

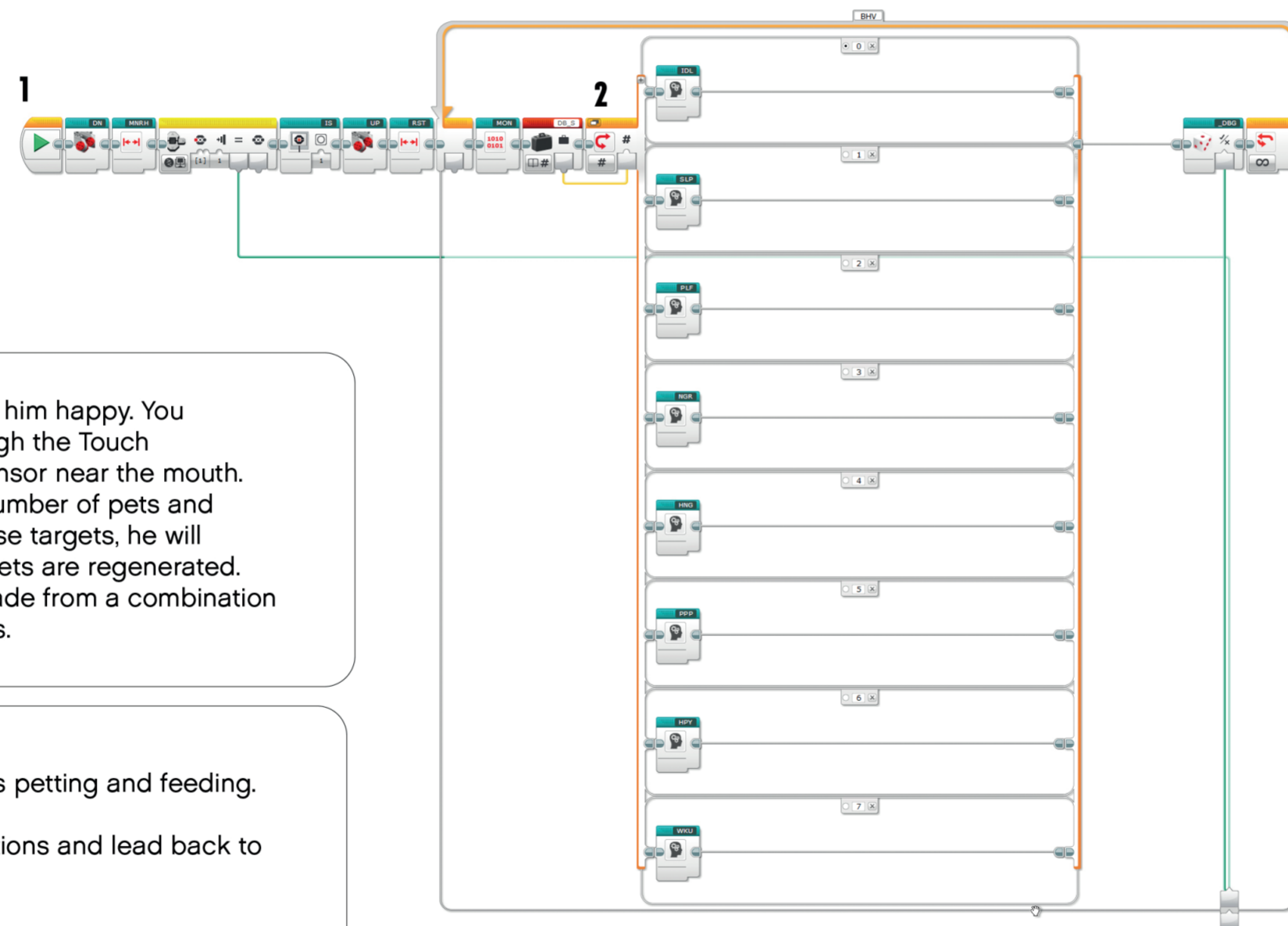
# Program Descriptions

## Puppy



### Overview

The puppy is an advanced behavior program that can move in and out of 8 possible behaviors. A series of My Blocks handle the simple movements and physical interactions while others dive into checking and changing the behavior status itself. NOTE: this image shows an expanded version of the program with the main switch in Flat Mode.



