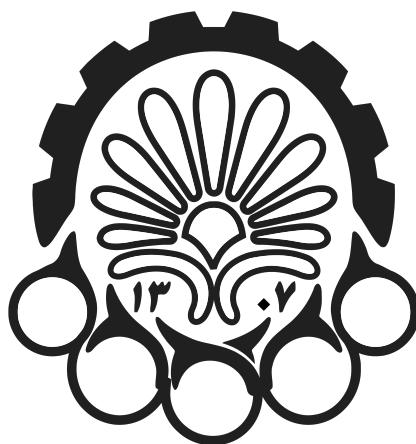


# معماری افزاره‌های شبکه دکتر صبا ئی



**دانشگاه صنعتی امیرکبیر**  
( پلی تکنیک تهران )  
دانشکده مهندسی کامپیوتر

رضا آدینه پور ۴۰۲۱۳۱۰۵۵

تمرین سری ششم

۱ دی ۱۴۰۳

## سوال اول

یک سوئیچ Shared Memory که قرار است به لینک‌های ATM با ظرفیت 125 Mbps متصل شود و از حافظه‌های با زمان دسترسی 16 ns استفاده نماید، حداکثر چند پورت می‌تواند داشته باشد؟

پاسخ

$$t_{\text{mem}} = 16 \text{ ns}, \quad L = 53 \text{ byte} = 424 \text{ bit}$$

$$r = 125 \text{ Mbps}$$

در حافظه‌های shared memory،  $N$  نوشتن در حافظه و  $N$  خواندن از حافظه در یک Cell Slot انجام می‌شود. بنابراین داریم:

$$(2N) \cdot t_{\text{mem}} \leq \frac{L}{r} \rightarrow N \leq \frac{L}{2r \cdot t_{\text{mem}}}$$

بنابراین:

$$N \leq \frac{424}{2 \times 125 \times 10^6 \times 16 \times 10^{-9}} \rightarrow N \leq 106$$

## سوال دوم

در یک سوئیچ Division-Time زمان دسترسی به حافظه 5 ns است. طول بسته‌های ورودی 200 بایت است و خطوط ورودی همگی 5 Gbps هستند. در هر یک از موارد زیر بیشترین تعداد خطوط ورودی به این سوئیچ را مشخص کنید:

- Shared-Memory Switch
- Shared Medium Switch

پاسخ

$$t_{\text{mem}} = 5 \text{ ns}$$

$$L = 200 \times 8 = 1600 \text{ bit}$$

$$r = 5 \times 10^9 \text{ bps}$$

**Shared Medium:**

$$(N + 1)t_{\text{mem}} \leq \frac{L}{r} \Rightarrow N + 1 < \frac{1600}{5 \times 10^9 \times 5 \times 10^{-9}}$$

$$N + 1 \leq 64 \Rightarrow N \leq 63$$

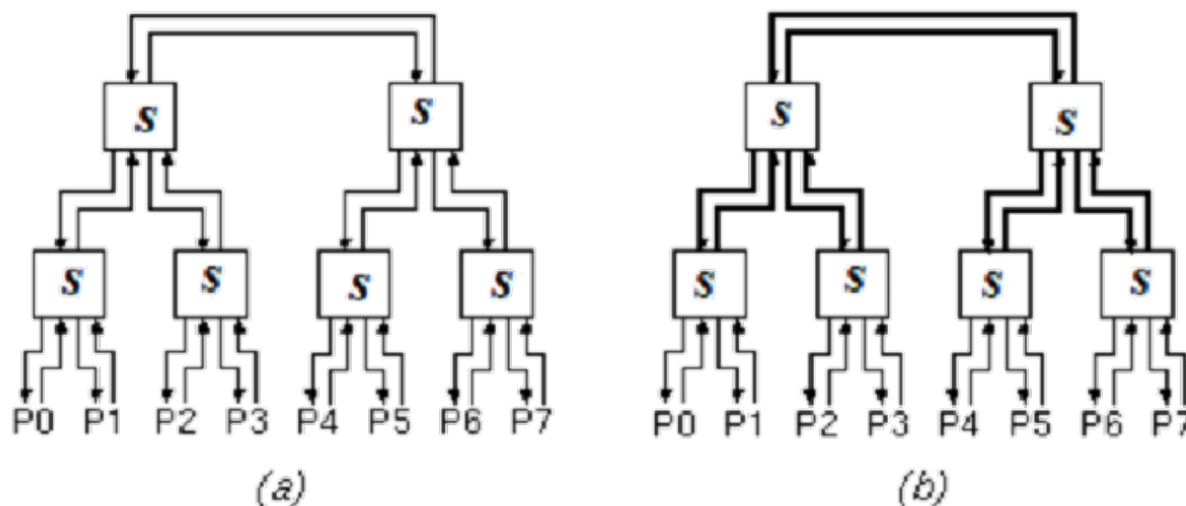
**Shared Memory:**

$$(2N)t_{\text{mem}} \leq \frac{L}{r} \Rightarrow N \leq \frac{1600}{2 \times 5 \times 10^9 \times 5 \times 10^{-9}}$$

$$N \leq 32$$

## سوال سوم

در شکل زیر یک سوئیچ  $8 \times 8$  را نشان می‌دهد. همان طور که مشخص است این سوئیچ دارای ساختاری درختی است. تمام لینک‌ها در هر شکل  $a$  ظرفیت عبور تنها یک بسته در هر برش زمانی را دارند.



شکل ۱: ساختار سوئیچ سوال سوم

۱. الگوی ترافیکی را مثال بزنید که تمام پورت‌های ورودی و خروجی اشغال باشند اما سوئیچ دچار Blocking نمی‌شود (فرض کنید الگویی که هر پورت ورودی به پورت خروجی هم‌نام خودش  $p(in)$  به  $p(out)$  وصل شده باشد امکان‌پذیر نباشد).

پاسخ

- (a)  $P_1(out)$  and  $P_0(in)$
- (b)  $P_3(out)$  and  $P_2(in)$
- (c)  $P_5(out)$  and  $P_4(in)$
- (d)  $P_7(out)$  and  $P_6(in)$

۲. الگوی ترافیکی را مثال بزنید که نشان دهد در شکل  $a$  سوئیچ دچار Internal Blocking می‌شود.

پاسخ

برای مثال اگر  $P_0$  پورت ورودی و  $P_4$  پورت خروجی باشد، با وجود اینکه پورت ورودی  $P_1$  آزاد است و به غیر از  $P_4$  تمام پورت‌های خروجی نیز آزاد هستند، به علت internal blocking از  $P_1$  به هیچ‌یک از پورت‌های  $P_2, P_3, P_5, P_6, P_7$  نمی‌توان بسته فرستاد.

۳. اگر در شکل  $b$  فرض کنیم خطوط پررنگ تر ظرفیت ارسال ۲ بسته در یک برش زمانی را دارند. آیا این تغییر سوئیچ شکل  $b$  دچار Internal Blocking نمی‌شود؟

پاسخ

بله، برای مثال دو انتقال زیر را در نظر بگیرید.

