

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Харківський державний університет харчування та торгівлі

МІКРОПРОЦЕСОРНА ТЕХНІКА

Доповнення. Частина 3.

Методичні вказівки до виконання лабораторних робіт з дисципліни
«Мікропроцесорна техніка». Для студентів напряму підготовки
6.050604 «Енергомашинобудування»

Харків
ХДУХТ
2017

Методичні вказівки для виконання лабораторних робіт з курсу «Мікропроцесорна техніка» доповнення, частина 3, для студентів напряму підготовки 6.050604 «Енергомашинобудування» [Електронний ресурс] / укладачі: Дьяков О.Г., Іштван Є.О.– Електрон. дані. – Х. : ХДУХТ, 2017. –1 електрон. опт. Диск (CD-ROM); 12 см. – Назва з тит. екрана.

Укладачі: О.Г. Дьяков,
Є.О. Іштван.

Рецензент: д.т.н, проф. В.О. Потапов

Кафедра фізико-математичних та інженерно-технічних дисциплін

© Дьяков О.Г., Іштван Є.О., 2017
©Харківський державний університет
харчування та торгівлі, 2017

Вступ

Зростання рівня розвитку сучасної мікропроцесорної техніки, її широке використання у всіх галузях сучасної техніки призводить до не уклінного росту та удосконаленню технічних засобів що дозволяють суттєво спрощувати роботу людини з комп'ютером. Необхідність у прийомі, обробці та передачі великого обсягу інформації у реальному масштабі часу призводить до розширення сфери використання перетворювачів, які дають змогу вводити інформацію до комп'ютера так і створювати певні сигнали для керування технічними приладами.

Принциповою відмінністю в методах управління з використанням комп'ютера для керування технічними засобами у порівнянні з іншими засобами є те, що комп'ютер може обробляти тільки цифрову інформацію. В той же час людина при роботі використовує и інші види загального спілкування.

При використанні комп'ютера здійснюється цифрова обробка інформації, що надходить до нього. В той же час переважна більшість технічних приладів, якими передбачається керування, потребують використання аналогової інформації (тобто сигнали повинні повільно змінюватися у часі за своєю величиною та знаком).

Рішення всіх цих задач здійснюється лінійними цифро-аналоговими та аналого-цифровими перетворювачами (ЦАП та АЦП). Підвищення рівня автоматизації всіх галузей промисловості, у тому числі і харчової, неможливе без широкого використання цих приладів. Вони дають можливість створити необхідний діалог між людиною і приладами, які можуть керувати певними технологічними процесами.

Цифрова обробка сигналів є однією із галузей цифрового проектування, на якій базуються сучасні методи комплексної автоматизації як технологічних процесів, так і загальної автоматизації виробництва цілих галузей.

Знайомство з програмуванням мікроконтролерних систем на базі ATmega, з використанням сучасних мікросхем (MAX6675, ADG408, ADG409) та ЦАП з АЦП дає змогу напрацювати навички та розуміння застосування сучасних мікроконтролерних систем та електросхематичних рішень у харчовій промисловості.

Будь-які сучасні системи керування технологічними процесами використовують мікроконтролери та різноманітні датчики. Коли фахівець розуміє процес на різних його рівнях, то стає більш кваліфікованим, саме на навчання таких фахівців розраховане цей методичний посібник.

1. ВИМІРЮВАННЯ ТЕМПЕРАТУРИ З ВИКОРИСТАННЯМ ТЕРМОПАРИ ТА АВТОМАТИЗАЦІЇ НА БАЗІ КОНТРЕЛЕРА АТМЕГА

У лабораторній роботі використовується термопара хромель-алюмель (ТХА), Контролер Arduino UNO, операційний підсилювач та зовнішнє АЦП на базі мікросхеми MAX6675.

Термопари – це чутливий елемент термоелектричного перетворювача у вигляді двох ізольованих провідників із різнорідних матеріалів, з'єднаних на одному кінці, принцип дії якого ґрунтується на використанні термоелектричного ефекту для вимірювання температури. Принцип дії термопари базується на термоелектричних явищах. Термопара складається з двох провідників, сполучених кінцями так, що вони утворюють два контакти. Контакти поміщають в середовища з різною температурою.

Термопара ТХА або К-типу має широкі температурні режими роботи та високу чутливість. Стаціонарний режим для температури з 0 до 1100 градусів Цельсія та краткостроковий з –180 градусів до +1300 градусів.

Контролер Arduino UNO базується на мікроКонтролері ATmega328, та має 14 цифрових входів/виходів (з них 6 можуть використовуватись, як ШИМ-виходи), 6 аналогових входів, кварцевий резонатор на 16 МГц, інтерфейси: USB, COM, ICSP, I2C. Та має зручне підключення до комп'ютера за допомогою USB-стандарту.

