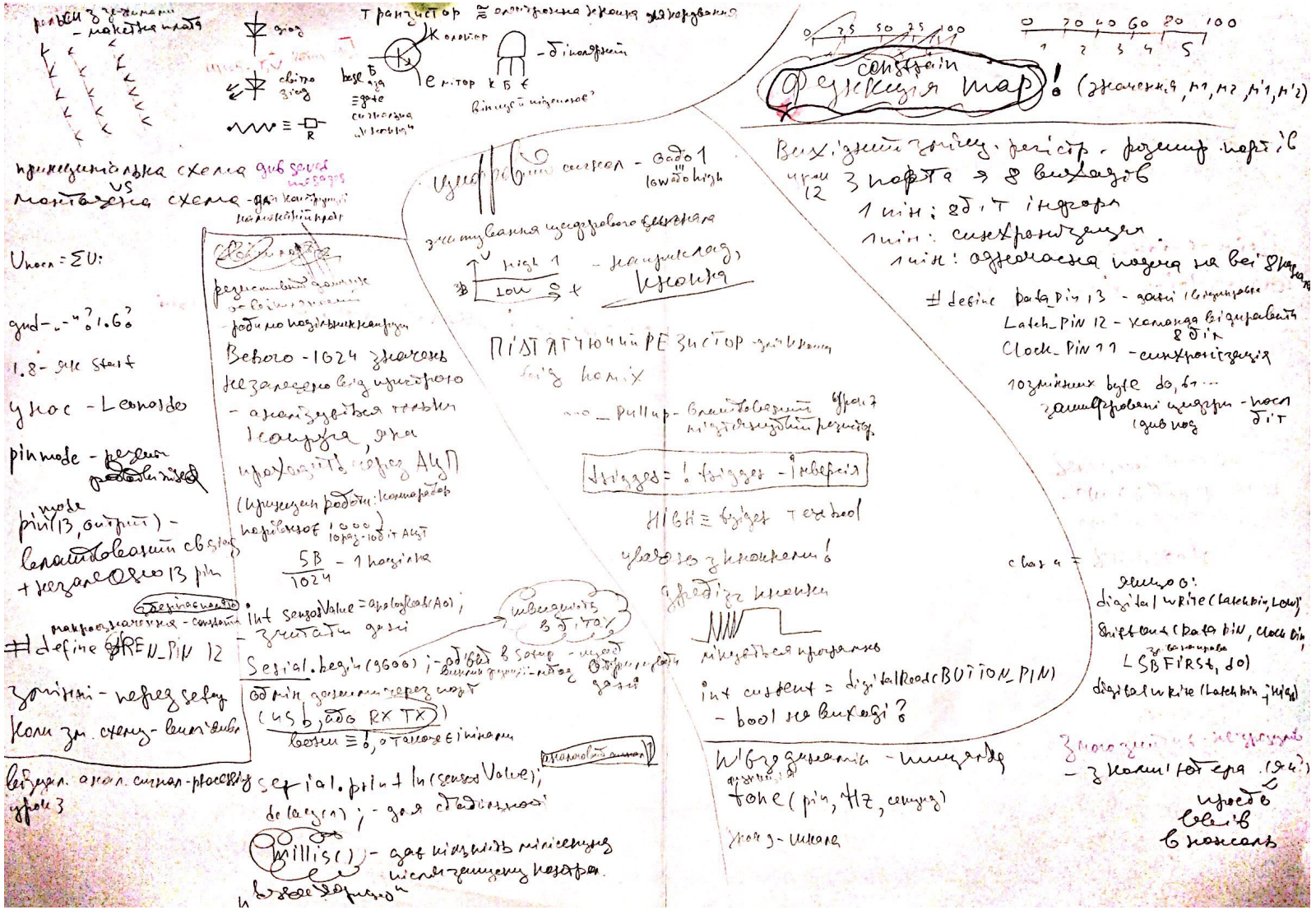


рельси з зажимами – макетна плата

діод

світло діод

транзистор ≅ електронна кнопка для керування

К колектор

Е мітор

– біполярний

К Б Е

вінцеві підсилює?

принципіальна схема див series

VS

монтажна схема – для конструкції на макетній платі

$U_{посл} = \Sigma U_i$

gnd – „–"? 1.6?

