև կարելի է ունենալ ավելի ճշգրիտ փուլային համապատասխանություն: Ստորև կատարվում է 3 և 4 ընտրումների դեպքում համաձայնեցման միջոցի աշխատանքի վերլուծությունը:

Առաջարկվող մեթոդի գործառույթում առանցքային դերը պատկանում է համապատասխանեցնող սխեմային: Այն ներկայացնում է համակցային սխեմա, որը ՍԸ-ի ելքային ազդանշանների միջոցով ընտրում է clk2-ի այն փուլային շեղումը, որի դեպքում clk1-ի ընտրումը մետակայուն չէր:



Նկ. 2.14. clk1 ազդանշանի երեք ընտրումները

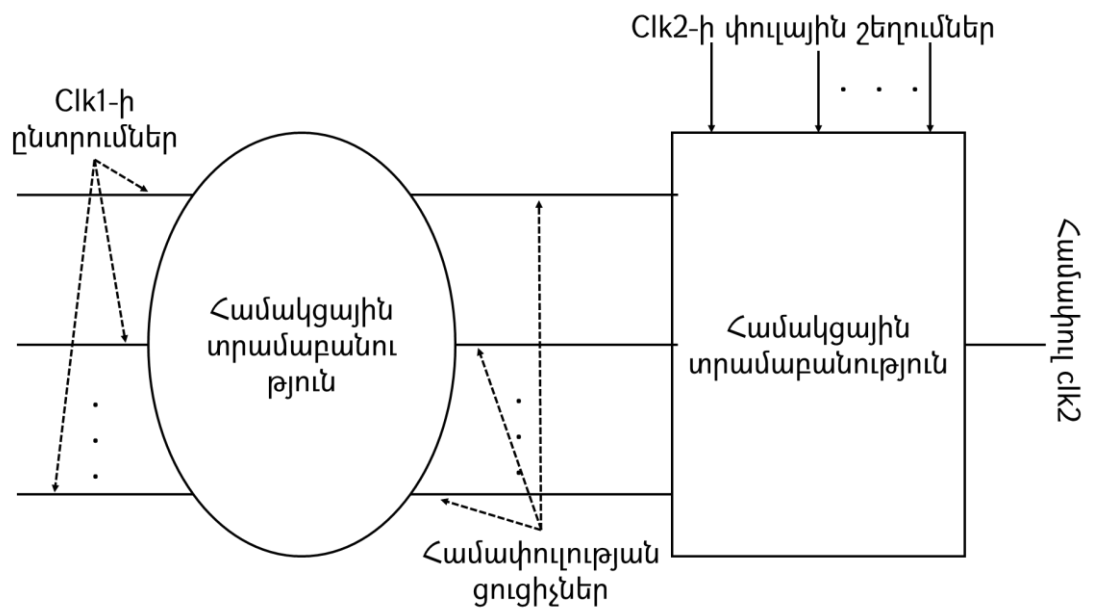
ՀԱ-ի սկզբունքային կառուցվածքը բերված է նկ. 2.15-ում: 3 ընտրում կատարելու դեպքում հնարավոր է ՍԸ-ի ելքային վեկտորի 6 հավաք (նկ. 2.16), որոնցից երեքի դեպքում clk1-ի և clk2-ի փուլային շեղումը փոքր է 180 աստիճանից, իսկ մյուս երեք դեպքում՝ մեծ: Նկ. 2.16-ում «Մ» տառը մատնանշում է տվյալ ընտրման արդյունքում համապատասխան տրիգերի ելքում մետակայունության առաջացումը: Դրանում պատկերված ընտրումների յուրաքանչյուր վեկտորի դեպքում ՀԱ-ն ընտրում է clk2-ի փուլային շեղումներից միայն մեկը: Հարկ է նշել, որ 1, 3, 5 դեպքերը իրար, իսկ 2, 4, 6 դեպքերը իրար փոխբացառող են:

ՍԸ-ի՝ նկ. 2.16-ին համապատասխանող իսկության աղյուսակը հետևյալն է՝

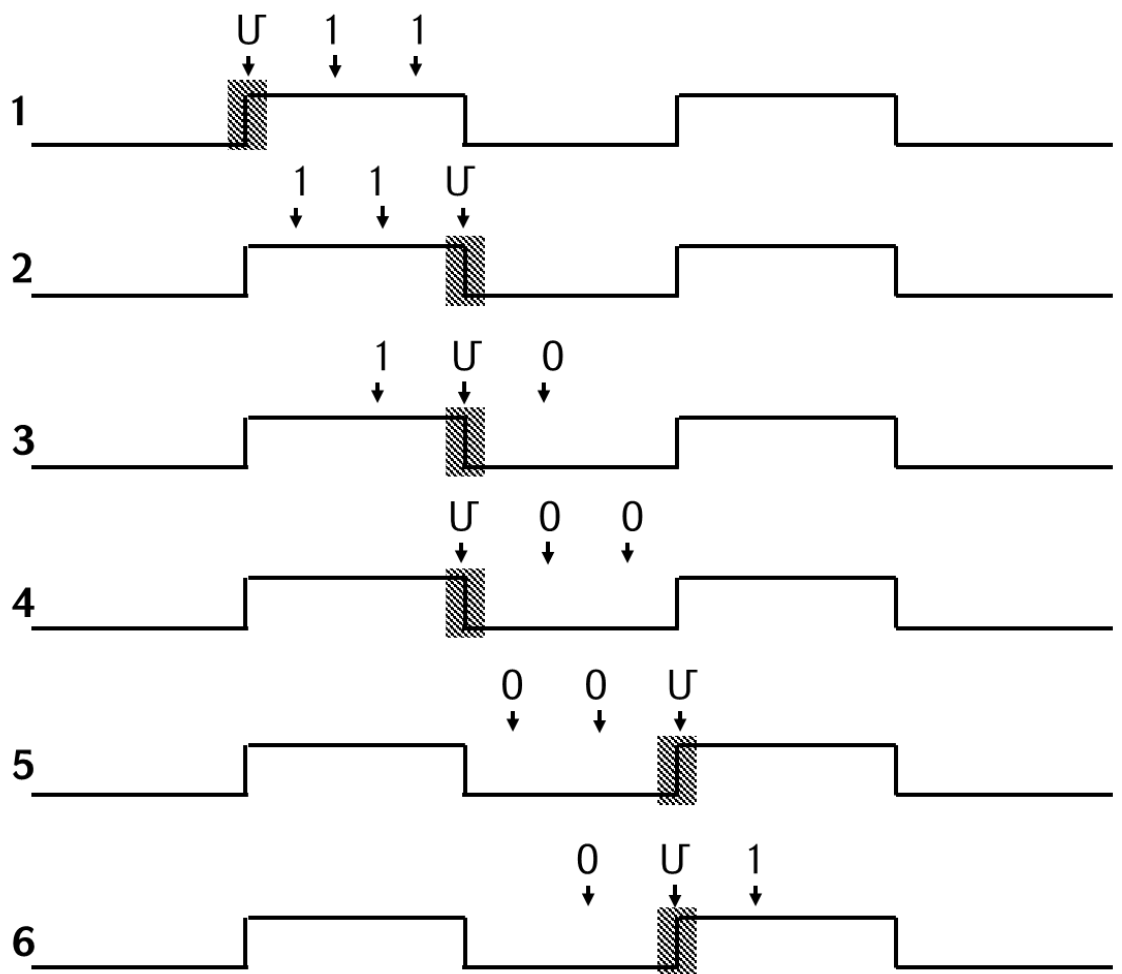
Աղյուսակ 2.4

ՀԱ-ի իսկության աղյուսակը 3 ընտրումների դեպքում

Դեպք #	Ընտրում 1	Ընտրում 2	Ընտրում 3
1	Մ	1	1
2	1	1	Մ
3	1	Մ	0
4	Մ	0	0
5	0	0	Մ
6	0	Մ	1



Նկ. 2.15. Համապատասխանեցնող սխեմայի սկզբունքային կառուցվածքը



Նկ. 2.16. ՍԸ-ի ելքային վեկտորների հավանական տարբերակները

Աղ.2.4-ում ստվերով նշված են clk2-ով clk1-ին համապատասխան կայուն ընտրումները: Վերջիններիցս նրանք, որոնց արժեքը 0 է, ցույց են տալիս, որ clk1-ի և clk2-ի փուլային շեղումը 180 աստիճանից մեծ է: Աղյուսակից երևում է նաև, որ յուրաքանչյուր դեպքի բուլյան ֆունկցիան կազմելու դեպքում, երկու հարակից դեպքերն ինչ-որ հավանականությամբ կարող են տրամաբանական 1 արժեքն ընդունել միաժամանակ: Օրինակ եթե դեպքերին համապատասխանող բուլյան ֆունկցիան նշանակվի Z, իսկ ընտրումները՝ a, b և c, ապա դեպք #1-ի համար կստացվի $Z1=bc$, իսկ #2-ի՝ $Z2=ab$: Եթե $a=b=c=1$, ապա Z1-ը և Z2-ը ընդունում են տրամաբանական բարձր արժեք: Միևնույն ժամանակ աղ. 2.4-ից այս երկու դեպքերում <Ս-ն ընտրում է տարբեր փուլային արժեքներ, ինչը վերջինիս ելքում հանգեցնում է հակասության: Նմանատիպ հակասությունների առաջացման դեպքում անհրաժեշտ է, որ <Ս-ն ընտրի clk2-ի այնպիսի փուլային շեղում, որը կբավարարի երկու դեպքերին էլ: Այդ նպատակով կազմվել է հակասություն առաջացնող ընտրումների վեկտորները <Ս-ի ելքին համապատասխանեցնող լրացուցիչ աղյուսակ (աղ.2.5), այսինքն վերջիններիս ընտրման դեպքում սխեմայի վարքն այլ է:

Աղյուսակ 2.5

Համարժեք բուլյան արժեքով դեպքերի միավորումը

Ընտրումների վեկտոր	Համափուլ ընտրում
1 1 1	Ընտրում 2
0 1 1	Ընտրում 3
1 1 0	Ընտրում 1
1 0 0	Ընտրում 3!
0 0 0	Ընտրում 2!
0 0 1	Ընտրում 1!

Աղյուսակում բացականչական նշանը մատնանշում է clk1-ի և clk2-ի միջև 180-ից մեծ փուլային շեղումը:

clk2-ի փուլային ածանցյալների միջև շեղման չափն ընտրվում է այնպես, որ հնարավոր լինի միայն մեկ մետակայուն ազդանշան, սակայն, 3 ընտրումից երկու ոչ

մետակայունները շարունակում են մնալ սահմանային դեպքեր՝ մեկը $clk1$ -ի դրական ճակատին մոտ, մյուսը՝ բացասական: Առաջարկվող մեթոդի վրա հիմնված համակարգի հուսալիությունը բարձրացնելու համար կարելի է մեծացնել կատարվող ընտրումների թիվը, ինչը կլինի համաձայնեցման համակարգի մակերեսի մեծացման հաշվին:

4 ընտրման դեպքում ՀՄ-ի իսկության աղյուսակը ցուցադրված է աղ. 2.6-ում: Այդ դեպքում առկա են մեկ մետակայուն և 3 կայուն ընտրումներ: Բոլոր տարբերակներում 3-ից երկուսը ընտրվում են $clk1$ -ի դրական և բացասական ճակատների հարևանությամբ՝ դրանցից ամենաքիչը Δt հեռավորության վրա, իսկ երրորդը՝ $clk1$ -ի տրամաբանական մակարդակներից մեկի միջանկյալ դիրքում, որտեղ մետակայունության առաջացումն ուղղակի բացառված է: Այս դեպքում իրար փոխբացառող դեպքերի հավաքածուներն են 1, 3, 5, 7 և 2, 4, 6, 8 տարբերակները:

4 ընտրումներով դեպքի համար ևս մշակվել է համարժեք արդյունքով բուլյան ֆունկցիաներին մեկ ընդհանուր ՀՄ-ի ելք համապատասխանեցնող աղյուսակ:

Նկատի ունենալով իրար փոխբացառող դեպքերի վերոհիշյալ հավաքածուները՝ կարելի է ասել, որ համաձայնեցման ժամանակ՝

- տեղի կունենա միայն առաջին հավաքածուից մեկ դեպք,
- տեղի կունենա միայն երկրորդ հավաքածուից մեկ դեպք,
- յուրաքանչյուր հավաքածուից տեղի կունենա մեկական դեպք:

Առաջին երկու դեպքերում ՀՄ-ը կընտրի համապատասխան փուլով ազդանշանը, իսկ երրորդ դեպքում՝ միաժամանակ տեղի ունեցած երկու դեպքերին բավարարող փուլով ազդանշանը:

Աղյուսակ 2.6

4 ընտրման դեպքում ՀՄ-ի իսկության աղյուսակը

Դեպք #	Ընտրում 1	Ընտրում 2	Ընտրում 3	Ընտրում 4
1	Մ	1	1	1
2	1	1	1	Մ
3	1	1	Մ	0
4	1	Մ	0	0

5	Մ	0	0	0
---	---	---	---	---

Աղ. 2.6-ի շարունակությունը

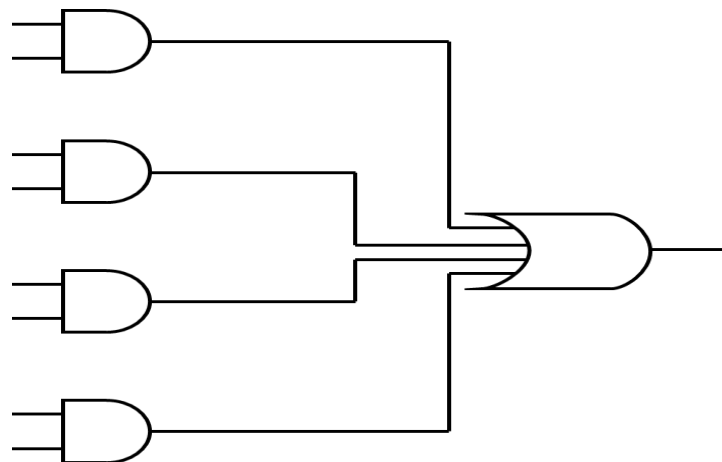
6	0	0	0	Մ
7	0	0	Մ	1
8	0	Մ	1	1

Աղյուսակ 2.7

4 ընտրումների դեպքում համընկնող բուլյան ֆունկցիաների և ՀՄ-ի ելքի
համապատասխանեցումը

Ընտրումների վեկտոր	Համափուլ ընտրում
1 1 1 1	Ընտրում 3
0 1 1 1	Ընտրում 4
1 1 1 0	Ընտրում 2
1 1 0 0	Ընտրում 1
1 0 0 0	Ընտրում 4!
0 0 0 0	Ընտրում 3!
0 0 0 1	Ընտրում 2!
0 0 1 1	Ընտրում 1!

ՀՄ-ի տրամաբանության՝ համափուլ սինքրոազդանշանը ընտրող հատվածը իրենից ներկայացնում է երկու մուտքանի տրաբանական ԵՎ տարրեր, որոնց ելքերը տրամաբանական ԿԱՄ փականի մուտքերն են (նկ. 2.17):



Նկ. 2.17. Համափուլ սինքրոազդանշանը ընտրող սխեման

Վերևում շարադրվածից սինքրոազդանշանի դինամիկ համաձայնեցմամբ մեթոդի մասին կարելի է ընդհանրացնել հետևյալը՝

- Համաձայնեցվում են սինքրոազդանշանները, այլ ոչ թե փոխանցվող տվյալները: Այս առանձնահատկությունը առաջարկվող մեթոդի հիմնական առավելություններից մեկն է, քանի որ սինքրոտիրույթների միջև տեղադրվող համաձայնեցման միջոցների քանակը կախված չէ փոխանցվող ազդանշանի կարգայնությունից և հավասար է 1-ի:
- Արագագործությունը ենթակա է լավարկման: Քանի որ համաձայնեցման միջոցի կառուցվածքի մեծ մասը համակցային տրամաբանություն է, բնական է ակնկալել, որ արագագործությունը գլխավորապես կախված է հենց այդ սխեմաներից, այսինքն անհրաժեշտության դեպքում նախագծողը կարող է լրացուցիչ լավարկել այն:

Համաձայնեցման այս միջոցը կիրառելի է կամայական հաճախականային հարաբերություններով սինքրոտիրույթների համաձայնեցման համար: Այն ունի առկա այլ միջոցների համեմատ տարածման փոքր հապաղում, սակայն տրիգերային համաձայնեցուցիչների համեմատ բարդ իրագործելի է և զբաղեցնում է մեծ մակերես:

2.4. Տարբեր հաճախություններով տակտավորվող թվային սխեմաների համաձայնեցման միջոցների կիրառմամբ նախագծման գործընթացը

Համաձայնեցման միջոցները թվային ԻՍ-երի և դրանց նախագծման գործընթացի [117] անբաժան մասն են կազմում: Դրանք առկա են թվային ստանդարտ բջիջների գրադարաններում [118] կամ նախագծման այլ հավաքածուներում: Ժամանակակից սխեմաների բարդությունը հաշվի առնելով՝ առաջարկվում է թվային սխեմաների նախագծման գործընթաց (նկ.2.18), որի դեպքում համաձայնեցման միջոցները սինքրոտիրույթների միջև ներմուծվում են դեռևս ԻՍ-ի վարքային մոդելի նախագծման փուլում՝ վարքային կամ կառուցվածքային նկարագրերի տեսքով: Ըստ նախագծման գործընթացի՝ համաձայնեցված ԻՍ-ի ռեգիստրային փոխանցման մակարդակի (ՌՓՄ) նկարագրության մշակմանը հաջորդում է ֆունկցիոնալ ստուգման

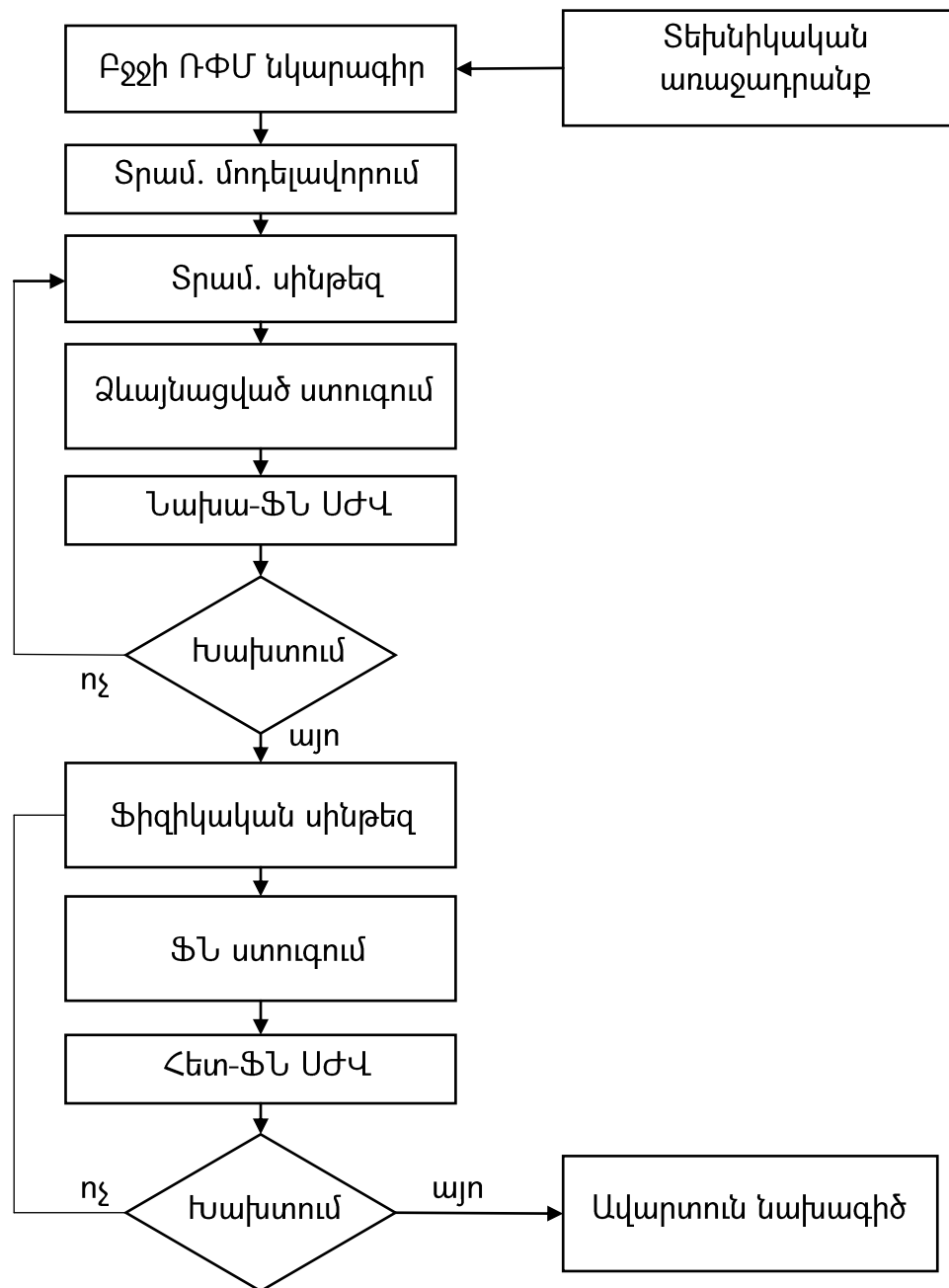
փուլը, որը ենթադրում է ձևայնացված մեթոդներով [119] և նմանակումների միջոցով սխեմայի ֆունկցիոնալության ստուգում: Այս փուլում են ստուգվում իրականացված արձանագրությունները, համաձայնեցուցիչների աշխատանքը և նաև սխեմայի աշխատանքը համաձայնեցման միջոցների առկայության դեպքում: Ֆունկցիոնալ ստուգումից հետո իրականացվում է ՌՓՄ նկարագրության տրամաբանական սինթեզ [120], որի արդյունքում գեներացվում է վարքային նկարագրությանը համարժեք փականային մակարդակի սխեմա: Այս փուլի ավարտից հետո կատարվում է նշված երկու մոդելների տրամաբանական ֆունկցիաները համեմատող ձևայնացված ստուգում: Նախագծման երթուղու հաջորդ փուլն է թեստավորմանը կողմնորոշված նախագծման [121] կառուցվածքների ներմուծումը: Դա որպես կանոն վերաբերում է սովորական տրիգերների՝ սկան տրիգերներով [122] փոխարինմանը, ինչը էապես բարելավում է արտադրված ԻՍ-ի թեստավորումը: Այնուհետև իրականացվում է ԻՍ-ի ֆիզիկական նախագծի (ՖՆ) հատակագծի կազմում [123], որի հիմնական նպատակն է նվազագույնի հասցնել ծրագծման կուտակումները [124-128] և համակարգի ընդհանուր մակերեսը: ՖՆ-ի հատակագծի կազմվելուց հետո տեղի է ունենում բոլոր բջիջների տեղաբաշխում [129] և միջմիացումների ծրագծում [125]: Սխեմայի ամենաերկար միջմիացումները պատկանում են սինթրոազդանշանի բաշխման ցանցին: ԻՍ-ի տարբեր հատվածներում տակտավորող ազդանշանի անկանոն հապաղումների շեղումներից խուսափելու նպատակով իրականացվում է սինթրոազդանշանի բաշխման ցանցի սինթեզ [130-135], որի արդյունքում միջմիացումների վրա տեղադրվում են բուֆերներ և շրջիչներ, որոնք ապահովում են սինթրոազդանշանի կայունությունը ԻՍ-ի ամբողջ մակերեսով: Սինթեզված ՖՆ-ը ենթարկվում է նախագծման կանոնների ստուգման [136]:

Ինչպես նշվեց, համաձայնեցման ուղիների կառուցումն անհրաժեշտ է կատարել նախագծի ՌՓՄ նկարագրության վրա, սակայն դա դեռևս բավարար չէ ՖՆ-ից հետո ԻՍ-ի անհրաժեշտ ֆունկցիոնալությունն ապահովելու համար:

Սինթրոտիրույթների համաձայնեցման սխեմաների ֆունկցիոնալ ստուգումը (ՖՍ) իրականացվում է նախագծի ՌՓՄ նկարագրության վրա: Վերջինս, համեմատած փականային մակարդակի հետ, ավելի պարզ ներկայացում է, և ֆունկցիոնալ

խոտանների մեծ մասը հնարավոր է հայտնաբերել հենց ՌՓՄ-ում: Համաձայնեցման ստուգումը (ՀՍտ) ներկայացնում է սինքրոտիրոյթների նույնականացում, սինքրոտիրոյթի հատում կատարող ազդանշանների հայտնաբերում և սինքրոտիրոյթների միջև առկա տրամաբանության ստուգում:

ՀՍ-ը ներառում է նաև սինքրոտիրոյթների միջև փոխանցվող ազդանշանների դասակարգումը համաձայնեցման արձանագրությունների խմբերի, արձանագրությունների հատկությունների պարզումը և ձևայնացված ստուգմամբ կամ նմանակման միջոցով դրանց որակի հավաստումը: Ֆունկցիոնալ ստուգման փուլում ստուգվում է համակարգի վարքը համաձայնեցուցիչների և սինքրոազդանշանի թրթռոցների առկայության դեպքում:



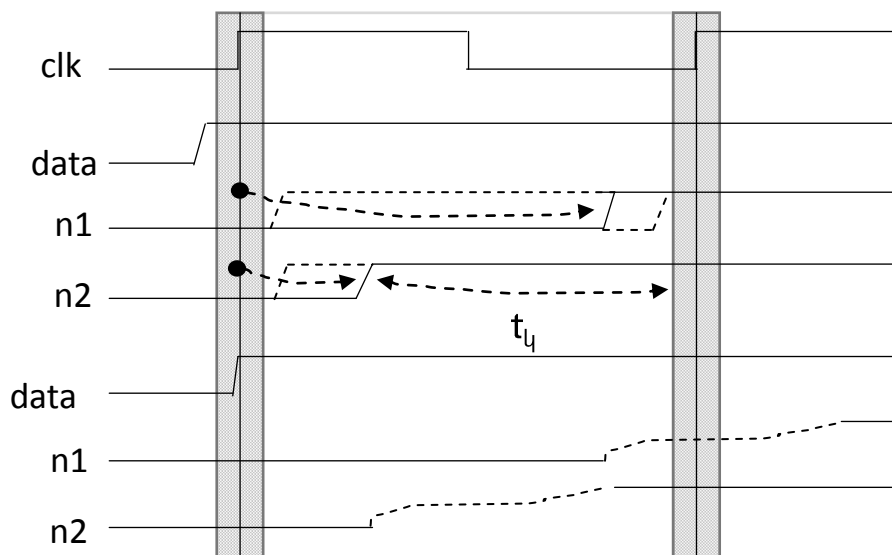
Նկ. 2.18. Թվային նախագծման երթուղին

Նախագծման հետագա փուլերը՝ տրամաբանական և ֆիզիկական սինթեզները, կարող են համաձայնեցուցիչների աշխատանքում առաջացնել որոշակի շեղումներ, որոնք հնարավոր չէր ստուգել ֆունկցիոնալ հավաստման փուլում: Այս խնդրի առկայությունը և ազդեցությունը գնահատելու համար անհրաժեշտ է կատարել նախագծի ժամանակային վերլուծություն:

Ստատիկ ժամանակային վերլուծությամբ (ՍԺՎ) [137-140] ստուգվում է փականային մակարդակի սխեմայի՝ առաջադրված բոլոր ժամանակային սահմանափակումներին բավարարելը: ՍԺՎ-ի հիմնական նպատակն է տվյալ

սինքրոտիզացիոն համար ապահովել ֆիզիկական նախագծումից հետո դրանում գործող տրիգերների հաստատման և պահպանման ժամանակների բավարարումը [138]: Տրիգերային համաձայնեցուցիչների դեպքում մուտքային ազդանշանի՝ տրիգերի ժամանակային սահմանափակումներին ենթարկվելը բավարար չէ՝ երաշխավորելու այդ սարքերի ճշգրիտ ֆունկցիոնալությունը: Համաձայնեցուցիչ՝ մետակայուն վիճակի հակված միջմիացումների՝ սինքրոնության ժամանակային պահանջներին բավարարելը հանգեցնում է ազդանշանի կայունացման ժամանակի նվազման, ինչը էքսպոնենցիալ օրենքով նվազեցնում է խափանման պարբերությունը:

Խնդիրը լուծելու համար համաձայնեցման սխեմաների վրա դրվում են լրացուցիչ ժամանակային սահմանափակումներ, որպեսզի մետակայունության առավելագույն հապաղումը լինի սինքրոազդանշանի պարբերությունից բավարար փոքր և ապահովվի բավարար մեծ կայունացման ժամանակ (նկ. 2.19):



Նկ. 2.19. Մետակայուն միջմիացման ժամանակային սահմանափակումները

Նկ. 2.19-ում պատկերված է data ազդանշանի՝ տրիգերի հաստատման ժամանակից ուշ փոխանջատումը, ինչը, առանց լրացուցիչ ժամանակային սահմանափակումների, հանգեցնում է փոքր կայունացման ժամանակով մետակայուն վիճակի, և այդ ազդանշանը չի հասցնում վերականգնվել մինչև սինքրոազդանշանի հաջորդ ճակատի պահը: Սակայն լրացուցիչ ժամանակային պահանջների սահմանման դեպքում ազդանշանի վերականգնումը տեղի է ունենում անհրաժեշտ պահին:

Կայունացման ժամանակի տևողության մեծացումը՝ ըստ (1.11)-ի, հանգեցնում է խափանման պարբերության մեծացման, և հետևաբար՝ ֆիզիկական սինթեզից հետո համակարգը պահպանում է կայունությունը:

ԻՍ-ի սինքրոտրոյթների հաղորդակցման ժամանակային բնութագրերը ևս ենթարկվում են ՍԺՎ-ի [138]: Սինքրոտրոյթի հատում կատարող ազդանշանի վրա դրվող ժամանակային սահմանափակումներն են՝

- ընդունող տրիգերի մուտքային ազդանշանի փոխանջատման տևողությունը
- համաձայնեցման արձանագրության մաս կազմող ազդանշանների ժամանակային փոխհարաբերությունները

Նշվածներից առաջինի կարևորությունն այն է, որ փոխանջատման երկար տևողությունները հանգեցնում են տրիգերի հաստատման և պահպանման մեծ արժեքների [141, 142], ինչը տրիգերի ելքը դարձնում է մետակայունության նկատմամբ անկայուն: Երկրորդ տիպի սահմանափակումները կարևոր են ֆիզիկական սինթեզից հետո ազդանշանների միջև համապատասխանությունը պահպանելու համար:

Ֆիզիկական սինթեզից հետո համաձայնեցուցիչների անխափան և հուսալի աշխատանքն ապահովելու նպատակով, դրանց վրա դրվում են մի շարք պահանջներ և սահմանափակումներ: Մակարդակային համաձայնեցուցիչն (նկ. 1.17) առաջադրվում են հետևյալ սահմանափակումները՝

- կայունացման բավարար մեծ ժամանակ,
- առաջին տրիգերում տրանզիստորների այնպիսի չափեր, որոնք էապես կնվազեցնեն տրիգերի՝ մետակայուն վիճակի նկատմամբ զգայունությունը,
- մեկ ամբողջական բջջի օգտագործում. ցանկալի չէ համաձայնեցուցիչը նախագծել երկու առանձին տրիգերներով,
- առաջին տրիգերի մուտքին փոխանջատման փոքր տևողություն,
- երկրորդ տրիգերը չպետք է փոխարինվի սկան տրիգերով:

Կայունացման ժամանակի (ԿԺ) ընդունելի արժեքի որոշման համար չկա որևէ հստակ մեթոդ: Դրա արժեքը որոշվում է տվյալ կիրառության համար անհրաժեշտ խափանման պարբերության արժեքից ելնելով: Սինթեզման ժամանակ, հնարավորության սահմաններում, անհրաժեշտ է ապահովել համաձայնեցուցիչներում

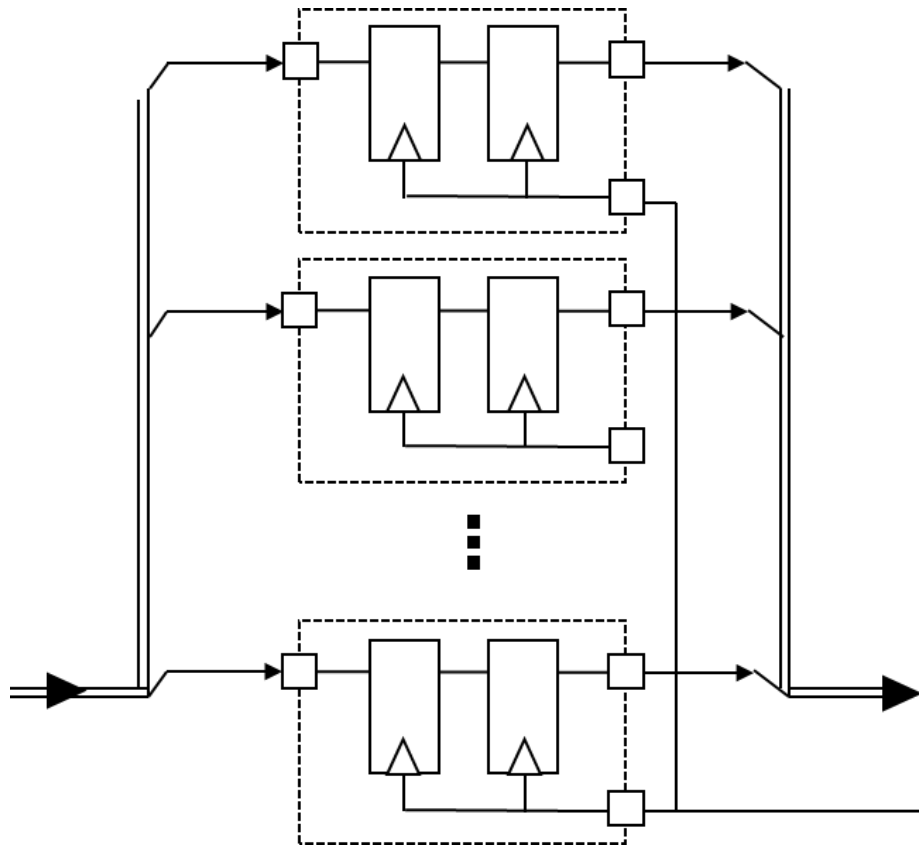
մետակայունության նկատմամբ պակաս զգայուն տրիգերների օգտագործումը: Այդ տրիգերները սովորաբար ավելի արագ են դուրս գալիս մետակայուն վիճակից և ունեն նեղ մետակայունության պատուհան [143]: Հարկ է նշել, որ ՌՓՄ նկարագրության ոչ բոլոր կառուցվածքներից է հնարավոր սինթեզել նախագծման ԹՄՏԳ-երում առկա մետակայուն տրիգերներ, քանի որ անհրաժեշտ ելուստներով կամ բնութագրերով տարրը հնարավոր է բացակայի գրադարանից: Նման դեպքերում այդ տրիգերները նախագծվում են հատուկ տվյալ կիրառության համար: Մյուս կողմից եթե ԹՄՏԳ-ում առկա են տրիգերային համաձայնեցուցիչների ամբողջական բջիջներ, սինթեզ կատարող ծրագրին [144] անհրաժեշտ է տալ ցուցումներ, որպեսզի դրանք հնարավորինս լիարժեք օգտագործվեն արդյունարար սխեմայում:

Նախագծման մեկ այլ մոտեցման դեպքում համաձայնեցուցիչների կառուցվածքային բջիջները տեղադրվում են անմիջապես համակարգի ՌՓՄ նկարագրի մեջ: Այս մոտեցումն ավելի է պարզեցնում սինթեզման գործընթացը և անմիջապես ապահովում է սարքի ժամանակային պարամետրերի հաշվի առնումը, սակայն ՌՓՄ նկարագրությունը դառնում է տեխնոլոգիական գործընթացից կախված: Համաձայնեցուցչի կառուցվածքային նկարագրի ներդրումը կիրառական և հարմար մոտեցում է կոնկրետ նախագծերի համար, սակայն ՄՍ բլոկերը [145] նախագծելիս ժամանակ դրա կիրառումը դառնում է աննպատակահարմար:

Տրիգերային համաձայնեցուցիչների մուտքային ազդանշանների փոխանջատման հապաղումները անհրաժեշտ արժեքի բերելու նպատակով կիրառում են բուֆերներ:

Սկան տրիգերների՝ մուլտիպլեքսորի հաշվին լրացուցիչ հապաղում ունենալու պատճառով խափանման պարբերության վատթարացման մեկ այլ աղբյուր է սկան կառուցվածքների ներմուծումը [146]: Տրիգերային համաձայնեցուցչի առաջին տրիգերի փոխարինումը սկան տրիգերով չունի կրիտիկական որևէ ազդեցություն: Այդ ազդեցությունը շոշափելի է երկրորդ տրիգերը սկան տրիգերով փոխարինելիս: Երկու դեպքում էլ սկան ներմուծումն ավելորդ է, քանի որ երկու տրիգերները միացված են տեղաշարժող ռեգիստրում, և դրանց միջև չկա համակցային սխեմա:

Բազմակարգ ազդանշանների համաձայնեցման ժամանակ յուրաքանչյուր կարգ կարելի է համաձայնեցնել մեկական մակարդակային համաձայնեցուցչով [147] (նկ. 2.20):



Նկ. 2.20. Բազմակարգ ազդանշանի համաձայնեցումը

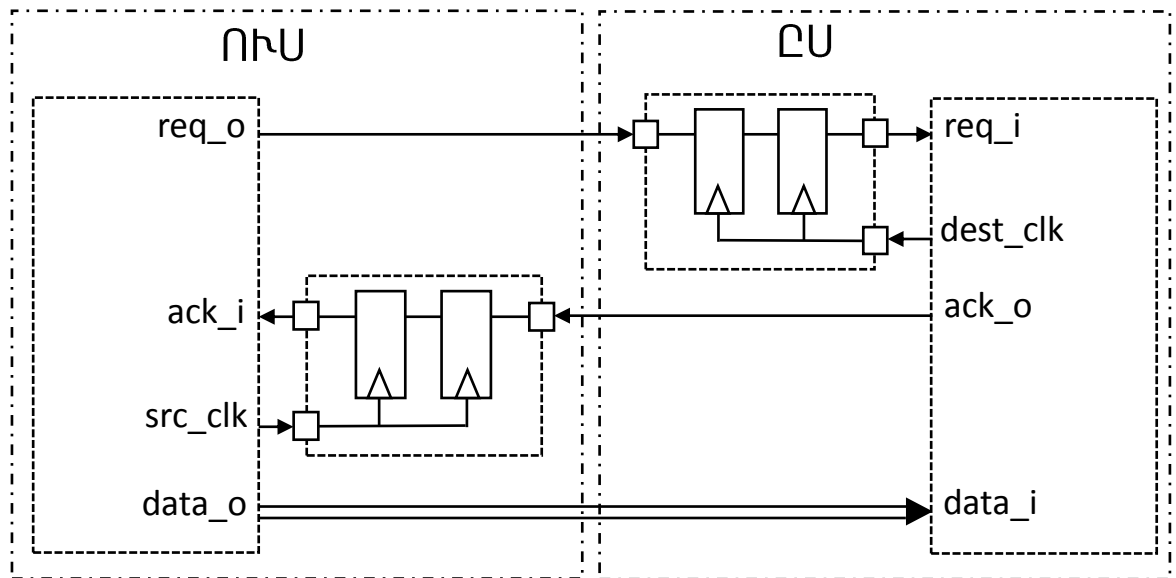
Նկ. 2.20-ում [148] պատկերված եղանակով բազմակարգ ազդանշանների համաձայնեցման ժամանակ դրվում է պահանջ, որ մուտքային կարգերից միաժամանակ միայն մեկը կարող է փոխանցատվել: Դա երաշխավորելու համար իրականացվում է ազդանշանի՝ Գրեյի մեթոդով կոդավորում [149]: Պատկերված զուգահեռ համաձայնեցուցիչը (ԶՀ) [148] ասինքրոն հերթահենք եղանակի հիմնական տարրն է: Դրա առաջ դրվում են հետևյալ ֆունկցիոնալ պահանջները.

- համաձայնեցման ուղու վրա չպետք է լինի համակցային սխեմա,
- մուտքային տվյալները պետք է Գրեյի կոդով ներկայացվեն:

ԶՀ-ի կիրառումը հանգեցնում է համաձայնեցվող ազդանշանի կորստի այն դեպքում, երբ ՌԻՍ ՍԱ-ի հաճախականությունը մեծ է ԸՍ ՍԱ-ինից:

Ֆիզիկական սինթեզի ժամանակ ուղիների լավարկումը ՋՀ-ի առանձին մասերում կարող է հանգեցնել սինքրոազդանշանի տարբեր հապաղումների, ինչը անխուսափելիորեն հանգեցնում է համաձայնեցուցչի խափանման:

Ինչպես նշվել է 1.2-ում, բազմակարգ ազդանշանների համաձայնեցման մեկ այլ եղանակ է համաձայնեցման արձանագրություն իրականացնող համաձայնեցուցիչը (ՀԱԻՀ) (նկ. 2.21) [150]:



Նկ. 2.21. Համաձայնեցման արձանագրություն իրականացնող համաձայնեցուցիչը

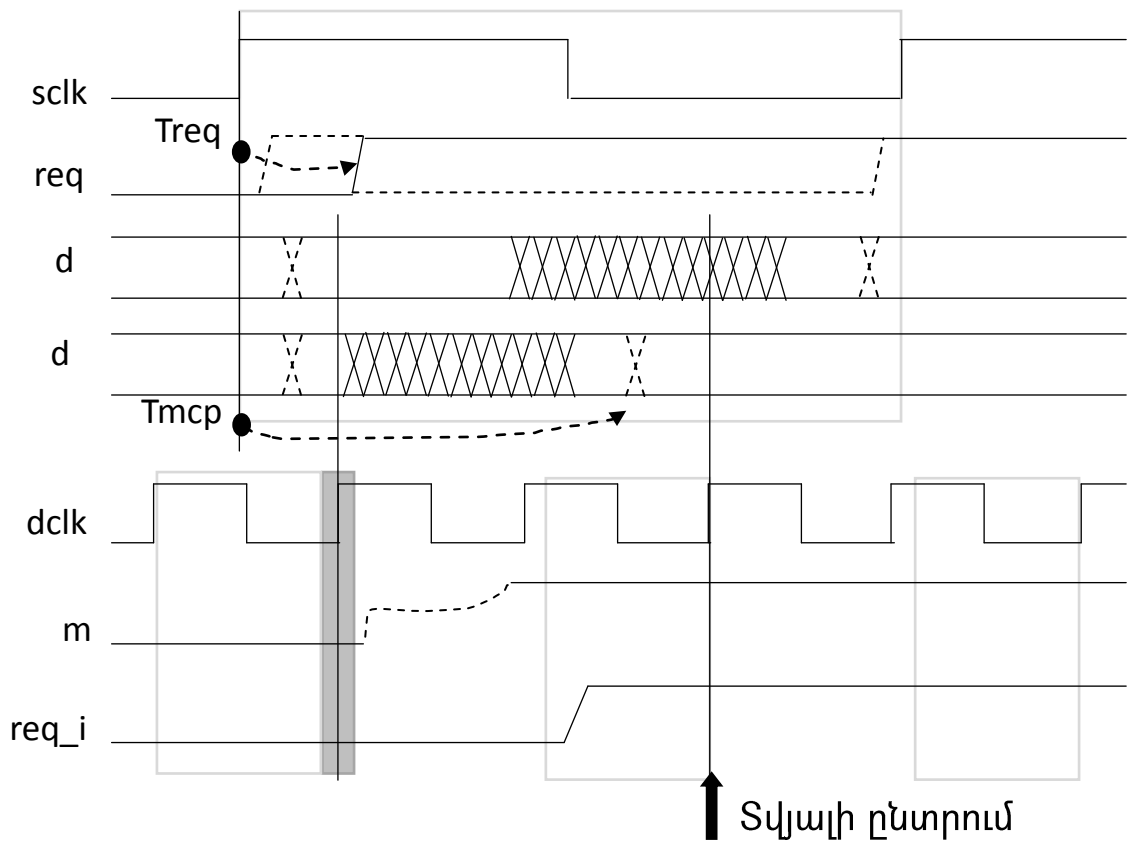
ՌՓՄ մակարդակում ՀԱԻՀ-երին առաջադրվում և ստուգվում են հետևյալ պահանջները՝

- ձեռքսեղմման արձանագրության իրականացում,
- տվյալային կարգերի կայունություն, որը պետք է ապահովվի, քանի դեռ «պատրաստ» ազդանշանը չի ակտիվացվել:

Համաձայնեցման այս եղանակի կիրառման դեպքում անհրաժեշտ է տվյալային ազդանշանի և ղեկավարող կարգերի միջև ապահովել որոշակի ժամանակային կապակցվածություն: Մասնավորապես, դրանք պետք է ունենան այնպիսի հապաղում, որը կապահովի ճիշտ ազդանշանի ընդունումը ԸՍ-ի միջոցով: Եթե համաձայնեցման ուղին սինթեզող ծրագրային գործիքում նշված է որպես «սխալ ուղի», և դրա վրա դրված են ոչ ճշգրիտ սահմանափակումներ, ապա ֆիզիկական սինթեզի արդյունքում

կարող է առաջանալ մեծ հապաղում՝ հանգեցնելով սխալ արժեքների ընդունման (նկ. 2.22):

Նկ. 2.22-ում req և d ազդանշանները դեկավարվում են համեմատաբար դանդաղ սինքրոազդանշանով: Այդ պատճառով req ազդանշանը ԸՍ-ին է հասնում ավելի շուտ, քան տվյալային վերջին կարգը՝ խախտելով ընդունող տրիգերի պահպանման ժամանակը: Դա առաջացնում է մետակայունություն, ինչը հանգեցնում է փոխանջատվող d ազդանշանի ընտրման և ձեռքսեղմման արձանագրությունը խափանման:



Նկ. 2.22. Ձեռքսեղմման արձանագրության՝ ուղու մեծ հապաղումներով պայմանավորված խափանումը

Նկ. 2.22-ում երկրորդ d ազդանշանի վրա դրված է ԸՍ ՍԱ-ի նկատմամբ լրացուցիչ սահմանափակում, ինչի արդյունքում տվյալային կարգերը ճիշտ համաձայնեցվում են:

Համաձայնեցուցիչների աշխատանքի ճշտությունը նախագծման ընթացքում կարելի է ստուգել մի քանի այլ եղանակներով, ինչպիսիք են՝ ՌՓՄ և փականային մակարդակներում իրականացվող ժամանակային ստուգումները, սինքրոտիզացիայի

հատման միջանկյալ կառուցվածքների վերլուծությունը և ՍԺՎ հավելյալ սահմանափակումները:

2.5. Տարբեր հաճախություններով տակտավորվող թվային սխեմաների համաձայնեցման առաջարկվող միջոցների արդյունավետության գնահատումը

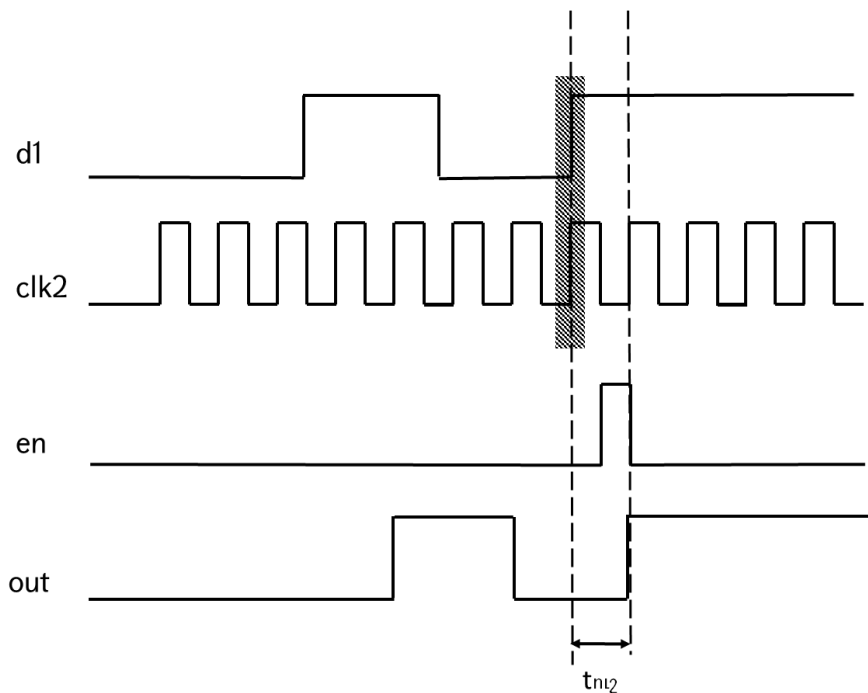
Ժամանակակից ԻՍ-երի կարևորագույն մակրոբնութագրերի՝ արագագործության և էներգասպառման վրա դրվող պահանջները բավարարելու նպատակով համաձայնեցնող սխեմաների որոշ բնութագրերին ևս ներկայացվում են խիստ սահմանափակումներ: Դրանց կարևորագույն բնութագրերն են [66].

- Արտադրողականություն. սահմանվում է որպես նվազագույն ժամանակ, որը պետք է առկա լինի երկու իրար հաջորդող տվյալների փոխանակման գործողությունների միջև, որպեսզի համակարգն աշխատի ըստ նախատեսված ալգորիթմի:
- Ուշացում կամ հապաղում, որի ընթացքում փոխանցվող ազդանշանը համաձայնեցման միջոցի (ՀՄ) մուտքից ելք է տեղափոխվում:
- Համաձայնեցվող հաճախականություններ. մատնանշում են ՌԻՍ-ի և ԸՍ-ի հաճախականային այն հարաբերությունները, որոնցով տակտավորվող սինքրոտիզմները կարող են համաձայնեցվել տվյալ համաձայնեցման միջոցով:
- Էներգասպառում. ՀՄ-երը, ներդրվելով սինքրոտիզմների միջև, էներգասպառման որոշակի բաժին են ավելացնում ընդհանուր համակարգի էներգասպառմանը: Այդ պատճառով նշված բնութագրի ուսումնասիրությունը խիստ կարևոր է ներկայացնում ԻՍ-երի նախագծման գործընթացում:

ԻՍ-երի արագագործության տեսանկյունից սխեմայի արտադրողականությունը դրա կարևորագույն բնութագրերից մեկն է: ՀՄ-երը համակարգերում չեն իրականացնում հաշվողական որևէ ֆունկցիա, դրանց նպատակն է նվազեցնել կամ վերացնել ասինքրոն սխեմաների տվյալների փոխանակման ժամանակ առաջացնող մետակայունության բացասական ազդեցությունները ԻՍ-ի ֆունկցիոնալ

գործունեության վրա: Այդ պատճառով, լինելով ոչ ֆունկցիոնալ սխեմաներ, դրանք պետք է հնարավորինս փոքր ազդեցություն ունենան սինքրոտիզացիայի միջև տվյալների փոխանակման արտադրողականության և ընդհանուր համակարգի էներգասպառման վրա և ունենան փոքր մուտք-ելք հապաղում:

Մետակայունության հայտնաբերմամբ և բնականոն վիճակի վերականգնմամբ ՀՄ-ի արտադրողականությունն ու մուտք-ելք հապաղումը հնարավոր է հաշվարկել դրա ժամանակային դիագրամից ելնելով (նկ. 2.23): Նկ.2.8-ում պատկերված համաձայնեցման եղանակի մուտքին հնարավոր են հետևյալ դեպքերը. 1) մուտքային ազդանշանն ունի նույն արժեքը, ինչ նախորդ պարբերության ժամանակ, 2) մուտքային ազդանշանը փոխանջատվում է:



Նկ. 2.23. Մետակայունության հայտնաբերմամբ և բնականոն վիճակի վերականգնմամբ ՀՄ-ի ժամանակային դիագրամը

- 1) Նախորդ և ներկա պարբերությունների ժամանակ ՀՄ-ի մուտքին տրվում է տրամաբանական 0 արժեք: Այդ դեպքում q2 տրիգերի ելքին միացված ԲԱՑԱՌՈՂ ԿԱՄ տարրի ելքը զրո է, իսկ q1 տրիգերի ԲԱՑԱՌՈՂ ԿԱՄ-ՈՉ տարրի ելքը՝ տրամաբանական 1: *en* ազդանշանն, այդ դեպքում, ընդունում է

տրամաբանական 0 արժեք, ինչը նշանակում է, որ կարգի կորուստ տեղի չի ունեցել: $e_n = 0$, հետևաբար՝ $d_2 = q_2$, իսկ d_1-d_2 հապաղումը՝

$$t_{d_1-d_2} = T_{UL} + t_{q_2} + t_{\text{համակցային}} \quad (2.3)$$

2) Տեղի է ունեցել փոխանջատում:

ա) q_2 -ի ելքը բնականոն փոխանջատվում է: Այդ դեպքում q_2 -ին միացված ԲԱՑԱՌՈՂ ԿԱՄ տարրի ելքը տրամաբանական զրո է, հետևաբար՝ $e_n = 0$ և $d_2 = q_2$, իսկ d_1-d_2 հապաղումը կրկին որոշվում է (2.3)-ով,

բ) q_2 -ի ելքը մետակայուն է, և տեղի է ունեցել կարգի կորուստ: Քանի որ q_2 -ը չի փոխանջատվել, ուստի դրան միացված ԲԱՑԱՌՈՂ ԿԱՄ տարրի ելքը տրամաբանական 1 է: Մակարդակով ղեկավարվող տրիգերի ելքը արդեն հավասարվել է մուտքին. ԲԱՑԱՌՈՂ ԿԱՄ-ՈՉ տարրի ելքը տրամաբանական 1 է: $e_n = 1$, հետևաբար՝ $d_2 = !q_2$, իսկ d_1-d_2 հապաղումը՝

$$t_{d_1-d_2} = T_{UL} + t_{\text{համակցային}} \quad (2.4)$$

Մետակայունության առաջացման դեպքում ՀՄ-ի հապաղումը փոքր է բնականոն դեպքի հապաղումից q_2 տրիգերի հապաղման չափով, քանի որ ազդանշանը ոչ թե տարածվում է q_2 տրիգերով, այլ ձևավորվում է համակցային տրամաբանության միջոցով:

Մշակված ՀՄ-ի աշխատանքը ստուգվել է սինքրոտիզմների միջև կարգերի՝ փոքր հաճախականությամբ սինքրոտիզմից մեծ հաճախականությամբ սինքրոտիզմի փոխանակությամբ՝ ընդունող սինքրոտիզմի մուտքում ապահովելով բնականոն և մետակայուն վիճակներ: Կատարվել են առաջարկվող ՀՄ-ի մոդելավորումներ, և ընդհանրացվել են ստացված արդյունքները (աղ. 2.8):

Աղյուսակ 2.8

Մոդելավորման արդյունքները

Ուղարկող սինքրոտիզմ ($U < g$)	Ստացող սինքրոտիզմ ($U < g$)	Պարբ. (նվ)	ՀԵ-ի հապաղում (նվ)	ՀԵ-ի հապաղում մետակայունության դեպքում (նվ)	%
330	467	2,14	2,24	2,23	0,44

330	568	1,76	1,86	1,84	1
330	735	1,36	1,46	1,45	0,6

Աղ. 2.8-ի շարունակությունը

330	870	1,15	1,25	1,24	0,8
330	1064	0,94	1,04	1,03	0,9
330	1408	0,71	0,809	0,808	0,12
330	1724	0,58	0,675	0,67	0,74
330	2080	0,48	0,578	0,575	0,52

Ստացված արդյունքները ցույց են տալիս, որ մշակված ՀՄ-ի գ2 տրիգերի ելքում բնականոն և մետակայուն վիճակներում դրա հապաղումը, ընդունող տարբեր հաճախականությունների դեպքում, առավելագույնը տարբերվում է 1% - ով:

Առաջարկվող ՀՄ-ն արձանագրություն իրագործող համաձայնեցման միջոցների համեմատ ունի ավելի փոքր մուտք-ելք հապաղում, որը հավասար է 1 տակտային պարբերության տևողության: Երկուական կարգերի յուրաքանչյուր հաջորդ փոխանցումը կարող է տեղի ունենալ նախորդից մեկ տակտային պարբերություն հետո: Այս մեթոդի կիրառման դեպքում ընդունող սինքրոազդանշանի հաճախականության և ուղարկող սինքրոազդանշանի հաճախականության հարաբերությունը կարող է լինել կամայական՝ ի տարբերություն հերթի վրա հիմնված ՀՄ-ի, որի դեպքում այդ հարաբերությունը կարող է հավասար լինել առավելագույնը երեքի:

Համեմատվել են հիշատակված համաձայնեցման միջոցների հապաղումները, ինչից երևում է առաջարկվող եղանակի առավելությունը (աղ. 2.9):

Աղյուսակ 2.9

Համաձայնեցման տարբեր միջոցների հապաղումները

Ուղարկող սինքրոտի րյթ (ՄՀg)	Ստացող սինքրոտի րյթ (ՄՀg)	Առաջարկվող եղանակ(նվ)	ՍՀ եղանակ (նվ)		Հերթահենք եղանակ (նվ)	
			Մին.	Մաքս.	Մին.	Մաքս.
330	467	2,24	4,28	6,42	6,42	8,56
330	568	1,86	3,52	5,28	5,28	7,04

330	735	1,46	2,72	4,08	4,08	5,44
330	870	1,25	2,3	3,45	3,45	4,6

Աղ. 2.9-ի շարունակությունը

330	1064	1,04	1,88	2,82	2,82	3,76
330	1408	0,809	1,42	2,13	2,13	2,84
330	1724	0,675	1,16	1,74	1,74	2,32
330	2080	0,578	0,96	1,44	1,44	1,92

Աղյուսակում զետեղված արդյունքներից երևում է, որ մետակայունության հայտնաբերմամբ համաձայնեցման մեթոդն ապահովում է համաձայնեցման նվազագույնը երկու անգամ ավելի փոքր ուշացում, քան հայտնի մյուս մեթոդները:

Մշակված ՀՄ-ի թերությունն այն է, որ ուղարկող սինքրոազդանշանի հաճախականությունը չի կարող լինել ավելի մեծ, քան ընդունողինը: Այս հանգամանքը առաջարկվող համաձայնեցման միջոցի սահմանափակումն է:

Սինքրոազդանշանների դինամիկ համաձայնեցման մեթոդի դեպքում համակարգի հապաղումը որոշվում է հետևյալ կերպ՝

$$t_{\text{հապ}} = t_{\text{շեղ}} + t_{\text{տրիգ}} + t_{\text{ՀՄ}}, \quad (2.5)$$

որտեղ $t_{\text{ՀՄ}}$ -ը ներկայացնում է ՀՄ-ի մուտք-ելք հապաղումը, իսկ $t_{\text{տրիգ}}$ -ը՝ ՍԸ-ի մեկ տրիգերի հապաղումը: Այս հապաղումը, պայմանավորված լինելով բացառապես համակցային տրամաբանությամբ, հնարավոր է լավարկել՝ դրա անհրաժեշտ մեծությունն ապահովելու նպատակով:

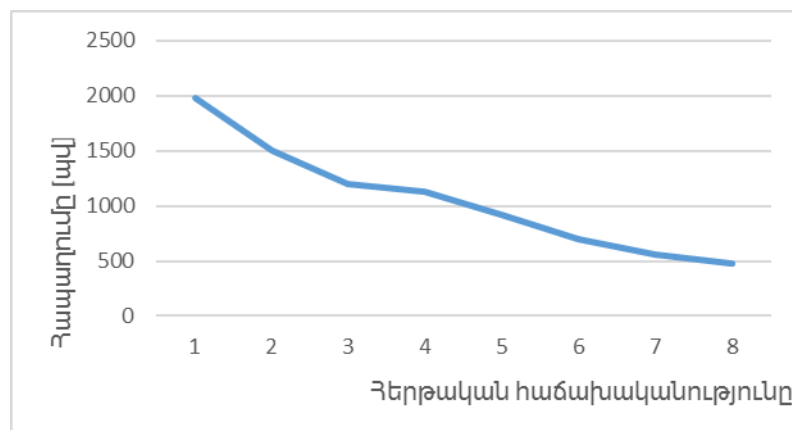
Համաձայնեցվող սինքրոազդանշանների հաճախականային սահմանը կարելի է որոշել (2.2) անհավասարությունից: Դրանից երևում է, որ ՈԽՍ ՍԱ-ի առավելագույն հաճախականությունը սահմանափակվում է կիսապարբերության ընթացքում առնվազն 3 ընտրում կատարելու անհրաժեշտությամբ և տրիգերների արգելված տիրույթի լայնությամբ: Կատարված մոդելավորումների արդյունքները (Աղյուսակ 2.10), որոնք ստացվել են տարբեր հաճախականությունների համար չափելով ՀՄ-ի հապաղումը և էներգասպառումը: Ստացված արդյունքները ցույց են տալիս, որ տվյալ ընտրման հաճախականության դեպքում ՀՄ-ի հապաղումը չի գերազանցում մեկ տակտային պարբերությունը:

Հաճախականությունից կախված՝ այս միջոցի հապաղումն ու էներգասպառումը կրում են հետևյալ փոփոխությունները (նկ. 2.24, 2.25):

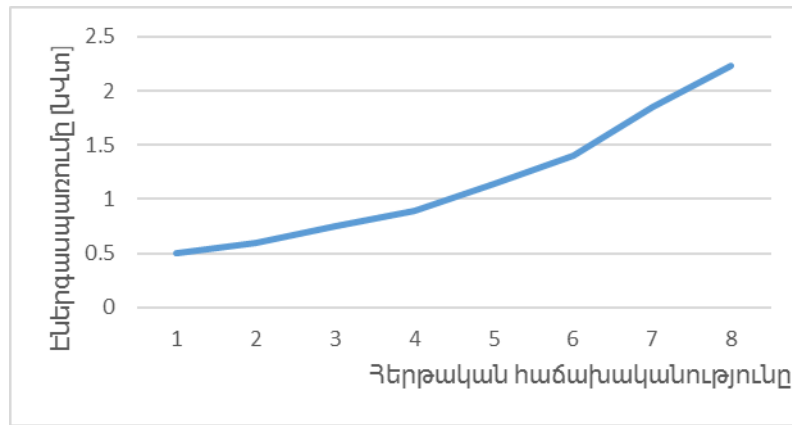
Աղյուսակ 2.10

Հաճախություններին համապատասխանող հապաղումներն ու էներգիան

Տակտավորման հաճախականություն (ՄՀց)	Հապաղում (սվ)	Էներգիա (նՎտ)
467	1980	0,5
568	1500	0,6
735	1200	0,75
870	1125	0,894
1064	920	1,14
1408	700	1,4
1724	564	1,85
2080	480	2,23



Նկ. 2.24. Հապաղման՝ հաճախականությունից ունեցած կախվածությունը



Նկ. 2.25. Էներգասպառման՝ հաճախականությունից ունեցած կախվածությունը

Այսպիսով մշակվել է համաձայնեցման երկու մեթոդ, որոնց դեպքում համաձայնեցման հապաղումը ընդունում է ընդունող սինքրոտիզացիոն տակտային պարբերությանը մոտ արժեք:

Եզրակացություններ

1. Առաջարկվել է մետակայուն վիճակի հայտնաբերմամբ և բնականոն վիճակի վերականգնմամբ համաձայնեցման մեթոդը, որի դեպքում ազդանշանի վերաթողարկման շնորհիվ ապահովվում է հապաղման առնվազն 2 անգամ շահում՝ ՀՄ-ի տրամաբանական փականների քանակի աննշան ավելացման հաշվին:
2. Մշակվել է սինքրոազդանշանի դինամիկ համապատասխանեցման համաձայնեցման մեթոդը, որը տրամաբանական փականների քանակի մեծացման և իրականացման որոշակի բարդության հաշվին ապահովում է ընդունող սինքրոազդանշանի պարբերությունը չգերազանցող համաձայնեցման հապաղում:
3. Առաջարկվել է նախագծման գործընթաց, որն ապահովում է համաձայնեցման մշակված միջոցների՝ գործընթացի, լարման և ջերմաստիճանի համեմատ զգայունության էական նվազեցում և համաձայնեցված համակարգի կայունություն:

4. Ստեղծված համաձայնեցման մեթոդների վերլուծությունը ցույց է տալիս, որ դրանք օժտված են բարձր արդյունավետությամբ, քանի որ զբաղեցրած մակերեսի աննշան ավելացման հաշվին ապահովում են հապաղման և, հետևաբար, արագագործության էական շահում:

ԳԼՈՒԽ 3. ՏԱՐԲԵՐ ՀԱՃԱԽՈՒԹՅՈՒՆՆԵՐՈՎ ՏԱԿՏԱՎՈՐՎՈՂ ԹՎԱՅԻՆ ՍԽԵՄԱՆԵՐԻ ՀԱՄԱՁԱՅՆԵՑՄԱՆ ՄԻՋՈՑՆԵՐԻ ԻՐԱԿԱՆԱՑՈՒՄԸ

Ժամանակակից ԻՍ-երի բարդ կառուցվածքը, դրանցում սինքրոտիզմների քանակը և փոխանցվող տվյալների կարգայնությունը դժվարացնում են նախագծողի կողմից համաձայնեցման միջոցների ներմուծումը համակարգ: Առաջացած բարդությունները ԻՍ-ի նախագծման գործընթացի վրա ունեն հետևյալ բացասական ազդեցությունները՝

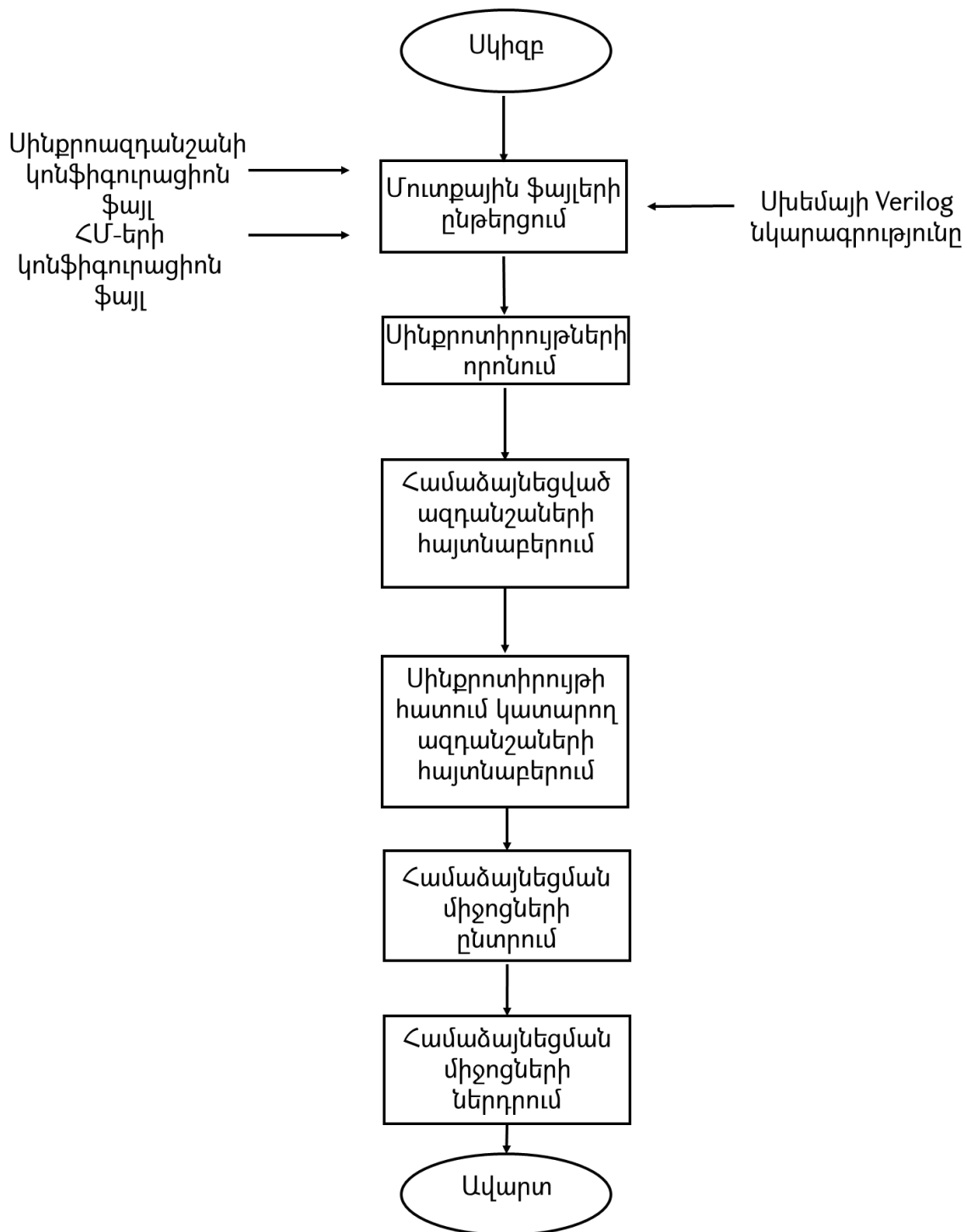
- նախագծման և ստուգման ժամանակների էական ավելացում,
- մարդկային սխալի մեծ հավանականություն:

Վերոհիշյալ ազդեցություններն ուղղակիորեն ազդում են ԻՍ-երի շուկա դուրս գալու ժամանակի և գնի վրա:

Խնդիրը լուծելու համար մշակվել է SmartSync ծրագրային գործիքը [75], որը, ստանալով համակարգի՝ Verilog սարքավորումների նկարագրման լեզվով գրված ռեգիստրային փոխանցման կամ փականային մակարդակի նկարագրությունը, և կոնֆիգուրացիոն ֆայլերը, հայտնաբերում է հաղորդակցվող սինքրոտիզմները, սինքրոտիզմների հատում կատարող ազդանշանները և սխեմայի՝ տարբեր տակտավորման հաճախականություն ունեցող ու ազդանշաններ փոխանակող հանգույցների միջև ներդնում է ատենախոսությունում առաջարկված համաձայնեցման միջոցները՝ ելքում ստանալով արդեն համաձայնեցված սինքրոտիզմներով ռեգիստրային փոխանցման կամ փականային մակարդակի նկարագրություն:

SmartSync ծրագրային միջոցը մշակվել է Python [151] օբյեկտա-կողմնորոշված ծրագրավորման լեզվով՝ օգտագործելով վերջինիս numpy և PyQt [152] ֆունկցիաների գրադարանները: Ծրագիրն ունի հարմարավետ և հեշտ յուրացվող ինտերֆեյս, ինչը հնարավորություն է տալիս նախագծողին կարճ ժամանակում վարժվել դրանով սխեմաներ համաձայնեցնելուն: SmartSync-ը կոմպիլացվել է երկու տարբերակով՝ CentOS 6.0 [153], Windows 7 [154] և ավելի նոր օպերացիոն համակարգերում աշխատելու նպատակով: Մշակված ծրագրային գործիքով իրականացված համաձայնեցումների գնահատումները ցույց են տվել, որ SmartSync-ով

համաձայնեցման միջոցների ներդնումը (նկ. 3.1) էապես կրճատում է ինտեգրալ սխեմայի նախագծման և սինթրոտիրոյթների համաձայնեցման ստուգման ժամանակները՝ բացառելով սինթրոտիրոյթի հատում կատարող ազդանշանների մետակայուն վիճակում հայտնվելու հավանականությունը: SmartSync գործիքի ստեղծած Verilog [155] նկարագրության ճշտությունը ստուգվում է VCS տրամաբանական մակարդակի մոդելավորման գործիքով և ապա նախագիծը մտնում է ԱՆԵ փուլ:



Նկ. 3.1. SmartSync ծրագրի աշխատանքը նկարագրող բլոկ-դիագրամը

Մուտքային կոնֆիգուրացիոն ֆայլերը և համաձայնեցման ենթակա սխեման ծրագրին տալուց հետո, վերջինս, համակարգի վարքային նկարագրության մեջ իրականացնում է սինքրոտիրույթների որոնում և սինքրոազդանշանների նույնականացում: Բոլոր սինքրոազդանշանները հայտնաբերելուց հետո կատարվում է դրանցով տակտավորվող ազդանշանների որոնում: Նշված տվյալները

հավաքագրելուց հետո իրականացվում է համաձայնեցման միջոցների ներդրում, որին հաջորդում է համաձայնեցված համակարգի VCS մոդելավորման փուլը:

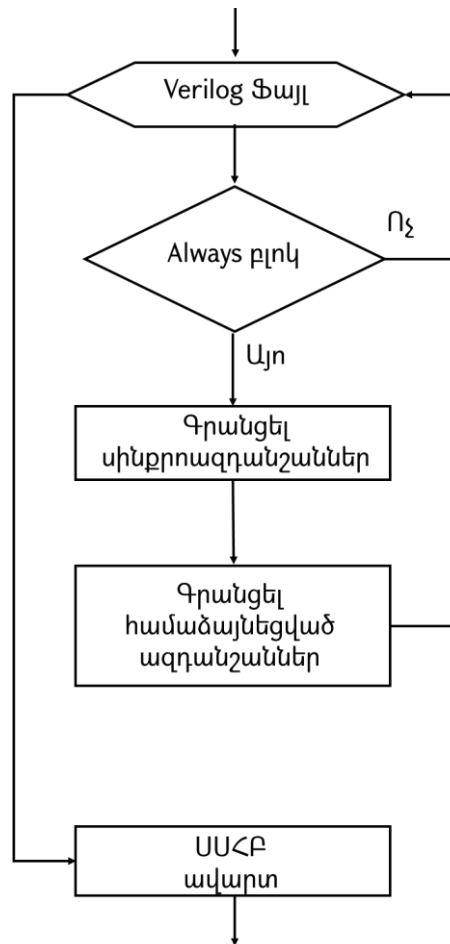
3.1. Տարբեր հաճախություններով տակտավորվող թվային սխեմաների համաձայնեցման ծրագրային միջոցի աշխատանքի սկզբունքը

SmartSync ծրագրային գործիքն ունի հետևյալ սկզբունքային կառուցվածքը՝

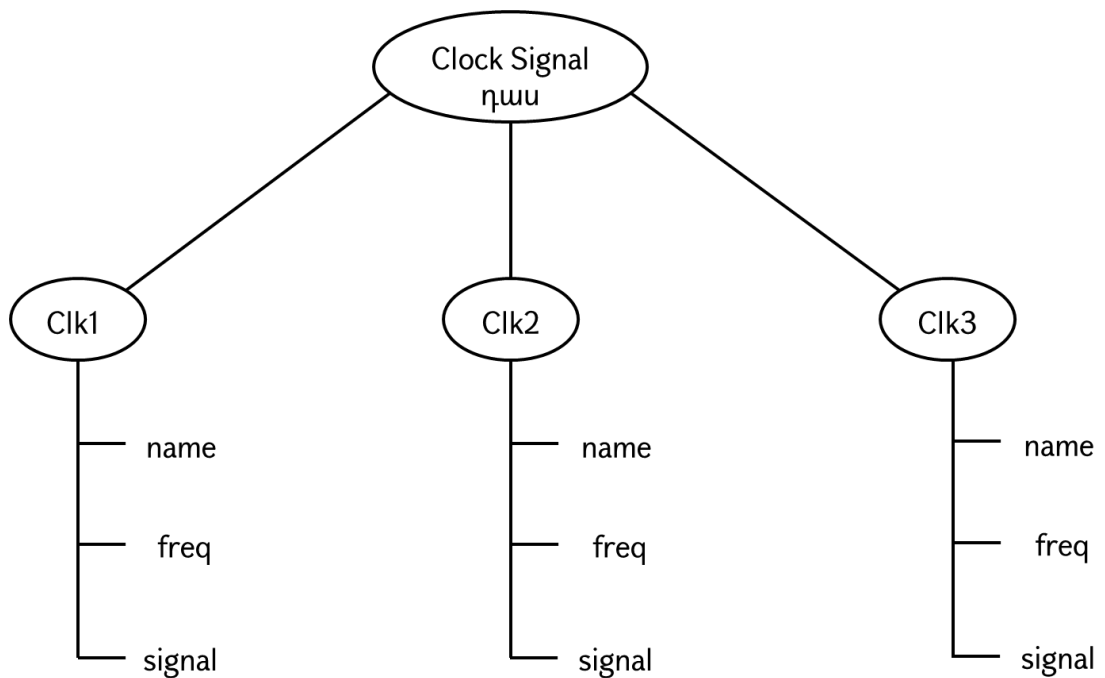
- սինքրոտիզմներ ու սինքրոազդանշաններ հայտնաբերող բլոկ (ՍՍՀԲ),
- վերլուծող բլոկ (ՎԲ),
- համաձայնեցման միջոցների ընտրման բլոկ (ՀՄԸԲ),
- համաձայնեցման միջոցների ներդրման բլոկ (ՀՄՆԲ):

ՍՍՀԲ-ը, օգտվելով կոնֆիգուրացիոն ֆայլերում տրված սինքրոազդանշանների անուններից, Verilog նկարագրությունից հայտնաբերում է դրանք: Ինչպես երևում է ՍՍՀԲ-ի աշխատանքը նկարագրող բլոկ-դիագրամից (նկ. 3.2), դա իրականացվում է Verilog-ի always բլոկերի վերլուծության միջոցով: Ծրագիրը, տող առ տող ընթերցելով ՌՓՄ նկարագրությունը, որոնում է always բլոկները, որոնց միջոցով հայտնաբերում է սինքրոազդանշանները և սինքրոտիզմի հատում կատարող ազդանշանները: Python լեզվի օբյեկտային կողմնորոշված լինելու շնորհիվ՝ ծրագրի ներսում սինքրոազդանշանները ներկայացվում են որպես ClockSignal դասի օբյեկտներ, որոնք ունեն հետևյալ հատկությունները՝ անուն, հաճախականություն և տվյալ սինքրոազդանշանով սինքրոնացվող ազդանշանների ցուցակ (նկ. 3.3): Վերջինս հետագայում ստեղծվում է ՎԲ-ի միջոցով՝ ՍՍՀԲ-ից ստացված սինքրոտիզմների վերլուծությամբ:

ՎԲ-ը ստեղծում է SynchronizedSignal դասին պատկանող օբյեկտներ, որոնցից նրանք, որոնք հանդիպում են երկու կամ ավելի տարբեր սինքրոտիզմներում, ստանում են հատուկ դրոշմային արժեք՝ CDCSignal, ինչը նշանակում է, որ տվյալ ազդանշանը կատարում է սինքրոտիզմի հատում:



Նկ. 3.2. ՍՍՀԲ-ի աշխատանքի ալգորիթմը

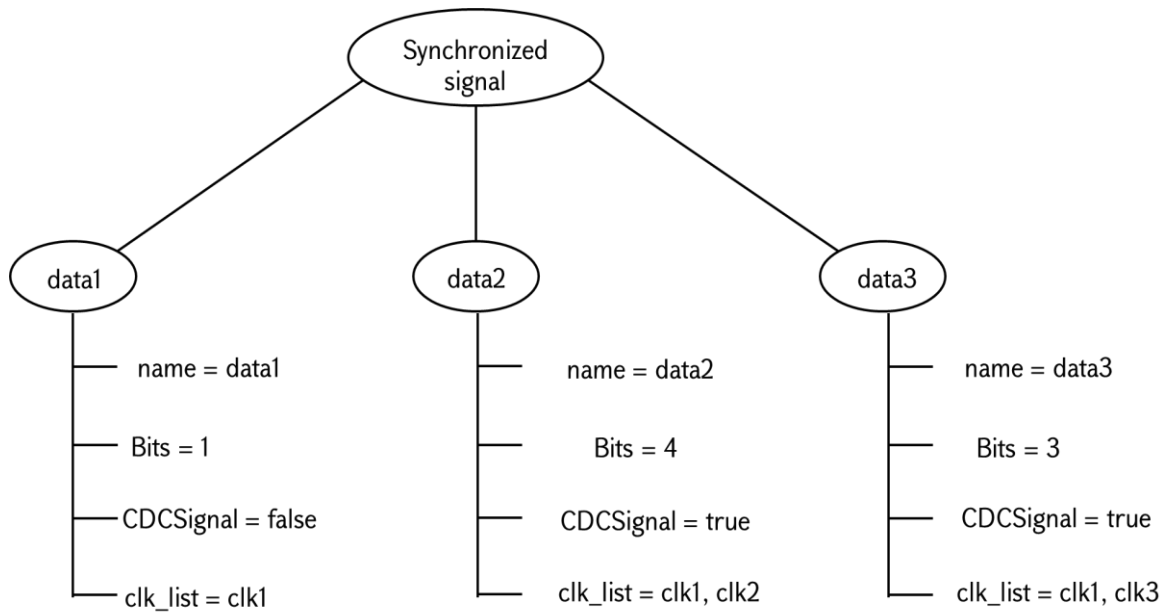


Նկ. 3.3. ClockSignal դասի օբյեկտները

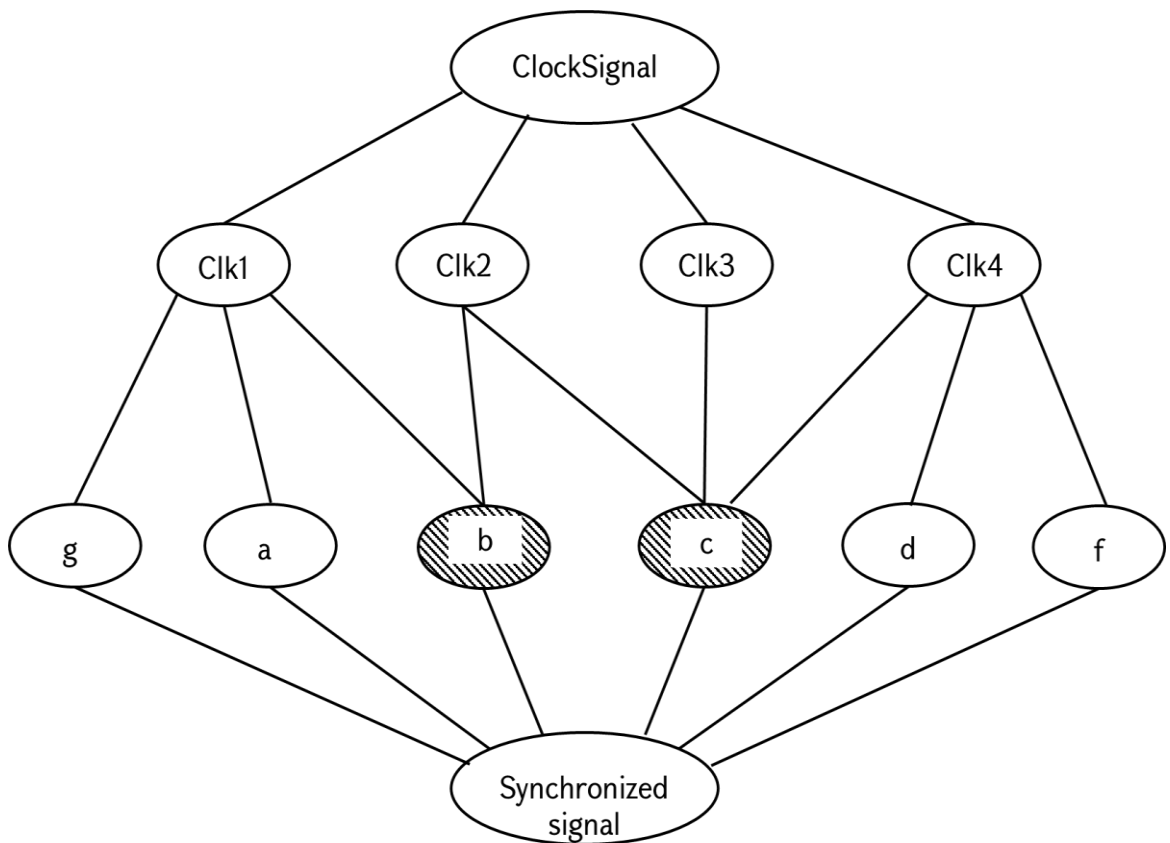
SynchronizedSignal դասի օբյեկտները (նկ. 3.4) ունեն հետևյալ հատկությունները՝ ազդանշանի անուն, կարգայնություն, CDCSignal դրոշմ և սինքրոտիզացիայի ցուցակ,