Також використовуються плата з мікросхемою MAX6675, що дає змогу швидкого монтажу принципової схеми та дослідження особливостей використання програмних засобів.

UNO PINOUT

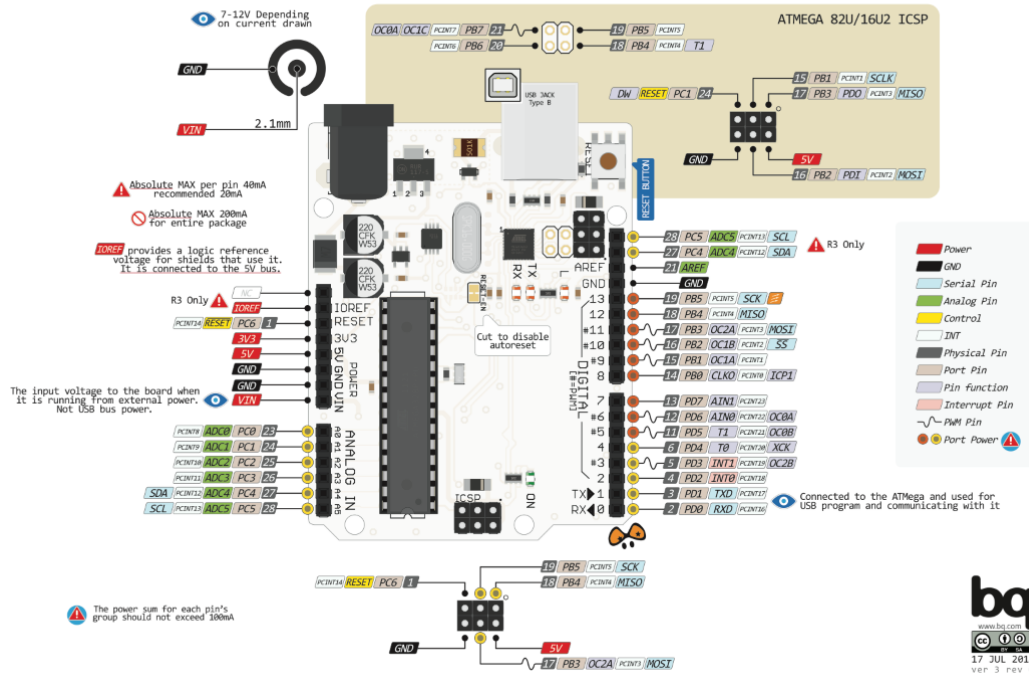


Рис. 1. Побудова контролеру Arduino UNO

Для програмування використовується середовище Arduino IDE, яке зображено на рисунку 2.

