



“Analog Clock”



High School Math Alignment

Domain: Geometry

Cluster: Apply geometric concepts in modelling situations

Standard: CCSS.MATH.CONTENT.HSG.MG.A.1 Use geometric shapes, their measures, and their properties to describe objects

Standard: CCSS.MATH.CONTENT.HSG.MG.A.3 Apply geometric methods to solve design problems

Level 2 CSTA Alignment

Concept: Control

Standard: 2-AP-12: Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals

Description

Students will build an analog clock from scratch that orients its hands based on input minute and hour **variables**. Students will have to use a series of **translations**, **rotations**, and **loops** to orient the clock's numbers around the circle. They will have to write **functions** using **math blocks** to rotate the hands the appropriate number of degrees depending on the time.

Learning Objectives

- Students will think creatively about how to apply what they know about translations and rotations to create a familiar arrangement of numbers
- Students will use loops to iterate commands
- Students will label and organize their code into component parts
- Students will write functions involving fractional degree measure in a circle

Videos

- Rotation Away from the Origin:
<https://www.youtube.com/watch?v=ywinUeZeG5Y>
- Loops: <https://www.youtube.com/watch?v=OQeyRI3j3FU>
- Using Variables: <https://www.youtube.com/watch?v=l3OeE52zlns&t=5s>

Review/Prior Knowledge

Students will need to be comfortable with basic BlocksCAD building and transformations - especially rotations. If you have never used BlocksCAD before with your students, you can take some time to introduce them to the interface and to basic building practices. They should also have some experience using variables, functions, and loops either in BlocksCAD or in other coding programs. Recommended BlocksCAD explorations to develop these skills include Pizza Printer and Ice Cream Machine.

Resources

- Student Worksheet: Analog Clock → Student guide for activity
- Analog Clock Solution.xml → Example solution (walkthrough below)

Teacher Notes

Introduction: Look at an analog clock in your classroom, or online, and ask students to think about how the two hands move. In particular, how does the hour hand know exactly where to be between one hour and the next? Tell students that they are going to use BlocksCAD to model an analog clock. They should be prepared to work independently or in small groups, following their intuition and trying out ideas before asking for help.

Exploration: The Student Worksheet provides a basic outline for students to follow and some cues to keep them on the right track. The facilitator, however, should be prepared to help students implement their ideas and brainstorm new ones. Make sure that students are using union blocks to bring together the blocks they make for each section. They should label and collapse these blocks each time they finish a section. The walkthrough attached below provides very detailed explanations of a

way to complete each section if you are newer to BlocksCAD. You can use the Table of Contents to navigate to certain sections. The available example solution BlocksCAD file also contains the blocks that students will need to build to get their clock working.

Reflection Questions and Activities

1. What were the biggest design challenges, and what ideas and blocks helped you overcome them?
2. How do you think the actual mechanisms in a physical analog clock help the hands move in the same way that your programmed clock worked?
3. What other objects with movie pieces could be modelled using BlocksCAD by entering variables that affect the design?

Teacher resources: *The complete build of the clock*

Clock Face and Boundary	4
Center Posts	6
Numbers	9
Orienting the Numbers Correctly	13
Hour and Minute Hands	14
Animating the Minute Hand	19
Animating the Hour Hand	21

Clock Face and Boundary

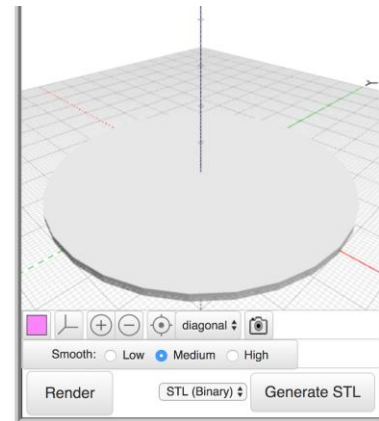
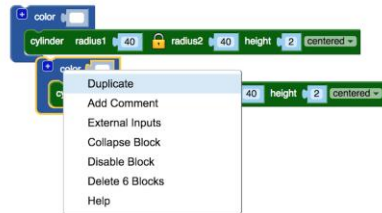
We begin with the main body of the clock. Drag and drop a cylinder from the 3D Shapes folder. Make the cylinders radius 40 and the height 2. Center it.



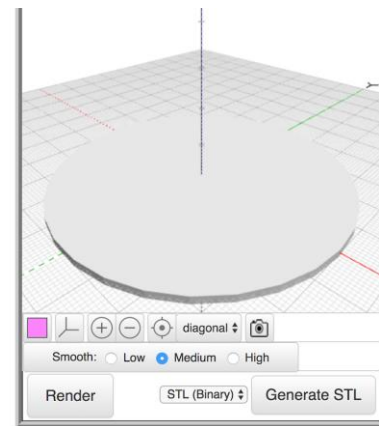
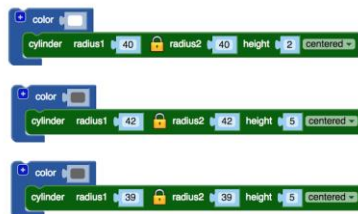
Drag and drop a white color block from the Transforms folder. Place it on the cylinder.



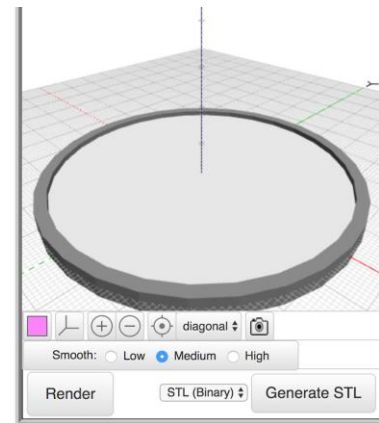
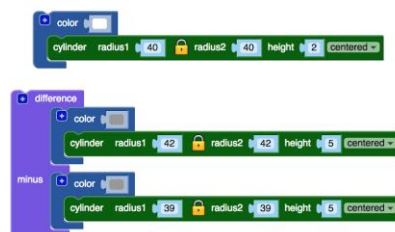
Duplicate this block two times and pick a different color from the original.



Make one of the cylinders radius 42 and height 5. Make the other cylinders radius 39 and height 5.



Drag and drop a difference block from the Set Ops folder. Subtract the smallest cylinder from the largest cylinder. Press render.



Drag and drop a union block from the Set Ops folder. Collapse the blocks and place them in the union block.



Add a comment to this union block and call it “Clock body”.

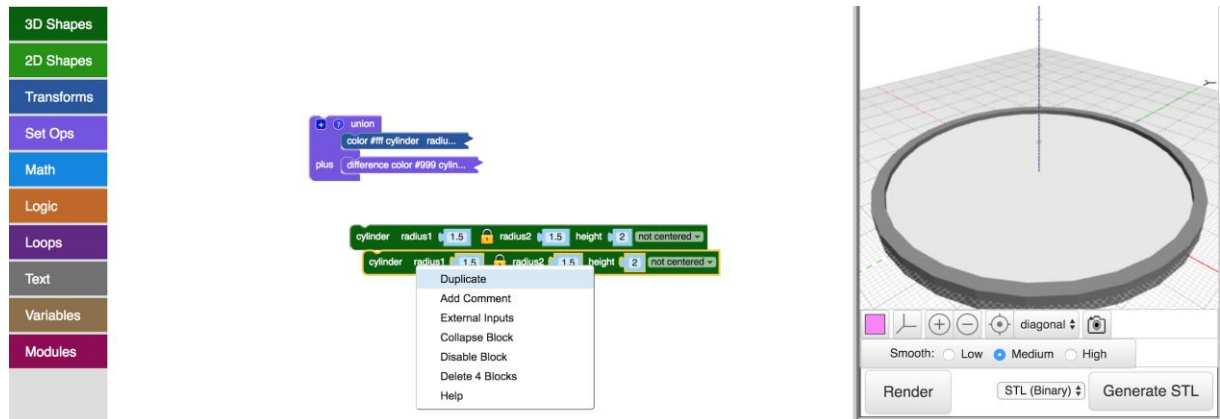


Center Posts

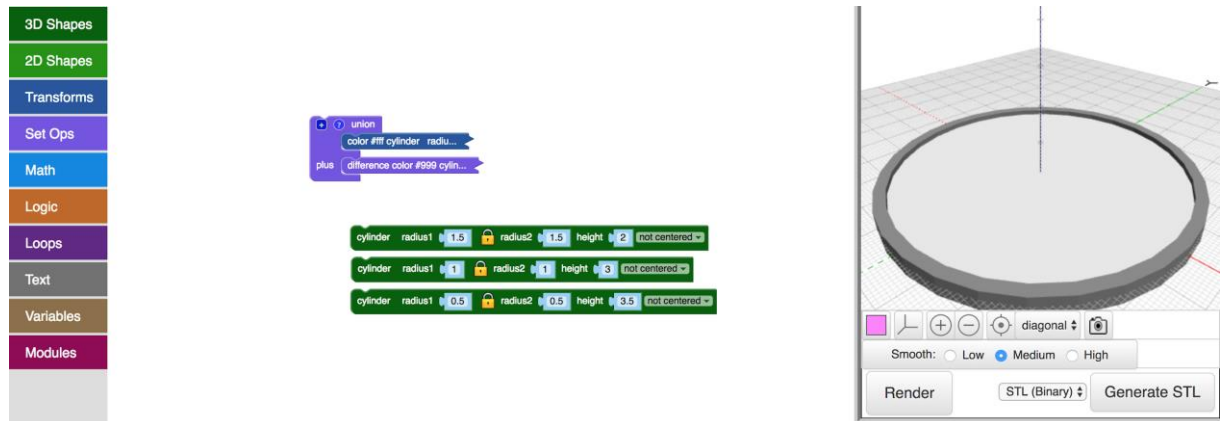
Drag and drop a cylinder from the 3D Shapes folder. Makes this new cylinder's radius 1.5 and height 2. Make it not centered.



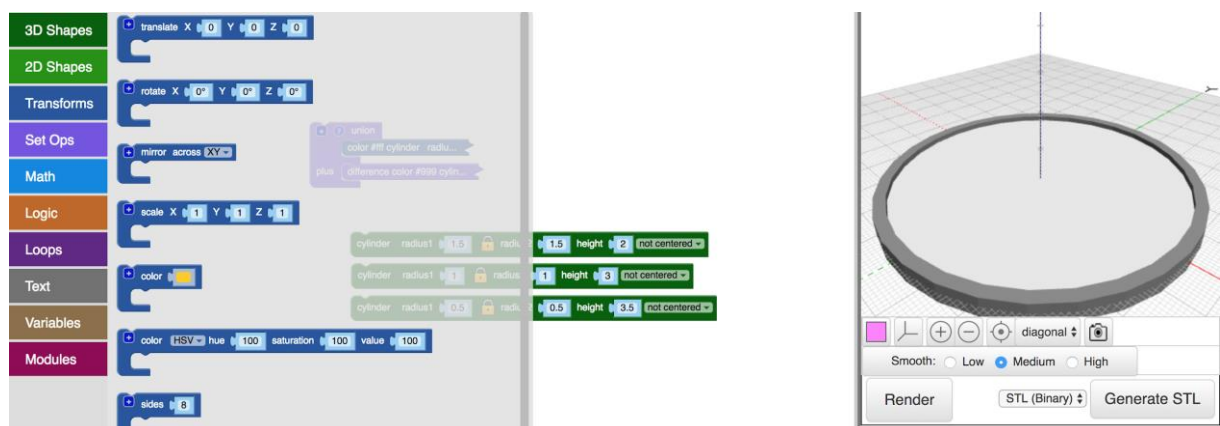
Duplicate this new cylinder two times.



