1.Аналіз предметної області

1.1Загальний аналіз плат *Arduino*

Стрімкий розвиток електроніки швидко змінює наше життя. Перше, що спадає на думку у цьому зв’язку, – це комп’ютери, інтернет і стільникові телефони. Людина вільна у пошуках необхідної інформації, має можливість вийти на зв'язок з бажаним абонентом, не зважаючи на місцезнаходження. Може отримувати дистанційну освіту і об’єднуватися у групи за професійними, соціальними або культурними інтересами. Все це стало можливим у значній мірі завдяки винаходу мікропроцесора і створенню мікропроцесорних систем.

Так, мікропроцесори і мікроконтролери широко використовуються у побутовій техніці, автомобільній електроніці, аерокосмічній та військовій галузях і, звичайно ж, у промисловому виробництві.

Отже, мікроконтролер – це спеціалізований мікроелектронний програмований прилад, що призначений для використання у керуючих пристроях, системах передачі даних та системах керування технологічними процесами.

Мікроконтролери використовують у побутовій техніці, медичних приладах, системах керування ліфтами, телефонах, раціях та інших засобах зв’язку, електронних музичних інструментах та автомагнітолах, комп’ютерній периферії (клавіатурах, джойстиках, принтерах, тощо), світлофорах, автоматичних воротах та шлагбаумах, інтерактивних дитячих іграшках, автомобілях, локомотивах та літаках, роботах та промислових верстатах.

Кожного року у світі продаються мільярди мікроконтролерів, а середній мешканець розвинутої країни протягом кожного дня десятки разів має справу з мікроконтролерами, які є невід’ємною частиною сучасного технологічного навколишнього середовища.

Освіта повинна взаємодіяти з використанням сучасних технологій. Людині непросто жити в сучасному світі. Їй потрібно постійно розвиватися і стежити за новітніми технологіями в електроніці, які відіграють дуже важливу роль в нашому суспільстві. Однією з таких новинок є сімейство контролерів *Arduino. Arduino* являє собою досить простий інструмент для створення електронних пристроїв і втілення в життя різних ідей. Це платформа побудована на друкованій платі з інтегрованим середовищем для написання програмного забезпечення. В основі апаратної частини лежить мікроконтролер сімейства *ATmega* і мінімально необхідна для роботи обв'язка.

*Arduino* може приймати цифрові і аналогові сигнали з різних пристроїв і має можливість керування різними виконуваними модулями. Існує велика кількість різних мікроконтролерів. Знайти потрібну і викладену в доступній формі інформацію про них буває досить важко, не кажучи вже про виконання будь-яких практичних завдань з їх використанням. *Arduino*, у свою чергу, спрощує процес роботи з мікроконтролерами і має ряд незаперечних переваг перед іншими пристроями для викладачів, студентів та любителів:

* низька вартість. Плати *Arduino* відносно дешеві в порівнянні з іншими платформами. Деякі готові модулі стоять менше 50 доларів. Найдешевшу версію можна зібрати вручну.
* кросплатформеність. З *Arduino* можна працювати на системах під управлінням *ОС Windows, Mac OS і Linux*.
* просте і зрозуміле середовище програмування. Середовище розробки спроектоване для новачків, не знайомих з розробкою програмного забезпечення. Однак це не заважає досвідченим користувачам створювати і досить складні проекти. Середовище являє собою додаток, що включає в себе редактор коду, компілятор і спеціальний модуль для прошивки плати. Мова програмування, що використовується в *Arduino*, є реалізацією *Wiring*. Говорячи простими словами, це *C / C ++*, доповнений деякими бібліотеками.
* можливість апаратного розширення. Можливості плат *Arduino* можна розширити за допомогою особливих мікросхем, які називають «шилдами» (від англ. *shields*). Шилди встановлюються поверх основної плати і дають нові можливості. Так, наприклад, існують плати-розширення для підключення до локальної мережі та інтернету (*Ethernet Shield*), для управління потужними моторами (*Motor Shield*), для отримання координат і часу з супутників *GPS* (модуль *GPS*) і багато інших.

У підсумку, навіть звичайні користувачі можуть розробити досвітчені зразки з метою економії коштів і розуміння роботи.

1.2Апаратна частина *Arduino*

*Arduino* - апаратна обчислювальна платформа, основними компонентами якої є плата вводу/виводу та середовище розробки на мові *Processing/Wiring*. Платформа *Arduino* застосовується для створення електронних пристроїв з можливістю прийому сигналів від різних цифрових і аналогових датчиків, які можуть бути підключені до неї, і управління різними пристроями*. Arduino* може використовуватися як для створення інтерактивних об'єктів автоматики, так і підключатися до програмного забезпечення на комп'ютері через стандартні дротові і бездротові інтерфейси (наприклад: *Adobe Flash, Processing, Max/MSP*, *Pure Data, SuperCollider*). Плата *Arduino* в основному складається з мікроконтролерів сімейства *Atmel AVR: ATmega328, ATmega168, ATmega2560*, *ATmega32U4, ATTiny85*, а також елементів для програмування та інтеграції з іншими пристроями. На багатьох платах наявний лінійний стабілізатор напруги +5В або +3,3В. Тактування здійснюється на частоті 16 або 8 МГц кварцовим резонатором. У мікроконтролер записаний завантажувач (*bootloader*), тому зовнішній програматор не потрібен. В рамках співпраці зі сторонніми виробниками в *Arduino IDE* була включена підтримка деяких апаратних засобів *Intel x86. Intel Galileo* та *Intel Edison – Arduino*-сумісні плати на *Intel x86* архітектурі. Плати механічно і електрично сумісні з периферійними платами *Arduino*. Плати функціонують під власною *ОС Linux*, поверх якої працює додаток, що дозволяє завантажувати і виконувати скетчі *Arduino*. Програми *Arduino* пишуться мовою програмування *C* або *C++.* Середовище розробки *Arduino* поставляється разом із бібліотекою програм, яка називається *"Wiring".* Користувачам необхідно визначити лише дві функції, для того щоб створити програму, яка буде працювати за принципом циклічного виконання: *setup():* функція виконується лише раз при старті програми і дозволяє задати початкові параметри; *loop():* функція виконується періодично доки плата не буде вимкнена

1.3Різновид плат *Arduino*

## 1.3.1.Arduino *Pro Mini*

*Arduino Pro Mini*(рис 1.1) - цей пристрій на базі мікроконтролера *ATmega328*. У його склад входить: 14 цифрових входів виходів (з них 6 можуть використовуватися в якості ШІМ-виходів), 8 аналогових входів, кварцовий резонатор, кнопка скидання і контактні площадки для впаювання роз'ємів. Шестиконтактний роз'єм може служити для живлення та взаємодії з платою через *USB* за допомогою *FTDI*-перехідника або макетної плати Sparkfun.

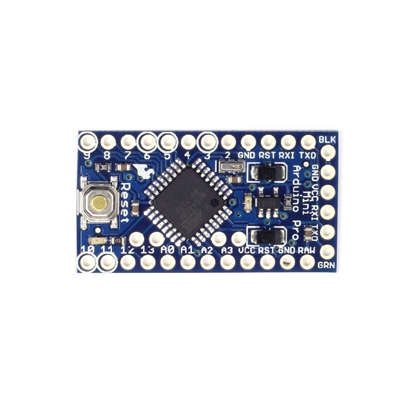


Рис.1.1 *Arduino Pro Mini*

Характеристики:

* Мікроконтролер *ATmega168* або *ATmega328*;
* Робоча напруга 3.3В або 5В (в залежності від моделі);
* Напруга харчування 3.35-12В (для моделі 3.3В) або 5 - 12В (для моделі 5В);
* Цифрові входи / виходи 14 (з них 6 можуть використовуватися в якості ШІМ-виходів);
* Аналогові входи 8;
* Максимальний струм одного виведення 40 мА;
* *Flash*-пам'ять 16 КБ (з яких 2 КБ використовуються загрузчиком);
* *SRAM* 1 КБ;
* *EEPROM* 512 байт;
* Тактова частота 8 МГц (для моделі 3.3В) або 16 МГц (в моделі 5В);

Живлення:

*Arduino Pro Mini* може живитися від різних джерел:

* через макетну плату;
* через перехідник *FTDI*, приєднаний до шестиконтактного роз'єму;

від стабілізованого джерела живлення з напругою 3.3В або 5В (в залежності від моделі), підключеного до виходу *Vcc*.

* Крім того, на платі є вбудований стабілізатор напруги, завдяки якому допускається подавати на плату напругу живлення величиною до 12В. Якщо для живлення плати використовується нестабілізоване джерело живлення, переконайтеся, що він приєднаний до виходу *"RAW",* а не *VCC.*

Нижче перераховані виходи живлення, розташовані на платі:

* + *RAW*. Для живлення плати від нестабілізованого джерела напруги.
  + *VСС*. Стабілізована напруга 3.3В або 5В.
  + *GND*. виходи землі.`

Пам’ять:

Обсяг флеш-пам'яті програм мікроконтролера *ATmega328* становить 32 КБ (з яких 2 КБ використовуються загрузчиком). Мікроконтролер також має 1 КБ пам'яті *SRAM* і 512 байт *EEPROM* (з якої можна зчитувати або записувати інформацію за допомогою бібліотеки *EEPROM*).

Входи і виходи:

З використанням функцій *pinMode ()*, *digitalWrite ()* і *digitalRead ()* кожен з 14 цифрових виходів *Pro Mini* може працювати в якості входу або виходу. Залежно від моделі, рівень напруги на висновках обмежений 3.3В або 5В. Максимальний струм, який може віддавати або споживати один вихід, становить 40 мА. ШІМ: виходи 3, 5, 6, 9, 10 і 11. За допомогою функції *analogWrite ()* можуть виводити 8-бітові аналогові значення в вигляді ШІМ-сигналу. Інтерфейс *SPI*: виходи 10 (*SS*), 11 (*MOSI*), 12 (*MISO*), 13 (*SCK*). Дані виходи дозволяють здійснювати зв'язок по інтерфейсу *SPI*. У пристрої реалізована апаратна підтримка *SPI*, проте на даний момент мова *Arduino* поки її не підтримує. В *Arduino Pro Mini* є 6 аналогових входів, кожен з яких може представити аналогову напругу у вигляді 10-бітного числа (1024 різних значення).

