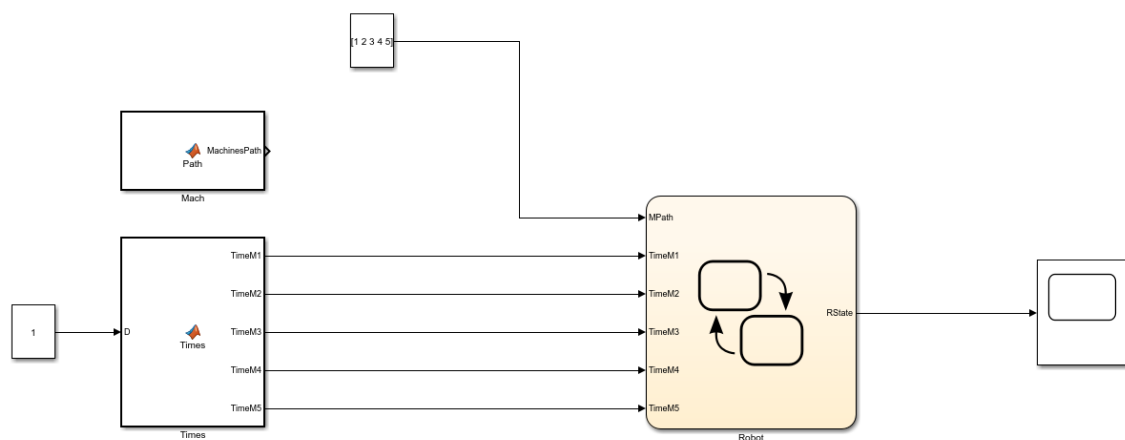


Event-Based Control

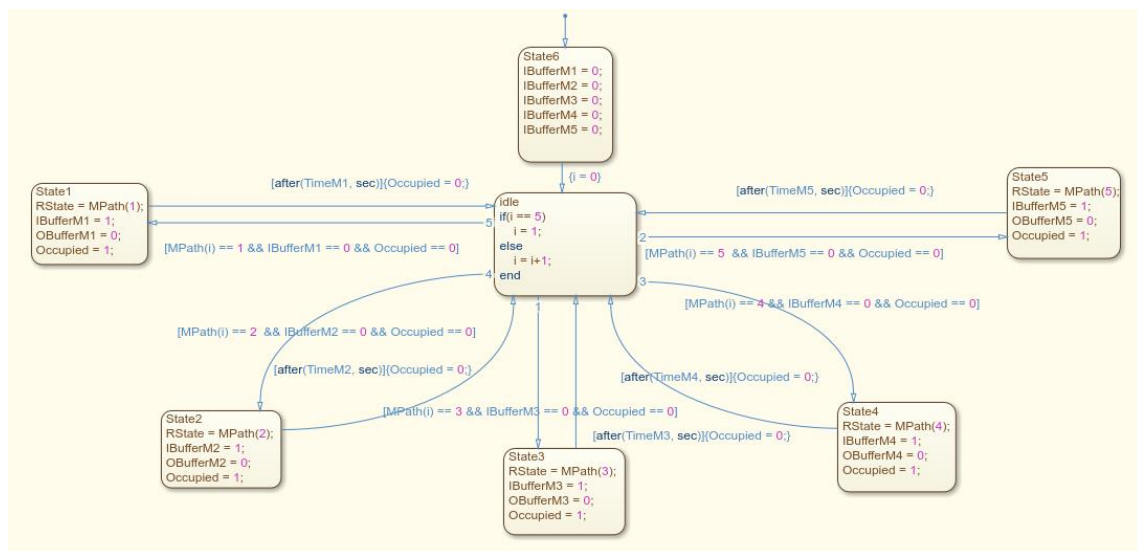
Fifth Task

The fifth task is to develop a controller that manipulates one transport robot and 5 machines, each one with different execution times. Each machine has an input and an output buffer.

To do this I made a program with SimuLink and StateFlow scripts. This program consists in one SimuLink module, one StateFlow module and 2 function. The program as 2 ways of working, by defining a constant (shown in the next figure) or by connecting the function and disconnecting the array. The global graph is the following:



To control the transport robot, I used the following module:



This graph only allows the transport robot to place the object to the next state after the machine finished the process. The machine follows the path given by the following function:

```
function MachinesPath = Path()
```

```

j = 0;
NrProcesses = 5;
MachinesPath = zeros(1,NrProcesses);
for k=1:NrProcesses
    MachinesPath(1,k) = randi(5,1);
end

for i=1:NrProcesses-1
    if(MachinesPath(1,i) == MachinesPath(1,i+1))
        while j == 0
            A = randi(5,1);
            if(A ~= MachinesPath(1,i+1))
                if(i > 1)
                    if(A ~= MachinesPath(1,i-1))
                        MachinesPath(1,i) = A;
                        j = 1;
                    end
                else
                    p = 2;
                    if(A ~= MachinesPath(1,p-1))
                        MachinesPath(1,i) = A;
                        j = 1;
                    end
                end
            end
        end
        j = 0;
    end
end
end

```

This function creates an array with 5 positions (Number of Processes), each position has random integers from 1 to 5 (Number of Machines). This function also as the capability of making sure there aren't 2 equal consecutive machines in the Machines Path array.