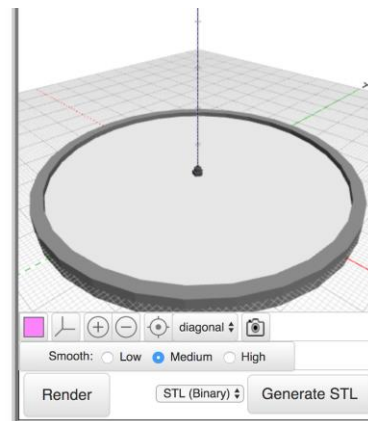
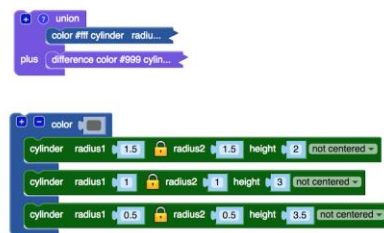
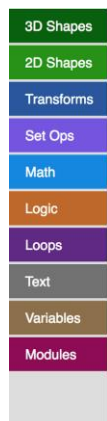
Make the radius of the second cylinder 1 and the height 3. Make the radius of the third cylinder 0.5 and the height 3.5. Make them not centred.



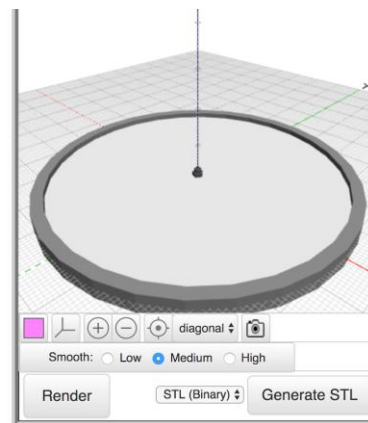
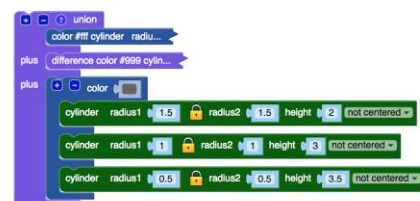
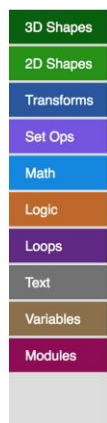
Drag and drop a color block from the Transforms folder.



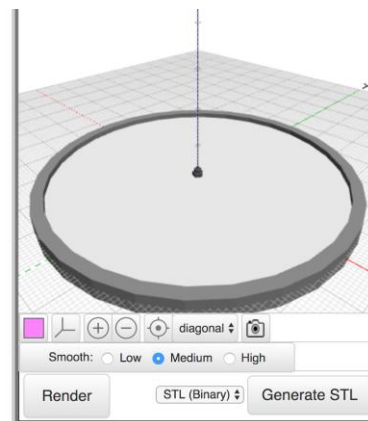
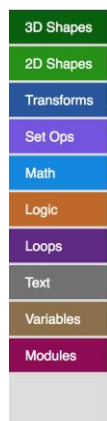
Choose a color and add all three small cylinders in it. Press render.



Add this new block to the “Clock body” union block.



Collapse this union block and put it to the side.

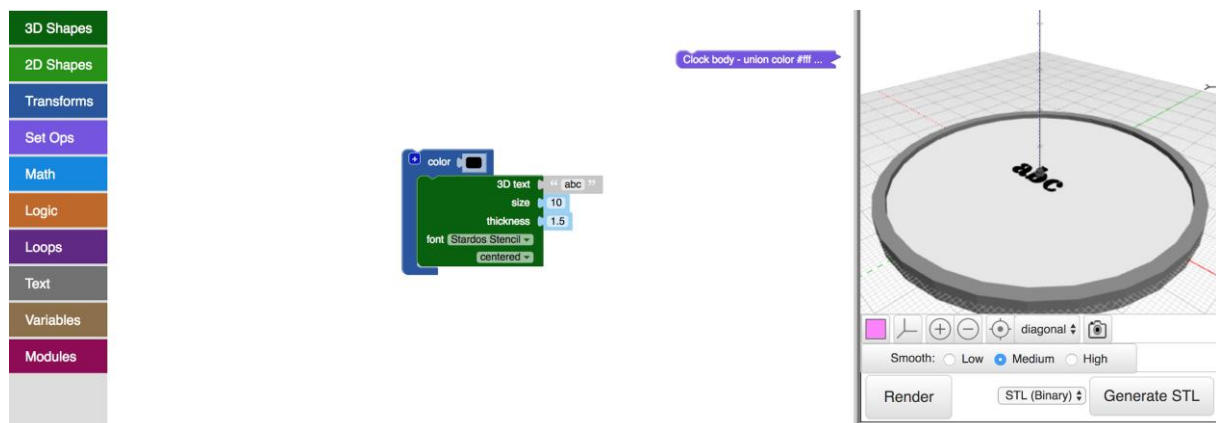


Numbers

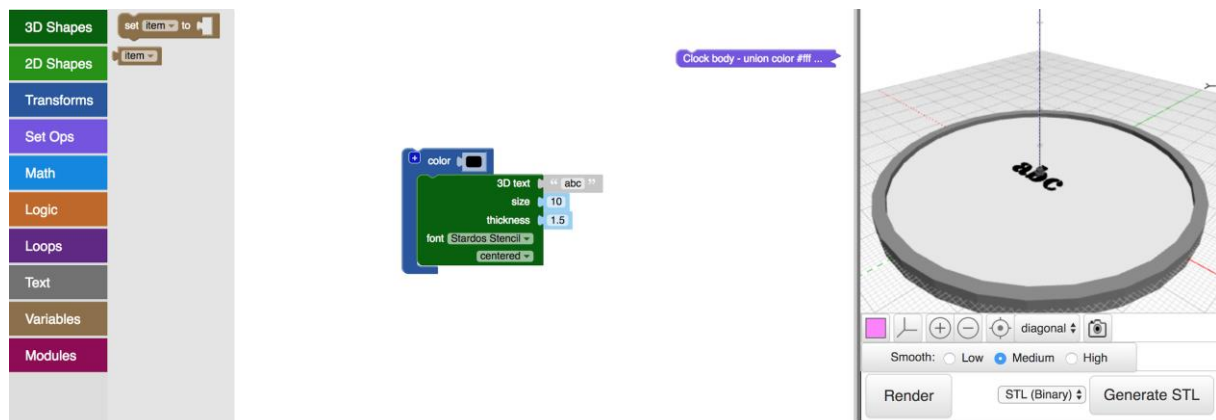
Drag and drop a 3D text block from the Text folder. Make the size 10 and the thickness 1.5. Center it. Pick a font you like.



Add some color to the text. Drag and drop a color block from the Transforms folder. Pick a color and press render.



Now we will place the numbers out and orient them properly. We will use the loop block to do this. We will count from 1 to 12 because that is the amount of numbers we need on the face of the clock. We will rotate the 12 numbers, then translate them out, and lastly rotate them to orient them properly. Drag and drop a variable from the Variables folder.



Place the variable in “abc” spot on the 3D text and make a new variable “i”.



Drag and drop a rotate block from the Transforms folder.

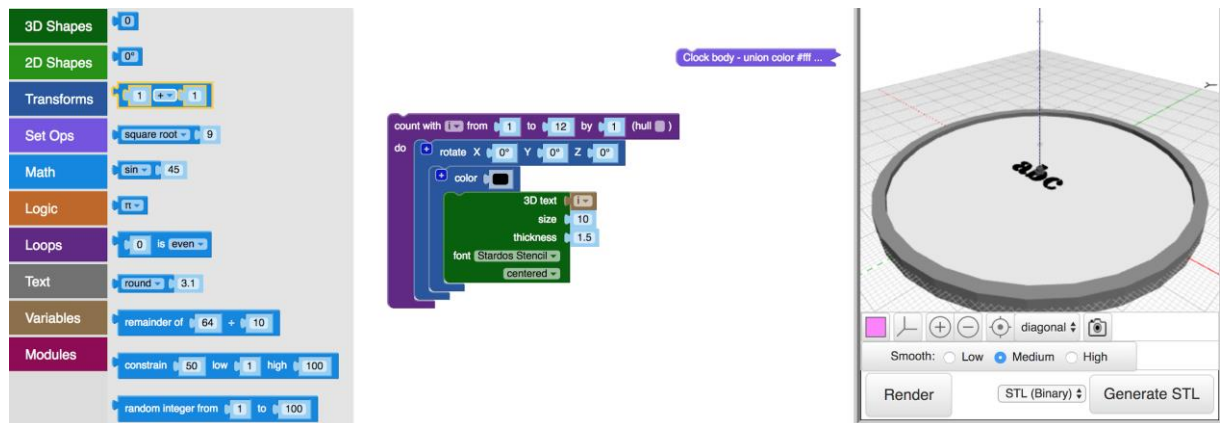


Drag and drop a loop block from the Loops folder.

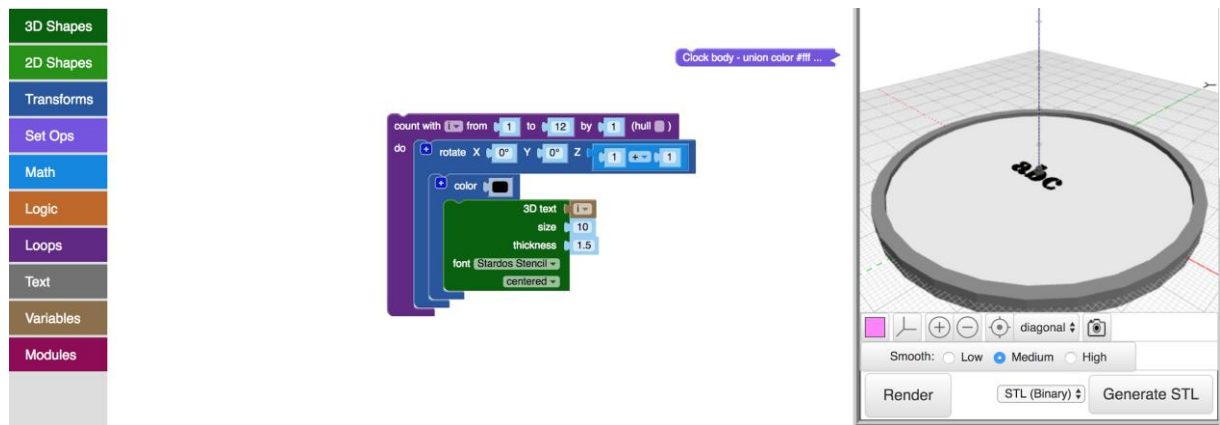


Loop from 1 to 12 by 1.

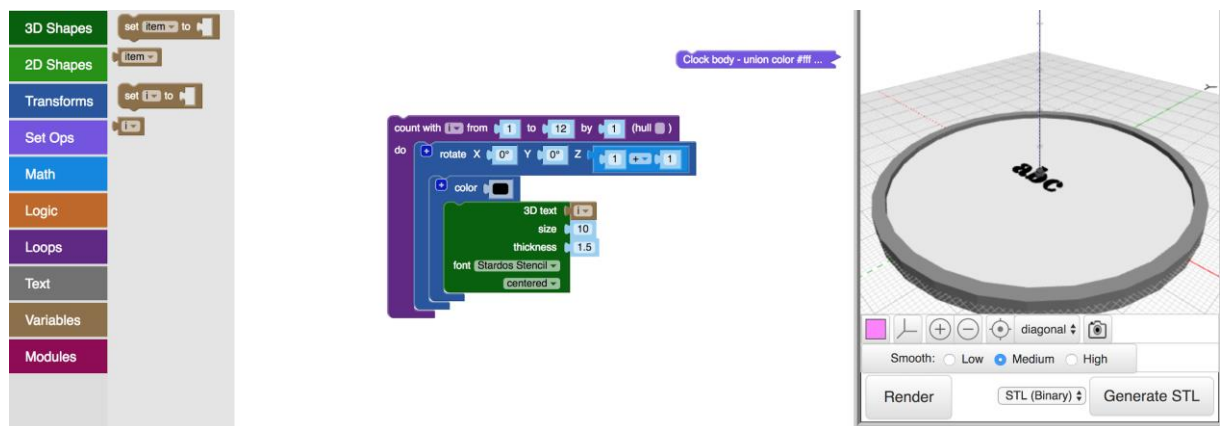
Drag and drop a summation block from the Math folder.



