

جواب سوال 1

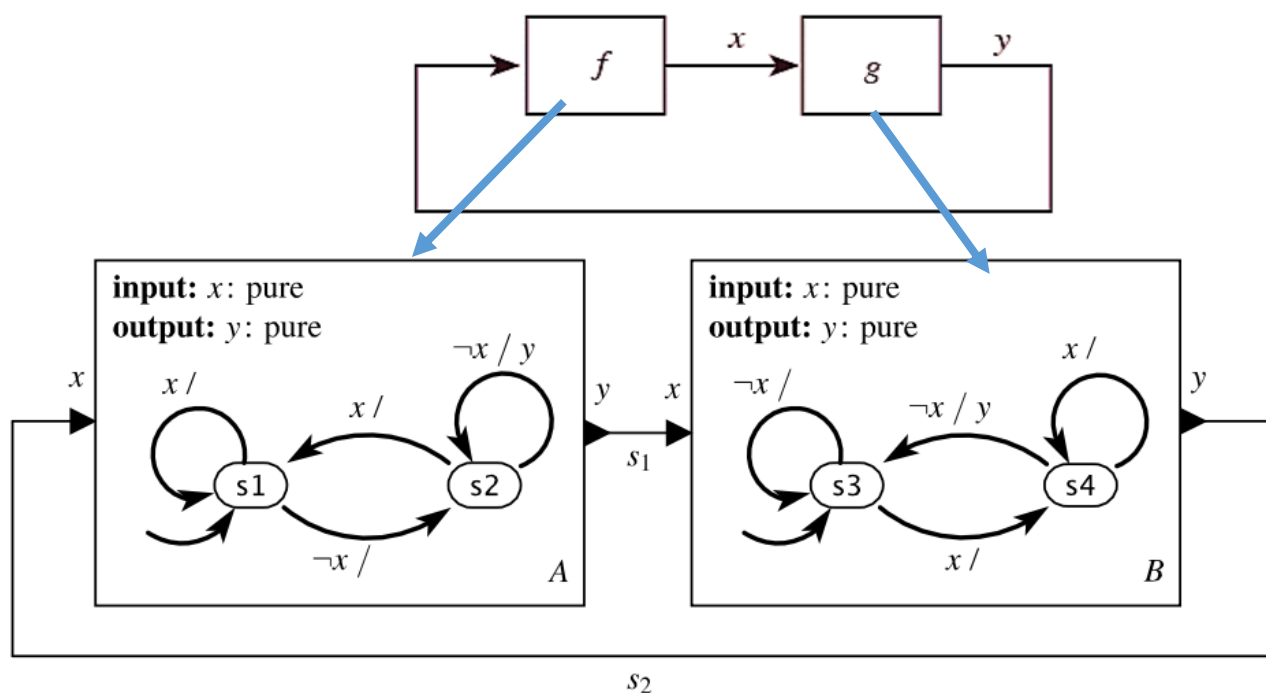
(الف)

برای بررسی خوش ساخت باید fixed point را برای تمام حالات بررسی کنیم. اگر بتوانیم برای هر مدل fixed point پیدا کنیم میتواند گفت که مدل خوش ساخت است

- A more complicated problem, involving two equations, is to find x and y , so that:

$$x = f(y), \text{ and } y = g(x).$$

- The analogous feedback composition has two state machines in feedback as:



