Heart Model Documentation

Table of Contents

[Lists 2](#_Toc94556945)

[Variables 2](#_Toc94556946)

[Setup Pins Function 3](#_Toc94556947)

[Read Input Function 3](#_Toc94556948)

[Toggle Speed Function 3](#_Toc94556949)

[Run Bundle of His Function 4](#_Toc94556950)

[Run Upper Chamber Function 4](#_Toc94556951)

[Run Left Branch Function 4](#_Toc94556952)

[Run Right Branch Function 5](#_Toc94556953)

[Show Light Function 5](#_Toc94556954)

[How to add a button? 6](#_Toc94556955)

# Lists

* Pins: This corresponds to the board pin that the buttons connected to.

const int Pins[6] = {2, 3, 4, 5, 6};

* Upper Chamber: This is a list of the light numbers that are in the upper chamber (SA/AV Nodes). You can use this if you want to only show that lights in the SA/AV nodes.

const int upperChamber[5] = {5, 4, 0, 14};

* Bundle Of His: This is a list of the light numbers that are in the middle of the heart (Bundle Of His). You can use this if you want to only show that lights in the middle of the heart.

const int bundleOfHis[9] = {1, 2, 3, 6, 7, 8, 9, 10};

* Left Branch: This is a list of the light numbers that are in the left of the heart. You can use this if you want to only show that lights in the left of the heart.

const int leftBranch[4] = {11, 12, 13};

* Right Branch: This is a list of the light numbers that are in the right of the heart. You can use this if you want to only show that lights in the right of the heart.

const int rightBranch[4] = {15, 16, 17};

# Variables

* New State: Is a flag used to identify whether a button has been pressed or not. It is used in the readInput function which awaits a button to be pressed to update the flag.
* Button: Is an integer that is set in the readInput function and used in the loop function to check which button got pressed and run a specific code.
* Is Normal Speed: Is a flag that is set by toggleSpeed function and implemented into every function to set the delay to control light speed.

# Setup Pins Function

Sets up the listeners on the pins that the buttons are connected to. If you want to increase the number of buttons add the pin number that the button is connected to and increase buttonsNbr to match the number of buttons.

void setupPins() {

int buttonsNbr = 6;

for (int i = 0; i < buttonsNbr; i++) {

pinMode(Pins[i], INPUT\_PULLUP);

}

}

# Read Input Function

Runs a loop to detect which button pressed and updates newState and button variables. If you increased the number of buttons then you need to increase buttonNbr variable in the function.

void readInput() {

/\* Loop to read buttons increase the number in the Pins list and below variable if you add more buttons. \*/

int buttonNbr = 6;

for (int i = 0; i < buttonNbr && newState == HIGH; i++) {

button = Pins[i];

newState = digitalRead(Pins[i]);

}

}

# Toggle Speed Function

Switches the isNormalSpeed flag to opposite value. Thereby, toggling it between true or false.

void toggleSpeed() {

isNormalSpeed = !isNormalSpeed;

}

# Run Bundle of His Function

Runs a loop that uses the bundleOfHis list to know which light to turn on and off. The function take the delayNbr to control light speed and lightColor to control light colour. By calling the function and passing the parameters the light in the middle of the heart (known as the bundle of His) will light up in order top to bottom. You do not need to modify this function just call it with the parameters.

void runBundleOfHis(int delayNbr, int color[4]) {

for(int a= 0; a<=7; a++) {

showLight(bundleOfHis[a], delayNbr, color);

}

}

# Run Upper Chamber Function

Runs a loop that uses the upperChamber list to know which light to turn on and off. The function take the delayNbr to control light speed and lightColor to control light colour. By calling the function and passing the parameters the light in the upper chamber of the heart (SA/AV Nodes). You do not need to modify this function just call it with the parameters.

void runUpperChamber(int delayNbr, int color[4]) {

for(int a= 0; a<=7; a++) {

showLight(upperChamber[a], delayNbr, color);

}

}

# Run Left Branch Function

Runs a loop that uses the leftBranch list to know which light to turn on and off. The function take the delayNbr to control light speed and lightColor to control light colour. By calling the function and passing the parameters the light in the left branch of the heart. You do not need to modify this function just call it with the parameters.

void runLeftBranch(int delayNbr, int color[4]) {

for(int a= 0; a<=7; a++) {

showLight(leftBranch[a], delayNbr, color);

}

}

# Run Right Branch Function

Runs a loop that uses the rightBranch list to know which light to turn on and off. The function take the delayNbr to control light speed and lightColor to control light colour. By calling the function and passing the parameters the light in the right branch of the heart. You do not need to modify this function just call it with the parameters.

void runLeftBranch(int delayNbr, int color[4]) {

for(int a= 0; a<=7; a++) {

showLight(leftBranch[a], delayNbr, color);

}

}

# Show Light Function

Show Light function takes the light number, delay which controls how fast the light turn off and on, and light colour, which take an RGB value.

{ red, green, blue }

To make it easier to pick a colour you can Google RGB colour picker and use the value and store it into a variable and pass it to the function. You don't need to modify this function.

void showLight(int lightNbr, int delayNbr, int color[4]) {

strip.setPixelColor(lightNbr, strip.Color(color[0], color[1], color[2]));

strip.show();

delay(delayNbr);

strip.setPixelColor(lightNbr, strip.Color(0, 0, 0));

strip.show();

delay(delayNbr);

}

# How to add a button?

To add a button you need:

* The light number (lightNbr variable).
* The delay (delayNbr variable).
* An if statement to toggle the delay between normal speed and slow speed.
* Light colour variable to set the colour of the light.
* Show light function that takes the previous parameters to turn on the light.

void your\_button\_name() {

int lightNbr = 5;

int delayNbr;

// This is used for the speed toggle button

if (isNormalSpeed == false) {

delayNbr = 225;

} else {

delayNbr = 25;

}

lightColor[4] = { 255, 0, 255 };

showLight(lightNbr, delayNbr, lightColor);

}

All the buttons have the same basic set up. But it differs in the execution.