1.8 – як start

у нас – Leonardo

pinmode – режим роботи піна

pin mode (13, output) – влаштовувати св. діод + незалежно 13 pin

(define) макроозначення – constant

#define GREN_PIN 12

змінні – перед setup

коли зм. схему – вимкнути

візуал. анал. сигнал – processing урок 3

Цифровий сигнал – 0 або 1 low або high

зчитування цифрового сигнала

high 1

low 0

– наприклад, кнопка

ПІДТЯГУЮЧИЙ РЕЗИСТОР – для кнопки від поміх

... – pullup – влаштовувати урок 7 підтягнутий резистор

triggers = ! triggers – інверсія

HIGH ≡ triggers тех bool

уважно з кнопками!

дребізг кнопки

лікується програмно

int current = digitalRead(BUTTON_PIN) – bool на виході?

Вихідний зміщ. регістр – розширення портів

урок 12 3 порта → 8 виходів

1 пін: 8 біт інформ

1 пін: синхронізація

1 пін: одночасна подача на всі 8 каналів

#define Data_Pin 13 – дані (відкривання)

Latch_PIN 12 – команда відкривати 8 біт

Clock_PIN 11 – синхронізація

10 змінних byte d0, b1 ...

зашифровані цифри – посл біт (див под)

Вебого – 1024 значень

незалежно від пристрою – аналізується тільки напруга, яка проходить через АЦП

(принцип роботи: компаратор порівнює 1000 10 раз – 10 біт АЦП)

$\frac{5В}{1024}$ – 1 поділка

int sensorValue = analogRead(A0); – зчитати дані

Serial.begin(9600); – об'єкт в setup – щоб викликати функції – метод

обмін даними через порт (usb, або RX TX)

вони ≡ !, а також є і пінами

serial.print ln(sensorValue);

delay(1); – для стабільності

millis() – дає кількість мілісекунд після запуску контрол.

„взаємодіяльно"

швидкість в бітах

аналоговий сигнал

0 25 50 75 100

0 20 40 60 80 100

1 2 3 4 5

constrain

Функція map! (значення, м1, м2, м'1, м'2)

Вихідний зміщ. регістр – розширення портів

урок 12 3 порта → 8 виходів

1 пін: 8 біт інформ

1 пін: синхронізація

1 пін: одночасна подача на всі 8 каналів

#define Data_Pin 13 – дані (відкривання)

Latch_PIN 12 – команда відкривати 8 біт

Clock_PIN 11 – синхронізація

10 змінних byte d0, b1 ...

зашифровані цифри – посл біт (див под)

chasa =

якщо 0:

digitalWrite(Latch_Pin, Low);

shiftout(Data_Pin, Clock_Pin, LSBFIRST, d0)

digitalWrite(Latch_Pin, High)

Змагання – незрозумів – з комп'ютера (як?) просто ввів в консоль

Нібзадинамік – пищалка

функція

tone(pin, Hz, секунд)

Знак з – шкала

Відладка

Serial.println("get data")

5V - 800 mA

на Arduino

Записувати аналоговий сигнал

≡ Користуватись ШІМ, щоб „змінювати" на щось інше

напруга на виходах = 5 або 0 В

частота однаковa

0-255 значень

0-100%

ШІМ - імітація аналогового сигналу

analogWrite(pin, значення)

0-255

3~ - ШІМ

Масиви - згладжування від шумів

ширина вікна згладж

int d[6]

int d[ ]=[2,4,8]

int d[6]=[...]

char S[6]="hello"

програму див в SoftedMessages

Void - нічого не повертає

int d(int x) - функція, що повертає

return - отв

Serial.h:

3,10 ШІМ

не працює

done()

- шім не

перед -

constrain(9, 2, 400)

