RGB ARDUINO CONTROL CODE

\_RGBL\_Digital.pde   
//--- bof RGBL - RGB Digital Preamble//RGB LED pinsint ledDigitalOne[] = {14, 15, 16}; //the three digital pins of the first digital LED 14 = redPin, 15 = greenPin, 16 = bluePinint ledDigitalTwo[] = {9, 10, 11}; //the three digital pins of the first digital LED 14 = redPin, 15 = greenPin, 16 = bluePinint ledDigitalThree[] = {3, 5, 6}; //the three digital pins of the first digital LED 14 = redPin, 15 = greenPin, 16 = bluePinconst boolean ON = LOW; //Define on as LOW (this is because we use a common Anode RGB LED (common pin is connected to +5 volts)const boolean OFF = HIGH; //Define off as HIGH//Predefined Colorsconst boolean RED[] = {ON, OFF, OFF}; const boolean GREEN[] = {OFF, ON, OFF}; const boolean BLUE[] = {OFF, OFF, ON}; const boolean YELLOW[] = {ON, ON, OFF}; const boolean CYAN[] = {OFF, ON, ON}; const boolean MAGENTA[] = {ON, OFF, ON}; const boolean WHITE[] = {ON, ON, ON}; const boolean BLACK[] = {OFF, OFF, OFF}; //An Array that stores the predefined colors (allows us to later randomly display a color)const boolean\* COLORS[] = {RED, GREEN, BLUE, YELLOW, CYAN, MAGENTA, WHITE, BLACK};//--- eof RGBL - RGB Digital Preamblevoid setup(){ for(int i = 0; i < 3; i++){ pinMode(ledDigitalOne[i], OUTPUT); //Set the three LED pins as outputs pinMode(ledDigitalTwo[i], OUTPUT); //Set the three LED pins as outputs pinMode(ledDigitalThree[i], OUTPUT); //Set the three LED pins as outputs }}void loop(){/\* Example - 1 Set a color Set the three LEDs to any predefined color\*/ setColor(ledDigitalOne, RED); //Set the color of LED one setColor(ledDigitalTwo, GREEN); //Set the color of LED two setColor(ledDigitalThree, BLUE); //Set the color of LED three/\* Exampe - 2 Go through Random Colors Set the LEDs to a random color\*/ //int rand = random(0, sizeof(COLORS) / 2); //get a random number within the range of colors // setColor(ledDigitalOne, COLORS[rand]); //Set the color of led one to a random color //rand = random(0, sizeof(COLORS) / 2); //Set the color of LED 2 to a random color // setColor(ledDigitalTwo, COLORS[rand]); //rand = random(0, sizeof(COLORS) / 2); //Set the color of LED 3 to a random color // setColor(ledDigitalThree, COLORS[rand]); //delay(1000); }/\* Sets an led to any color led - a three element array defining the three color pins (led[0] = redPin, led[1] = greenPin, led[2] = bluePin) color - a three element boolean array (color[0] = red value (LOW = on, HIGH = off), color[1] = green value, color[2] =blue value)\*/void setColor(int\* led, boolean\* color){ for(int i = 0; i < 3; i++){ digitalWrite(led[i], color[i]); }}/\* A version of setColor that allows for using const boolean colors\*/void setColor(int\* led, const boolean\* color){ boolean tempColor[] = {color[0], color[1], color[2]}; setColor(led, tempColor);}

