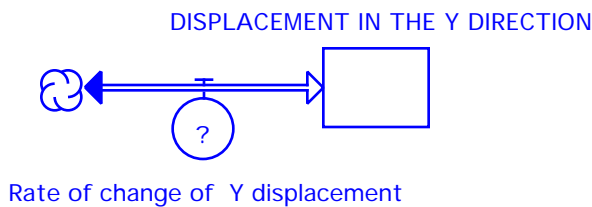
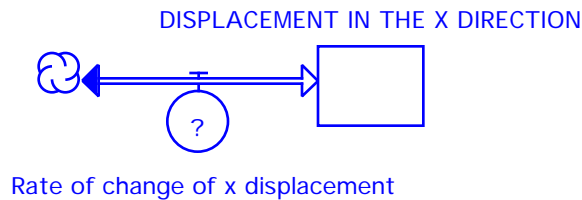


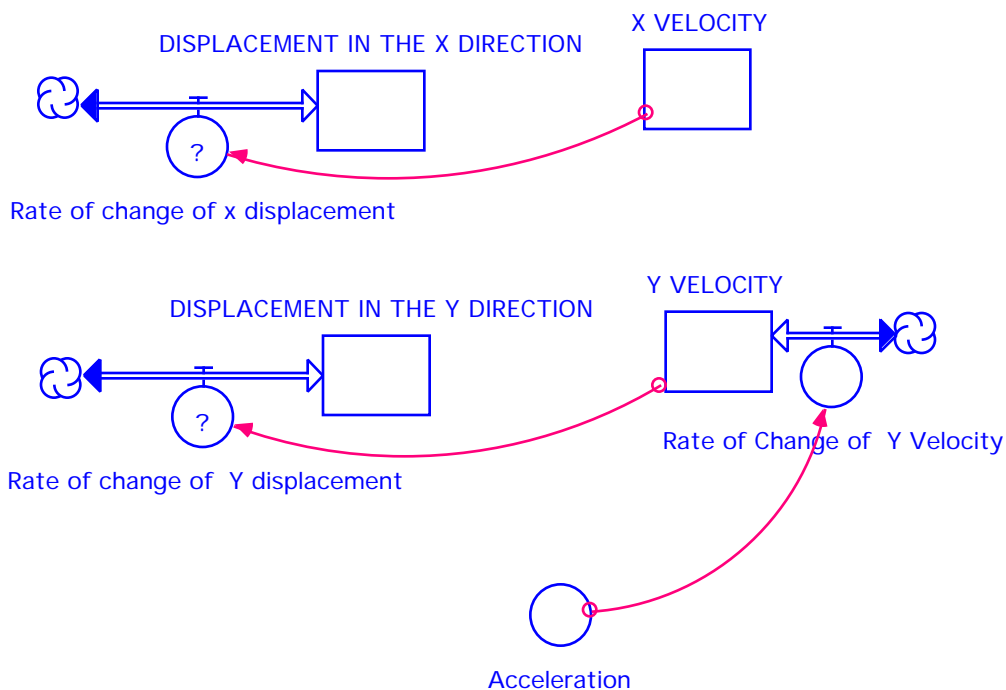
## PROJECTILE MOTION STUDENT GUIDE

Open up Stella and let's start with a new window. Be sure to click on the world and change it to  $X^2$ . Now build two simple models as you did in the kinematics model that look like the following. Notice that the flows are bi-flows.



You have now created a very simple kinematics model for changes in displacement in the x and y directions. Double click on each stock and set their initial values to zero.

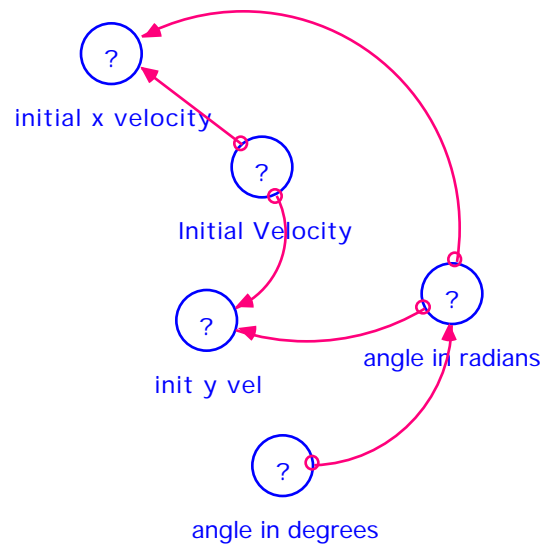
We now have to define the rate of change of each variable. Create two new stocks for the velocity in the x direction and the velocity in the y direction. We know that the velocity in the x direction will be a constant, however, the velocity in the y direction will vary over time due to the acceleration due to gravity. Build your model so that it looks like the following:



We need to input some values. In the acceleration converter, double click and place in the value  $-9.8 \text{ {m/s/s}}$ . This is the acceleration due to gravity and, of course, it is directed downward.

Double click on the flow titled “Rate of Change of Y Velocity”. Click on acceleration in the Required Inputs box so that it is entered as the value for that flow. Do the same for each of the other flows that have question marks.

The problem is, the first time we tried to help the second string quarterback, he threw the ball with an initial velocity of  $100 \text{ m/s}$  at an angle of  $30^\circ$ . This means that the velocity has two parts. One part is upward, in the y direction, and the other part is in the x direction. We need to set up some converters that will break the velocity vector into its components. On the right hand side of your screen, please build the following:



Please enter the following values:

- Initial velocity =  $100 \text{ {m/s}}$
- angle in degrees =  $30$
- angle in radians =  $\text{angle in degrees} * (\pi/180)$
- initial y velocity =  $\text{Initial velocity} * \sin(\text{angle in radians})$
- initial x velocity =  $\text{Initial velocity} * \cos(\text{angle in radians})$

Once this has been completed, all of the question marks in the converters should have disappeared.

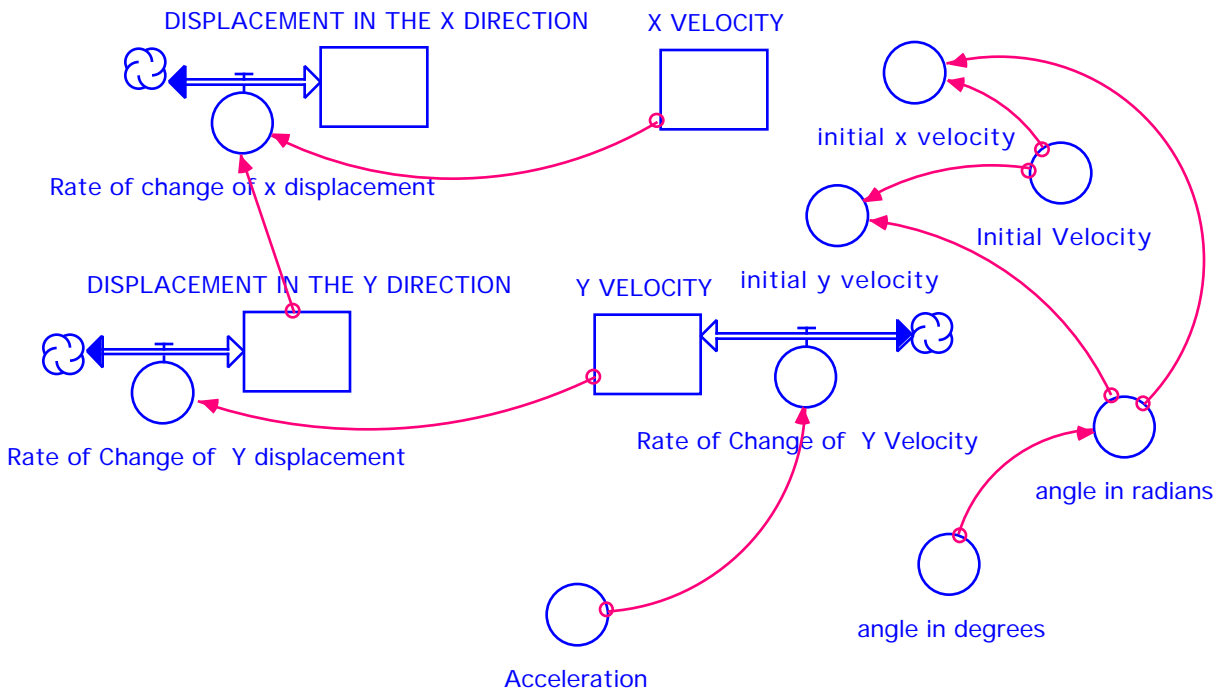
Now, double click on the stock for the x velocity and click on `initial_x_velocity` in the Allowable Inputs box for the initial value. Do similarly for the y velocity stock, but also click on the Non-negative box so that the checkmark disappears. Since acceleration is directed downward, y velocity will eventually become negative.

