

Using 21st Century Technology to Preserve 20th Century Slave Clocks Without a Master Clock

Bob Pritzker

Bob Pritzker, past President of Toronto Chapter 33 addressed a unique way of driving slave clocks when there is no master clock available.

Slave clocks were ubiquitous in factories, schools, train station and many more locales in the late nineteenth and early twentieth centuries. They were all driven by a master clock which was maintained in a separate and controlled environment. While most of the slave clocks were utilitarian in appearance, many were housed in beautiful and ornate cases.

Today, collectors can find these clocks at antique shows and fairs, but how do you make them useful if you have no master to send pulses to the slave to mark the time? Bob's presentation addressed this issue thoroughly.

There are two general types of slave clock mechanisms... impulse driven and Synchronous. Bob distinguished the two and described how each worked. He then demonstrated how, using a tiny Arduino processor, he could simulate the pulsing of a master clock. As the two types of slave clocks operate differently Bob described the two pulse generators separately.

Bob provided the following handout identifying all the components required for each circuit, graphics showing the links between the parts, and the detailed instructions needed for the Arduino to provide the correct pulse.

//Arduino Blink Sketch

// the setup function runs once when you press reset
or power the board

void setup() {

```
// initialize digital pin 13 as an output.
pinMode(13, OUTPUT);
}
```

// the loop function runs over and over again forever

void loop() {

```
digitalWrite(13, HIGH); // turn the LED on (HIGH is
the voltage level)
delay(1000);           // wait for a second
digitalWrite(13, LOW); // turn the LED off by
making the voltage LOW
delay(1000);           // wait for a second
}
```

//Arduino Minute Impulse Generator

```
#include <DS3231.h>
```

```
#include <Wire.h>
```

```
#include <Arduino.h>
```

```
#define ONE_HZ_SW 10
```

```
int polar = 0;
```

```
int pulseCount = 0;
```

void setup() {

```
Wire.begin();
```

```
pinMode(ONE_HZ_SW, INPUT);
```

```
pinMode(8, INPUT); // set time switch
```

```
pinMode(6, OUTPUT);
```

```
digitalWrite(ONE_HZ_SW, HIGH);
```

```
digitalWrite(8, HIGH);
```

```
Wire.beginTransmission(0x68);
```

```
Wire.write(0x0E);
```

```
Wire.write(0x00);
```

```
Wire.endTransmission();
```

```
digitalWrite (6, LOW);
```

```
}
```

void loop() {

```
if(digitalRead(8)==LOW){
```

```
advClk();
```

```
}
```

```
if(digitalRead(8)==HIGH){
```

```
while(digitalRead(ONE_HZ_SW)) {
```

```
} // wait or LOW
```

```
{digitalWrite (6, LOW);
```

```
pulseCount++;}
```

```
while(!digitalRead(ONE_HZ_SW)) {
```

```
} // wait or HIGH
```

```
if(pulseCount == 60){
```

```
pulseCount = 0;
```

```
digitalWrite (6, HIGH);
```

```
}
```

```
}
```

```
void advClk() {
```

```
digitalWrite(6,HIGH);
```

```
delay (750);
```

```
digitalWrite (6,LOW);
```

```
delay (500); //
```

```
pulseCount = 0;
```

```
}
```