**1** The first set of blocks initialize the robot. First sitting the robot down with My Block DN and then allowing the user to move the head to the right position with My Block MNRH. The next Sensor Block allows a debug mode triggered within the Loop BHV. The next My Blocks change the eye display, stand the robot up and reset all motors, variables, and timers.

**2** Loop BHV has all the behaviors. My Block MON checks the counts of petting and feeding and outputs to Variable DB\_S. The Switch In Number Mode takes the variable and outputs to 8 different behaviors. Each My Block is a series of display, move, and checks for variables. My Block \_\_DBG displays the relevant variables and targets so you can see what is happening and where you are in the robot behavior. It is only activated if you press and hold the left brick button when resetting the puppy head before the loop starts.

The goal of the puppy is to make him happy. You can pet and feed the robot through the Touch Sensor on his back and Color Sensor near the mouth. There is a randomly generated number of pets and feeds as targets. If you reach those targets, he will become happy and then the targets are regenerated. Each of the other behaviors is made from a combination of the petting and feeding targets.

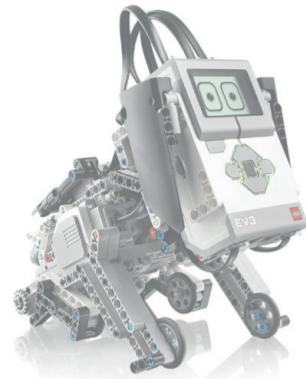
#### My Blocks in Switch

IDL- Idle - here the puppy checks petting and feeding.

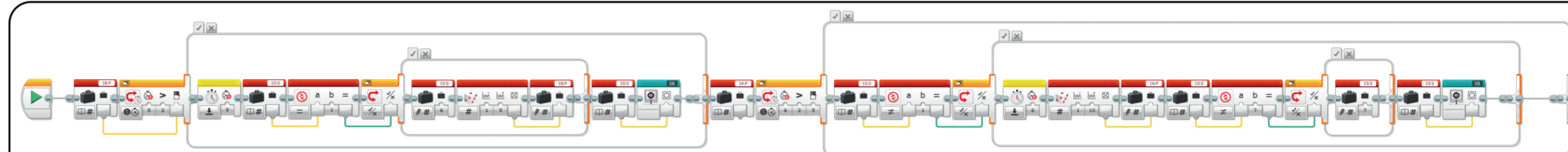
The rest of the My Blocks are actions and lead back to My Block IDL  
 SLP-Sleeping  
 PLF-Playful  
 NGR-Angry  
 HNG-Hungry  
 PPP- Bathroom  
 HPY-Happy  
 WKU-Waking Up