For example, if we run the first part of the code and we get an array of [5 5 3 5 1], the second part of the code will convert it to [2 5 3 5 1]

For the time control I also have a MATLAB Function, the function is the following:

```

function [TimeM1, TimeM2, TimeM3, TimeM4, TimeM5] =
Times(D)

```

```

Upper = D+0.1*D;
Lower = D-0.1*D;
Range = Upper-Lower;

```

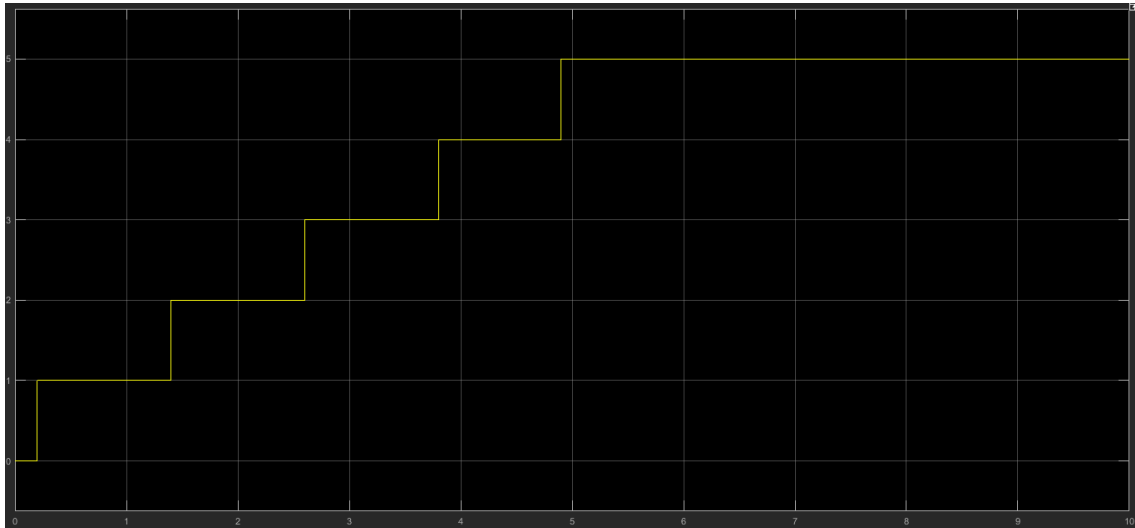
```

TimeM1 = Range*rand(1,1)+Lower;
TimeM2 = Range*rand(1,1)+Lower;
TimeM3 = Range*rand(1,1)+Lower;
TimeM4 = Range*rand(1,1)+Lower;
TimeM5 = Range*rand(1,1)+Lower;

```

This function gives random values to the machine processing time between $[D-0.1*D, D+0.1*D]$.

Using the first way of working (with the constant [1 2 3 4 5]). The results I had were the following:

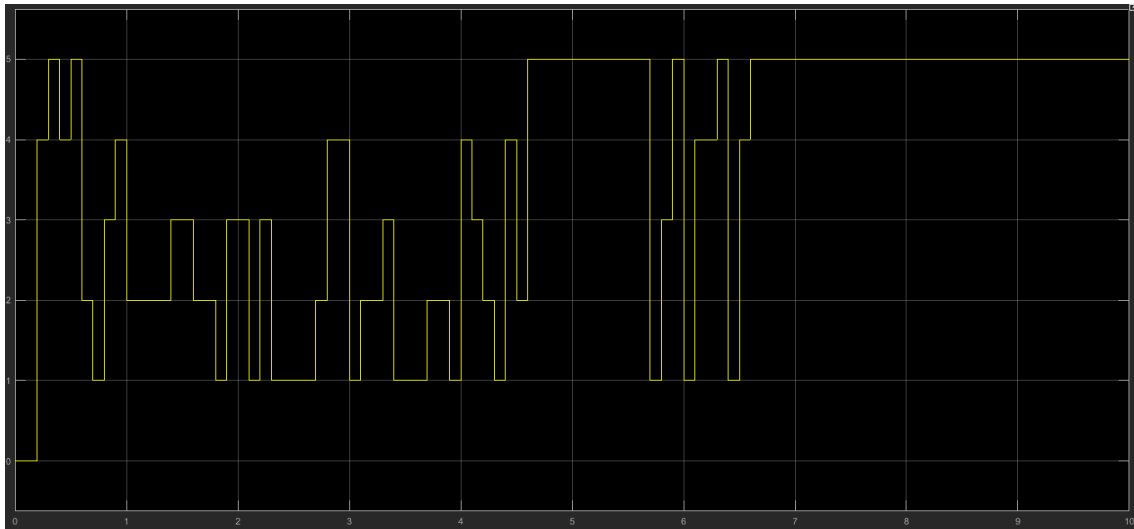


Using the tools from Matlab I was able to check the time of each process:

Machine	Time (s)
1	1.192
2	1.202
3	1.202
4	1.097
5	1.116

Of course, these values will be different every time I run the program because these values are defined by a random parameter.

When I use the second method of working (with the function) the results I got were the following:



This method doesn't work as well as the first one, because the function Mach is always running and always defining a different MachinesPath, I couldn't make the function to run just once.

In general we can see that the problem was well solved.