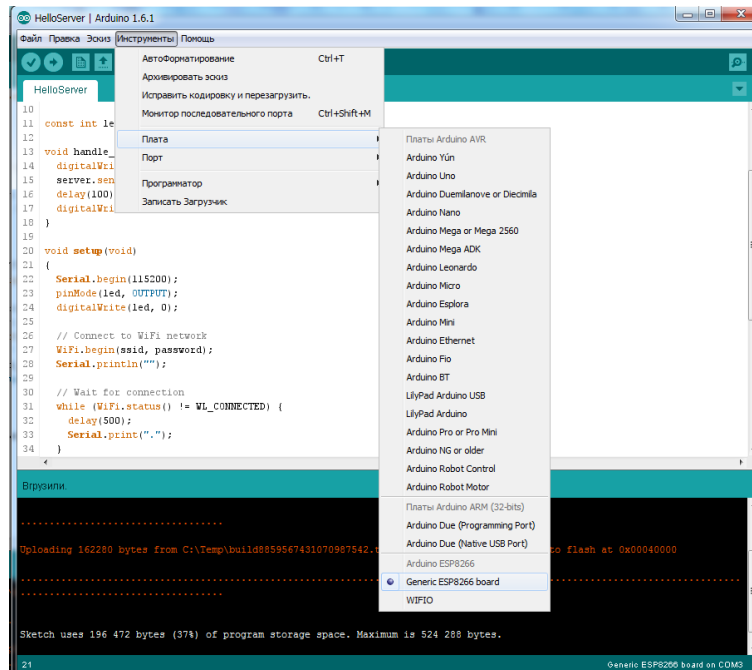


Рис. 2. Середовище програмування Arduino IDE

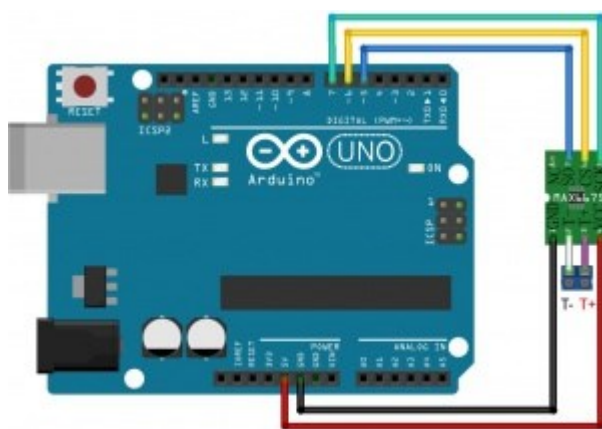


Рис. 3. Принципова схема підключення MAX6675 до Arduino UNO

Перед сборкою схеми слід уважно ознайомитися з маркуванням на всіх пристроях. Перед підключенням до джерела живлення слід ознайомитися зі схемою та переконатися, що вона зібрана вірно та збережена полярність підключення.

Слід переконатися, що до середовища Arduino IDE підключено бібліотеку для обміну даних із мікросхемою MAX6675.

t, °C	ЭДС, мВ										
	0	1	2	3	4	5	6	7	8	9	
-20	-1,27	-1,35	-1,39	-1,46	-1,52	-1,58	-1,64	-1,70	-1,77	-1,83	
-10	-0,64	-0,70	-0,77	-0,83	-0,89	-0,96	-1,02	-1,08	-1,14	-1,21	
-0	0	-0,06	-0,13	-0,19	0,26	-0,32	-0,38	0,45	-0,51	-0,58	
+0	0	0,07	0,13	0,20	0,26	0,33	0,39	0,46	0,52	0,59	
10	0,65	0,72	0,78	0,85	0,91	0,98	1,05	1,11	1,18	1,24	
20	1,31	1,38	1,44	1,51	1,57	1,64	1,70	1,77	1,84	1,91	
30	1,98	2,05	2,12	2,18	2,25	2,32	2,38	2,45	2,52	2,59	
40	2,66	2,73	2,80	2,87	2,94	3,00	3,07	3,14	3,21	3,28	
50	3,35	3,42	3,49	3,56	3,63	3,70	3,77	3,84	3,91	3,98	
60	4,05	4,12	4,19	4,26	4,33	4,41	4,48	4,55	4,62	4,69	
70	4,76	4,83	4,90	4,98	5,05	5,12	5,20	5,27	5,34	5,41	
80	5,48	5,56	5,63	5,70	5,78	5,85	5,92	5,99	6,07	6,14	
90	6,21	6,29	6,36	6,43	6,51	6,58	6,65	6,73	6,80	6,87	
100	6,95	7,03	7,10	7,17	7,25	7,32	7,40	7,47	7,54	7,62	
110	7,69	7,77	7,84	7,91	7,99	8,06	8,13	8,21	8,28	8,35	
120	8,43	8,50	8,58	8,65	8,73	8,80	8,88	8,95	9,03	9,10	
130	9,18	9,25	9,33	9,40	9,48	9,55	9,63	9,70	9,78	9,85	
140	9,93	10,00	10,08	10,16	10,23	10,31	10,38	10,46	10,54	10,61	
150	10,69	10,77	10,85	10,92	11,00	11,08	11,15	11,23	11,31	11,38	
160	11,46	11,54	11,62	11,69	11,77	11,85	11,93	12,00	12,08	12,16	
170	12,24	12,32	12,40	12,48	12,55	12,63	12,71	12,79	12,87	12,93	
180	13,03	13,11	13,19	13,27	13,36	13,44	13,52	13,60	13,68	13,76	
190	13,84	13,92	14,00	14,08	14,16	14,23	14,33	14,42	14,50	14,58	
200	14,66	14,74	14,82	14,90	14,98	15,06	15,14	15,22	15,30	15,38	

Рис. 4. Таблица градування ТХА

Завдяки Open Source підходу, всі креслення лінії платформ Ардуіно знаходяться у відкритому доступі, що призвело до створення великої кількості аналогів подібних платформ.

“UNO” перекладається як один з італійського і розробники тим самим натякають на майбутній вихід Arduino 1.0. Нова плата стала флагманом лінійки плат Ардуіно.

Характеристики

Мікроконтролер	ATmega328
Робоча напруга	5 В
Вхідна напруга (рекомендована)	7-12 В
Вхідна напруга (гранична)	6-20 В
Цифрові Входи/Виходи	14 (6 з ШІМ)
Аналогові виходи	6
Постійний струм вхід/вихід	40 мА
Постійний струм для 3.3 В	50 мА
Флеш-пам'ять	32 Кб
ОЗУ	2 Кб (ATmega328)
EEPROM	1 Кб (ATmega328)
Тактова частота	16 МГц

Живлення

Arduino Uno може отримувати харчування через підключення USB або від зовнішнього джерела живлення. Джерело живлення вибирається автоматично.