# Program Descriptions

## Puppy

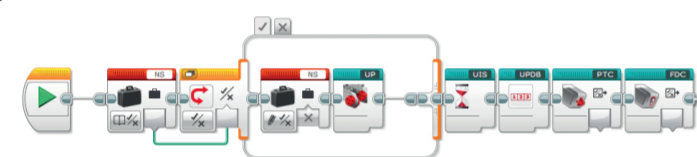


UIS



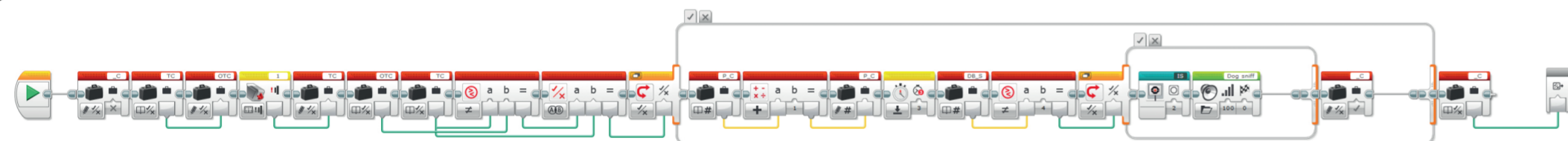
My Block UIS changes the eyes to give the robot more life. The robot randomly changes which direction the eyes look and blink. The first switch in Timer Mode will be true if the timer is greater than the value given to Variable IBP. Timer 5 is reset and Variable ISS is compared to 1; checking if the eyes are closed already. If so, the robot will look right and send a random value to Variable IBP, otherwise the eyes will close. The eye value is then updated to My Block IIS. The next switch should be activated with Variable IAS in Timer Mode using Timer 6. In the true case, it checks if the robots eyes are not closed. If so, Timer 6 is reset, a random number is wired to Variable IAS, and a last check to see if the eyes are looking left, or ISS is 7. It will switch the value of ISS to 6 or 7 as needed. The value is then wired to My Block IS.

IDL



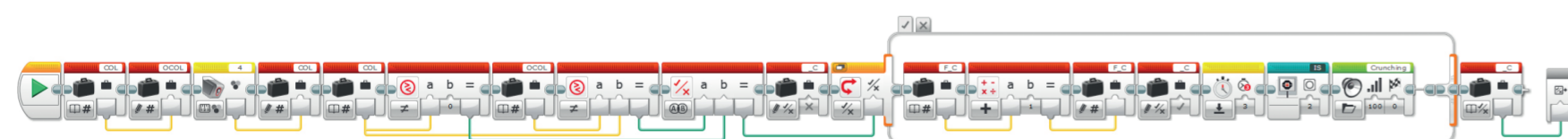
My Block IDL is the idle state of the robot. Here it makes sure the robot is standing and will update the eyes of the robot with My Block UIS. It will then check the pet and feed count combinations with My Block UPDB. Finally, it will check the sensors as needed if the robot will be pet or fed with My Blocks PTC and FDC.

PTC



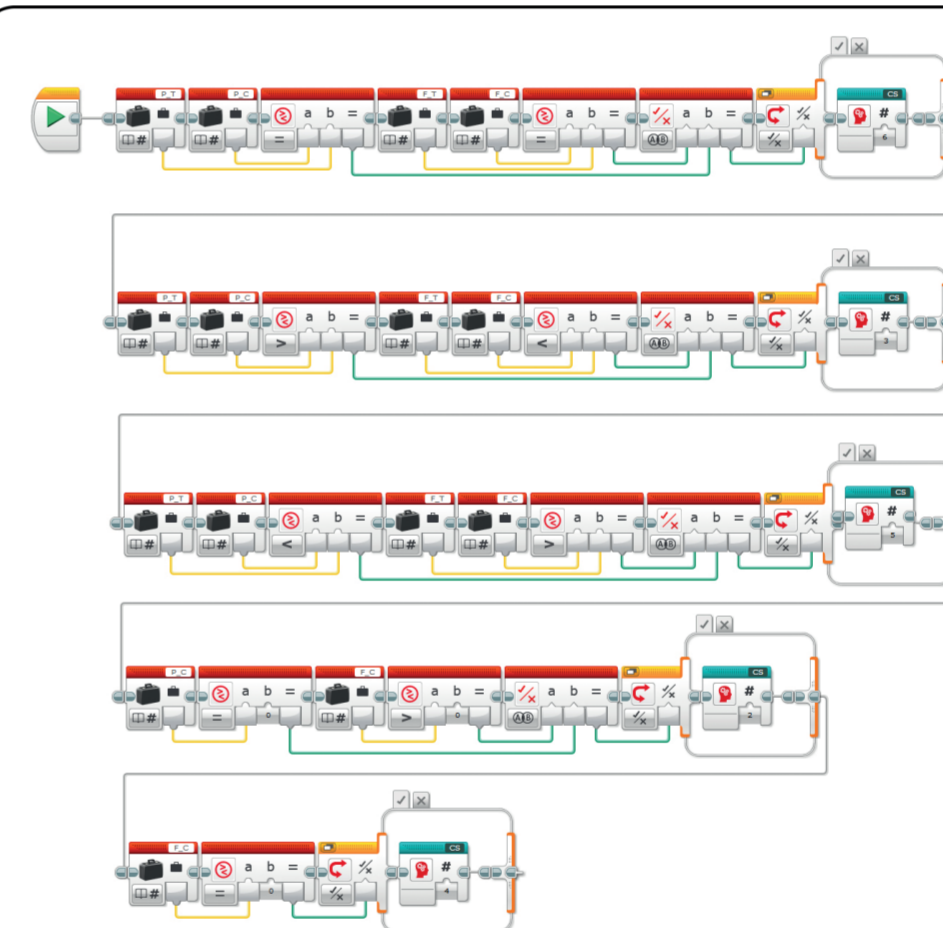
My Block PTC checks the Touch Sensor of the robot and adds to the pet count Variable P\_C. First, it checks to see if the Touch Sensor was already pushed. If it is, the pet will not count. This is done with a series of Variables and Compare Blocks. By storing the previous touch state into variables at different times, their value can be checked. If the old states do not match the new state, 1 is added to P\_C, then Timer 3 is reset, the eyes change, and a sound is played. Variable \_C is an output of truth, to show the My Block has been activated.

