Adventní kalendář s Arduino UNO

Adventní kalendář s Arduino UNO a stručný návod na stavbu chytrého adventního kalendáře s časovačem v podobě zimní vánoční vesničky řízený Arduinem.

Přináším druhou evoluci loňského „hloupého“ adventního kalendáře, který jen nečinně stál na skříňce a svítil pouze tehdy, když byl ručně sepnut obvod s několika LED zakoupený v KIKu. Pro letošní rok jsem tedy naplánoval razantní změnu a celkovou automatizaci postavenou na Arduino UNO, tímto je adventní kalendář mnohem více „fancy“ !! a vo to jde

Původní papírový kalendář

Před rokem, tedy advent 2019 jsem pomocí vyřezávacího plotru vyrobil adventní kalendář v podobě jednoduché zimní vesničky. Každá chaloupka má své číslo, velmi překvapivě, od jedné do dvaceti tří. Kostelík má výsadu honosit se dvacet čtverkou. Chaloupky jsou velké tak akorát na pytlík čaje, či kostičku ledové čokolády. Filozofie je jednoduchá, každý den si najdete chaloupku s překvapením (nakonec jsme se ustálili na různých druzích čaje). Kostelík je větší, tam se vešly i náušnice

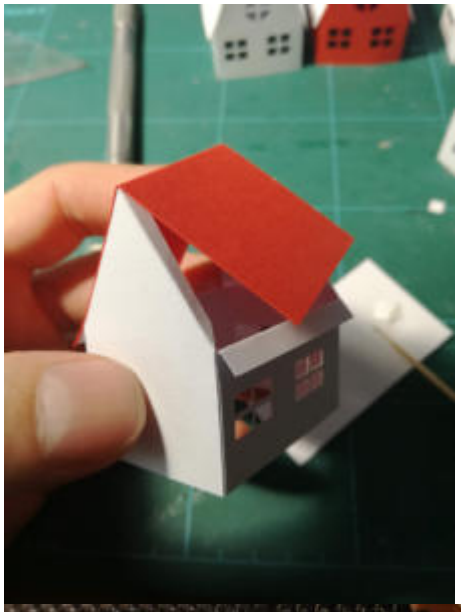
Slepení jednotlivých domečků byl víceméně relax, protože jsou opravdu jednoduché. Vyďobávání vyřezaných okýnek z lepivé řezací podložky už bylo horší Lepení domečků je snadné. Nejprve se slepí obvodové zdivo a potom postupně na každou stranu při lepí střechy. Lepidlem není třeba jakkoli plýtvat a promáčet jim papír. Stačí párátkem rozetřít pár kapek.

Zde je k dispozici zdrojová grafika v dxf pro chaloupky. Jsou tam tři tipy chaloupek, střecha pro každou z nich a kostel. Nemají na sobě čísla, to si musíte dle potřeby dokreslit a nakopírovat podle toho jak budete tisknout, nebo plotrovat.

>>DXF ke stažení zde<<

Výsledek „hloupého“ kalendáře vypadá takto.


EVOLVEO







Elektrifikace do každého omu!

Jelikož žijeme v 21. století, což dnes zarytě ignorují už jen úřady a Česká Pošta, je potřeba jít s dobou a kalendář trochu vyšperkovat. Zadání je jednoduché. Každý dům bude mít vlastní LED, které se budou rozsvěcet dle různých podmínek. Pro tento projekt jsem zvolil jako nejvhodnější adresovatelné LED, takže jsem dostal nekonečně mnoho řešení. Nakonec jsem se ustálil na čtyřech funkčních režimech:

1. Ráno 6:00 – 6:15 svítí celá vesnička bílou barvou a chaloupka s datem svítí modře (Červená se bila s barvou střech a špatně se hledala). Ten čas není náhodný, v 6h totiž vstávám do práce a nemusím po chodbě tápat podél zdi jako nebožáci ze sci-fi románu Den Trifidů
2. Dopoledne a přes den zhasnuto, svítí jen chaloupka s datem. Pro změnu červeně (mám k dispozici 255 barev, tak proč se nerozšoupnout).
3. Večer od 17:00 do 22:00 svítí opět celá vesnice jako ráno. Tedy chaloupky bílou, datum modrou.
4. A konečně poslední je režim nočního klidu. Vše je zhasnuto od 22:00 až do 6:00 ráno.

Program a hardware

Nápad je velmi jednoduchý, řešení už bylo horší. Jelikož jsem věčný programátorský začátečník, tak jsem musel zase načíst tuny internetů, ale nějak jsem to slepil ze všech možných příkladů + samozřejmě doprogramoval k obrazu svému.