Зовнішнє живлення (не USB) може подаватися через перетворювач напруги AC/DC (блок живлення) або акумуляторною батареєю. Перетворювач напруги підключається за допомогою роз'єму 2.1 мм з центральним позитивним полюсом. Провід від батареї підключаються до висновків Gnd і Vin роз'єму живлення.

Платформа може працювати при зовнішньому живленні від 6 В до 20 В. При напрузі живлення нижче 7 В, висновок 5V може видавати менше 5 В, при цьому платформа може працювати нестабільно. При використанні напруги вище 12 В регулятор напруги може перегрітися і пошкодити плату. Рекомендований діапазон від 7 В до 12 В.

Піни живлення :

- VIN. Вхід використовується для подачі живлення від зовнішнього джерела (за відсутності 5 В від роз'єму USB або іншого регульованого джерела живлення). Подача напруги живлення відбувається через даний висновок.
- 5V. Регульоване джерело напруги, що використовується для живлення мікроконтролера і компонентів на платі. Харчування може подаватися від виведення VIN через регулятор напруги, або від роз'єму USB, або іншого регульованого джерела напруги 5 В.
- 3V3. Напруга на виводі 3.3 В генерується вбудованим регулятором на платі. Максимальне споживання струму складає – 50 мА.
- GND. Піни заземлення.

Пам'ять

Мікроконтролер ATmega328 має 32 кБ флеш-пам'яті, з яких 0.5 кБ використовується для зберігання завантажувача, а також 2 кБ ОЗУ (SRAM) і 1 Кб EEPROM (яка читається і записується за допомогою бібліотеки EEPROM).

Входи та виходи

Кожен з 14 цифрових висновків Uno може налаштований як вхід або вихід, використовуючи функції `pinMode ()`, `digitalWrite ()`, і `digitalRead ()`, Піни працюють при напрузі 5 В. Кожен висновок має навантажувальний резистор (за замовчуванням відключений) 20 – 50 кОм і може пропускати до 40 мА.

Деякі піни мають особливі функції:

- Послідовна шина: 0 (RX) і 1 (TX). Піни використовуються для отримання (RX) і передачі (TX) даних TTL. Дані піни підключені до відповідних висновків мікросхеми послідовної шини ATmega8U2 USB-to-TTL.
- Зовнішнє переривання: 2 і 3. Дані піни можуть бути налаштовані на виклик переривання або на молодшому значенні, або на передньому чи задньому фронті, або при зміні значення. Детальна інформація знаходиться в описі функції `attachInterrupt ()`.
- ШІМ: 3, 5, 6, 9, 10, і 11. Будь-який з висновків забезпечує ШІМ з роздільною здатністю 8 біт за допомогою функції `analogWrite ()`.
- SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). За допомогою даних висновків здійснюється зв'язок SPI, для чого використовується бібліотека SPI.
- LED: 13. Вбудований світлодіод, підключений до цифрового висновку 13. Якщо значення на виведення має високий потенціал, то світлодіод горить.

На платформі Uno встановлені 6 аналогових входів (позначених як A0..A5), кожен дозволом 10 біт (тобто може приймати 1 024 різних значення). Стандартно піни мають діапазон вимірювання до 5 В щодо землі, проте є можливість змінити верхню межу за допомогою виведення AREF і функції `analogReference ()`. Деякі піни мають додаткові функції:

- I2C: 4 (SDA) і 5 (SCL). За допомогою висновків здійснюється зв'язок I2C (TWI), для створення якої використовується бібліотека `Wire`.

Додаткова пара висновків платформи:

- AREF. Опорна напруга для аналогових входів. Використовується з функцією `analogReference ()`.

- Reset. Низький рівень сигналу на виводі перезавантажує мікроконтролер. Зазвичай застосовується для підключення кнопки

перезавантаження на платі розширення, що закриває доступ до кнопки на самій платі Arduino.

Зв'язок

На платформі Arduino Uno встановлено кілька пристроїв для здійснення зв'язку з комп'ютером, іншими пристроями Arduino або мікроконтролерами. ATmega328 підтримують послідовний інтерфейс UART TTL (5 В), здійснюваний висновками 0 (RX) і 1 (TX). Встановлена на платі мікросхема ATmega8U2 направляє даний інтерфейс через USB, програми на стороні комп'ютера «спілкуються» з платою через віртуальний COM порт. Прошивка ATmega8U2 використовує стандартні драйвера USB COM, ніяких сторонніх драйверів не потрібно, але на Windows для підключення потрібно файл ArduinoUNO.inf. Моніторинг послідовної шини (Serial Monitor) програми Arduino дозволяє посилати і отримувати текстові дані при підключенні до платформи. Світлодіоди RX і TX на платформі будуть мигати при передачі даних через мікросхему FTDI або USB підключення (але не при використанні послідовної передачі через шини 0 і 1).

