

INFÖR LABORATION 2

Följande uppgifter ur kursboken ska vara utförda som förberedelse för laborationen. Du ska på begäran av laborationshandledare redogöra för dessa.

Uppgifter i kursboken						
6.5						
6.6						
6.1	(6.8)	7.3	7.10	(7.6)	7.12	(7.24)
6.2	6.9	7.4	7.11	(7.9)	7.20	7.25
6.3					7.22	7.26
6.4						

Uppgifter inom parentes är inte strikt obligatoriska, men om man gjort dem blir arbetet med de obligatoriska uppgifterna lättare.

Nedanstående hemuppgifter i detta PM som ska vara utförda innan laborationen påbörjas.

Hemuppgifter	
2.A	2.B

OBS! Hemuppgift 2.B är utspridd över flera deluppgifter. Den består i att fylla i RTN och styrsignaler i tabellerna som finns i deluppgifterna 2.2.2, 2.4.2, 2.4.3 och 2.4.5 i detta PM.

LABORATION 2

INLEDNING

Denna laboration består av fem deluppgifter. I uppgift 1 kommer du att undersöka en JK-vippa, samt bygga en räknare med hjälp av två JK-vippor. För den här uppgiften ska du använda dig av utrustning du redan har bekantat dig med under laboration 1, ett chip (SN74HC112) samt ett datablad som innehåller specifikationerna för chippet.

Under uppgifterna 2, 3, 4 och 5 kommer du att bekanta dig med LU2-modulen. Du ska först bygga en enkel dataväg och undersöka hur ett värde kan överföras mellan registerna, sen ska du koppla ihop datavägen med ALU:n för att utföra enkla operationer. Till slut kommer du att utforska hur minne fungerar och hur läs- och skrivcykler fungerar.

Om du inte kan svara på följdfrågor, gå gärna vidare med de andra uppgifterna och be en handledare att hjälpa dig.

LABORATIONSUPPGIFT 2.1 VIPPOR OCH RÄKNARE

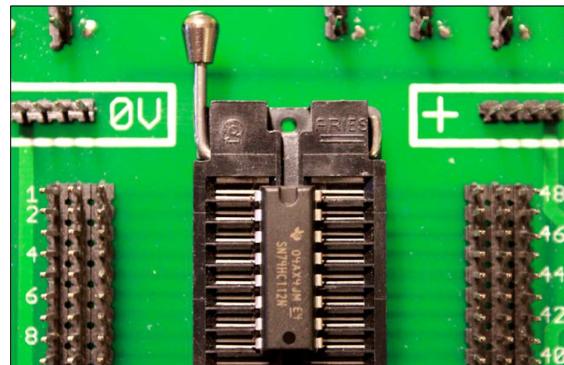
Du förbereder dig för denna laborationsuppgift genom att studera kursbokens kapitel 6. Undersök också noga databladet för chippet SN74HC112. Varje chip består av 2 JK-vippor. Lägg märke till att vippa 1 har de flesta av sina pinnar på vänstersidan av chippet, medan vippa 2 har de flesta av sina pinnar på högersidan.

Medan du följer nedanstående instruktioner, se till att strömtillförseln till LU1-modulen är avstängd.

KOPPLA UPP SN74HC112

Lyft upp hävarmen på sockeln på LU1-modulen och placera chippet som i figuren nedan, det vill säga med den halvcirkelformade markeringen uppåt och så att chippets ben längst upp till vänster hamnar rätt för stifttrad 1. Då ska chippets ben längst upp till höger hamna rätt för stifttrad 48.

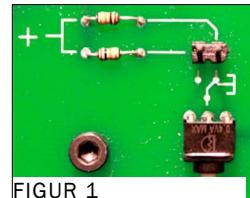
Fäll ner hävarmen och försäkra dig om att chippet sitter stabilt och ligger med kroppen rakt över fåran i mitten.



CHIPPET SN74HC112 I SIN SOCKEL

DELUPPGIFT 2.1.1: STABILISERA KLOCKPULSEN

Längst ner på LU1-modulen finns en återfjädrande knapp som ska användas för att ge klockpulser till vipporna. Eftersom knappen är mekanisk uppstår mekaniskt studs. Signalen behöver därför stabiliseras med hjälp av en SR-latch, som i Figur 2. I fall signalen inte stabiliseras kan du uppleva att varje knapptryckning resulterar i flera klockpulser.



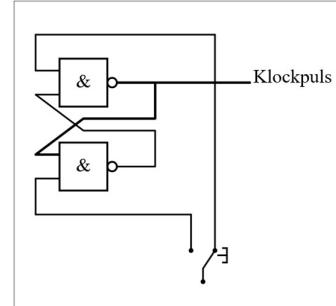
FIGUR 1
KLOCKPULSKNAPP

Nu ska du bygga en SR-latch och koppla dess utgång ("klockpuls") till en lysdiod. Använd de två NAND-grindarna närmast knappen.