Зв'язок:

*Arduino Pro Mini* надає ряд можливостей для здійснення зв'язку з комп'ютером, ще одним Arduino або іншими мікроконтролерами. У *ATmega328* є приймач *UART*, що дозволяє здійснювати послідовний зв'язок за допомогою цифрових виходів 0 *(RX)* і 1 *(TX).* У пакет програмного забезпечення *Arduino* входить спеціальна програма, що дозволяє зчитувати і відправляти на *Arduino* прості текстові дані через *USB*-з'єднання.

Бібліотека *SoftwareSerial* дозволяє реалізувати послідовний зв'язок на будь-яких цифрових виходах *Arduino Pro Mini*.У мікроконтролері *ATmega328* також реалізована підтримка послідовних інтерфейсів *I2C (TWI)* і *SPI*. У програмне забезпечення *Arduino* входить бібліотека *Wire*, що дозволяє спростити роботу з шиною *I2C*

1.3.2.*Arduino Nano*

*Arduino Nano* (рис.1.2)- це повнофункціональний мініатюрний пристрій на базі мікроконтролера *ATmega328 (Arduino Nano 3.0)* або *ATmega168 (Arduino Nano* *2.x),* адаптоване для використання з макетної платі. За функціональністю пристрій схожий на Arduino Duemilanove, і відрізняється від нього розмірами, відсутністю роз'єму живлення, а також іншим типом (Mini-B) USB-кабелю. Arduino Nano розроблено і випускається фірмою Gravitech.

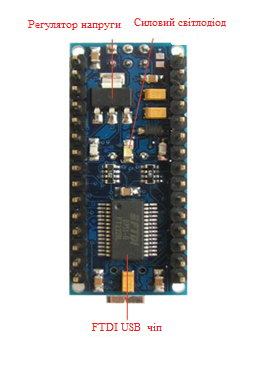
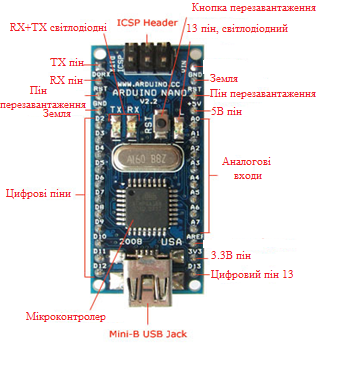


Рис. 1.2 *Назви портів Arduino Nano*

Характеристики:

* Мікроконтролер *Atmel ATmega168* або *ATmega328*
* Робоча напруга (логічний рівень) 5В
* Напруга живлення (рекомендований) 7-12В
* Напруга живлення (граничне) 6-20В
* Цифрові входи / виходи 14 (з яких 6 можуть використовуватися як ШІМ-виходи)
* Аналогові входи 8
* Максимальний струм одного виведення 40 мА
* Flash-пам'ять 16 КБ *(ATmega168)* або 32 КБ *(ATmega328)* з яких 2 КБ використовуються загрузчиком
* *SRAM* 1 КБ *(ATmega168)* або 2 КБ *(ATmega328)*
* *EEPROM* 512 байт *(ATmega168)* або 1 КБ *(ATmega328)*
* Тактова частота 16 МГц
* Розміри плати 1.85 см х 4.3 см

Живлення:

*Arduino Nano* може бути заживлений через кабель *Mini-B USB*, від зовнішнього джерела живлення з нестабілізованою напругою 6-20В (через вихід 30) або зі стабілізованою напругою 5В (через вихід 27). Пристрій автоматично вибирає джерело живлення з найбільшою напругою.

Напруга на мікросхему *FTDI FT232RL* подається тільки в разі живлення *Arduino Nano* через *USB*. Тому при харчуванні пристрою від інших зовнішніх джерел (НЕ *USB*), вихід 3.3В (формований мікросхемою *FTDI*) буде неактивний, в результаті чого світлодіоди *RX* і *TX* можуть мерехтіти при наявності високого рівня сигналу на виходах 0 і 1.

Пам'ять:

Обсяг пам'яті програм мікроконтролера *ATmega168* становить 16 КБ (з них 2 КБ використовуються загрузчиком); в *ATmega328* - цей обсяг становить 32 КБ (з яких 2 КБ також відведені під завантажувач). Крім цього, *ATmega168* має 1 КБ оперативної пам'яті *SRAM* і 512 байт *EEPROM* (для взаємодії з якої служить бібліотека *EEPROM*); а мікроконтролер *ATmega328* - 2 КБ *SRAM* і

1 КБ *EEPROM*.

Входи і виходи:

З використанням функцій *pinMode ()*, *digitalWrite ()* і *digitalRead ()* кожен з 14 цифрових виходів *Arduino Nano* може працювати в якості входу або виходу. Робоча напруга виходів - 5В. Максимальний струм, який може віддавати або споживати один вихід, становить 40 мА. Послідовний інтерфейс виходи 0 (*RX*) і 1 (*TX*). Використовуються для отримання (*RX*) і передачі (*TX*) даних по послідовному інтерфейсу. Ці виходи з'єднані з відповідними виходами мікросхеми-перетворювача *USB-UART* від *FTDI*.

Зовнішні переривання: виходи 2 і 3. Дані виходи можуть бути налаштовані в якості джерел переривань, що виникають при різних умовах: при низькому рівні сигналу, по фронту, по спаду або при зміні сигналу. ШІМ: виходи 3, 5, 6, 9, 10 і 11. За допомогою функції *analogWrite ()* можуть виводити 8-бітові аналогові значення в вигляді ШІМ-сигналу. Інтерфейс *SPI*: виходи 10 (*SS*), 11 (*MOSI*), 12 (*MISO*), 13 (*SCK*). Дані виходи дозволяють здійснювати зв'язок по інтерфейсу *SPI*. У пристрої реалізована апаратна підтримка *SPI*, проте на даний момент мова *Arduino* поки її не підтримує. Світлодіод: вихід 13. Вбудований світлодіод, приєднаний до цифрового виходу 13. При відправці значення *HIGH* світлодіод вмикається, при відправці *LOW* - вимикається.

Зв'язок:

*Arduino Nano* надає ряд можливостей для здійснення зв'язку з комп'ютером, ще одним *Arduino* або іншими микроконтроллерами. У *ATmega168* і *ATmega328* є приймач *UART*, що дозволяє здійснювати зв'язок з послідовним інтерфейсів за допомогою цифрових виходів 0 (*RX*) і 1 (*TX*). Мікросхема *FTDI FT232RL* забезпечує зв'язок приймача з *USB*-портом комп'ютера, і при підключенні до ПК дозволяє *Arduino* визначатися як віртуальний *COM*-порт (драйвера *FTDI* включені в пакет програмного забезпечення *Arduino*).

# 1.3.3. *Arduino Uno*

*Arduino Uno* (рис.1.3 )- це пристрій на основі мікроконтролера *ATmega328* (*datasheet*). У його склад входить все необхідне для зручної роботи з мікро контролером: 14 цифрових входів / виходів (з них 6 можуть використовуватися в якості ШІМ-виходів), 6 аналогових входів, кварцовий резонатор на 16 МГц, роз'єм *USB,* роз'єм живлення, роз'єм для внутрисхемного програмування (*ICSP*) і кнопка скидання. Для початку роботи з уcтройством досить просто подати живлення від *AC / DC*-адаптера або батарейки, або підключити його до комп'ютера за допомогою *USB*-кабелю.

На відміну від всіх попередніх плат *Arduino, Uno* в якості перетворювача інтерфейсів *USB-UART* використовує мікроконтролер *ATmega16U2* (*ATmega8U2* до версії *R2*) замість мікросхеми *FTDI.*

Зміни на платі версії R3 перераховані нижче:

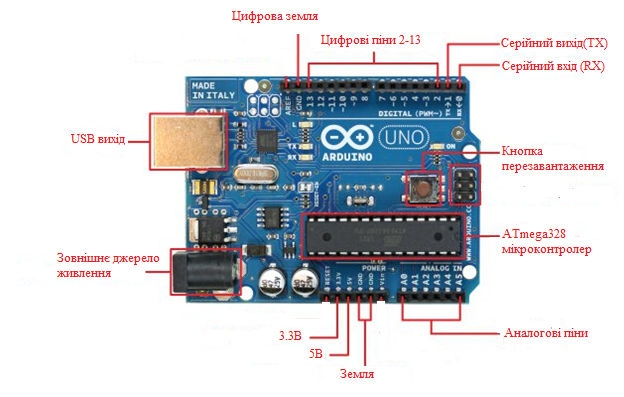


Рис. 1.3  *Назви портів Arduino Uno*

Терморегулятори 1.0: додані виходи *SDA* і *SCL* (біля виходу AREF), а також два нових виходи, розташованих біля виходів *RESET*. Перший - *IORE*F - дозволяє платам розширення підлаштовуватися під робочу напругу *Arduino*. Даний висновок передбачений для сумісності плат розширення як з 5В*- Arduino* на базі мікроконтролерів *AV*R, так і з 3.3В-платами *Arduino Due*. Другий вихід ні до чого не приєднаний і зарезервований для майбутніх цілей.Покращена стійкість ланцюга скидання.Мікроконтролер *ATmega8U2* замінений на *ATmega16U2."Uno"* (в перекладі з італійської - "один") названий з нагоди майбутнього випуску *Arduino 1.0*. Спільно з *Arduino 1.0* дані пристрої будуть базовими версіями *Arduino. Uno* - еталонна модель платформи *Arduino* і є останньою в серії *USB*-плат

Характеристики:

* мікроконтролер *ATmega328*
* Робоча напруга 5В
* Напруга живлення (рекомендований) 7-12В
* Напруга живлення (граничне) 6-20В
* Цифрові входи / виходи 14 (з них 6 можуть використовуватися в якості ШІМ-виходів)
* Аналогові входи 6
* Максимальний струм одного виведення 40 мА
* Максимальний вихідний струм виводу 3.3V 50 мА
* *Flash*-пам'ять 32 КБ (*ATmega*328) з яких 0.5 КБ використовуються загрузчиком
* *SRAM* 2 КБ (*ATmega328*)
* *EEPROM* 1 КБ (*ATmega328*)
* Тактова частота 16 МГц

Живлення:

*Arduino Uno* може живиться від *USB* або від зовнішнього джерела живлення - тип джерела вибирається автоматично.В якості зовнішнього джерела живлення *(НЕ USB)* може використовуватися мережевий *AC / DC*-адаптер або акумулятор. Штекер адаптера (діаметр - 2.1мм, центральний контакт - позитивний) необхідно вставити у відповідний роз'єм живлення на платі. У разі живлення від акумулятора, її проводу необхідно під'єднати до виходів *Gnd* і *Vin* роз'єму *POWER.*Напруга зовнішнього джерела живлення може бути в межах від 6 до 20 В. Однак, зменшення напруги живлення нижче 7В призводить до зменшення напруги на виводі 5V, що може стати причиною нестабільної роботи пристрою. Використання напруги більше 12В може призводити до перегріву стабілізатора напруги і виходу плати з ладу. З огляду на це, рекомендується використовувати джерело живлення з напругою в діапазоні від 7 до 12В.

Нижче вказані виходи живлення, розташовані на платі:

Живити пристрій через виходи 5V або 3V3 не рекомендується, оскільки в цьому випадку не використовується стабілізатор напруги, що може привести до виходу плати з ладу. 3.3В, що надходять від стабілізатора напруги на платі. Максимальний струм, споживаний від цього виходу, становить 50 мА.

Пам'ять:

Обсяг флеш-пам'яті *ATmega328* становить 32 КБ (з яких 0.5 КБ використовуються загрузчиком). Мікроконтролер також має 2 КБ пам'яті *SRAM* і 1 КБ *EEPROM* (з якої можна зчитувати або записувати інформацію за допомогою бібліотеки *EEPROM*).

Входи і виходи:

З використанням функцій *pinMode ()*, *digitalWrite ()* і *digitalRead ()* кожен з 14 цифрових висновків може працювати в якості входу або виходу. Рівень напруги на виходах обмежений 5В. Максимальний струм, який може віддавати або споживати один вихід, становить 40 мА. Послідовний інтерфейс: виходи 0 (*RX*) і 1 (*TX*). Використовуються для отримання (*RX*) і передачі (*TX*) даних по послідовному інтерфейсу. Ці виходи з'єднані з відповідними виходами мікросхеми *ATmega8U2*, яка виконує роль перетворювача *USB-UART*.Зовнішні переривання: виходи 2 і 3. ШІМ: виходи 3, 5, 6, 9, 10 і 11. За допомогою функції *analogWrite ()* можуть виводити 8-бітові аналогові значення в вигляді ШІМ-сигналу.Інтерфейс *SPI*: виходи 10 (*SS*), 11 (*MOSI*), 12 (*MISO*), 13 (*SCK*). Із застосуванням бібліотеки *SPI* дані виходи можуть здійснювати зв'язок по інтерфейсу *SPI*.Світлодіод: 13 вбудований світлодіод, приєднаний до виходу 13. При відправці значення *HIGH* світлодіод включається, при відправці *LOW* - вимикається. В *Arduino Uno* є 6 аналогових входів (A0 - A5), кожен з яких може представити аналогову напругу у вигляді 10-бітного числа (1024 різних значення). За замовчуванням, вимір напруги здійснюється щодо діапазону від 0 до 5 В. Проте, верхню межу цього діапазону можна змінити, використовуючи вихід *AREF* і функцію *analogReference ().* Крім цього, деякі з аналогових входів мають додаткові функції:*TWI*: виходи A4 або *SDA* і вихід A5 або *SCL*. З використанням бібліотеки *Wire* дані виходи можуть здійснювати зв'язок по інтерфейсу *TWI*.

Крім перерахованих на платі існує ще кілька виходів:

* *AREF*. Опорна напруга для аналогових входів. Може бути задіяний функцією *analogReference ().*
* *Reset*. Формування низького рівня *(LOW)* на цьому висновку призведе до перезавантаження мікроконтролера. Зазвичай цей висновок служить для функціонування кнопки скидання на платах розширення

Зв'язок

*Arduino Uno* надає ряд можливостей для здійснення зв'язку з комп'ютером, ще одним *Arduino* або іншими мікроконтроллерами. У *ATmega328* є приймач *UART*, що дозволяє здійснювати послідовний зв'язок за допомогою цифрових виходів 0 *(RX)* і 1 *(TX).* Мікроконтролер *ATmega16U2* на платі забезпечує зв'язок цього приймача з *USB*-портом комп'ютера, і при підключенні до ПК дозволяє *Arduino* визначатися як віртуальний *COM*-порт. Прошивка мікросхеми 16U2 використовує стандартні драйвера *USB-COM*, тому встановлення зовнішніх драйверів не потрібно. На платформі *Windows* необхідний тільки відповідний .*inf-файл*. У пакет програмного забезпечення *Arduino* входить спеціальна програма, що дозволяє зчитувати і відправляти на *Arduino* прості текстові дані. При передачі даних через мікросхему-перетворювач *USB-UART* під час *USB*-з'єднання з комп'ютером, на платі будуть мигати світлодіоди *RX* і *TX*. (При послідовній передачі даних за допомогою виходів 0 і 1, без використання *USB*-перетворювача, дані світлодіоди задіюються).Бібліотека *SoftwareSerial* дозволяє реалізувати послідовний зв'язок на будь-яких цифрових виходах *Arduino Uno*.У мікроконтролері *ATmega328* також реалізована підтримка послідовних інтерфейсів I2C (TWI) і SPI. У програмне забезпечення *Arduino* входить бібліотека *Wire*, що дозволяє спростити роботу з шиною I2C.

1.3.4 *Arduino Mega 2560*

*Arduino Mega 2560* (рис.1.4)- це пристрій на основі мікроконтролера *ATmega2560 (datasheet).* У його склад входить все необхідне для зручної роботи з мікро контролером: 54 цифрових входу / виходу (з яких 15 можуть використовуватися в якості ШІМ-виходів), 16 аналогових входів, 4 *UART* (апаратних приймача для реалізації послідовних інтерфейсів), кварцовий резонатор на 16 МГц , роз'єм *USB*, роз'єм живлення, роз'єм *ICSP* для внутрішньосхемного програмування і кнопка скидання. Для початку роботи з пристроєм досить просто подати живлення від *AC / DC*-адаптера або батарейки, або підключити його до комп'ютера за допомогою *USB*-кабелю. *Arduino* *Mega* сумісний з більшістю плат розширення, розроблених для *Arduino Duemilanove і Diecimila*.



Рис. 1.4 Платформа *Arduino Mega 2560*

*Arduino Mega 2560* відрізняється від усіх попередніх плат тим, що в ньому для перетворення інтерфейсів *USB-UART* замість мікросхеми *FTDI* використовується мікроконтролер *ATmega16U2* (*ATmega8U2* в версіях плати *R1* і *R2*).На платі *Mega 2560* версії *R2* доданий резистор, що підтягує до землі лінію *HWB* мікроконтролера *8U2*. Подібний захід дозволяє спростити процес оновлення прошивки і перехід пристрою в режим *DFU*.

Зміни на платі версії *R3* перераховані нижче:

* Терморегулятори 1.0: додані виходи *SDA* і *SCL* (біля виходів *AREF*), а також два нових виходи, розташованих біля виходу *RESET*. Перший - *IOREF* - дозволяє платам розширення підлаштовуватися під робочу напругу *Arduino*. Даний вихід передбачений для сумісності плат розширення як з 5V- *Arduino* на базі мікроконтролерів *AVR*, так і з 3.3V-платами *Arduino* *Due*. Другий вихід ні до чого не приєднаний і зарезервований для майбутніх цілей.
* Покращена стійкість ланцюга скидання.
* Мікроконтролер *ATmega16U2* замінений на *8U2*.

Характеристики:

* мікроконтролер *ATmega2560*
* Робоча напруга 5В
* Напруга живлення (рекомендована) 7-12В
* Напруга живлення (гранична) 6-20В
* Цифрові входи / виходи 54 (з яких 15 можуть використовуватися в якості ШІМ-виходів)
* Аналогові входи 16
* Максимальний струм одного вихода 40 мА
* Максимальний вихідний струм виходу 3.3V 50 мА
* *Flash*-пам'ять 256 КБ з яких 8 КБ використовуються загрузчиком
* *SRAM* 8 КБ
* *EEPROM* 4 КБ
* Тактова частота 16 МГц

Живлення:

*Arduino* *Mega* може живитися від *USB* або від зовнішнього джерела живлення - тип джерела вибирається автоматично.В якості зовнішнього джерела живлення (*НЕ* *USB*) може використовуватися мережевий *AC / DC*-адаптер або акумулятор / батарея. Штекер адаптера (діаметр - 2.1мм, центральний контакт - позитивний) необхідно вставити у відповідний роз'єм живлення на платі. У разі живлення від акумулятора / батареї, її проводу необхідно під'єднати до висновків *Gnd* і *Vin* роз'єму *POWER*.Напруга зовнішнього джерела живлення може бути в межах від 6 до 20 В. Однак, зменшення напруги живлення нижче 7В призводить до зменшення напруги на виводі 5V, що може стати причиною нестабільної роботи пристрою. Використання напруги більше 12В може призводити до перегріву стабілізатора напруги і виходу плати з ладу. З огляду на це, рекомендується використовувати джерело живлення з напругою в діапазоні від 7 до 12В.