PWM RGB LED

\_RGBL\_AnalogTest  
*//---bof---RGBL-Analog Preamble//RGB LED pinsint ledAnalogOne[] = {3, 5, 6}; //the three pins of the first analog LED* 3 = redPin, 5 = greenPin, 6 = bluePin //These pins must be PWMint ledAnalogTwo[] = {9, 10, 11}; //the three pins of the second analog LED 9 = redPin, 10 = greenPin, 11 = bluePin //These pins must be PWM//Defined Colors (different RGB (red, green, blue) values for colors//(to add your own ie. fuscia experiment and then add to the list)const byte RED[] = {255, 0, 0}; const byte ORANGE[] = {83, 4, 0}; const byte YELLOW[] = {255, 255, 0}; const byte GREEN[] = {0, 255, 0}; const byte BLUE[] = {0, 0, 255}; const byte INDIGO[] = {4, 0, 19}; const byte VIOLET[] = {23, 0, 22}; const byte CYAN[] = {0, 255, 255}; const byte MAGENTA[] = {255, 0, 255}; const byte WHITE[] = {255, 255, 255}; const byte BLACK[] = {0, 0, 0}; const byte PINK[] = {158, 4, 79}; //---eof---RGBL-Analog Preamblevoid setup(){ for(int i = 0; i < 3; i++){ pinMode(ledAnalogOne[i], OUTPUT); //Set the three LED pins as outputs pinMode(ledAnalogTwo[i], OUTPUT); //Set the three LED pins as outputs } setColor(ledAnalogOne, BLACK); //Turn off led 1 setColor(ledAnalogTwo, BLACK); //Turn off led 2}void loop(){/\* Example 1 - Defined Colors Set to a known color (you can use any of the above defined colors)\*/ setColor(ledAnalogOne, MAGENTA); /\* Example 2 - Any Color Set the LED to any color you like\*/ //byte tempColor[] = {12,34,12}; //the RGB (red, gren blue) value for a color to display //setColor(ledAnalogOne, tempColor); /\*Example 3 - Fading Fade the LED between two colors (this will go from red to green to blue then back to red)\*/ //fadeToColor(ledAnalogOne, RED, GREEN, 10); //fadeToColor takes 4 parameters //ledAnalogOne - an array with 3 values defining the red, green and blue pins of the LED //RED - This is the start color //GREEN - This is the end color //10 - the delay (in milliseconds between updates) (determines the fade speed) //fadeToColor(ledAnalogOne, GREEN, BLUE, 10); //Fades from Green to Blue //fadeToColor(ledAnalogOne, BLUE, RED, 10); //Fades from Blue to Red}/\* Sets the color of the LED to any RGB Value led - (int array of three values defining the LEDs pins (led[0] = redPin, led[1] = greenPin, led[2] = bluePin)) color - (byte array of three values defing an RGB color to display (color[0] = new Red value, color[1] = new Green value, color[2] = new Red value\*/void setColor(int\* led, byte\* color){ for(int i = 0; i < 3; i++){ //iterate through each of the three pins (red green blue) analogWrite(led[i], 255 - color[i]); //set the analog output value of each pin to the input value (ie led[0] (red pin) to 255- color[0] (red input color) //we use 255 - the value because our RGB LED is common anode, this means a color is full on when we output analogWrite(pin, 0) //and off when we output analogWrite(pin, 255). }}/\* A version of setColor that takes a predefined color (neccesary to allow const int pre-defined colors \*/void setColor(int\* led, const byte\* color){ byte tempByte[] = {color[0], color[1], color[2]}; setColor(led, tempByte);}/\* Fades the LED from a start color to an end color at fadeSpeed led - (int array of three values defining the LEDs pins (led[0] = redPin, led[1] = greenPin, led[2] = bluePin)) startCcolor - (byte array of three values defing the start RGB color (startColor[0] = start Red value, startColor[1] = start Green value, startColor[2] = start Red value endCcolor - (byte array of three values defing the finished RGB color (endColor[0] = end Red value, endColor[1] = end Green value, endColor[2] = end Red value fadeSpeed - this is the delay in milliseconds between steps, defines the speed of the fade\*/ void fadeToColor(int\* led, byte\* startColor, byte\* endColor, int fadeSpeed){ int changeRed = endColor[0] - startColor[0]; //the difference in the two colors for the red channel int changeGreen = endColor[1] - startColor[1]; //the difference in the two colors for the green channel int changeBlue = endColor[2] - startColor[2]; //the difference in the two colors for the blue channel int steps = max(abs(changeRed),max(abs(changeGreen), abs(changeBlue))); //make the number of change steps the maximum channel change for(int i = 0 ; i < steps; i++){ //iterate for the channel with the maximum change byte newRed = startColor[0] + (i \* changeRed / steps); //the newRed intensity dependant on the start intensity and the change determined above byte newGreen = startColor[1] + (i \* changeGreen / steps); //the newGreen intensity byte newBlue = startColor[2] + (i \* changeBlue / steps); //the newBlue intensity byte newColor[] = {newRed, newGreen, newBlue}; //Define an RGB color array for the new color setColor(led, newColor); //Set the LED to the calculated value delay(fadeSpeed); //Delay fadeSpeed milliseconds before going on to the next color } setColor(led, endColor); //The LED should be at the endColor but set to endColor to avoid rounding errors}/\* A version of fadeToColor that takes predefined colors (neccesary to allow const int pre-defined colors \*/void fadeToColor(int\* led, const byte\* startColor, const byte\* endColor, int fadeSpeed){ byte tempByte1[] = {startColor[0], startColor[1], startColor[2]}; byte tempByte2[] = {endColor[0], endColor[1], endColor[2]}; fadeToColor(led, tempByte1, tempByte2, fadeSpeed);}