Slå på strömmen till LU1-modulen.

Vilket logikvärdet visas på lysdioden när du trycker in knappen?

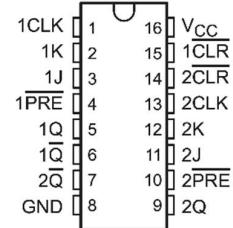
Vilket logikvärdet visas på lysdioden när du släpper knappen?



DELUPPGIFT 2.1.2: JK-VIPPA

För denna uppgift ska du undersöka en JK-vippa med asynkrona S- och R-ingångar, liknande dem du studerat i kursbokens uppgift 6.1.

OBS! Pinnarna på högersidan av chippet SN74HC112, som i databladet markerats med nummer 9 till 16, motsvarar stiftraderna 41 till 48 på sockeln på LU1-modulen. Lägg också märke till att PRE motsvarar S och CLR motsvarar R. Dessa signaler är "aktiv lågt".



- Låt din SR-latch för klockpuls vara inkopplad, men koppla bort strömtillförseln från modulen.
- Koppla upp vippa 1, så att $\overline{1PRE}$, $\overline{1CLR}$, $1J$ och $1K$ -signalerna går till switcharna 3, 2, 1 och 0.
- Koppla pinnen $1CLK$ så att den får signal från SR-latchen du byggde, alltså koppla den till ett av de lediga stiftens på lysdioden som visar latchens logikvärdet.
- Koppla utgången $1Q$ till en annan av lysdioderna på LU1.
- Koppla till slut pinnarna GND till OV och V_{CC} till $+V$, så att chippet drivs med ström.
- Ställ nu samtliga switchar i läge 1 och koppla på strömtillförseln till modulen.

UNDERSÖKNING AV INGÅNGAR

Undersök först insignalerna \bar{S} och \bar{R} , alltså $\overline{1PRE}$ och $\overline{1CLR}$.

Ställ \bar{S} till aktiv (switch 3 i läge 0) och \bar{R} till inaktiv (switch 2 i läge 1). Vad händer? _____

Ställ \bar{S} till inaktiv och \bar{R} till aktiv. Vad händer? _____

Behöver du ge en klockpuls för att vippans utsignal Q ska förändras? _____

Undersök nu insignalerna J och K och fyll i kolumnen Q^+ i nedanstående funktionstabell. Låt både \bar{S} och \bar{R} vara inaktiva under tiden, så att de inte påverkar signalerna J och K.

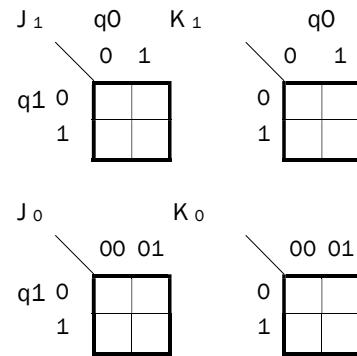
J	K	Q^+
0	0	
0	1	
1	0	
1	1	

HEMUPPGIFT 2.A

Implementera en 2-bitarsräknare som matar ut följande sekvens: 00, 01, 10, 11, 00,...

Analysera nu räknesekvenserna och fyll i följande tabeller.

q1	q0	q1 ⁺	q0 ⁺	J ₁	K ₁	J ₀	K ₀



Bestäm uttryck för J- och K-ingångarna.

J ₁ =
K ₁ =
J ₀ =
K ₀ =

Rita en skiss över hur du tänker genomföra sekvensen.

DELUPPGIFT 2.1.3: KOPPLA UPP RÄKNARE

Nu ska du koppla upp din räknare. Slå först av strömmen och ta bort alla kablar, förutom dem som ingår i SR-latchen som ska användas för att ge klockpuls. Välj också vilken av chippets två vippor som ska realisera din q_1 - respektive q_0 -vippa.

- Koppla in klockpulsen genom att ansluta en kabel till insignalerna 1CLK och 2CLK från lediga pinnar på lysdioden ansluten till SR-latchen.
- Båda vippornas R-insignaler måste nu vara inaktiva under hela utförandet. Alltså måste insignalerna $\overline{1\text{CLR}}$ och $\overline{2\text{CLR}}$ kopplas till positiv potential.
- Koppla båda vippornas S-insignaler, alltså $\overline{1\text{PRE}}$ och $\overline{2\text{PRE}}$ till switch nummer 0, så att vipporna kan ges ett initialvärde.
- Koppla också GND till jord (OV) och V_{cc} till positiv potential ($+V$).
- Koppla båda vippornas utsignaler $1Q$ och $2Q$ till lysdioder.
- Slå på strömmen och ställ switch 0 i läge 0 (aktiv S-ingång).
- Verifiera att vippornas tillstånd är 11.
- Ställ switch 0 i läge 1 (inaktiv S-ingång).
- Stega igenom din räknares sekvens genom att ge upprepade klockpulser.