որոնց պատկանում է ազդանշանը: ClockSignal և SynchronizedSignal դասերի օբյեկտների փոխկապակցվածությունը (նկ. 3.5) ապահովում է համաձայնեցման սխեմաների ներդրման տեղի և անհրաժեշտ համապատասխանեցման միջոցի ընտրությունը:

ՀՄԸԲ-ը (նկ. 3.6), օգտվելով CDCSignal դրոշմն ունեցող SynchronizedSignal դասի օբյեկտների սինքրոազդանշանների հաճախականություններից, դրանց ավելացնում է ևս մի լրացուցիչ հատկություն, որը պարունակում է տվյալ ազդանշանը փոխանակող սինքրոտիրույթների միջև անհրաժեշտ համաձայնեցման միջոցների անունները: Համաձայնեցման միջոցների ընտրությունն իրականացվում է՝ ելնելով համապատասխան տակտավորող ազդանշանների հաճախականություններից: Եթե ուղարկող սինքրոտիրույթի սինքրոազդանշանի հաճախականությունը մեծ է ընդունող սինքրոտիրույթի սինքրոազդանշանից, ապա սինքրոտիրույթի հատում կատարող ազդանշանի միակարգ լինելու դեպքում դրա համաձայնեցման համար ընտրվում է մետակայությունության հայտնաբերմամբ և ազդանշանի վերականգնմամբ միջոցը, իսկ բազմակարգ ազդանշանների դեպքում՝ սինքրոազդանշանների համապատասխանեցում իրականացնող միջոցը: Այն դեպքում, երբ ՈԻՍ ՍԱ-ի հաճախականությունը փոքր է ԸՍ-ի հաճախականությունից, անկախ սինքրոտիրույթի հատում կատարող ազդանշանի կարգայնությունից՝ այն համաձայնեցվում է սինքրոազդանշանների համապատասխանեցում կատարող համաձայնեցման միջոցով: Երբ բոլոր ազդանշանների համաձայնեցման միջոցներն արդեն հայտնի են ակտիվացվում է ՀՄԸԲ-ը (նկ. 3.7): Սինքրոտիրույթի հատում կատարող ազդանշանին համապատասխանող SynchronizedSignal դասի օբյեկտի համապատասխան հատկության միջոցով համաձայնեցման միջոցների գրադարանից ընտրվում է նշված համաձայնեցման միջոցը և տեղադրվում հաղորդակցվող սինքրոտիրույթների միջև՝ ստանալով համապատասխան միացումները: SmartSync-ի կարևոր առանձնահատկություններից է այն, որ համաձայնեցման միջոցների գրադարանը կարող է ցանկացած պահի համալրվել նոր ՀՄ-երով՝ առանց առաջացնելու ծրագրային գործիքի վերակոմպիլացման անհրաժեշտություն:

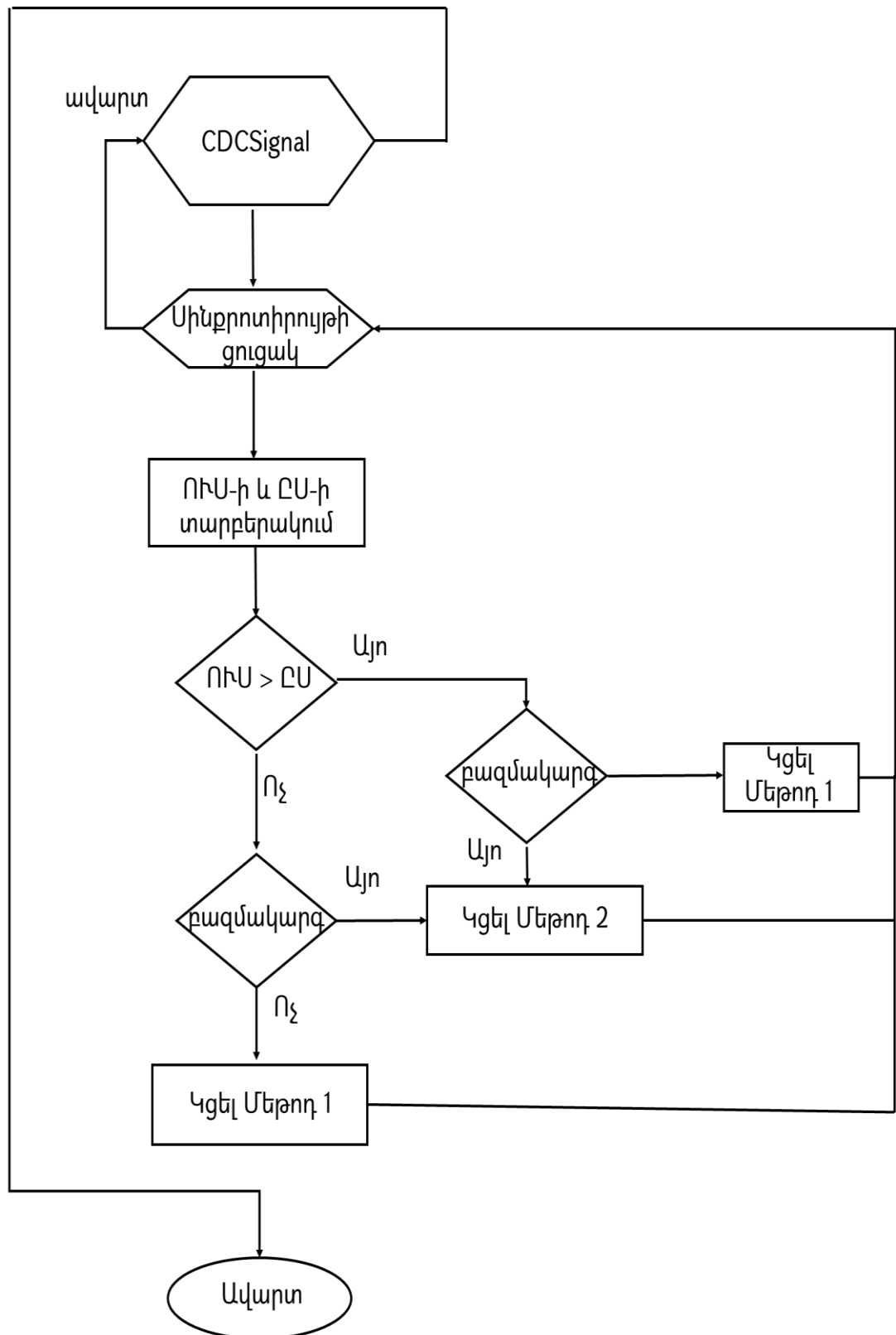


Նկ. 3.4. SynchronizedSignal դասի օբյեկտները

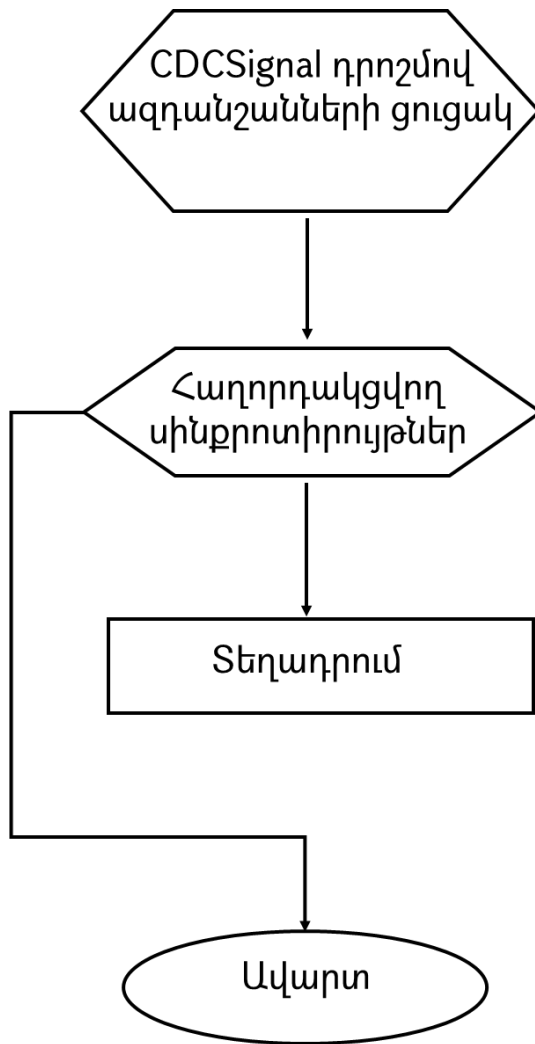


Նկ. 3.5. ClockSignal և SynchronizedSignal դասերի օբյեկտների
փոխկապակցվածությունը

Մշակված ծրագրային գործիքի մուտքային կոնֆիգուրացիոն ֆայլերն են՝ համաձայնեցման միջոցների կոնֆիգուրացիոն ֆայլը (<ՄԿՖ) և համակարգի սինքրոտիզմի կոնֆիգուրացիոն ֆայլը (<ՍԿՖ):



Նկ. 3.6. <ՄԸԲ-ի աշխատանքի սկզբունքը



Նկ. 3.7. ՀՄՆԲ-ի աշխատանքի սկզբունքը

ՀՄԿՖ ֆայլի տիպային օրինակը հետևյալն է.

NAME: MET_DETECTION_REPAIR

MIN_FREQ: 330

MAX_FREQ: 2000

DIRECTION: LOW_TO_HIGH

SYNC_METHOD: BITWISE

SYNC_DATA: BOTH

=====

NAME: FIFO_SYNC

MIN_FREQ: 330

MAX_FREQ: 2000

DIRECTION: HIGH_TO_LOW

SYNC_METHOD: MULTIBIT

SYNC_DATA: MULTIBIT

Այս ֆայլում յուրաքանչյուր ՀՄ նկարագրվում է NAME, MIN_FREQ, MAX_FREQ, DIRECTION, SYNC_METHOD, SYNC_DATA պարամետրերով որոնք համապատասխանաբար ներկայացնում են ՀՄ-ի անունը գրադարանում, նվազագույն համաձայնեցվող հաճախականությունը, առավելագույն համաձայնեցման հաճախականությունը, համաձայնեցման ուղղությունը, փոխանցվող ազդանշանի համաձայնեցման ձևը և կարգայնությունը: Վերոհիշյալ պարամետրերն ընդունում են հետևյալ թույլատրելի արժեքները.

- **NAME.** ընդունում է տառային սիմվոլով սկսվող ցանկացած տառաթվային կոմբինացիա:
- **MIN_FREQ.** ընդունում է 0-ից մեծ դրական թվեր, որոնք համապատասխանում են Մեգահերցային մասշտաբին:
- **MAX_FREQ.** ընդունում է 0-ից մեծ դրական թվեր, որոնք համապատասխանում են Մեգահերցային մասշտաբին:
- **DIRECTION.** ընդունում է LOW_TO_HIGH, HIGH_TO_LOW և BOTH արժեքներից որևէ մեկը, որոնք ցույց են տալիս, որ ՀՄ-ը կիրառելի է համապատասխանաբար, եթե ՈՒՍ-ի հաճախականությունը ավելի մեծ է քան ԸՍ-ինը, ԸՍ-ի հաճախականությունը ավելի մեծ է, քան ՈՒՍ-ինը և երկու դեպքերում էլ:
- **SYNC_METHOD.** ընդունում է SINGLEBIT, MULTIBIT և BOTH արժեքները, որոնք ցույց են տալիս, թե մեկ ՀՄ-ը քանի տվյալային կարգ է համաձայնեցնում:
- **SYNC_DATA.** ընդունում է SINGLEBIT, MULTIBIT և BOTH արժեքները, որոնք ցույց են տալիս համաձայնեցվող ազդանշանի հնարավոր կարգայնությունը:

ՀՄԿՖ-ի պարունակությունն օգտագործվում է ՀՄԸԲ-ի կողմից՝ անհրաժեշտ համաձայնեցման միջոցն ընտրելիս:

ՀՄԿՖ-ի՝ երկու սինքրոտիզացիոն պարունակող տիպային օրինակը հետևյալն է՝
CLKNAME: clk1

FREQ: 500

=====

CLKNAME: clk2

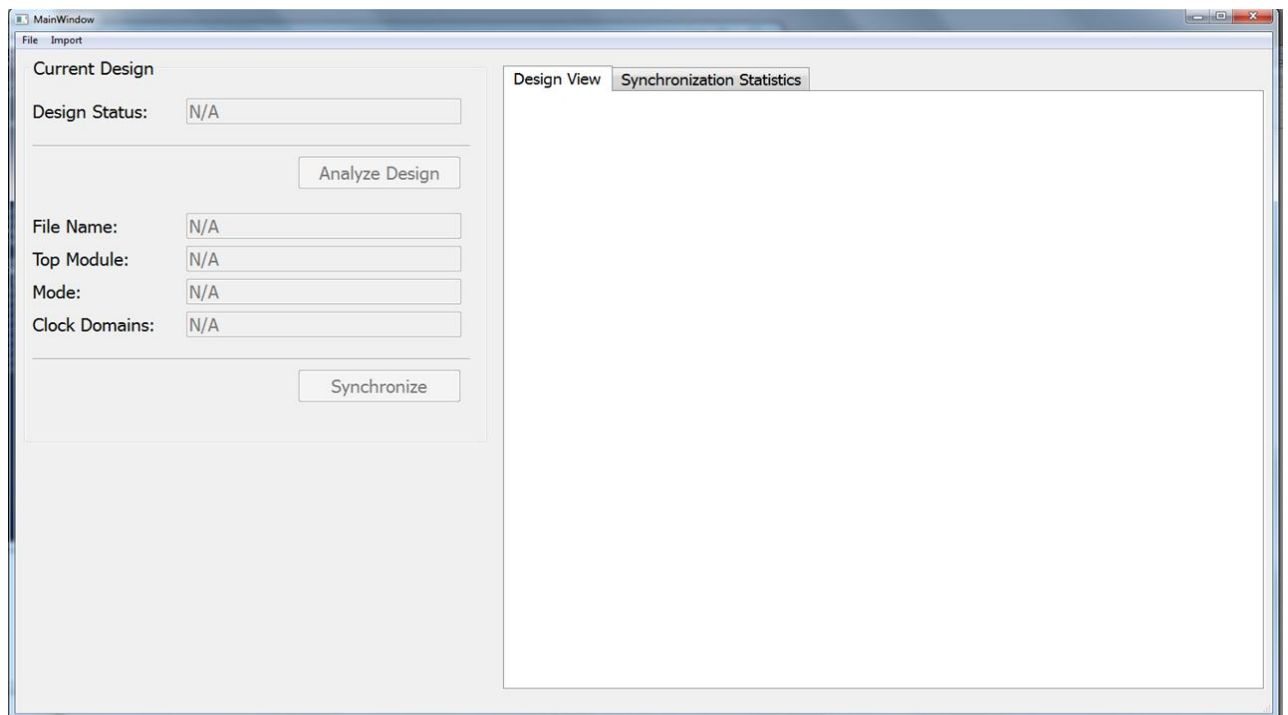
FREQ: 3500

Այս ֆայլում յուրաքանչյուր սինքրոտիզացիոն նկարագրվում է հետևյալ երկու պարամետրերով՝ CLKNAME և FREQ, որոնք ներկայացնում են նկարագրվող սինքրոտիզացիոն սինքրոազդանշանի անունը և հաճախականությունը՝ Մեգահերցերով արտահայտված:

3.2. Տարբեր հաճախություններով տակտավորվող թվային սխեմաների համաձայնեցման ծրագրային միջոցի կառուցվածքը

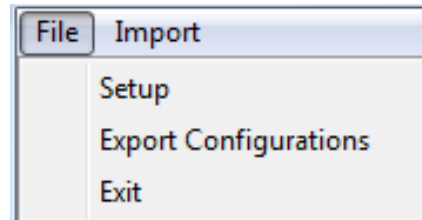
SmartSync ծրագրային գորիքային միջոցի պարզ և հարմարավետ ինտերֆեյսը թույլ է տալիս օգտագործողին նվազագույն ջանքերի միջոցով ծանոթանալ նախագծում առկա սինքրոտիզացիաներին, դրանց քանակին, և կատարել նաև սինքրոտիզացիաների համաձայնեցում:

Գործիքի միացման ժամանակ բացվում է դրա հիմնական պատուհանը (նկ. 3.8):

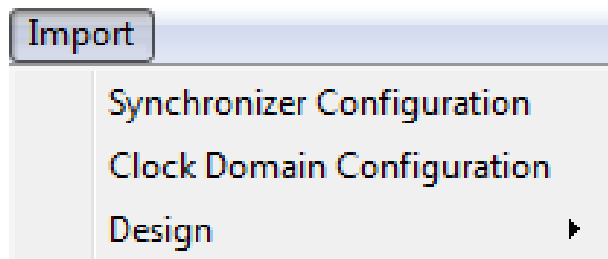


Նկ. 3.8. SmartSync ծրագրի գրաֆիկական ինտերֆեյսը
Այս պատուհանն ունի հետևյալ կառուցվածքը՝

- File և Import մենյուներ (նկ. 3.9 և նկ. 3.10),
- ընթացիկ նախագծի տվյալներ,
- նախագծի վերլուծություն սկսող Analyze Design կոճակ,
- նախագծի համաձայնեցում սկսող Synchronize կոճակ,
- Design View և Synchronization Statistics ենթաէջեր:



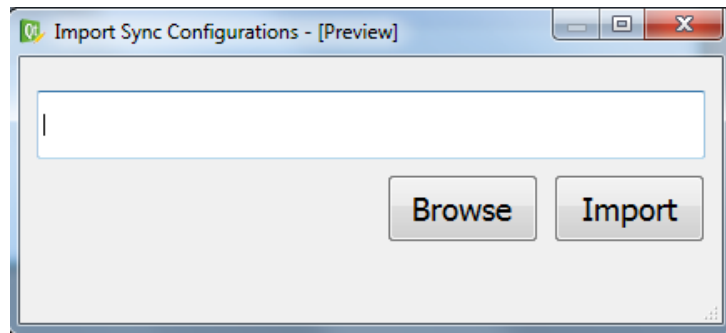
Նկ. 3.9. Ֆայլ մենյուն



Նկ. 3.10. Ֆայլերի ներմուծման մենյուն

File մենյուն բաղկացած է Setup, Export Configurations և Exit ընտրություններից, որոնք ներկայացնում են համապատասխանաբար, նախագծի վերլուծության և համաձայնեցման համար անհրաժեշտ կոնֆիգուրացիաների պատուհան, ներմուծված կոնֆիգուրացիաները ֆայլի մեջ գրանցելու հրահանգ և ծրագրից դուրս գալու հրահանգ:

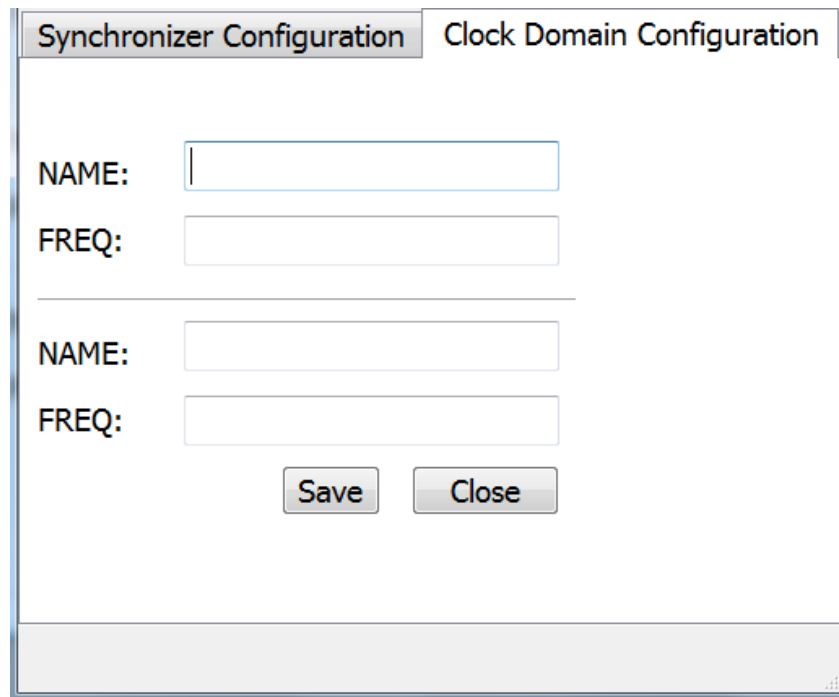
Import մենյուի Synchronizer Configuration և Clock Domain Configuration ընտրությունները սեղմելով բացվում են միանման պատուհաններ (նկ. 3.11), որոնցում հնարավոր է ներմուծել կամ նշել համապատասխան կոնֆիգուրացիոն ֆայլի անունը: Այդ ֆայլերը գրված են 3.1-ում նկարագրված ֆորմատներով, իսկ դրանց պարունակած ինֆորմացիան հիշվում է ծրագրի ներքին հիշողության մեջ:



Նկ. 3.11. Ֆայլի ներմուծման պատուհանը

Ծրագրային գործիքը հնարավորություն է տալիս կոնֆիգուրացիոն տվյալները ներմուծել ձեռքով (նկ. 3.12, 3.13)

Նկ. 3.12. Համաձայնեցման մեթոդների կոնֆիգուրացիաների պատուհանը

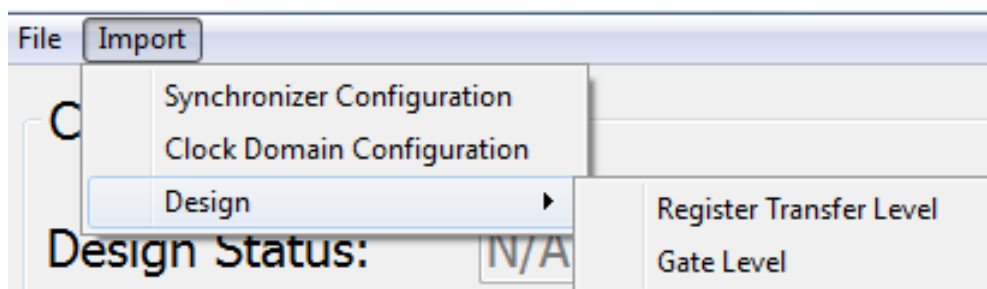


Նկ. 3.13. Սինքրոտիզացիայի կոնֆիգուրացիաների պատուհանը

Design ենթամենյուի ընտրության տարբերակների միջոցով կարելի է ներմուծել ռեգիստրային փոխանցման մակարդակի կամ փականային մակարդակի նկարագրություններ:

Մշակված ծրագրի աշխատանքի սկզբում նախագծի տվյալների առջև գրված N/A նշում է ցույց է տալիս, որ դեռևս չի ներմուծվել համաձայնեցման ենթակա նախագիծը:

SmartSync ծրագիրն աշխատեցնելու համար անհրաժեշտ է ներմուծել համաձայնեցման ենթակա ԻՍ-ի Verilog նկարագրությունը: Դա արվում է Import մենյուի Design ենթամենյուից RTL կամ Gate Level տարբերակներն ընտրելով (նկ. 3.14):



Նկ. 3.14. ՌԴՄ կամ ՓՄ նկարագրության ներմուծումը

Համապատասխան նկարագրությունն ընտրելուց հետո նախագծի կարգավիճակը ցույց տվող Design Status ցուցիչն ընդունում է Available արժեքը (նկ. 3.15), որը նշանակում է, որ նախագիծը պատրաստ է վերլուծության համար:

Նախագծի վերլուծությունը ենթադրում է նախագծում առկա սինքրոտիզացիաների և սինքրոազդանշանների հայտնաբերում և սինքրոտիզացիաների ու սինքրոազդանշանների հատում կատարող ազդանշանների արտապատկերում: Վերջինս հնարավորություն է տալիս նախագծողին, առանց համակարգի մանրամասները տեսնելու, ակնառու պատկերի միջոցով ուսումնասիրել համակարգի կառուցվածքը և փոխկապակցված սինքրոտիզացիաները:

Current Design

Design Status:

File Name:

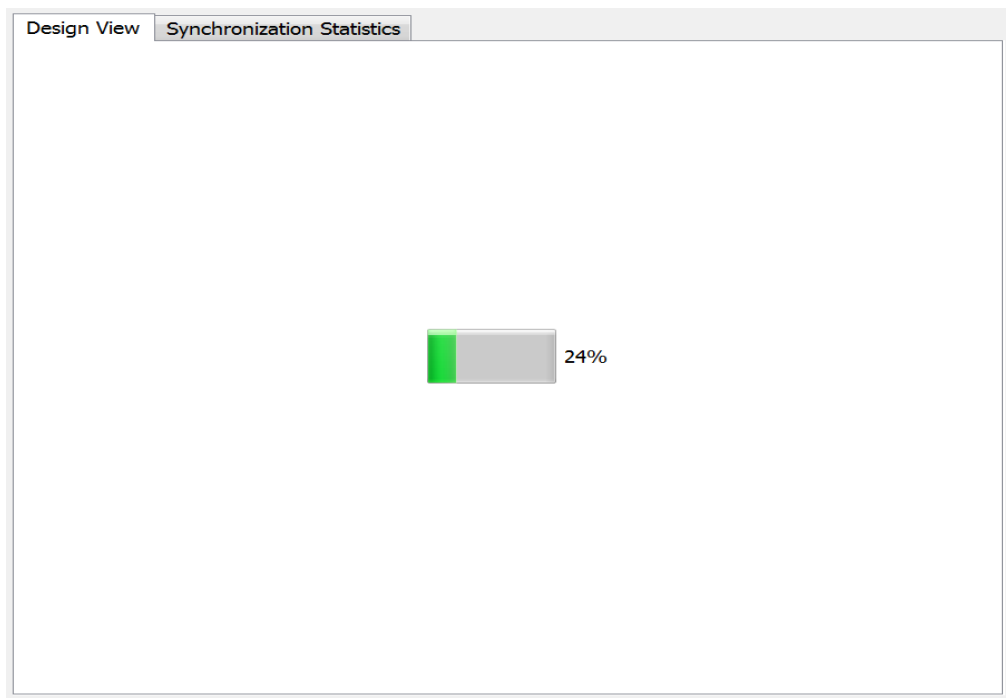
Top Module:

Mode:

Clock Domains:

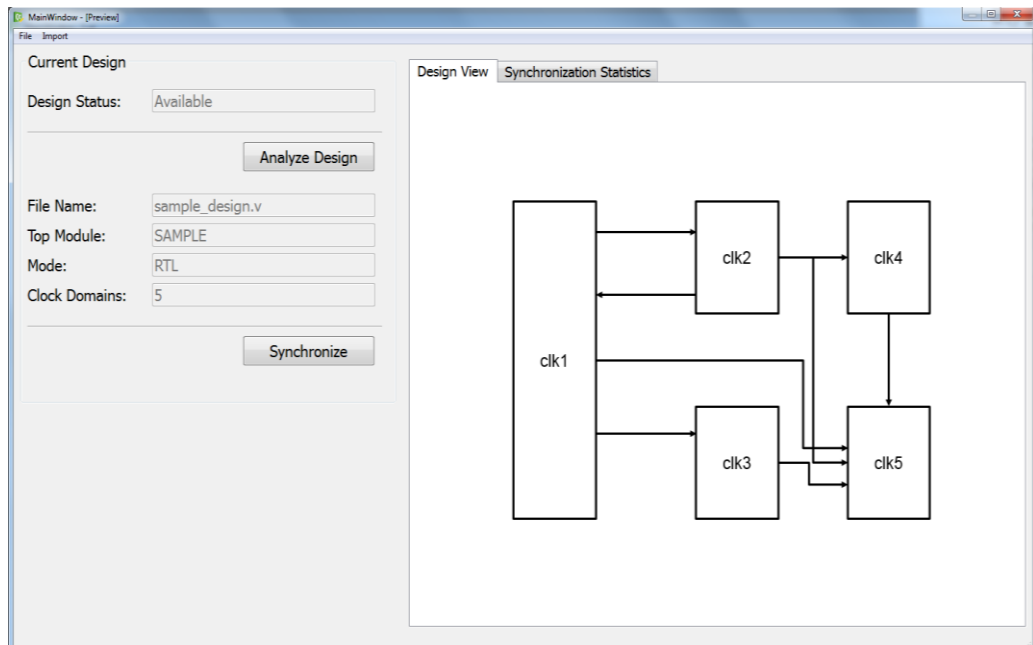
Նկ. 3.15. Նախագծի կարգավիճակի ցուցիչը

Analyze Design կոճակը սեղմելուց հետո Design View դաշտում հայտնվում է սխեմայի վերլուծության ընթացքի մասին վկայող ցուցիչը (նկ. 3.16):



