Добрый день. Меня зовут Никита Сергеевич, мне 14 лет и в этом посте я имею желание рассказать вам о разработанною мною устройстве(и приложении), потому что готовое устройство спаял и собрал мой отец. Хотя возможно такое устройство вы уже видели не раз, но мне это не мешает.

Создавалось все это для автоматического включения и выключения света в курятнике, но применимо и для других целей.

Готовое и собранное устройство может подать на указанный цифровой пин arduino 1 или 0 в период от и до указанного времени в точности до минут(вне периода подается значение противоположное указанному в приложении при добавлении таймера ).

Все параметры указываются в приложении. А вот отсюда поподробнее...

Есть приложение для android в котором необходимо подключиться к блютус модулю и добавить таймер. Приложение интуитивно понятное и думаю проблем с ним не возникнет.

[](https://community.alexgyver.ru/attachments/screenshot_2019-02-21-20-48-13-png.2452/?hash=282e2c0cb4fe271102861a805b9dd528)

.apk файл будет в папке этого проекта.

Необходимые элементы:

1.arduino;

2.rtc module ds1307;

3.bluetooth module HC-05;

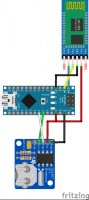
4.соединительные элементы(например провода);

5.прошивка для arduino; скетч на мой взгляд сложный и лучше в подробности не вдаваться.

6.приложение для android;

Разбираемся с arduino и подключаемыми модулями.

Ага, пункты 1-4:



arduino - hc-05

         5v - vcc

      gnd - gnd

      TX1 - RXD

      RX0 - TXD

arduino - ds1307

         5v - 5v

      gnd - gnd

        A5 - scl

        A4 - sda

 Не забываем при загрузке скетча отключать пин RX0 на arduino. Вы сказали "скетч"?

5.Скетч в студию:

[CODE=cpp]#include <Wire.h>                                            //библиотека для работы с шиной i2c

#include <iarduino\_RTC.h>                                    //библиотека для rtc модуля(если cчто, лежит в папке проекта(Timer->firmware\_for\_timer->iarduino\_RTC))

#include <EEPROM.h>                                          //библиотека для работы с энергонезависимой памятью

iarduino\_RTC time(RTC\_DS1307);                               //создание объекта для работы с модулем реального времени

int values;                                                  //переменная для получения данных

unsigned int to\_minutes(unsigned int hours, unsigned int minutes)//функция для перевода часов и минут в минуты

{

  return (60\*hours)+minutes;

}

void setup()

{

  time.begin();                                             //начало работы часов

  Serial.begin(9600);                                       //открытие сериал порта

}

void loop()

{

  if(Serial.available() > 0)                                //если что-то пришло

  {

    values = Serial.read();

    time.settime(0,-1, values);                             //считывание и установка часов

    delay(50);

    values = Serial.read();

    time.settime(0, values);                                //считывание и установка минут

    delay(50);

    //очищаем энергонезависимую память

    for (int i = 0 ; i < EEPROM.length() ; i++)

    {

      EEPROM.write(i, 0);

    }

    //////////////////////////////////

    values = Serial.read();

    delay(50);

    //запись данных в энергонезависимую память, полученных с телефона

    while (values > 0)                                        //пока есть что получать

    {

      for(int i = 0; i < 7; ++i)

      {

        EEPROM.write(EEPROM.read(EEPROM.length() - 1) \* 7 + i, Serial.read());

        delay(50);

      }

      EEPROM.write(EEPROM.length() - 1, EEPROM.read(EEPROM.length() - 1) + 1);

      values = Serial.read();

      delay(50);

    }

    /////////////////////////////////////////

  }

  //проход по каждому таймеру

  for(int i = 0; i < EEPROM.read(EEPROM.length() - 1); ++i)

  {

    if(EEPROM.read(i \* 7 + 0) == 1)                                                                               //если таймер нужно проверять

    {

      bool find = 0;                                                                                              //флаг

      time.gettime();

      if((to\_minutes(EEPROM.read(i \* 7 + 1), EEPROM.read(i \* 7 + 2)) <= to\_minutes(time.Hours, time.minutes)) && //если пришло время для работы

         (to\_minutes(EEPROM.read(i \* 7 + 3), EEPROM.read(i \* 7 + 4)) > to\_minutes(time.Hours, time.minutes)))

      {

        pinMode(EEPROM.read(i \* 7 + 5), OUTPUT);                                                                  //настраиваем пин таймера как выход

        digitalWrite(EEPROM.read(i \* 7 + 5), EEPROM.read(i \* 7 + 6));                                             //посылаем на пин нужное значение

      }

      else                                                                                                        //иначе если время не пришло

      {

        //ищем таймер с таким же пином

        for (int j = 0; j < EEPROM.read(EEPROM.length() - 1); ++j)

        {

          if(EEPROM.read(i \* 7 + 5) == EEPROM.read(j \* 7 + 5) && (i != j) && (EEPROM.read(j \* 7 + 0) == 1))

          {

            //и если нашли таймер с таким же пином и если для него время пришло

            if((to\_minutes(EEPROM.read(j \* 7 + 1), EEPROM.read(j \* 7 + 2)) <= to\_minutes(time.Hours, time.minutes)) &&

               (to\_minutes(EEPROM.read(j \* 7 + 3), EEPROM.read(j \* 7 + 4)) > to\_minutes(time.Hours, time.minutes))) //если пришло время для работы

            {

              pinMode(EEPROM.read(i \* 7 + 5), OUTPUT);                                                             //настраиваем пин таймера как выход

              digitalWrite(EEPROM.read(i \* 7 + 5), EEPROM.read(j \* 7 + 6));                                        //посылаем на пин нужное значение

              find = 1;                                                                                            //поднимаем флаг

            }

          }

        }

        if(find == 0)                                                                                              //если таймер с таким же пином не найден

        {

          pinMode(EEPROM.read(i \* 7 + 5), OUTPUT);                                                                 //настраиваем пин таймера как выход

          digitalWrite(EEPROM.read(i \* 7 + 5), !EEPROM.read(i \* 7 + 6));                                           //посылаем на пин значение противоположное нужному

        }

      }

    }

  }

}[/CODE]

Но все это без приложения будет холодным и неподвижным как камень, поэтому идем дальше.

6.

[QUOTE] Приложение интуитивно понятное и думаю проблем с ним не возникнет.  [/QUOTE]

Но лучше разобраться сразу. При первом входе в приложение необходимо будет подключиться к блютус модулю, для этого нажимаем "синхронизировать устройство"

и выбираем наш модуль. При галочке "подключаться автоматически" телефон сам будет пытаться подключиться к блютус модулю, при открытии приложения(эту фишку взял у alex gyver)."таймеры" здесь список всех таймеров и их поиск по имени. При нажатии на имя таймера открывается экран изменения этого таймера. Здесь же(где список таймеров) в правом нижнем углу расположен "+" нажав на который открывается экран добавления таймера, где указывается:

 -его имя,

- от какого времени(часы минуты),

- до какого времени(часы минуты),

- управляемый пин,

 -значение для него

-и состояние(будет arduino включать его или не будет в нужное время).

На этом хочу поблагодарить за внимание и сказать что, если нашли какие-то косяки или баги - смело пишите об этом, будем исправлять.