Бібліотекою SoftwareSerial можливо створити послідовну передачу даних через будь-який з цифрових висновків Uno.

ATmega328 підтримує інтерфейси I2C (TWI) і SPI. В Arduino включена бібліотека Wire для зручності використання шини I2C.

Програмування

Платформа програмується за допомогою ПЗ Arduino. З меню Tools> Board вибирається "Arduino Uno" (згідно зі встановленим мікроконтролеру).

Мікроконтролер ATmega328 поставляється з записаним завантажувачем, що полегшує запис нових програм без використання зовнішніх програматорів. Зв'язок здійснюється оригінальним протоколом STK500.

Є можливість не використовувати завантажувач і запрограмувати мікроконтролер через піни ICSP (внутрішньосхемне програмування).

Автоматична (програмна) перезавантаження

Uno розроблена таким чином, щоб перед записом нового коду перезавантаження здійснювалася самою програмою Arduino на комп'ютері, а не натисканням кнопки на платформі. Одна з ліній DTR мікросхеми ATmega8U2, керуючих потоком даних (DTR), підключена до висновку перезавантаження мікроконтролеру ATmega328 через 100 нФ конденсатор. Активація даної лінії, тобто подача сигналу низького рівня, перезавантажує мікроконтролер. Програма Arduino, використовуючи дану функцію, завантажує код одним натисканням кнопки Upload в самому середовищі програмування. Подача сигналу низького рівня по лінії DTR скоординована з початком запису коду, що скорочує таймаут завантажувача.

Функція має ще одне застосування. Перезавантаження Uno відбувається кожного разу при підключенні до програми Arduino на комп'ютері з ОС Mac X або Linux (через USB). Наступні півсекунди після перезавантаження працює завантажувач. Під час програмування відбувається затримка декількох перших байтів коду щоб уникнути отримання платформою некоректних даних (всіх, окрім коду нової програми). Якщо проводиться разова налагодження скетчу, записаного в платформу, або введення будь-яких інших даних при першому запуску, необхідно переконатися, що програма на комп'ютері очікує протягом секунди перед передачею даних.

На Uno є можливість відключити лінію автоматичної перезавантаження розривом відповідної лінії. Контакти мікросхем з обох кінців лінії можуть бути з'єднані з метою відновлення. Лінія маркована "RESET-EN". Вимкнення автоматичного перезавантаження також можливо підключивши резистор 110 Ом між джерелом 5 В і даною лінією.

Струмівий захист роз'єму USB

В Arduino Uno вбудований самовостанавливаючийся запобіжник (автомат), що захищає порт USB комп'ютера від струмів короткого замикання і надструмів. Хоча практично всі комп'ютери мають подібний захист, проте, даний запобіжник забезпечує додатковий бар'єр. Запобіжник срабативат при проходженні струму більше 500 мА через USB порт і розмикає ланцюг до тих поки нормальні значення струмів НЕ будуть відновлено.

Фізичні характеристики

Довжина і ширина друкованої плати Uno складають 6.9 і 5.3 см відповідно. Роз'єм USB і силовий роз'єм виходять за межі даних розмірів. Чотири отвори в платі дозволяють закріпити її на поверхні. Відстань між цифровими висновками 7 і 8 дорівнює 0,4 см, хоча між іншими висновками воно становить 0,25 см.

Знайомство з цифровим і аналоговим Ардуіно

```
/* Миготіння LED
```

```
* -----
```

```
*
```

```
* Включає і вимикає світлодіод (LED) приєднаний
```

```
* До виходу 13, з інтервалом в 2 секунди
```

```
*
```

```
*/
```

```
int ledPin = 13; // LED приєднаний до висновку 13
```

```
void setup ()
```

```
{
```

```
  pinMode (ledPin, OUTPUT); // Встановлюємо висновок 13 як вихід
```

```
}
```

```

void loop ()
{
  digitalWrite (ledPin, HIGH); // Включаємо LED
  delay (1000); // Пауза 1 секунда (1000 мс)
  digitalWrite (ledPin, LOW); // Вимикаємо LED
  delay (1000); // Пауза 1 секунда (1000 мс)
}

```

Функція delay (n) призупиняє обробку програми на n мілісекунд. Все це відбувається у вічному циклі loop ().

