دانشگاه کردستان
University of Kurdistan
زانکۆی کوردستان

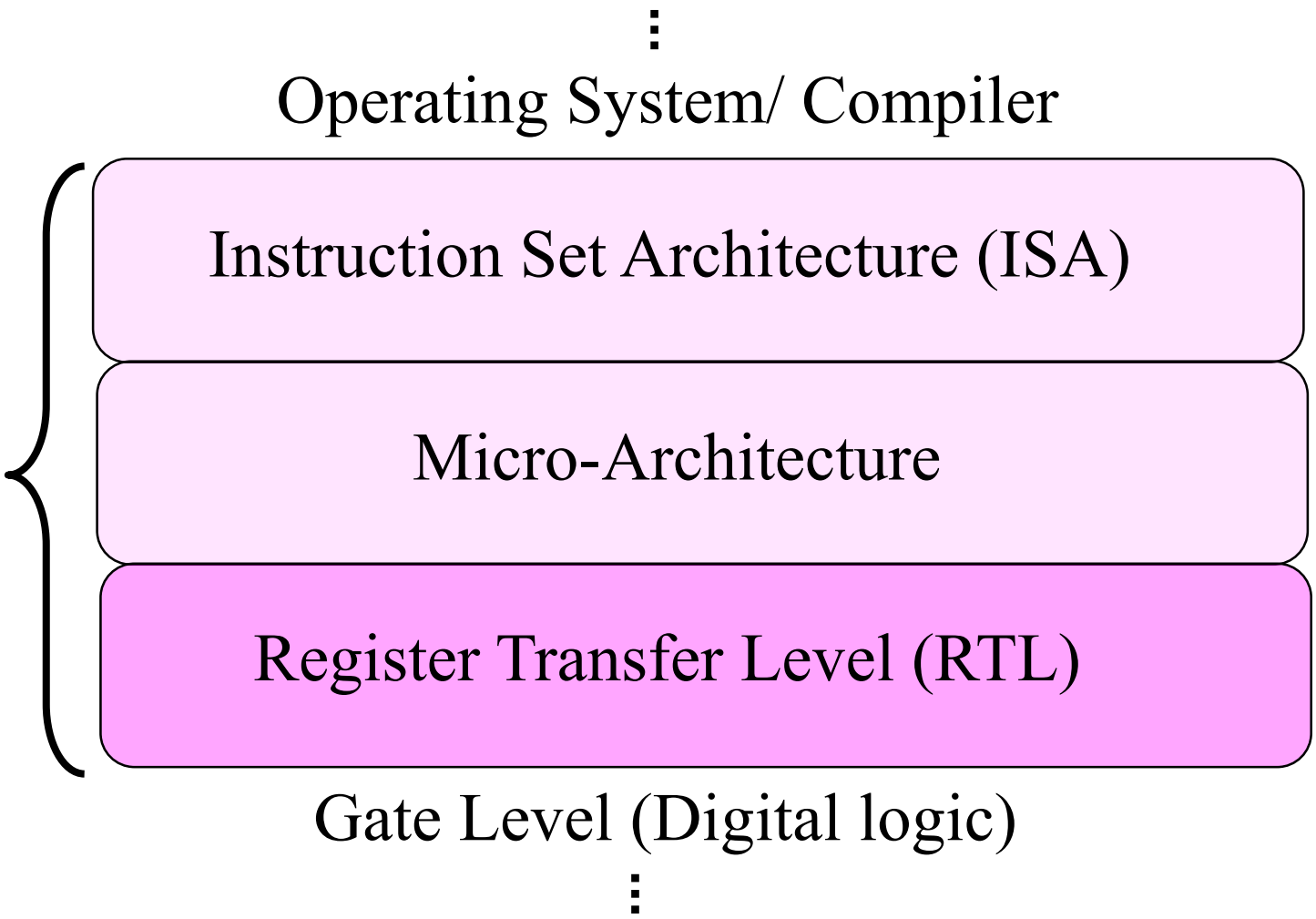
Department of Computer Engineering
University of Kurdistan

Computer Architecture

Register Transfer Language (RTL)

By: Dr. Alireza Abdollahpouri

لایه های معماری کامپیوتر

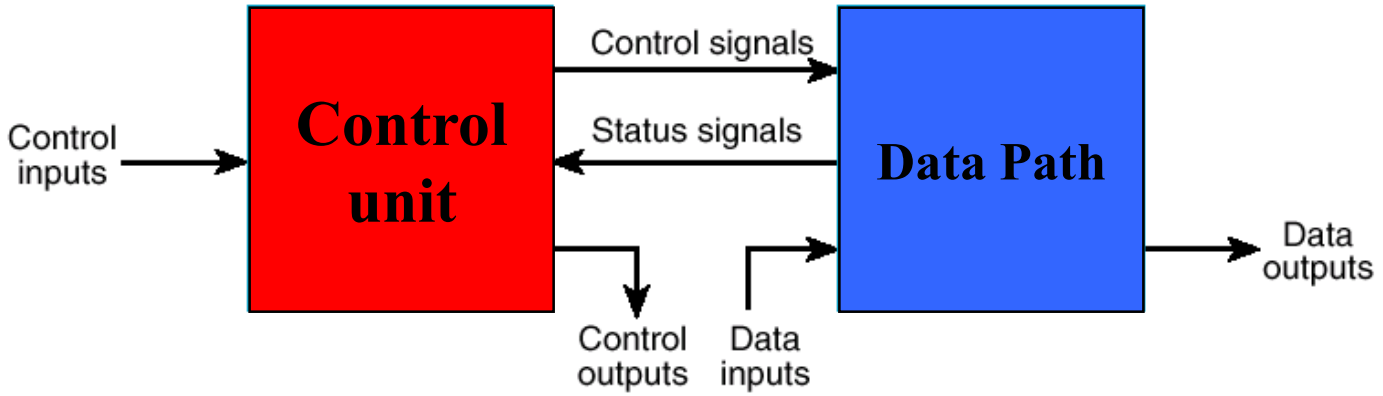


دو بخش اساسی یک پردازنده

- **Datapath (مسیر داده)**

  The ALU, registers and logic to perform operations on them

- **Control unit (واحد کنترل)**
  - **Generates signals to control datapath**
  - **Accepts status signals to perform sequencing**



RTL (Register Transfer Language) زبان انتقال رجیستر

تعریف ریزعملیات(Micro Operation)

یک عملیات پایه که روی داده های ذخیره شده روی رجیسترها در یک کلاک انجام میشود مانند: Shift , Load , Clear , Inc , Dec

برای مشخص کردن یک سیستم دیجیتال باید موارد زیر تعیین شوند:

۱- مجموعه رجیسترهای آن و توانایی های هر رجیستر

۲- دنباله میکرواپریشن های قابل اجرا بر روی اطلاعات ذخیره شده در هر رجیستر

۳- شرط کنترلی که باعث انجام مجموعه ای متوالی از میکرواپریشن ها می شود

زبان انتقال رجیستر RTL

برای سهولت در توصیف دنباله ای از عملوندها و نقل و انتقال داده بین رجیسترها (در datapath) از سیستم نمایش ویژه ای به اسم RTL استفاده میشود. استفاده از RTL سبب سهولت توصیف عملکرد کامپیوتر و راحتی کار طراحی آن میگردد.

زبان انتقال رجیستر (Register Transfer Language) RTL

- زبان انتقال رجیستر (RTL) در حقیقت توصیف اتفاقاتی است که در هر کلاک در مسیر داده (data-path) روی می دهد.

- این اتفاقات را به صورت مجموعه ای از ریزعملیاتها نشان می دهیم.

