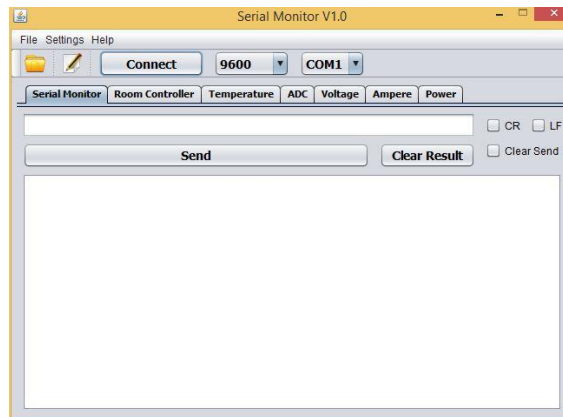


របៀបប្រើប្រាស់កម្មវិធី Serial Monitor V1.0

មុខងាររបស់កម្មវិធីមាន៖

- Serial Monitor
- Room Controller
- Temperature
- ADC
- Voltage
- Ampere
- Power

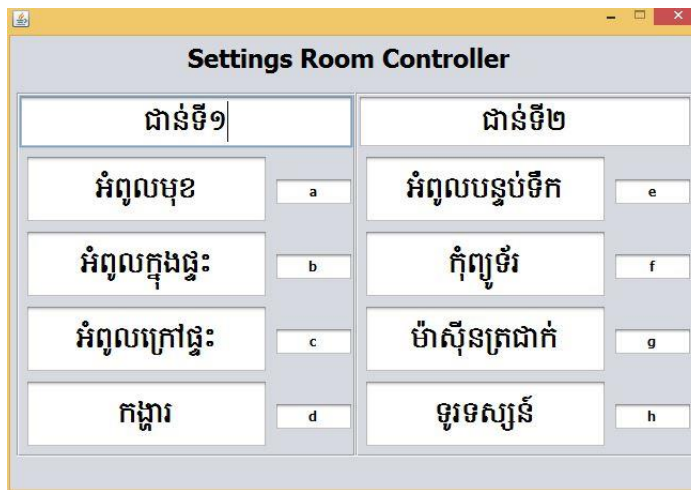
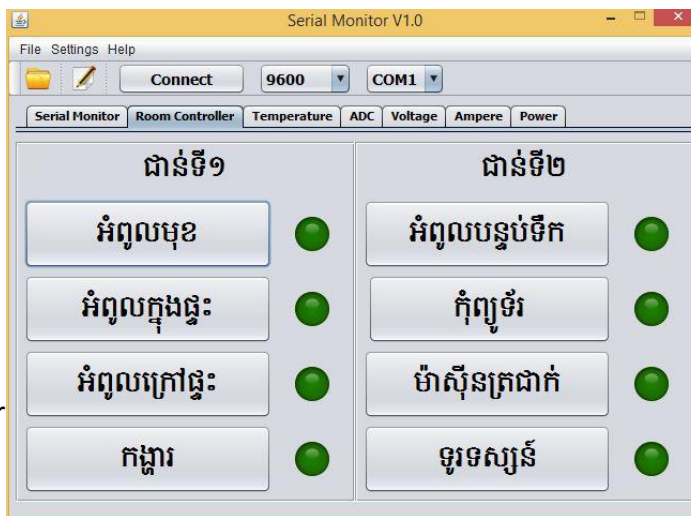


➤ Serial Monitor គឺជាកន្លែងដែលអាចឱ្យយើងមានលទ្ធភាពទទួលទិន្នន័យនិងបញ្ជូនទិន្នន័យតាមរយៈ Serial Port (RS232) ។ យើងអាចមើលឃើញគ្រប់ទិន្នន័យទាំងអស់ដែលអ្នកប្រើប្រាស់បានបញ្ជូនតាមរយៈ Serial Port ឬ Bluetooth Module ជាដើម។ Serial Monitor ក៏មានលទ្ធភាពអាចរក្សាទុកនូវទិន្នន័យទាំងអស់ផងដែរ។