FTC



My Block FDC checks the Color Sensor of the robot and adds to the feed count Variable F\_C. First, it checks to see if the color shown is the same as it was on the last check. If it is, the feed will not count. This is done with a series of Variables and Compare Blocks. By sorting the colors into variables at different times, their value can be checked. If the old colors do not match, 1 is added to F\_C, Timer 3 is reset, and a sound is played. Variable \_C is an output of truth, to show the My Block has been activated.

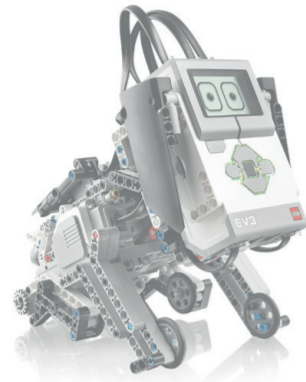
UPDB



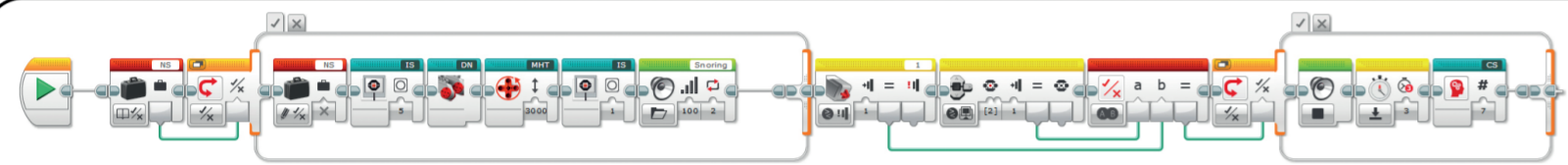
My Block UPDB checks the various combinations of pet counts and feed counts which lead up to the various behaviors. The Variable CS is for the state, and combinations are as follows in terms of the variables:  
P\_C=P\_T and F\_C=F\_T CS=6 (Happy)  
P\_C>P\_T and F\_C<F\_T CS=3 (Angry)  
P\_C<P\_T and F\_C>F\_T CS=5 (PP)  
P\_C=0 and F\_C>0 CS=2 (Playful)  
F\_C=0 CS=4 (Hungry)  
The other states are addressed in other My Blocks.  
CS=1 (Sleeping) is addressed in My Block MON and  
CS=7 (Wake Up) is in My Block SLP. More combinations here can make more behaviors for the robot.