قوانین RTL

اسامی رجیسترها:

اسامی رجیسترها با حروف بزرگ که معمولا ساده شده کار رجیستر است مشخص میشوند. نظیر:

MAR=Memory Address Register **PC= Program Counter**

IR= Instruction Register **R1=(رجیستر شماره ۱)**

نمایش رجیسترها:

PC

برای نمایش یک رجیستر از یک مستطیل استفاده میشود که نام رجیستر در وسط آن نوشته میشود

نمایش بیت ها:

7	6	5	4	3	2	1	0

بیت های یک رجیستر n بیتی از 0 (سمت راست) تا n-1 (سمت چپ ترین بیت) شماره گذاری میشوند

15 ... 8	7 ... 0
PC(H)	PC(L)

گاهی یک رجیستر ۱۶ بیتی به ۲ بایت کم ارزش و پر ارزش بصورت زیر تقسیم میشود

از پرانتز و شماره بر ای مشخص کردن یک قسمت از رجیستر استفاده میشود

PC(H)= PC(8-15)

PC(L)= PC(0-7)

قوانین RTL

- انتقال اطلاعات یک رجیستر به دیگری بصورت زیر نمایش داده میشود:

$$R2 \leftarrow R1$$

عبارت فوق نشان دهنده انتقال محتویات رجیستر R1 به رجیستر R2 است. محتوی R1 بدون تغییرباقی می ماند ولی محتوی R2 برابر با محتوی R1 میشود.

لازمه سخت افزاری انتقال :

۱- موجود بودن مسیری از خروجی R1 به R2

۲- رجیستر R2 باید قابلیت بارکردن موازی (Parallel Load) داشته باشد.

- دو ریزعملیات همزمان با کاما از هم جدا می شوند:

$R1 \leftarrow R1+R2,\ \ R3 \leftarrow 0$

University of Kurdistan

تابع کنترلی

شرط انتقال:

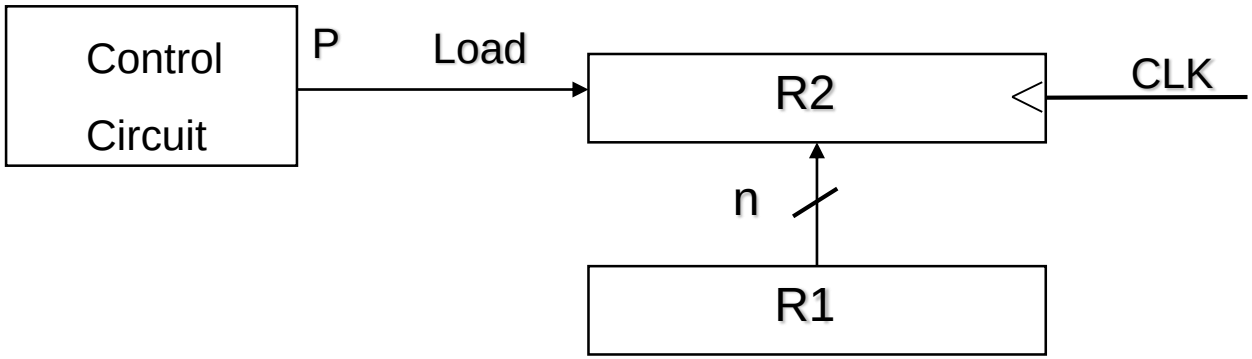
گاهی اوقات انتقال فقط باید در صورت تحقق یک شرط کنترلی خاص انجام شود نظیر عبارت:

If (P=1) then (R2← R1)

تابع کنترلی:

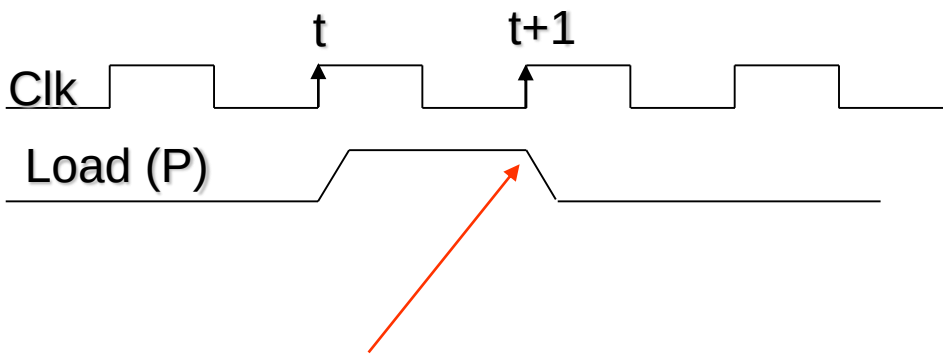
یک عبارت بولی است که مقدار صفر یا یک دارد این تابع بصورت زیر در عبارت میکرواپریشن گنجانیده میشود

P: R2 ← R1



University of Kurdistan

زمان بندی



مدار کنترل در زمان t سیگنال P را فعال میکند لذا وقتی کلاک t+1 به وقوع می پیوندد با P فعال روبرو شده و باعث انتقال R1 به R2 میشود.

عمل انتقال در این زمان اتفاق می افتد

این انتقال در لبه کلاک t+1 اتفاق می افتد. اگر P بیش از یک Clk فعال باشد عمل انتقال در سایر کلاک ها هم اتفاق خواهد افتاد.

فعال بودن به لبه باعث میشود تا بتوان عمل را در یک کلاک پالس انجام داد.

T: R2 ← R1 , R1 ← R2

University of Kurdistan

Multiplexer-Based Transfers

- **Consider**

$$if\ (K1 = 1)\ then\ (R0 \leftarrow R1)\ else\ if\ (K2 = 1)\ then\ R0 \leftarrow R2$$

- **Which can also be expressed as**

$$K1 : R0 \leftarrow R1,$$

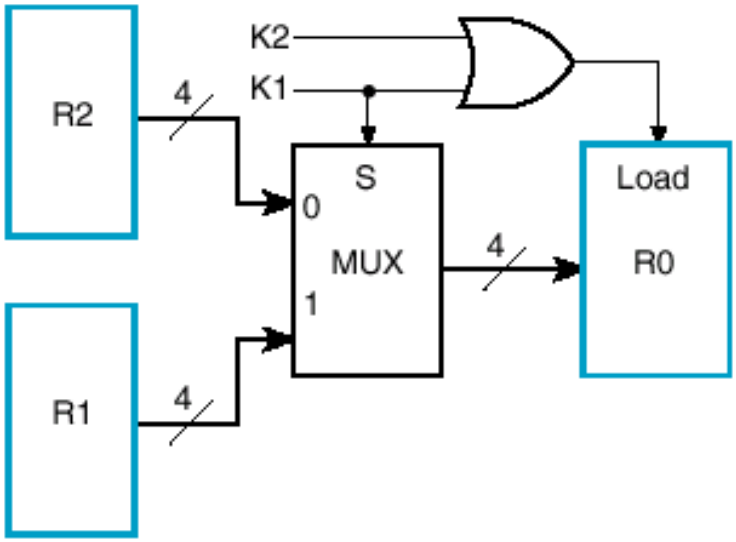
$$\overline{K1}\ K2 : R0 \leftarrow R2$$