Виходи живлення, розташовані на платі, перераховані нижче:

* *VIN*. Напруга, що надходить в *Arduino* безпосередньо від зовнішнього джерела живлення (не пов'язане з 5В від *USB* або іншим стабілізованою напругою). Через цей вихід можна як подавати зовнішнє живлення, так і споживати струм, коли пристрій живиться від зовнішнього адаптера.
* 5V. На цей вихід надходить напруга 5В від стабілізатора напруги на платі, поза незалежності від того, як живиться пристрій: від адаптера (7 - 12В), від *USB* (5В) або через вихід *VIN* (7 - 12В). Живити пристрій через виходи 5V або 3.3V не рекомендується, оскільки в цьому випадку не використовується стабілізатор напруги, що може привести до виходу плати з ладу.
* 3.3V що надходять від стабілізатора напруги на платі. Максимальний струм, споживаний від цього виходу, становить 50 мА.
* *GND*. Виходи землі.
* *IOREF*. Цей вихід надає платам розширення інформацію про робочу напругу мікроконтролера *Arduino*.

Пам'ять

У мікроконтролері *ATmega2560* є 256 КБ флеш-пам'яті програм (з яких 8 КБ використовуються загрузчиком), 8 КБ пам'яті *SRAM* і 4 КБ *EEPROM* (для роботи з цією пам'яттю служить бібліотека *EEPROM*).

Входи і виходи

З використанням функцій *pinMode (),* *digitalWrite ()* і *digitalRead ()* кожен з 54 цифрових виходів *Arduino* *Mega* можна налаштувати на роботу в якості входу або виходу. Рівень напруги на виходах обмежений 5В. Максимальний струм, який може віддавати або споживати один вихід, становить 40 мА.

Крім цього, деякі виходи *Arduino* можуть виконувати додаткові функції:

* Послідовний інтерфейс *Serial*: виходи 0 *(RX)* і 1 *(TX)*; *Serial* 1: 19 *(RX)* і 18 *(TX);* *Serial* 2: 17 *(RX)* і 16 *(TX); Serial* 3: 15 *(RX)* і 14 *(TX).* Дані виходи використовуються для отримання *(RX)* і передачі *(TX)* даних по послідовному інтерфейсу. Виходи 0 і 1 також з'єднані з відповідними виходами мікросхеми *ATmega16U2*, яка виконує роль перетворювача *USB-UART*.
* Зовнішні переривання: виходи 2 (переривання 0), 3 (переривання 1), 18 (переривання 5), 19 (переривання 4), 20 (переривання 3) і 21 (переривання 2). Ці виходи можуть використовуватися в якості джерел переривань, що виникають при різних умовах: при низькому рівні сигналу, при фронті, спаді або зміні сигналу. ШІМ: виходи 2 - 13 і 44 - 46. За допомогою функції *analogWrite ()* можуть виводити 8-бітові аналогові значення в вигляді ШІМ-сигналу.
* Інтерфейс *SPI*: виходи 50 *(MISO),* 51 *(MOSI),* 52 *(SCK),* 53 *(SS).* Із застосуванням бібліотеки *SPI* дані виходи дозволяють здійснювати зв'язок по інтерфейсу *SPI*. Лінії *SPI* також виведені на роз'єм *ICSP*, сумісний з *Arduino* *Uno*, *Duemilanove* і *Diecimila*.
* Світлодіод: 13. Вбудований світлодіод, приєднаний до виходу 13. При відправці значення *HIGH* світлодіод включається, при відправці *LOW* - вимикається.
* *TWI*: виходи 20 *(SDA)* і 21 *(SCL).* З використанням бібліотеки *Wire* дані виходи дозволяють здійснювати зв'язок по інтерфейсу *TWI*.
* В *Arduino Mega 2560* є 16 аналогових входів, кожен з яких може представити аналогову напругу у вигляді 10-бітного числа (1024 різних значення). За замовчуванням, вимір напруги здійснюється щодо діапазону від 0 до 5 В. Проте, верхню межу цього діапазону можна змінити, використовуючи висновок *AREF* і функцію *analogReference ().*

Крім перерахованих на платі існує ще кілька виходів:

* *AREF*. Опорна напруга для аналогових входів. Може бути задіяний функцією *analogReference ().*
* *Reset*. Формування низького рівня *(LOW)* на цьому виході призведе до перезавантаження мікроконтролера. Зазвичай цей вихід служить для функціонування кнопки скидання на платах розширення.

Зв'язок

*Arduino Mega 2560* надає ряд можливостей для здійснення зв'язку з комп'ютером, ще одним *Arduino* або іншими мікроконтроллерами. У *ATmega2560* є чотири апаратних приймача *UART* для реалізації послідовних інтерфейсів (з логічним рівнем *TTL* 5В). Мікроконтролер *ATmega16U2* (або *ATmega8U2* на платах версії *R1* і *R2*) забезпечує зв'язок одного з приймачів з *USB*-портом комп'ютера, і при підключенні до ПК дозволяє *Arduino* визначатися як віртуальний *COM*-порт (для цього операційній системі *Windows* потрібно відповідний *.inf-файл*, в відміну від *OSX* і *Linux*, де розпізнавання плати в якості *COM*-порту відбувається автоматично). У пакет програмного забезпечення *Arduino* входить спеціальна програма *SerialMonitor*, що дозволяє зчитувати і відправляти на *Arduino* прості текстові дані. При передачі даних через мікросхему *ATmega8U2* *ATmega16U2* під час *USB*-з'єднання з комп'ютером, на платі будуть мигати світлодіоди *RX* і *TX*. (При послідовній передачі даних за допомогою висновків 0 і 1, без використання *USB*-перетворювача, дані світлодіоди задіюються).

1.3.5. ***Raspberry Pi***

*Raspberry Pi* - це одноплатний комп'ютер, серцем якого є процесор архітектури ARM. Через роз'єм *HDMI* комп'ютер підключають до будь-якого сучасного телевізора. Клавіатуру і мишу можна підключити через *USB*. Існують два різновиди - *Model A* і *Model B*, які відрізняються обсягом оперативної пам'яті (256MB і 512MB), крім того у другій моделі є роз'єм *Ethernet*, через який його можна підключити до локальної мережі. Модель А можна пов'язати з зовнішнім світом тільки через спеціальний мережевий *USB*-адаптер.Замість жорсткого диска використовується *SD*-карта пам'яті.

У 2011 році вперше був сконструйований і представлений одноплатний мікрокомп'ютер *Raspberry Pi*. Призначався цей девайс для навчання інформатиці, розміром був всього лише з *USB*-флешку**. Він повинен був стати бюджетним інструментом, але виявився популярний і за межами шкільних кабінетів.** Популярності в інших сферах посприяли робототехніки та інженери, а також перемога на виставці *ARM TechCon* в тому ж 2011 році. І сьогодні мікрокомп'ютер *Raspberry Pi* продовжує активно розвиватися і використовуватися. ***Raspberry Pi*** зображено на рисунку 1.5.



Рис. 1.5 *Одноплатний комп'ютер* ***Raspberry Pi***

В принципі цей комп'ютер можна використовувати як ПК (персональний комп'ютер), але деякими застереженнями. Те, що він побудований на архітектурі *ARM*, не дозволяє встановити звичайні версії *Windows* і звичні програми для цієї системи. Можна встановити лише спеціальні версії *Linux*, які скомпільовані для цього комп'ютера. І оперативної пам'яті, ОЗУ (*RAM*), у нього замало за нинішніми мірками. Так, можна з нього виходити в Інтернет. Так що персональним комп'ютером його можна назвати з натяжкою. За своїми характеристиками це планшетний комп'ютер, тільки без екрану і акумулятора. ***Raspberry Pi*** може працювати під ***Windows CE***, ***Debian***, ***Fedora***, ***Gentoo***, ***Arch* *Linux***, ***RISC OS***, ***AROS*** або***FreeBSD***, навіть існує ***Android*** для ***Raspberry Pi***. Також розроблені ОС які базуються на  ***Debian*** *(****Raspbian****)* і  ***Fedora***(***FedoraRemix****,****Pidora***) оптимізовані під ***Raspberry Pi****.* Оскільки крім стандартних, притаманних комп’ютеру інтерфейсів, ***Raspberry Pi*** має “ноги” для підключення зовнішніх приладів, це розширює сферу застосування  ***Raspberry Pi***  у системах автоматизації та при побудові інших цікавих речей від розумного дому та роботизованих систем до автопілотів.

Характеристики:

* Живлення – +5В. через звичайний шнур з *microUSB*. Можна ввімкнути до *USB* комп’ютера, або використовувати зарядний пристрій для смартфона;
* *USB* – два порти, які зразу були зайняті клавіатурою і мишкою;
* *HDMI* – для підключення до монітора чи телевізора;
* *Video* – можна підключити до стандартного *Video* входу будь-якого пристрою;
* *Stereo Audio* – Звуковий вихід;
* Тримач *SD* картки. На *SD* картку встановлюється операційна система.
* *GPIO* – 26 ніг для приєднання до них будь-чого (сенсорів, мікросхем, мікроконтролерів, тощо). Підтримуються також *UART*, *I2C*, *SPI*.

1.3.6. ***Raspberry Pi* і *Arduino***

Безумовно, *Raspberry Pi* є повнофункціональнім комп'ютером,структура показана на рис.1.6.. Він володіє всіма атрибутами справжнього комп'ютера: віділенім процесором, пам'яттю і графічнім драйвером для виведення через *HDMI*. На *Raspberry Pi* працює спеціальна версія операційної системи Linux. Тому на Raspberry Pi легко встановити більшість програм для *Linux.* *Raspberry* *Pi* можна використовуват як повноцінній медіа-сервер або емулятор відеоігор.Хоча в *Raspberry Pi* і відсутнє внутрішнє сховище даних, але можна використовуват смарт-карти в якості флеш-пам'яті, яка Обслуговує всю систему. Таким чином, можна швидко завантажувати для налагодження різні версії операційної системи або програмних оновлень. Оскількі цей пристрій забезпечує незалежну з'єднуваність по мережі, його можна налаштовуваті и для доступу по *SSH*, або пересилати на нього файли по протоколу *FTP*.