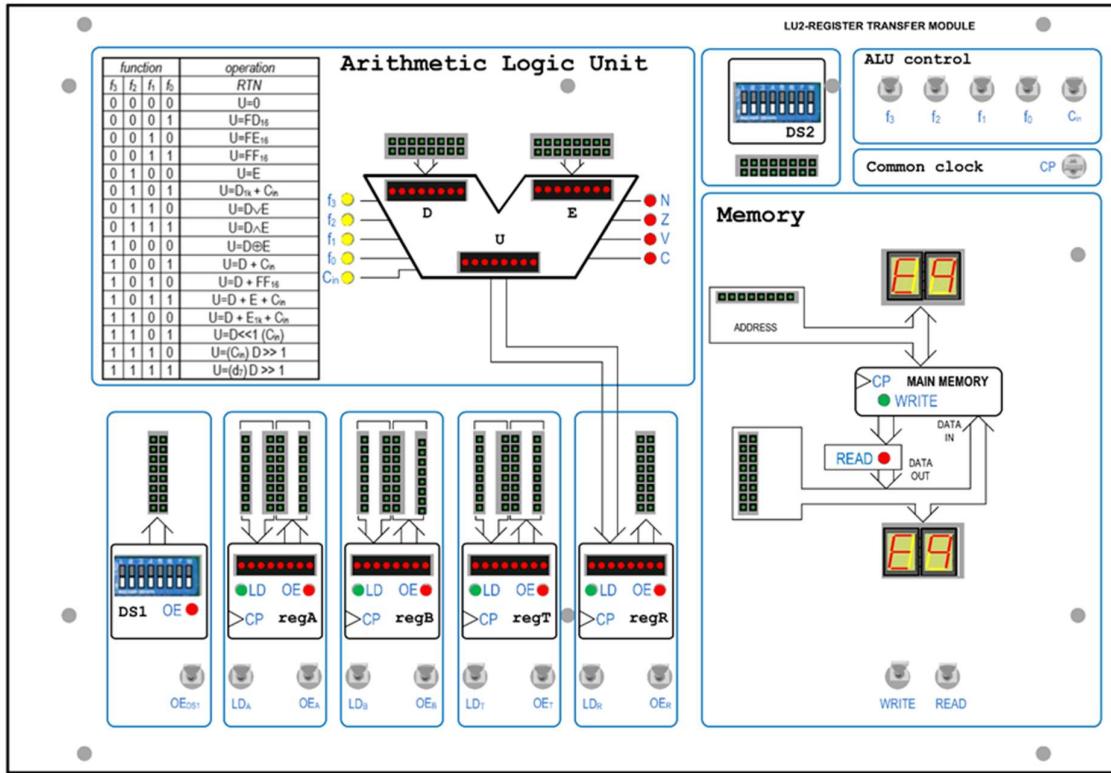
Ange utsekvenserna för räknaren:

Kontrollera att din sekvens stämmer med specifikationerna. Innan du går vidare, be en handledare verifiera din lösning.

Glöm inte att sedan slå ifrån strömmen, avlägsna alla kablar och placera allt material i ordning, innan du går vidare med nästa laborationsuppgift.

BESKRIVNING AV LABORATIONSMASKINEN

LU2-REGISTER TRANSFER MODULE



LU2 består av fyra olika sorters komponenter:

- Data Source-moduler (DS1, DS2)
- 8-bitars register (regA, regB, regT och regR)
- Arithmetic logic unit (ALU)
- Minnesmodul

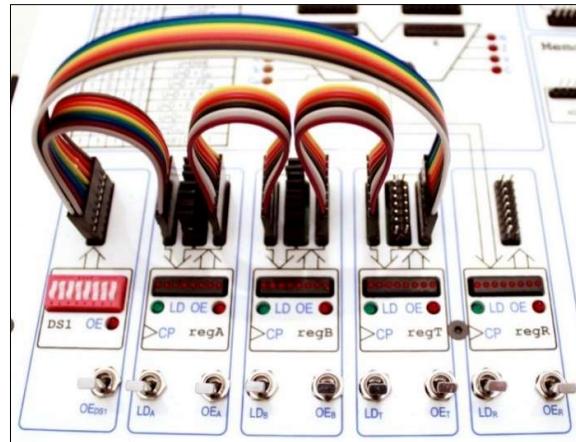
LABORATIONSUPPGIFT 2.2: DATAVÄGEN

För denna uppgift ska du undersöka hur flera register kan kopplas samman för att konstruera en enkel dataväg, liknande den du har sett i kursbokens kapitel 7.

Vi ska använda oss av:

- Data Source (DS1)
- Registren A, B och T, samtliga hopkopplade med 8-poliga kablar, som i Figur 3.