Розглянемо зразок програми з регуляцією яскравості горіння світлодіода залежно від подання сигналу на аналоговий вхід. Для цього експерименту нам буде потрібно подавати на аналоговий вхід A0 змінну напругу, це можна реалізувати, підключивши на вхід дільник напруги або потенціометр.

```

const int analogInPin = A0; // На аналоговий вхід A0 підключаємо
потенціометр

```

```

const int analogOutPin = 9; // Пін 9 підключаємо до цифрового 9-му
порту з ШІМ

```

```

int sensorValue = 0; // Змінна для зчитування аналогового сигналу
int outputValue = 0; // Змінна задає сигнал на аналоговий вихід 9

```

```

void setup () {
  // Ініціалізація серійного порту на швидкості 96000 біт за секунду:
  Serial.begin (9600);
}

```

```

void loop () {
  // Зчитуємо значення аналогового сигналу в змінну:

```

```

    sensorValue = analogRead (analogInPin);
    // Масштабується функцією map вихідний сигнал, «стискаючи» його з
1024 до 256:
    outputValue = map (sensorValue, 0, +1023, 0, 255);
    // Виводимо отриманий сигнал на пін на analogOutPin:
    analogWrite (analogOutPin, outputValue);

    // Виводимо результати в серійний порт:
    Serial.print ("sensor =");
    Serial.print (sensorValue);
    Serial.print ("\ t output =");
    Serial.println (outputValue);

    // Чекаємо 2 мілісекунди перед наступним циклом для АЦП перед
НОВИМ зчитуванням:
    delay (2);
}

```

4. Підключення ендокера до ARDUINO UNO

Згадаймо підключення LED до Ардуіно, між включенням і вимиканням якого була секундна затримка. У використовуваній програмі (див. код нижче) був один дуже великий недолік, а саме була використана функція delay (). Під час виконання цієї функції контролер не може виконувати інші команди в головній функції loop ()

```

/* Миготіння LED
* -----
*
* Включає і вимикає світлодіод (LED) приєднаний
* До виходу 13, з інтервалом в 2 секунди
*
* /
int ledPin = 13; // LED приєднаний до висновку 13

```

```

void setup ()
{
pinMode (ledPin, OUTPUT); // Встановлюємо висновок 13 як вихід
}
void loop ()
{
digitalWrite (ledPin, HIGH); // Включаємо LED
delay (1000); // Пауза 1 секунда
digitalWrite (ledPin, LOW); // Вимикаємо LED
delay (1000); // Пауза 1 секунда
}

```

Перше, що ми зробимо, це внесемо невеликі коректування у функцію loop (). Щоб зробити код більш компактним і витонченим, замінимо 2 пари рядків на одну пару. Замість установки значення в HIGH, а потім назад в LOW, ми отримаємо поточне значення ledPin і проінвертуємо його. Тобто якщо воно було HIGH, то стане LOW і навпаки

```

void loop ()
{
digitalWrite (ledPin,! digitalRead (ledPin)); // Включаємо / вимикаємо LED
delay (1000); // Затримка 1 с
}

```

Тепер ми вдосконалюємо функцію delay (). Замість цього, ми будемо використовувати функцію millis (). Ця функція повертає кількість мілісекунд, що минув з моменту запуску поточної програми. Функція переповниться (повернеться в нуль) приблизно через 50 діб роботи програми.

Альтернативною функцією є micros (), яка повертає кількість мікросекунд, що минув з моменту запуску поточної програми. Функція переповниться (повернеться в нуль) приблизно через 70 хвилин роботи програми.

У нашому уроці ми будемо використовувати функцію millis ():

```

/*

```



```

** Енкодер
** Для керуванням яскравістю LED використовується енкодер Sparkfun
* /

int brightness = 120; // Яскравість LED, починаємо з половини
int fadeAmount = 10; // Крок зміни яскравості LED
unsigned long currentTime;
unsigned long loopTime;
const int pin_A = 12; // Pin 12
const int pin_B = 11; // Pin 11
unsigned char encoder_A;
unsigned char encoder_B;
unsigned char encoder_A_prev = 0;

void setup () {
  // Declare pin 9 to be an output:
  pinMode (9, OUTPUT); // Встановлюємо pin 9 як вихід
  pinMode (pin_A, INPUT);
  pinMode (pin_B, INPUT);
  currentTime = millis ();
  loopTime = currentTime;
}
void loop () {
  currentTime = millis ();
  if (currentTime >= (loopTime + 5)) { // перевіряємо кожні 5мс (200 Гц)
    encoder_A = digitalRead (pin_A); // Зчитуємо стан виходу А енкодера
    encoder_B = digitalRead (pin_B); // Зчитуємо стан виходу В енкодера
    if ((! encoder_A) && (encoder_A_prev)) { // якщо стан змінилося з позитивного до
нуля
      if (encoder_B) {
        // Вихід У в полож. упоряд., значить обертання за годинниковою стрілкою
        // Збільшуємо яскравість, не більше ніж до 255
        if (brightness + fadeAmount <= 255) brightness += fadeAmount;
      }
    }
  }
}