Place it in the Z-value of the rotate block.



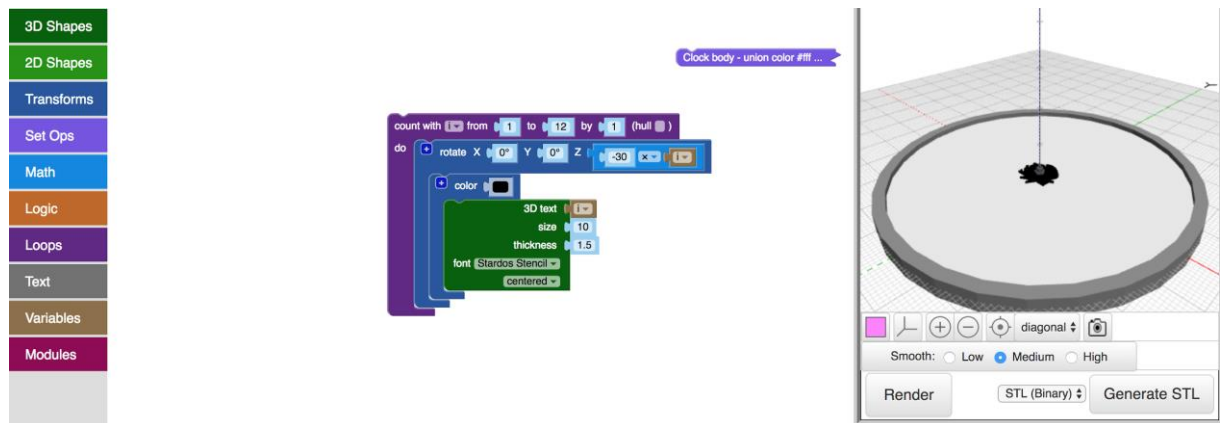
Drag and drop a variable "i" from the Variables folder.



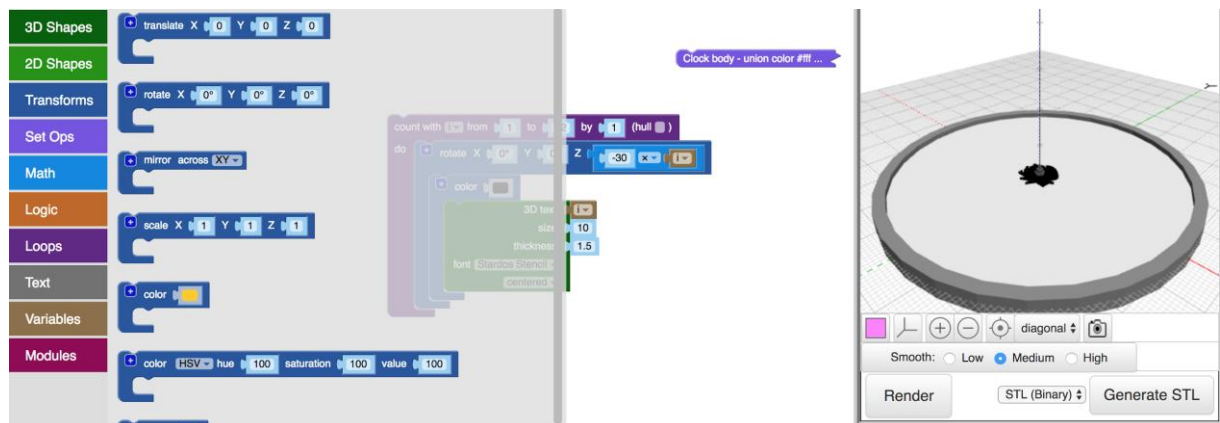
Place it in the summation block and make the block multiply. Multiply the i with -30.

We do this because $30 * 12 = 360$, and this will rotate the numbers around the face of the clock as the numbers are counted. The negative sign makes the loop rotate the numbers the correct way.

Press render.

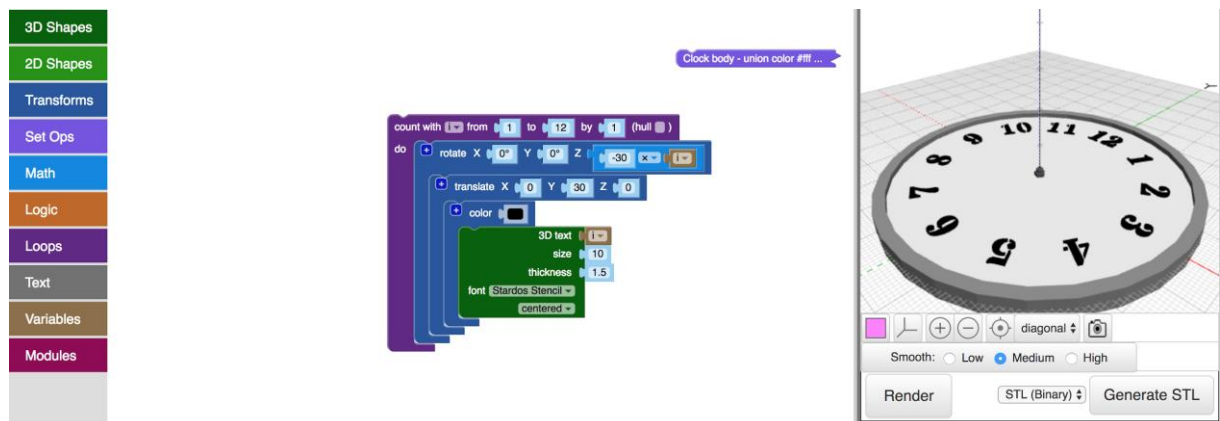


We need to translate the numbers out. Drag and drop a translate block from the Transforms folder.



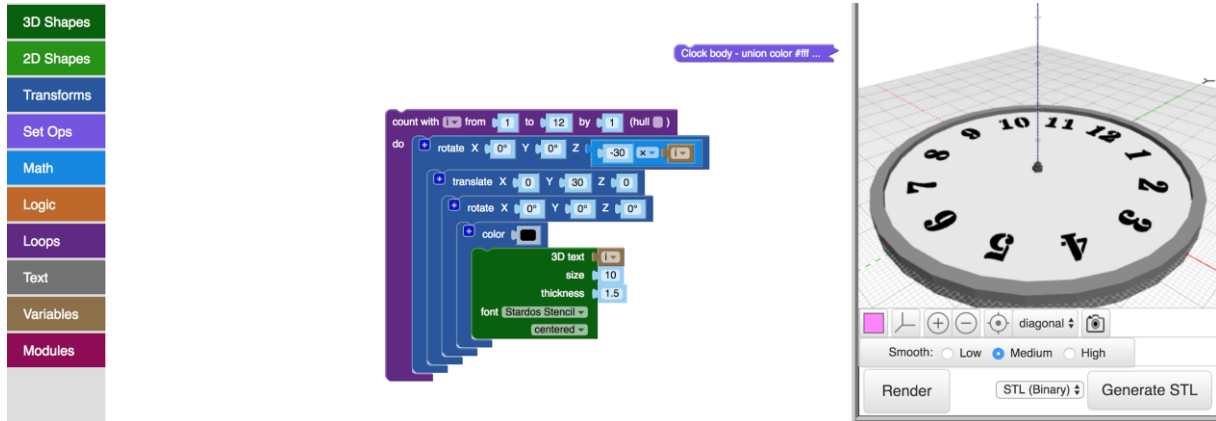
Place it between the rotate block and the color block. Translate 30 in the Y-direction.

Press render.

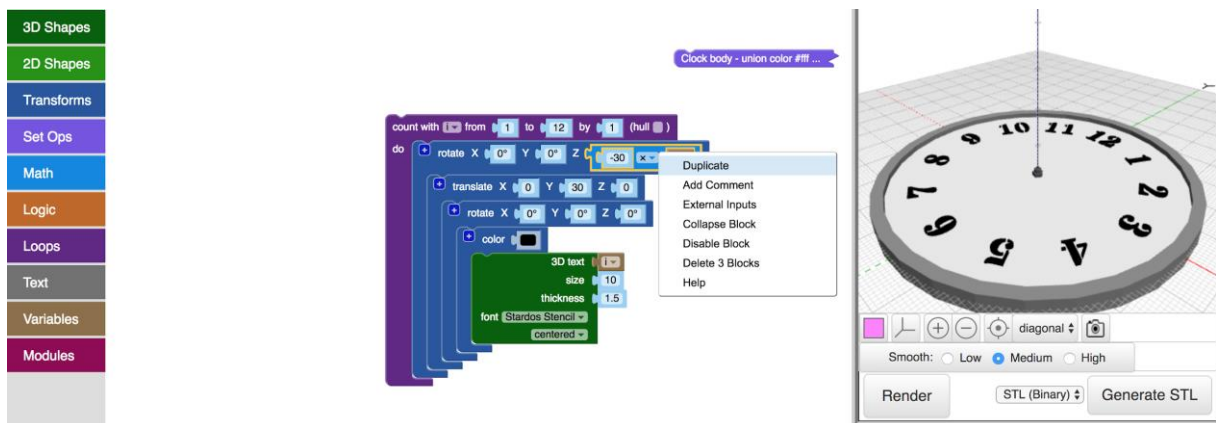


Orienting the Numbers Correctly

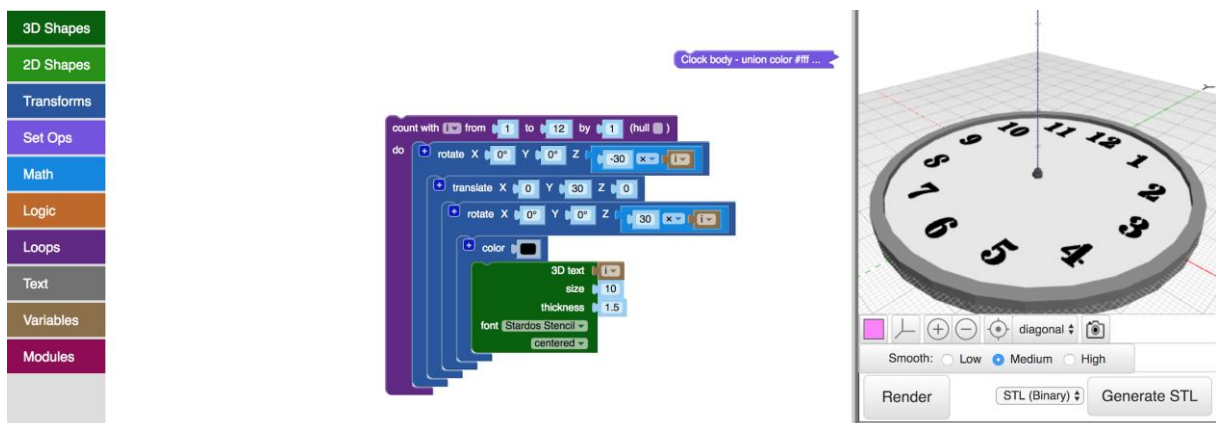
Now we will orient the numbers correctly. Drag and drop a rotate block from the Transforms folder. Place it between the translate block and the color block.