Նկ. 3.16. Սխեմայի վերլուծության ընթացքի ցուցիչը

Վերլուծության գործընթացի ավարտից հետո ընթացիկ նախագծի տվյալային դաշտերն ընդունում են համապատասխան արժեքներ, իսկ Design View դաշտում հայտնվում է սինքրոտիզացիայի և դրանց փոխկապակցվածությունների սկզբունքային սխեման (նկ. 3.17):



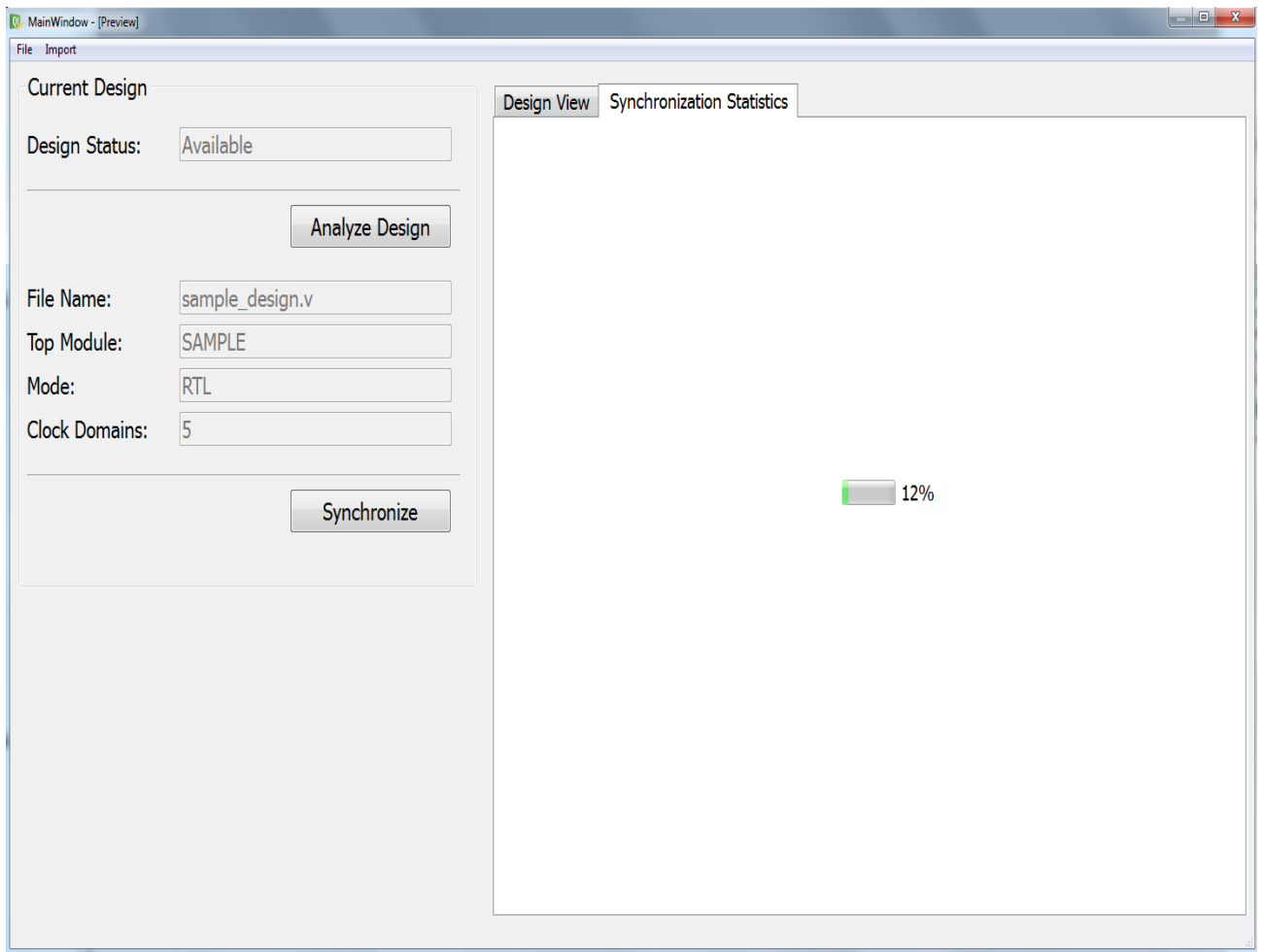
Նկ. 3.17. Համաձայնեցման միջոցների ներդրմանը պատրաստ նախագիծը

Վերոհիշյալ տվյալների հայտնվելը վկայում է այն մասին, որ նախագիծը պատրաստ է համաձայնեցման միջոցների ներմուծման համար:

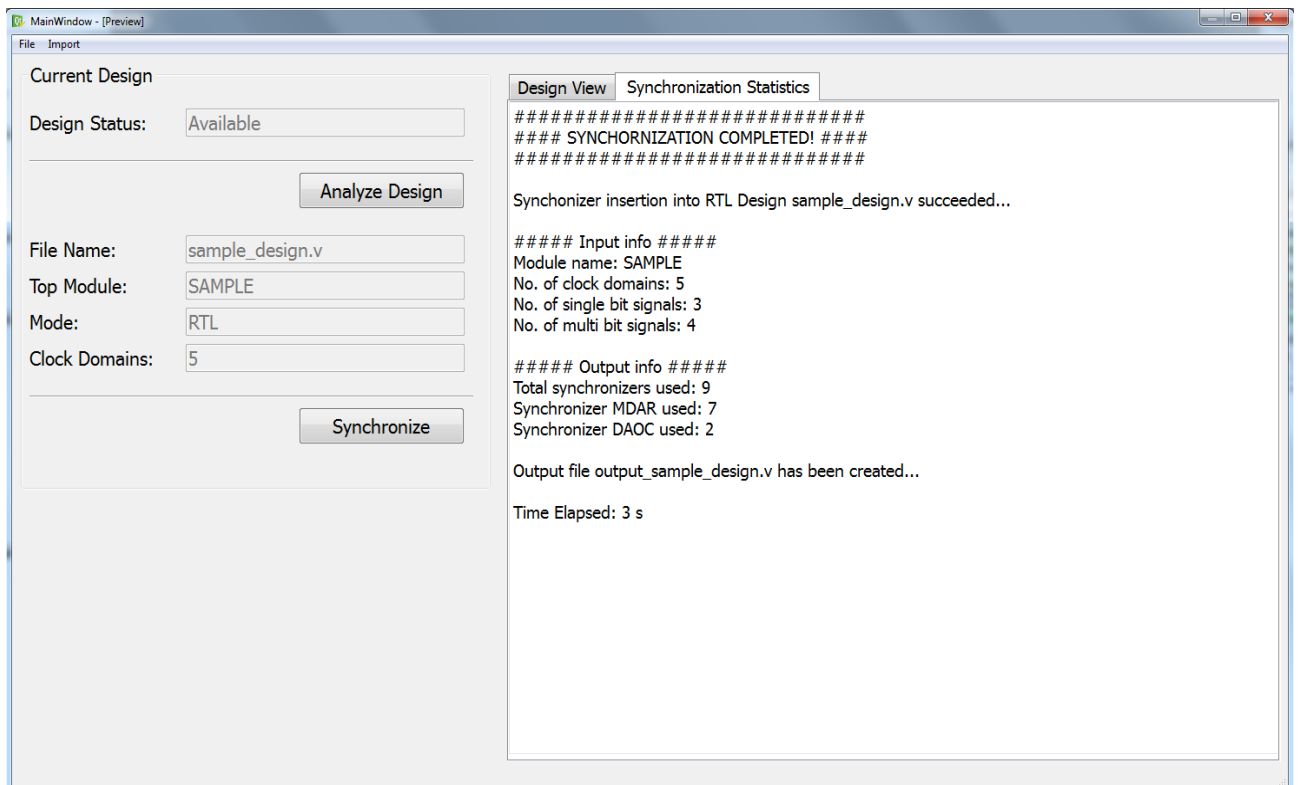
Համաձայնեցման միջոցների ներմուծումը սկսելու համար պետք է սեղմել Synchronize կոճակը, որի արդյունքում կատարվում է անցում Design View դաշտից Synchronization Statistics դաշտ, որի վրա կրկնում է նախագծի համաձայնեցման կարգավիճակը (նկ. 3.18):

Համաձայնեցման գործընթացի ավարտից հետո Synchronization Statistics պատուհանում հայտնվում է հետևյալ ինֆորմացիան (նկ. 3.19).

- ԻՍ-ի Verilog նկարագրությունը պարունակող նախաձևական ֆայլի անունը,
- վերլուծության արդյունք հանդիսացող տվյալները,
- կիրառված համաձայնեցման միջոցների ընդհանուր քանակը,
- ՀՄ-երի գրադրադարանում առկա յուրաքանչյուր մեթոդի կիրառությունների քանակը:



Նկ. 3.18. ՀՄ-երի ներդրման ընթացքի ցուցիչը



Նկ. 3.19. Համաձայնեցման ավարտից հետո SmartSync ծրագրի պատուհանը

SmartSync ծրագրային միջոցի ելքային արդյունքը համաձայնեցման միջոցներ պարունակող սխեմայի ՌՓՄ կամ ՓՄ նկարագրությունն է, որի օրինակը բերված է ստորև:

```
module rtl_uart (rst, sender_clk, en_send_d, d_send, send_en, sender_output,
sender_buf_empty, receiver_clk, en_receive_d, receiver_data, en_rec, get_in,
receive_buf_empty );
```

```
input rst, sender_clk, en_send_d;
```

```
input [3:0] d_send;
```

```
input send_en;
```

```
output sender_output, output sender_buf_empty;
```

```
input receiver_clk, en_receive_d;
```

```
output [3:0] receiver_data;
```

```
input en_rec, get_in;
```

```
output receive_buf_empty;
```

```
reg [3:0] reg_rec;
```

```
reg sender_buf_empty;
```

```
reg sender_output;
```

```
reg [3:0] receiver_data;
```

```
reg receive_buf_empty;
```

```
always @ (posedge receiver_clk or posedge rst)
```

```
if (rst) begin
```

```
    receiver_data    <= 0;
```

```
    receive_buf_empty    <= 1;
```

```
end else begin
```

```
    if (en_receive_d) begin
```



```

    receiver_data <= sender_reg;
    receive_buf_empty <= 1;
end
if (en_rec) begin
    sender_busy    <= 1;
end
// Start of frame detected, Proceed with rest of data
if (sender_block_busy) begin
    sender_block_sample_cnt <= sender_block_sample_cnt + 1;
    // Logic to sample at middle of data
    if (sender_block_sample_cnt == 7) begin
if ((sender_block_d2 == 1) && (sender_block_cnt == 0)) begin
    sender_block_busy <= 0;
end else begin
    sender_block_cnt <= sender_block_cnt + 1;
    // Start storing the sender_block data
    if (sender_block_cnt > 0 && sender_block_cnt < 9) begin
        sender_block_reg[sender_block_cnt - 1] <= sender_block_d2;
    end
    if (sender_block_cnt == 9) begin
        sender_block_busy <= 0;
        // Check if End of frame received correctly
        if (sender_block_d2 == 0) begin
            sender_block_frame_err <= 1;
        end else begin
            receive_buf_empty    <= 0;
            sender_block_frame_err <= 0;
            // Check if last sender_block data was not unloaded,
            sender_block_over_run <= (receive_buf_empty) ? 0 : 1;

```

```

        end
    end
end

    end

    end
end
if (!en_rec) begin
    sender_block_busy <= 0;
end
end
always @ (posedge sender_clk or posedge rst)
if (rst) begin
    reg_rec      <= 0;
    sender_buf_empty    <= 1;
    receiver_block_over_run  <= 0;
    sender_output    <= 1;
    receiver_block_cnt    <= 0;
end else begin
    if (en_send_d) begin
        if (!sender_buf_empty) begin
            receiver_block_over_run <= 0;
        end else begin
            reg_rec <= d_send;
            sender_buf_empty <= 0;
        end
    end
end
if (send_en && !sender_buf_empty) begin
    receiver_block_cnt <= receiver_block_cnt + 1;
    if (receiver_block_cnt == 0) begin

```

```

    sender_output <= 0;
end
if (receiver_block_cnt > 0 && receiver_block_cnt < 9) begin
    sender_output <= reg_rec[receiver_block_cnt -1];
end
if (receiver_block_cnt == 9) begin
    sender_output <= 1;
    receiver_block_cnt <= 0;
    sender_buf_empty <= 1;
end
end
if (!send_en) begin
    receiver_block_cnt <= 0;
end
end
MDAR x1(sender_output, receiver input, clk);
endmodule

```

Վերևում գրված Verilog նկարագրության մեջ առկա է երկու սինքրոտիզացիա, որոնք փոխանակում են մեկ 8-կարգանի ազդանշան: Սինքրոտիզացիա հատում կատարող այդ ազդանշանները համաձայնեցվում են MDAR համաձայնեցման միջոցի գործադրմամբ, որին տրված են համապատասխան միացումները:

3.3. Տարբեր հաճախություններով տակտավորվող թվային սխեմաների համաձայնեցման միջոցների արդյունավետության գնահատումը

Առաջարկված SmartSync ծրագրային միջոցի արդյունավետությունը ստուգվել է հետևյալ բազմասինքրոն թվային համակարգերի վրա՝ OpenSPARC T1, OpenSPARC T2 և ORCA պրոցեսորներ:

Վերոհիշյալ երեք պրոցեսորները ներկայացնում են անվճար լիցենզավորմամբ պրոցեսորներ, որոնք, սինքրոտիզացիաների բաժանված լինելու շնորհիվ, նպատակահարմար է օգտագործել SmartSync ծրագրի թեստավորման ժամանակ:

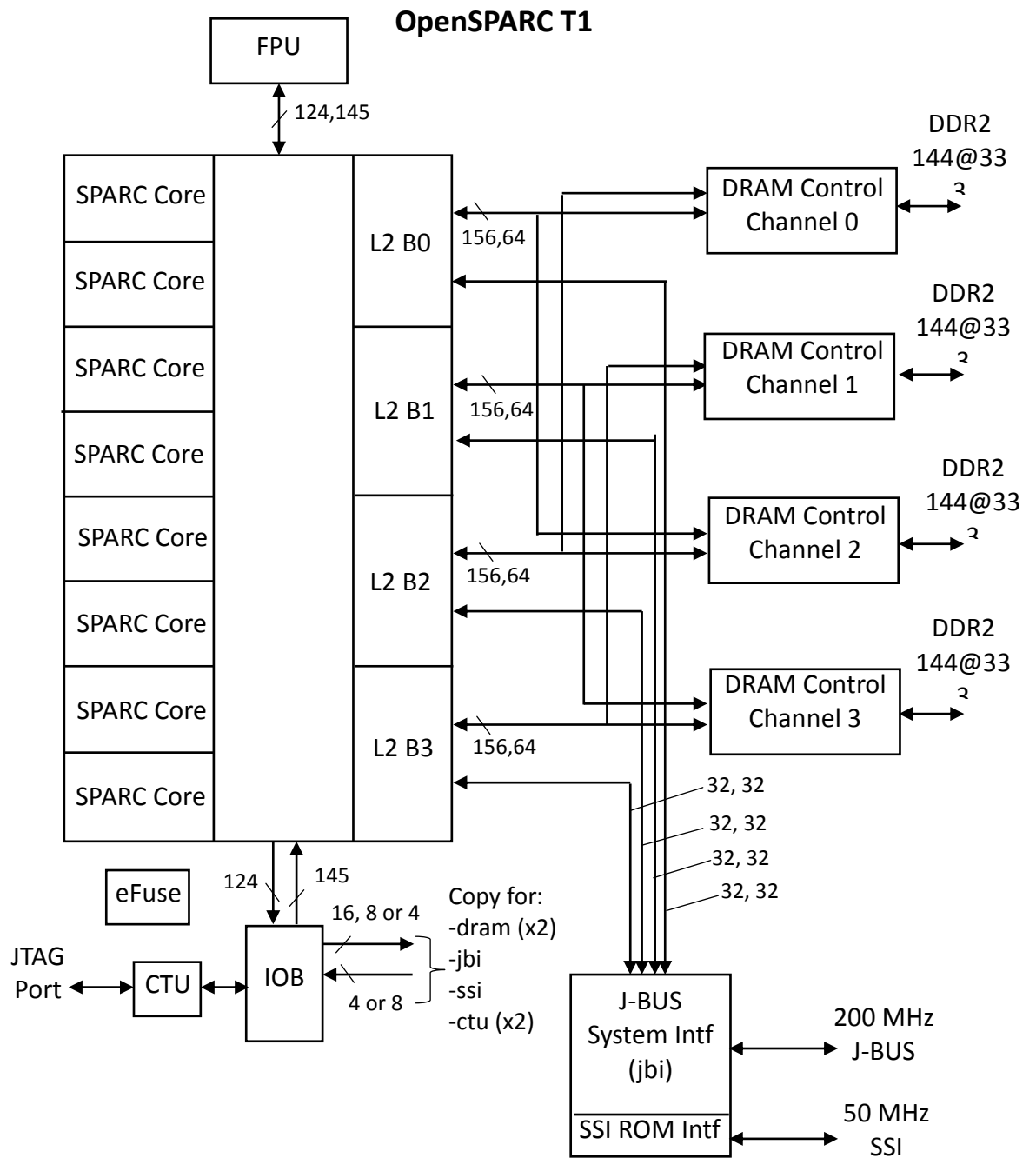
Մասնավորապես, OpenSPARC T1 պրոցեսորը (նկ. 3.20) 64 կարգայնությամբ բազմամիջուկ համակարգ է: Այն բաղկացած է 8 միջուկներից:

Միջուկներից յուրաքանչյուրը ներկայացնում է SPARC միջուկ, որն ունի 4 զուգահեռ ուղիներ սպասարկող սարքային ապահովում: Այդ ապահովումը բաղկացած է լիարժեք ռեգիստրային ֆայլից, օժանդակող վիճակների ռեգիստրներից և արտոնյալ ռեգիստրներից: SPARC միջուկներից յուրաքանչյուրը բաղկացած է 6-կասկադանի կոնվերտներից (նկ. 3.21): Այդ կասկադներն են.

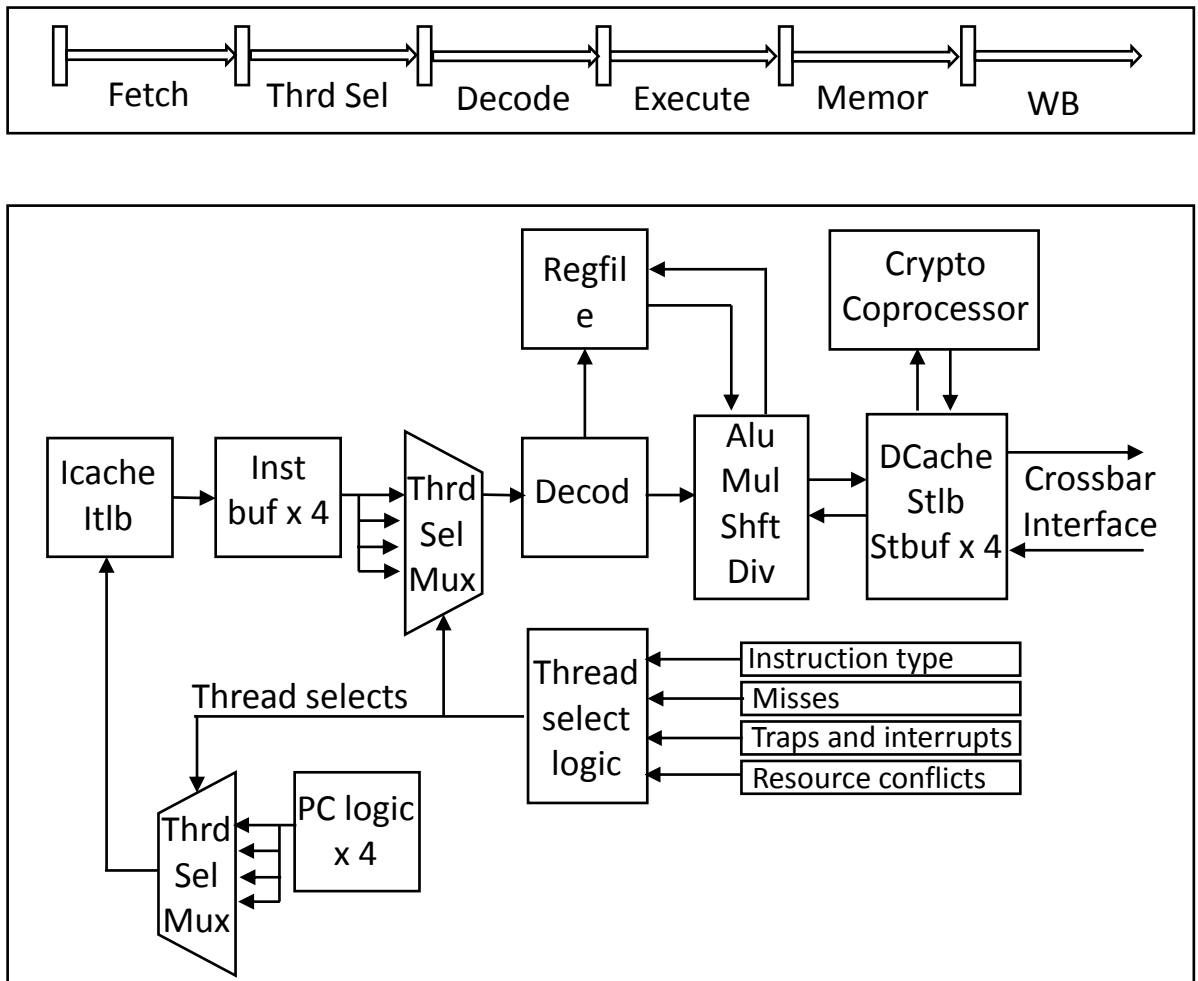
- վերծանում,
- ուղու ընտրում,
- դեկոդավորում,
- իրականացում,
- հիշողություն,
- հետադարձ գրանցում:

Միջուկների կազմի մեջ մտնում են հետևյալ միավորները.

- Հրահանգի վերծանման միավորը (ՀՎՄ), որը ներառում է վերծանման, ուղու ընտրման և դեկոդավորման կասկադները:
- Իրականացման միավորը (ԻՄ), որը ներառում է իրականացման կասկադը:
- Բեռնում/տեղադրման միավորը (ԲՏՄ), որում ընդգրկված են հիշողության և հետադարձ գրանցման կասկադները:
- Ընդհատումների տրամաբանական միավորը (ԸՏՄ), որի կազմում են ընդհատման տրամաբանությունը և ընդհատման հաշվիչը:
- Հիշողության ղեկավարման միավորը (ՀՂՄ):



Նկ. 3.20. OpenSPARC T1 պրոցեսորի կառուցվածքը



Նկ. 3.21. SPARC միջուկի կառուցվածքը

Նշված համակարգերի ՌՓՄ նկարագրությունների սինթեզն ու փականային մակարդակների մոդելավորումներն իրականացվել են 28 նմ տեխնոլոգիայի համար, տիպային լարման 1.05 Վ արժեքի դեպքում: Արդյուքների ստացումն ու դրանց արդյունավետության գնահատումն իրականացվել են հետևյալ կերպ.