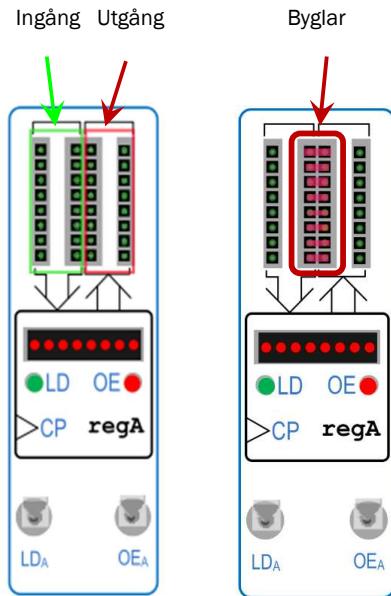
Samma dataväg används vid läsning från och skrivning till register. Varje register har en ingång och en utgång, som visas i Figur 4. För att kunna skapa en obruten dataväg, måste alla stift vara sammankopplade. Genom att koppla ihop ingång och utgång på ett register med byglar, som visas i Figur 5, kan vi undvika att använda ett stort antal kablar.



FIGUR 3 ENKEL DATAVÄG

Vi ska nu lägga till följande kopplingar för att bygga en dataväg:

- Kontrollera att regA har byglar över terminalerna i mitten.
- Anslut en kort kabel mellan DS1 och ingången på regA.
- Kontrollera att regB är byglad på samma sätt som regA.
- Anslut en kort kabel mellan utgång på regA och ingång på regB.
- Anslut en kort kabel mellan utgång på regB och ingång på regT. NOTERA ATT REGT INTE SKA HA IN- OCH UTGÅNGAR BYGLADE!
- Anslut en lång kabel mellan den lediga raden i DS1 och utgång på regT.



FIGUR 4
REGISTER-
MODULEN OCH
DESS IN- OCH
UTGÅNGAR.

FIGUR 5
BYGLAD
REGISTER-
MODUL

OBS! För flera registerutgångar som är kopplade till en gemensam buss är det viktigt att kunna försätta dem i högimpedanstillstånd så att endast ett register i taget kan tillåtas "driva" bussen.

Att låta flera register samtidigt driva databussen kan, i ogynnsamma fall, resultera i att kretsarna förstörs och utrustningen blir oanvändbar.

Av denna anledning har LU2 försetts med en kontrollfunktion som gör att, om två OE-signaler aktiveras samtidigt kommer dessa att blinka. Observera att detta **kan, men inte måste**, vara en felkoppling. Om du har kopplat registerutgångarna till en gemensam anslutning ("buss") ska du då omedelbart deaktivera de OE-utgångar som orsakar konflikten.

Prova att aktivera OE_{DS1} tillsammans med någon eller några av de andra styrsignalerna för att "driva" bussen (OE_A , OE_B , OE_R). Om dioden för OE blinkar någonstans, slå av OE.

DELUPPGIFT 2.2.1

Kontrollera att alla switchar är i avstängt läge och observera innehållet i samtliga register. Aktivera OE_{DS1} . Ställ in värdet 00001111_2 på DS1.

Ändrades något registerinnehåll (ja, vilket/nej)? _____

Aktivera nu styrsignalerna LD_A och LD_B och ge en klocksignal. Ändrades innehållet i något register (ja, vilket/nej)? _____

Vilka slutsatser kan du dra? _____

DELUPPGIFT 2.2.2 (DEL AV HEMUPPGIFT 2.B)

Du ska nu undersöka hur registren A och B kan byta innehåll med varandra (swap), med en serie av operationer. För att kunna åstadkomma detta krävs ett tredje register för temporär lagring.

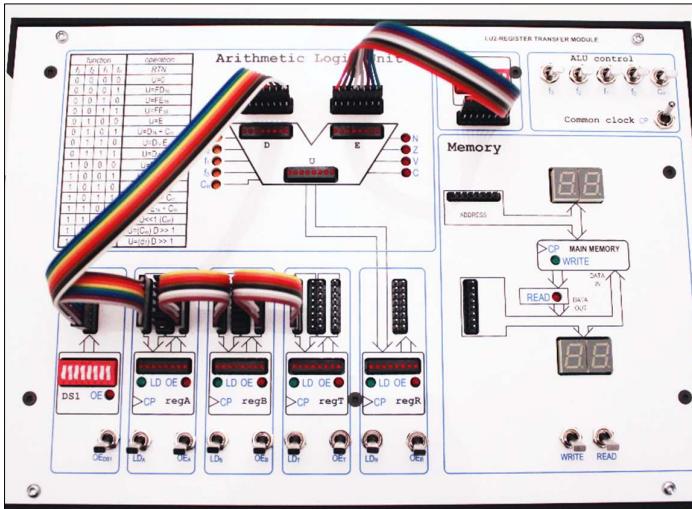
Börja med att tilldela följande värden till registren: 81_{16} till register A, $3C_{16}$ till register B.

Komplettera sedan följande tabell med RTN-beskrivning, samt de nödvändiga styrsignalerna och utför styrsekvenserna.