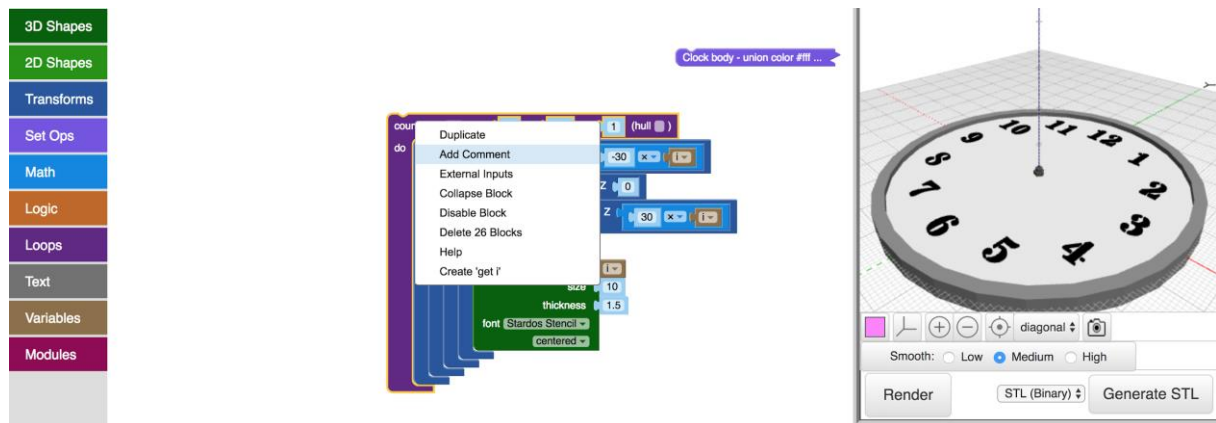
Duplicate the summation block in the larger rotate block.



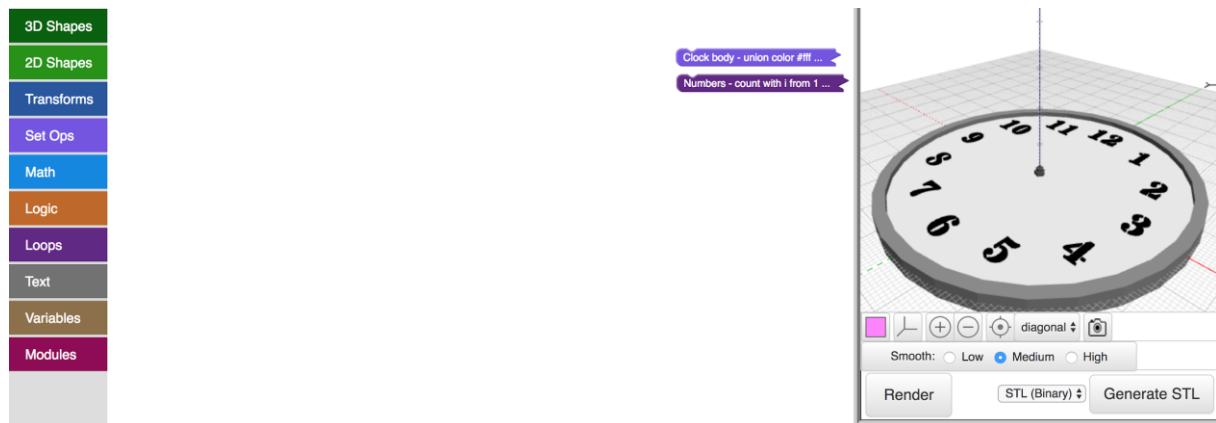
Place this new summation block in the Z-value of the smaller rotate block. Remove the minus sign so "i" is multiplied with 30. Press render.



Add a comment to the loop block. Name it “Numbers”.



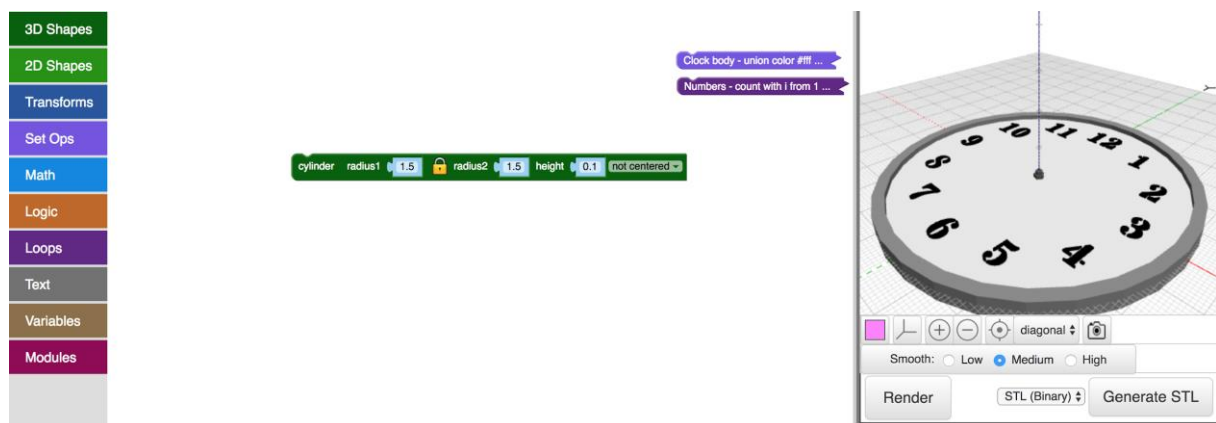
Collapse the loop block and put it to the side.



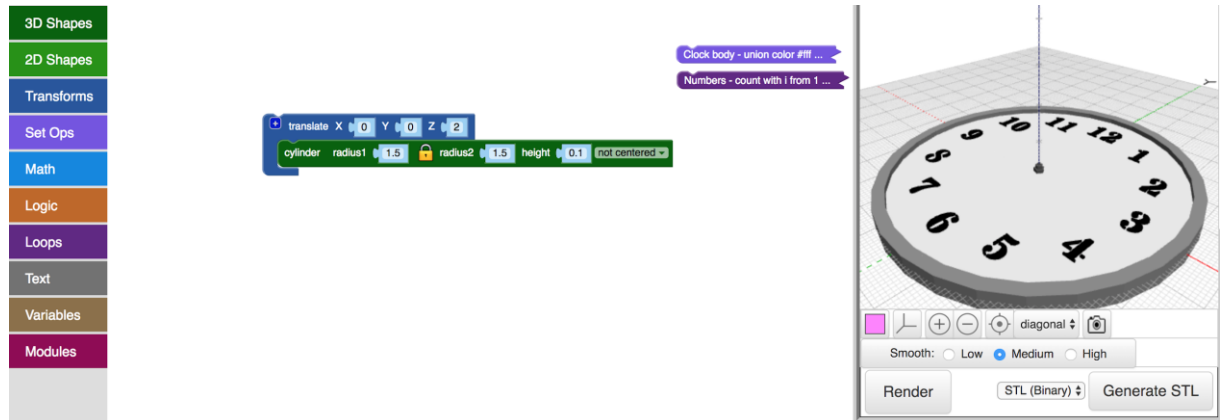
Hour and Minute Hands

Now we will build the hour and minute hand. This walkthrough builds hour and minute hands by hulling cylinders on the ends of the hands to make it more realistic. Students can also just use basic cylinders lying in the XY plane as hands.

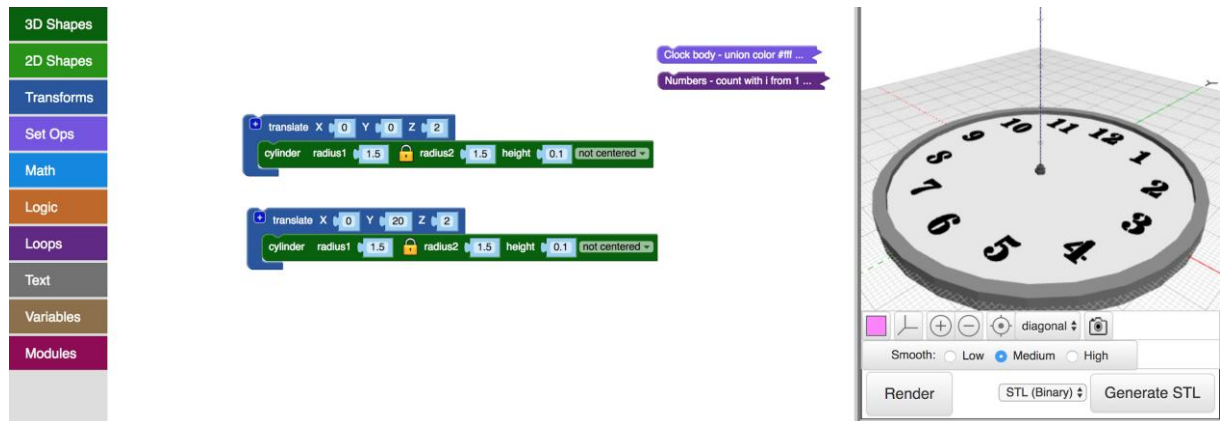
Drag and drop a cylinder from the 3D Shapes folder. Make radius 1.5 and the height 0.1. Make it not centered.



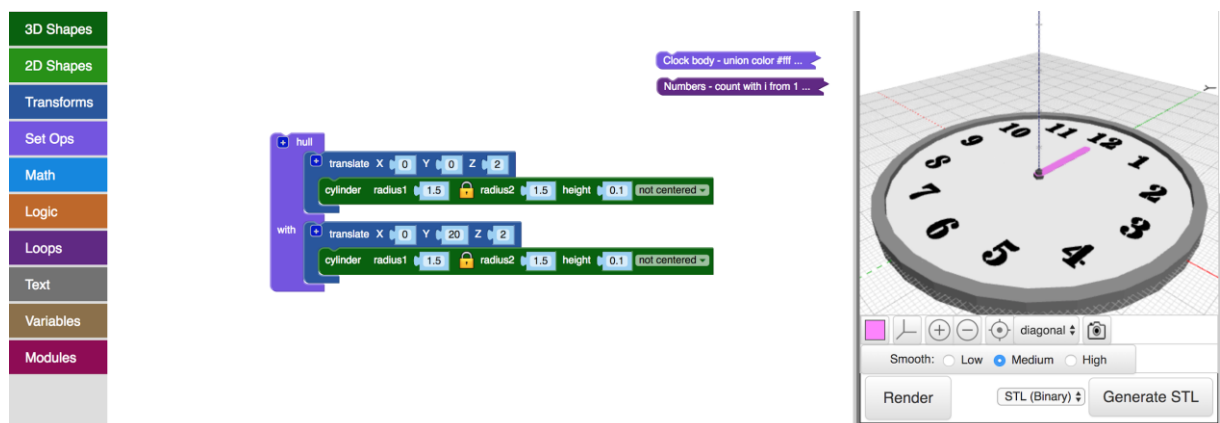
Drag and drop a translate block from the Transforms folder. Translate the cylinder 2 in the Z-direction.



Duplicate this block. Translate the new cylinder 20 in the Y-direction and 2 in the Z-direction.