# Program Descriptions

## Puppy

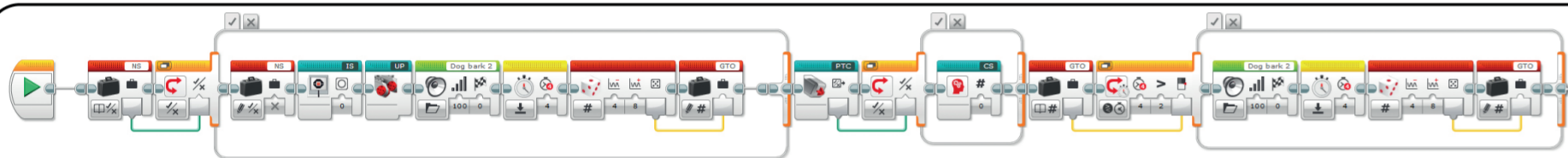


### SLP



My Block SLP makes the robot sleep. The eyes will close, the head will move down and the robot will sit. The robot will also snore. My Blocks IS, DN, and MHT are used and inputs are placed in the input box. If the robot is pet via the touch sensor or hit on the nose via the center brick button, he will stop snoring and wake up. Sensor Blocks and a Logic Block are used here, and the My Block CS is given an input of 7.

### PLF



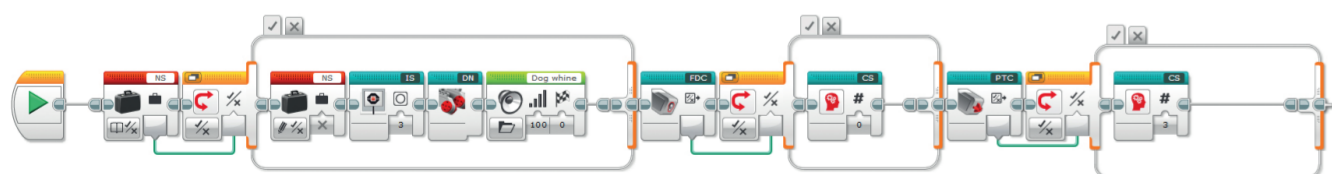
My Block PLF will make the robot playful. The robot will make sure the eyes are open fully, it will stand up if it is not already and bark. Timer 4 is reset and a random number is placed into Variable GTO. The robot checks if it is pet. If it is, the robot will go back to idle before it changes state. The robot waits for a time as given from Variable GTO and barks again. Timer 4 is reset, and GTO is randomized again.

### NGR



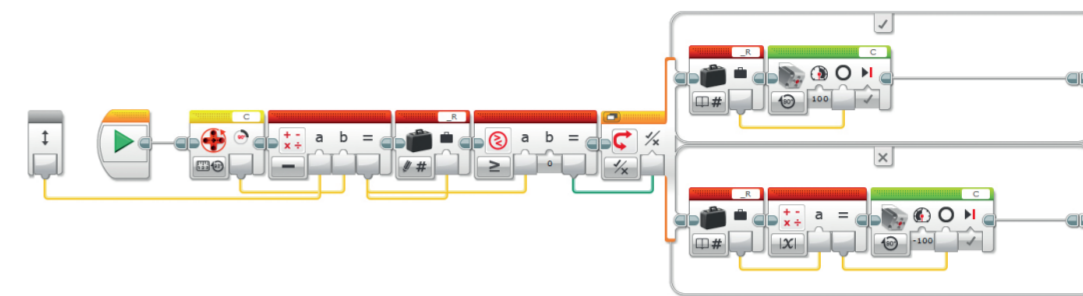
My Block NGR makes the robot angry. The eyes get an angry expression and the robot growls and barks. The pet count will also decrease by 1, and the robot goes back to an idle state.

### HNG



My Block HNG makes the robot hungry. The robot will look sad, sit down and whine. If it is fed, the robot will go back to idle. If it is pet, the robot will become playful.