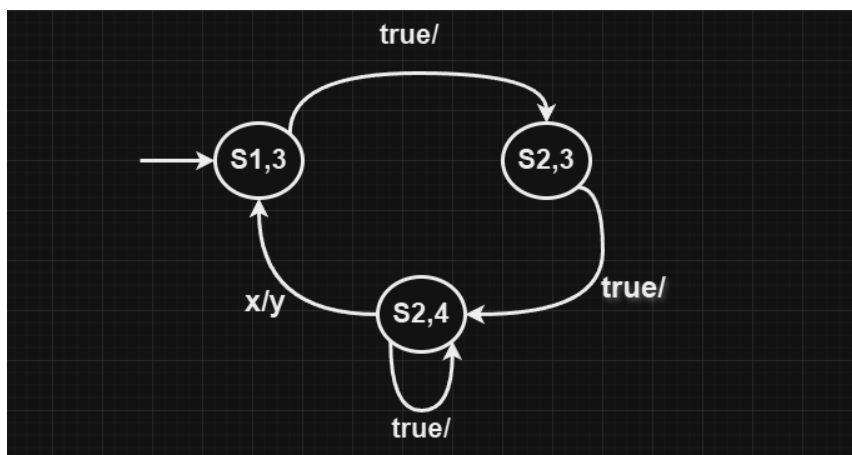
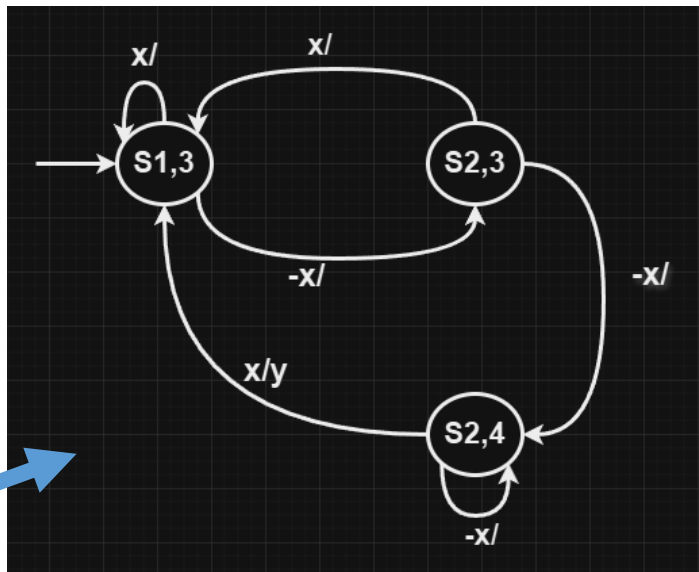
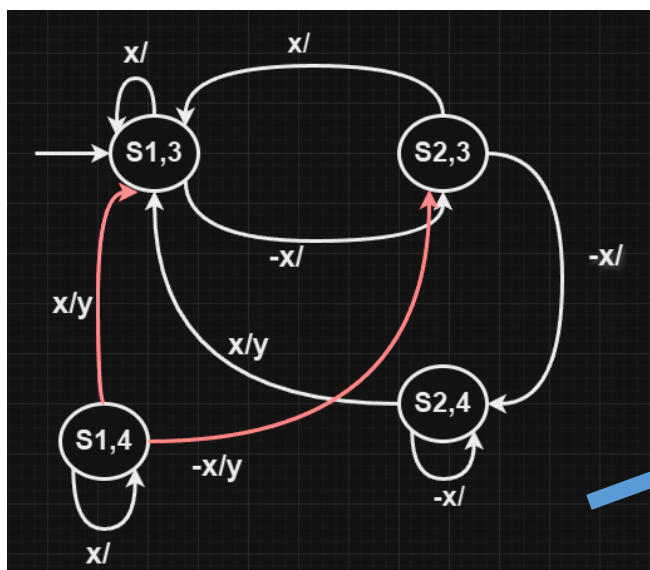
(بررسی Fixed Point)

- 1: ($x = \text{absent}$) \rightarrow In State **S1**, ($S1 \rightarrow S2$) \rightarrow ($y = \text{absent}$) \rightarrow In State **S3**, ($S3 \rightarrow S3$) $\rightarrow y = \text{absent}$
- 2: ($x = \text{absent}$) \rightarrow In State **S1**, ($S1 \rightarrow S2$) \rightarrow ($y = \text{absent}$) \rightarrow In State **S4**, ($S4 \rightarrow S3$) $\rightarrow y = \text{Present}$
- 3: ($x = \text{Present}$) \rightarrow In State **S1**, ($S1 \rightarrow S1$) \rightarrow ($y = \text{absent}$) \rightarrow In State **S3**, ($S3 \rightarrow S3$) $\rightarrow y = \text{absent}$
- 4: ($x = \text{Present}$) \rightarrow In State **S1**, ($S1 \rightarrow S1$) \rightarrow ($y = \text{absent}$) \rightarrow In State **S4**, ($S4 \rightarrow S3$) $\rightarrow y = \text{Present}$
- 5: ($x = \text{absent}$) \rightarrow In State **S2**, ($S2 \rightarrow S2$) \rightarrow ($y = \text{persent}$) \rightarrow In State **S3**, ($S3 \rightarrow S4$) $\rightarrow y = \text{absent}$
- 6: ($x = \text{absent}$) \rightarrow In State **S2**, ($S2 \rightarrow S2$) \rightarrow ($y = \text{persent}$) \rightarrow In State **S4**, ($S4 \rightarrow S4$) $\rightarrow y = \text{absent}$
- 7: ($x = \text{Present}$) \rightarrow In State **S2**, ($S2 \rightarrow S1$) \rightarrow ($y = \text{absent}$) \rightarrow In State **S3**, ($S3 \rightarrow S3$) $\rightarrow y = \text{absent}$
- 8: ($x = \text{Persent}$) \rightarrow In State **S2**, ($S2 \rightarrow S1$) \rightarrow ($y = \text{absent}$) \rightarrow In State **S4**, ($S4 \rightarrow S3$) $\rightarrow y = \text{Persent}$