- **Block diagram?**

Multiplexer Block Diagram

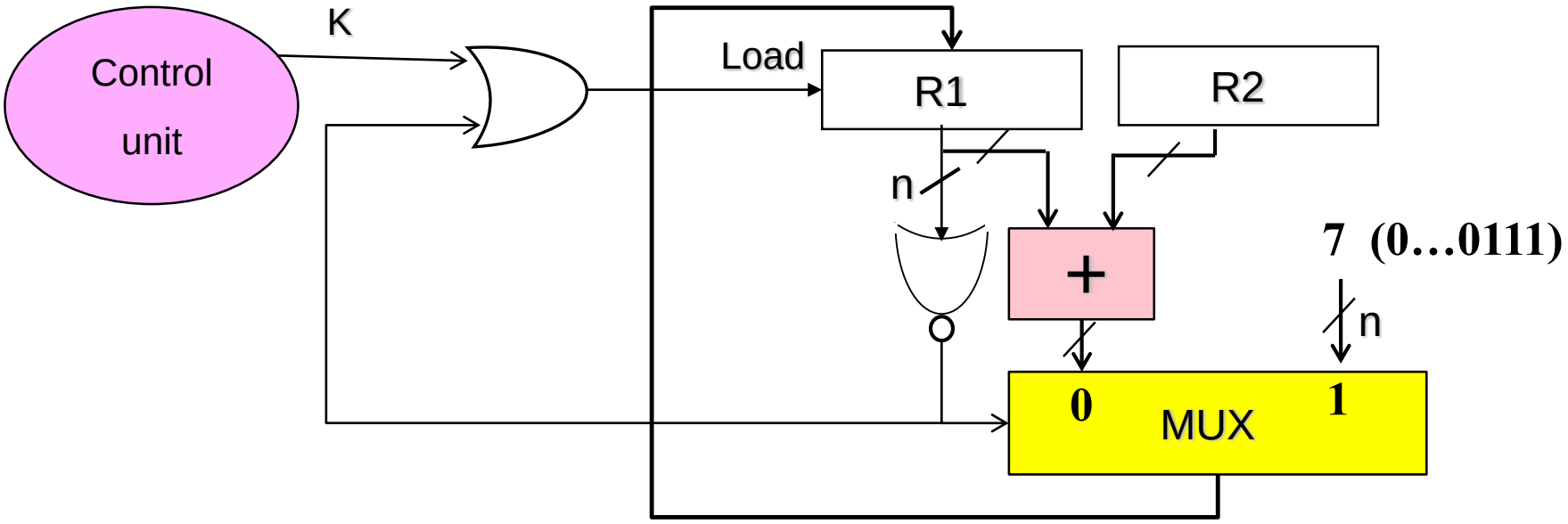
$$K1: R0 \leftarrow R1,$$

$$\overline{K1}\, K2: R0 \leftarrow R2$$



تمرین

$$if\ (R1 = 0)\ then\ (R1 \leftarrow 7)\ else\ if\ (K = 1)\ then\ R1 \leftarrow R1 + R2$$



انتقال از طریق گذرگاه (BUS)

- در یک کامپیوتر تعداد زیادی رجیستر وجود دارد که همه آنها باید به نوعی به هم متصل شوند برای سهولت در اتصال از **Bus** استفاده می شود.

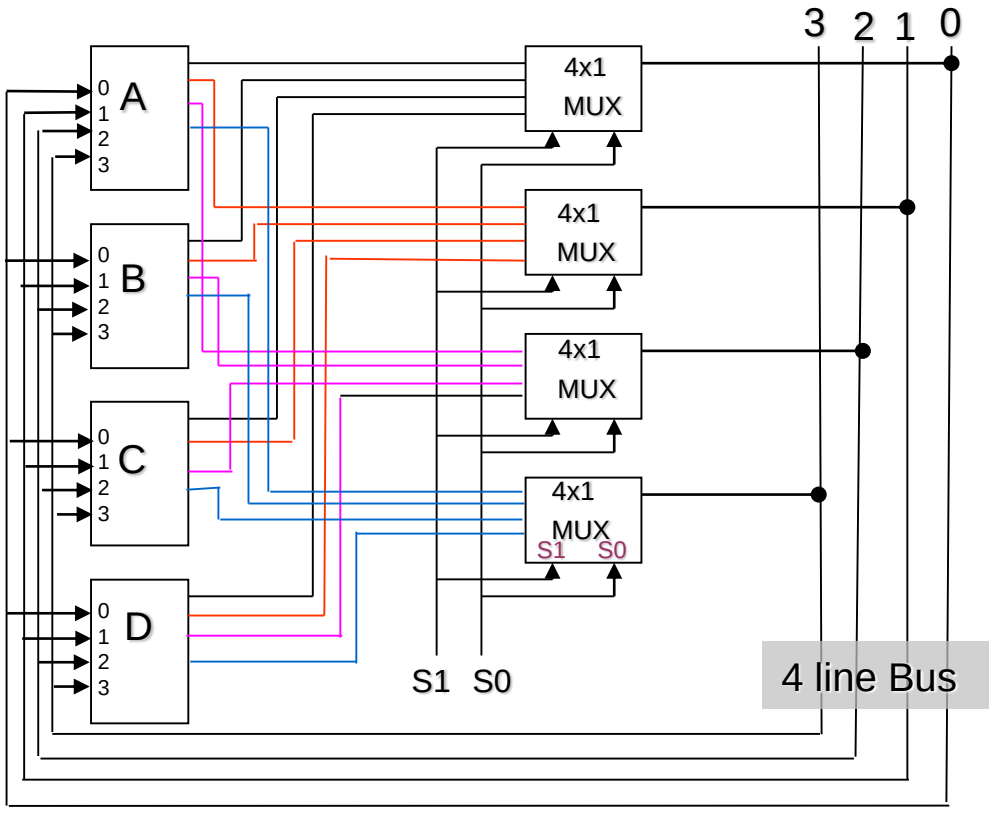
- **Bus** به مجموعه ای از خطوط مشترک اتلاق میشود که با داشتن **n** خط می توانند اطلاعات رجیسترهای **n** بیتی را به طور همزمان منتقل کنند.

- برای پیاده سازی **Bus** از دو طریق استفاده می شود:

۱. مالتی پلکسرها
۲. گیت های با خروجی ۳ حالته

University of Kurdistan

BUS با استفاده از MUX



S1 S0	Reg Selected
0 0	A
0 1	B
1 0	C
1 1	D

تعداد MUXها : به تعداد بیتهای رجیسترها (Mux 0 بیت صفر تمام رجیسترها را روی bus قرار میدهد و ...)

تعداد ورودی MUX ها : به تعداد رجیسترها

انتقال از طریق BUS

- معمولا ورودی همه رجیسترهای مقصد به **BUS** متصل میباشند. از اینرو کافیست S_1S_0 بنحوی انتخاب شود که خروجی رجیستر روی **BUS** قرار گیرد. همزمان ورودی **Load** رجیستر مقصد نیز فعال شود تا داده از مبدا به مقصد انتقال یابد. برای مثال برای انتقال رجیستر **A** به رجیستر **B** از طریق باس به ترتیب زیر عمل میشود:

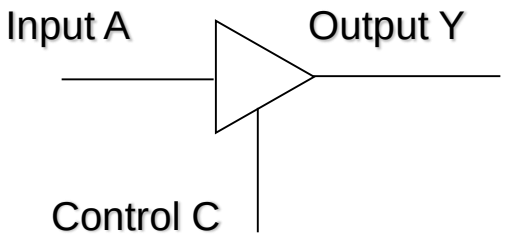
$$BUS \leftarrow A \quad , B \leftarrow BUS$$

در صورتی که وجود **BUS** را بصورت ضمنی قبول کنیم بصورت ساده می نویسیم:

$$B \leftarrow A$$

پیاده سازی BUS با استفاده از گیتهای ۳ حالته: TriState Buffer

- گیت سه حالته : مداری است که خروجی آن علاوه بر ۲ حالت صفر و یک می تواند دروضعیت سومی قرار گیرد که اصطلاحا حالت امپدانس بالا نامیده شده و در این حالت خروجی گیت بصورت مدار باز عمل میکند یعنی نه به صفر وصل است نه به یک.