1. SmartSync ծրագրի միջոցով իրականացվել է վերհիշյալ 3 սխեմաների ՌՓՄ նկարագրություններում սինթրոտիրույթների համաձայնեցում՝ առաջարկված երկու մեթոդներով և հերթահենք համաձայնեցման մեթոդով:
2. Ծրագրի ելքային Verilog նկարագրությունները այնուհետև սինթեզվել են Design Compiler ծրագրային գործիքային միջոցի գործադրմամբ:
3. Արձանագրվել և համեմատվել են համաձայնեցման միջոցներով սինթեզված սխեմայի փականների քանակը և սինթրոտիրույթների միջև առավելագույն հապաղումը:

Նկարագրված ալգորիթմով և SmartSync ծրագրով իրականացված համաձայնեցման արդյունավետությունը գնահատելու համար VCS ծրագրային գործիքի միջոցով յուրաքանչյուր սխեմայի համար իրականացվել է փականային մակարդակի մոդելավորում:

Աղ. 3.1, 3.2-ում ցուցադրված արդյունքները վկայում են SmartSync ծրագրի և մշակված համաձայնեցման մեթոդների արդյունավետության մասին:

Աղյուսակ 3.1

Համաձայնեցման տարբեր մեթոդների կիրառման դեպքում սխեմայում փականների ընդհանուր քանակը

Նախագիծ	Հերթահեն մեթոդ	Մեթոդ 1	Մեթոդ 2	Մեթոդ 1- հերթահեն	Մեթոդ 2- հերթահեն
OpenSPARC T1	48867	52350	50030	7,1%	2,38 %
OpenSPARC T2	50850	54785	51427	7,74 %	1,13 %
ORCA	63150	67830	65490	7,4 %	3,7 %

Աղյուսակ 3.2

Համաձայնեցման տարբեր մեթոդների կիրառման դեպքում սինքրոտիզմների միջև առավելագույն հապաղումները

Նախագիծ	Հերթահեն մեթոդ(նվ)	Մեթոդ 1 (նվ)	Մեթոդ 2 (նվ)	Մեթոդ 1- հերթահեն	Մեթոդ 2- հերթահեն
OpenSPARC T1	1,73	0,45	0,34	73 %	80 %
OpenSPARC T2	1,14	0,3	0,22	74 %	80,7 %
ORCA	0,8	0.24	0.15	70 %	81 %

Սխեմայում փականների քանակի տեսանկյունից առաջարկված մեթոդները հերթահեն մեթոդի համեմատ օգտագործում են 1,13% - 7,74 % ավելի շատ փականներ, իսկ սինքրոտիզմների միջև վատագույն հապաղումներում՝ 70% - 81 % շահում: Այս արդյունքները խոսում են համաձայնեցման առաջարկված մեթոդների և SmartSync ծրագրային գործիքով դրանց ներդրման արդյունավետության մասին:

Մշակված համաձայնեցման մեթոդները, հայտնի մյուս մեթոդների համեմատ, մակերեսի ոչ էական մեծացման հաշվին ԻՍ են ներմուծում փոքր հապաղում և բավարարում են ժամանակակից համակարգերին առաջադրվող արագագործության և մակերեսի պահանջներին:

Եզրակացություններ

1. Մշակված սկզբունքների և մեթոդների հիման վրա կառուցված համաձայնեցման միջոցների՝ SmartSync ծրագրային գործիքի միջոցով ներդրումը, վերջինիս պարզ և հարմարավետ ինտերֆեյսի և ելքային Verilog նկարագրության մոդելավորման շնորհիվ, ապահովում է նախագծման ժամանակի էական կրճատում:
2. OpenSPARC T1, OpenSPARC T2 և ORCA պրոցեսորներում մետակայունության հայտնաբերմամբ և բնականոն վիճակի վերականգնմամբ մեթոդով համաձայնեցման իրականացումը SmartSync ծրագրային գործիքով հերթահենք համաձայնեցման մեթոդի իրականացման համեմատ օգտագործված փականների քանակի 7,1 - 7,74% ավելացման հաշվին ապահովում է համաձայնեցման հապաղման 70 – 74% նվազեցում:
3. Սինքրոազդանշանների համապատասխանեցման մեթոդով պրոցեսորներում սինքրոտիրոյթների համաձայնեցման դեպքում կիրառվում են 1,13 – 3,7 % ավելի շատ փականներ քան համաձայնեցման հերթահենք մեթոդի դեպքում, սակայն առաջարկվող մեթոդն ապահովում է 80 – 81 %-ով փոքր հապաղում:

ԵԶՐԱՀԱՆԳՈՒՄ

1. Ձևակերպվել են տարբեր հաճախություններով տակտավորվող թվային սխեմաների սինքրոտիրոյթների համաձայնեցման միջոցներին առաջադրվող պահանջները, որոնց բավարարումն ապահովում է ինտեգրալ սխեմաների անհրաժեշտ արագագործություն ու կիսահաղորդչային բյուրեղի վրա զբաղեցրած մակերես:
2. Առաջարկվել է մետակայուն վիճակի հայտնաբերմամբ և բնականոն վիճակի վերականգնմամբ համաձայնեցման մեթոդ, որի դեպքում ազդանշանի վերաթողարկման շնորհիվ՝ ապահովվում է համաձայնեցման միջոցի հապաղման առնվազն 2 անգամ շահում, օգտագործված տրամաբանական փականների քանակի՝ 8 %-ը չգերազանցող ավելացման հաշվին:
3. Մշակվել է սինքրոագդանշանի դինամիկ համապատասխանեցմամբ սինքրոտիրոյթների համաձայնեցման մեթոդը, որն օգտագործված տրամաբանական փականների քանակի մեծացման և իրականացման որոշակի բարդության հաշվին ապահովում է ընդունող սինքրոագդանշանի պարբերությունը չգերազանցող համաձայնեցման հապաղում:
4. Առաջարկվել է համաձայնեցման միջոցների նախագծման գործընթաց, որն ապահովում է վերջիններիս՝ գործընթացի, լարման և ջերմաստիճանի նկատմամբ զգայունության էական նվազեցում և համաձայնեցված համակարգի կայունություն:
5. Ատենախոսությունում մշակված սկզբունքների և մեթոդների հիման վրա կառուցված համաձայնեցման միջոցների՝ SmartSync ծրագրային գործիքի միջոցով ներդնումը, վերջինիս պարզ և հարմարավետ ինտերֆեյսի շնորհիվ ապահովում է նախագծման ժամանակի շոշափելի կրճատում՝ այն հասցնելով մի քանի վայրկյանից մինչև մի քանի րոպե տևողության: OpenSPARC T1, OpenSPARC T2 և ORCA պրոցեսորներում սինքրոտիրոյթների համաձայնեցումն իրականացվում է առաջարկված մետակայունության հայտնաբերմամբ ու բնականոն վիճակի վերականգնմամբ և սինքրոագդանշանների դինամիկ

համապատասխանեցմամբ մեթոդներով ու հայտնի հերթահենք միջոցով: SmartSync ծրագրային գործիքով առաջարկված առաջին մեթոդով համաձայնեցման կառուցվածքների ներդնումը, հերթահենք համաձայնեցման մեթոդի իրականացման համեմատ, օգտագործված փականների քանակի 7,1 - 7,74 % ավելացման հաշվին ապահովում է համաձայնեցման հապաղման 70 - 74% նվազեցում, իսկ սինքրոազդանշանների համապատասխանեցման մեթոդով համաձայնեցման դեպքում, հերթահենք մեթոդի համեմատ 1,13 - 3,7 % ավելի շատ փականներ օգտագործելու պարագայում ապահովվում է 80 - 81 %-ով փոքր համաձայնեցման հապաղում:

ՕԳՏԱԳՈՐԾՎԱԾ ԳՐԱԿԱՆՈՒԹՅԱՆ ՑԱՆԿ

1. Taur Y., Ning T.H. Fundamentals of modern VLSI devices.- Cambridge university press. 2013. -678p
2. Emre S., Friedman E. High Performance Integrated Circuit Design.- McGraw Hill Professional, 2012. -736p
3. Niraj K., Chen D. Nanoelectronic circuit design. - Springer Science & Business Media, 2010. -485p
4. Hoefflinger B. Chips 2020: A Guide to the Future of Nanoelectronics. – Berlin: Springer, New York, 2012. –505p.
5. Fujita S. Nano-electronics challenge: chip designers meet real nano-electronics in 2010s? // Design, Automation and Test in Europe Conference. -2009. -P.431-432.
6. Rupp K., Selberherr S. The Economic Limit to Moore's Law // IEEE transactions on semiconductor manufacturing. - February, 2011. - Vol. 24, No 1. - P. 1-4.
7. Ayala J., Alonso A. VLSI-SoC: Forward-Looking Trends in IC and Systems Design // 18th IFIP WG 10.5/IEEE International Conference on Very Large Scale Integration. – New York: Springer, 2012. – P.27-29.
8. Rabaey J.M. Digital Integrated Circuits. 2nd edition. -Prentice-Hall, 2003. –761p.
9. Baker J. CMOS: Circuit Design, Layout and Simulation. 3rd edition. –Wiley-IEEE Press, 2010. –1208p.
10. Sze S., Kwok K. Physics of Semiconductor Devices. -New Jersey: John Wiley and Sons, 2006. –832p.
11. Kebaili M., Brignone J.C., Morin-Allory K. Clock domain crossing formal verification: a meta-model. // IEEE High Level Design Validation and Test Workshop (HLDVT). -2016. - P.136-141.
12. Spars J., Furber S. Principles Asynchronous Circuit Design. Kluwer Academic Publishers. - Kluwer Academic Publisher, 2002. -337p.
13. Myers C.J. Asynchronous circuit design. -John Wiley & Sons, 2004. -424p.
14. Birtwistle G., Davis A., editors. Asynchronous digital circuit design. -Springer Science & Business Media, 2013. -265p.
15. Wang X., Lu J. Analysis and improvement against the attack on 4-way handshaking protocol of 802.11 i // Jisuanji Gongcheng/ Computer Engineering. -2007. -Vol.32, No.3. - P.169-171.
16. He C., Mitchell J.C. Analysis of the 802.11 i 4-Way Handshake. // 3rd ACM workshop on Wireless security, 2004. -P.43-50.
17. Ginosar R. Fourteen ways to fool your synchronizer. // IEEE Ninth International Symposium Asynchronous Circuits and Systems, 2003. -P.89-96.

18. Wang Z., He X., Sechen C. A new approach for gate-level delay-insensitive asynchronous logic. // Circuits, Systems, and Signal Processing. -2015 May 1. -Vol.34 No.5. -P.1431-1459.
19. Caohuu T., Edwards J. Implementation of an Efficient Library for Asynchronous Circuit Design with Synopsys. // Progress in Systems Engineering. Springer International Publishing. -2015. -P.465-471.
20. Sokolov D., Mokhov A., Yakovlev A., Lloyd D. Towards asynchronous power management. // Faible Tension Faible Consommation (FTFC). -2014 IEEE. -P.1-4.
21. Di J., Smith S.C. // Ultra-low power multi-threshold asynchronous circuit design. United States patent US 8,664,977. 2014 Mar 4.
22. Cortadella J., Singhal V., Tuncer E., Lavagno L. // Variability-aware scheme for high-performance asynchronous circuit voltage regulation. United States patent US 8,572,539. 2013 Oct 29.
23. Shih C.H., Lai Y.H., Jiang J.H. SPOCK: Static performance analysis and deadlock verification for efficient asynchronous circuit synthesis. // Proceedings of the IEEE/ACM International Conference on Computer-Aided Design. -2015. -P.442-449.
24. Liu S., Ma Y., Hong X., Wang Y. Simultaneous slack budgeting and retiming for synchronous circuits optimization. // Design Automation Conference (ASP-DAC) 15th Asia and South Pacific. -2010. -P.49-54.
25. Krstic M., Grass E., Gürkaynak F.K., Vivet P. Globally asynchronous, locally synchronous circuits: Overview and outlook. // IEEE Design & Test of Computers. -2007. -Vol.24, No.5. -P.430-41.
26. Mengali U. Synchronization techniques for digital receivers. -Springer Science & Business Media, 2013. -520p.
27. Singh D.B., Date A.S., Sabnis H.S., inventors; Lsi Corporation, assignee. Methods and structure for source synchronous circuit in a system synchronous platform. United States patent US 8,497,704. 2013 Jul 30.
28. Scheffer L., Lavagno L., Martin G., editors. EDA for IC system design, verification, and testing. -CRC press, 2016. -544p.
29. Foster H.D. Trends in functional verification: a 2014 industry study. // Proceedings of the 52nd Annual Design Automation Conference. -2015. -p.48.
30. Tu Y.H., Cheng K.H., Hsu C.H., Huang H.Y. A low supply voltage synchronous mirror delay with quadrature phase output. // Design and Diagnostics of Electronic Circuits & Systems, 17th International Symposium. -2014. P.163-166.
31. Piguet C. Low-power electronics design. – CRC Press, 2005. –841p.
32. Rabaey J. Low Power Design Essentials. – Berlin: Springer, 2009. –288p.

33. Меликян В.Ш., Матевосян А.В., Чобанян С.Г., Петросян Т.А. Комбинированный способ снижения потребляемой мощности цифровых КМОП схем // Тематический выпуск XXVII Международной научно-технической конференции “Проблемы электроники”. - Киев, 2007. - Часть 1. - С. 89-92.
34. Sulistyo J.B., Ha D.S. A new characterization method for delay and power dissipation of standard library cells. // VLSI design. -2002. -Vol.15, No.3. -P.667-678.
35. Sharma V.K., Pattanaik M., Raj B. PVT variations aware low leakage INDEP approach for nanoscale CMOS circuits. // Microelectronics reliability. -2014. -Vol.54, No.1. -P.90-99.
36. Kumar R., inventor; Apple Inc., assignee. Power source for clock distribution network. United States patent US 9,419,589. 2016 Aug 16.
37. Noh K.L., Chaudhari Q.M., Serpedin E., Suter B.W. Novel clock phase offset and skew estimation using two-way timing message exchanges for wireless sensor networks. // IEEE transactions on communications. -2007. -Vol.55, No.4. -P.766-777.
38. Roy K., Prasad S. Low-Power CMOS VLSI Circuit Design. - // New York: Wiley Inerscience Publication, 2000.- 247p.
39. Tsai Y-F., Duarte D., Vijaykrishnan N., Irwin M. J. Impact of process scaling on the efficiency of leakage reduction schemes // International Conference on Integrated Circuit Design and Technology. - Texas, 2004. - P. 3-11.
40. Մելիքյան Վ.Շ., Հարությունյան Ա.Գ., Մաթևոսյան Ա.Վ., Պետրոսյան Յ.Պ. Ցածր էներգասպառմամբ ինտեգրված խմբական լախագծում, ՀԱՃՀ, Երևան, 2012. - 203 էջ :
41. Le Sueur E. , Heiser G. Dynamic voltage and frequency scaling: The laws of diminishing returns. // Proceedings of the 2010 international conference on Power aware computing and systems. -2010. -P.1-8.
42. Keating M., Flynn D., Aitken R. Low power methodology manual. – Berlin: Springer, 2007. –300p.
43. Henzler S. Power Management of Digital Circuits in Deep Sub-Micron CMOS Technologies. – Netherlands: Springer, 2009. –204p.
44. Choi K. , Soma R. , Pedram M. Dynamic voltage and frequency scaling based on workload decomposition. // Proceedings of the 2004 international symposium on Low power electronics and design. -2004. -P.174-179.
45. Ren Z. , Krogh B. H. , Marculescu R. Hierarchical adaptive dynamic power management. // IEEE Transactions on Computers. -2005. -Vol.54, No.4. -P.409-420.
46. Krstic M. , Grass E. , Gürkaynak F. K. , Vivet P. Globally asynchronous, locally synchronous circuits: Overview and outlook. // IEEE Design & Test of Computers. -2007. - Vol.24, No.5. -P.430-441.

47. Muttersbach J. , Villiger T. , Fichtner W. Practical design of globally-asynchronous locally-synchronous systems. // Advanced Research in Asynchronous Circuits and Systems, 2000.(ASYNC 2000) Proceedings. Sixth International Symposium. -2000. -P.52-59.
48. Chong K. S. , Chang K. L. , Gwee B. H. , Chang JS. Synchronous-logic and globally-asynchronous-locally-synchronous (GALS) acoustic digital signal processors. // IEEE Journal of Solid-state circuits. -2012. -Vol.47, No.3. -P.769-780.
49. Salman E. , Dasdan A. , Taraporevala F. , Kucukcakar K. , Friedman E. G. Exploiting setup–hold-time interdependence in static timing analysis. // IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. -2007. -Vol.26, No.6. -P.1114-1125.
50. Nishi Y., Doering R. Handbook of Semiconductor Manufacturing Technology. // 2nd edition. - CRC Press, 2007. –1720p.
51. Ortmanns M. , Gerfers F. , Manoli Y. A continuous-time/spl Sigma//spl Delta/Modulator with reduced sensitivity to clock jitter through SCR feedback. // IEEE Transactions on Circuits and Systems I: Regular Papers. -2005. -Vol.52, No.5. -P.875-884.
52. Ginosar R. Metastability and synchronizers: A tutorial. // IEEE Design & Test of Computers. -2011. -Vol.28, No.5. -P.23-35.
53. Lin J. F. Low-power pulse-triggered flip-flop design based on a signal feed-through. // IEEE transactions on very large scale integration (vlsi) systems. -2014. -Vol.22, No.1. - P.181-185.
54. Cantoni A. , Walker J. , Tomlin TD. Characterization of a flip-flop metastability measurement method. // IEEE Transactions on Circuits and Systems I: Regular Papers. - 2007. -Vol.54, No.5. -P.1032-1040.
55. Golson S. Synchronization and Metastability. // Synopsys Users Group (SNUG) Silicon Valley. 2014. -48p.
56. Beer S. , Cannizzaro M. , Cortadella J. , Ginosar R. , Lavagno L. Metastability in better-than-worst-case designs. // Asynchronous Circuits and Systems (ASYNC), 2014 20th IEEE International Symposium. -2014. -P.101-102.
57. Restle P. J. , McNamara T. G. , Webber D. A. , Camporese P. J. , Eng K. F. , Jenkins K. A. , Allen D. H. , Rohn M. J. , Quaranta M. P. , Boerstler D. W. , Alpert C. J. A clock distribution network for microprocessors. // IEEE Journal of Solid-State Circuits. - 2001. -Vol.36, No.5. -P.792-799.
58. Piguet C. Low-Power CMOS Circuits: Technology, Logic Design and CAD Tools. - CRC Press, 2005. –440p.
59. Zarate-Roldan J. , Carreon-Bautista S. , Costilla-Reyes A. , Sánchez-Sinencio E. A power management unit with 40 dB switching-noise-suppression for a thermal harvesting array. //

- IEEE Transactions on Circuits and Systems I: Regular Papers. -2015. -Vol.62, No.8. -P.1918-1928.
60. Lin H. N. , Kuo C. W. , Chen H. C. Electromagnetic interference noise suppression of writing/reading flash memory with multiple-execution control program. // IET Science, Measurement & Technology. -2015. -Vol.9, No.8. -P.1032-1038.
 61. Chelcea T. , Nowick S. M. A low-latency FIFO for mixed-clock systems. // VLSI, 2000. Proceedings. IEEE Computer Society Workshop. -2000. -P.119-126.
 62. Weltin-Wu C. , Zhao G. , Galton I. A 3.5 GHz Digital Fractional-N PLL Frequency Synthesizer Based on Ring Oscillator Frequency-to-Digital Conversion. // IEEE Journal of Solid-State Circuits. -2015. -Vol.50, No.12. -P.2988-3002.
 63. Best R. E. Phase locked loops. -McGraw-Hill Professional, 2007. -490p.
 64. Gangopadhyay S. , Somasekhar D. , Tschanz J. W. , Raychowdhury A. A 32 nm embedded, fully-digital, phase-locked low dropout regulator for fine grained power management in digital circuits. // IEEE Journal of Solid-State Circuits. -2014. -Vol.49, No.11. -P.2684-2693.
 65. Sharif N. , Ramzan N. , Lodhi F. K. , Hasan O. , Hasan S. R. Quantitative analysis of state-of-the-art synchronizers: Clock domain crossing perspective. // Emerging Technologies (ICET), 2011 7th International Conference. -2011. -P.1-6.
 66. Al-bayati Z. Design and Verification of Clock Domain Crossing Interfaces (Doctoral dissertation, Concordia University Montréal, Québec, Canada). 2012. -77p.
 67. Kapschitz T. , Ginosar R. Formal verification of synchronizers. // Advanced Research Working Conference on Correct Hardware Design and Verification Methods. Springer Berlin Heidelberg. -2005. -P.359-362.
 68. Taskin B. Advanced Timing and Synchronization Methodologies for Digital VLSI Integrated Circuits (Doctoral dissertation, University of Pittsburgh). 2005. -199p.
 69. Cummings C. E. Clock domain crossing (cdc) design & verification techniques using systemverilog. SNUG-2008, Boston. 2008. -56p.
 70. Al-bayati Z. , Mohamed O. A. , Hasan S. R. , Savaria Y. Design of a C-element based clock domain crossing interface. // Microelectronics (ICM), 2012 24th International Conference. -2012. -P.1-4.
 71. Karimi N. , Chakrabarty K. Detection, Diagnosis, and Recovery From Clock-Domain Crossing Failures in Multiclock SoCs. // IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. -2013. -Vol.32, No.9. -P.1395-1408.
 72. Sarwary S. , Verma S. Critical clock-domain-crossing bugs. // EDN. -2008. -Vol.53, No.7. -P.55-64.
 73. Herlev M. , Poulsen C. K. , Spars J. Open core protocol (OCP) clock domain crossing interfaces. // NORCHIP. IEEE. -2014. -P.1-6.

74. Hsiao S. F. , Lin C. G. , Wu P. H. , Wen C. S. Asynchronous AHB bus interface designs in a multiple-clock-domain graphics system. // Circuits and Systems (APCCAS), 2012 IEEE Asia Pacific Conference. -2012. -P.408-411.
75. Մելիքյան Վ., Խաժակյան Տ., Հախճախյան Տ., Մանուկյան Ս. Տարբեր հաճախականությամբ աշխատող տակտավորվող թվային սխեմաների համաձայնեցման եղանակ // ՀՀ ԳԱԱ և ՀԱՊՀ տեղեկագիր. Տեխնիկական գիտությունների սերիա.-2016. -Հ.69, N4, ISSN 0002-306X. -Էջ 381-392
76. Melikyan V., Babayan E., Khazhakyan T., Manukyan S. Analysis of the impact of metastability phenomenon on the latency and power consumption of synchronizer circuits // East-West Design & Test Symposium, Yerevan, Armenia. -2016. -P.100-102.
77. Parulkar I. , Wood A. , Hoe J. C. , Falsafi B. , Adve S. V. , Torrellas J. , Mitra S. OpenSPARC: An open platform for hardware reliability experimentation. // Fourth Workshop on Silicon Errors in Logic-System Effects (SELSE). -2008. -P.1-6
78. <https://github.com/VectorBlox/orca>
79. Kang W. , Zhang L. , Klein J. O. , Zhang Y. , Ravelosona D. , Zhao W. Reconfigurable codesign of STT-MRAM under process variations in deeply scaled technology. // IEEE Transactions on Electron Devices. -2015. -Vol.62, No.6. -P.1769-1777.
80. Lorenz J. , Bär E. , Burenkov A. , Evanschitzky P. , Asenov A. , Wang L. , Wang X. , Brown A. R. , Millar C. , Reid D. Simultaneous simulation of systematic and stochastic process variations. // Simulation of Semiconductor Processes and Devices (SISPAD), 2014 International Conference. -2014. -P.289-292.
81. Gonzalez J. , Cruz J. , Vázquez D. , Rueda A. Analysis of process variations' impact on a 2.4 GHz 90 nm CMOS LNA. // Circuits and Systems (LASCAS), 2013 IEEE Fourth Latin American Symposium. -2013. -P.1-4.
82. Shur M. Terahertz compact SPICE model. // Mixed Design of Integrated Circuits and Systems, 2016 MIXDES-23rd International Conference. -2016. -P.27-31.
83. Yang K. , Blaauw D. , Sylvester D. An all-digital edge racing true random number generator robust against pvt variations. // IEEE Journal of Solid-State Circuits. -2016. - Vol.51, No.4. -P.1022-1031.
84. Gunther S. H. , Jourdan S. , Greiner R. , Burton E. A. , Deval A. S. , Cornaby M. , Shrall J. , Ramadorai R. , inventors; Intel Corporation, assignee. Integrated circuit performance improvement across a range of operating conditions and physical constraints. United States patent US 8,949,635. 2015 Feb 3.
85. Babayan E., Khazhakyan T., Melikyan N., Beglaryan N. High PSRR Bandgap Reference Circuit Based on SAED32/28nm EDK // International Conference on Semiconductor Micro & Nanoelectronics.- Yerevan, Armenia, 2013. - P.187-189.

86. Goldman R., Bartleson K., Wood T., Melikyan V., Babayan E., Khazhakyan T. Experience of low power design teaching // European Workshop on Microelectronics Education.- Grenoble, France, 2012. -P.141-144.
87. Tschanz J. , Kim N. S. , Dighe S. , Howard J. , Ruhl G. , Vangal S. , Narendra S. , Hoskote Y. , Wilson H. , Lam C. , Shuman M. Adaptive frequency and biasing techniques for tolerance to dynamic temperature-voltage variations and aging. // Solid-State Circuits Conference, 2007. ISSCC 2007. Digest of Technical Papers. IEEE International. - 2007, -P.292-604.
88. Gevorgyan A., Baghdasaryan A., Avagyan D., Khazhakyan T., Milic M. Thermal Aware Task Scheduling for 3D ICs // International Conference on Electrical, Electronic and Computing Engineering. -Zlatibor, Serbia, 2013. EL3.7.1-5.
89. Kumar R. , Kursun V. Reversed temperature-dependent propagation delay characteristics in nanometer CMOS circuits. // IEEE Transactions on Circuits and Systems II: Express Briefs. -2006. -Vol.53, No.10. -P.1078-1082.
90. Shafique M. , Garg S. , Henkel J. , Marculescu D. The EDA challenges in the dark silicon era: Temperature, reliability, and variability perspectives. // Proceedings of the 51st Annual Design Automation Conference. -2014. -P.1-6.
91. Salehi S. , DeMara R. F. Energy and Area Analysis of a Floating-Point Unit in 15nm CMOS Process Technology. // SoutheastCon 2015. -2015. -P.1-5.
92. Shafique M. , Garg S. , Henkel J. , Marculescu D. The EDA challenges in the dark silicon era: Temperature, reliability, and variability perspectives. // Proceedings of the 51st Annual Design Automation Conference. -2014. -P.1-6.
93. Zimpeck A. L. , Meinhardt C. , Reis R. A. Impact of PVT variability on 20nm FinFET standard cells. // Microelectronics Reliability. -2015. -Vol.55, No.9. -P.1379-1383.
94. Sharma V. K. , Pattanaik M. , Raj B. PVT variations aware low leakage INDEP approach for nanoscale CMOS circuits. // Microelectronics reliability. -Vol.54, No.1. -P.90-99.
95. Augustine C. , Tokunaga C. , Malavasi A. , Raychowdhury A. , Khellah M. , Tschanz J. , De V. Characterization of PVT variation & aging induced hold time margins of flip-flop arrays at NTV in 22nm tri-gate CMOS. // Electron Devices Meeting (IEDM), 2016 IEEE International. -2016. -P.35-5.
96. Gitterman R. , Teman A. , Meinerzhagen P. , Fish A. , Burg A. A process compensated gain cell embedded-DRAM for ultra-low-power variation-aware design. // Circuits and Systems (ISCAS), 2016 IEEE International Symposium. -2016. -P.1006-1009.
97. Radfar M. , Shah K. , Singh J. A highly sensitive and ultra low-power forward body biasing circuit to overcome severe process, voltage and temperature variations and extreme voltage scaling. // International Journal of Circuit Theory and Applications. -2015. -Vol.43, No.2. -P.233-252.

98. Tang A. , Yang Y. , Lee C. Y. , Jha N. K. McPAT-PVT: Delay and power modeling framework for FinFET processor architectures under PVT variations. // IEEE Transactions on Very Large Scale Integration (VLSI) Systems. -2015. -Vol.23, No.9. -P.1616-1627.
99. Zimpeck A. L. , Meinhardt C. , Posser G. , Reis R. FinFET cells with different transistor sizing techniques against PVT variations. // Circuits and Systems (ISCAS), 2016 IEEE International Symposium. -2016. -P.45-48.
100. Alioto M. , Consoli E. , Palumbo G. . Variations in nanometer CMOS flip-flops: Part I—impact of process variations on timing. // IEEE Transactions on Circuits and Systems I: Regular Papers. -2015. -Vol.62, No.8. -P.2035-2043.
101. Melikyan V., Sahakyan A., Hekimyan A., Trdatyan D., Shishmanyanyan A., Khazhakyan T. Low power duty cycle adjustment simple method in high speed serial links // East-West Design & Test Symposium, Batumi, Georgia, 2015.- P.43-46.
102. Lin T.T., Tu W.P., Huang S.H. Self-adjusting mechanism to dynamically suppress the effect of PVT variations on clock skew. // Circuits and Systems (APCCAS), 2014 IEEE Asia Pacific Conference. 2014. -P.308-311.
103. Zhou N.Y., Restle P., Palumbo J., Kozhaya J., Qian H., Li Z., Alpert C.J., Sze C. PACMAN: Driving Nonuniform Clock Grid Loads for low-skew robust clock network. // System Level Interconnect Prediction (SLIP), 2014 ACM/IEEE International Workshop. 2014 -P.1-5.
104. Tu W.P., Huang S.H., Lu H.H. PVT-variations-tolerant clock design using self-correcting adjustable delay buffers. // Next-Generation Electronics (ISNE), 2014 International Symposium. 2014. -P.1-2.
105. Alioto M., Consoli E., Palumbo G. Hold Time Issues and Impact of Variations on Flip-Flop Topologies. // Flip-Flop Design in Nanometer CMOS. Springer International Publishing. 2015. -P.199-227.
106. Alioto M., Consoli E., Palumbo G. Variations in nanometer CMOS flip-flops: part II—energy variability and impact of other sources of variations. // IEEE Transactions on Circuits and Systems I: Regular Papers. -2015. -Vol.62, No.3. -P.835-843.
107. Sengupta A., Pande P. Comparative Study of the Methodologies Used in Low-Power Master–Slave Flip-Flops. // Proceeding of International Conference on Intelligent Communication, Control and Devices. Springer Singapore. 2017. -P.127-133.
108. Lin T.T., Tu W.P., Huang S.H. Self-adjusting mechanism to dynamically suppress the effect of PVT variations on clock skew. // Circuits and Systems (APCCAS), 2014 IEEE Asia Pacific Conference. Nov 17., 2014. –P.308-311.
109. Augustine C., Tokunaga C., Malavasi A., Raychowdhury A., Khellah M., Tschanz J., De V. Characterization of PVT variation & aging induced hold time margins of flip-flop arrays at

- NTV in 22nm tri-gate CMOS. // Electron Devices Meeting (IEDM), 2016 IEEE International Dec. 3, 2016. -P.35.
110. Huang S.Y., Zheng C.C. Die-to-Die Clock Skew Characterization and Tuning for 2.5 D ICs. // Asian Test Symposium (ATS), 2016 IEEE 25th. Nov 21, 2016. -P.221-226.
 111. Bai Z., Zhou X., Mason R.D., Allan G. Low-phase noise clock distribution network using rotary traveling-wave oscillators and built-in self-test phase tuning technique. // IEEE Transactions on Circuits and Systems II: Express Briefs. -2015. -Vol.62, No.1. -P.41-45.
 112. Huang Z., Jiang B., Li L., Luong H.C. 2.3 A 4.2 μ s-settling-time 3rd-order 2.1 GHz phase-noise-rejection PLL using a cascaded time-amplified clock-skew sub-sampling DLL. // Solid-State Circuits Conference (ISSCC), 2016 IEEE International. 2016. -P.40-41.
 113. Hossain M., Chen E.H., Navid R., Leibowitz B., Chou A., Li S., Park M.J., Ren J., Daly B., Su B., Shirasgaonkar M. A 4 $\times$ 40 Gb/s quad-lane CDR with shared frequency tracking and data dependent jitter filtering. // VLSI Circuits Digest of Technical Papers, 2014 Symposium. 2014. -P.1-2.
 114. Chung S.H., Kim Y.J., Kim Y.H., Kim L.S.. A 10-Gb/s 0.71-pJ/bit forwarded-clock receiver tolerant to high-frequency jitter in 65-nm CMOS. // IEEE Transactions on Circuits and Systems II: Express Briefs. -2016. -Vol.63, No.3. -P.264-268.
 115. Al-Alem Y., Shubair R.M., Kishk A. Clock jitter correction circuit for high speed clock signals using delay units and time selection window. // Microwave Symposium (MMS), 2016 16th Mediterranean. 2016. -P.1-3.
 116. Khazhakyan T. Research of metastability timing characteristics for threshold voltage and process-voltage-temperature variations. // East-West Design & Test Symposium (EWDTS), IEEE October 14-17, -Yerevan, Armenia. 2016. -P.84-87
 117. McElroy C., Neiryneck D., McLaughlin M. Comparison of wireless clock synchronization algorithms for indoor location systems. // Communications Workshops (ICC), 2014 IEEE International Conference. 2014. -P.157-162.
 118. Kiamehr S., Ebrahimi M., Firouzi F., Tahoori M.B. Extending standard cell library for aging mitigation. // IET Computers & Digital Techniques. -2015. -Vol.9, No.4. -P.206-212.
 119. Braibant T., Chlipala A. Formal verification of hardware synthesis. // International Conference on Computer Aided Verification. Springer Berlin Heidelberg. 2013. -P.213-228.
 120. Hassoun S., Sasao T. Logic synthesis and verification. -Springer Science & Business Media, 2012. -454p.
 121. Saeed S.M., Sinanoglu O. Design for testability support for launch and capture power reduction in launch-off-shift and launch-off-capture testing. // IEEE Transactions on Very Large Scale Integration (VLSI) Systems. -2014. -Vol.22, No.3. -P.516-521.