- ورودی  $P_0$  و خروجی  $P_4$

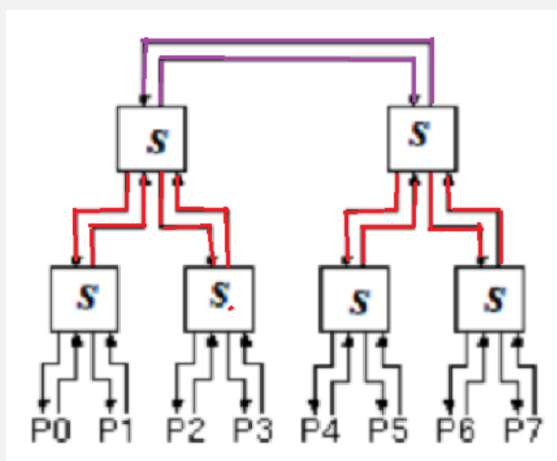
- ورودی  $P_1$  و خروجی  $P_5$

در این صورت، به دلیل internal blocking با وجود آزاد بودن پورت ورودی  $P_2$  و پورت خروجی  $P_6$  امکان انتقال بسته از  $P_2$  به  $P_6$  وجود ندارد، زیرا نیاز دارد از بالاترین خط همزمان ۳ بسته در یک برش زمانی ارسال شود که بیش از ظرفیت لینک (دو بسته) است.

۴. کمترین ظرفیتی که می‌توان به سوئیچ قسمت a اضافه کرد که سوئیچ دچار Internal Blocking نشود چیست؟

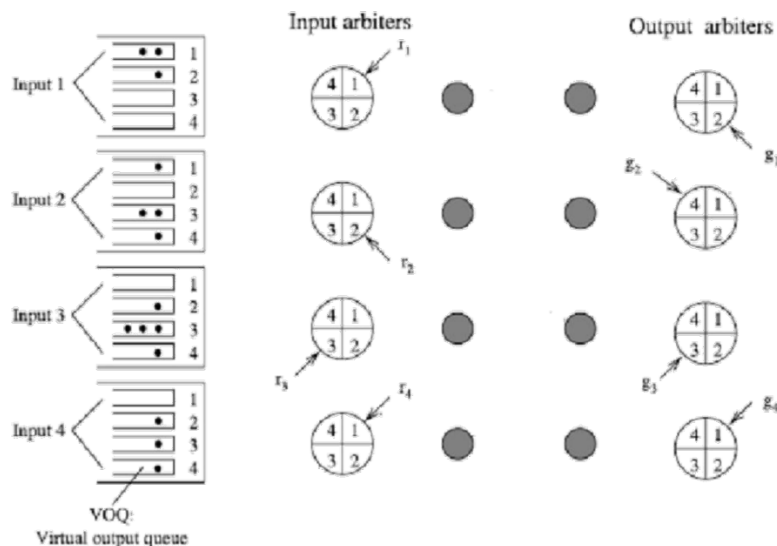
پاسخ

اگر ظرفیت لینک‌های قرمز ۲ بسته در یک برش زمانی و ظرفیت لینک‌های بنفش ۴ بسته در یک برش زمانی باشد، internal blocking رخ نخواهد داد.



## سوال چهارم

الگوریتم DRRM را بر روی شکل زیر اعمال کنید. این الگوریتم را تا دو مرحله اجرا کنید. هر مرحله شامل دو Iteration است.