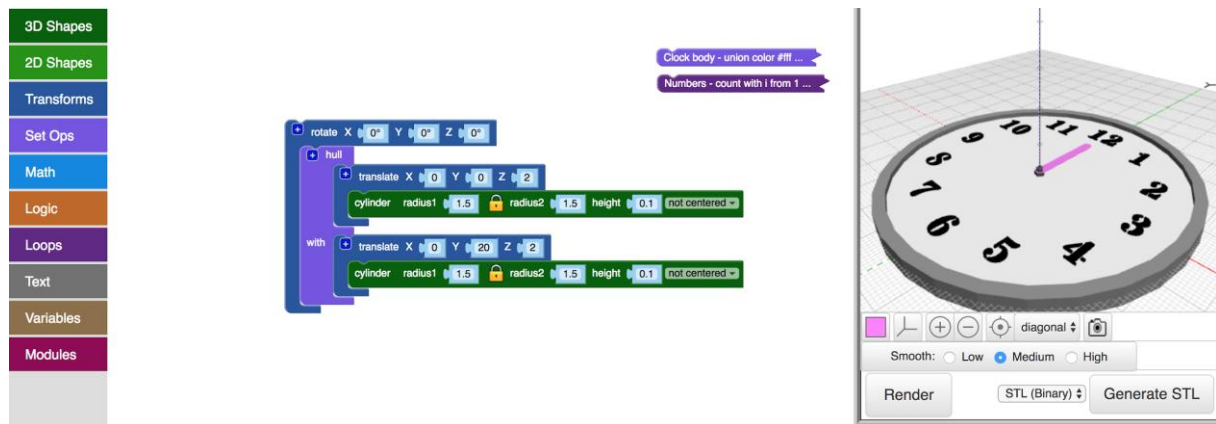


These two cylinders will make the hour hand. Drag and drop a hull block from the Set Ops folder. Hull the two translated cylinders. Press render.



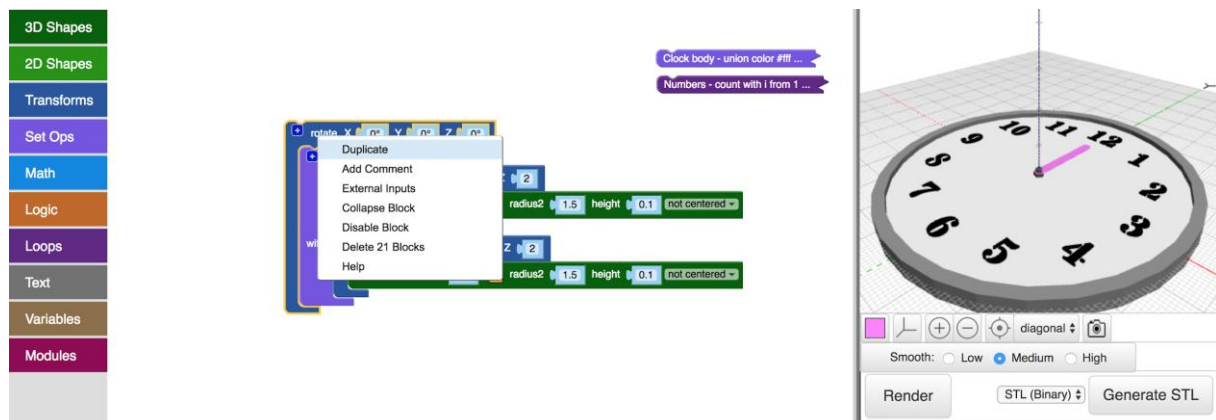
We need to rotate this in order to animate the movement of the clock.

Drag and drop a rotate block from the Transforms folder. We will leave this for later.

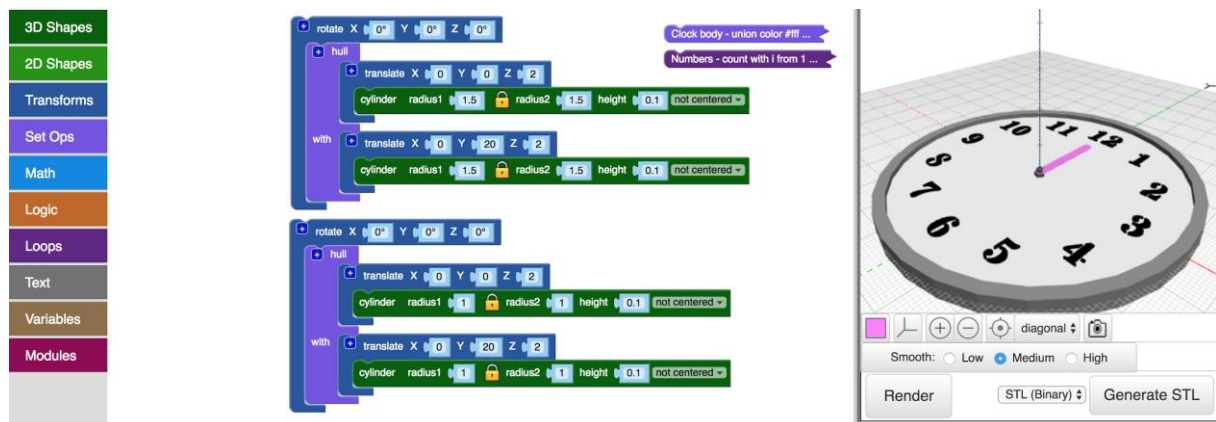


Now we will create the minute hand.

Duplicate the rotated block.

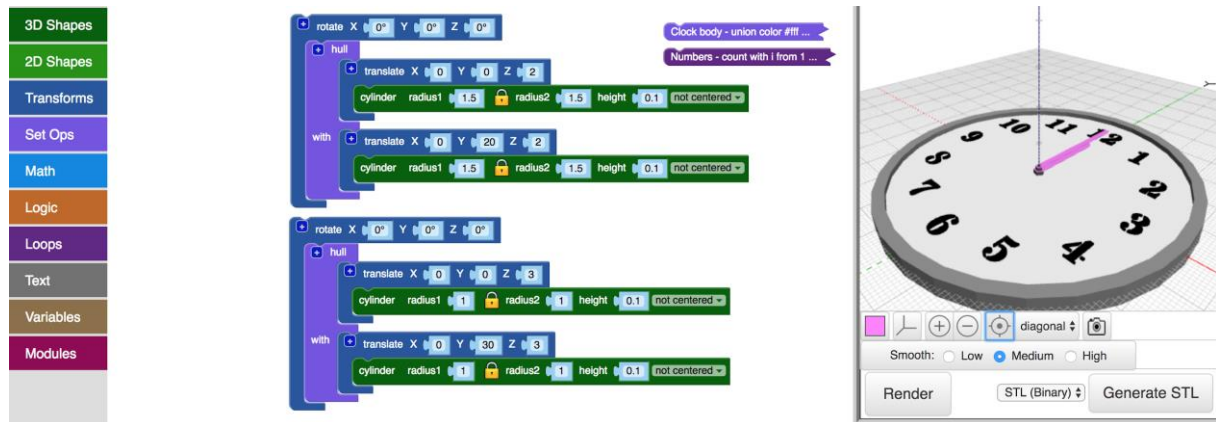


Change the radius of the new cylinders to 1 and leave the height at 0.1.

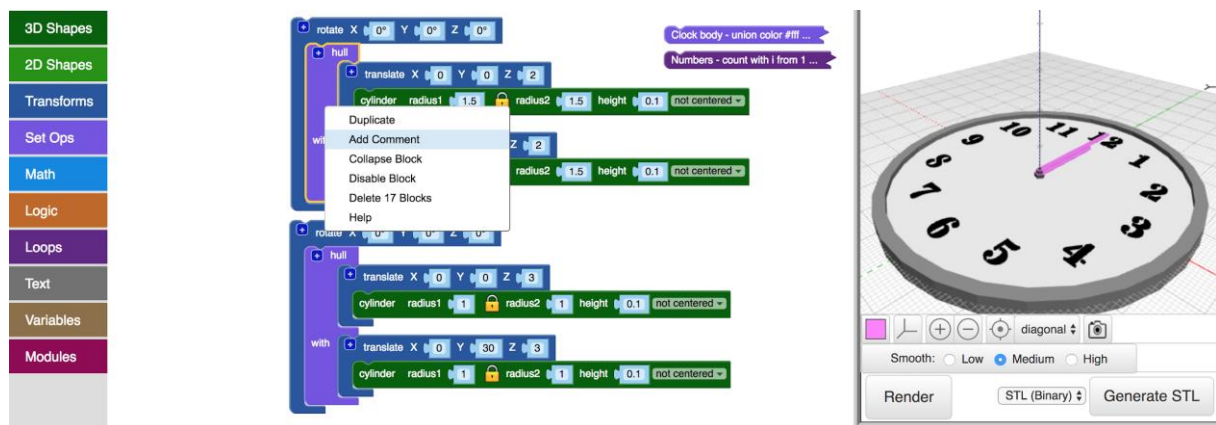


Change the translate blocks in the new rotate block.

The first should translate 3 in the Z-direction. The second should translate 30 in the Y-direction and 3 in the Z-direction. Press render.



Name the first hull block “Hour hand” and the second hull block “Minute hand”.



Collapse them to save space.



Now we will animate the rotation. We will create a spot to input the time and this will in turn decide the degree of rotation of the two hands about the Z-axis. In other words we will have two variables that change the Z-value of the rotate blocks.

Drag and drop two set blocks from the Variables folder. Create two new variables, one in each of the set block.



Name one variable “Hours” and the other “Minutes”.



Drag and drop two number blocks from the Math folder. Place one in each of the set blocks.



Now we will build the input in the Z-value of the rotations.

Animating the Minute Hand

We begin with the minute hand.

We need to be able to translate a number to an angle. We will do this with a fraction. A circle is 360° and there are 60 minutes in an hour. This means that the angle of 1 minute is:

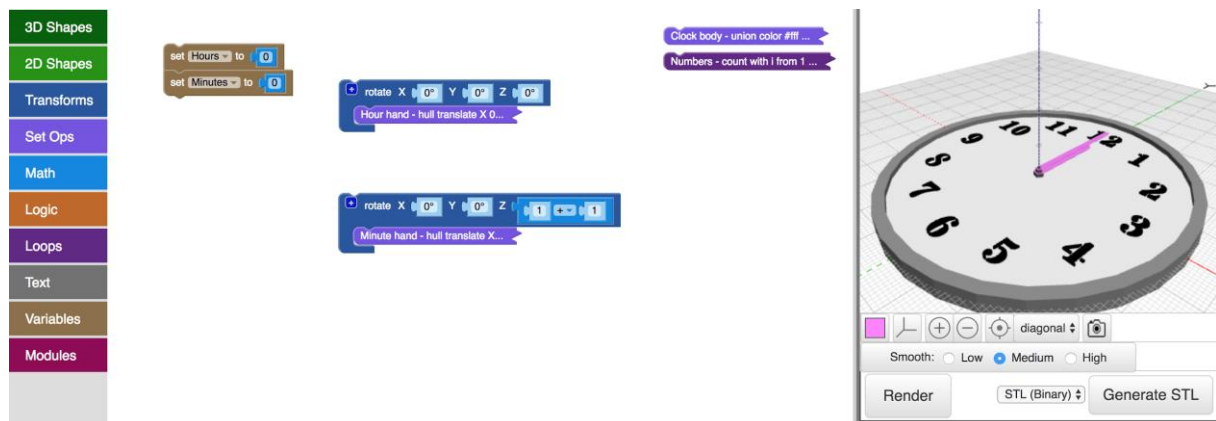
$$\frac{360^\circ}{60}$$

If we input the minute value and multiply it with this fraction we will get the desired angle:

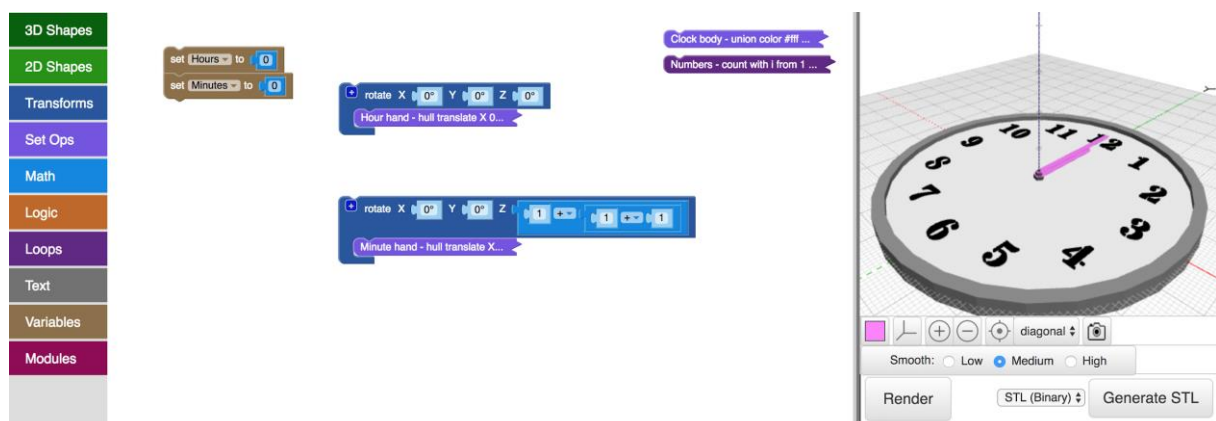
$$minutes \times \frac{360^\circ}{60}$$

We will build this in the Z-value of the rotate block.