Steg	RTN	OE_A	OE_B	OE_T	OE_R	LD_A	LD_B	LD_T	LD_R
1									
2									
3									

Vad är värdet på register A efter operationen? _____ Register B? _____

LABORATIONSUPPGIFT 2.3 UNDERSÖKNING AV ALU



Nu ska vi undersöka hur en ALU fungerar. För den här uppgiften ska vi använda oss av:

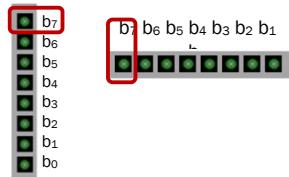
- Data Source. Två moduler (DS1, DS2).
- ALU. ALU-modulen har två 8-bitars dataingångar, D och E, samt en 8-bitars utgång U. Flaggorna (N, Z, V, C) visas på ALU:ns högersida.

Flytta kabeln från utgången på register T till ALU(D), samt koppla en ny kabel mellan DS2 och ALU(E). Kontrollera att OE_{DS1} är aktiv.

OBS! Den mest signifikanta biten (bit7) finns åt vänster i D- respektive E-delarna av ALU:n, samt i DS2.

För lodräta kontaktdon, finns den mest signifikanta biten *uppåt*.

Detta gäller för alla register, samt DS1.



SAMMANSTÄLLNING AV EN 8-BITARS ALU

Observera ALU-funktioner i tabellen nedan:

funktion				operation	utsignaler											
f ₃	f ₂	f ₁	f ₀	RTN	u ₇	u ₆	u ₅	u ₄	u ₃	u ₂	u ₁	u ₀	N	Z	V	C
0	0	0	0	U = 0	0	0	0	0	0	0	0	0	0	1	0	0
0	0	0	1	U = FD ₁₆	1	1	1	1	1	1	0	1	1	0	0	0
0	0	1	0	U = FE ₁₆	1	1	1	1	1	1	1	0	1	0	0	0
0	0	1	1	U = FF ₁₆	1	1	1	1	1	1	1	1	1	0	0	0
0	1	0	0	U = E	e ₇	e ₆	e ₅	e ₄	e ₃	e ₂	e ₁	e ₀	u ₇	(1)	0	0
0	1	0	1	U = D _{1k} + C _{in}												
0	1	1	0	U = D ∨ E	d ₇ ∨e ₇	d ₆ ∨e ₆	d ₅ ∨e ₅	d ₄ ∨e ₄	d ₃ ∨e ₃	d ₂ ∨e ₂	d ₁ ∨e ₁	d ₀ ∨e ₀	u ₇	(1)	0	0
0	1	1	1	U = D ∧ E	d ₇ ∧e ₇	d ₆ ∧e ₆	d ₅ ∧e ₅	d ₄ ∧e ₄	d ₃ ∧e ₃	d ₂ ∧e ₂	d ₁ ∧e ₁	d ₀ ∧e ₀	u ₇	(1)	0	0
1	0	0	0	U = D ⊕ E	d ₇ ⊕e ₇	d ₆ ⊕e ₆	d ₅ ⊕e ₅	d ₄ ⊕e ₄	d ₃ ⊕e ₃	d ₂ ⊕e ₂	d ₁ ⊕e ₁	d ₀ ⊕e ₀	u ₇	(1)	0	0
1	0	0	1	U = D + C _{in}												
1	0	1	0	U = D + FF ₁₆												
1	0	1	1	U = D + E + C _{in}												
1	1	0	0	U = D + E _{1k} + C _{in}												
1	1	0	1	U = D « 1 (C _{in})	d ₆	d ₅	d ₄	d ₃	d ₂	d ₁	d ₀	C _{in}	u ₇	(1)	(6)	d ₇
1	1	1	0	U = (C _{in}) D » 1	c _{in}	d ₇	d ₆	d ₅	d ₄	d ₃	d ₂	d ₁	u ₇	(1)	(6)	d ₀
1	1	1	1	U = (d ₇) D » 1	d ₇	d ₇	d ₆	d ₅	d ₄	d ₃	d ₂	d ₁	u ₇	(1)	0	d ₀

(1) $Z = \overline{u_7} \wedge \overline{u_6} \wedge \overline{u_5} \wedge \overline{u_4} \wedge \overline{u_3} \wedge \overline{u_2} \wedge \overline{u_1} \wedge \overline{u_0}$, dvs Z = 1 då samtliga bitar i register U är 0, Z = 0 annars.

(2) V ettställs om D = (10000000)₂, V nollställs annars.

(3) C nollställs om D = 0, C ettställs annars.

(4) $V = (\overline{u_7} \wedge d_7 \wedge e_7) \vee (u_7 \wedge \overline{d_7} \wedge \overline{e_7})$, dvs V-flaggan sätts enligt reglerna för tvåkomplementsaritmetik.