شکل ۲: شکل مورد نظر

پاسخ

## Phase 1 - Iteration 1

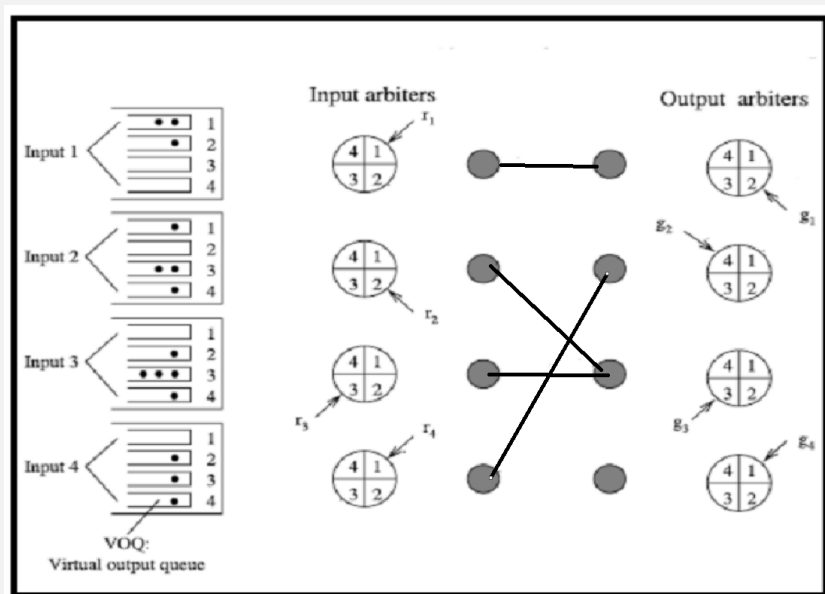


Figure 3: Step 1

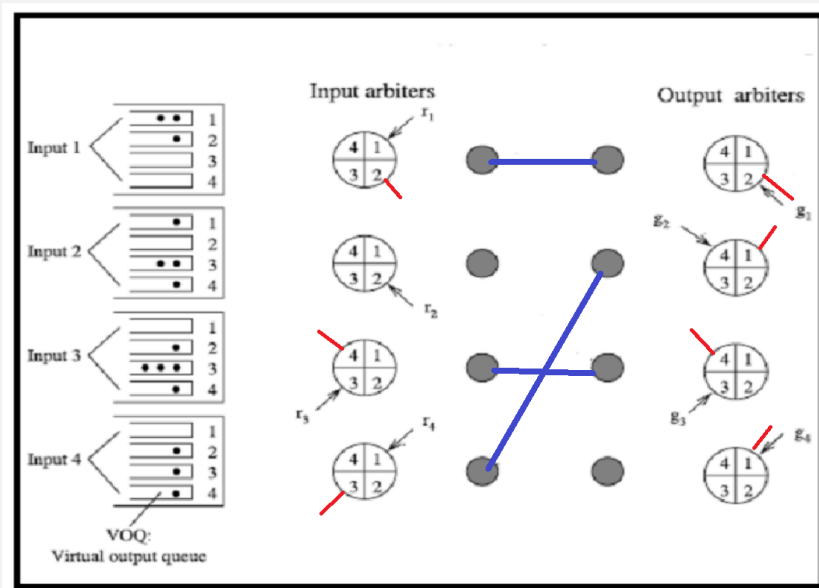


Figure 4: Step 2

### Phase 1 - Iteration 2

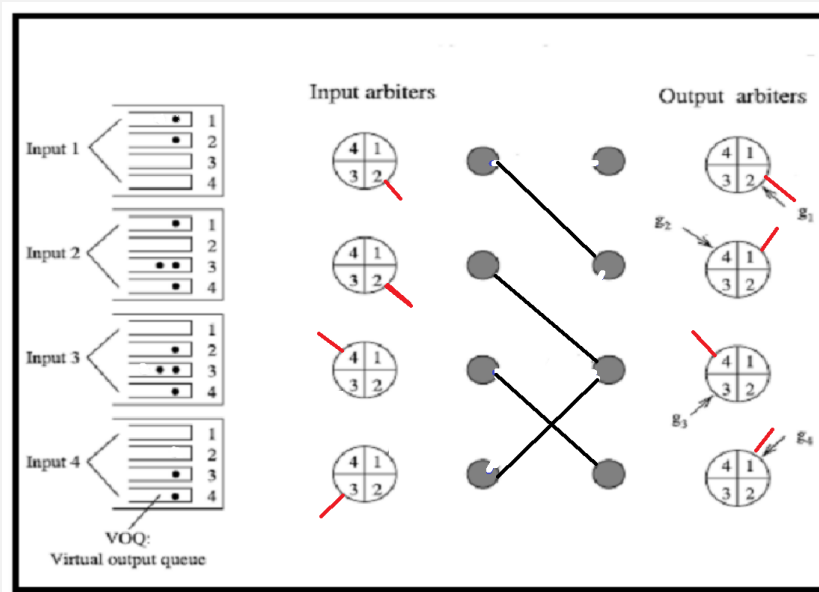


Figure 5: Step 1

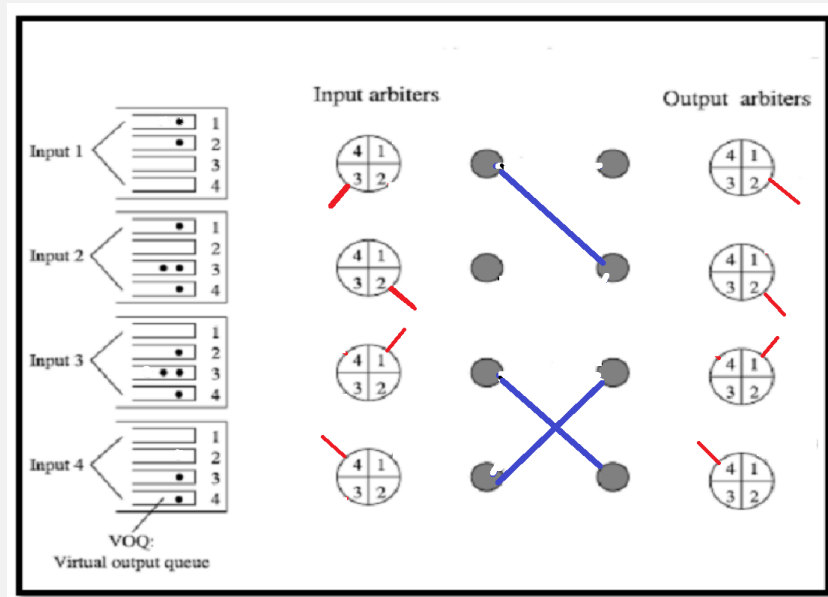


Figure 6: Step 2

### Phase 2 - Iteration 1

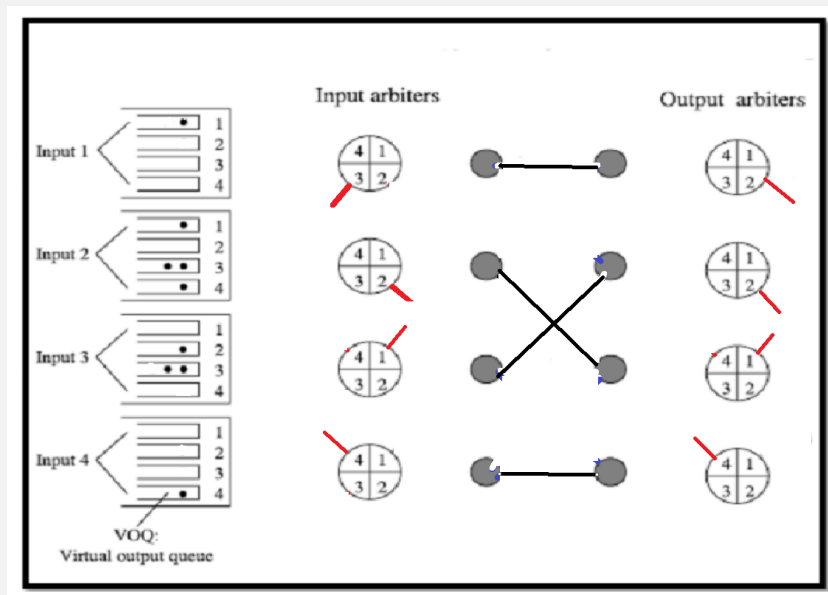


Figure 7: Step 1