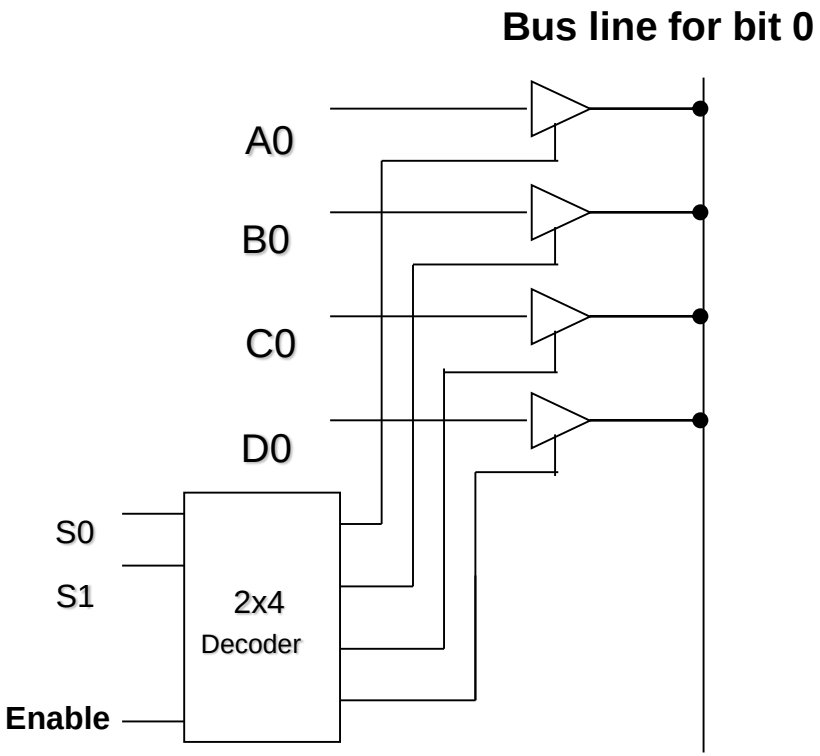


Y=A c=1

Y=High Impedance c=0

یک بافر ۳ حالته و عملکرد آن

پیاده سازی BUS با استفاده از گیتهای ۳ حالته: TriState Buffer



- استفاده از گیتهای سه حالته سبب میشود تا بتوان یک Bus را براحتی از طریق اتصال خروجی رجیسترهای سه حالته و یا استفاده از یک بافر سه حالته بوجود آورد

- فقط باید مواظب بود که در هر لحظه فقط یکی از گیت های سه حالته فعال باشد این اطمینان با استفاده از یک Decoder حاصل میشود

انواع میکرواپریشن یا ریزعملیات

- میکرواپریشن های متداول پیاده سازی شده در کامپیوتر های دیجیتال عبارتند از:

۱. انتقال رجیستر: اطلاعات را از یک رجیستر به دیگری منتقل می کنند.
۲. محاسباتی: عملیات محاسباتی را روی داده ذخیره شده در رجیسترها انجام میدهند.
۳. منطقی: عملیات دستکاری روی بیت های رجیستر را انجام می دهد.
۴. hift : عمل شیفت را روی داده ذخیره شده روی رجیسترها انجام می دهند.

University of Kurdistan

انتقال حافظه

می توان برای کار با حافظه از دو رجیستر استفاده کرد: AR ، DR

عمل خواندن حافظه: Read

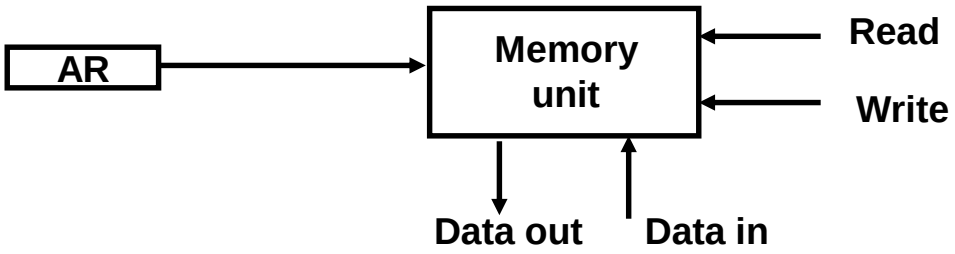
برای خواندن از یک محل مشخص از حافظه باید آدرس آن محل مشخص شده در رجیستر خاصی به نام Address Register نوشته شود. داده خوانده شده از حافظه نیز در رجیستر DR نوشته می شود.

عمل نوشتن درحافظه: Write

- آدرس محل مورد نظر در AR ریخته میشود. این محل حافظه بصورت M[AR] نشان داده میشود. داده مورد نظر میتواند در هر یک از رجیسترها قرار داشته باشد.
- میکرو اپریشن های مربوطه بصورت زیر خواهند بود:

Read : $DR \leftarrow M[AR]$

Write : $M[AR] \leftarrow R_1$



میکرواپریشن های محاسباتی

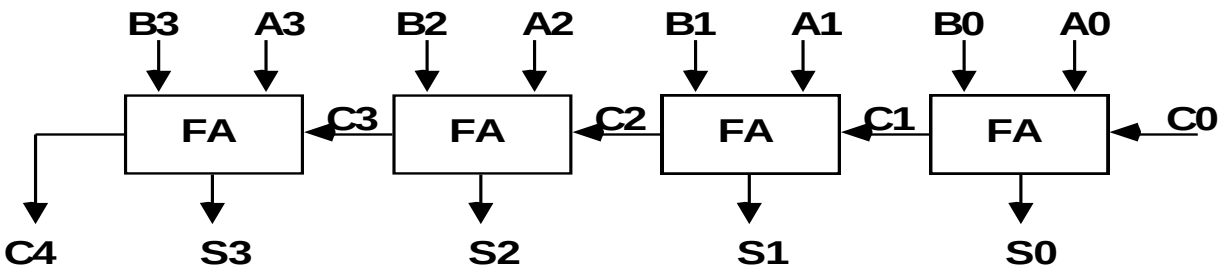
- میکرو اپریشن های محاسباتی اصلی عبارتند از:

  جمع، تفریق، افزایش، کاهش و شیفت

- میکرواپریشن جمع:
- بصورت روبرو نشان داده میشود: R3 ← R1+R2

احتیاجات سخت افزاری:

سه رجیستر و مدارات دیجیتالی که عمل جمع را انجام دهند.



میکرواپریشن های محاسباتی

میکرو اپریشن تفریق :

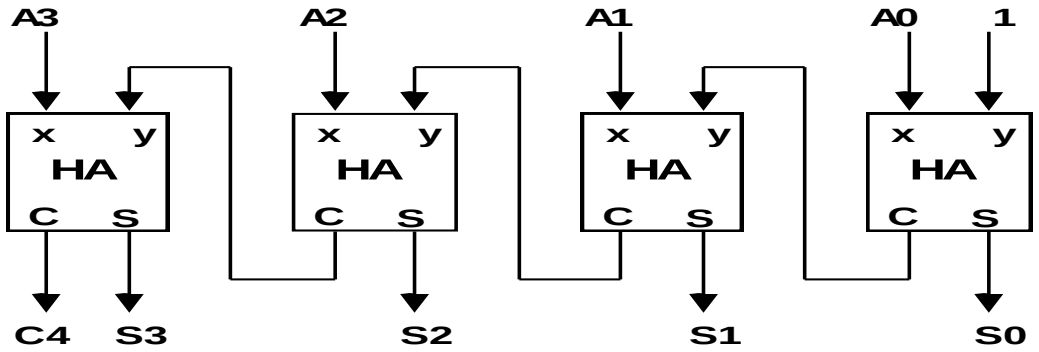
اغلب از طریق متمم گیری و جمع انجام می شود:

$R3 \leftarrow R1+\overline{R2}+1$ **معادل است با** $R3 \leftarrow R1-R2$

میکرواپریشن INC ، DEC :