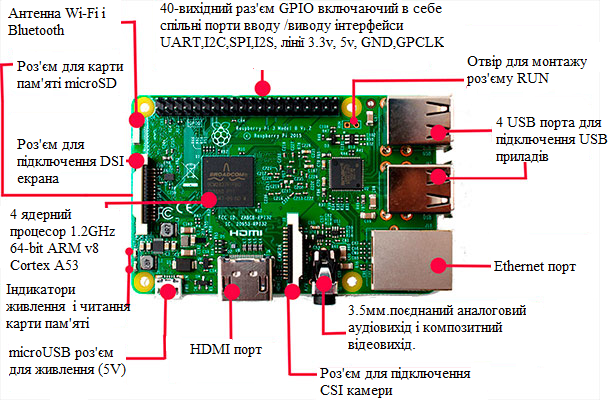


Рис. 1.6 ***Raspberry Pi* структура**

Плати *Arduino* - це мікроконтролери, а не повноцінні комп'ютери. На них немає операційної системи як такої, Arduino просто виконує код, що інтерпретується прошивкою. В даному випадку не має в розпорядженні базових інструментів, що надаються операційною системою, але, з іншого боку, таке безпосереднє виконання нескладного коду протікає простіше, а при роботі не виникає ніяких витрат, пов'язаних з операційною системою. Основне призначення плати *Arduino* - взаємодія з сенсорами і пристроями, тому *Arduino* відмінно підходить для апаратних проектів, де потрібно просто реагувати на різні сигнали сенсорів і ручне введення. Може здатися, що в цьому немає нічого особливого, проте на справі Вона відмінно підходить саме для підключення інших пристроїв і виконавчих механізмів, де операційної системи просто не потрібно, так як мова йде просто про реєстрацію дій і реагування на них.

Живлення:

Вимоги до електроживлення для цих двох систем дуже відрізняються. *Raspberry Pi* для роботи потрібно постійна напруга 5V, більш того, робота *Raspberry Pi* завершується програмним процесом - як у звичайного комп'ютера. *Arduino*, в свою чергу, починає виконувати код відразу після включення і припиняє роботу, коли ви виймаєте штепсель з розетки. Щоб розширити функціонал пристрою з *Arduino*, ви підключаєте пристрій контактом самої плати *Arduino* або до плат розширень для неї. Існують сотні різноманітних плат розширень, кожена з яких призначений для вирішення специфічного завдання, може взаємодіяти з тими чи іншими сенсорами, а також з іншими платами розширення, які разом утворюють повноцінний керуючий блок.*Raspberry Pi* складно переносити з місця на місце, так як ви не зможете просто вставити в нього дві батарейки . Для роботи цього комп'ютера необхідно забезпечити безперебійне живлення, а також підключити додаткове обладнання, яке гарантує подачу постійного струму. У випадку з *Arduino* весь процес дещо спрощується, так як система вимагає всього лише комплекту батарей, що забезпечує напругу не нижче певного рівня; також потрібен простий *Shild* для управління електроживленням. Навіть при аварійному відключенні струму на *Arduino* ви не ризикуєте ні пошкодити операційну систему, ні отримати будь-які програмні помилки. Варто заново підключити *Arduino* до джерела енергії і прилад просто відновить роботу.

У *Raspberry* *Pi* є вбудований *Ethernet*-порт, який забезпечує легкий доступ до будь-якої мережі і практично не вимагає настройки. Провести бездротовий Інтернет на *Raspberry* *Pi* також не складає труднощів: купуєте *USB*-адаптер для *WiFi* і встановлюєте відповідний драйвер. Як тільки це зроблено, можете використовувати операційну систему для підключення до веб-серверів, обробляти *HTML* або просто що-небудь писати в Інтернеті. Можна навіть використовувати *Raspberry* *Pi* для створення віртуальної приватної мережі або в якості сервера друку.

На жаль, система *Arduino* без додаткових модифікацій не пристосована для роботи по мережі. З нею потрібно як слід повозитися, щоб встановити надійне з'єднання, але це цілком можливо. Потрібна додаткова схема, оснащена *Ethernet*-портом, також знадобиться підключити деякі кабелі і написати потрібний код, щоб все запрацювало. Загалом, це досить складна робота, тому деякі компанії випускають платформи Arduino з уже вбудованим функціоналом Ethernet.

Сенсори:

Як *Raspberry* *Pi*, так і *Arduino* володіють набором інтерфейсних портів, проте аналогові сенсори набагато простіше підключати саме до *Arduino*. Мікроконтролер з легкістю інтерпретує різні сигнали на основі написаного вами коду і реагує на них. Тому *Arduino* відмінно підходить для тих випадків, коли ви плануєте виконувати серії команд або реагувати на показання сенсорів, в залежності від яких буде коригуватися робота сервоприводів і пристроїв.

У свою чергу, *Raspberry* *Pi* для ефективної взаємодії з такими пристроями вимагає спеціального програмного забезпечення ймовірно, це зайві складності, якщо потрібно просто автоматизувати поливання в саду. У багатьох проектах *Arduino* і *Raspberry* *Pi* використовуються спільно, причому *Arduino* виступає в якості керуючої плати, на якій виконуються команди, що видаються по *Raspberry* *Pi*. Інформація з сенсорів подається на *Raspberry* *Pi*, де вона записується, або у відповідь на неї виконуються ті чи інші операції.

Підсумки:

Отже, якщо ваша основна задача - зчитувати дані сенсорів, змінювати значення на двигуні або інших пристроях. З огляду на вимоги *Arduino* до електроживлення і простоту обслуговування цієї системи, пристрій цілком можна експлуатувати не вимикаючи, при цьому майже не втручаючись в його роботу.

При вирішенні таких завдань, які було б логічно виконувати на персональному комп'ютері. *Raspberry* *Pi* спрощує управління потоком операцій в різних ситуаціях: якщо підключаєтеся до Інтернету для зчитування або запису даних, програвайте якусь медіа-інформацію або підключаєтеся до зовнішнього дисплею.

З огляду на, те *Arduino* і *Raspberry* *Pi* вирішують різні завдання, в певних ситуаціях зручно використовувати ці пристрої спільно. Існує ряд можливостей з'єднання двох пристроїв; в такому випадку отримуєте клієнтський доступ до налаштувань і коду через *Raspberry* *Pi*, в той час як *Arduino* контролює управління робочими органами і збирає інформацію з сенсорів. Існує чимало варіантів такого з'єднання: по *USB*, локальної мережі або у вигляді простого підключення портів введення / виводу *Arduino* до *Raspberry* *Pi*.

Висновок

В останні роки використання *Arduino* експоненціально зростає. Протягом декількох років платформа використовувалася для створення багатьох тисяч проектів, починаючи від конструювання простих побутових девайсів до розробки складних наукових інструментів. Навколо платформи побудувалася співтовариство студентів, вчених і просто зацікавлених з усього світу. І вибирають саме цю платформу за зручність і багатофункціональність.

# 

# 2. ВИБІР ТЕХНОЛОГІЙ ТА ІНСТРУМЕНТАЛЬНИХ ЗАСОБІВ ДЛЯ РОЗРОБКИ ПРОГРАМНОГО МОДУЛЮ

## 2.1 Програмне забезпечення для *Arduino*

Для роботи з платформою Arduino вам не обов'язково ставитиме собі певний програмне забезпечення. Почати програмувати можна з   Arduino Web Editor, що дозволяє зберігати скетчі в хмарному сховищі. інструмент постійно оновлюється онлайн, нічого завантажувати і встановлювати заново не доводиться. Але для роботи, звичайно, буде потрібно постійне інтернет-з'єднання.

Однак, якщо ви віддаєте перевагу програмувати оффлайн, вам слід завантажити останню версію програми для робочого столу Arduino . Це відкрите програмне забезпечення, сумісне з наступними операційними системами:

* Windows;
* Mac OS (Lion або більш ранні версії);
* Linux 32 bit;
* Linux 64 bit;
* Linux ARM.

Актуальна версія додатка для робочого столу буде працювати з будь-якою версією мікрокомп'ютера Arduino, ніякі додаткові програми під певний залізо встановлювати не потрібно. Головна перевага всієї цієї відкритої платформи саме в її простоті. Якщо ви тільки починаєте працювати з Arduino, вас напевно порадує величезна кількість зрозумілих інструкцій, офіційно переведених навіть на російську мову.