پاسخ

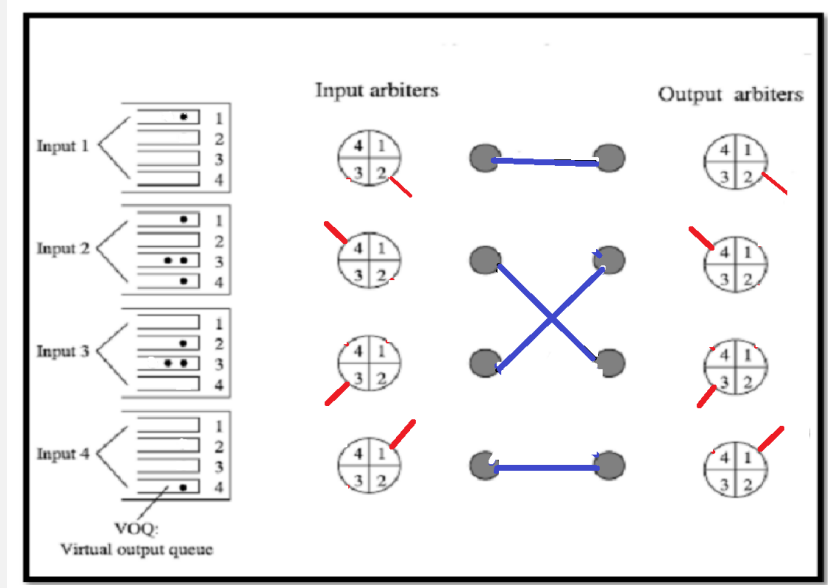


Figure 8: Step 2

## Phase 2 - Iteration 2

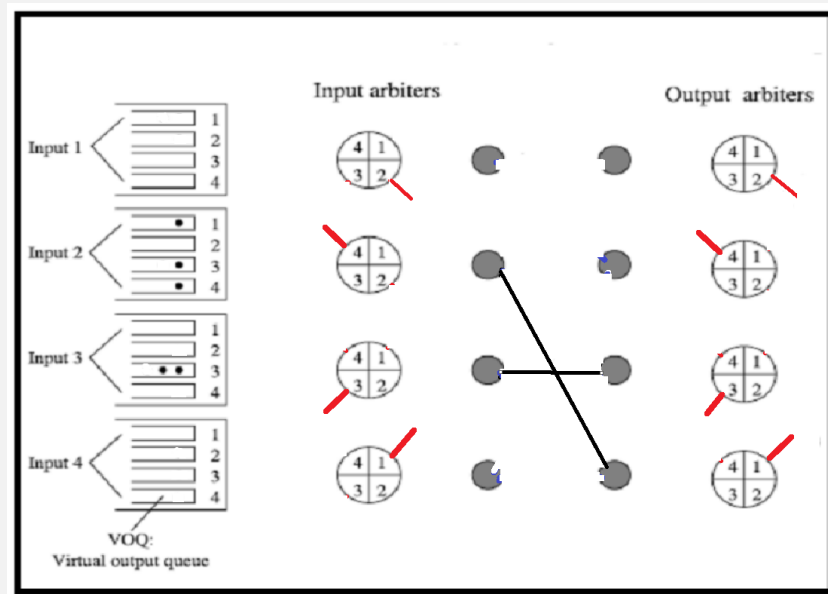


Figure 9: Step 1

پاسخ

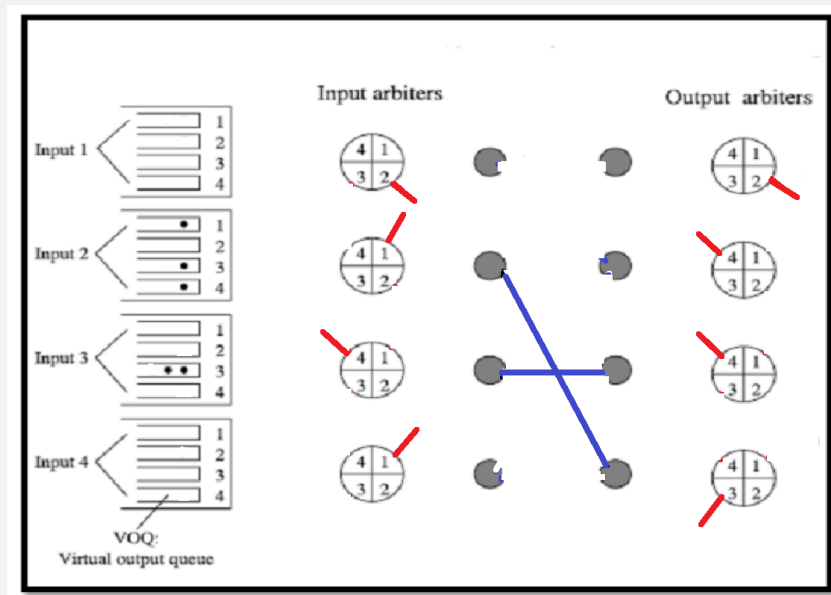
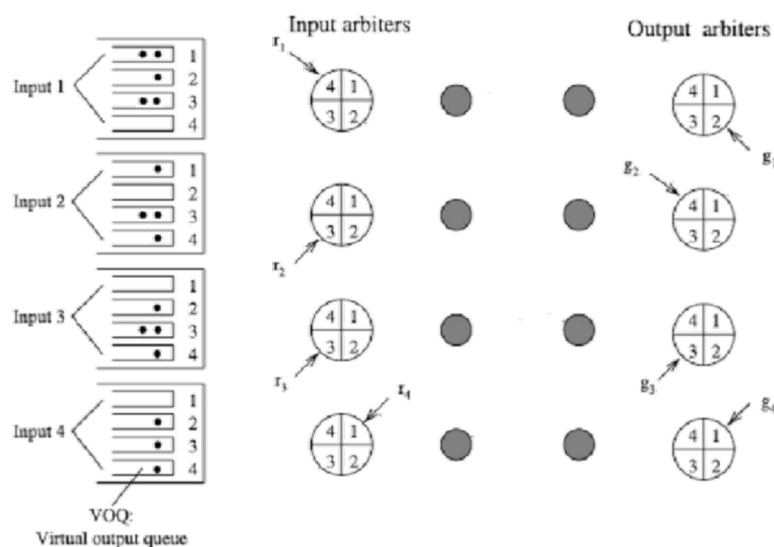


Figure 10: Step 2

## سوال پنجم

الگوریتم EDRRM را بر روی شکل زیر اعمال کنید. این الگوریتم را تا دو مرحله اجرا کنید. هر مرحله شامل یک Iteration است.



