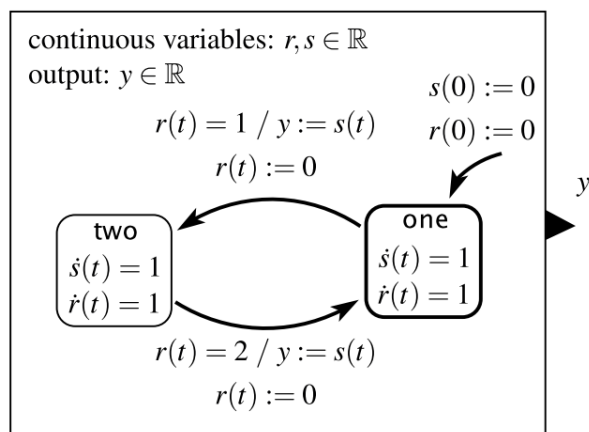


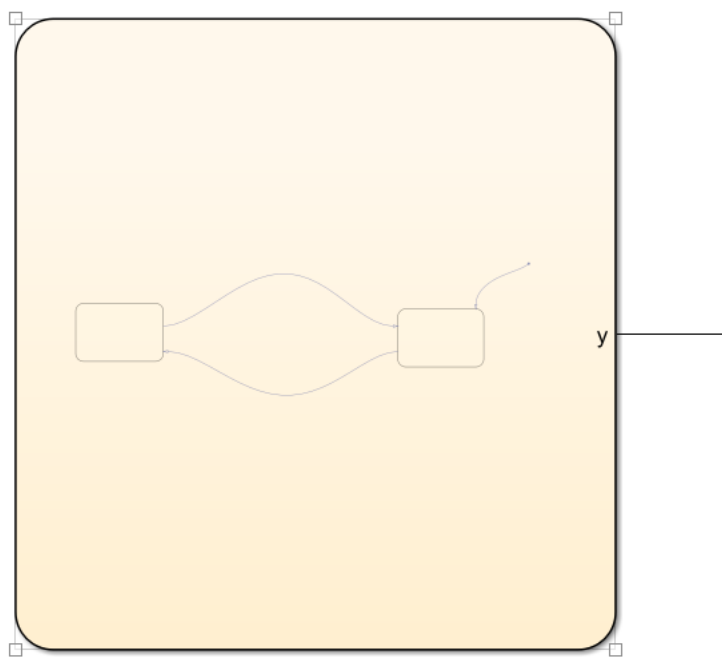
گزارش سوال دوم تمرین سری دوم

در این تمرین از ما خواسته شده مسئله دوم از فصل چهارم را پیاده سازی کنیم :

شکل کتاب به صورت زیر است :

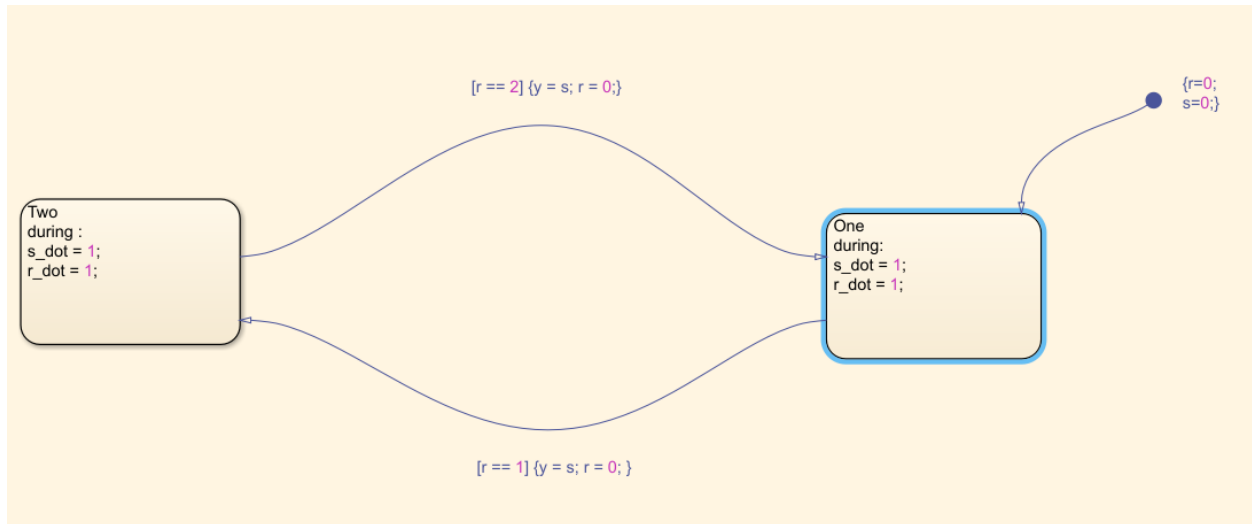


برای پیاده سازی state ها در Simulink از chart استفاده میکنیم :



