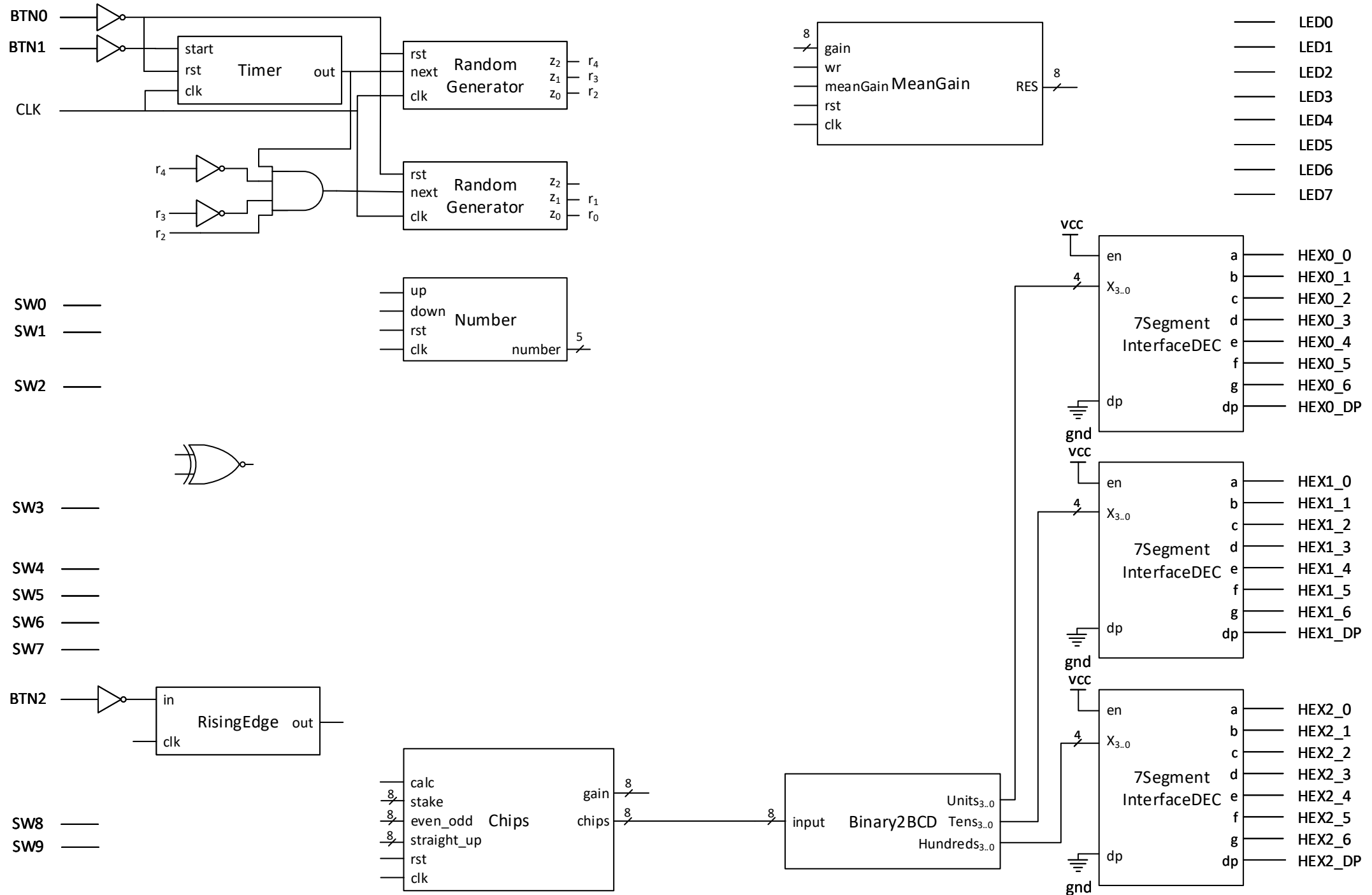


Практикум из основа рачунарске технике – домаћи задатак 2019/2020

- пета лабораторијска вежба -

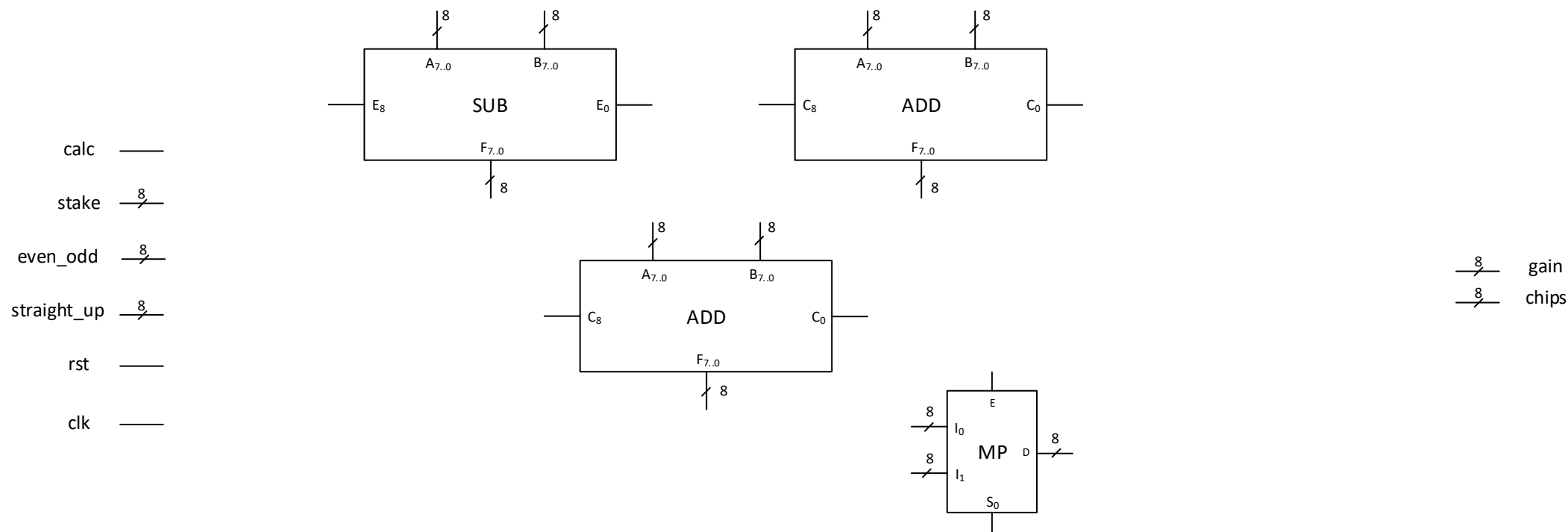
Задатак

Потребно је допунити дату шему која симулира један рад рулета машине са цилиндром од 32 броја **0₁₀-31₁₀** (неки делови шеме реализују исте функционалности са четврте лабораторијске вежбе, док се неке функционалности разликују). На основу улога играча, стања играча, броја који је пао, као и одабира опција на које је играч играо, потребно је да се на седмосегментним дисплејима прикаже ново стање играча (број жетона). Ново стање играча представља стање играча након добитка или губитка. На седмосегментним дисплејима **HEX2**, **HEX1** и **HEX0** приказује се број жетона играча. Прекидачима **SW4-SW7** играч бира један од могућих предефинисаних улога и то **1₁₀**, **2₁₀**, **4₁₀** или **6₁₀** жетона, респективно. Да би улагање било исправно, максимално један прекидач може да буде у позицији један (чиме се бира одређен улог), док остали прекидачи морају да буду у позицији 0. У случају да су сви прекидачи у позицији 0, онда се сматра да играч прескаче спин (бацање лоптице), па је улог једнак нули. Такође, ако улог премашује стање играча и онда је улог једнак нули. На рулету постоје две опције које играч може да игра. Прва опција је да играч погоди број који ће да падне у наредном спину, док је друга опција да играч погоди да ли ће број у наредном спину бити паран или непаран. У случају да је играч играо погађање броја и том приликом погоди број који падне, његов добитак представља тридесетодвоструки улог (улог * 32), у случају да је промашио добитак је једнак нули. Играч бира број на основу стања прекидача **SW0** и **SW1**. Прелазак прекидача **SW0** из позиције 0 у позицију 1 увећава жељени број за један, док прелазак прекидача **SW1** из позиције 0 у позицију 1 смањује жељени број за један. Одабрани број се приказује у бинарном систему на **LED4-LED0** (**LED4** представља бит највеће тежине). Описану функционалност бирања жељеног броја реализује модул **Number** (реализован модул са четврте лабораторијске вежбе). У случају да је играч играо погађање парност броја и том приликом погоди парност броја који падне, његов добитак представља дупли улог, у случају да је промашио добитак је једнак нули. Играч бира парност на основу стања прекидача **SW3** (позиција 0 одговара предвиђању парног броја, а позиција 1 одговара предвиђању непарног броја). Играч може да игра или прву, или другу или обе опције у зависности од стања прекидача **SW8** и **SW9**. Прекидач **SW8** у позицији један означава да играч игра опцију погађања броја, а прекидач **SW9** у позицији један означава да игра опцију погађања парности броја. Ако су оба прекидача у позицији један играч игра обе опције при чему је укупан добитак збир оба добитка (улог је исти за обе опције). У случају да је број освојених жетона једнак 0, онда је потребно смањити стање играча за уложени број жетона. Функционалност рачунања стања играча реализује модул **Chips**. Играч покреће спин на сваки притисак дугмета **BTN1** чиме рулет машина генерише насумичан број након насумичног броја тактова. Број се генерише на основу предефинисане секвенце из опсега **0₁₀-31₁₀**. Функционалност генерисања насумичног броја тактова реализује модул **Timer**. Добитак је потребно запамтити у меморији. Играч може да израчуна аритметичку средину добитка који је остварио у