Although this is a good model at this point, we need to do some fine tuning. Connect the Displacement in the Y direction stock to the rate of change of x displacement flow. A question mark will appear in the flow. Double click on the flow and enter the following statement:

if (DISPLACEMENT\_IN\_THE\_Y\_DIRECTION >=0.2 {m}) then (X\_VELOCITY) else 0

The reason for this is that we want to know how far the quarterback can throw the ball in the air. Stella will keep on calculating the displacement even after the ball has started rolling on the ground. So, by including this statement, you've told Stella to stop calculating once the ball hits the ground.

Hopefully, this is what your final model looks like:



Before you run the model, create a graph located directly below the model. Don't forget to pin it down. For your first graph, choose the "Scatter" option, click in the Comparative box, and choose the displacement in the x direction and the displacement in the y direction as the X and Y variables, respectively.

Create a second graph and pin it down to the right of the model. Choose Scatter and Comparative, but choose angle in degrees and displacement in the x direction as the X and Y variables, respectively.

Finally, try graphs 3, 4, etc. and different combinations of other variables and see what the shapes of the graphs tell you. Your goal is to answer the questions listed below for an angle of 30 degrees. In the menu bar at the top, click and hold on Run, then drag down to Run to execute the model.

**Q1.** How far did he throw the ball?

**Q2.** How long was it in the air?

**Q3.** How high did it go?

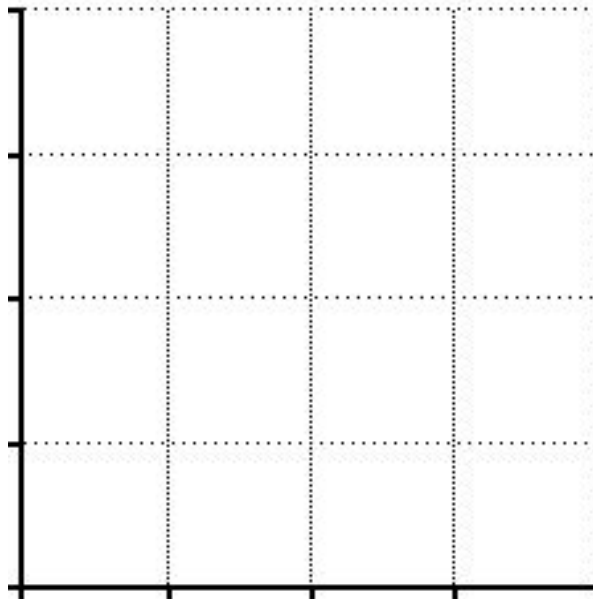
**Q4.** Is this as far as he can throw the ball?

We are now ready to answer our original question. What angle will give our struggling young quarter back the greatest range for his arm?

In the menu bar at the top, click and hold on Run. Scroll down to Sensi Specs. Type 7 as your number of runs. Double-click on angle\_in\_degrees in the Allowable box to move it to the Selected box. Click once on angle\_in\_degrees in the Allowable box. In the box labeled Start, type 15 and in the box labeled End, type 75. Click on Set and you will see seven different values for your angles appear in the small window to the right. Click OK.

Click and hold on Model in the top menu bar and scroll down to Restore ... Graphs & Tables. From the Run menu, select S-Run and the model will execute seven times.

**Q5.** On the axes provided, sketch your graph of y-displacement vs. x-displacement.



**Q6.** Which angle gave the greatest range?

**Q7.** What happened to each of the angles above and below the angle with maximum range? What are these angles called?

**Q8.** Looking at the graph for x-displacement vs. angle in degrees, what patterns do you notice?

**Q9.** Suppose you were to coach baseball. If the outfielder wants to throw the ball home to nail a runner, how would you advise them to throw the ball? Why?

**Q10.** If our struggling young quarterback wants to throw the bomb, how should he throw the ball? Why? Support your answers from the results and your graphs.