122. Mishra A., Sinha N., Singh V., Chakravarty S., Singh A.D. Modified scan flip-flop for low power testing. // Test Symposium (ATS), 2010 19th IEEE Asian. Dec 1, 2010. -P.367-370.
123. Murali S., Meloni P., Angiolini F., Atienza D., Carta S., Benini L., De Micheli G., Raffo L. Designing application-specific networks on chips with floorplan information. // IEEE/ACM international conference on Computer-aided design. 2006. -P.355-362.
124. Liu W.H., Wei Y., Sze C., Alpert C.J., Li Z., Li Y.L., Viswanathan N. Routing congestion estimation with real design constraints. // 50th Annual Design Automation Conference. 2013. -P.92.
125. Qi Z., Cai Y., Zhou Q. Accurate prediction of detailed routing congestion using supervised data learning. // Computer Design (ICCD), 2014 32nd IEEE International Conference. 2014. -P.97-103.
126. Chaitanya N.K., Varadarajan S., Sreenivasulu P. Adaptive multi-path routing for congestion control. // Advance Computing Conference (IACC), 2014 IEEE International. 2014. -P.189-192.
127. Taghavi T, Li Z, Alpert C, Nam GJ, Huber A, Ramji S. New placement prediction and mitigation techniques for local routing congestion. // Computer-Aided Design (ICCAD), 2010 IEEE/ACM International Conference. 2010. -P.621-624.
128. Li W., Dhar S., Pan D.Z. UTPlaceF: A routability-driven FPGA placer with physical and congestion aware packing. // Computer-Aided Design (ICCAD), 2016 IEEE/ACM International Conference. 2016. -P.1-7.
129. Markov I.L., Hu J., Kim M.C. Progress and challenges in VLSI placement research. // Proceedings of the IEEE. -2015. -Vol.103, No.11. -P.1985-2003.
130. Sitik C., Filippini L., Salman E., Taskin B. High performance low swing clock tree synthesis with custom D flip-flop design. // VLSI (ISVLSI), 2014 IEEE Computer Society Annual Symposium. -2014. -P.498-503.
131. Ravi S., Mittal Y., Kittur H.M. Low power obstacle and skew aware clock tree synthesis. // Communication and Computing Systems. CRC Press. 2016. -P.593-597.
132. Shang Y., Zhang C., Yu H., Tan C.S., Zhao X., Lim S.K. Thermal-reliable 3D clock-tree synthesis considering nonlinear electrical-thermal-coupled TSV model. // Design Automation Conference (ASP-DAC), 2013 18th Asia and South Pacific. 2013. -P.693-698.
133. Chen F.W., Hwang T. Clock-tree synthesis with methodology of reuse in 3D-IC. // ACM Journal on Emerging Technologies in Computing Systems (JETC). -2014. -Vol.10, No.3. -P.22.
134. Lu T., Srivastava A. Gated low-power clock tree synthesis for 3D-ICs. // Low Power Electronics and Design (ISLPED), 2014 IEEE/ACM International Symposium. 2014.-P. 319-322.

135. Sitik C., Lerner S., Taskin B. Timing characterization of clock buffers for clock tree synthesis. // Computer Design (ICCD), 2014 32nd IEEE International Conference. 2014. -P. 230-236.
136. Fischbach R., Heinig A., Schneider P. Design rule check and layout versus schematic for 3D integration and advanced packaging. // 3D Systems Integration Conference (3DIC), 2014 International. 2014. -P.1-7.
137. Luo G., Jin B., Zhang W. A Fast and Simple Block-Based Approach for Common Path Pessimism Removal in Static Timing Analysis. // Computer-Aided Design and Computer Graphics (CAD/Graphics), 2015 14th International Conference. 2015. -P.234-235.
138. Ramprasath S., Vijaykumar M., Vasudevan V. A Skew-Normal Canonical Model for Statistical Static Timing Analysis. // IEEE Transactions on Very Large Scale Integration (VLSI) Systems. -2016. -Vol.24, No.6. -P.2359-2368.
139. Oh D.K., Jin M.W., Kim J.H. Extracting the K-most Critical Paths in Multi-corner Multi-mode for Fast Static Timing Analysis. // JOURNAL OF SEMICONDUCTOR TECHNOLOGY AND SCIENCE. -2016. -Vol.16, No.6. -P.771-780.
140. Bhatnagar P., Garg S. Dynamic Threshold Delay Characterization Model for Improved Static Timing Analysis. // Journal of Electronic Testing. -2014. -Vol.30, No.5. -P.495-504.
141. Zhang G.L., Li B., Schlichtmann U. PieceTimer: A holistic timing analysis framework considering setup/hold time interdependency using a piecewise model. // Computer-Aided Design (ICCAD), 2016 IEEE/ACM International Conference. 2016. -P.1-8.
142. Dhamija N., Lalani G., Nelson M., Brown J., Spruth H., Sharma P. Test circuits to characterize setup/hold/access times, minimum voltage and maximum frequency of operation for memory compilers. // Microelectronic Test Structures (ICMTS), 2016 International Conference. 2016. -P.122-125.
143. Waters A., Muhlestein J., Moon U.K. Analysis of Metastability Errors in Conventional, LSB-First, and Asynchronous SAR ADCs. // IEEE Transactions on Circuits and Systems I: Regular Papers. -2016. -Vol.63, No.11. -P.1898-1909.
144. Amarú L., Gaillardon P.E., De Micheli G. MIXSyn: An efficient logic synthesis methodology for mixed XOR-AND/OR dominated circuits. // Design Automation Conference (ASP-DAC), 2013 18th Asia and South Pacific. 2013. -P.133-138.
145. Chakraborty R.S., Bhunia S. RTL hardware IP protection using key-based control and data flow obfuscation. // VLSI Design, 2010. VLSID'10. 23rd International Conference. 2010. -P.405-410.
146. Tehranipoor M., Salmani H., Zhang X. Design for Hardware Trust: Dummy Scan Flip-Flop Insertion. In Integrated Circuit Authentication. Springer International Publishing. 2014. -P. 39-67.

147. Tudruj M., Kopanski D., Borkowski J. Dynamic Workflow Control with Global States Monitoring. // Parallel and Distributed Computing, 2007. ISPD'07. Sixth International Symposium. 2007. -P.44-44.
148. Estes D.L., inventor; Don Estes & Associates, Inc., assignee. Method and system for data synchronization. United States patent application US 15/099,560. 2016 Apr 14.
149. Das J.C., De D. Reversible binary to grey and grey to binary code converter using QCA. // IETE Journal of Research. -2015. -Vol.61, No.3. -P.223-229.
150. Tarawneh G., Yakovlev A., Mak T. Eliminating synchronization latency using sequenced latching. // IEEE Transactions On Very Large Scale Integration (Vlsi) Systems. -2014. - Vol.22, No.2. -P.408-419.
151. <https://www.python.org/>
152. <https://wiki.python.org/moin/PyQt>
153. <https://www.centos.org/>
154. <https://www.microsoft.com/en-us/software-download/windows7>
155. Thomas D., Moorby P. The Verilog® Hardware Description Language. -Springer Science & Business Media, 2008. -380p

ՀԱՎԵԼՎԱԾ 1. ՆԵՐԴՐՄԱՆ ԱԿՏ

SYNOPSYS®

№ 339/12

"26" 09 2017



Հաստատում եմ՝

«ՍԻՆՈՓՍԻՍ ԱՐՄԵՆԻԱ» ՓԲԸ

Գլխավոր տնօրեն՝

Հ. Մուսայեյան

Տիգրան Ռուբենի Խաժակյանի «Տարբեր հաճախություններով տակտավորվող թվային սխեմաների համաձայնեցման միջոցների մշակումը» թեմայով թեկնածուական ատենախոսության արդյունքների

ՆԵՐԴՐՄԱՆ ԱԿՏ

ԵՊՀ «Էլեկտրոնիկա, միկրո և նանոէլեկտրոնիկա» մասնագիտությամբ 3-րդ կուրսի ասպիրանտ Ս. Ռ. Խաժակյանի կողմից մշակված նախագծման SmartSync ծրագրային միջոցը ներդրված է «ՍԻՆՈՓՍԻՍ ԱՐՄԵՆԻԱ» ՓԲԸ-ում: Այն օգտագործվում է բարձր արագագործություն պահանջող ԻՍ-երում հաղորդակցվող սինքրոտիզմների համաձայնեցման համար, երբ արագագործության հետ մեկտեղ անհրաժեշտ է ապահովել ԳԼՋ շեղումների նկատմամբ անտարբերություն, և երբ մակերեսի որոշակի մեծացումն ընդունելի է: Մշակված SmartSync ծրագրային միջոցի կիրառումը զգալիորեն պարզեցրել է նման համակարգերում սինքրոտիզմների համաձայնեցումը: Ծրագրային միջոցը կազմում է ինտեգրալ սխեմաներում սինքրոնացման գործընթացի կարևոր մասը և բավարարում է ժամանակակից էլեկտրոնային նախագծման բնագավառում կիրառվող ծրագրային գործիքային միջոցին ներկայացվող բոլոր պահանջներին:

Անալոգային և խառը ազդանշանային սխեմաների

նախագծման դեպարտամենտի տնօրեն՝

Վ. Մուսայան

"ՍԻՆՈՓՍԻՍ ԱՐՄԵՆԻԱ" ՓԲԸ
0026, ԶՂ, ԵՐԵՎԱՆ, ԱՐՇԱԿՈՒՆՅԱՆ 41
ՖԵՌ. (+374 10) 49 21 00, ՖԱՔՍ (+374 10) 49 26 96
ՅԱԳՅ 02236362

"SYNOPSYS ARMENIA" CJSC
41 ARSHAKUNYATS AVE., YEREVAN, ARMENIA, 0026
TEL.: (+374 10) 49 21 00, FAX: (+374 10) 49 26 96
TAX PAYER'S ID 02236362

ՀԱՎԵԼՎԱԾ 2. SMARTSYNC ԾՐԱԳՐՈՎ ՀԱՄԱՁԱՅՆԵՑՎԱԾ ORCA

ՊՐՈՑԵՍՈՐԻ VERILOG ՆԿԱՐԱԳՐՈՒԹՅՈՒՆԻՑ ՀԱՏՎԱԾ

```
module orca_module
(
    src1,
    src2,

    src_val,
    src_ec,
    a_ct,
    a_st,
    a_dest,
    a_sdest,
    a_dest_val,
    a_d,
    clk,
    rst
);

input clk1, clk2, clk3;
input rst;

input [31:0] src1;
input [31:0] src2;
input [31:0] src2;
input      src_val;
input      src_ec;

input [31:0] a_ct;
input      a_st;

output [31:0] a_dest;
output      a_sdest;
output      a_dest_val;
output      a_d;

reg [31:0] a_dest;
reg      a_dest_val;
reg      a_d;

reg [31:0] muin1;
reg [31:0] muin2;
wire [63:0] mout;
reg [1:0] mcyc;
wire mbus;
wire md;
reg mop;
```



```

wire [31:0]  tc;
wire [31:0]  tc2;
            wire [31:0]          tcc;

```

```

reg [31:0] dt;
reg [31:0] dt;
reg [31:0] dt;
reg [31:0] dt;
reg [31:0] dt;
reg [31:0] dt;
reg [31:0] dt;

```

```

            assign tc = ~src1 + 32'd1;
assign tc2 = ~src2 + 32'd1;
assign tcc = ~src2 + 32'd1;

```

```

            assign a_sdest = a_dest_val;

```

```

always @(posedge clk3) begin
  casex(a_ct[31:24])
    {`A_FORMAT} :
      begin
        dt <= a_ct[`A_ABS1_POS]
              ? (src1[31] ? tc : src1)
              : src1;
        dt <= a_ct[`A_ABS2_POS]
              ? (src2[31] ? tc2 : src2)
              : src2;
        dt <= a_ct[`A_ABS3_POS]
              ? (src2[31] ? tcc : src2)
              : src2;
      end
    default :
      begin
        dt <= src1;
        dt <= src2;
        dt <= src2;
      end
  endcase
end

```

```

always @* begin
  casex(a_ct[31:24])
    {`A_FORMAT} :
      begin
        dt <= a_ct[`A_NEG1_POS] ? (~dt + 32'd1) : dt;
        dt <= a_ct[`A_NEG2_POS] ? (~dt + 32'd1) : dt;
      end
  end

```

```

    dt <= a_ct[`A_NEG3_POS] ? (~dt + 32'd1) : dt;
    dt <= a_ct[`A_NEG1_POS] ? tc : src1;
    dt <= a_ct[`A_NEG2_POS] ? tc2 : src2;
    dt <= a_ct[`A_NEG3_POS] ? tcc : src2;
end
{`B_FORMAT} :
begin
    dt <= a_ct[`B_NEG1_POS] ? (~dt + 32'd1) : dt;
    dt <= a_ct[`B_NEG2_POS] ? (~dt + 32'd1) : dt;
    dt <= a_ct[`B_NEG3_POS] ? (~dt + 32'd1) : dt;
    dt <= a_ct[`B_NEG1_POS] ? tc : src1;
    dt <= a_ct[`B_NEG2_POS] ? tc2 : src2;
    dt <= a_ct[`B_NEG3_POS] ? tcc : src2;
end
default :
begin
    dt <= dt;
    dt <= dt;
    dt <= dt;
    dt <= src1;
    dt <= src2;
    dt <= src2;
end
endcase
end

always @(posedge clk2) begin
    casex({src_ec, a_ct[31:24], a_ct[11:0]})
        {1'b0, 8'h??, 12'h???} :
            begin
                a_d <= 1'b0;
                a_dest <= {32{1'bx}};
                a_dest_val <= src_val;
            end
        {1'b1, `_FORMAT, 12'h001} :
            begin
                a_d <= 1'b1;
                a_dest <= src1;
                a_dest_val <= src_val;
            end
        {1'b1, `_FORMAT, 12'h025} :
            begin
                a_d <= 1'b1;
                {a_dest_val, a_dest} <= dt + dt;
            end
        {1'b1, `_FORMAT, 12'h026} :
            begin
                a_d <= 1'b1;
                {a_dest_val, a_dest} <= dt - dt;
            end
        {1'b1, `_FORMAT, 12'h01B} :

```

```

begin
  a_d <= 1'b1;
  a_dest <= src1 & src2;
  a_dest_val <= src_val;
end
{1'b1, `A_FORMAT, 12'h11B} :
begin
  a_d <= 1'b1;
  a_dest <= src1 & src2;
  a_dest_val <= src_val;
end
{1'b1, `_FORMAT, 12'h01C} :
begin
  a_d <= 1'b1;
  a_dest <= src1 | src2;
  a_dest_val <= src_val;
end
{1'b1, `_FORMAT, 12'h01A} :
begin
  a_d <= 1'b1;
  a_dest <= src2 << src1[4:0];
  a_dest_val <= src_val;
end
{1'b1, `_FORMAT, 12'h016} :
begin
  a_d <= 1'b1;
  a_dest <= src2 >> src1[4:0];
  a_dest_val <= src_val;
end
{1'b1, `_FORMAT, 12'h018} :
begin
  a_d <= 1'b1;
  a_dest <= dt >> dt[4:0];
  a_dest_val <= src_val;
end
{1'b1, `_FORMAT, 12'h014} :
begin
  a_d <= 1'b1;
  a_dest <= (dt >= dt) ? dt : dt;
  a_dest_val <= src_val;
end
{1'b1, `_FORMAT, 12'h012} :
begin
  a_d <= 1'b1;
  a_dest <= (dt >= dt) ? dt : dt;
  a_dest_val <= src_val;
end
{1'b1, `A_FORMAT, 12'h114} :
begin
  a_d <= 1'b1;
  a_dest <= (dt >= dt) ? dt : dt;
end

```

```

    a_dest_val <= src_val;
end
{1'b1, `_FORMAT, 12'h013} :
begin
    a_d <= 1'b1;
    a_dest <= (dt <= dt) ? dt : dt;
    a_dest_val <= src_val;
end
{1'b1, `_FORMAT, 12'h113} :
begin
    a_d <= 1'b1;
    a_dest <= (dt <= dt) ? dt : dt;
    a_dest_val <= src_val;
end
{1'b1, `_FORMAT, 12'h000} :
begin
    a_d <= 1'b1;
    a_dest <= src_val ? src2 : src1;
    a_dest_val <= src_val;
end
{1'b1, `C_FORMAT, 12'h080} :
begin
    a_d <= 1'b1;
    a_dest <= {32{1'bx}};
    a_dest_val <= 1'b0;
end
{1'b1, `C_FORMAT, 12'h081} :
begin
    a_d <= 1'b1;
    a_dest <= {32{1'bx}};
    a_dest_val <= (dt < dt);
end
{1'b1, `C_FORMAT, 12'h082} :
begin
    a_d <= 1'b1;
    a_dest <= {32{1'bx}};
    a_dest_val <= (dt == dt);
end
{1'b1, `C_FORMAT, 12'h083} :
begin
    a_d <= 1'b1;
    a_dest <= {32{1'bx}};
    a_dest_val <= (dt <= dt);
end
{1'b1, `C_FORMAT, 12'h084} :
begin
    a_d <= 1'b1;
    a_dest <= {32{1'bx}};
    a_dest_val <= (dt > dt);
end
{1'b1, `C_FORMAT, 12'h085} :

```

```

begin
  a_d <= 1'b1;
  a_dest <= {32{1'bx}};
  a_dest_val <= (dt != dt);
end
{1'b1, `C_FORMAT, 12'h086} :
begin
  a_d <= 1'b1;
  a_dest <= {32{1'bx}};
  a_dest_val <= (dt >= dt);
end
{1'b1, `C_FORMAT, 12'h087} :
begin
  a_d <= 1'b1;
  a_dest <= {32{1'bx}};
  a_dest_val <= 1'b1;
end
{1'b1, `C_FORMAT, 12'h0C0} :
begin
  a_d <= 1'b1;
  a_dest <= {32{1'bx}};
  a_dest_val <= 1'b0;
end
{1'b1, `C_FORMAT, 12'h0C6} :
begin
  a_d <= 1'b1;
  a_dest <= {32{1'bx}};
  a_dest_val <= ({1'b0, dt} >= {1'b0, dt});
end
{1'b1, `C_FORMAT, 12'h0C7} :
begin
  a_d <= 1'b1;
  a_dest <= {32{1'bx}};
  a_dest_val <= 1'b1;
end
{1'b1, `A_FORMAT, 12'h080} :
begin
  a_d <= 1'b1;
  a_dest <= {32{1'bx}};
  a_dest_val <= 1'b0;
end
{1'b1, `A_FORMAT, 12'h081} :

begin
  a_d <= 1'b1;
  a_dest <= {32{1'bx}};
  a_dest_val <= (dt >= dt);
end
{1'b1, `A_FORMAT, 12'h087} :
begin
  a_d <= 1'b1;

```

```

    a_dest <= {32{1'bx}};
    a_dest_val <= 1'b1;
end
{1'b1, `A_FORMAT, 12'h0C0} :
begin
    a_d <= 1'b1;
    a_dest <= {32{1'bx}};
    a_dest_val <= 1'b0;
end
{1'b1, `A_FORMAT, 12'h0C1} :
begin
    a_d <= 1'b1;
    a_dest <= {32{1'bx}};
    a_dest_val <= ({1'b0, dt} < {1'b0, dt});
end
{1'b1, `A_FORMAT, 12'h0C2} :
begin
    a_d <= 1'b1;
    a_dest <= {32{1'bx}};
    a_dest_val <= ({1'b0, dt} == {1'b0, dt});
end
{1'b1, `A_FORMAT, 12'h0C3} :
begin
    a_d <= 1'b1;
    a_dest <= {32{1'bx}};
    a_dest_val <= ({1'b0, dt} <= {1'b0, dt});
end
{1'b1, `A_FORMAT, 12'h0C4} :
begin
    a_d <= 1'b1;
    a_dest <= {32{1'bx}};
    a_dest_val <= ({1'b0, dt} > {1'b0, dt});
end
{1'b1, `A_FORMAT, 12'h0C5} :
begin
    a_d <= 1'b1;
    a_dest <= {32{1'bx}};
    a_dest_val <= ({1'b0, dt} != {1'b0, dt});
end
{1'b1, `A_FORMAT, 12'h0C6} :
begin
    a_d <= 1'b1;
    a_dest <= {32{1'bx}};
    a_dest_val <= ({1'b0, dt} >= {1'b0, dt});
end
{1'b1, `A_FORMAT, 12'h0C7} :
begin
    a_d <= 1'b1;
    a_dest <= {32{1'bx}};
    a_dest_val <= 1'b1;
end

```

```

{1'b1, `A_FORMAT, 12'h16A} :
begin
  a_d <= md;
  a_dest <= mout[63:32];
  a_dest_val <= src_val;
end
{1'b1, `_FORMAT, 12'h009} :
begin
  a_d <= md;
  a_dest <= mout[31:0];
  a_dest_val <= src_val;
end
{1'b1, `A_FORMAT, 12'h109} :
begin
  a_d <= md;
  a_dest <= mout[31:0];
  a_dest_val <= src_val;
end
{1'b1, `A_FORMAT, 12'h16B} :
begin

  a_d <= md;
  a_dest <= mout[31:0];
  a_dest_val <= src_val;
end
{1'b1, `A_FORMAT, 12'h169} :
begin
  a_d <= md;
  a_dest <= mout[31:0];
  a_dest_val <= src_val;
end
{1'b1, `A_FORMAT, 12'h148} :
begin
  a_d <= 1'b1;
  a_dest <= (dt >> dt[4:0]) & ((1<<dt[4:0])-1);
  a_dest_val <= src_val;
end
{1'b1, `A_FORMAT, 12'h149} :
begin
  a_d <= 1'b1;
  a_dest <= (dt >>> dt[4:0]) & ((1 <<< dt[4:0])-1);
  a_dest_val <= src_val;
end
{1'b1, `A_FORMAT, 12'h14A} :
begin
  a_d <= 1'b1;
  a_dest <= (src1 & src2) | (~src1 & src2);
  a_dest_val <= src_val;
end
{1'b1, `_FORMAT, 12'h028} :

```

```

begin
  a_d <= 1'b1;
  {a_dest_val, a_dest} <= dt + dt + src_val;
end

{1'b1, `_FORMAT, 12'h027} :
begin
  a_d <= 1'b1;
  {a_dest_val, a_dest} <= dt - dt;
end

default :
begin
  a_d <= 1'b0;
  a_dest <= {32{1'bx}};
  a_dest_val <= 1'bx;
end
endcase
end

```

```

always @* begin
  casex({src_ec, a_ct[31:24], a_ct[11:0]})
  {1'b1, `_A_FORMAT, 12'h16A} :
  begin
    mop <= 1'b1;
    muin1 <= dt;
    muin2 <= dt;
  end
  {1'b1, `_FORMAT, 12'h009} :
  begin
    mop <= 1'b1;

    if (src1[31])
    begin
      muin1[23:0] <= tc[23:0];
      muin1[31:24] <= 'd0;
    end
    else
    begin
      muin1[23:0] <= src1[23:0];
      muin1[31:24] <= 'd0;
    end
    if (src2[31])
    begin
      muin2[23:0] <= tc2[23:0];
      muin2[31:24] <= 'd0;
    end
    else
    begin
      muin2[23:0] <= src2[23:0];
      muin2[31:24] <= 'd0;
    end
  end
end

```



```

        end
    end
    {1'b1, `A_FORMAT, 12'h109} :
    begin
        mop <= 1'b1;

        if (src1[31])
            begin
                muin1[23:0] <= tc[23:0];
                muin1[31:24] <= 'd0;
            end
        else
            begin
                muin1[23:0] <= src1[23:0];
                muin1[31:24] <= 'd0;
            end
        end
        if (src2[31])
            begin
                muin2[23:0] <= tc2[23:0];
                muin2[31:24] <= 'd0;
            end
        else
            begin
                muin2[23:0] <= src1[23:0];
                muin2[31:24] <= 'd0;
            end
        end
    end
    {1'b1, `A_FORMAT, 12'h16B} :
    begin
        mop <= 1'b1;
        if (src1[31])
            muin1 <= tc;
        else
            muin1 <= src1;
        if (src2[31])
            muin2 <= tc2;
        else
            muin2 <= src2;
        end
    end
    {1'b1, `A_FORMAT, 12'h169} :
    begin
        mop <= 1'b1;
        muin1 <= src1;
        muin2 <= src2;
    end
    default :
    begin
        mop <= 1'b0;
        muin1 <= 'd0;
        muin2 <= 'd0;
    end
end

```

```

        endcase
    end

    assign mout = muin1 * muin2;

    assign mbus = |mcy;
    assign md = (mcy == 2'd2);

    always @(posedge clk1 or posedge rst)
    if (rst)
        begin
            mcy <= 'd0;
        end
    else
        begin
            if (~mop)
                begin
                    mcy <= 'd0;
                end
            else
                begin
                    if (mcy == 2'd2)
                        mcy <= 'd0;
                    else
                        mcy <= mcy + 'd1;
                    end
                end
            end
        end
    end

endmodule