последњих шеснаест игри притиском на дугме **BTN2**. Средњи добитак се приказује на **LED7-LED0** (**LED7** представља бит највеће тежине). Прекидач **SW2** у позицији 1 приказује аритметичка средина добитка, а у позицији 0 изабрани број на лед диодама. Функционалност чувања добитка у меморији и проналажења аритметичке средине добитка реализује модул **MeanGain** (сматрати да се неће покренути игра и проналажење аритметичке средине добитка у исто време). Рулет машина се ресетује притиском дугмета **BTN0** чиме се ресетујују све секвенцијалне мреже на одговарајућу вредност. Сматрати да играч прво ресетује рулет машину пре почетка њеног коришћења. На датој шеми су модули **RisingEdge**, **RandomGenerator**, **Binary2BCD** и **7SegmentInterfaceDEC** са којима се студент упознао на претходним лабораторијским вежбама. При реализацији главне шеме и модула **Timer**, **MeanGain**, **Chips** дозвољено је коришћење свих стандардних комбинационих, флип-флопова, регистара са стандардним контролним улазима, И, ИЛИ и НЕ кола, **RisingEdge** компоненте као и **FallingEdge** компоненте. Обратити пажњу да су вредности регистара ажурају у адекватним тренутцима тј. да се у њих уписују добре вредности. Није потребно обраћати пажњу на могућа прекорачења вредности који се добијају као последица сабирања. Дозвољено је именовати излазе логичких елемената и користи их на другим деловима структурне шеме. Дозвољено је спајати или издвајати сигнале и тако формирати нове групе сигнала. Студент је дужан да црта прегледно коришћењем графитне оловке. Урађен домаћи задатак на формулару донети у термину пете лабораторијске вежбе.



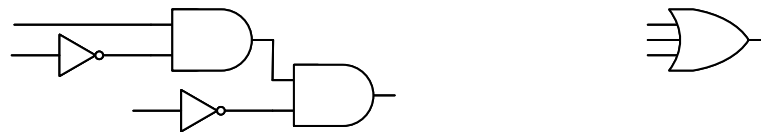
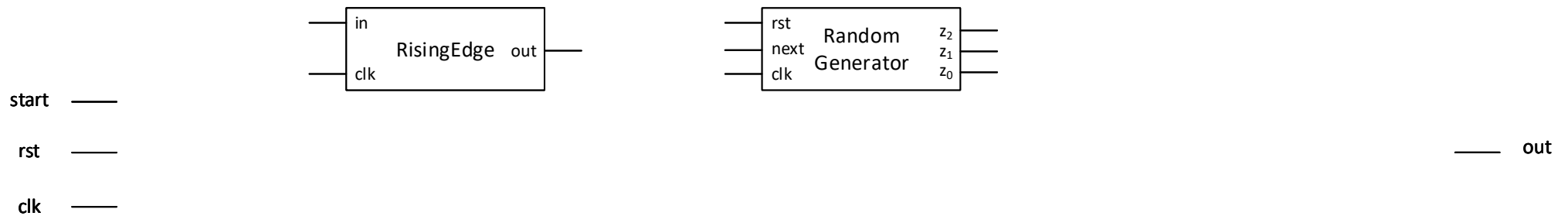
Задатак