Použité komponenty

- Arduino UNO (klon)
- RTC modul reálného času – Mini RTC Hodiny reálného času DS1307
- Adresovatelný LED pásek – Adresovatelný LED pásek 18W/m WS2812B RGB 1,66cm
- Vodiče – osvědčil se mi kroucený modelářský třípramenný servo kabel 0,24
- konektor pro 5,5 standardní arduino 5V konektor

- Adaptér do zásuvky s 5,5mm kolíkem na 5V

Program

Jak už bylo zmíněno, nějak jsem to slepil dohromady, ale nakonec to funguje. Bylo třeba si trochu pohrát s tím, že se kalendář sem tam odpojí ze zásuvky. Popíšu jen pár bodů které bych vyzdvihl, zbytek je okomentován v kódu.

První úskalí je v hardwaru. Můj LED pásek měl v datasheetu frekvenci 800KHz, která se nastavuje v parametrech knihovny. Nakonce po hodině laborování jsem zkusil dát 400KhHz ….. a voila problém vyřešen. Pokud tedy LEDky jen poblikávají a chovají se velmi „divně“ i s nahraným vzorovým demo koódem – problém bude ve frekvenci 400/800

Nastavení RTC je zakomentováno, aby se neprovádělo po každém připojení Arduina k napětí. Nastaví se tedy jednou, nahraje do desky a poté zakomentuje a zase nahraje do desky. Tím je RTC nastaven a pomocí interní baterky si drží čas.

Druhá zvláštnost oproti jednoduchému řešení je použití pole pro identifikaci rozsvícené LED. Důvod je jednoduchý. Domečky nejsou poskládány popořadě, ale data jsou rozházená různě po podložce. Abych nemusel tahat kilometry kabelu, spájel jsem co nejkratšího LED hada a propojil chaloupky nejkratší cestou. Tím pádem ID led diody už se nerovná číslu dne podle RTC. Zdroj data z hodin RTC tedy určuje pouze pořadí v poli LED ID.

příklad:

je 4.12. proměnná a[now.day()] se tedy rovná hodnotě s pořadovým číslem čtyři (počítá se od nuly) tedy hodnota a=15, do proměnné „den“ se tedy načte číslo „a“ tedy 15 a v programu potom rozsvítí patnáctou LED v pořadí nad kterou je chaloupka číslo 4

Je nutné si samozřejmě toto rozvrhnou dopředu, kde bude která chaloupka stát. Pokud chcete rozsvěcet LED postupně, tak odpadnou hrátky s polem a dáte že jednoduše „den=now.day() “ a tím pádem šestého se rozsvítí LED v pořadí šestá..