حالت های 6 و 8 شرط یکتا بودن Fixed Point را برهم میزند و مدل Ill-formed است

ب) چون مدل خوش ساخت نیست پس برساختنی هم نیست.

ج) State S1,4 قابل درسترس نیست پس این اسیت را حذف می کنیم. پس داریم:



د

$x_{1,3}(\text{absent}) \Rightarrow S1 \rightarrow \text{absent} \Rightarrow S2 \rightarrow \text{absent}$
 $x_{1,4}(\text{persent}) \Rightarrow S1 \rightarrow \text{absent} \Rightarrow S2 \rightarrow \text{absent}$
 $x_{2,3}(\text{absent}) \Rightarrow S1 \rightarrow \text{persent} \Rightarrow S2 \rightarrow \text{absent}$
 $x_{2,4}(\text{persent}) \Rightarrow S1 \rightarrow \text{absent} \Rightarrow S2 \rightarrow \text{persent}$

جواب سوال 2

الف) یک FSM مسطح معادل رفتار آن ارائه کنید.

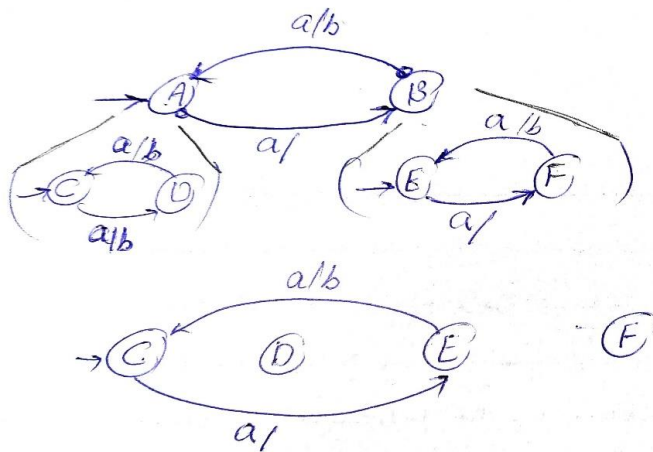
ب) این ماشین چه حالت‌های دسترس پذیری دارد؟

چون ماشین حالت از نوع Preemptive می باشد هیچگاه وارد حالت های D و F نمی شود.

پس با هر بار ورودی به حالت های C و E دسترسی پذیری دارد.

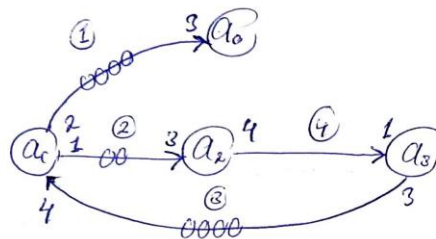
ج) رفتار ورودی/خروجی این ماشین را در یک جمله توصیف کنید.

این ماشین حالت با هر بار گرفتن ورودی a یک در میان b خروجی میدهد.



جواب سوال 3

الف) ماتریس وقوع (ماتریس توپولوژی) گراف Γ را استخراج کنید و مرتبه آن را مشخص نمایید.



$$q_A p_C = q_B c_C$$

Producer Consumer

$$\textcircled{1} \quad a_1 \times 2 = a_0 \times 3 \rightarrow -3a_0 + 2a_1 + 0a_2 + 0a_3 = 0$$

$$\textcircled{2} \quad a_1 \times 1 = a_2 \times 3 \rightarrow 0a_0 + 1a_1 - 3a_2 + 0a_3 = 0$$

$$\textcircled{3} \quad a_3 \times 3 = a_2 \times 4 \rightarrow 0a_0 + 0a_1 - 4a_2 + 3a_3 = 0$$