(با مدار ترکیبی و با با یک شمارنده بالا- پائین شمار ساخته می شوند)

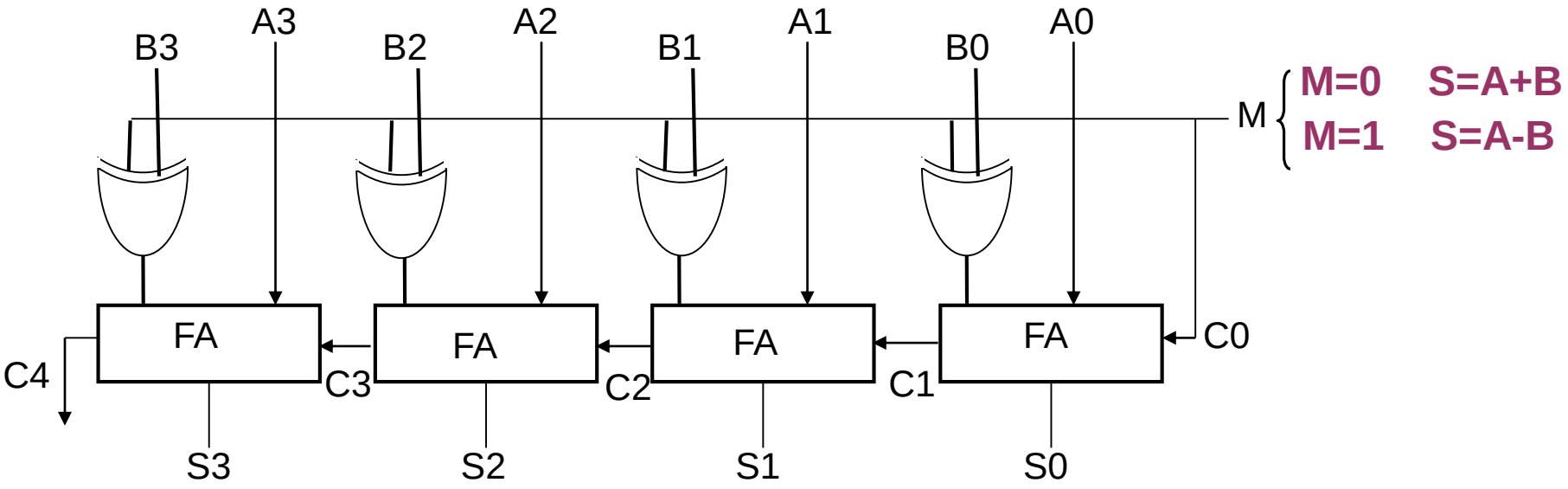
$R1 \leftarrow R1+1$ $R1 \leftarrow R1-1$



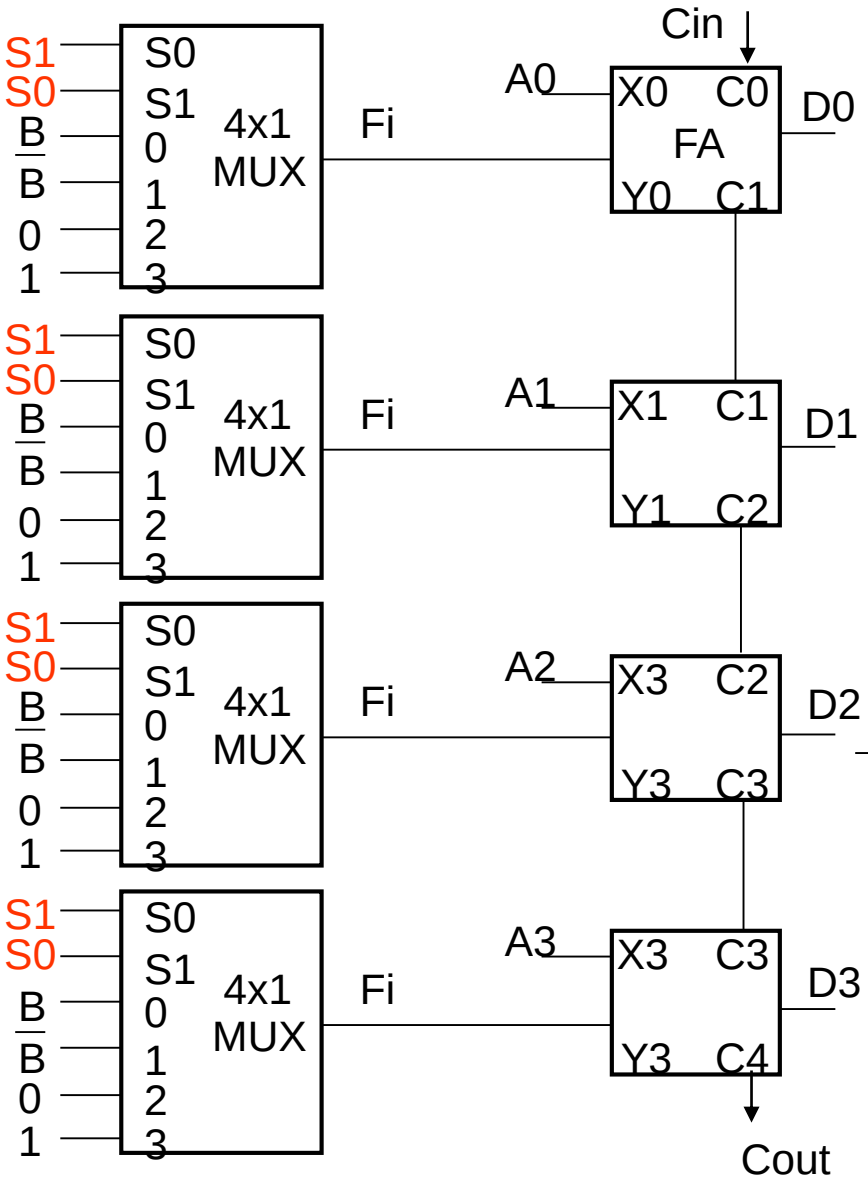
جمع کننده و تفریق کننده باینری

برای اجرای ریزعملوند جمع نیاز به مداری است که عمل جمع را روی رجیسترها انجام دهد. این مداراز n مدار تمام جمع کننده FA ساخته می شود.

برای اضافه نمودن عمل تفریق، کافی است از تعدادی گیت XOR و یک ورودی کنترلی M مطابق شکل زیر کمک بگیریم.



مدار واحد محاسباتی



می توان با استفاده از مدار روبرو انواع میکرو اپریشنهای محاسباتی را یکجا پیاده سازی کرد. خروجی این مداربصورت جدول زیر است.

