***Digital Watch/Clock***

**Components:**

* Arduino UNO
* 4-Digit 7 Segment Display
* Breadboard/ Prototyping PCB stripboard
* Shift Register-74HC595
* Connecting Wires
* DS3231 Real Time Clock (RTC) Module

**Connection Diagram:**

Diagram, schematic

Description automatically generated

**Libraries Used:**

#include<Wire.h>

#include<DS3231.h>

**Code:**

//Four-Digit 7 Segments Multiplexing using Arduino: Display time in HH:MM

#include <Wire.h>    //Library for SPI communication

#include <DS3231.h>   //Library for RTC module

#define latchPin 5

#define clockPin 6

#define dataPin 4

#define dot 2

DS3231 RTC;         //Declare object RTC for class DS3231

int h;              //Variable declared for hour

int m;              //Variable declared for minute

int thousands;

int hundreds;

int tens;

int unit;

bool h24;

bool PM;

void setup ()

{

    Wire.begin();

    pinMode(9,OUTPUT);

    pinMode(10,OUTPUT);

    pinMode(11,OUTPUT);

    pinMode(12,OUTPUT);

    pinMode(latchPin, OUTPUT);

    pinMode(clockPin, OUTPUT);

    pinMode(dataPin, OUTPUT);

    pinMode(dot,OUTPUT);

}

void loop ()

{

    digitalWrite(dot,HIGH);

    int h= RTC.getHour(h24, PM);  //To get the Hour

    int m = RTC.getMinute();      //TO get the minute

    int number = h\*100+m;         //Converts hour and minute in 4-digit

    int thousands = number/1000%10; //Getting thousands digit from the 4 digit

    int hundreds = number/100%10;  //Getting hundreds digit from 4 digit

    int tens = number/10%10;        //Getting tens digit from 4-digit

    int unit = number%10;           //Getting last digit from 4-digit

    int t= unit;

    int u= tens;

    int v= hundreds;

    int w= thousands;

//Converting the individual digits into corresponding number for passing it through the shift register so LEDs are turned ON or OFF in seven segment

switch (t)

{

  case 0:

  unit = 63;

  break;

  case 1:

  unit = 06;

  break;

  case 2:

  unit =91;

  break;

  case 3:

  unit=79;

  break;

  case 4:

  unit=102;

  break;

  case 5:

  unit = 109;

  break;

  case 6:

  unit =125;

  case 7:

  unit = 07;

  break;

  case 8:

  unit = 127;

  break;

  case 9:

  unit =103;

  break;

  }

switch (u)

{

  case 0:

  tens = 63;

  break;

  case 1:

  tens = 06;

  break;

  case 2:

  tens =91;

  break;

  case 3:

  tens=79;

  break;

  case 4:

  tens=102;

  break;

  case 5:

  tens= 109;

  break;

  case 6:

  tens =125;

  case 7:

  tens = 07;

  break;

  case 8:

  tens = 127;

  break;

  case 9:

  tens =103;

  break;

  }

  switch (v)

  {

  case 0:

  hundreds = 63;

  break;

  case 1:

  hundreds = 06;

  break;

  case 2:

  hundreds =91;

  break;

  case 3:

  hundreds=79;

  break;

  case 4:

  hundreds=102;

  break;

  case 5:

  hundreds = 109;

  break;

  case 6:

  hundreds =125;

  case 7:

  hundreds = 07;

  break;

  case 8:

  hundreds = 127;

  break;

  case 9:

  hundreds =103;

  break;

  }

  switch (w)

  {

  case 0:

  thousands = 63;

  break;

  case 1:

  thousands = 06;

  break;

  case 2:

  thousands =91;

  break;

  case 3:

  thousands=79;

  break;

  case 4:

  thousands=102;

  break;

  case 5:

  thousands = 109;

  break;

  case 6:

  thousands =125;

  case 7:

  thousands = 07;

  break;

  case 8:

  thousands= 127;

  break;

  case 9:

  thousands =103;

  break;

  }

    digitalWrite(9, LOW);

    digitalWrite(latchPin, LOW);

    shiftOut(dataPin, clockPin, MSBFIRST,thousands);  // The thousand digit is sent

    digitalWrite(latchPin, HIGH);  // Set latch pin HIGH to store the inputs

    digitalWrite(9, HIGH);         // Turinig on that thousands digit

    delay(5);                      // delay for multiplexing

    digitalWrite(10, LOW);

    digitalWrite(latchPin, LOW);

    shiftOut(dataPin, clockPin, MSBFIRST,hundreds );    // The hundered digit is sent

    digitalWrite(latchPin, HIGH);

    digitalWrite(10, HIGH);

    delay(5);

    digitalWrite(11, LOW);

    digitalWrite(latchPin, LOW);

    shiftOut(dataPin, clockPin, MSBFIRST,tens);   // The tens digit is sent

    digitalWrite(latchPin, HIGH);

    digitalWrite(11, HIGH);

    delay(5);

    digitalWrite(12, LOW);

    digitalWrite(latchPin, LOW);

    shiftOut(dataPin, clockPin, MSBFIRST,unit);   // The last digit is sent

    digitalWrite(latchPin, HIGH);

    digitalWrite(12, HIGH);

    delay(5);

}

**The END**