2.1.1.Середовище розробки *ARDUINO WEB EDITOR*

Якщо чималий набір різних скетчів для *Arduino*, використовують середовище розробки , як *Arduino Web Editor*. Цей інструмент, виходячи з його назви, являє собою веб-редактор з досить зручною функціональністю. Він дає можливість розробляти свої проекти в єдиному місці (при роботі на різних ПК), збирати необхідні бібліотеки в єдине місце, публікувати розробки просто поділившись посиланням на скетч. Це дає найбільш швидкий і зручний доступ до стандартних бібліотек і прикладами з них. Швидке визначення порта з підключеним контролером. Недоліками можна назвати необхідність пам'ятати логін і пароль, можливі проблеми з роботою сторонніх бібліотек, особливо з сторонніми контролерами типа *ATtiny85*. Також роботу всередині бібліотек (створення, редагування). Напевно повністю відмовитись від свого IDE поки не вдасться тим, хто копається в бібліотеках. В цілому використовувати цей інструмент для якихось швидких невеликих проектів, публікації скетчів встаттях в інтернеті. Середовище розробки *ARDUINO WEB EDITOR* зображено на рис.2.1

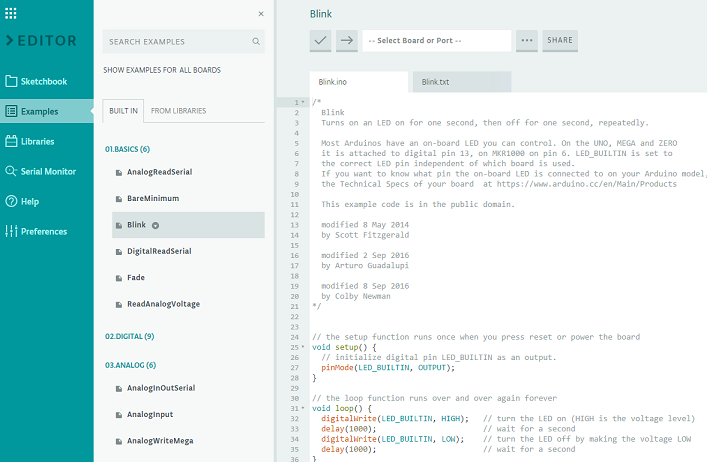


Рис.2.1 Середовище розробки *ARDUINO WEB EDITOR*

Закладка *Sketchbook* показує список своїх скетчів. У цієї системи буде обмеження на зберігання 100 скетчів. Також в цій закладці є кнопки створення нового скетча і папки. Тут же є можливість підвантажити свої проекти і бібліотеки з комп'ютера. Обмеження на зберігання даних - 100 МБ.Кнопка *Examples* переносить нас в список стандартних прикладів скетчів. Приклади майже не відрізняються від побачених нами в *Arduino IDE*. Також тут список прикладів з стандартних бібліотек.. Там вам будуть доступні дійсно шикарні приклади з ідеальних бібліотек. Такі як, наприклад, веб-сервер на *GPRS* Далі йде кнопка бібліотек *Libraries*. В ній бачимо список рідних бібліотек *Arduino*, а також маємо можливість в окремій вкладці побачити свої підвантажені бібліотеки або підвантажити при необхідності інші. Загалом тут передбачені всі зручності для пошуку рідних бібліотек, завантаження сторонніх і включення їх в свій скетч. Далі йдуть закладки *Serial* *Monitor*, допомога і налаштування вигляду веб-віконця.

2.1.2.Середовище розробки *ARDUINO IDE*

Середовище розробки *Arduino* представляє собою текстовий редактор програмного коду, область повідомлень, вікно виведення тексту (консоль), панель інструментів і кілька меню. Для завантаження програм і зв'язку середовище розробки підключається до апаратної частини *Arduino*.

Середовище розробки *ARDUINO IDE* зображено на рис.2.2

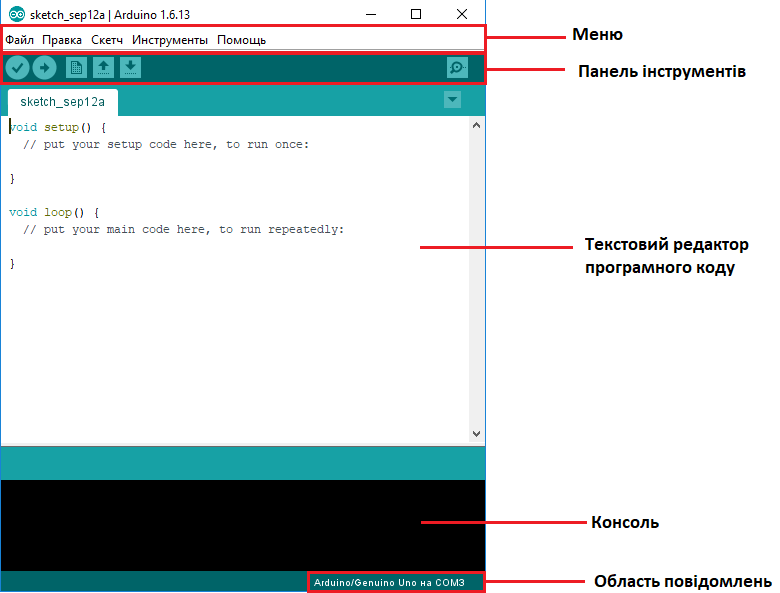


Рис.2.2 Середовище розробки *ARDUINO IDE*

Основний функціонал *Arduino IDE*.

Меню "Файл".(рис.2.3) Опустивши очевидні пункти меню, хотілося б згадати такий пункт, як "Папка зі скетчами". За замовчуванням *Arduino IDE* зберігає кожен скетч в окрему папку. Ім'я папки збігається з ім'ям, зазначеним для скетчу при збереженні. Змінити робочу директорію для папок з скетчами можна в пункті меню "Налаштування".

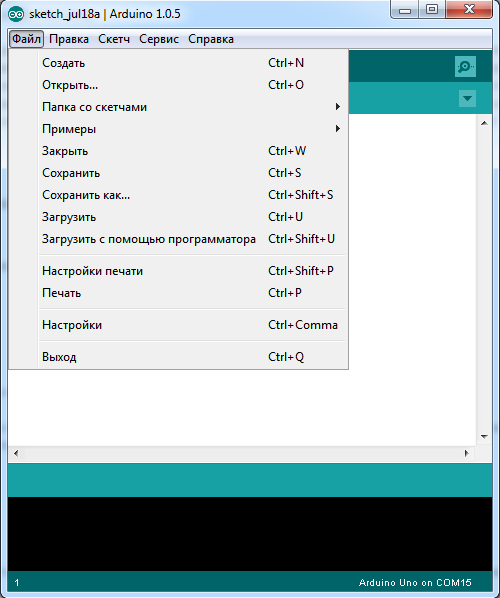


Рис2.3Меню файл

В меню "Правка"(рис.2.4) розташовані команди для роботи з кодом вашої програми. Часто використовувані команди зручні наявністю комбінацій для швидкого доступу за допомогою клавіатури. Зручними функціями є можливість копіювання для форумів і в *html* форматі, що дозволяють ділиться вашими скетчами, зберігаючи наочність розмітки у вигляді *BB* кодів або *html* розмітки відповідно

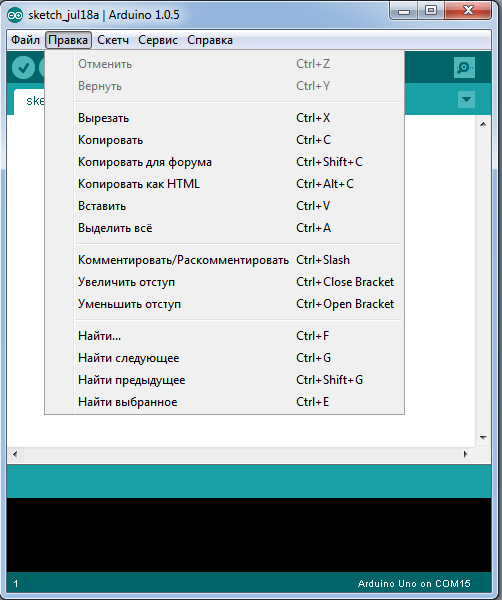


Рис2.4.Меню правка

Меню "Скетч".(рис.2.5) В даному меню продубльовано команда з панелі управління "Перевірити Компілювати". Виконання якої призведе до перевірки вашого коду на помилки, і в разі їх відсутності - до компіляції.Пункт меню "Показати папку скетчів" відкриє робочу директорію *Arduino IDE*, зазначену в налаштуваннях."Додати файл ..." дозволяє відкрити текстовий файл (або скетч) в окремій вкладці

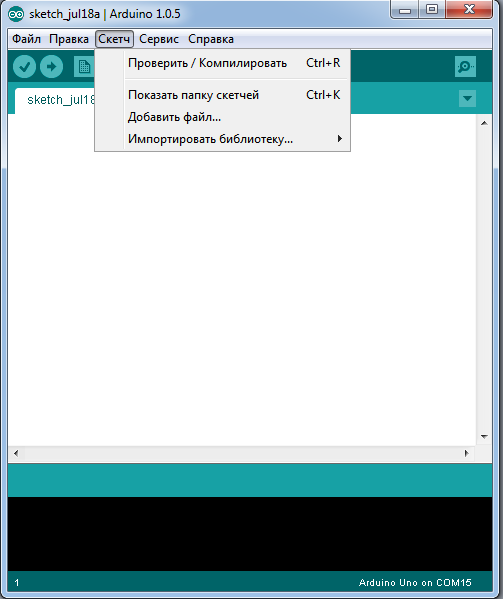


Рис2.5.Меню скетч

Імпорт бібліотек. *Arduino IDE* містить безліч попередньо бібліотек. Бібліотеки додають додаткову функціональність скетчам, наприклад, при роботі з апаратною частиною або при обробці даних. Одна або кілька директив *#include* будуть розміщені на початку коду скетчу з подальшою компіляцією бібліотек і разом зі скетчем. Завантаження бібліотек вимагає додаткового місця в пам'яті *Arduino*. Для встановлення сторонніх бібліотек можна скористатися командою "Імпортувати бібліотеку ...*"Add* *Library ...":* Далі необхідно вказати на папку або *zip* файл з бібліотекою. В результаті бібліотека стане доступна для імпорту через пункт меню (командою *#include*). Імпорт бібліотек. *Arduino IDE* відображено на рис.2.6

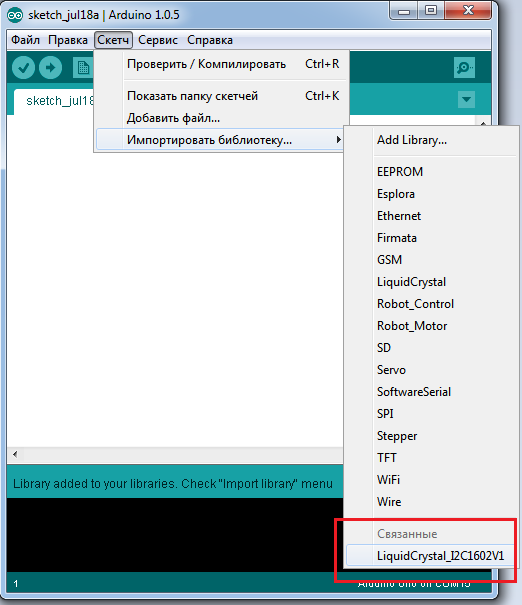


Рис.2.6 Імпорт бібліотек. *Arduino IDE*

У меню "Сервіс" (рис.2.7)необхідно вказати модель вашої *Arduino* плати, а так само *COM* порт, до якого вона підключена. Зручною функцією є автоформатування, яка дозволяє виправити огріхи в розмітці скетчу і привести його в легкий для читання вигляд. Особливо актуально при копіюванні сторонніх програм.Середа *Arduino* *IDE* дозволяє залити *bootloader* на *atmega* мікроконтролери. У пункті меню "Програматор" вказується використовуваний пристрій. Команда "Записати завантажувач" здійснює безпосередньо заливку *bootloader* для зазначеної моделі плати.