D= A+Y+Cin

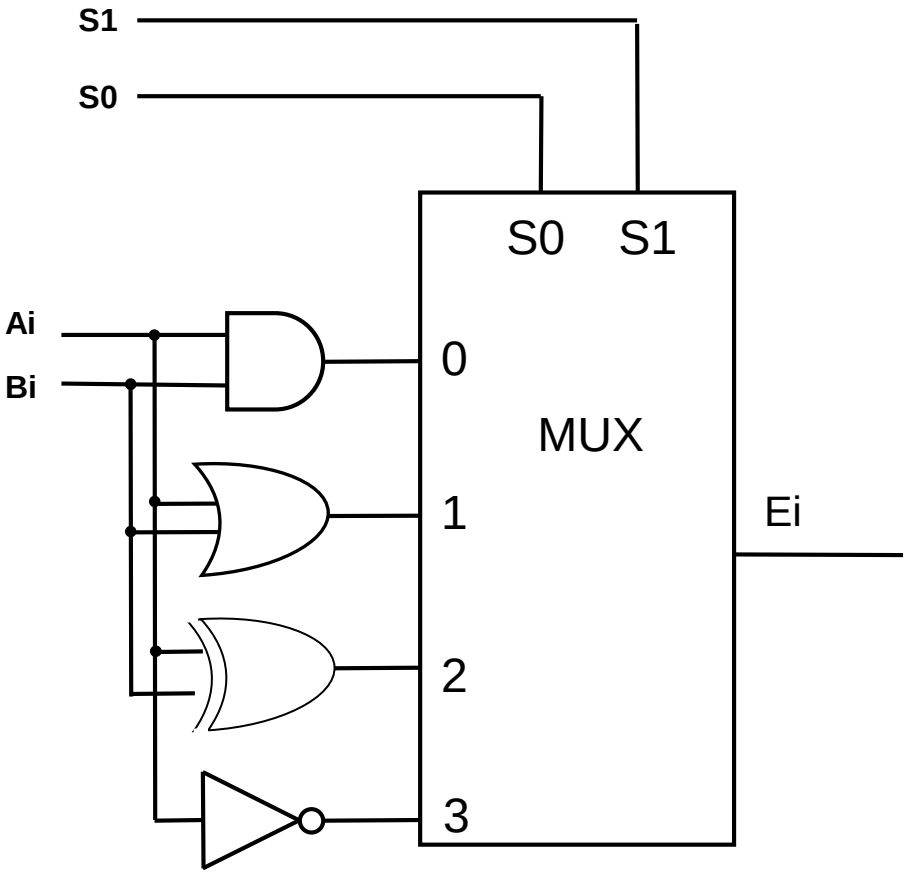
S0	S1	CIN	Y	D	Micro Operatioon
0	0	0	B	A+B	Add
0	0	1	$\overline{B}$	A+$\overline{B}$+1	Add With Carry
0	1	0	$\overline{B}$	A+$\overline{B}$	Sub with Borrow
0	1	1	B	A+B+1	Sub
1	0	0	0	A	
1	0	1	0	A+1	Inc
1	1	0	1	A-1	Dec
1	1	1	1	A	Transfor
					.

Logic MicroOperations میکرواپریشن های منطقی

- این میکرو اپریشن ها عملیات باینری منطقی را روی بیت های ذخیره شده در رجیسترها انجام میدهند. عمل باینری بصورت جداگانه بر روی هر بیت اعمال میگردد.

- میکرواپریشن XOR $P: R1 \leftarrow R1 \oplus R2$
- میکرواپریشن OR $P: R1 \leftarrow R1 \vee R2$
- میکرواپریشن AND $P: R1 \leftarrow R1 \wedge R2$

پیاده سازی سخت افزاری میکرواپریشن های منطقی



S1	S0	OutPut
0	0	$E = A \wedge B$
0	1	$E = A \vee B$
1	0	$E = A \oplus B$
1	1	$E = \overline{A}$

برخی کاربردهای میکرو اپریشن های منطقی

: Selective Set

برای یک کردن بیتهایی از رجیستر A بکار میرود. بیتهایی را که میخواهیم یک شوند، توسط رجیستر B مشخص میکنیم.

A	1010
B (دو بیت سمت چپ A را مجبور به یک شدن می کنیم، بقیه دست نمی خورد.)	1100
A (بعد از عمل)	1110

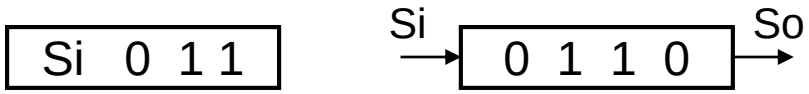
اینکار با $A \leftarrow A \vee B$ انجام میشود.

سوال: برای selective clear به چه صورت عمل می کنیم؟

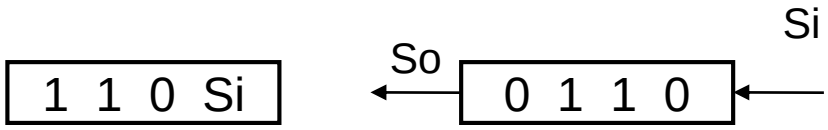
میکرو اپریشن های شیفت

- این میکرو اپریشن ها برای انتقال سریال داده بکار میروند

شیفت به راست :



شیفت به چپ :



قبل از شیفت

بعد از شیفت

انواع شیفت

شیفت منطقی Logical Shift :

- مقدار $S_i=0$ در نظر گرفته میشود.

شیفت منطقی به سمت چپ R1 ⟵ Shl R1

در RTL میکرواپریشن هر دو رجیستر مبدا و مقصد باید یکسان باشند.

شیفت منطقی به سمت راست R2 ⟵ Shr R2

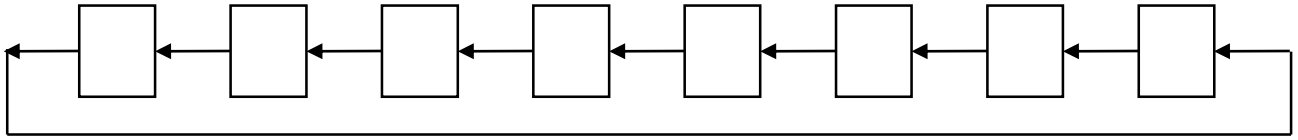
University of Kurdistan

انواع شیفت

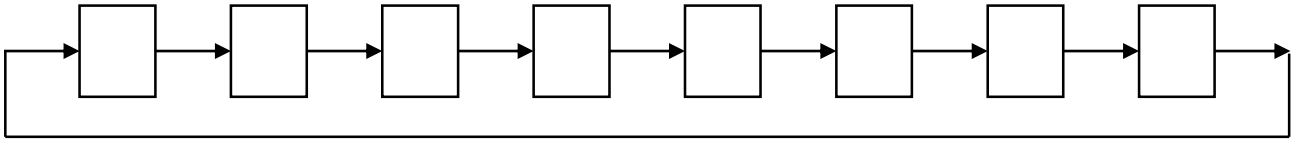
شیفت چرخشی Circular Shift :

- Si به So وصل میشود.

شیفت چرخشی به سمت چپ Cil R1 ⟵ R1



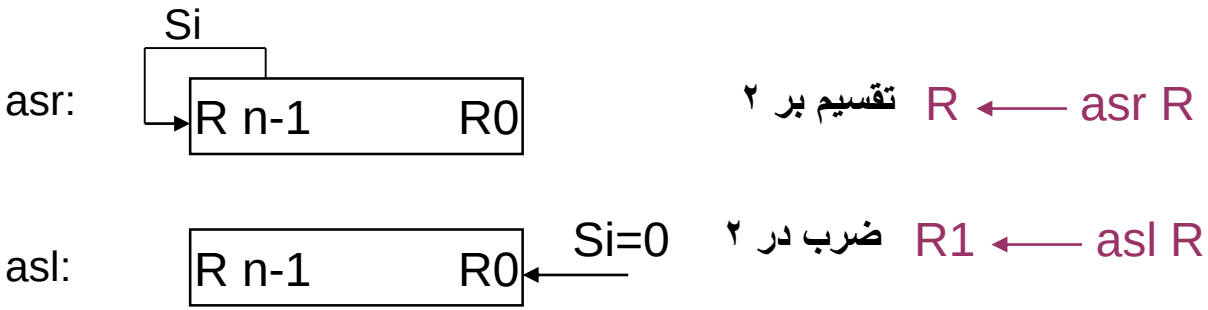
شیفت چرخشی به سمت راست Cir R2 ⟵ R2



انواع شیفت

شیفت محاسباتی Arithmetic Shift :