### MHT



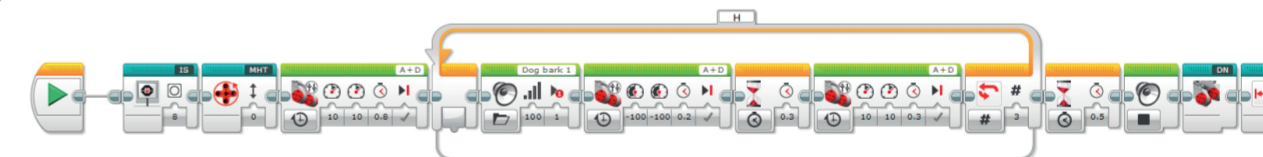
My Block MHT moves the head of the robot. By taking an input and subtracting that from the current position of the head, as read by the Motor Sensor Block, the motor is moved that amount of degrees at full power. The Compare Block and switch ensure the proper direction of movement.

### PPP



My Block PPP will have the robot 'go to the bathroom'. The eyes will go up, the robot stands, and lifts its leg. Loop P will make the leg shake 3 times, and then the leg goes down. The feed count will go to 1 and the robot will go back to idle.

### HPY



My Block HPY will make the robot happy. The eyes will display as hearts, and the robot will sit down. The head is checked to be in the right position as well. Once sitting, it will hop in place and bark 3 times as seen in loop H. After waiting for .5 seconds, the robot will sit properly and reset the pet and feed counts again. This is all done in My Block RST. The robot will also end in its idle state.

### WKR



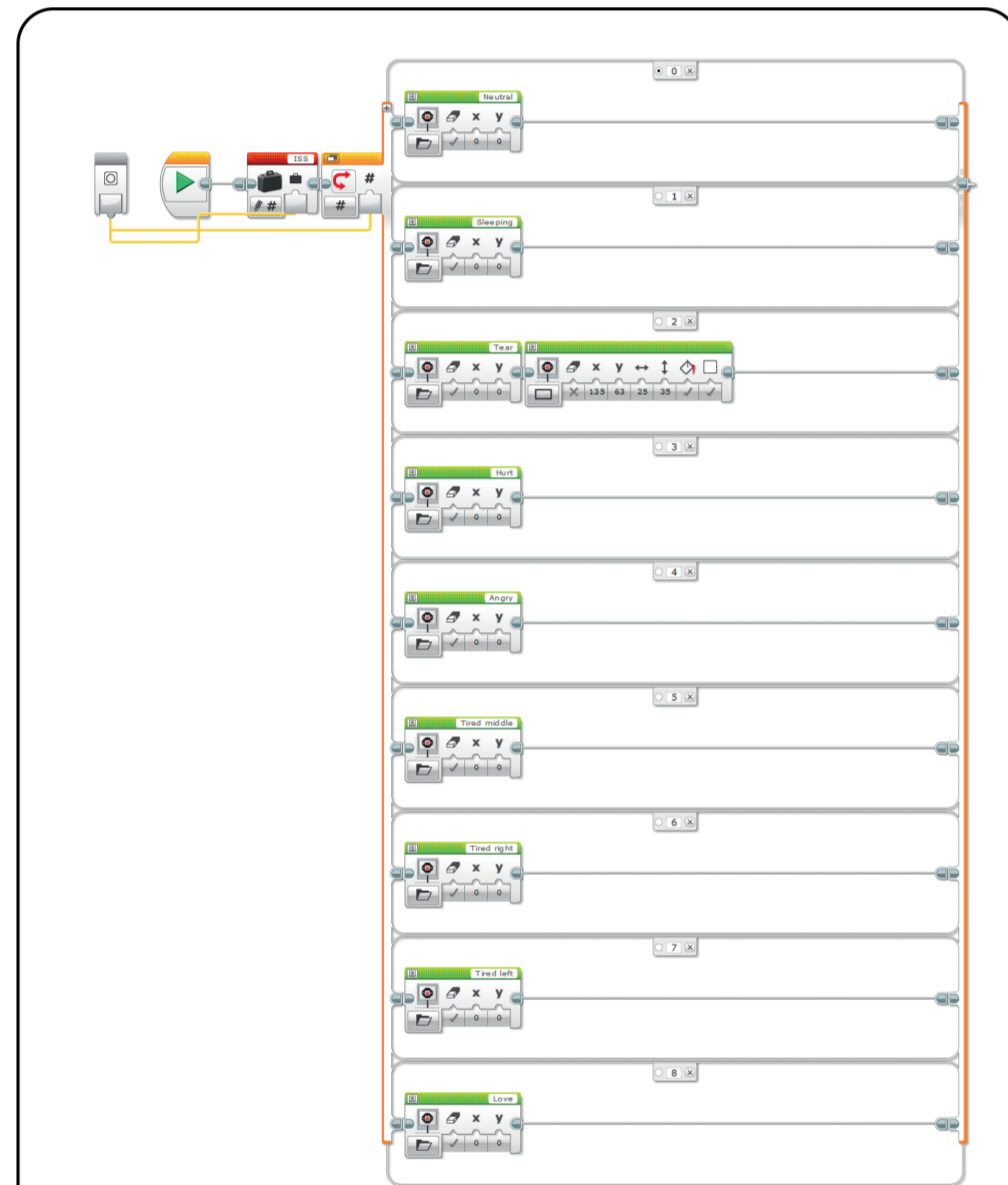
My Block WKR will wake up the robot after it is sleeping. The eyes will be tired, it will whine, and move the head back to its starting position. The robot will sit and stretch with My Block STL. It will wait for 1 second before standing up and go back to idle.