شکل ۱۱: شکل مورد نظر

پاسخ

## Phase 1 - Iteration 1

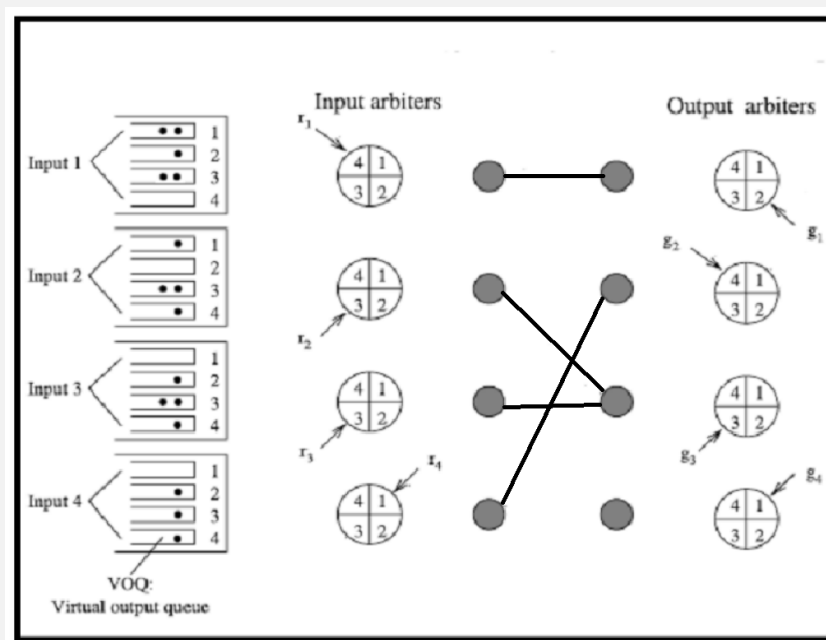


Figure 12: Step 1

پاسخ

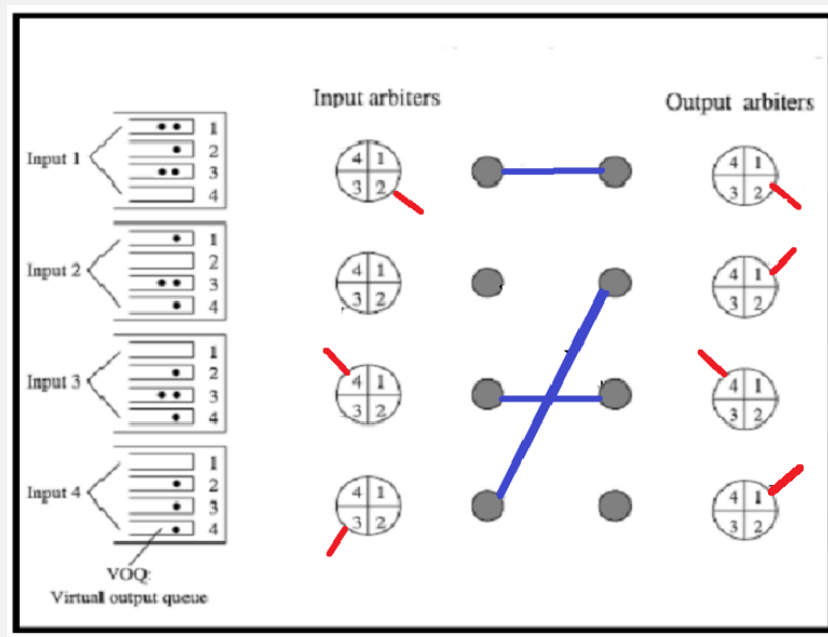


Figure 13: Step 2

### Phase 1 - Iteration 2

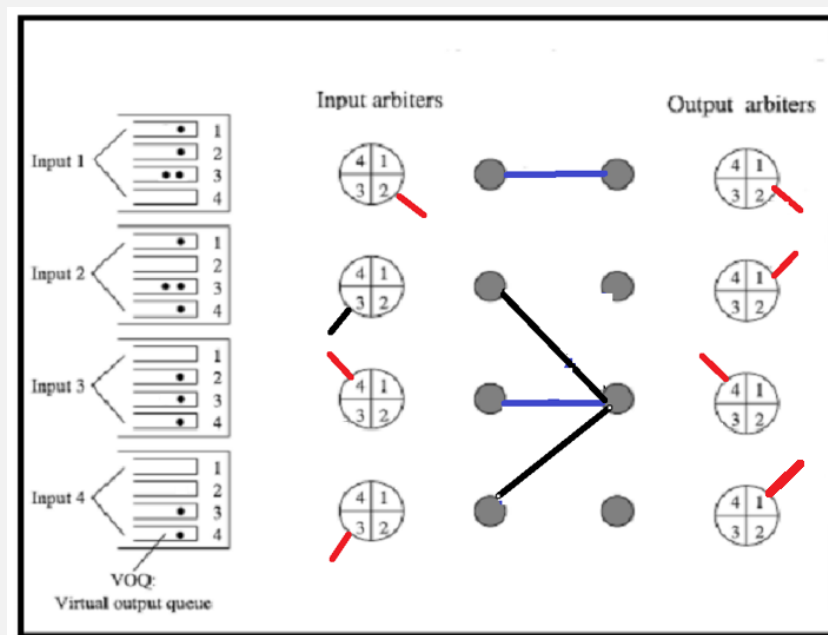


Figure 14: Step 1

پاسخ

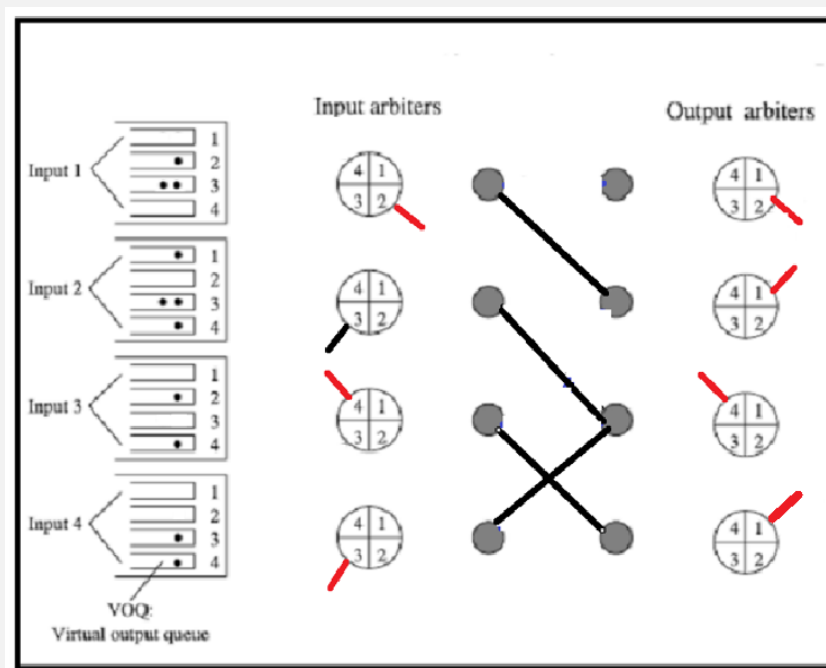


Figure 15: Step 2

## Phase 2 - Iteration 1

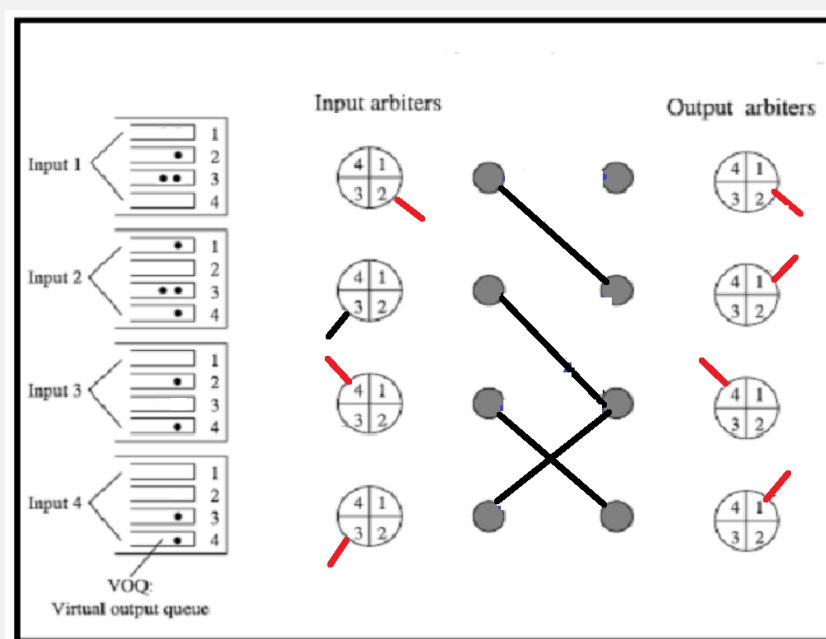


Figure 16: Step 1

پاسخ

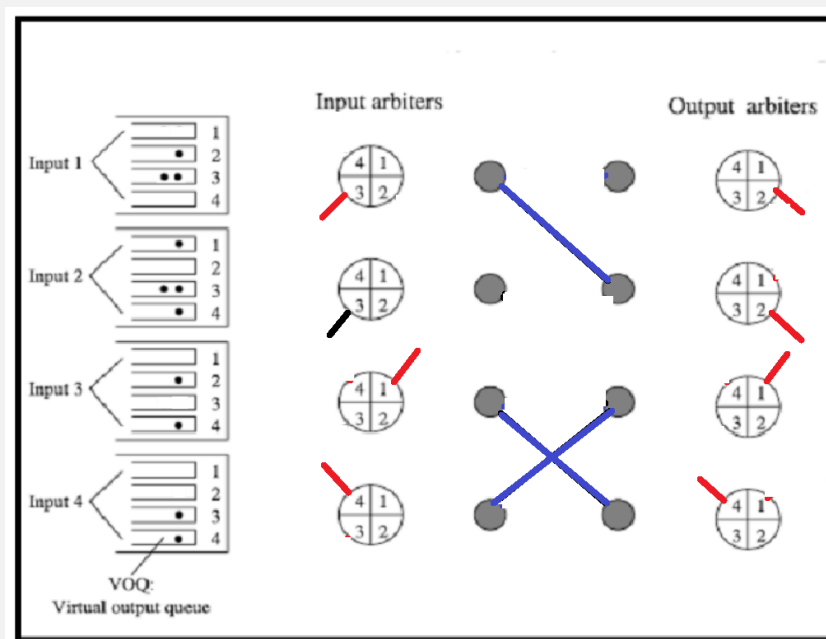


Figure 17: Step 2

## Phase 2 - Iteration 1

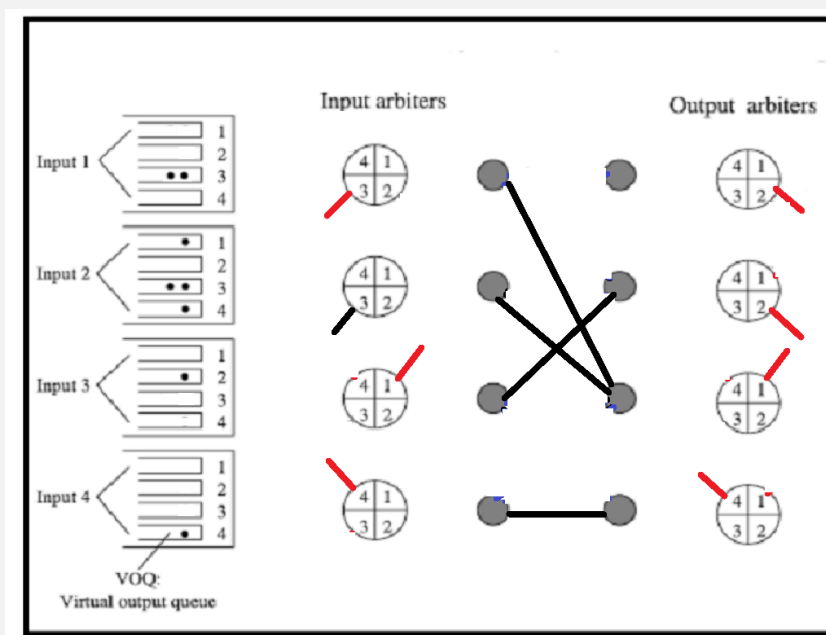


Figure 18: Step 1

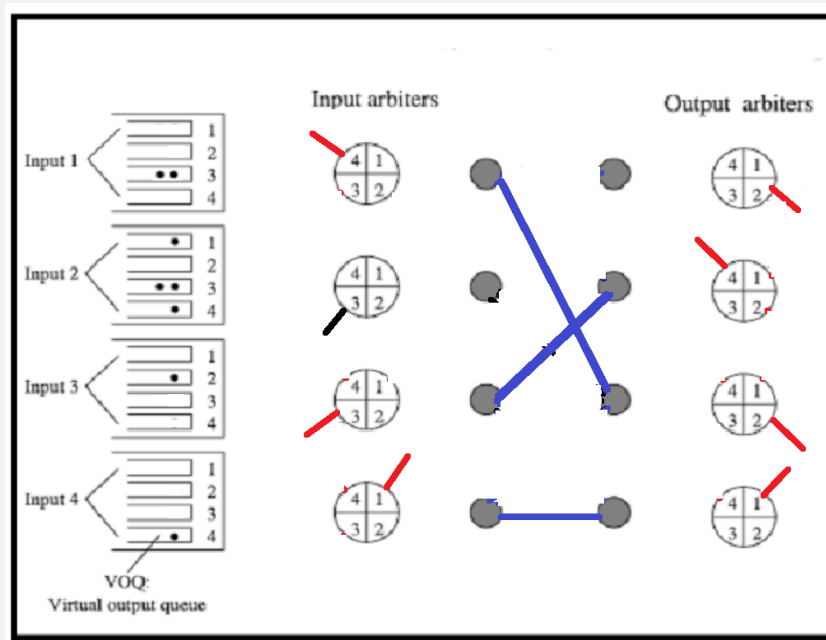


Figure 19: Step 2

### Phase 2 - Iteration 1

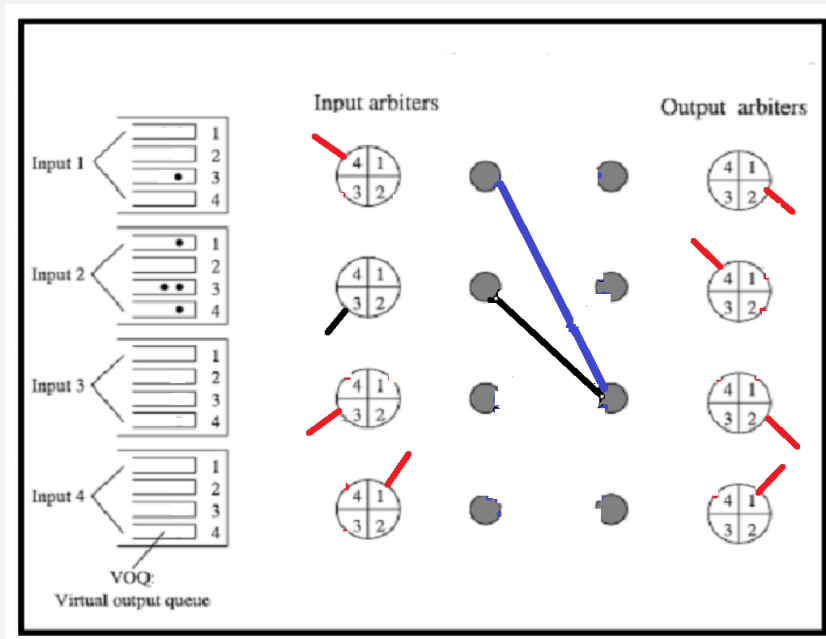


Figure 20: Step 1

پاسخ

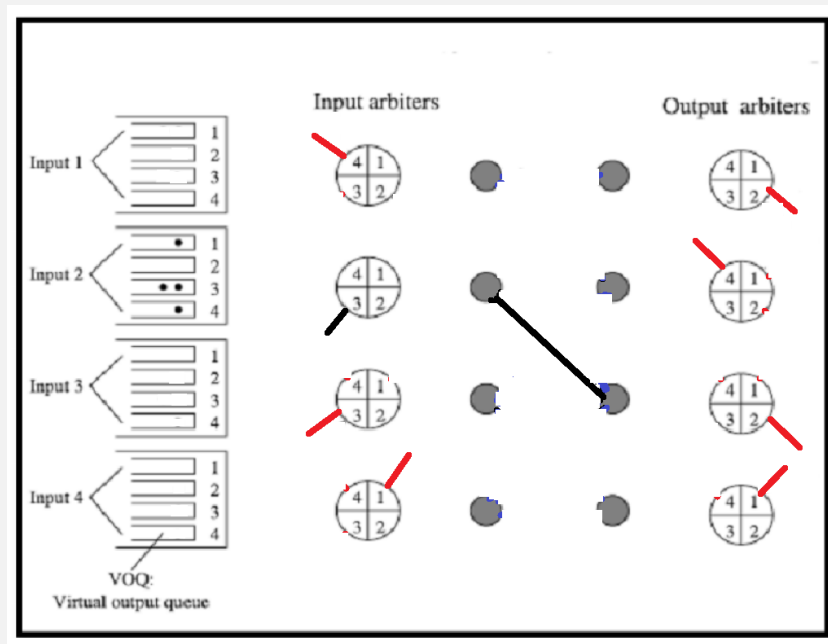


Figure 21: Step 2



## سوال ششم

- (الف) مزایا و معایب سوئیچ‌های Banyan را شرح دهید.

پاسخ

۱. مزایا:

- کارایی بالا: Banyan به دلیل طراحی چندمرحله‌ای، تأخیر کمتری نسبت به سوئیچ‌های تک‌مرحله‌ای دارد.
- پیاده‌سازی ساده: معماری ساده‌ای داشته و نیاز به اجزای پیچیده ندارد.
- قابلیت مقیاس‌پذیری: امکان گسترش اندازه سوئیچ با اضافه کردن مراحل یا گره‌ها وجود دارد.
- حداقل مسیریابی: ساختار مرتب‌شده‌ای دارد که مسیریابی را آسان و با حداقل تأخیر ممکن می‌سازد.

۲. معایب:

- بلاک شدن داخلی (Internal Blocking): اگر چند بسته بخواهند از یک لینک مشترک استفاده کنند، ممکن است بلاک شدن رخ دهد.
- عدم تحمل خطا: خرابی یک گره یا لینک می‌تواند کل سیستم را مختل کند.
- الگوهای ترافیکی محدود: الگوهای خاص ترافیک ممکن است بهره‌وری و عملکرد را کاهش دهند.
- پیچیدگی در کنترل ترافیک: برای جلوگیری از بلاک شدن داخلی، به کنترل‌کننده‌های پیچیده نیاز است.

- (ب) یک سوئیچ Banyan  $16 \times 16$  رسم کنید که شامل Shuffle و Unshuffled باشد.

پاسخ

سوئیچ Banyan با اندازه  $16 \times 16$  شامل چهار مرحله است زیرا  $16 = 4^2$  و هر مرحله از سوئیچ‌های  $2 \times 2$  تشکیل شده است.