//Arduino Simplex Synchron Driver

```
#include <DS3231.h>
#include <Wire.h>
#include <Arduino.h>
#include <TM1637Display.h>
#define CLK 4
#define DIO 3
#define ONE_HZ_SW 10

TM1637Display display(CLK, DIO);
DS3231 Clock;
bool h12, PM;
int second,minute,hour,hrs;
uint8_t segto;

void setup() {

  Wire.begin();
  display.setBrightness(0x8);
  pinMode(ONE_HZ_SW, INPUT);
  pinMode(8, INPUT);// set hour switch
  pinMode(7, INPUT);

  digitalWrite(ONE_HZ_SW, HIGH);
  digitalWrite(8, HIGH);
  digitalWrite(7, HIGH);

  pinMode(13, OUTPUT);
  pinMode(5, OUTPUT);// pins 5 & 6 to relay
  pinMode(6, OUTPUT);

  Wire.beginTransmission(0X68);
  Wire.write(0x0E);
  Wire.write(0x00);
  Wire.endTransmission();

  digitalWrite (5, LOW);
  digitalWrite (6, HIGH);

  delay (500);
}

void loop()
{

  if (!digitalRead(8)) set_hour(); // hold the switch up to
  set hour
  if (!digitalRead(7)) set_minute(); // hold the switch up
  to set minute

  ReadDS3231();

  while(digitalRead(ONE_HZ_SW)) {
  } // wait or LOW
  digitalWrite(13, LOW);
  digitalWrite (5, LOW);
  digitalWrite (6, LOW);

  display.showNumberDec(hour*100+minute, true);
```

```
if((minute == 57) && (second == 54)) {
  digitalWrite (5,HIGH);//turn run motor on
}

if((hrs == 5) && (minute == 58) && (second == 8)) {
  digitalWrite (6, HIGH);//turn correction off
}

if((hrs != 5) && (minute == 58) && (second == 2)) {
  digitalWrite (6, HIGH);//turn correction off
}

while(!digitalRead(ONE_HZ_SW)) {
}
digitalWrite(13, HIGH);
segto = 0x80 | display.encodeDigit(hour%10);
display.setSegments(&segto, 1, 1); //add colon to
separate values
digitalWrite (5, LOW);
digitalWrite (6, LOW);
}

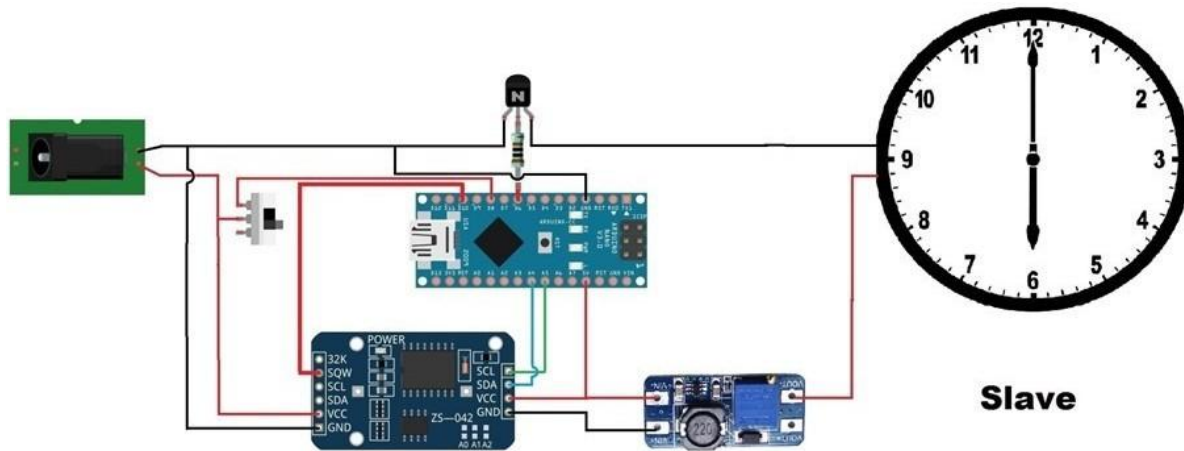
void ReadDS3231()
{

  second=Clock.getSecond();
  minute=Clock.getMinute();
  hour=Clock.getHour(h12, PM);
  hrs = hour;
  if(hour>12){
    hour=hour-12;
  }
}

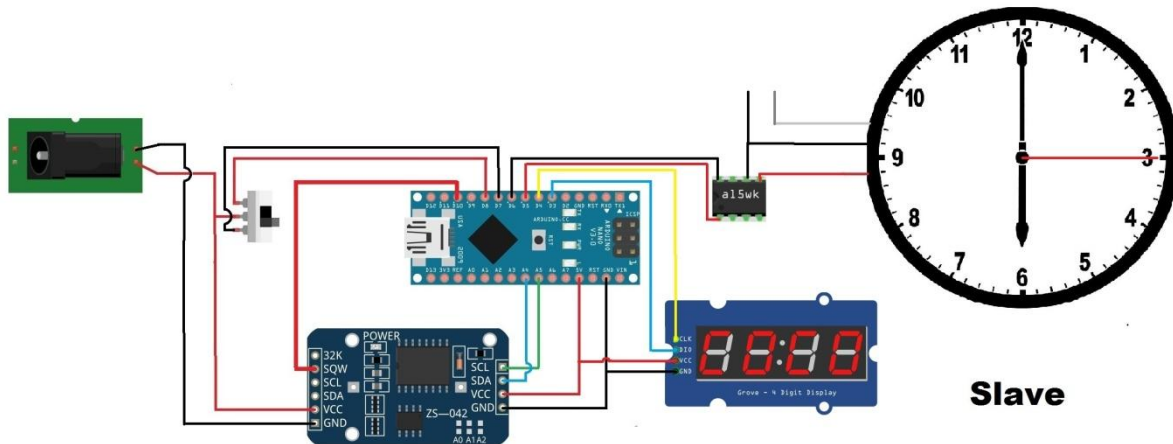
void set_hour()
{
  while (!digitalRead(8)) // set time switch must be
  released to exit
  {
    hrs++;
    if (hrs > 24) hrs = 1;
    display.showNumberDec(hrs*100+minute, true);
    delay(750);
  }
  Clock.setSecond(0); Clock.setHour(hrs);
  //delay(1000);
}

void set_minute()
{
  while (!digitalRead(7)) // set time switch must be
  released to exit
  {
    minute++;
    if (minute == 60) minute = 0;
    display.showNumberDec(hour*100+minute, true);
    delay(750);
  }
  Clock.setSecond(0); Clock.setMinute(minute);
}
}
```

Circuit to Drive Impulse Slave Clock



Circuit to Drive Synchronous Slave Clock



Sources of Information

Arduino: <http://www.arduino.cc>
<http://www.arduino.cc/en/Guide/HomePage>
<http://tronixstuff.com/2010/04/04/getting-started-with-arduino-chapter-zero/>

Real Time Clocks: <http://tronixstuff.com/2014/12/01/tutorial-using-ds1307-and-ds3231-real-time-clock-modules-with-arduino/>
<http://howtomechatronics.com/tutorials/arduino/arduino-ds3231-real-time-clock-tutorial/>

Slave Clocks: https://clockhistory.com/setclocks/secondary_clocks/
http://www.ats-usa.com/pdf_apnotes/MCMODES.pdf
http://www-03.ibm.com/ibm/history/exhibits/cc/pdf/cc_2407MLN2.pdf

Contact: Bob Pritzker timeman@live.com