Zbytek už jsou jen příkazy pro ovládání LED pásku a podmínky pro časovač, tam už není co vysvětlovat.

```
// NeoPixel modul s 8 RGB LED WS2812
// navody.arduino-shop.cz - studnice vědění pro všechny Arduino nooby jako já
:-)
#include <Wire.h>
// připojení potřebné knihovny
#include <Adafruit_NeoPixel.h>
// nastavení propojovacího pinu
#define pinDIN 6
// nastavení počtu LED modulů
#define pocetLED 50
// inicializace LED modulu z knihovny
// pozor, některé čipy jedou na 400KHz, některé na 800KHz, já měl v
datasheetu 800 ale nakonec fungovaly až na 400 tak na to pozor
Adafruit_NeoPixel rgbWS = Adafruit_NeoPixel(pocetLED, pinDIN, NEO_GRB +
NEO_KHZ400);


#include "RTClib.h"
RTC_DS1307 RTC;

uint32_t barva2;
//
uint8_t R = 255;
uint8_t G = 255;
uint8_t B = 255;


uint32_t low = rgbWS.Color(0, 0, 0);
uint32_t high = rgbWS.Color(150, 150, 50);

int den;
int hodina;
int minuta;
int stav; //1 - rano, svitim; 2 - dopoledne nesvítím; 3 - odpoledne svítím; 4
- noc nesvítím
```

```
bool svitim = false;

// pole LED, domečky nejsou seřazeny popořadě dle data, ale rozsvící se na přezkáčku podle pozice na desce -> LED ID 1 = 6.12. ;LED ID 2 = 4.12. atd
int a[25] = {0, 6, 4, 20, 15, 19, 7, 1, 17, 13, 9, 8, 11, 5, 22, 21, 3, 10, 23, 2, 14, 16, 18, 12, 24};



#define NUM_LIGHTS 25




void setup() {

  // následující příkaz v pořadí rok, měsíc, den, hodina, minuta, vteřina
  //  RTC.adjust(DateTime(2020, 12, 16, 17, 48, 20));
  //delay(3000);

  // zahájení komunikace s LED modulem
  rgbWS.begin();

  rgbWS.show();            // Turn OFF all pixels ASAP
  rgbWS.setBrightness(255); // Set BRIGHTNESS to about 1/5 (max = 255)
  svitim = false; // zapamatování stavu aby program neustále nezapínal LED, led se zapnou jednou a pomocí "svitim" si zapamatuje že svítí a čeká na změnu času
  stav = 1; // stav časovače
  rgbWS.fill(0, 0, 24);
  rgbWS.show();
  DateTime now = RTC.now();
//----------------------po dosažení štědrého dne zůstane datum na 24.12. pro účely posledního příkazu setup, nutno nastavit i v Loop----------------------
----------------
  if (now.day() > 24) {
    den = 24;
  }
```

```
  else {
    den = a[ now.day()];
  }
//--------------------------------------------------------------------------
----------------------------
  Serial.begin(9600);
// zapnout všechny ledky postupně
  for ( int i = 0; i < NUM_LIGHTS; i++) {
    rgbWS.setPixelColor(i, high);
    rgbWS.show();
    delay(100);
  }
// vypnout všechny ledky
  for ( int i = 0; i < NUM_LIGHTS; i++) {
    rgbWS.setPixelColor(i, low);
    rgbWS.show();
    delay(100);
  }
// rozsvítit dané datum, pro případ že připojíme arduino ve stavu 2 -
dopoledne nesvítím -> program by šel do break a nenastavil datum
  nastavRGB(255, 0  , 0 , den);
  rgbWS.show();


}

void loop() {



  DateTime now = RTC.now();
//----------------------po dosažení štědrého dne zůstane datum na 24.12.------
--------------------------------
  if (now.day() > 24) {
    den = 24;
  }

  else {
    den = a[ now.day()];
```

```
  }
//--------------------------------------------------------------------------------
----------------------------------------
  hodina = now.hour();
  minuta = now.minute();
  Serial.print("Den: ");
  Serial.println(now.day());
  Serial.print("pozice: ");
  Serial.print(den);
  Serial.print(",");
  Serial.print(hodina);
  Serial.print(" : ");
  Serial.println(minuta);
  Serial.print("Stav: ");
  Serial.println(stav);
  Serial.print("Svitim: ");
  Serial.println(svitim);


  if (hodina == 6 && minuta < 15)  {
    stav = 1; //rano svítím
  }

  else if (hodina == 6 && minuta >= 15 || (hodina > 6 && hodina < 17)) {
    stav = 2; //dopoledne/den nesvítím
  }
  else if (hodina >= 17 && hodina < 22) {
    stav = 3;// odpoledne svítím
  }
  else {
    stav = 4; // noc nesvítím
  }



  switch (stav) {

    case 1: //rozsviť všechny LED + datum modře
      if (svitim == false) {
```

```
      for ( int i = 0; i < NUM_LIGHTS; i++) {
        rgbWS.setPixelColor(i, high);
        rgbWS.show();
        delay(400);
        nastavRGB(0, 0  , 255  , den);
        rgbWS.show();
        svitim = true;
      }

    }
    else {

    }
    break;

  case 2://zhasni všechny LED + rozsviť datum červeně

    if (svitim == true) {
      for ( int i = 0; i < NUM_LIGHTS; i++) {
        rgbWS.setPixelColor(i, low);
        rgbWS.show();
        delay(400);
        nastavRGB(255, 0  , 0  , den);

        rgbWS.show();

        svitim = false;
      }
    }
    break;

  case 3://rozsviť všechny LED + datum modře
    if (svitim == false) {

      for ( int i = 0; i < NUM_LIGHTS; i++) {
        rgbWS.setPixelColor(i, high);
        rgbWS.show();
        delay(400);
        nastavRGB(0, 0  , 255  , den);
```

```
          rgbWS.show();
          svitim = true;
        }

      }
      else {

      }
      break;

    case 4://zhasni všechny LED vč. data
      if (svitim == true) {

        for ( int i = 0; i < NUM_LIGHTS; i++) {
          rgbWS.setPixelColor(i, low);
          rgbWS.show();
          delay(400);
          //nastavRGB(0, 0  , 255  , den);
          rgbWS.show();
          svitim = false;
        }

      }
      else {

      }
      break;
  }
}


//--------------------------------------------------------------------------
---------

// funkce pro nastavení zadané barvy na zvolenou diodu
void nastavRGB (byte r, byte g, byte b, int cislo) {
  // vytvoření proměnné pro ukládání barev
  uint32_t barva;
  // načtení barvy do proměnné
```