Q2.a

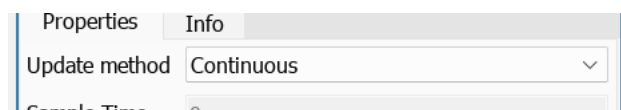
حال داخل chart ما state هارا اوکی میکنیم :



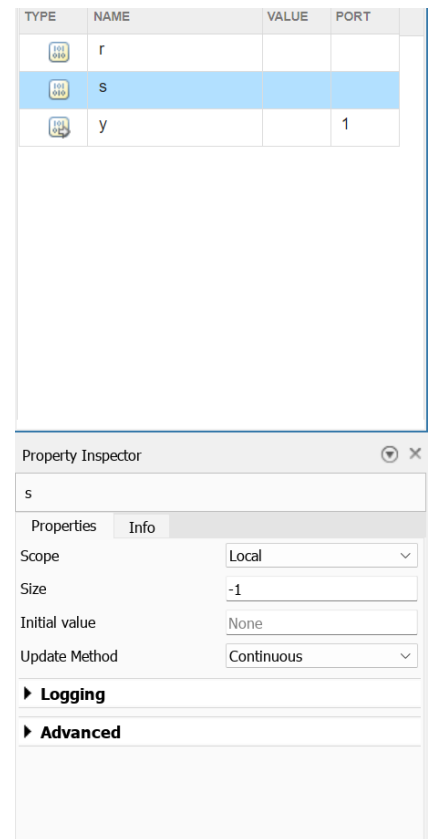
Stateflow های زیر را با متغیر های r و s و y داریم و s_dot و r_dot نیز مشتق های اول ما هستند.

مقادیر را طبق کتاب میدهیم و برای شرط و action های هر دو استیت مقادیر لازم را پیاده سازی میکنیم و شرط آغازین را به استیت one وصل میکنیم.

به یکسری مشکلات برمیخوریم که متغیرها شناسایی نشدند و برای حل این مشکلات باید update method در بخش property inspector را به حالت continuous بگذاریم :



و سپس در بخش symbols مقدار y را output کنیم و r و s را continuous local data کنیم تا Simulink مقادیر s_dot و r_dot را بشناسد :



در ادامه در قسمت زیرین Simulink کلیک کرده و auto را fixed step auto میکنیم :

FixedStepAuto

سپس سایز را 1 گذاشته و زمان را 0 تا 10 میگذاریم :

Simulation time

Start time: 0.0 Stop time: 10.0

Solver selection

Type: Fixed-step Solver: auto (Automatic solver selection)

▼ Solver details

Fixed-step size (fundamental sample time): 1

Zero-crossing options

☐ Enable zero-crossing detection for fixed-step simulation

حال مشکل حل شد و خروجی داریم.

حال میاییم و خروجی stateflow را محاسبه میکنیم و میبینیم که خروجی بر اساس زمان به صورت زیر است :

$Y(t)$:

$\{ t \text{ if } t=3k \text{ for } k \text{ in } N, t \text{ if } t=3k+1 \text{ for } k \text{ in } N, \text{ otherwise absent } \}$