$$\textcircled{4} \quad a_2 \times 4 = a_3 \times 1 \rightarrow 0a_0 + 0a_1 + 4a_2 - 1a_3 = 0$$

$$\Gamma = \begin{bmatrix} -3 & 2 & 0 & 0 \\ 0 & 1 & -3 & 0 \\ 0 & -4 & 0 & 3 \\ 0 & 0 & 4 & -1 \end{bmatrix}$$

$$\text{Rank}(\Gamma) = 3$$

$$\left(\begin{array}{cccc|c} -3 & 2 & 0 & 0 & 0 \\ 0 & 1 & -3 & 0 & 0 \\ 0 & -4 & 0 & 3 & 0 \\ 0 & 0 & 4 & -1 & 0 \end{array} \right) \xrightarrow{\times(4)} \left(\begin{array}{cccc|c} -3 & 2 & 0 & 0 & 0 \\ 0 & 1 & -3 & 0 & 0 \\ 0 & 0 & -12 & 3 & 0 \\ 0 & 0 & 4 & -1 & 0 \end{array} \right) \xrightarrow{R_3 - (-4) \cdot R_2 \rightarrow R_3} \left(\begin{array}{cccc|c} -3 & 2 & 0 & 0 & 0 \\ 0 & 1 & -3 & 0 & 0 \\ 0 & 0 & -12 & 3 & 0 \\ 0 & 0 & 4 & -1 & 0 \end{array} \right) \xrightarrow{\times(\frac{1}{3})} \left(\begin{array}{cccc|c} -3 & 2 & 0 & 0 & 0 \\ 0 & 1 & -3 & 0 & 0 \\ 0 & 0 & -4 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{array} \right) \xrightarrow{R_4 - (-\frac{1}{3}) \cdot R_3 \rightarrow R_4} \left(\begin{array}{cccc|c} -3 & 2 & 0 & 0 & 0 \\ 0 & 1 & -3 & 0 & 0 \\ 0 & 0 & -4 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{array} \right)$$

ب) مقدار q با کوچکترین مقادیر صحیح مثبت را به دست آورید به گونه ای که $\Gamma q = 0$ باشد.

$$\Gamma q = 0$$

$$-3x_1 + 2x_2 = 0$$

$$x_2 - 3x_3 = 0$$

$$-4x_2 + 3x_4 = 0$$

$$4x_3 - 1x_4 = 0$$

$$\text{if } x_3 = 1 \rightarrow \begin{cases} 4(1) - x_4 = 0 & x_4 = 4 \\ -4x_2 + 3(4) = 0 & x_2 = 3 \\ 3 - 3x_3 = 0 & x_3 = 1 \\ -3x_1 + 2(3) & x_1 = 2 \end{cases} \Rightarrow q = \begin{bmatrix} 2 \\ 3 \\ 1 \\ 4 \end{bmatrix}$$

ج) برای این گراف SDF یک زمانبندی تکرارشونده ترتیبی را به همراه شرایط اولیه بافرها مشخص کنید. حداقل مجموع بافرهای مورد نیاز برای اجرای زمانبندی به دست آمده چقدر است؟

$$\begin{array}{ccccccc} \text{Buffer} = \begin{bmatrix} 4 \\ 2 \\ 4 \\ 0 \end{bmatrix} & \xrightarrow{a_1} & \begin{bmatrix} 6 \\ 3 \\ 0 \\ 0 \end{bmatrix} & \xrightarrow{a_0} & \begin{bmatrix} 3 \\ 3 \\ 0 \\ 0 \end{bmatrix} & \xrightarrow{a_0} & \begin{bmatrix} 0 \\ 3 \\ 0 \\ 0 \end{bmatrix} & \xrightarrow{a_2} & \begin{bmatrix} 0 \\ 0 \\ 0 \\ 4 \end{bmatrix} & \xrightarrow{a_3} & \begin{bmatrix} 0 \\ 0 \\ 3 \\ 3 \end{bmatrix} & \xrightarrow{a_3} & \begin{bmatrix} 0 \\ 0 \\ 6 \\ 2 \end{bmatrix} \\ q = \begin{bmatrix} 2 \\ 3 \\ 1 \\ 4 \end{bmatrix} & \xrightarrow{\text{Fire}} & \begin{bmatrix} 2 \\ 2 \\ 1 \\ 4 \end{bmatrix} & \xrightarrow{\text{Fire}} & \begin{bmatrix} 1 \\ 2 \\ 1 \\ 4 \end{bmatrix} & \xrightarrow{\text{Fire}} & \begin{bmatrix} 0 \\ 2 \\ 4 \\ 4 \end{bmatrix} & \xrightarrow{\text{Fire}} & \begin{bmatrix} 0 \\ 2 \\ 0 \\ 4 \end{bmatrix} & \xrightarrow{\text{Fire}} & \begin{bmatrix} 0 \\ 2 \\ 0 \\ 3 \end{bmatrix} & \xrightarrow{\text{Fire}} & \begin{bmatrix} 0 \\ 2 \\ 0 \\ 2 \end{bmatrix} \end{array}$$