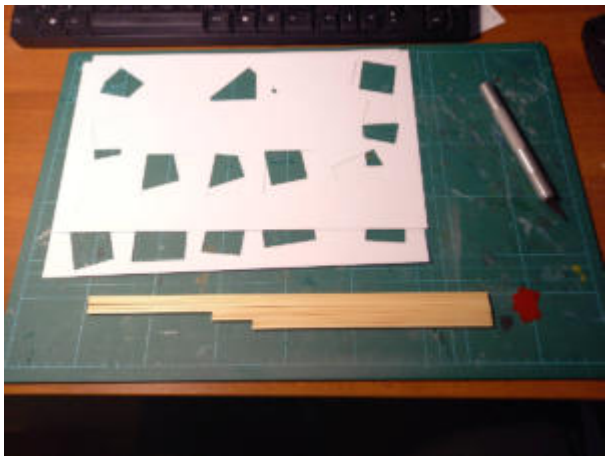
```
  barva = rgbWS.Color(r, g, b);
  // nastavení barvy pro danou LED diodu,
  // číslo má pořadí od nuly
  rgbWS.setPixelColor(cislo - 1, barva);
  // aktualizace barev na všech modulech
  rgbWS.show();
}
```

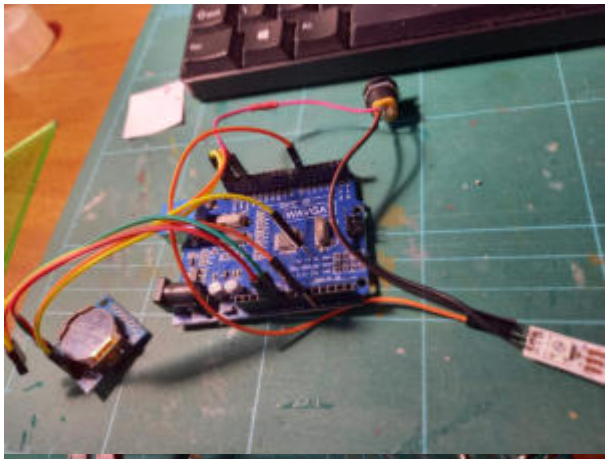
POKUS

Stavba

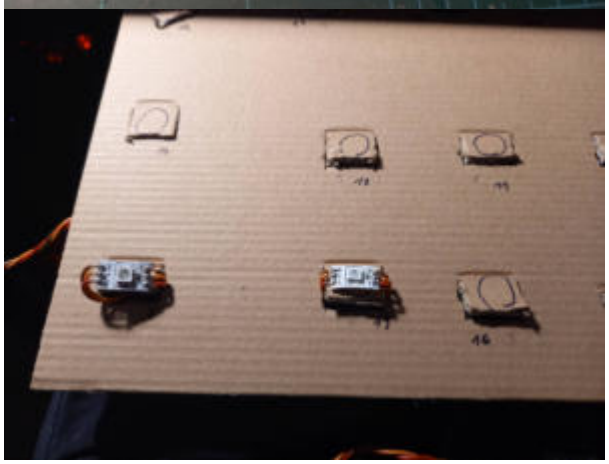
Neměl jsem po ruce moc kartonu, takže místo řešení „za pomoci trojúhelníku“ jsem zvolil kombinaci tvrdého výkresu a 3×3 smrkových nosníčků. Do škaredého kartonu jsem připevnil LEDky které jsem nastříhal a propojil mezi sebou do jednoho hada. A nahoru jsem pomocí oboustranné pásky přilepil dva „rámy“ ve kterých drží chaloupky. Šlo o to aby domečky byly do podložky zapuštěny a držely i když se bude s podložkou hýbat, nebo odněkud foukne.

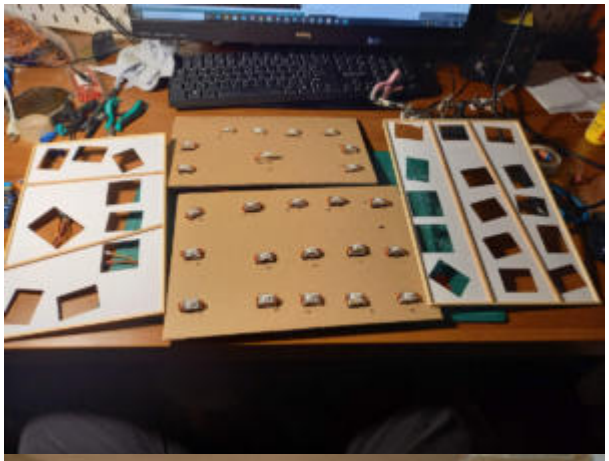
Podstavec jsem šmiknul z velké kartonové krabice. Zůstaly defacto boky a asi 1cm okraj, zbytek dna je vyříznutý. Zde platí, že fotky jsou více než tisíc slov.















Závěr

Blbinka dobrá, hlavně se člověk naučí pracovat s adresovatelnými LED a RTC a lze tak vytvořit jakékoli variace svíticích kalendářů, časomíry a tak podobně. Už se mi rodí v hlavě nápad na kalendář v3.0

Papírový adventní kalendář s Arduino UNO









Total Page Visits: 4751 - Today Page Visits: 0