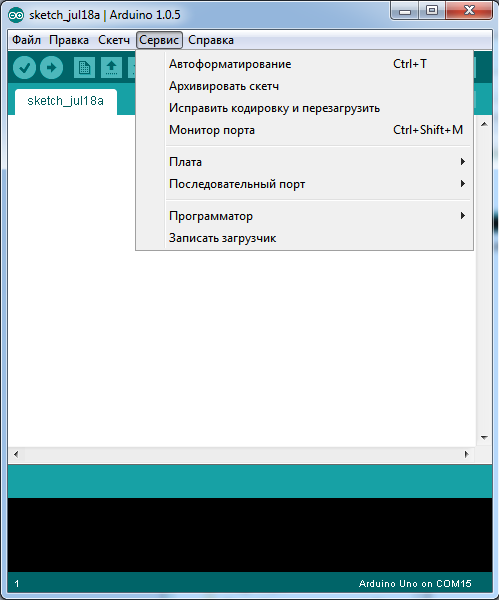


Рис.2.7.Меню сервіс

Пункт меню "Монітор порту" викликає вікно для обміну повідомленнями з *arduino* через *COM* порт.(рис.2.7)

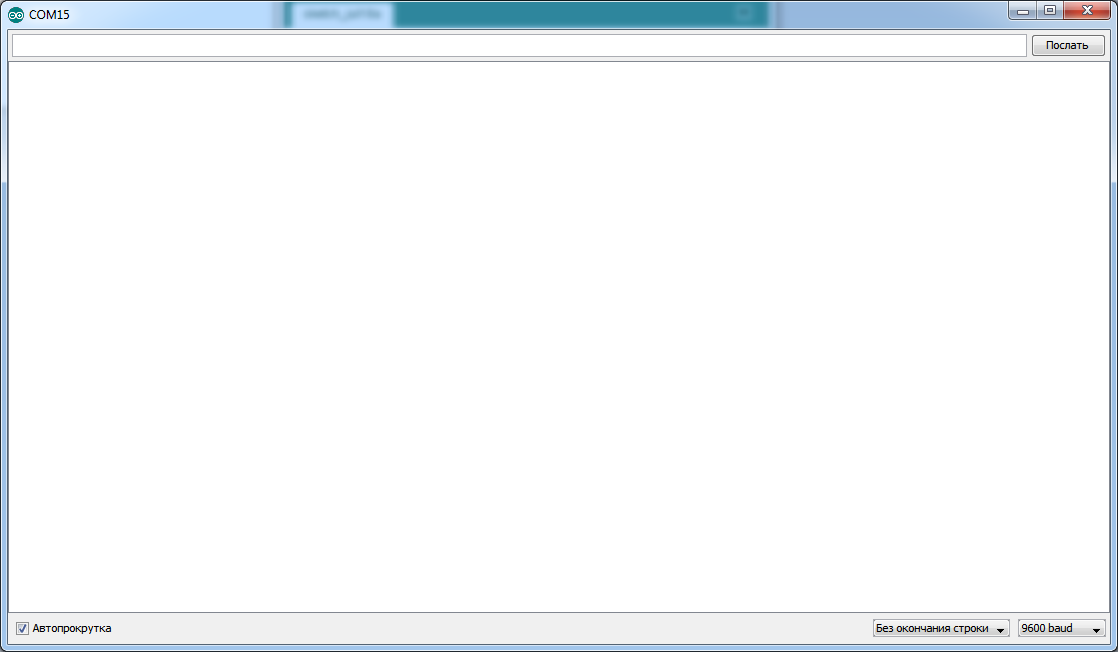


Рис.2.7.Монітор порта

Меню "Довідка"(рис.2.8): Пункти меню ведуть на відповідні статті на офіційний сайт.

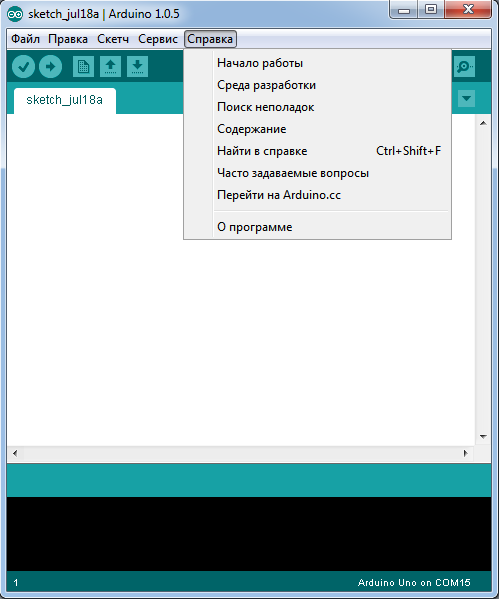


Рис.2.8.Меню довідка

Команди панелі управління, як ми бачимо, дублюють найбільш актуальні пункти меню зображено на рис 2.9

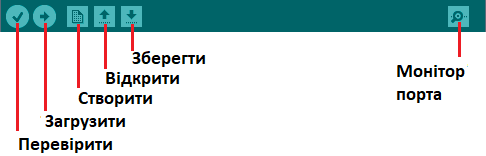


Рис.2.9. Команди панелі управління

2.1.3.*FLProg*

*FLProg* — середовища візуального програмування плат *Arduino*. (рис.2.10)

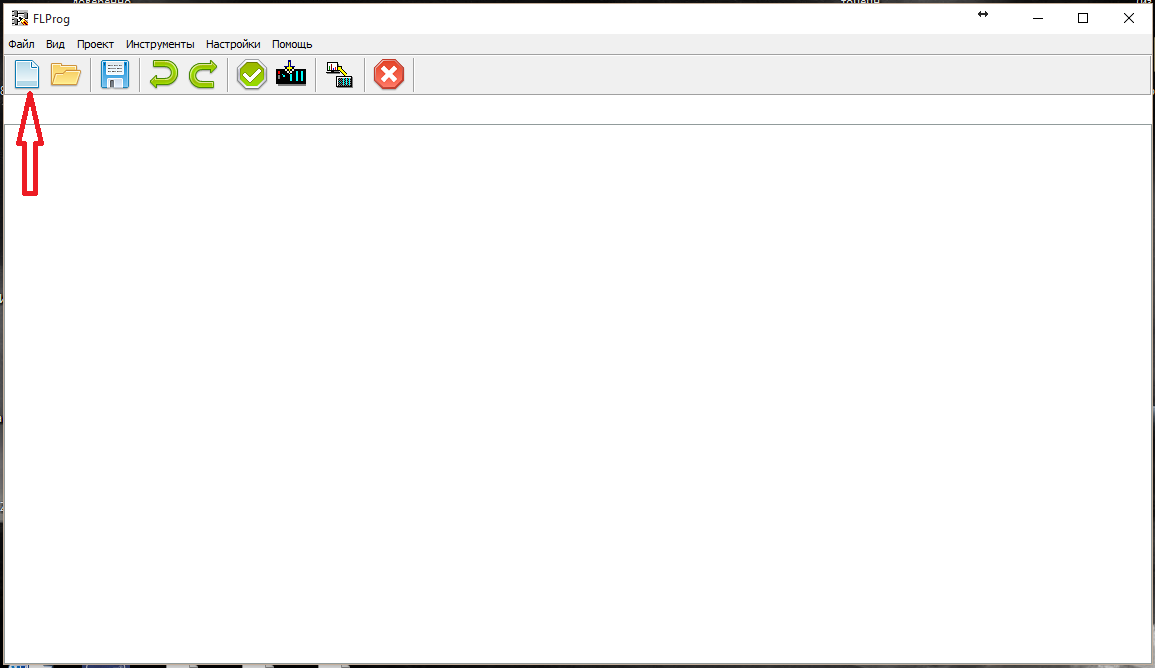


Рис.2.10 Середовище розробки *FLProg*

Щоб створити новий проект натискаємо кнопку «Створити новий проект», як показано на рис.2.11 Відкриється вікно вибору контролера і мови програмування проекту.

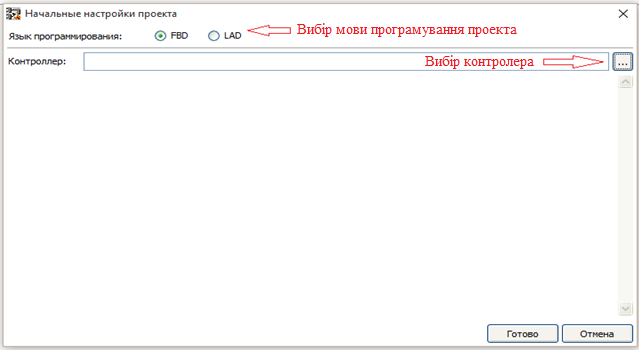


Рис.2.11 Вікно вибору мови програмування і контролера

Для створення проекту можна використовувати будь-який з двох мов програмування *(FBD і LAD)* є стандартами в області програмування промислових контролерів.