$$\begin{array}{cccc} \begin{bmatrix} 0 \\ 0 \\ 9 \\ 1 \end{bmatrix} & \xrightarrow{a_3} & \begin{bmatrix} 0 \\ 0 \\ 12 \\ 0 \end{bmatrix} & \xrightarrow{a_1} & \begin{bmatrix} 2 \\ 1 \\ 8 \\ 0 \end{bmatrix} & \xrightarrow{a_1} & \begin{bmatrix} 4 \\ 2 \\ 4 \\ 0 \end{bmatrix} \\ \xrightarrow{\text{Fire}} & \begin{bmatrix} 0 \\ 2 \\ 0 \\ 1 \end{bmatrix} & \xrightarrow{\text{Fire}} & \begin{bmatrix} 0 \\ 2 \\ 0 \\ 0 \end{bmatrix} & \xrightarrow{\text{Fire}} & \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} & \xrightarrow{\text{Fire}} & \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \end{array}$$

Scheduled $(a_1, a_0, a_0, a_2, a_3, a_3, a_3, a_3, a_1, a_1)$

The Schedule sequence is $(a_1, a_0, a_0, a_2, a_3, a_3, a_3, a_3, a_1, a_1)$

$$\begin{aligned} \text{Minimum buffer for every edge is} &= \max(4, 6, 3, 0, 0, 0, 0, 0, 0, 2, 1) = 6 \\ &= \max(2, 3, 3, 3, 0, 0, 0, 0, 0, 1, 2) = 3 \\ &= \max(4, 0, 0, 0, 0, 3, 6, 9, 12, 8, 4) = 12 \\ &= \max(0, 0, 0, 0, 4, 3, 2, 1, 0, 0, 0) = 4 \end{aligned}$$

در نتیجه به حداقل بافرهای به سباز 4، 3، 6، 12 نیاز هست.

د) با فرض موجود بودن توابع $a_1()$ ، $a_2()$ ، $a_3()$ ، $a_4()$ ، که حلقه زمانبندی تکرار شونده گراف را به زبان C بنویسید. از آرایه ها برای پیاده سازی بافرها استفاده کنید.

```
#include <stdio.h>
void a0();
void a1();
void a2();
void a3();
void printBuffer();

// declare buffers
int buffer1[6];    // a1 -> a0
int buffer2[3];    // a1 -> a2
int buffer3[12];   // a3 -> a1
int buffer4[4];    // a2 -> a3

// pointer for writer and reader on buffers
int index_b1 = 4, index_b2 = 2, index_b3 = 4, index_b4 = 0;

int main() {

// initialization buffers
    for(int i = 0; i < 4; i++){
        buffer1[i] = 1;
    }
    for(int i = 0; i < 2; i++){
        buffer2[i] = 1;
    }

    for(int i = 0; i < 4; i++){
        buffer4[i] = 1;
    }

    // print valuse all buffer initia
    printf("frist valuse: ");
    printBuffer();

    while(true){
        // these functions update all buffer, wb and rb
        a1();
        printf("fire a1: ");
        printBuffer();

        a0();
        printf("fire a0: ");
        printBuffer();

        a0();
        printf("fire a0: ");
        printBuffer();
    }
}
```

```

        a2();
        printf("fire a2: ");
        printBuffer();

        a3();
        printf("fire a3: ");
        printBuffer();

        a3();
        printf("fire a3: ");
        printBuffer();

        a3();
        printf("fire a3: ");
        printBuffer();

        a3();
        printf("fire a3: ");
        printBuffer();

        a1();
        printf("fire a1: ");
        printBuffer();

        a1();
        printf("fire a3: ");
        printBuffer();
        printf("-----> End of Trun <-----");
        printf("\n-----\n");

        i++;
    }

    return 0;
}

void a0(){
// counsume
    for (int i = 0; i < 3 ; i++)
    {
        index_b1 = index_b1-1;
        buffer1[index_b1] = 0;
    }
}

void a1(){
// counsume