```

ՀԱՎԵԼՎԱԾ 3. SMARTSYNC ԾՐԱԳՐԱՅԻՆ ԳՈՐԾԻՔԻ PYTHON

ՆԿԱՐԱԳՐՈՒԹՅՈՒՆՆԻՑ ՀԱՏՎԱԾ

```
from PyQt5 import QtCore, QtGui, QtWidgets
```

```
class Ui_MainWindow(object):
    def setupUi(self, MainWindow):
        MainWindow.setObjectName("MainWindow")
        MainWindow.resize(1469, 831)
        self.centralwidget = QtWidgets.QWidget(MainWindow)
        self.centralwidget.setObjectName("centralwidget")
        self.frame = QtWidgets.QFrame(self.centralwidget)
        self.frame.setGeometry(QtCore.QRect(0, 0, 1461, 811))
        self.frame.setFrameShape(QtWidgets.QFrame.StyledPanel)
        self.frame.setFrameShadow(QtWidgets.QFrame.Raised)
        self.frame.setObjectName("frame")
        self.groupBox = QtWidgets.QGroupBox(self.frame)
        self.groupBox.setGeometry(QtCore.QRect(10, 10, 541, 471))
        font = QtGui.QFont()
        font.setPointSize(16)
        self.groupBox.setFont(font)
        self.groupBox.setObjectName("groupBox")
        self.label = QtWidgets.QLabel(self.groupBox)
        self.label.setGeometry(QtCore.QRect(10, 190, 101, 31))
        self.label.setObjectName("label")
        self.label_2 = QtWidgets.QLabel(self.groupBox)
        self.label_2.setGeometry(QtCore.QRect(10, 230, 121, 31))
        self.label_2.setObjectName("label_2")
        self.label_3 = QtWidgets.QLabel(self.groupBox)
        self.label_3.setGeometry(QtCore.QRect(10, 270, 121, 31))
        self.label_3.setObjectName("label_3")
        self.label_4 = QtWidgets.QLabel(self.groupBox)
        self.label_4.setGeometry(QtCore.QRect(10, 310, 151, 31))
        self.label_4.setObjectName("label_4")
        self.lineEdit = QtWidgets.QLineEdit(self.groupBox)
        self.lineEdit.setEnabled(False)
        self.lineEdit.setGeometry(QtCore.QRect(190, 190, 321, 31))
        self.lineEdit.setObjectName("lineEdit")
        self.lineEdit_2 = QtWidgets.QLineEdit(self.groupBox)
        self.lineEdit_2.setEnabled(False)
        self.lineEdit_2.setGeometry(QtCore.QRect(190, 230, 321, 31))
        self.lineEdit_2.setObjectName("lineEdit_2")
        self.lineEdit_3 = QtWidgets.QLineEdit(self.groupBox)
        self.lineEdit_3.setEnabled(False)
        self.lineEdit_3.setGeometry(QtCore.QRect(190, 270, 321, 31))
```

```

self.lineEdit_3.setObjectName("lineEdit_3")
self.lineEdit_4 = QtWidgets.QLineEdit(self.groupBox)
self.lineEdit_4.setEnabled(False)
self.lineEdit_4.setGeometry(QtCore.QRect(190, 310, 321, 31))
self.lineEdit_4.setObjectName("lineEdit_4")
self.label_5 = QtWidgets.QLabel(self.groupBox)
self.label_5.setGeometry(QtCore.QRect(10, 50, 151, 31))
self.label_5.setObjectName("label_5")
self.lineEdit_5 = QtWidgets.QLineEdit(self.groupBox)
self.lineEdit_5.setEnabled(False)
self.lineEdit_5.setGeometry(QtCore.QRect(190, 50, 321, 31))
self.lineEdit_5.setObjectName("lineEdit_5")
self.line = QtWidgets.QFrame(self.groupBox)
self.line.setGeometry(QtCore.QRect(10, 100, 511, 16))
self.line.setFrameShape(QtWidgets.QFrame.HLine)
self.line.setFrameShadow(QtWidgets.QFrame.Sunken)
self.line.setObjectName("line")
self.pushButton = QtWidgets.QPushButton(self.groupBox)
self.pushButton.setEnabled(True)
self.pushButton.setGeometry(QtCore.QRect(320, 120, 191, 41))
self.pushButton.setObjectName("pushButton")
self.pushButton_2 = QtWidgets.QPushButton(self.groupBox)
self.pushButton_2.setEnabled(True)
self.pushButton_2.setGeometry(QtCore.QRect(320, 380, 191, 41))
self.pushButton_2.setObjectName("pushButton_2")
self.line_2 = QtWidgets.QFrame(self.groupBox)
self.line_2.setGeometry(QtCore.QRect(10, 360, 511, 16))
self.line_2.setFrameShape(QtWidgets.QFrame.HLine)
self.line_2.setFrameShadow(QtWidgets.QFrame.Sunken)
self.line_2.setObjectName("line_2")
self.tabWidget = QtWidgets.QTabWidget(self.frame)
self.tabWidget.setGeometry(QtCore.QRect(570, 20, 891, 761))
font = QtGui.QFont()
font.setPointSize(14)
self.tabWidget.setFont(font)
self.tabWidget.setObjectName("tabWidget")
self.tab = QtWidgets.QWidget()
self.tab.setObjectName("tab")
self.label_6 = QtWidgets.QLabel(self.tab)
self.label_6.setGeometry(QtCore.QRect(140, 80, 701, 571))
self.label_6.setText("")
self.label_6.setPixmap(QtGui.QPixmap("2circuitsample.png"))
self.label_6.setObjectName("label_6")
self.tabWidget.addTab(self.tab, "")
self.tab_2 = QtWidgets.QWidget()
self.tab_2.setObjectName("tab_2")
self.textEdit = QtWidgets.QTextEdit(self.tab_2)
self.textEdit.setGeometry(QtCore.QRect(0, 0, 891, 721))
self.textEdit.setObjectName("textEdit")
self.tabWidget.addTab(self.tab_2, "")
MainWindow.setCentralWidget(self.centralwidget)

```

```

self.statusbar = QtWidgets.QStatusBar(MainWindow)
self.statusbar.setObjectName("statusbar")
MainWindow.setStatusBar(self.statusbar)
self.menubar = QtWidgets.QMenuBar(MainWindow)
self.menubar.setGeometry(QtCore.QRect(0, 0, 1469, 21))
self.menubar.setObjectName("menubar")
self.menuFile = QtWidgets.QMenu(self.menubar)
self.menuFile.setObjectName("menuFile")
self.menuImport_2 = QtWidgets.QMenu(self.menubar)
self.menuImport_2.setObjectName("menuImport_2")
self.menuDesign = QtWidgets.QMenu(self.menuImport_2)
self.menuDesign.setObjectName("menuDesign")
MainWindow.setMenuBar(self.menubar)
self.actionSetup = QtWidgets.QAction(MainWindow)
self.actionSetup.setObjectName("actionSetup")
self.actionRegister_Transfer_Level = QtWidgets.QAction(MainWindow)
self.actionRegister_Transfer_Level.setObjectName("actionRegister_Transfer_Level")
self.actionGate_Level = QtWidgets.QAction(MainWindow)
self.actionGate_Level.setObjectName("actionGate_Level")
self.actionSynchronizer_Configuration = QtWidgets.QAction(MainWindow)
self.actionSynchronizer_Configuration.setObjectName("actionSynchronizer_Configuration")
self.actionClock_Domain_Configuration = QtWidgets.QAction(MainWindow)

self.actionClock_Domain_Configuration.setObjectName("actionClock_Domain_Configuration")
self.actionRegister_Transfer_Level_2 = QtWidgets.QAction(MainWindow)
self.actionRegister_Transfer_Level_2.setObjectName("actionRegister_Transfer_Level_2")
self.actionGate_Level_2 = QtWidgets.QAction(MainWindow)
self.actionGate_Level_2.setObjectName("actionGate_Level_2")
self.actionExit = QtWidgets.QAction(MainWindow)
self.actionExit.setObjectName("actionExit")
self.actionExport_Configurations = QtWidgets.QAction(MainWindow)
self.actionExport_Configurations.setObjectName("actionExport_Configurations")
self.menuFile.addAction(self.actionSetup)
self.menuFile.addAction(self.actionExport_Configurations)
self.menuFile.addAction(self.actionExit)
self.menuDesign.addAction(self.actionRegister_Transfer_Level_2)
self.menuDesign.addAction(self.actionGate_Level_2)
self.menuImport_2.addAction(self.actionSynchronizer_Configuration)
self.menuImport_2.addAction(self.actionClock_Domain_Configuration)
self.menuImport_2.addAction(self.menuDesign.menuAction())
self.menubar.addAction(self.menuFile.menuAction())
self.menubar.addAction(self.menuImport_2.menuAction())

self.retranslateUi(MainWindow)
self.tabWidget.setCurrentIndex(1)
QtCore.QMetaObject.connectSlotsByName(MainWindow)

def retranslateUi(self, MainWindow):
    _translate = QtCore.QCoreApplication.translate
    MainWindow.setWindowTitle(_translate("MainWindow", "MainWindow"))
    self.groupBox.setTitle(_translate("MainWindow", "Current Design"))

```

```

self.label.setText(_translate("MainWindow", "File Name:"))
self.label_2.setText(_translate("MainWindow", "Top Module:"))
self.label_3.setText(_translate("MainWindow", "Mode:"))
self.label_4.setText(_translate("MainWindow", "Clock Domains:"))
self.lineEdit.setText(_translate("MainWindow", "sample_design.v"))
self.lineEdit_2.setText(_translate("MainWindow", "SAMPLE"))
self.lineEdit_3.setText(_translate("MainWindow", "RTL"))
self.lineEdit_4.setText(_translate("MainWindow", "5"))
self.label_5.setText(_translate("MainWindow", "Design Status:"))
self.lineEdit_5.setText(_translate("MainWindow", "Available"))
self.pushButton.setText(_translate("MainWindow", "Analyze Design"))
self.pushButton_2.setText(_translate("MainWindow", "Synchronize"))
self.tabWidget.setTabText(self.tabWidget.indexOf(self.tab), _translate("MainWindow",
"Design View"))
self.textEdit.setHtml(_translate("MainWindow", "<!DOCTYPE HTML PUBLIC \"-
//W3C//DTD HTML 4.0//EN\" \"http://www.w3.org/TR/REC-html40/strict.dtd\">\n"
"<html><head><meta name=\"qrichtext\" content=\"1\" /><style type=\"text/css\">\n"
"p, li { white-space: pre-wrap; }\n"
"</style></head><body style=\" font-family:'MS Shell Dlg 2'; font-size:14pt; font-weight:400;
font-style:normal;\">\n"
"<p style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-
indent:0; text-indent:0px;\">#####</p>\n"
"<p style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-
indent:0; text-indent:0px;\">#### SYNCHORNIZATION COMPLETED! ####</p>\n"
"<p style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-
indent:0; text-indent:0px;\">#####</p>\n"
"<p style=\"-qt-paragraph-type:empty; margin-top:0px; margin-bottom:0px; margin-left:0px;
margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><br /></p>\n"
"<p style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-
indent:0; text-indent:0px;\">Synchronizer insertion into RTL Design sample_design.v
succeeded...</p>\n"
"<p style=\"-qt-paragraph-type:empty; margin-top:0px; margin-bottom:0px; margin-left:0px;
margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><br /></p>\n"
"<p style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-
indent:0; text-indent:0px;\">##### Input info #####</p>\n"
"<p style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-
indent:0; text-indent:0px;\">Module name: SAMPLE</p>\n"
"<p style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-
indent:0; text-indent:0px;\">No. of clock domains: 5</p>\n"
"<p style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-
indent:0; text-indent:0px;\">No. of single bit signals: 3</p>\n"
"<p style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-
indent:0; text-indent:0px;\">No. of multi bit signals: 4</p>\n"
"<p style=\"-qt-paragraph-type:empty; margin-top:0px; margin-bottom:0px; margin-left:0px;
margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><br /></p>\n"
"<p style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-
indent:0; text-indent:0px;\">##### Output info #####</p>\n"
"<p style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-
indent:0; text-indent:0px;\">Total synchronizers used: 9 </p>\n"
"<p style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-
indent:0; text-indent:0px;\">Synchronizer MDAR used: 7</p>\n"

```

```

"<p style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;\">Synchronizer DAOC used: 2</p>\n"
"<p style=\"-qt-paragraph-type:empty; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><br /></p>\n"
"<p style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;\">Output file output_sample_design.v has been created...</p>\n"
"<p style=\"-qt-paragraph-type:empty; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><br /></p>\n"
"<p style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;\">Time Elapsed: 3 s</p>\n"
"<p style=\"-qt-paragraph-type:empty; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><br /></p></body></html>"))
    self.tabWidget.setTabText(self.tabWidget.indexOf(self.tab_2), _translate("MainWindow",
"Synchronization Statistics"))
    self.menuFile.setTitle(_translate("MainWindow", "File"))
    self.menuImport_2.setTitle(_translate("MainWindow", "Import"))
    self.menuDesign.setTitle(_translate("MainWindow", "Design"))
    self.actionSetup.setText(_translate("MainWindow", "Setup"))
    self.actionRegister_Transfer_Level.setText(_translate("MainWindow", "Register Transfer
Level"))
    self.actionGate_Level.setText(_translate("MainWindow", "Gate Level"))
    self.actionSynchronizer_Configuration.setText(_translate("MainWindow", "Synchronizer
Configuration"))
    self.actionClock_Domain_Configuration.setText(_translate("MainWindow", "Clock Domain
Configuration"))
    self.actionRegister_Transfer_Level_2.setText(_translate("MainWindow", "Register Transfer
Level"))
    self.actionGate_Level_2.setText(_translate("MainWindow", "Gate Level"))
    self.actionExit.setText(_translate("MainWindow", "Exit"))
    self.actionExport_Configurations.setText(_translate("MainWindow", "Export
Configurations"))

```

ՀԱՎԵԼՎԱԾ 4. ՕԳՏԱԳՈՐԾՎԱԾ ՆԿԱՐՆԵՐԻ, ԱՂՅՈՒՍԱԿՆԵՐԻ ԵՎ ՀԱՊԱՎՈՒՄՆԵՐԻ ՑԱՆԿԸ

Նկարների ցանկ

Նկ. 1.1. Ասինքրոն սխեմաների՝ ՁԱ-ով համաձայնացումը.....	10
Նկ. 1.2. Քառափուլ-միագիծ ձեռքսեղմման արձանագրությունը.....	11
Նկ. 1.3. Երկփուլ միագիծ ձեռքսեղմման արձանագրությունը.....	11
Նկ. 1.4. Քառափուլ-երկգծանի ձեռքսեղմման արձանագրությունը.....	12
Նկ. 1.5. Երկփուլ-երկգծանի ձեռքսեղմման արձանագրությունը.....	13
Նկ. 1.6. Նույն սինքրոտիրույթում գտնվող հանգույցները.....	14
Նկ. 1.7. Տարրական համաձայնեցված սխեմայի կառուցվածքը.....	15
Նկ. 1.8. ՀՀԲ-ի՝ շրջիչ լինելու դեպքում տարրական համաձայնեցված սխեմայի Ժամանակային դիագրամները.....	16
Նկ. 1.9. Էներգասպառման՝ հաճախականությունից ունեցած կախվածությունը..	18
Նկ. 1.10. ԻՍ-ում ծանրաբեռնվածությունը գնահատող սխեման.....	19
Նկ. 1.11. Սինքրոագղանշանի էներգասպառման՝ սինքրոտիրույթների քանակից ունեցած կախվածությունը.....	21
Նկ. 1.12. Երկու սինքրոտիրույթ ունեցող ԳԱՏՍ համակարգը.....	21
Նկ. 1.13. Երկտակտ տրիգերում մետակայուն վիճակի առաջացումը.....	22
Նկ. 1.14. Կայունացման ժամանակի՝ մուտքի և սինքրոագղանշանի ճակատների հեռավորությունից կախումը.....	24
Նկ. 1.15. ԳԱՏՍ համակարգերի նախագծման երթուղին.....	26
Նկ. 1.16. Հետադարձ կապով մուլտիպլեքսորով տրիգերային համաձայնեցուցիչը.....	28
Նկ. 1.17. Մակարդակային համաձայնեցուցիչը.....	29
Նկ. 1.18. Ճակատ գրանցող համաձայնեցուցիչը.....	29
Նկ. 1.19. Իմպուլս գեներացնող համաձայնեցուցիչը.....	30
Նկ. 1.20. Սինքրոագղանշանի ընդհատմամբ համաձայնեցման ինտերֆեյսը.....	32

Նկ. 1.21. Սահմանային համաձայնեցմամբ ինտերֆեյսը.....	34
Նկ. 1.22. Համաձայնեցման հերթահենք ինտերֆեյս.....	35
Նկ. 1.23. ԲՄԱ արձանագրությամբ հաղորդակցվող բլոկերի համաձայնեցման ինտերֆեյսը.....	36
Նկ. 1.24. ԲՄԱ-ի ինտերֆեյսի աշխատանքը.....	37
Նկ. 1.25. Առանց տվյալների բուֆերացման ԲՄԱ-ի համաձայնեցման ինտերֆեյսը.....	38
Նկ. 1.26. Տվյալների բուֆերացմամբ ԲՄԱ-ի համաձայնեցման ինտերֆեյսը	39
Նկ. 1.27. Ասինքրոն հերթի սխեման.....	40
Նկ. 1.28. ՄՍ բլոկի «տեր» հանգույց լինելու դեպքում միջսինքրոտիրոյթային գրանցման գործողության համաձայնեցումը.....	40
Նկ. 1.29. ՄՍ բլոկի «ծառա» հանգույց լինելու դեպքում միջսինքրոտիրոյթային գրանցման գործողության համաձայնեցումը.....	41
Նկ. 1.30. ՄՍ բլոկի՝ «տեր» հանգույց լինելու դեպքում ընթերցման գործողության համաձայնեցումը.....	41
Նկ. 1.31. ՄՍ բլոկի՝ «ծառա» հանգույց լինելու դեպքում ընթերցման գործողության համաձայնեցումը.....	42
Նկ. 1.32. Գրանցման գործողության դեպքում AHB և ԲՄԱ հիմքով համաձայնեցման ինտերֆեյսների արտադրողականությունների՝ ՄՍ բլոկի և դրդի հաճախականային հարաբերությունից կախումը.....	42
Նկ. 1.33. Ընթերցման գործողության դեպքում AHB և ԲՄԱ հիմքով համաձայնեցման ինտերֆեյսների արտադրողականությունների՝ ՄՍ բլոկի և դրդի հաճախականային հարաբերությունից կախումը.....	43
Նկ. 2.1. ԱԳՇ-երից կախված՝ շեմային լարման տատանումները.....	49
Նկ. 2.2. Տարբեր ուղիներով սինքրոազդանշանի տարածման հապաղումները...	52
Նկ. 2.3. Երկտակտ տրիգերի կառուցվածքը.....	53
Նկ. 2.4. Երկտակտի տրիգերում մետակայունության առաջացումը.....	54
Նկ. 2.5. Տրիգերի հապաղման՝ դրա ժամանակային սահմանափակումների խախտման չափից կախվածությունը.....	54

Նկ. 2.6. Մետակայունության հայտնաբերմամբ մեթոդի բլոկ-դիագրամը.....	57
Նկ. 2.7. Մետակայունության հայտնաբերմամբ համաձայնեցման մեթոդն իրականացնող համակարգի կառուցվածքը.....	57
Նկ. 2.8. Մետակայունության հայտնաբերմամբ համաձայնեցման միջոցի փականային մակարդակի սխեման.....	59
Նկ. 2.9. Առաջարկված համաձայնեցման միջոցի աշխատանքի ժամանակային դիագրամը.....	60
Նկ. 2.10. N-կարգանի ազդանշանի համաձայնեցումը.....	61
Նկ. 2.11. Սինքրոազդանշանների համապատասխանեցմամբ համաձայնեցման համակարգի սկզբունքային կառուցվածքը.....	63
Նկ. 2.12. Սինքրոազդանշանի ընդունիչը.....	64
Նկ. 2.13. Տարբեր փուլային շեղումներով clk1-ի ընդունումը.....	65
Նկ. 2.14. clk1 ազդանշանի երեք ընտրումները.....	66
Նկ. 2.15. Համապատասխանեցնող սխեմայի սկզբունքային կառուցվածքը.....	67
Նկ. 2.16. ՍԸ-ի ելքային վեկտորների հավանական տարբերակները.....	67
Նկ. 2.17. Համափուլ սինքրոազդանշանը ընտրող սխեման.....	70
Նկ. 2.18. Թվային նախագծման երթուղին.....	73
Նկ. 2.19. Մետակայուն միջմիացման ժամանակային սահմանափակումները.....	74
Նկ. 2.20. Բազմակարգ ազդանշանի համաձայնեցումը.....	77
Նկ. 2.21. Համաձայնեցման արձանագրություն իրականացնող համաձայնեցուցիչը.....	78
Նկ. 2.22. Ձեռքսեղմման արձանագրության՝ ուղու մեծ հապաղումներով պայմանավորված խափանումը.....	79
Նկ. 2.23. Մետակայունության հայտնաբերմամբ և բնականոն վիճակի վերականգնմամբ ՀՄ-ի ժամանակային դիագրամը.....	81
Նկ. 2.24. Հապաղման՝ հաճախականությունից ունեցած կախվածությունը.....	85
Նկ. 2.25. Էներգասպառման՝ հաճախականությունից ունեցած կախվածությունը.....	85
Նկ. 3.1. SmartSync ծրագրի աշխատանքը նկարագրող բլոկ-դիագրամը.....	88
Նկ. 3.2. ՍՍՀԲ-ի աշխատանքի ալգորիթմը.....	90

Նկ. 3.3. ClockSignal դասի օբյեկտները.....	90
Նկ. 3.4. SynchronizedSignal դասի օբյեկտները.....	91
Նկ. 3.5. ClockSignal և SynchronizedSignal դասերի օբյեկտների փոխկապակցվածությունը.....	92
Նկ. 3.6. ՀՄԸԲ-ի աշխատանքի սկզբունքը.....	92
Նկ. 3.7. ՀՄՆԲ-ի աշխատանքի սկզբունքը.....	94
Նկ. 3.8. SmartSync ծրագրի գրաֆիկական ինտերֆեյսը.....	96
Նկ. 3.9. Ֆայլ մենյուն.....	97
Նկ. 3.10. Ֆայլերի ներմուծման մենյուն.....	97
Նկ. 3.11. Ֆայլի ներմուծման պատուհանը.....	97
Նկ. 3.12. Համաձայնեցման մեթոդների կոնֆիգուրացիաների պատուհանը.....	98
Նկ. 3.13. Սինքրոտիզացիաների կոնֆիգուրացիաների պատուհանը.....	98
Նկ. 3.14. ՌՓՄ կամ ՓՄ նկարագրության ներմուծումը.....	99
Նկ. 3.15. Նախագծի կարգավիճակի ցուցիչը.....	99
Նկ. 3.16. Սխեմայի վերլուծության ընթացքի ցուցիչը.....	100
Նկ. 3.17. Համաձայնեցման միջոցների ներդրմանը պատրաստ նախագիծը.....	100
Նկ. 3.18. ՀՄ-երի ներդրման ընթացքի ցուցիչը.....	101
Նկ. 3.19. Համաձայնեցման ավարտից հետո SmartSync ծրագրի պատուհանը..	102
Նկ. 3.20. OpenSPARC T1 պրոցեսորի կառուցվածքը.....	107
Նկ. 3.21. SPARC միջուկի կառուցվածքը.....	108

Աղյուսակների ցանկ

Աղյուսակ 1.1. Մուտքային կարգի երկկարգանի ներկայացումը.....	12
Աղյուսակ 2.1. Տարբեր շեմային լարումների և ջերմաստիճանների դեպքում բնականոն և մետակայուն վիճակներում տրիգերի հապաղման արժեքները	55
Աղյուսակ 2.2. Տարբեր շեմային լարումների և ջերմաստիճանների դեպքում տրիգերի մետակայուն ելքի կայունացման ժամանակի արժեքները.....	55
Աղյուսակ 2.3. Բնականոն և մետակայուն վիճակների առաջացման դեպքում տրիգերային համաձայնեցուցիչների հապաղումները.....	56

Աղյուսակ 2.4. ՀՍ-ի իսկության աղյուսակը 3 ընտրումների դեպքում.....	66
Աղյուսակ 2.5. Համարժեք բուլյան արժեքով դեպքերի միավորումը.....	68
Աղյուսակ 2.6. 4 ընտրման դեպքում ՀՍ-ի իսկության աղյուսակը.....	69
Աղյուսակ 2.7. 4 ընտրումների դեպքում համընկնող բուլյան ֆունկցիաների և ՀՍ-ի ելքի համապատասխանեցումը.....	70
Աղյուսակ 2.8. Մոդելավորման արդյունքները.....	82
Աղյուսակ 2.9. Համաձայնեցման տարբեր միջոցների հապաղումները.....	83
Աղյուսակ 2.10. Հաճախություններին համապատասխանող հապաղումներն ու էներգիան.....	84
Աղյուսակ 3.1. Համաձայնեցման տարբեր մեթոդների կիրառման դեպքում սխեմայում փականների ընդհանուր քանակը.....	109
Աղյուսակ 3.2. Համաձայնեցման տարբեր մեթոդների կիրառման դեպքում սինքրոտիզմների միջև առավելագույն հապաղումները.....	109

Օգտագործված հապավումների ցանկը

ԻՍ՝ ինտեգրալ սխեմա

ԹՄ՝ թվային սխեմա

ՄՕԿ՝ մետաղ-օքսիդ-կիսահաղորդիչ

ԿՄՕԿ՝ կոմպլեմենտար մետաղ-օքսիդ-կիսահաղորդիչ

ՁԱ՝ ձեռքսեղմման արձանագրություն

ՈԻՏ՝ ուղարկող տարր

ԸՏ՝ ընդունող տարր

ԳԱՏՄ՝ գլոբալ ասինքրոն, տեղային սինքրոն

ՀՀԲ՝ Համակցային հաշվողական բլոկ

ՄԵՀ՝ մուտք-ելք հապաղում

ԼՀԴՄ՝ լարման և հաճախականության դինամիկ մասշտաբավորում

ՍԱ՝ սինքրոնազդանշան

ՈԻՍ՝ ուղարկող սինքրոտիրույթ
 ԸՍ՝ ընդունող սինքրոտիրույթ
 ՄՀ՝ մակարդակային համաձայնեցուցիչ
 ՃԳ՝ ճակատ գրանցող
 ՃԳՀ՝ ճակատ գրանցող համաձայնեցուցիչ
 ԻԳ՝ իմպուլս գեներացնող
 ՍԸԻ՝ սինքրոագդանշանի ընդհատմամբ ինտերֆեյս
 ՍՀԻ՝ սահմանային համաձայնեցմամբ ինտերֆեյս
 ՀՀԻ՝ համաձայնեցման հերթահենք ինտերֆեյս
 ԳԼՋ՝ գործընթաց, լարում, ջերմաստիճան
 ՀՍ՝ համաձայնեցնող սխեմա
 ՄՍ՝ մտավոր սեփականություն
 ՀԻՍ՝ Համակարգային ինտեգրալ սխեմա
 ԲՄԱ՝ բաց միջուկի արձանագրություն
 ՀԻ՝ համաձայնեցման ինտերֆեյս
 ՀԻՍ՝ համակարգային ինտեգրալ սխեմա
 ՎԱ՝ վերջավոր ավտոմատ
 ԱԳՇ՝ արտադրական գործընթացի շեղումները
 ԱԸ՝ ազդանշանի ընդունիչ
 ՀաԲ՝ համեմատող համակցային բլոկ
 ՓՇ՝ փուլային շեղիչ
 ՍԸ՝ սինքրոագդանշանի ընդունիչը
 ՀՍ՝ համապատասխանեցնող սխեմա
 ՌՓՄ՝ ռեգիստրային փոխանցման մակարդակ
 ՖՆ՝ ֆիզիկական նախագիծ

ՖՍ՝ ֆունկցիոնալ ստուգում
 ՀՍտ՝ համաձայնեցման ստուգում
 ՍԺՎ՝ ստատիկ ժամանակային վերլուծություն
 ԿԺ՝ կայունացման ժամանակ
 ԶՀ՝ զուգահեռ համաձայնեցուցիչ
 ՀԱԻՀ՝ համաձայնեցման արձանագրություն իրականացնող համաձայնեցուցիչ
 ՀՄ՝ համաձայնեցման միջոց
 ՍՍՀԲ՝ սինքրոտիրույթներ ու սինքրոազդանշաններ հայտնաբերող բլոկ
 ՎԲ՝ վերլուծող բլոկ
 ՀՄԼԲ՝ համաձայնեցման միջոցների ներդրման բլոկ
 ՀՄԿՖ՝ համաձայնեցման միջոցների կոնֆիգուրացիոն ֆայլը
 ՀՍԿՖ՝ համակարգի սինքրոտիրույթների կոնֆիգուրացիոն ֆայլը
 ՀՄՎ՝ Հրահանգի վերձանման միավորը
 ԻՄ՝ իրականացման միավոր
 ԲՏՄ՝ Բեռնում/տեղադրման միավորը
 ԸՏՄ՝ Ընդհատումների տրամաբանական միավորը
 ՀՂՄ՝ Հիշողության ղեկավարման միավորը