Потребно је допунити дату шему за реализацију модула ***Chips***. Овај модул представља секвенцијалну мрежу Муровог типа која има улазне сигнале ***calc***, ***stake***, ***even_odd***, ***straight_up***, ***rst*** и ***clk*** (сигнал такта) и два излазна осмобитна сигнала ***chips*** и ***gain***. Овај модул има могућност ажурирања тренутног стања играча. Ажурирање се задаје активном вредношћу сигнала ***calc***. Улог играча, добитак играча при погађању парности и добитак играча при погађању вредности броја се задаје улазним осмобитним сигналимa ***stake***, ***even_odd*** и ***straight_up*** (ради се о оствареним, а не о потенцијалним добицима), респективно. При ажурирању стања треба узети у обзир све наведене вредности. Излазни осмобитни сигнал ***chips*** представља тренутно стање играча, а излазни осмобитни сигнал ***gain*** представља укупни остварени добитак. Улазни сигнал ***rst*** служи за ресетовање мреже, чијем се активном вредношћу мрежа доводи у подразумевано стање (почетно стање играча је ***100*₁₀** жетона) неvezано за остале улазне сигнале (***rst*** је вишег приоритета).



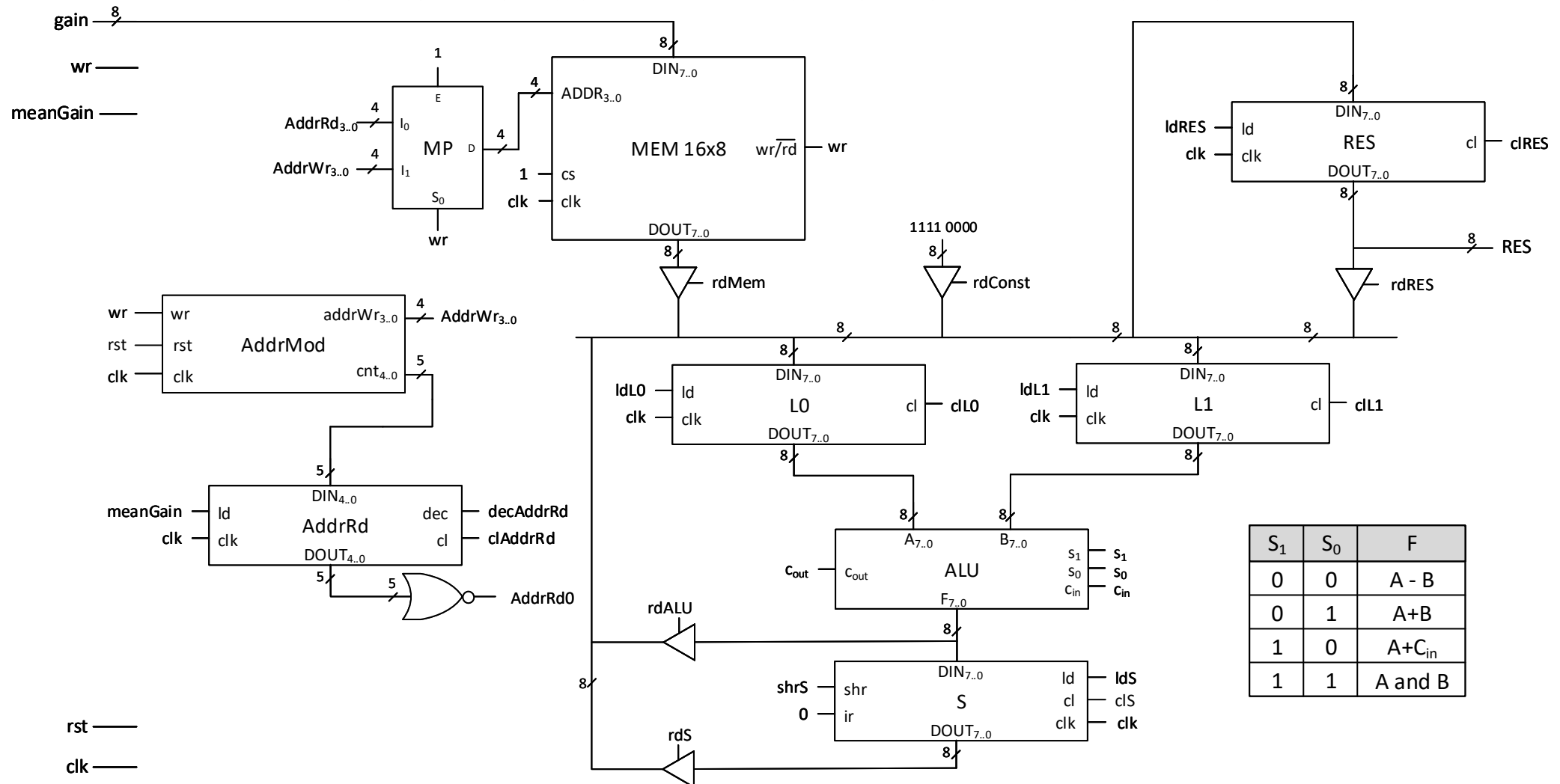
Задатак

Потребно је допунити дату шему за реализацију модула **Timer**. Овај модул представља секвенцијалну мрежу Муровог типа која има улазне сигнале **start**, **rst** и **clk** (сигнал такта) и један излазни једнобитни сигнал **out**. Овај модул има могућност активирања (активна вредност је 1) излазног сигнала **out** у трајању насумичног броја тактова (**z** тактова). Активном вредношћу улазног сигнала **start** у трајању једне периоде сигнала такта је сигнал мрежи да активира излазни сигнал **out** и држи овај сигнал активан **z** број тактова. Број **z** одеђује генератор случајних бројева. Обезбедити да једном активирањем **Timer** не може да се прекине поновним активирањем сигнала **start**. Улазни сигнал **rst** служи за ресетовање мреже, чиме се активним вредношћу мреже доводи у подразумевано стање (сви регистри се поставе на вредност нула) неvezано за остале улазне сигнале (**rst** је највишег приоритета). Улазни сигнал **next** генератора случајних бројева треба да се активира сваки пут када се активира **Timer**.