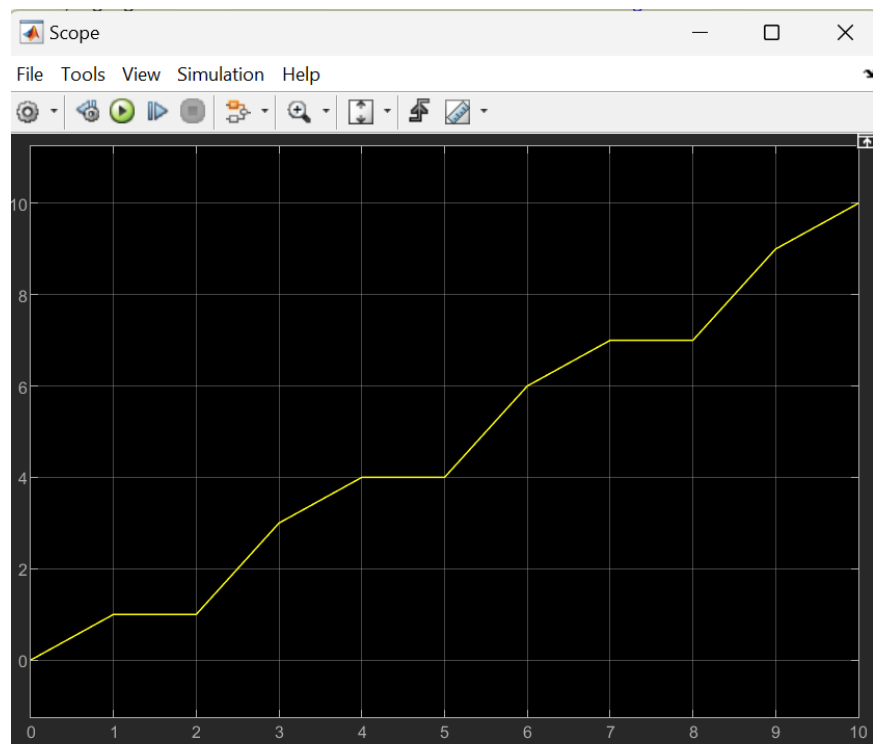
پس در زمان های زیر خروجی های زیر را داریم :

$(1,3,4,6,7,9,10,\dots)$

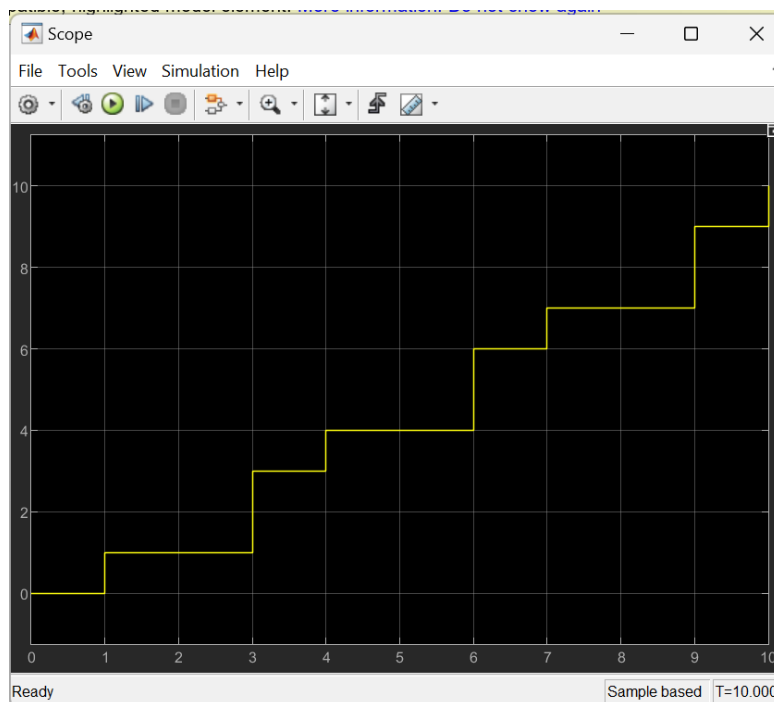
at times

$1,3,4,6,7,9,10,\dots$.

حال chart خود را به یک scope وصل میکنیم و خروجی را مشاهده میکنیم :



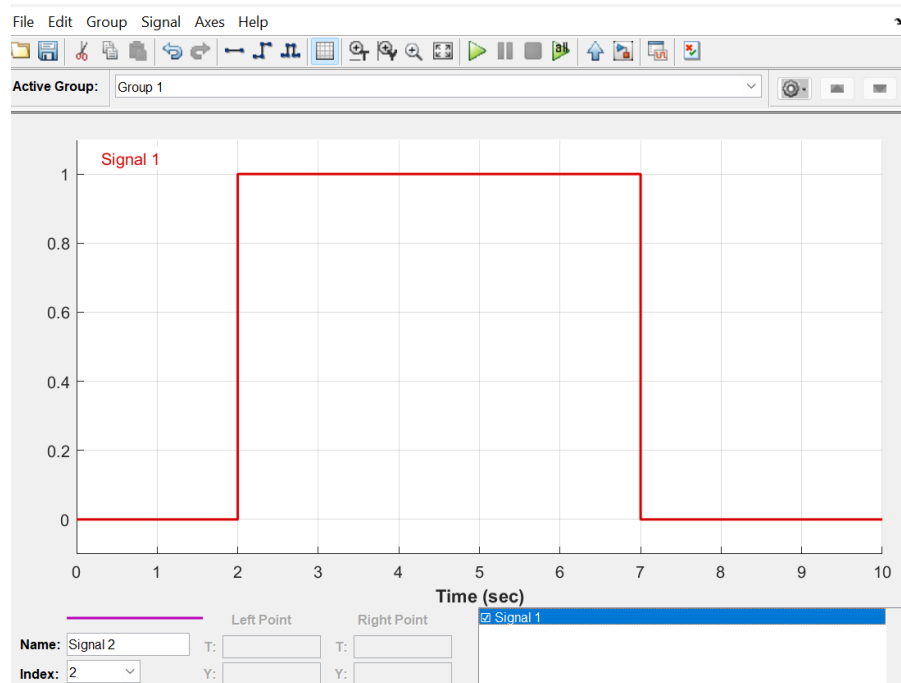
میبینیم که مقدار به صورت پله ای نیست. برای اینکه خروجی پله ای شود باید از quantizer block استفاده میکنیم که تابع را پله ای میکند :