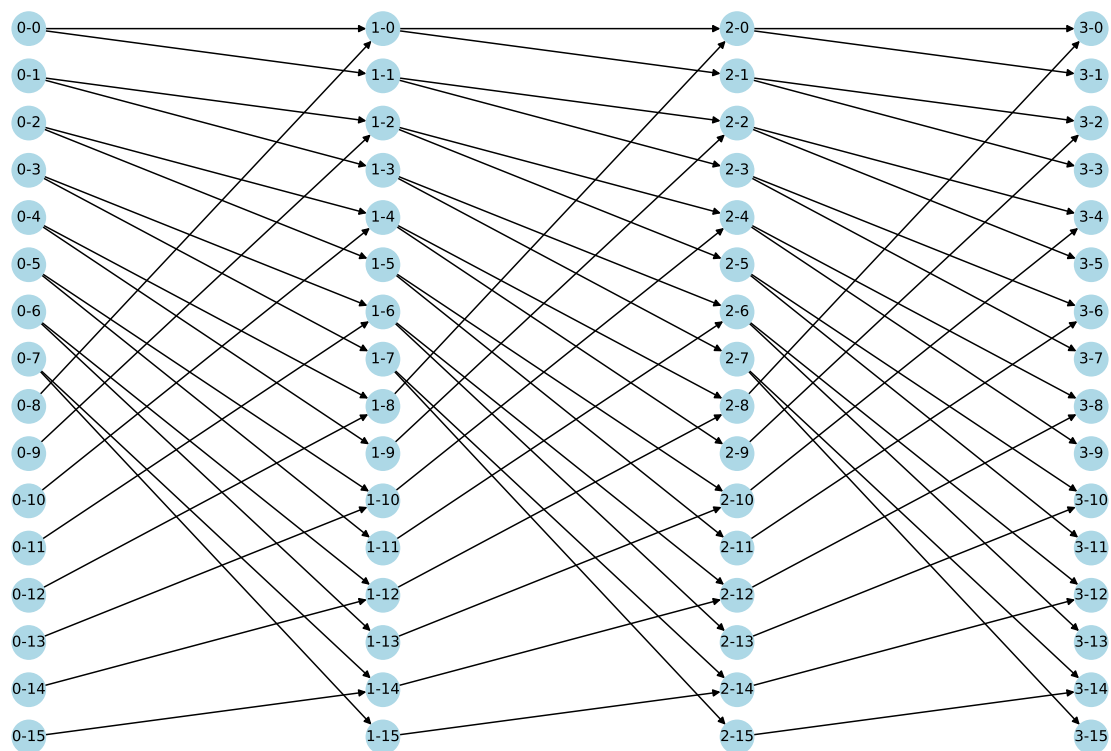
۱. تعداد مراحل برابر با  $\log_2(16) = 4$  است.

۲. در هر مرحله، ۸ سوئیچ  $2 \times 2$  مورد نیاز است.

۳. اتصالات Shuffle و Unshuffle به این صورت انجام می‌شود:

- Shuffle: خروجی  $i$  به ورودی  $(i \times 2) \bmod 16$  وصل می‌شود.

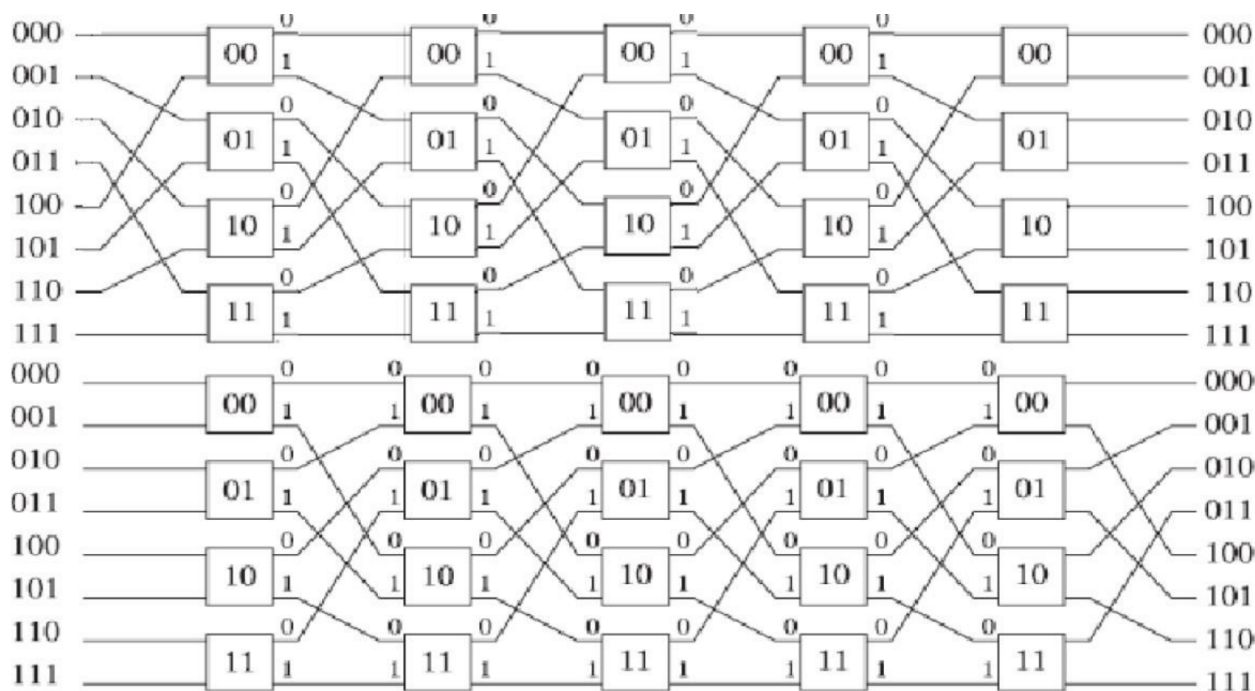
- Unshuffle: خروجی  $i$  به ورودی  $(i/2)$  یا  $i/2 + 8$  (برای اندیس‌های فرد) متصل می‌شود.



## سوال هفتم

به ازای حالت‌های زیر نحوه خروج بسته‌ها از سوئیچ را مشخص کنید.

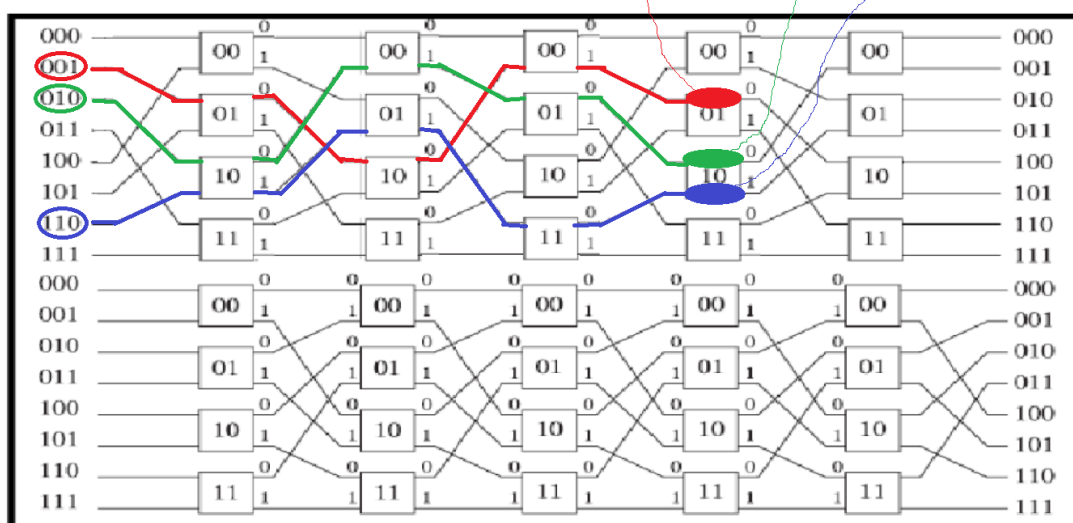
- A)  $001 \rightarrow 000$ ,  $100 \rightarrow 001$   
 B)  $110 \rightarrow 110$ ,  $100 \rightarrow 111$   
 C)  $010 \rightarrow 011$ ,  $110 \rightarrow 001$