Задатак

Дата је шема операционе јединице модула **MeanGain** која има могућност реализација две операције. Прва операција је учитавање добитка у меморијски модул **mem_16x8**, а други је рачунање аритметичке средине добити у последњих шеснаест одиграних игри. Потребно је нацртати дијаграм тока фазе извршења наредбе **MeanGain** која налази аритметичку средину добити у последњих шеснаест игри и резултат смешта у регистар **RES**. У случају да није одиграна ниједна игра резултат треба да буде једнак нули. **Дата шема се не сме мењати.** Нацртати дијаграм тока управљачких сигнала ове наредбе. Реализовати структурну шему управљачке јединице реализовану као стандардна секвенцијална прекидачка мрежа коришћењем Т флип-флопова код којих је логичка јединица активна вредност (потребно је предвидети у управљачкој јединици сигнал **rst** који активном вредношћу поставља мрежу у следећем такту у корак T_0). **Напомена:** ради лакше реализације операције аритметичке средине, суму добитака је потребно увек поделити са шеснаест без обзира на број сабирака у суми. Такође није потребно разматрати могућа прекорачења приликом сабирања добитака.

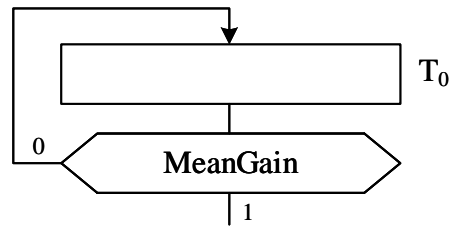


Задатак (модул MeanGain, наставак)

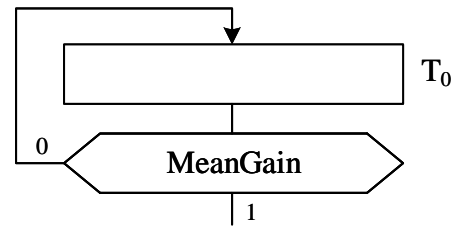
Модул **AddrMod** (овај модул је имплементиран и можете га преузети са *Moodle* платформе) на основу улазног сигнала **wr** генерише адресу **addrWr_{3..0}** и број **cnt_{4..0}**. Овај модул има могућност ажурирања адресе за упис у меморију. На излазним линијама **addrWr_{3..0}** се налази адреса добитка који је најдуже био у меморији или слободна локација и самим тим на ову локацију може да се упише следећи добитак. Активним улазним сигналом **wr** се ажурира адреса за упис. На излазним линијама **cnt_{4..0}** се налази број елемената у меморији.

Задатак

Дијаграм тока микроперација



Дијаграм тока управљачких сигнала



Управљачка јединица

Практикум из основа рачунарске технике – домаћи задатак 2019/2020

- пета лабораторијска вежба -

Студент:

Индекс: _____ / _____
 гггг 6666