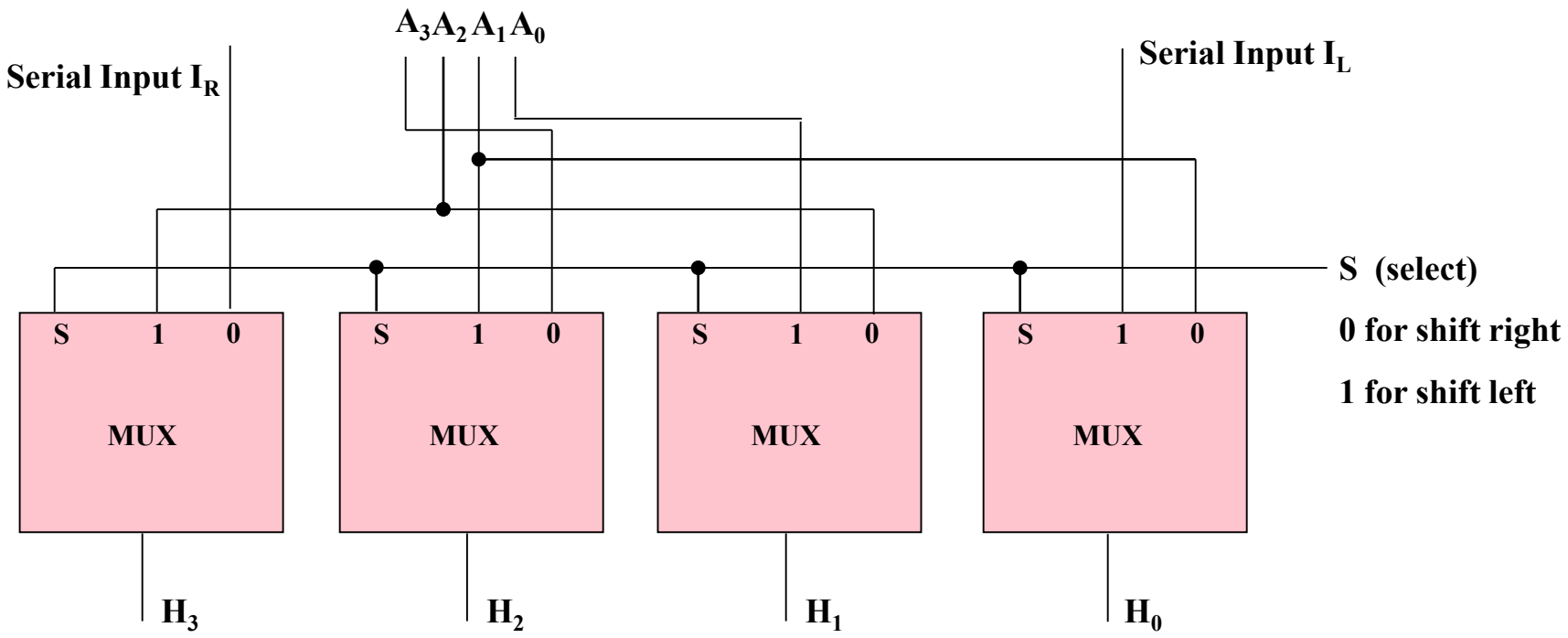
- در این نوع شیفت، شیفت به راست به منزله تقسیم بر 2 و شیفت به سمت چپ به منزله ضرب در 2 است. از اینرو محتوی بیت علامت رجیستر در هنگام شیفت بدون تغییر باقی میماند.



- اگر در asl مقدار Rn-1 عوض شود OverFlow اتفاق می افتد.
برای تشخیص این امر از رابطه زیر استفاده میشود :(قبل از شیفت)

$$V \leftarrow R_{n-1} \oplus R_{n-2}$$

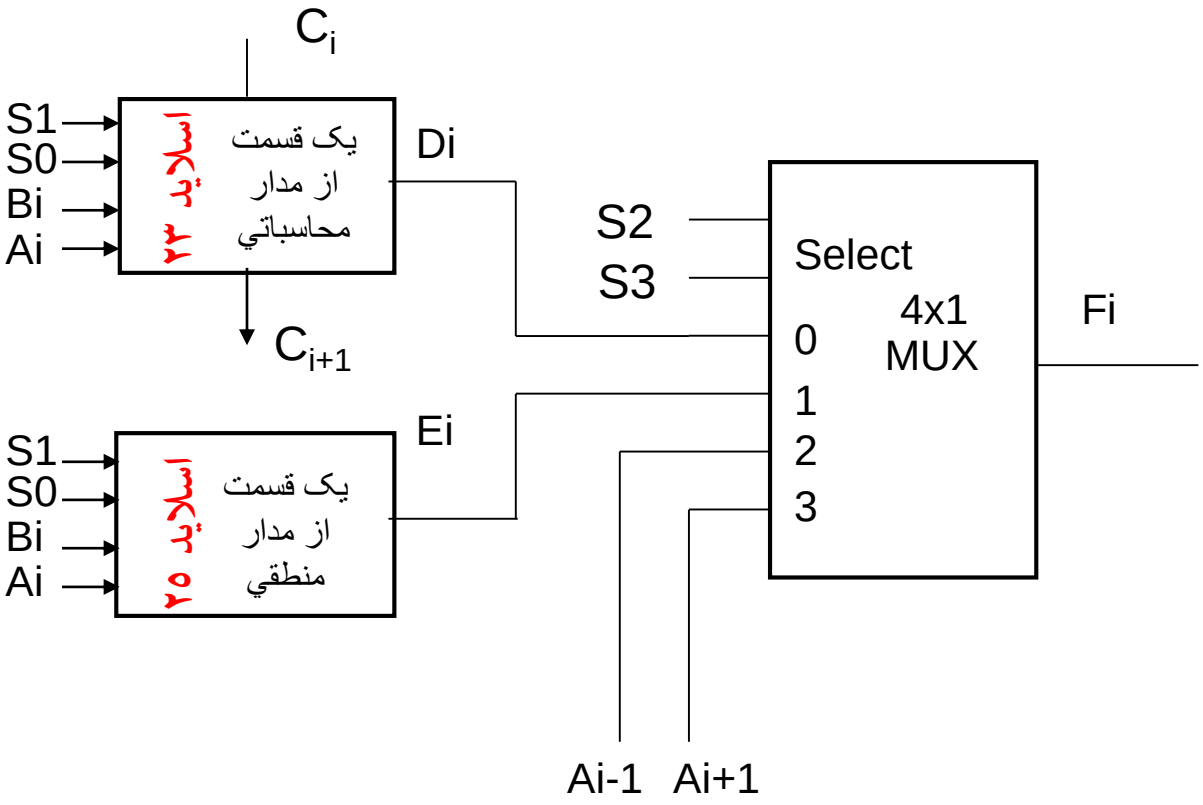
پیاده سازی سخت افزاری میکرو اپریشن های شیفت



مدار ترکیبی برای شیفت 4 بیتی

واحد محاسبه و منطق (ALU)

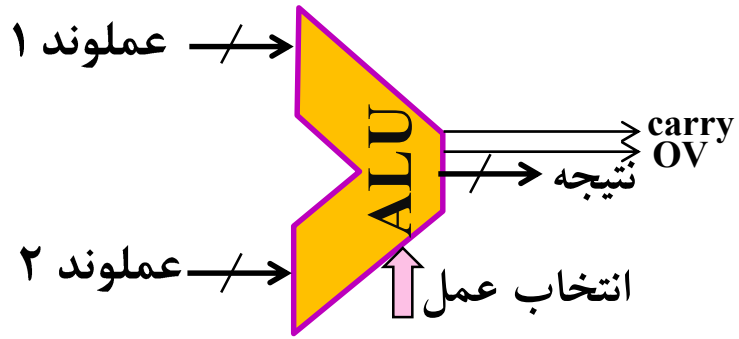
مدار سخت افزاری برای یک بیت ALU :