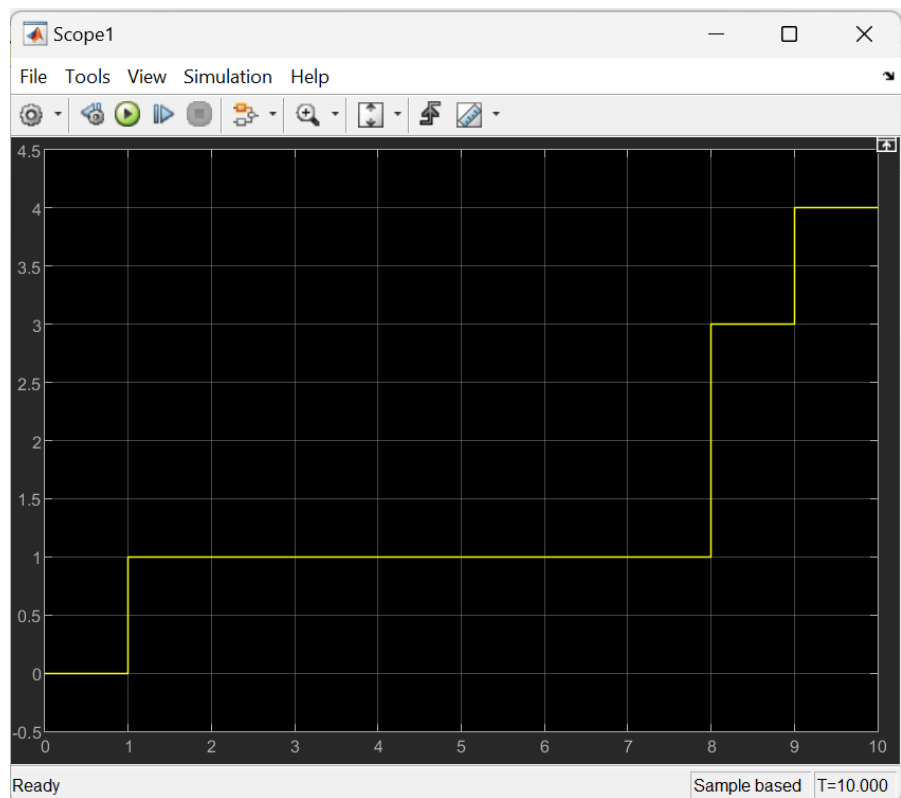
میبینیم که مقادیر با زمان متناسب است پس خروجی درست است.

برای بخش b همین مراحل را طی میکنیم و شرط reset را نیز اضافه میکنیم و 0 و 1 را برای false و true بودن آن میگذاریم.

خروجی نیز همان y است اما در اینجا یک متغیر ورودی reset داریم که باید مقادیر 0 و 1 را در زمان 0 تا 10 ثانیه بگیرد. ورودی باید به صورت پله ای باشد و در لحظه تغییر مقدار بدهد تا کارایی تابع خراب نشود. برای این کار از signal generator استفاده میکنیم که میگوید هر لحظه مقدار چه چیزی باشد. مقدار را به صورت زیر میدهیم :



در ثانیه های 2 تا 7 خروجی 1 است که true است. حال خروجی signal را مشاهده میکنیم :



میبینیم در زمان هایی که 1 دادیم سیگنال reset شده و دوباره سپس شروع به افزایش کرده.