(5) $C = c_8$, dvs carry ut från additionen av de mest signifikanta bitarna.

(6) $V = C_{in} \oplus d_7$ vid högerskift, $V = d_7 \oplus d_6$ vid vänsterskift, dvs sätts till 1 om skiftet föranleder teckenbyte.

DELUPPGIFT 2.3.1

ALU-OPERATIONER

Utgå ifrån kursbokens uppgift 7.10 och betrakta följande tabell. Fyll i styrsignalerna för att utföra operationerna och använd ALU:n för att fylla i resultaten i tabellen (kolumnen Utgång U), samt besvara följdfrågorna.

Operation	ALU-funktion					Ingång D								Ingång E								Utgång U							
	f ₃	f ₂	f ₁	f ₀	C _{in}	Bin								Bin								Bin							
1 U = E																													
2 U = DVE																													
3 U = D<<1(C _{in})					1	1	1	0	1	0	1	0	1	0	1	0	1	0	1	1	0	0	0						
4 U = (C _{in})D>>1					0																								
5 U = (d ₇)D>>1																													

Behöver du ge en klockpuls för att läsa resultatet på U-utgången i ALU:n? (ja, varför / nej, varför inte)

Observera flaggorna för rad 5. Vilka flaggor blir tända och varför?

DELUPPGIFT 2.3.2

ARITMETIK I ALU:N

Utgå nu från kursbokens uppgift 7.11. Fyll i styrsignalerna för att utföra operationerna samt deras resultat på ALU(U), inklusive flaggor.

	Ingång D								Ingång E								Op	ALU		Utgång U								Flaggor			
	Dec	Bin							Dec	Bin								F	C _{in}	Bin				Dec	Dec	N	Z	V	C		
1 27	0	0	0	1	1	0	1	1	55	0	0	1	1	0	1	1	+			0	1	0	1	0	0	1	0	82	82		
2 -100									-55								-														
3 75									55								+														
4 90									-55								-														

Tolka nu bitmönster i resultaten på utgång U som tal MED TECKEN.

I vilka fall blir resultaten felaktiga? _____

I dessa fall, vilken flagga tänds och vilken information ger den om resultatet?

Tolka nu istället alla bitmönster som tal UTAN TECKEN.

I vilka fall blir resultaten felaktiga? _____

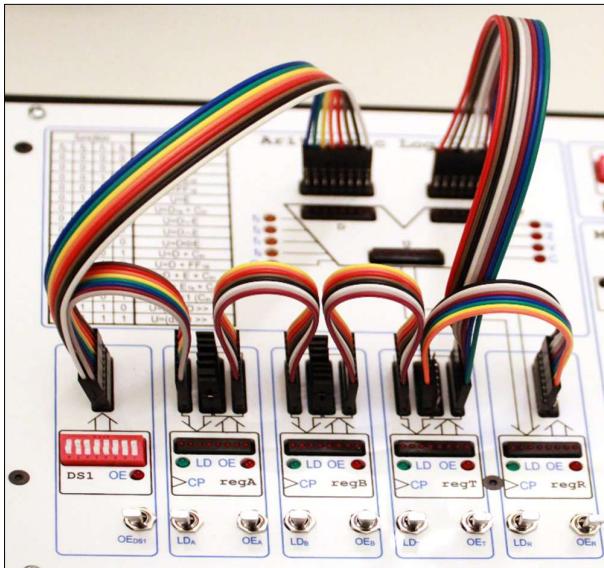
Vilken flagga tittar du för att bestämma om resultatet är rätt eller fel? _____

I vilka fall tänds N-flaggan och hur påverkar detta tolkningen av resultatet?

LABORATIONSUPPGIFT 2.4

DATAVÄG MED ALU

I den här uppgiften ska du undersöka hur ALU:n kan användas i datavägen för databearbetning. Betrakta figuren nedan och studera hur registren, DS1-modulen och ALU:n är kopplade för att skapa en sammanhängande dataväg.



FIGUR 6 DATAVÄG MED ALU

- Låt kabeln mellan DS1 och ALU(D) sitta kvar.
- Flytta kabeln från DS2 till utgången på regT.
- Anslut en kort kabel mellan utgången på regR och den lediga raden ingångspinnar på regT.

Om du är osäker på om din koppling stämmer, fråga gärna en handledare.

Notera att regT nu har sin egen utdataväg. Som en funktion av detta kan OE_T lämnas aktiv hela tiden.

DELUPPGIFT 2.4.1

ENKLA REGISTERÖVERFÖRINGAR

Ställ in värdet OF₁₆ på DS1, överför värdet till regT.

Vilken ALU-funktion kan du använda för att överföra värdet från regT till regR? _____

Vilka styrsignaler ska vara aktiva? _____

Utför den operationen.

Vilka styrsignaler ska vara aktiva för att föra över värdet från regR till regA och regB samtidigt?

Utför den operationen.