S3	S2	S1	S0	Cin	F Operation
0	0	0	0	0	A+B
0	0	0	0	1	A+B+1
0	0	0	1	0	$A+\overline{B}$
0	0	0	1	1	A - B
0	0	1	0	0	A
0	0	1	0	1	A+1
0	0	1	1	0	A -1
0	0	1	1	1	A
0	1	0	0	x	A^ B
0	1	0	1	x	A∨B
0	1	1	0	x	A⊕B
0	1	1	1	x	$\overline{A}$
1	0	x	x	x	shl A
1	1	x	x	0	shr A

واحد محاسبه و منطق (ALU)

- به جای اینکه هر رجیستر جداگانه عملیات را انجام دهد، میتوان از یک واحد محاسباتی منطقی مستقل برای انجام عملیات بر روی رجیسترهای مختلف استفاده کرد. این واحد ALU نامیده میشود.
- برای انجام یک میکرو اپریشن خروجی رجیستر مبدا به ورودی ALU متصل شده و خروجی ALU به ورودی رجیستر مقصد متصل میشود.
- مدار ترکیبی ALU عمل مورد نظر را که توسط ورودیهای کنترل مشخص میشود انجام میدهد.
- عمل شیفت میتواند توسط ALU یا به صورت مجزا توسط واحد دیگری انجام شود.

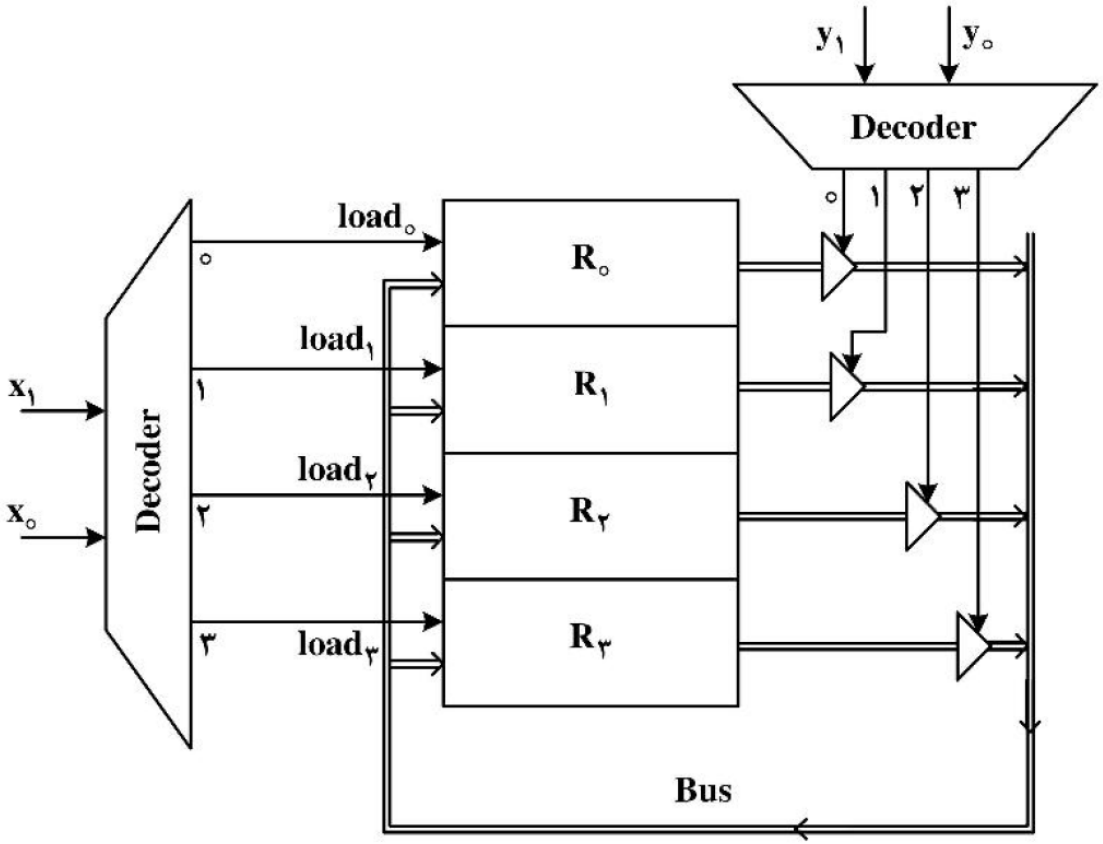


University of Kurdistan

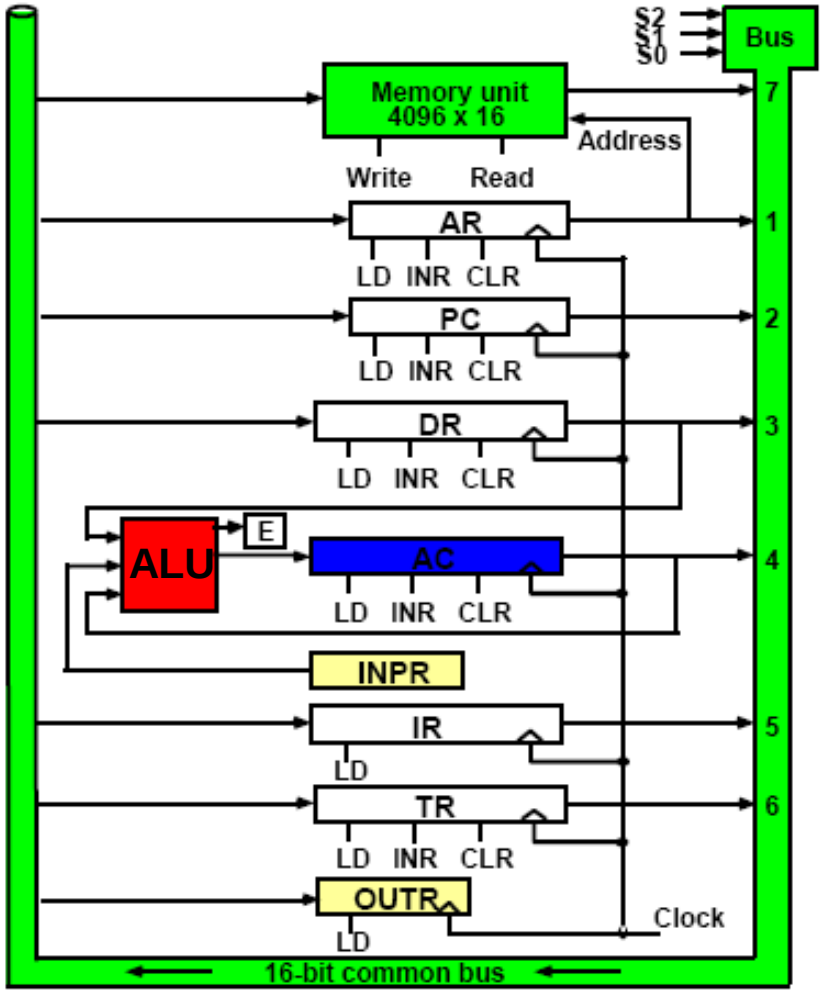
نمونه سوال کنکور

در شکل زیر، اگر قسمت Opcode دستورالعمل به شکل | x_1 | x_0 | y_1 | y_0 | باشد، دستور R_1 , R_3 mov
(جهت انتقال از راست به چپ) دارای چه Opcode به Hex است؟

۱) D
۲) E
۳) ۷
۴) ۱



نمونه سوال



در مسیرداده روبرو کدام عمل در یک پالس ساعت امکان پذیر است؟

IR← M[PC]

DR← DR+ AC

AC ← TR

DR← M[AR]

AR← PC

AC← DR+ AC

AC← M[AR]

Questions