# Program Descriptions

## Puppy



IS

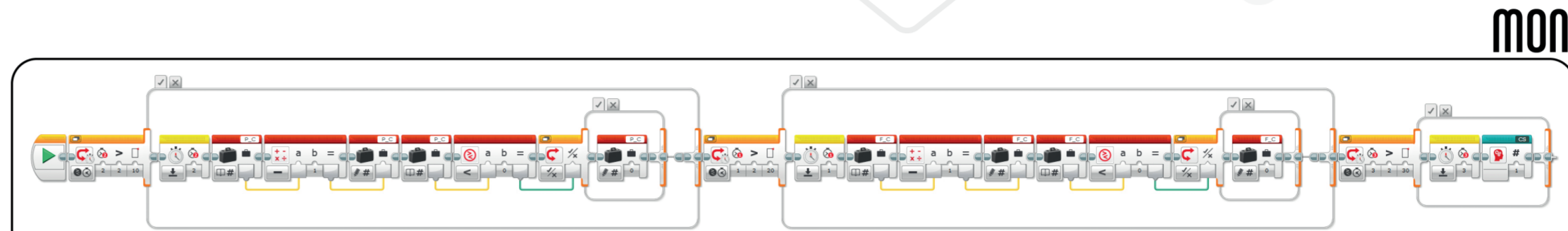


My Block IS takes an input for the eyes of the robot. A Switch Block in Number Mode can allow for many options. In case 3, the second Display Block clears the tear from the display.

RST



My Block RST resets the pet and feed variables as well as the timers and the My Block CS. A random number between 3 and 6 is set for the pet and feed targets.



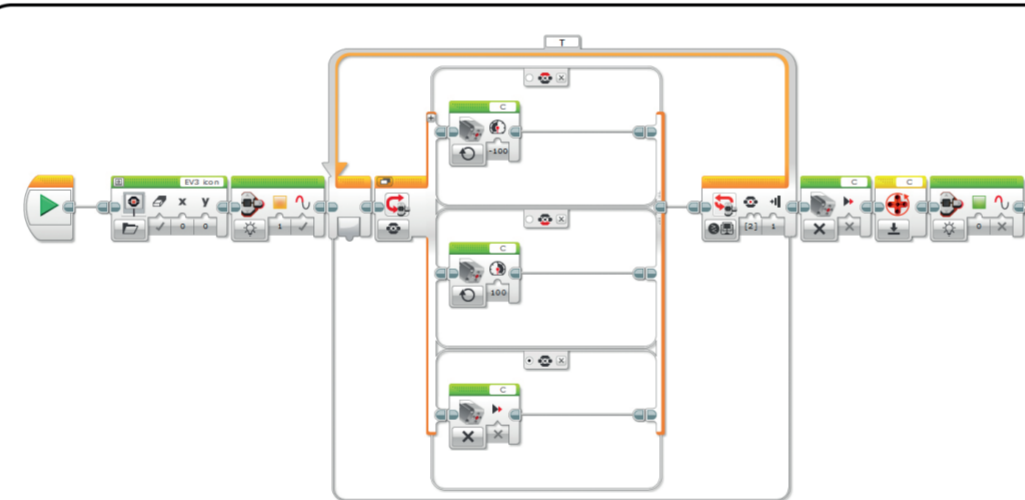
My Block MON monitors the pet count and feed count. Each have their own timer and after 2 seconds of idleness, the timers will reset and 1 is taken away from the count. An independent timer allows each count to run separately when interacted with. Lastly, if there is no activity, timer 3 will run, and CS=1 will make the robot go to sleep.