Vad innehåller nu register A och register B? _____

DELUPPGIFT 2.4.2 (DEL AV HEMUPPGIFT 2.B)

Ange en styrsignalsekvens som placeras värdet 12_{16} i register A, värdet 62_{16} i register R och värdet AA_{16} i register B. Kontrollera innehåll i respektive register innan du går vidare med nästa uppgift.

Steg	RTN	DS1	OE_{DS1}	OE_A	OE_B	OE_T	OE_R	LDA	LD_B	LD_T	LD_R	f_3	f_2	f_1	f_0	C_{in}
1	$12_{16} \rightarrow A$															
2	$62_{16} \rightarrow R$															
3	$AA_{16} \rightarrow B$															

DELUPPGIFT 2.4.3 (DEL AV HEMUPPGIFT 2.B)

Du ska nu på nytt undersöka hur registren A och B kan byta innehåll med varandra. För att kunna åstadkomma detta krävs, precis som i deluppgift 2.2.2, ett tredje register för temporär lagring. Undersök noggrant hur datavägen är byggd och hur ALU:n kan vara till hjälp för att utföra samma uppgift.

Antag att register A och register B har samma värden som efter sekvensen i föregående tabell. Komplettera tabellen nedan med den nya sekvensen av styrsignaler.

Steg	RTN	OE_{DS1}	OE_A	OE_B	OE_T	OE_R	LDA	LD_B	LD_T	LD_R	f_3	f_2	f_1	f_0	C_{in}
1															
2															
3															

Vilket värde finns nu i register A? _____ Register B? _____

DELUPPGIFT 2.4.4 (DEL AV HEMUPPGIFT 2.B)

Undersök en styrsignalsekvens som adderar innehållet i register A med innehållet i register B och lägger resultatet i A:

$$A + B \rightarrow A$$

Denna instruktion skulle kunna skrivas "ADDA B" (instruktionen ingår inte i den vanliga instruktionsuppsättningen i FLISP, utan används bara här).

Initiera register A med värdet 11010101_2 .

Initiera register B med värdet 01001111_2 .

Vilket resultat förväntar du dig? _____

Fyll i följande tabell. Komplettera också med värdena på register A, B, R och T efter varje steg:

Steg	RTN	DS1	OE _{DS1}	OE _A	OE _B	OE _T	OE _R	LDA	LD _B	LD _T	LD _R	f ₃	f ₂	f ₁	f ₀	C _{in}	A	B	R	T
1																				
2																				
3																				

Utför styrsignalsekvensen. Matchar värdet på regA det förväntade resultatet?

DELUPPGIFT 2.4.5 (DEL AV HEMUPPGIFT 2.B)

I denna uppgift vill vi beräkna medelvärdet av A och B, samt analysera beteendet hos C-flaggan.

Register A innehåller initialt värdet 92_{16} och register B innehåller värdet 94_{16} .

Vilket resultat förväntar du dig som medelvärde av dessa två tal? _____

Vilka operationer behöver du göra? _____

Ange en styrsignalsekvens som beräknar medelvärdet av A och B, fyll i RTN och styrsignaler för operationen i tabellen. Testa gärna i simulatorn i DigiFlisp.

Steg	RTN	DS1	OE _{DS1}	OE _A	OE _B	OE _T	OE _R	LDA	LD _B	LD _T	LD _R	f ₃	f ₂	f ₁	f ₀	C _{in}	A	B	R	T	C _{out}
1	$92_{16} \rightarrow A$																				
2	$94_{16} \rightarrow B$																				
3																					
4																					
5																					
6																					
7																					

VID LABORATIONSPLATSEN

Utför operationerna och fyll i värdena i registren A, B, R och T, samt C-flaggan från ALU:n i den högra delen av tabellen.

Jämför nu resultatet du har fått i dina beräkningar med det som du har fått efter styrsignalsekvensen.
Vad blev slutresultatet i register A efter sekvensen? _____

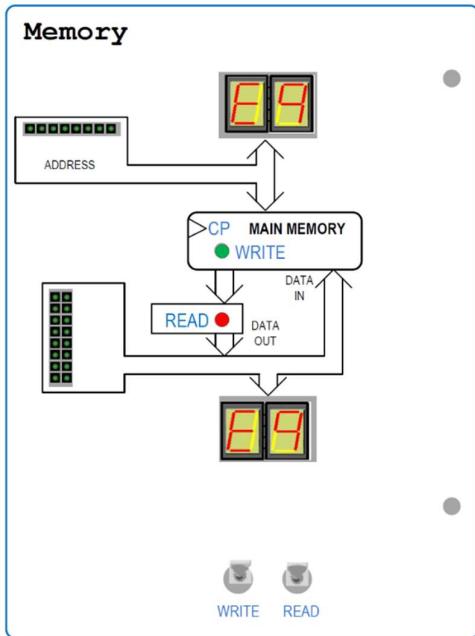
Är resultatet samma som det du beräknade tidigare?(Ja/Nej och varför)

LABORATIONSUPPGIFT 2.5

MINNESMODUL

BESKRIVNING AV MINNESMODUL OCH TERMINALANSLUTNING

I labbsystemet ingår en modul med läs- och skrivbart minne, RWM, med kapaciteten $2^8 = 256$ st 8-bitars minnesord. Blockschemat för RWM-modulen och dess frontpanel visas nedan.