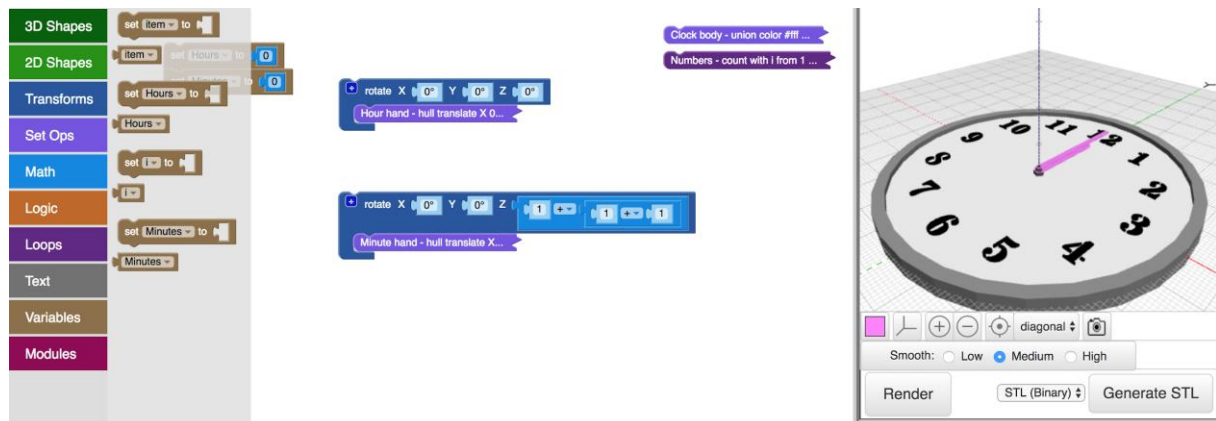
Drag and drop a summation block from the Math folder. Place it in the Z-value of the rotate block that rotates the minute hand.



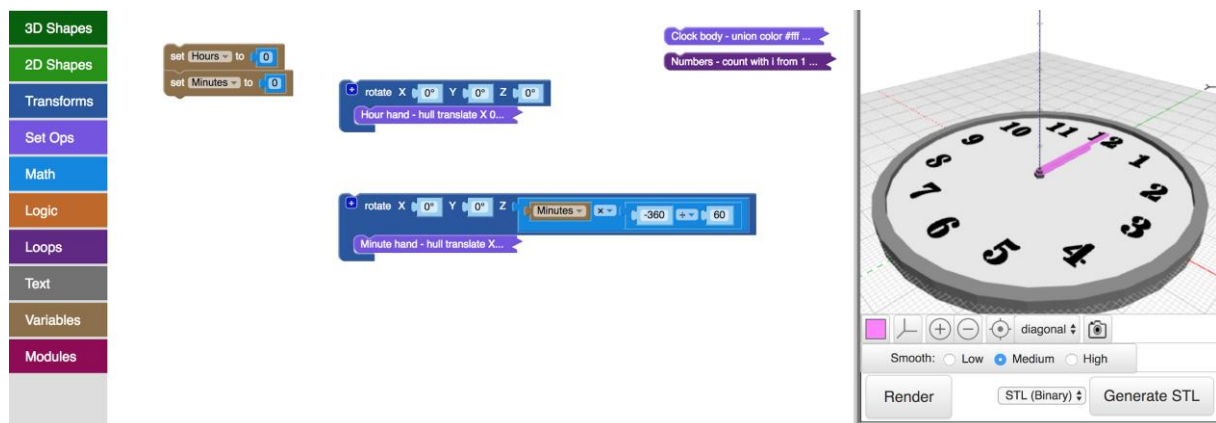
Drag and drop another summation block and place it in the last term of the first summation block.



Drag and drop a “minutes” variable from the Variables folder.

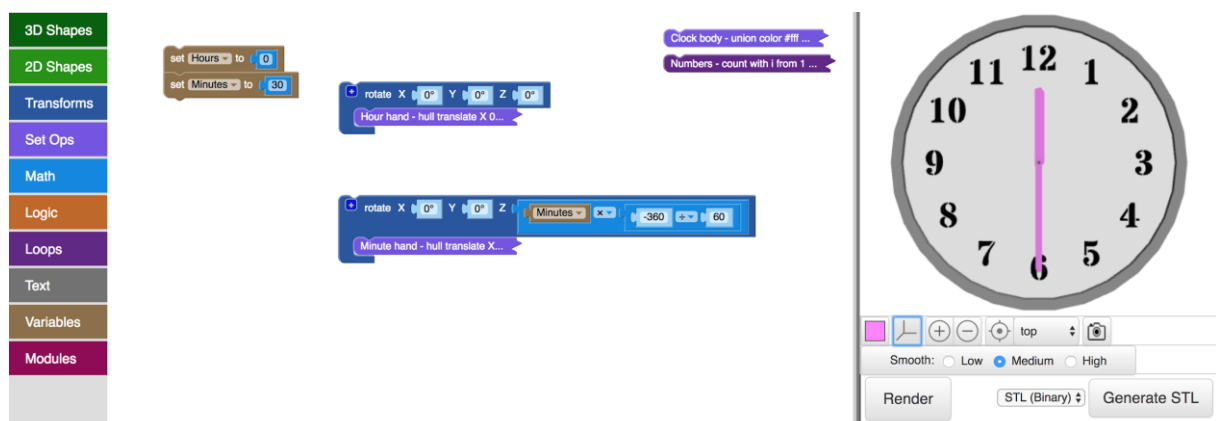


Place it in the first term of the Z-value. Multiply this variable with -360 divided by 60.



We need the minus sign to turn the minute hand the right direction about the Z-axis.

Test the function by typing 30 in the Minutes set block. Press render and check that the minute hand points straight down.



Make the minute value 0, render and set the minute hand block to the side.

Animating the Hour Hand

This one is slightly more complicated, because the movement is decided by both the hour and the minute. We know that a clock at 01:30 does not show the hour hand over 12 or 1, but between 12 and 1. We need to animate this.

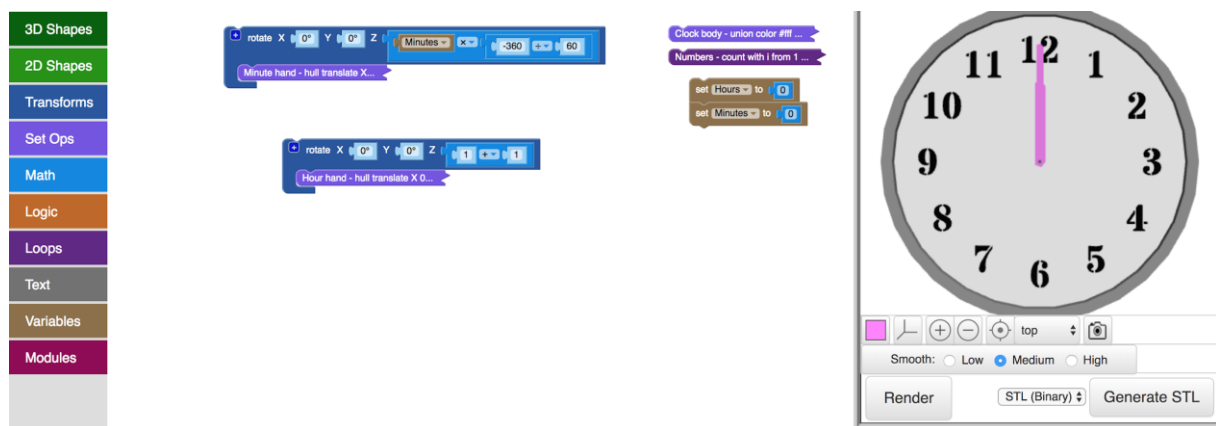
We need to take a number between 0 and 12 and translate it to an angle. Similar to the minute hand we do this with this fraction:

$$\frac{360^\circ}{12}$$

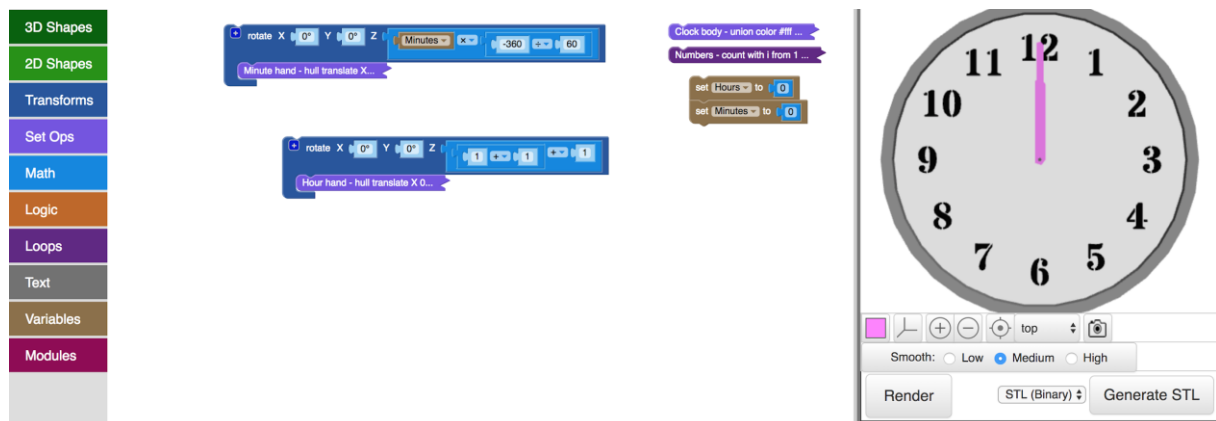
This equals the incremental movement of each hour. If we multiply this with the “hours” variable we will get:

$$hours \times \frac{360^\circ}{12}$$

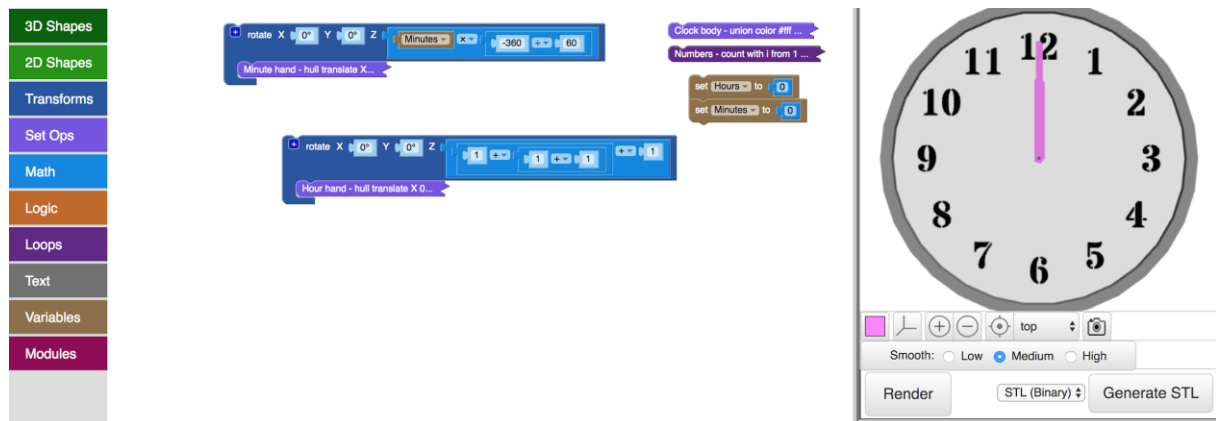
Drag and drop a summation block from the Math folder. Place it in the Z-value of the rotate block that rotates the hour hand.