DN



My Block DN sits the robot down and resets the motors.

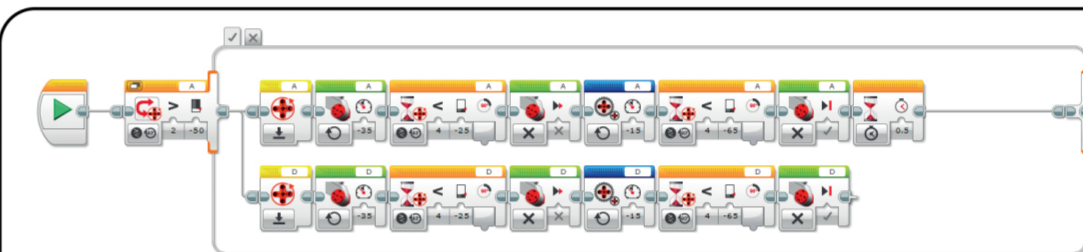
MNRH



My Block MNRH is a manual reset for the head of the robot. The up and down brick buttons move the head and the center button will end Loop T. The motor is then released and reset. Lining up the head of the robot gives the best position for the rest of the behaviors.

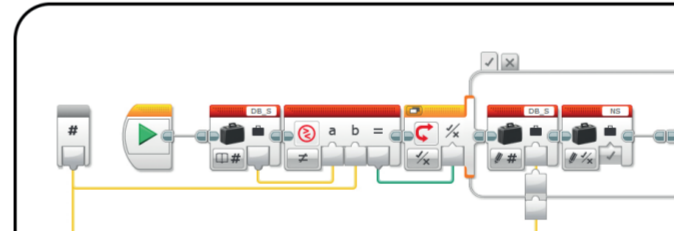
MON

UP



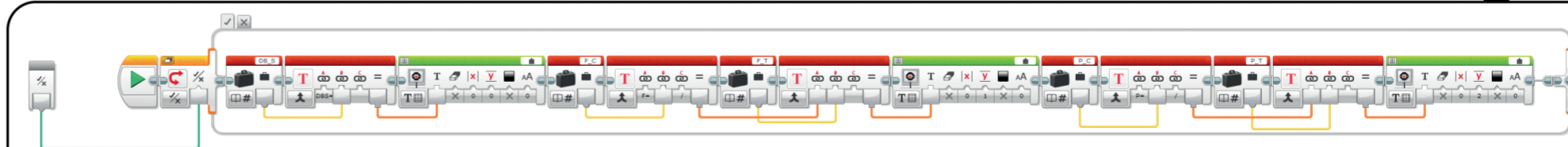
My Block UP stands the robot up. Using the motor in On Mode with a Wait For Block ensures a smooth transition between the motor power. This smooth transition is needed for making sure the robot does not fall on its face when standing up.

CS



My Block CS changes the state of the robot. It checks if the state was the same as it was in the previous loop. If it is the same, it will pass through until the state changes, otherwise it will change the state and turn Variable NS to true, which allows the states to run.

DBG



My Block \_DBG, when activated, will display the values of the 5 variables. The pet count, P\_C; feed count, F\_C; as well as the pet target, P\_T; feed target, F\_T; and lastly the state variable, DB\_S.