```

```

    for (int i = 0; i < 4 ; i++){
        index_b3 = index_b3-1;
        buffer3[index_b3] = 0;
    }

// Produce a1 to a0
    for (int i = 0; i < 2 ; i++){
        buffer1[index_b1] = 1;
        index_b1 = index_b1 + 1;
    }

// Produce a1 to a2
    for (int i = 0; i < 1 ; i++){
        buffer2[index_b2] = 1;
        index_b2 = index_b2 + 1;
    }
}

void a2(){
// counsume
    for (int i = 0; i < 3 ; i++){
        index_b2 = index_b2-1;
        buffer2[index_b2] = 0;
    }

// Produce a1 to a0
    for (int i = 0; i < 4 ; i++){
        buffer4[index_b4] = 1;
        index_b4 = index_b4 + 1;
    }
}

void a3(){
// counsume
    index_b4 = index_b4-1;
    buffer4[index_b4] = 0;

// Produce a3 to a1
    for (int i = 0; i < 3 ; i++){
        buffer3[index_b3] = 1;
        index_b3 = index_b3 + 1;
    }
}

void printBuffer(){

    printf("{%d, %d, %d, %d}\n", index_b1, index_b2, index_b3, index_b4);

}

```

```

frist valuse: {4, 2, 4, 0}
fire a1: {6, 3, 0, 0}
fire a0: {3, 3, 0, 0}
fire a0: {0, 3, 0, 0}
fire a2: {0, 0, 0, 4}
fire a3: {0, 0, 3, 3}
fire a3: {0, 0, 6, 2}
fire a3: {0, 0, 9, 1}
fire a3: {0, 0, 12, 0}
fire a1: {2, 1, 8, 0}
fire a3: {4, 2, 4, 0}
-----> End of Trun <-----
-----
fire a1: {6, 3, 0, 0}
fire a0: {3, 3, 0, 0}
fire a0: {0, 3, 0, 0}
fire a2: {0, 0, 0, 4}
fire a3: {0, 0, 3, 3}
fire a3: {0, 0, 6, 2}
fire a3: {0, 0, 9, 1}
fire a3: {0, 0, 12, 0}
fire a1: {2, 1, 8, 0}
fire a3: {4, 2, 4, 0}
-----> End of Trun <-----
-----

```

قسمت امتیازی :

با استفاده از امکانات domain dataflow سعی میکنیم شبیه سازی بکنیم. هر node را یک subsystem dataflow قرار میدهیم.

The screenshot shows the 'Block Data Types' dialog box for the 'FIR Decimation (mask)' block. The 'Column View' tab is active, displaying a table of block data types. The 'Main' tab is also visible, showing parameters for the FIR filter.

Name	BlockType	OutDataTypeStr	OutMin	OutMax	LockScale	DataType	Min	Max	AccumDataTypeStr	ParamDataTypeStr	ParamMin	ParamMax
FIR a1_a0	S-Function				<input checked="" type="checkbox"/>							
FIR a1_a2	S-Function				<input checked="" type="checkbox"/>							
FIR a2_a3	S-Function				<input checked="" type="checkbox"/>							
FIR a3_a1	S-Function				<input checked="" type="checkbox"/>							