від межі зн. змінної

QuadDisplay.h

Arduino.cc

processing.org - візуалізація

потужний

шина

0 2

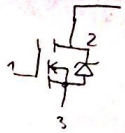
1 C

бібліотека є Wire.h

2 піна → багато пристроїв

WEEK 4

MOSFET транзистор



- посередник між Arduino і потужним навантаженням

тобто на 1 цифровий пін подається HIGH або LOW - як вимикач

digitalWRITE

можна використовувати ШІМ

~220В не можна

VIN - можна забирати всу потужність

ОБОВ: ВСІ ЗЕМЛІ МАЮТЬ БУТИ СПІЛЬНІ

(якщо різні джерела живлення)

ethernet shield

Датчик вологості грунту - через опір - аналоговий сигнал

Read serial - Basics

WEB - локальна мережа <SPI.h> - для інтерфейсу Shi

<ETHERNET.h>

ETHERNET → WEB SERVER

byte mac[ ]={...} - мак - адреса

16 систем

що написано на shield

отвіт:

IPAddress ip{192,168,0,150}; - 192.168.0.1 → розділ DHCP там є ooo

ooo відправка і отримання

Коллекторні мотори пост. струму L298D?

≥ 100 мА – не підходить ?

driver двигуна – зразу 2 двигуна

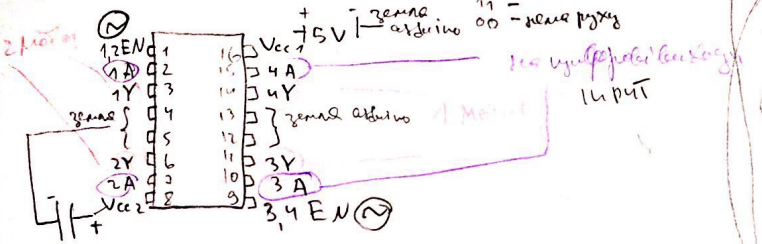
21.04.2015 керує швидкістю і напрямком

V_{CC2} – силове живлення

V_{CC1} – логічне живлення 5V аналогічно

EN – швидкість 2х двигунів – на ШІМ (analog) аналогічно

A – подається 0,1 => напрям руху 01 – право, 10 – ліво, 11 – , 00 – нема руху



5 EN } output
3 IN1 } цифрові
4 IN2 }

щоб запустити мотор треба намб 1 і 0 на IN1, IN2, це є задасть напрям

analog write EN – швидкість 0 – 255

motor shield L298 2 двигуна 2 IN

тільки 1 IN – цифровий фіксований HIGH – вперед, LOW – назад

! External Interrupts attachinterrupt (interrupt, ISR, mode)

N переривання, функція, подія (HIGH, LOW)

(0, 1, 2, 3, 4, 5)

Leonardo 3 2 0 1 7 0 – pins цифрові

delay заборонено

Volatile? ключове слово для змінної кількості обертів

Void drive (int left, int right) – функція керування швидкістю і напрямком двох колес

-255 ÷ 255

{

left = constrain(left, -255, 255);

right =

if (left >= 0) digitalWrite(IN2, HIGH) else low

Додаємо дальномір ультразв. – відстань в см

< порогове зн. – відїзд назад

релейний алгоритм – рух по лінії

Аналоговий дейтинг: фотоелемент інфракр

Залежить від висоти

білий – min

чорний – MAX

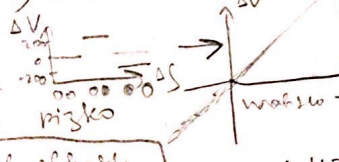
high – MAX (білий діагональ) 0 – 1000

lineSensor() – функція: чорне чи біле? порогове значення 700

2 датчика лінії

регулятор пропорційності > оптимально ΔV

плавне регулювання ΔV (це дозволить аналоговий сигнал датчика)



плавно – про

ΔS – помилка e

$u = k_1 e + k_2 e^3$ → 0,0001

pow(e,3)

– краще проходить всі типи поворотів

Bluetooth

таке це не по нав

HC-06

RX TX GROUND 5V

RX TX на аналогічно

receive transmit

Software serial – example

mySerial

Sep driver двигуна

Зам Кроковий двигун

Нап Біполярний двофазний

4.4V

Ш можна задавати скільки кроків

ан 200 кроків / аберт

Ма 3 pins → кількість кроків
→ напрям
→ режим утримування положення

FDM плата RAMPS

V G-code

5
3
4

Компас-3D

{ drive (int left, int

-255 ÷ 255

325

$u = 0{,}3e + 0{,}00001e^3$

Δu 312 456

72 760 -23

4