*FBD (Function Block Diagram)* - графічна мова програмування. Призначена для програмування програмованих логічних контролерів (ПЛК). Програма утворюється зі списку ланцюгів, які виконуються послідовно зверху вниз. Ланцюги можуть мати мітки. Інструкція переходу на мітку дозволяє змінювати послідовність виконання ланцюгів для програмування умов і циклів.При програмуванні використовуються набори бібліотечних блоків і власні блоки, також написані на *FBD* або іншими мовами МЕК 61131-3. Блок (елемент) - це підпрограма, функція або функціональний блок (І, АБО, НЕ, тригери, таймери, лічильники, блоки обробки аналогового сигналу, математичні операції і ін.).Кожна окрема ланцюг являє собою вираз, складене графічно з окремих елементів. До виходу блоку підключається наступний блок, утворюючи ланцюг. Всередині кола блоки виконуються строго в порядку їх сполуки. Результат обчислення ланцюга записується у внутрішнє змінну або подається на вихід ПЛК.

*Ladder Diagram (LD, LAD, РКС)* - мова релейної логіки. Синтаксис мови зручний для заміни логічних схем, виконаних на релейній техніці. Мова орієнтована на фахівців з автоматизації, які працюють на промислових підприємствах. Забезпечує наочний інтерфейс логіки роботи контролера, який полегшує не тільки завдання власне програмування і введення в експлуатацію, але і швидкий пошук неполадок в підключеному до контролера обладнанні. Програма на мові релейної логіки має наочний і інтуїтивно зрозумілий інженерам-електрикам графічний інтерфейс, який представляє логічні операції, як електричний ланцюг з замкнутими і роз'єднаними контактами. Перебіг або відсутність струму в цьому ланцюзі відповідає результату логічної операції (істина - якщо струм тече; брехня - якщо струм не тече). Основними елементами мови є контакти, які можна образно порівняти з парі контактів реле або кнопки. Пара контактів ототожнюється з логічної змінної, а стан цієї пари - зі значенням змінної. Розрізняються нормально замкнуті і нормально розімкнені контактні елементи, які можна зіставити з нормально замкнутими і нормально розімкнутими кнопками в електричних ланцюгах.

Після створення проекту на одній з мов змінити його буде неможливо.При натисканні кнопки вибору контролера відкриється відповідне вікно, в якому будуть представлені підтримувані програмою плати.В даному списку вибираємо потрібний контролер.Обраний для проекту контролер можна змінити в будь-який момент. Це відображено на рис.2.12.



Рис.2.12 Вікно вибору контролера

Робоче вікно програми *FLProg* складається з декількох полів:

* Основне меню програми
* Дерево проекту
* Дерево встановленого обладнання. У ньому представлено обладнання (проміжні реле, реле часу, генератори ...).
* Бібліотека блоків. У ній знаходиться обладнання, яке можливо застосувати в проекті.
* Область схеми, в якій і буде власне малюватися схема. Схема в *FLProg* являє собою набір плат з обладнанням.

Після завершення роботи над проектом проводиться його компіляція. Після компіляції автоматично відкриється програма *"Arduino IDE"* з завантаженим скетчем вашого проекту. У програмі *"Arduino IDE"* необхідно буде вказати номер *COM* порту, до якого підключений ваш контролер, вибрати його тип, і зробити заливку скетчу в контролер.

В даний час програмою підтримуються такі версії *Arduino*:

* *Arduino Diecimila*
* *Arduino Duemilanove*
* *Arduino Leonardo*
* *Arduino Lilypad*
* *Arduino Mega 2560*
* *Arduino Micro*
* *Arduino Mini*
* *Arduino Nano (ATmega168)*
* *Arduino Nano (ATmega328)*
* *Arduino Pro Mini*
* *Arduino Pro ( ATmega168)*
* *Arduino Pro (ATmega328)*
* *Arduino UNO*

Крім того нещодавно в списку підтримуваних контролерів з'явилася плата *Intel Galileo gen2*. Надалі передбачається поповнення і цього списку, і, можливо, додавання плат, заснованих на контролерах *STM*.

Проект в програмі *FLProg* являє собою набір своєрідних плат, на кожній з яких зібраний закінчений модуль загальної схеми. Для зручності роботи кожна плата має найменування і коментарі.

Для розробки проекту використовується середовище розробки Arduino IDE

Вона найбільш проста у викорисані та має багатий функціонал Також для завантаження програм і зв’язку середовище розробки підключається до апаратної частини *Arduino*.

2.1.4.Вибір платформи для розробки .

Для реалізації проекту потрібно підібрати відповідну характеристикам платформу *Arduino.* Підібрано кілька варіантів .

Платформа *Arduino* *Mini* це пристрій, що використовується в проектах, що вимагають максимальної економії місця. Вона містить 14 цифрових входів / виходів і 4 аналогових вхідних контакту. (Ще чотири доступні, але не виведені.) Пристрій не має ні USB-роз'єму, ні регулятора потужності ні навіть гребінки для підключення периферії. Програмування здійснюється за допомогою зовнішнього *USB* або *RS232* через *TTL* послідовний адаптер.

*Arduino* *Uno* є стандартною платою *Arduino* і можливо найбільш поширеною. Вона заснована на чіпі *Atmel* *ATmega328*, що має на борту 32 КБ флеш-пам'яті, 2 КБ *SRAM* і 1 Кбайт *EEPROM* пам'яті. На периферії має 14 дискретних (цифрових) каналів введення / виводу і 6 аналогових каналів введення / виводу, це дуже різнобічно-корисні девайси, що дозволяють перекривати більшість аматорських завдань в області мікроконтролерній техніки. Чіп *ATmega16u2* на борту управляє послідовної зв'язком. Дана плата контролера є однією з найбільш дешевих і найбільш часто використовуваних.

*Arduino* *Mega* *2560* має значно більше каналів введення - виведення. Вона має в цілому 54 цифрових ліній введення / виводу і 16 аналогових входів. Вона також має велику кількість флеш-пам'яті: 256 КБ, що дозволяє зберігати великі програми, ніж *Uno*. Вона також має чималу *SRAM* і *EEPROM*: 8 КБ і 4 КБ, відповідно. Вона також має 4 апаратних *UART* порту, що робить її ідеальною платформою для комунікацій з декількома пристроями паралельно.Плати *Arduino* *Mega* використовуються там, де необхідна велика кількість входів і виходів.

За вимогами проекту була обрана платформа *Arduino* *Mega* *2560* , вона дає можливість підключити усі потрібні нам модулі , а також достатньо пам'яті для зберігання скетчу проекту.

2.1.5.Вибір каналу передачі даних

Щоб забезпечити канал зв’язку потрібно підібрати найбільш безпечніший і досить потужний варіант . Розглядається кілька варіантів.

*ESP8266* - мікроконтролер китайського виробника *Espressif* з інтерфейсом *Wi-Fi.* Крім *Wi-Fi* мікроконтролер відрізняється можливістю виконувати програми з зовнішньої флеш-пам'яті з інтерфейсом SPI. ESP8266 може працювати як в ролі точки доступу так і кінцевої станції. При нормальній роботі в локальній мережі *ESP8266* конфігуруєтся в режим кінцевої станції. Для цього пристрою необхідно задати *SSID* *Wi-Fi* мережі і, в закритих мережах, пароль доступу. Для початкового конфігурування цих параметрів зручний режим точки доступу. У режимі точки доступу пристрій видно при стандартному пошуку мереж в планшетах і комп'ютерах. Залишається підключитися до пристрою, відкрити HTML сторінку конфігурації і задати параметри мережі. Після чого пристрій штатно підключиться до локальної мережі в режимі кінцевої станції.

*Bluetooth*-модуль *HC*-06 - З *HC*-06 є можливість керувати роботом прямо зі свого смартфона. Поставивши на телефон або планшет одну з численних програм для керування через *bluetooth*, ви можете перетворити його в справжній продвинутий джойстик, і ваш робот зможе слухняно повертати слідом за поворотом смартфона. Точно так само можна зв'язатися зі своїм пристроєм з ноутбука, або з будь-якого іншого приладу, який може підключатися до *bluetooth*-пристроїв.

Сфера застосування цього модуля не вичерпується управлінням. Його можна використовувати і для пересилання показань різноманітних сенсорів.

Робоча напруга цього *bluetooth*-модуля - 3,3 В, але його входи толерантні до 5 В, тому він сумісний з усіма платами *Arduino*.

Модуль *nRF24L01* - це цифровий приймач і передавач, укладений в одній маленькій мікросхемі. Розмір плати, на якій розміщується мікросхема, необхідна обв'язка і невелика антена становить всього 15 x 29 мм. Коротка специфікація радіомодуля:

* несуча частота: 2.4 ГГц;
* робоча напруга: від 3.3 до 3.6 В;
* дальність: до 100 м на відкритому просторі, і до 30 м в приміщенні;
* швидкість: 2 Мб / c;
* кількість каналів: 125.

Радіомодуль використовують для управління мобільним роботом: рухаємо джойстик на пульті - робот їде в потрібному напрямку. У зворотний бік цей же робот може відправляти дані телеметрії: показання одометра, інклінометра, і різних інших датчиків.

Інший варіант застосування - збір даних дистанційних датчиків. *nRF24L01* вміє працювати з топологією один-до-багатьох. Це означає, що один радіомодуль може тримати зв'язок одночасно з безліччю інших. Завдяки цьому, датчики температури, вологості і освітленості, встановлені по всій площі великого цеху або теплиці можуть передавати показання на базову станцію для подальшого аналізу. Ще одна можливість використання погодна станція у дворі житлового будинку, або на його даху. Станція збирає дані і кожні 30 хвилин і відправляє їх на інший *nRF24L01*, що знаходиться в квартирі - теж гідний варіант, адже модуль може перемикатися в режим низького споживання енергії, що дозволить залишити його без обслуговування на довгий час.

Проект реалізована на модулі *nRF24L01* , тому що модуль має можливість передачі даних на великих відстаннях, не такий вразливий до перешкод і має функцію передачі даних в зворотній бік , може перемикатися в режим низького споживання енергії.