شکل ۲۲: شکل مورد نظر

پاسخ

- A)  $001 \rightarrow 000$ ,  $100 \rightarrow 001$   
 B)  $110 \rightarrow 110$ ,  $100 \rightarrow 111$   
 C)  $010 \rightarrow 011$ ,  $110 \rightarrow 001$



## سوال هشتم

اجزای یک سوئیچ OpenFlow نسخه 5.1 را نشان دهید و هر کدام را شرح دهید.

پاسخ

۱. **Flow Table (جدول جریان):** این جدول، قوانین مربوط به جریان‌ها را ذخیره کرده و تصمیم‌گیری‌های لازم برای بسته‌های ورودی را انجام می‌دهد.

- **Match Fields:** فیلدهای تطبیق مانند آدرس IP و شماره پورت.
- **Actions:** اقداماتی مانند ارسال به پورت مشخص یا حذف بسته.
- **Counters:** شمارنده‌هایی برای ثبت تعداد و حجم بسته‌های پردازش‌شده.

۲. **Group Table (جدول گروه):** برای انجام عملیات پیشرفته‌تر مانند Multicast یا Load Balancing.

- تعریف گروه‌هایی از اقدامات.
- ارسال بسته به چندین مقصد به صورت همزمان.

۳. **Meter Table (جدول اندازه‌گیری):** مدیریت پهنای باند و اعمال سیاست‌های QoS.

- اندازه‌گیری نرخ جریان داده.
- اولویت‌بندی جریان‌ها.

۴. **Packet Buffer (بافر بسته):** ذخیره موقت بسته‌هایی که در انتظار پردازش یا ارسال به کنترل‌کننده هستند.

۵. **OpenFlow Channel (کانال ارتباطی):** ارتباط بین سوئیچ و کنترل‌کننده SDN.

- ارسال و دریافت پیام‌های کنترل.
- تضمین ارتباط امن.

۶. **Pipeline (پایپ‌لاین):** مجموعه‌ای از جدول‌های جریان که به صورت متوالی پردازش می‌شوند.

۷. **Statistics Collection (جمع‌آوری آمار):** جمع‌آوری آمار مربوط به جریان‌ها، پورت‌ها و پهنای باند.

۸. **Secure Channel (کانال امن):** ارتباط امن بین کنترل‌کننده و سوئیچ با استفاده از رمزنگاری.

## سوال نهم

معیارهای ارزیابی سوئیچ‌های کنونی و سوئیچ‌های نسل جدید SDN را با هم مقایسه کنید.

پاسخ

## ۱. معماری سوئیچینگ

- سوئیچ‌های سنتی: از معماری سخت‌افزاری ثابت و اختصاصی استفاده می‌کنند. تصمیم‌گیری‌ها در سطح سوئیچ و توسط سخت‌افزارهای داخلی انجام می‌شود. انعطاف‌پذیری کمی دارند.
- سوئیچ‌های SDN: دارای معماری نرم‌افزارمحور هستند. تصمیم‌گیری‌ها توسط کنترل‌کننده مرکزی انجام می‌شود. انعطاف‌پذیری بسیار بالایی دارند، چرا که قوانین و سیاست‌ها به‌صورت دینامیک توسط نرم‌افزار تنظیم می‌شوند.

## ۲. مدیریت و کنترل

- سوئیچ‌های سنتی: مدیریت و کنترل به‌صورت توزیع‌شده انجام می‌شود. نیازمند پیکربندی دستی و زمان‌بر هستند. تغییرات در مقیاس بزرگ دشوار است.
- سوئیچ‌های SDN: مدیریت و کنترل از طریق کنترل‌کننده مرکزی انجام می‌شود. پیکربندی به‌صورت خودکار و از طریق رابط‌های برنامه‌نویسی (API) صورت می‌گیرد. تغییرات و بروزرسانی‌ها سریع و کارآمد است.

## ۳. عملکرد و مقیاس‌پذیری

- سوئیچ‌های سنتی: عملکرد به سخت‌افزار وابسته است. برای مدیریت ترافیک بالا نیازمند سخت‌افزارهای گران‌قیمت هستند. مقیاس‌پذیری محدود است.
- سوئیچ‌های SDN: امکان مدیریت ترافیک بهینه از طریق نرم‌افزار وجود دارد. مقیاس‌پذیری بالا به دلیل کنترل مرکزی. نیاز به سخت‌افزارهای پیچیده کمتری دارند.

## ۴. انعطاف‌پذیری و قابلیت برنامه‌ریزی

- سوئیچ‌های سنتی: قوانین و سیاست‌ها ثابت و سخت‌افزاری هستند. برنامه‌ریزی و تغییرات محدود است.
- سوئیچ‌های SDN: بسیار انعطاف‌پذیر و قابل برنامه‌ریزی از طریق نرم‌افزار. امکان تعریف و اجرای قوانین جدید بدون تغییر در سخت‌افزار.

## ۵. امنیت

- سوئیچ‌های سنتی: امنیت به‌صورت محلی و در سطح هر سوئیچ مدیریت می‌شود. آسیب‌پذیری بیشتر در برابر حملات پیچیده.
- سوئیچ‌های SDN: امنیت به‌صورت متمرکز و توسط کنترل‌کننده مدیریت می‌شود. قابلیت به‌روزرسانی سریع قوانین امنیتی.

## پاسخ

## ۶. هزینه

- سوئیچ‌های سنتی: هزینه بالای سخت‌افزار. هزینه‌های عملیاتی و نگهداری بیشتر.
- سوئیچ‌های SDN: کاهش هزینه به دلیل استفاده از سخت‌افزارهای عمومی (Commodity Hardware). کاهش هزینه‌های عملیاتی با مدیریت ساده‌تر.

## ۷. جمع‌آوری و تحلیل داده

- سوئیچ‌های سنتی: جمع‌آوری داده‌ها محدود و زمان‌بر است. تحلیل داده‌ها نیازمند ابزارهای جداگانه است.
- سوئیچ‌های SDN: جمع‌آوری داده‌ها به‌صورت متمرکز و در زمان واقعی انجام می‌شود. تحلیل داده‌ها ساده‌تر و کارآمدتر است.

## سوال دهم

معماری سوئیچ‌های نسل جدید Huawei و Intel را بررسی کرده و نوع پیاده‌سازی و ویژگی‌های سوئیچ‌های OpenFlow مانند عملیات Pipelining را شرح دهید.

## سوال یازدهم

معماری سوئیچ‌های مبتنی بر چارچوب ForCES را بررسی کنید و ویژگی‌های این چارچوب را شرح دهید.



## سوال دوازدهم

تفاوت‌های چارچوب ForCES و OpenFlow را شرح دهید.