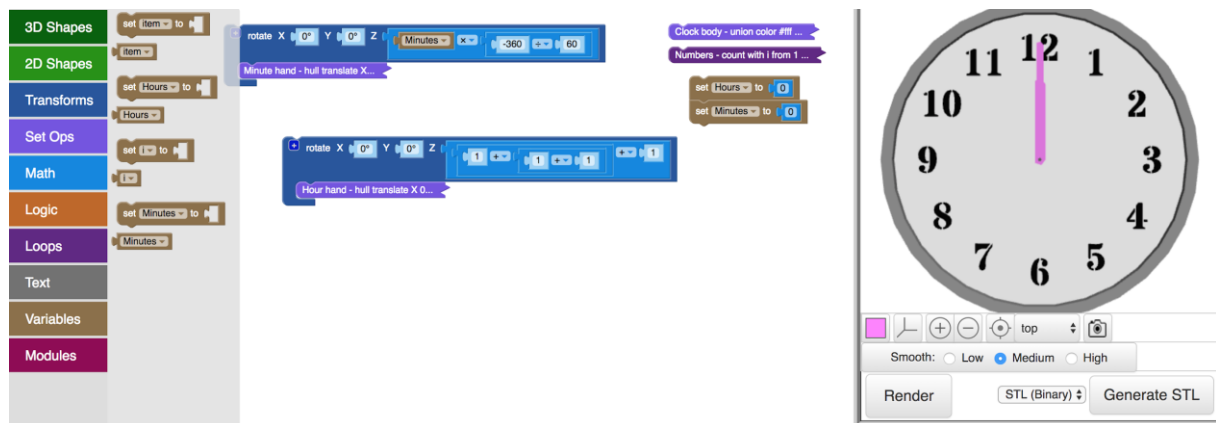
Drag and drop another summation block and place it in the first term of the summation block.



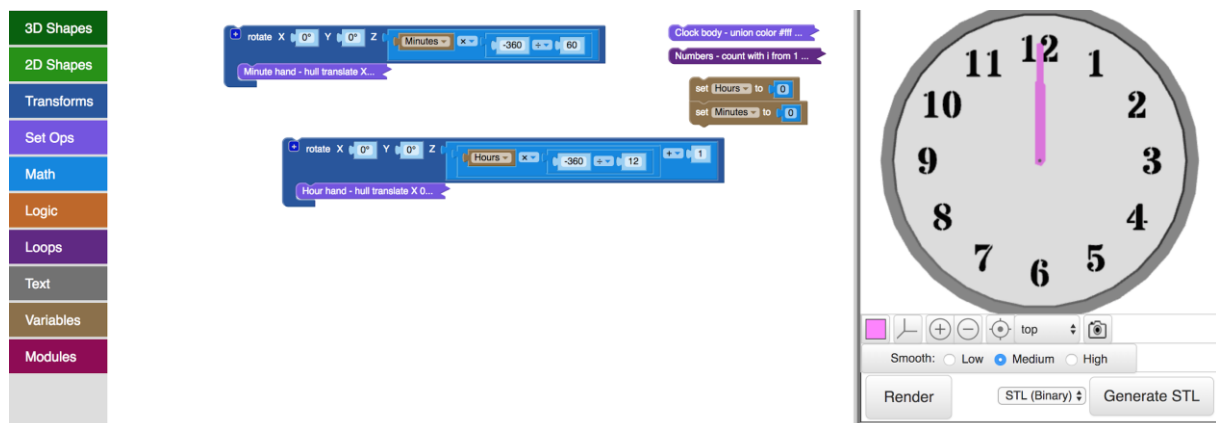
Drag and drop yet another summation block and place it in the middle term of the summation block.



Drag and drop a “Hours” variable from the Variables folder.

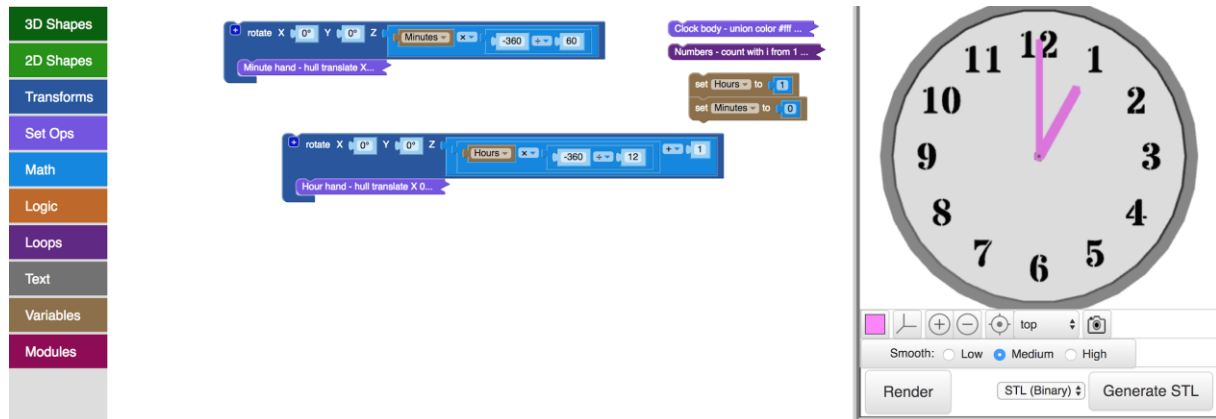


Place it in the first term of the Z-value. Multiply this variable with -360 divided by 12.

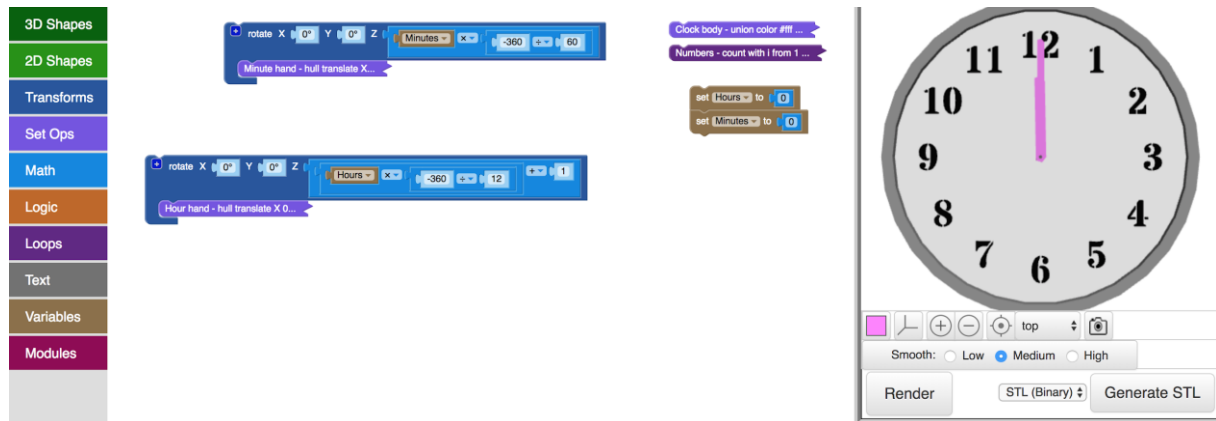


We need the minus sign to turn the hour hand the right direction about the Z-axis.

Test the function by typing 1 in the Hours set block. Press render and check that the hour hand points at 1.



Make the hour value 0 and press render. Now we will take care of the effect from the minute value on the hour value.



We have this equation:

$$hours \times \frac{360^\circ}{12}$$

We also know this:

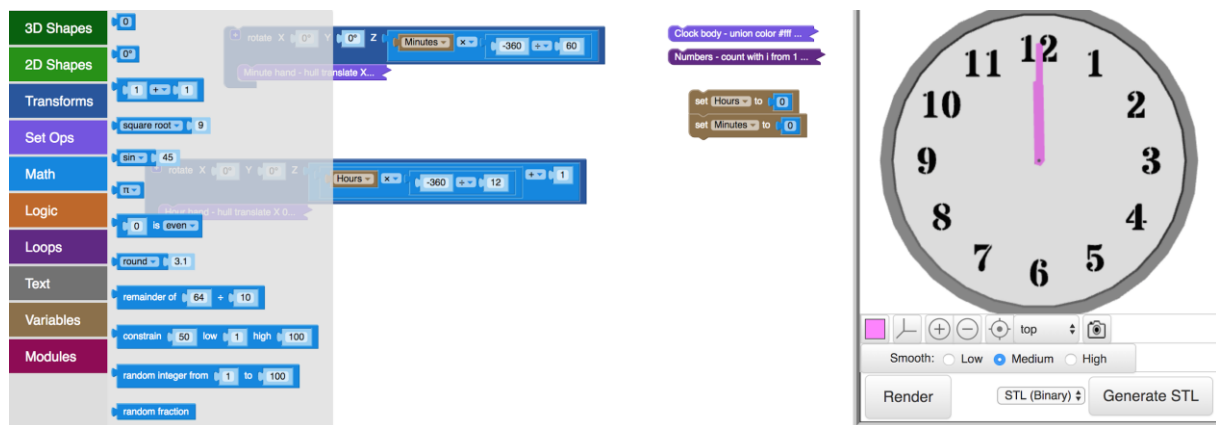
$$hours = \frac{minutes}{60}$$

If we insert the second equation into the first equation we get an equation for the movement of the hour hand with respect to the minute value:

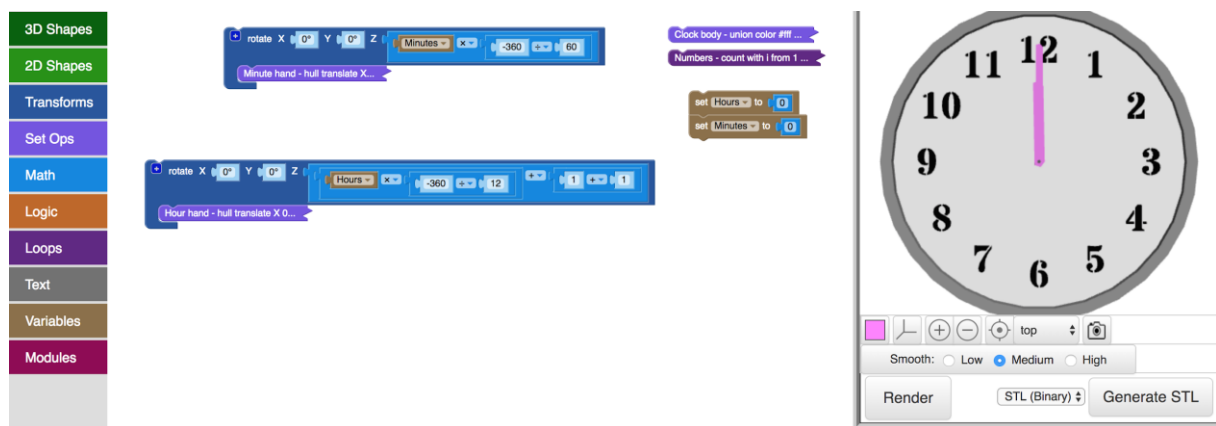
$$\frac{minutes}{60} \times \frac{360^\circ}{12}$$

Let's write this in BlocksCAD.