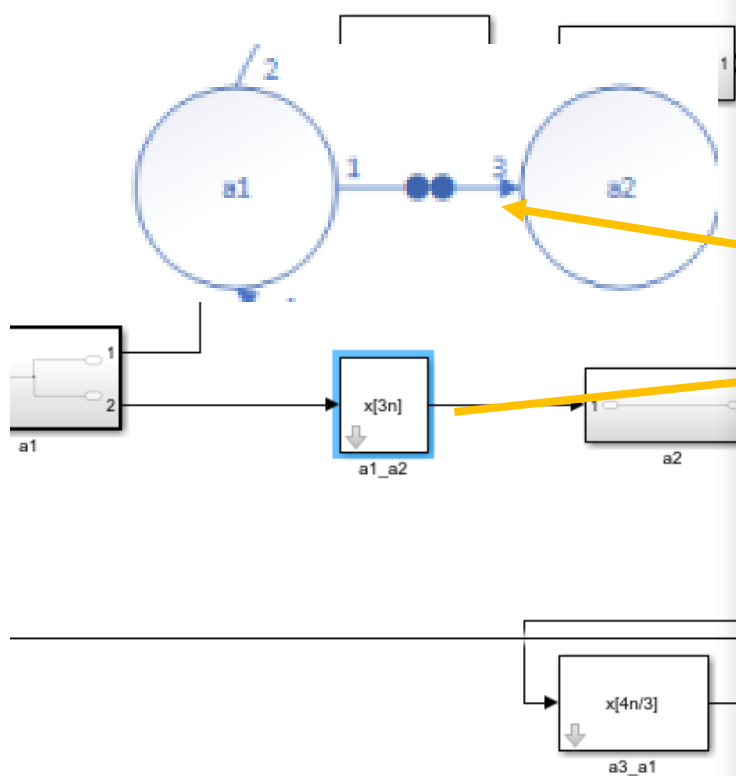
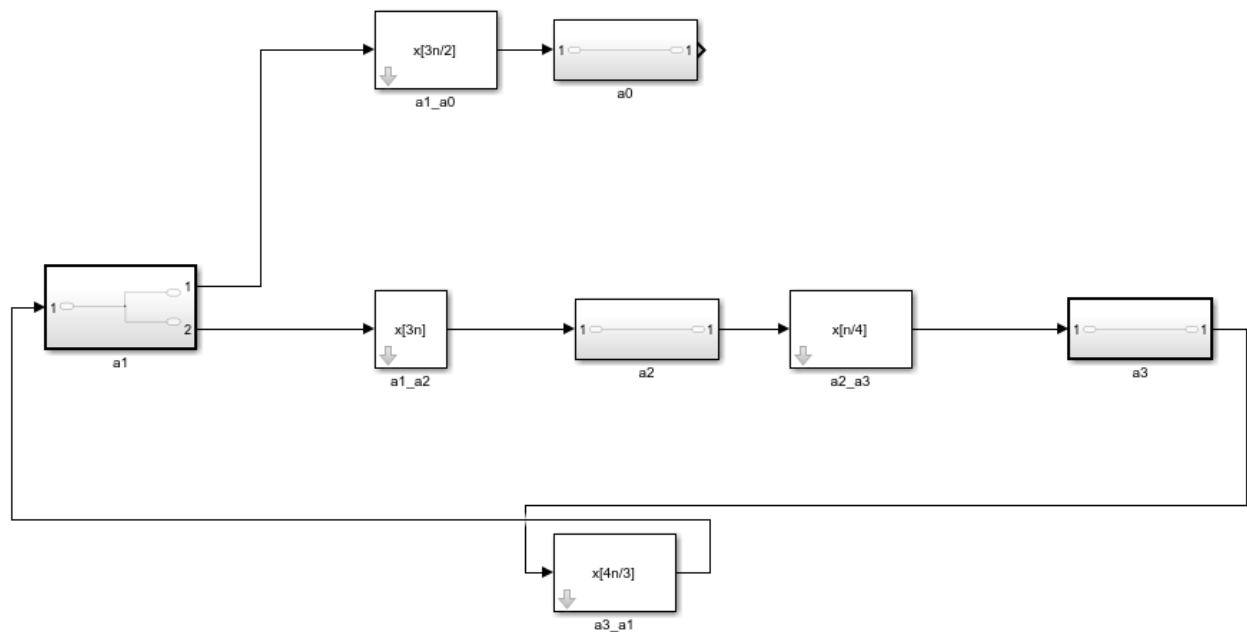
FIR Decimation (mask) (link)
Apply an FIR filter to the input signal, then downsample the signal by an integer-valued factor. The block uses a polyphase filter structure.

Coefficient source
☒ Dialog parameters
☐ Input port
☐ Filter object
☐ Auto

Main Data Types

Parameters
 FIR filter coefficients: 3
 Decimation factor: 4
 Filter structure: Direct form
 Input processing: Columns as channels (frame based)
 Rate options: Enforce single-rate processing
☐ Allow arbitrary frame length for fixed-size input signals

[View Filter Re](#)



Block Parameters: a1_a2

FIR Rate Conversion (mask) (link)

Upsample the input signal by an integer-valued factor, apply factor. The block implements the FIR filter using a polyphase

Coefficient source

- ☐ Dialog parameters
- ☐ Filter object
- ☒ Auto

Main Data Types

Parameters

Interpolation factor:

Decimation factor:

Rate options

☒ Allow multirate processing