```

```

else {
    // Вихід У в 0 сост., Значить обертання проти годинникової стрілки
    // Зменшуємо яскравість, але не нижче 0
    if (brightness - fadeAmount >= 0) brightness -= fadeAmount;
}

}

encoder_A_prev = encoder_A; // Зберігаємо значення А для наступного циклу

analogWrite (9, brightness); // Встановлюємо яскравість на 9 ніжку

loopTime = currentTime;
}
}

```

У даному прикладі ми ввели дві додаткові змінні `currentTime` і `loopTime`. У функції `setup ()` обидві змінні мають одне і теж значення. У функції `loop ()`, змінна `currentTime` кожен раз оновлюється в циклі. Коли `currentTime` більше ніж `loopTime` на 1 секунду (`loopTime + 1000`), то LED змінює свій стан, а змінної `loopTime` присвоюється поточне значення `currentTime`.

Зверніть увагу, що в даному прикладі ми не використали функцію `delay ()` і процесор може виконувати інші операції.

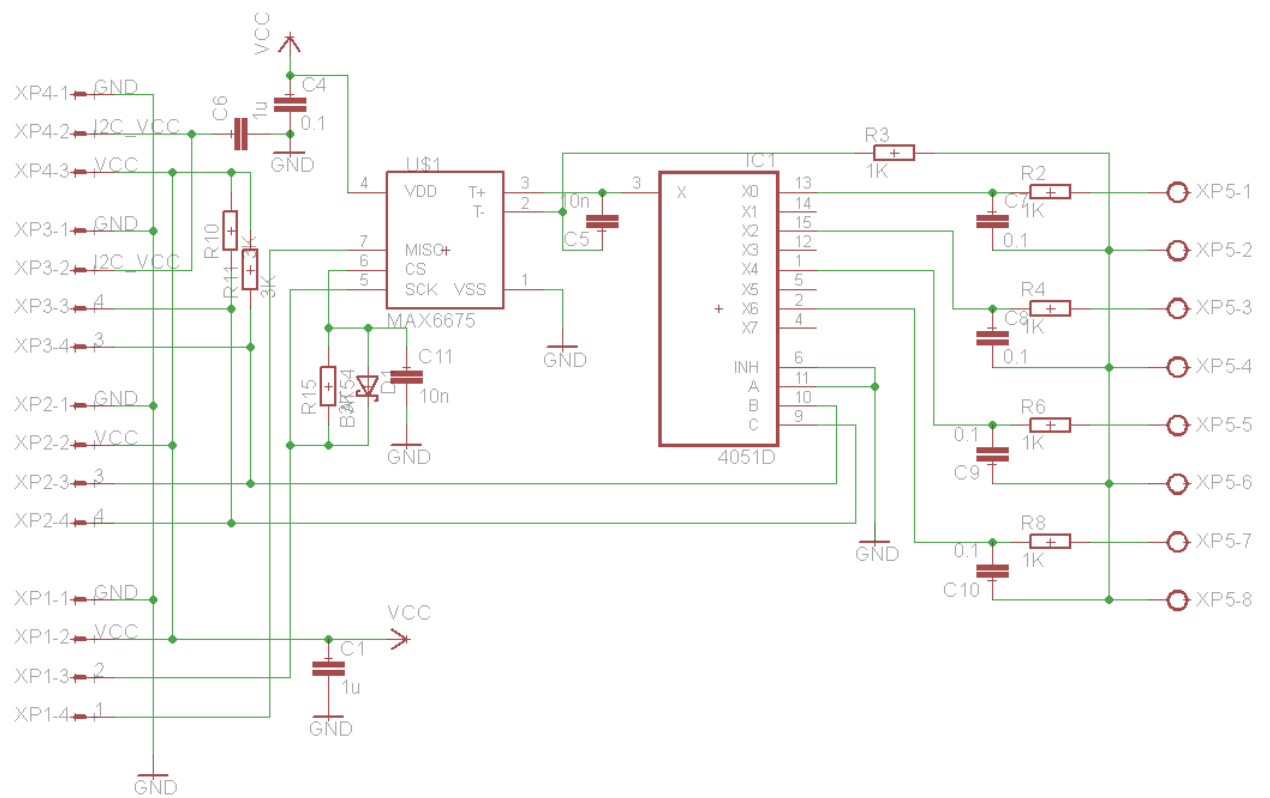


Рис.5. Принципова схема підключення термопар через ADG408 та MAX6675 до контролера.