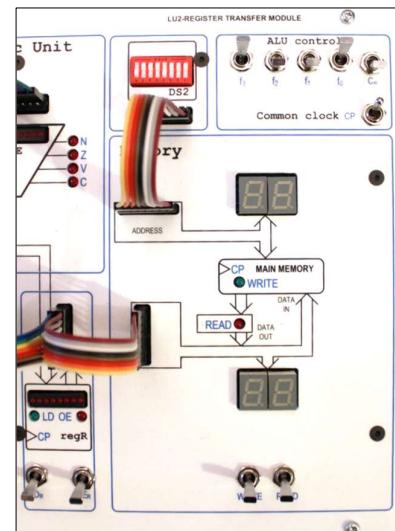


FIGUR 7 MINNESMODUL

DELUPPGIFT 2.5.1 UNDERSÖKNING AV MINNESINNEHÅLL

I denna uppgift undersöks läsning och skrivning till minnet. Du förbereder dig genom att studera kursbokens avsnitt 7.4, och utföra uppgifterna 7.24 till och med 7.28.

- Låt samtliga kablar sitta kvar på LU2.
- Koppla den lediga raden utgångspinnar på regR till dataingången på minnesmodulen, samt DS2 till addressingången.
- Ställ in adressen 10_{16} på DS2, som är ansluten till adresskontakten, samt dataordet 15_{16} på DS1, som är ansluten till datavägen. Glöm inte att aktivera styrsignalen OE_{DS1} .
- Aktivera switchen WRITE på minnesmodulen.
- Ge en klockpuls. Vad har hänt?



FIGUR 8 ANSLUTNINGAR TILL
MINNESMODULEN

DELUPPGIFT 2.5.2

Du ska nu utföra uppgifterna 7.25 och 7.26 från kursboken. Låt laborationssystemets modul DS2 motsvara simulatorns register TA. Eftersom adressen nu läggs direkt till minnets adressbuss räcker det med ett steg.

LÄSCYKELN

Fyll i laborationssystemets styrsignalsekvens för operationen (kursbokens uppgift 7.25):

$M(10_{16}) \rightarrow A$

Steg	RTN	DS1	DS2	OE _A	OE _B	OE _T	OE _R	LDA	LD _B	LD _T	LD _R	f ₃	f ₂	f ₁	f ₀	READ	WRITE
1																	

- Använd Digiflisp för att lägga in värdet 15_{16} på adress 10_{16} i minnet.
- Nollställ register A.
- Utför styrsignalsekvensen ovan.

Vilket värde innehåller nu register A? _____

SKRIVCYKELN

Fyll i laborationssystemets styrsignalsekvens för operationen kursbokens uppgift 7.26):

$A \rightarrow M(11_{16})$

Steg	RTN	DS1	DS2	OE _A	OE _B	OE _T	OE _R	LDA	LD _B	LD _T	LD _R	f ₃	f ₂	f ₁	f ₀	READ	WRITE
1																	

- Använd Digiflisp för att lägga in värdet FF_{16} på adress 11_{16} i minnet.
- Utför styrsignalsekvensen ovan. Klicka på *Update*.

Vilket värde finns nu i minnet på adress 11_{16} ? _____

DELUPPGIFT 2.5.3

Du ska nu undersöka och implementera en enkel minnesoperation, som ökar innehållet i ett minnesord med ett;

$M(Adr) + 1 \rightarrow M(Adr)$

Fyll i laborationssystemets styrsignalsekvens för operationen:

$M(11_{16}) + 1 \rightarrow M(11_{16})$

Steg	RTN	DS1	DS2	OE _A	OE _B	OE _T	OE _R	LDA	LD _B	LD _T	LD _R	f ₃	f ₂	f ₁	f ₀	C _{in}	READ	WRITE
1																		
2																		

- Låt minnesinnehållet förbli oförändrat efter föregående uppgift.
- Utför styrsignalsekvensen ovan. Klicka på *Update*.

Vilket värde finns nu i minnet på adress 11_{16} ? _____

SAMMANFATTNING

I denna laboration har du undersökt JK-vippor, byggt en enkel räknare, samt manipulerat data genom att använda en dataväg, register och en ALU.

Skriv nu en kort sammanfattning av de tre viktigaste sakerna du lärt dig under laborationen. Detta kan hjälpa dig att lägga viktiga koncept på minnet. Ta gärna hjälp av handledarna eller andra laborationsgrupper.

1. _____
2. _____
3. _____