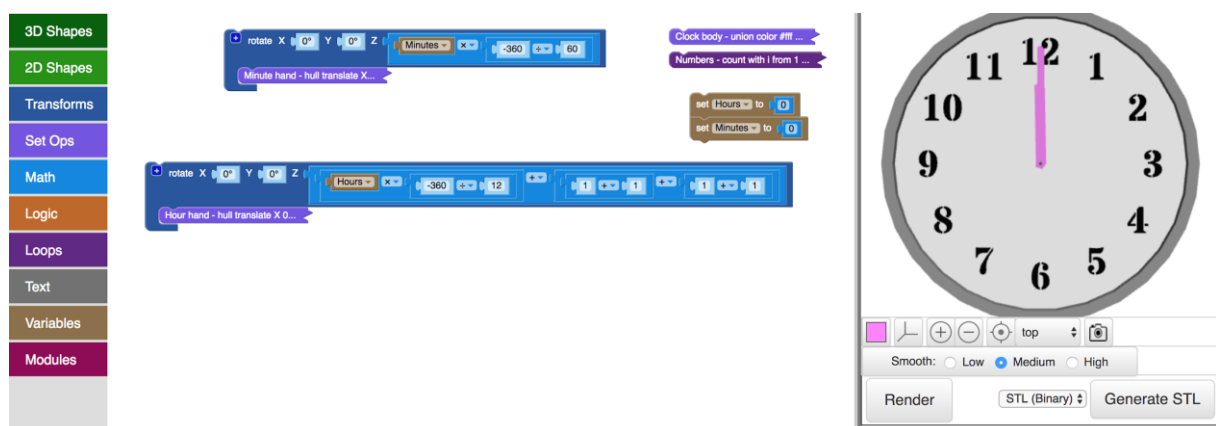
Drag and drop a summation block from the Math folder.



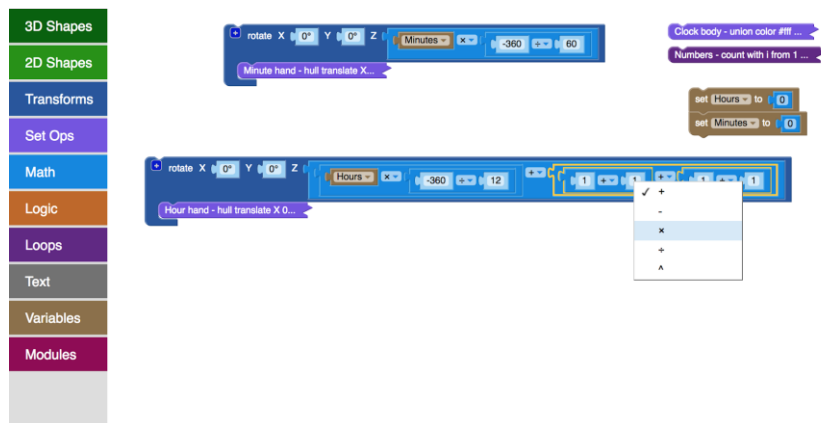
Insert it in the last term of the summation block in the Z-value of the rotate block.



Drag and drop another two summations blocks and insert them in the two term of the new summation block.

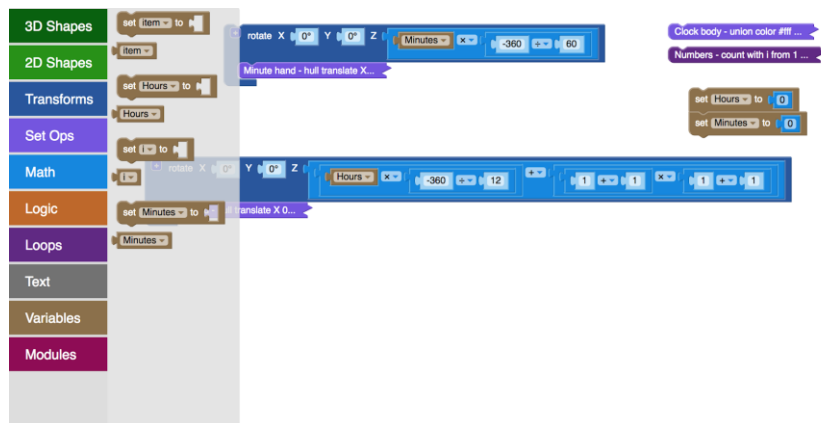


Multiply the two blocks.




Scratch code editor showing a clock project. The 'Minutes' variable is being multiplied by 1 in the 'Hour hand - hull translate X 0...' block. The 'Hours' variable is being multiplied by 1 in the 'Minute hand - hull translate X...' block. The 'Clock body - union color #ff...' and 'Numbers - count with i from 1...' blocks are also visible.

Drag and drop a "Minutes" variable from the Variables folder.



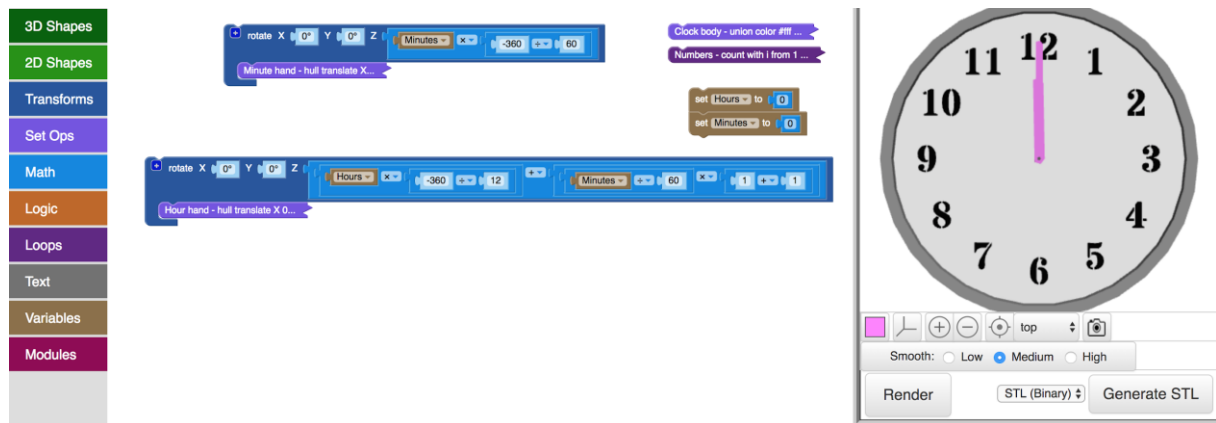
Scratch code editor showing a clock project. The 'Minutes' variable is being multiplied by 1 in the 'Hour hand - hull translate X 0...' block. The 'Hours' variable is being multiplied by 1 in the 'Minute hand - hull translate X...' block. The 'Clock body - union color #ff...' and 'Numbers - count with i from 1...' blocks are also visible.

Insert it in the first term of the second term in the summation block.

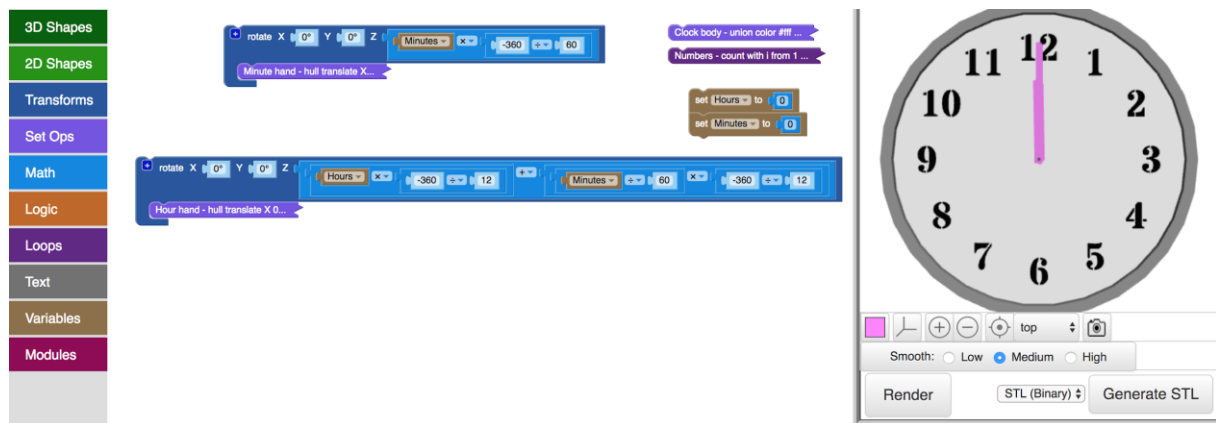


Scratch code editor showing a clock project. The 'Minutes' variable is being multiplied by 1 in the 'Hour hand - hull translate X 0...' block. The 'Hours' variable is being multiplied by 1 in the 'Minute hand - hull translate X...' block. The 'Clock body - union color #ff...' and 'Numbers - count with i from 1...' blocks are also visible.

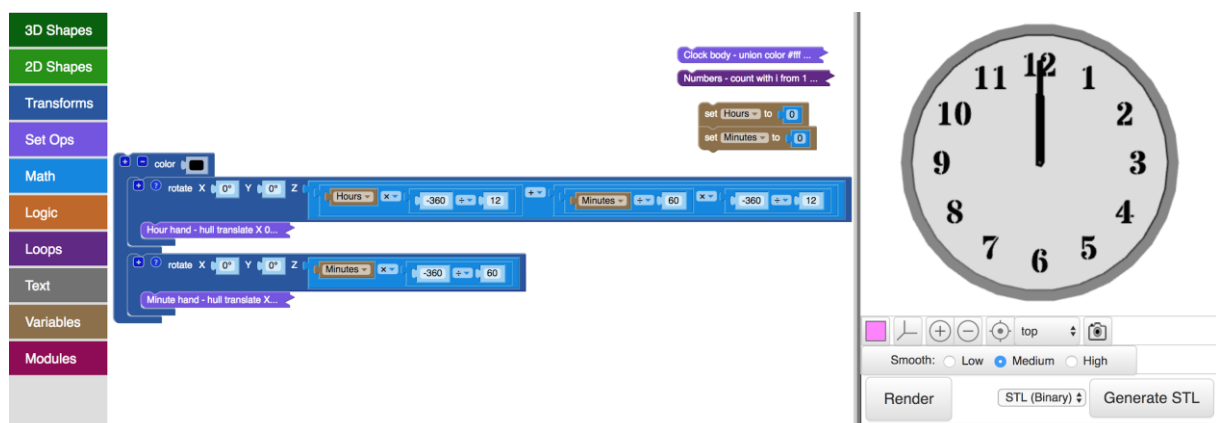
Divide the “Minutes” variable by 60.



Multiply this fraction by -360 divided by 12. The minus sign makes the rotation the correct way.



Let's add some color to the hour and minute hands. Drag and drop a color block from the Transforms folder. Put the two rotate blocks in this color block and pick a color. Press render.



Add a comment to the color block. Name the color block “Hands” and collapse it.

Clean up the window.



You have now built an analogue clock. You can input your desired time and render the clock to see the time animated.

Example: 4:43