Дослідження роботи Arduino UNO з мікросхемою MAX6675 та термопарю К-типу

1. Запустити програму Arduino IDE
2. Відкрити зразок з бібліотеки MAX6675 з обміном даних за Serial port.
3. Зібрати схему згідно з рисунком 3.
4. Завантажити програму у вигляді скетчу на контролер.
5. Відкрити Serial port та переконатись, що отримуєте дані з датчика, якщо данні не доходять, то перевірте швидкість обміну даними (9600 бот) та вірність підключення.
6. Зафіксувати зміну температури при доторканні до датчику.
7. За таблицею, зображеною на рисунку 4, знайти значення напруги, які відповідають отриманим значенням температури.
8. Зробіть висновок: в якому випадку зручно використовувати Контролер та мікросхему MAX6675?

Дослідження роботи Arduino UNO з мікросхемами ADG408 чи ADG409, MAX6675 та термопарами К-типу

1. Запустити програму Arduino IDE
2. Відкрити зразок з бібліотеки MAX6675 з обміном даних за Serial port.
3. Зібрати схему, згідно з рисунком 4. (використати 2 термопари К-типу)
4. Завантажити програму у вигляді скетчу на Контролер.
5. Перемичками встановити код 001 на піни А В С, що відповідає першій термопарі.

6. Відкрити Serial port та переконатись, що отримуєте данні з датчика, якщо данні не доходять, то перевірте швидкість обміну даними (9600 бот) та вірність підключення.

7. Змінити адресу на пінах А В С, на 002, що відповідає другій термопарі. Зафіксувати отримані значення

8. За таблицею, зображеною на рисунку 4, знайти значення напруги, які відповідають отриманим значенням температури

9. Зробіть висновки, в якому випадку доцільно використовувати мікросхеми ADG408 та ADG409?

СПИСОК ЛІТЕРАТУРИ

1. Манойлов В. В. Апаратні засоби систем автоматизації аналітичних приладів / В.В. Манойлов – СПб. : НДУ ІТМО, 2012. – 125 с.
2. Карлащук В. І. Електронна лабораторія на ІВМ РС. Інструментальні засоби і моделювання елементів практичних схем / В.І. Карлащук, С.В, Карлащук – М. СОЛОН-ПРЕСС, 2008. – 144 с.
3. Мілих В. І. Електротехніка, електроніка та мікропроцесорна техніка / В.І. Мілих, О.О. Шавьолкин – Київ : Каравелла, 2008. – 688 с.
4. Марков Б. Г. Моделирование электронных систем в программе «Electronics Workbench». – Волгоград: Перемена, 2006. – 78 с.
5. Будіщев М. С. Електротехніка, електроніка та мікропроцесорна техніка. – Львів : Афіша, 2001. – 423 с.
6. Онанченко Є. Л. Методичні вказівки до виконання лабораторних робіт «Дослідження роботи ЦАП і АЦП» з дисципліни «Електронні системи» / Є.Л. Онанченко, І.Є. Бражнік – Суми : Сумський державний університет, 2012. – 30 с.
7. Решетилов А. Р., Ольшевский Н. И. Методическое пособие к лабораторной работе по теме «Аналогово-цифровые преобразователи» / А. Р.Решетилов, Н. И. Ольшевский – Минск : Белорусский государственный университет информатики и радиоэлектроники, 2012. – 31 с.

ДОДАТКОВА

8. Федорков Б. Г. Микросхемы ЦАП, АЦП: функционирование, параметры, применение / Б.Г. Федорков, В.А. Телец. – М. : Энергоатомиздат, 1990. – 320 с.
9. Олсон Г. Цифровые системы автоматизации и управления / Г. Олсон, Д. Пиани – СПб. : Невский диалект. 2001. – 557 с.

Наукове електронне видання комбінованого використання
Можна використовувати в локальному та мережному режимах

МІКРОПРОЦЕСОРНА ТЕХНІКА.

Доповнення, частина 3.

Методичні вказівки до виконання лабораторних робіт для студентів
для студентів напряму підготовки 6.050604 «Енергомашинобудування»

Укладачі:

ДЬЯКОВ Олександр Георгійович

ШТВАН Єгор Олексійович

Відповідальний за випуск зав. кафедри М.І. Погожих
Техн. редактор А.О. Гончарова

План 2017 р., поз. ____/____

Підп. до друку 30. 12. 2017 р. Один електронний оптичний диск (CD-ROM);
супровідна документація. Об'єм даних 556 кб. Тираж 200 прим.

Видавець і виготівник

Харківський державний університет харчування та торгівлі
вул. Клочківська, 333, Харків, 61051.

Свідоцтво суб'єкта видавничої справи ДК № 4417 від 10.10.2012 р.