➤ Room Controller គឺជាកន្លែងដែលអាចឱ្យយើងមានលទ្ធភាពបញ្ជា (Send Command to Microcontroller) ឧបករណ៍អគ្គិសនីបានដោយយើងត្រូវមានប្រព័ន្ធ Microcontroller ជាប្រព័ន្ធដែលបញ្ជាដោយផ្ទាល់ទៅឧបករណ៍អគ្គិសនី។ Room Controller អាចបញ្ជាតាមរយៈខ្សែ Serial ឬ តាមរយៈ Bluetooth ។ យើងអាចកែប្រែឈ្មោះប៉ុក្កងនិងឃ្លាបញ្ជា(Command) បានដោយចូលទៅក្នុង Settings ។ អំពូលអាចក្តីបានលុះត្រាតែមានបញ្ជាពីខាងក្រៅ(Microcontroller Feedback) ។ ឃ្លាបញ្ជាមានទំរង់៖

```
_load=cmd\r\n
```

ពាក្យថា load គឺជាឃ្លាបញ្ជាសម្រាប់បញ្ជាអំពូល LED ។ ពាក្យថា cmd វាការបញ្ជាឱ្យអំពូល LED ភ្លឺ ឬ រលត់ ។



cmd មានទំហំ 8 bit cmd ត្រូវតែជាលេខគោលដប់។

\r\n ជាឃ្លាដែលបញ្ជាក់ប្រាប់ឱ្យកម្មវិធី Room Controller ស្គាល់ពីការបញ្ជា(Command) ។
ឧទាហរណ៍៖

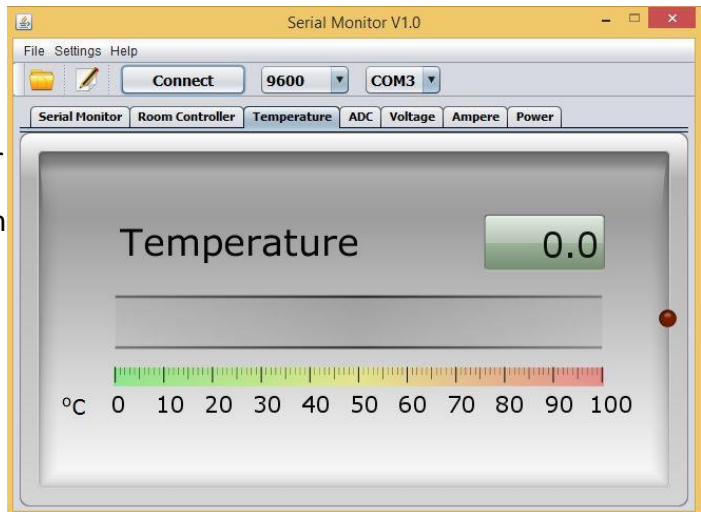
_load=1\r\n 1₁₀ = 00000001₂ អំពូលមួយភ្លឺ(អំពូលមុខ)

_load=40\r\n 1₁₀ = 00101000₂ អំពូលពីរភ្លឺ(កុំព្យូទ័រ កង្ហារ)

- Temperature គឺជាកន្លែងដែលអាចឱ្យយើងមានលទ្ធភាពបង្ហាញសីតុណ្ហភាពពីប្រព័ន្ធ Microcontroller តាមរយៈខ្សែ Serial ឬ តាម Bluetooth និងអាចរក្សាទុកនូវទិន្នន័យទាំងអស់ឃ្លាបញ្ជាមានទំរង់៖

`_temperature=cmd\r\n`

ពាក្យថា temperature គឺជាឃ្លាបញ្ជាសម្រាប់ទំរង់សីតុណ្ហភាព។ ពាក្យថា cmd វាជាតំរូវបស់សីតុណ្ហភាព។
ឧទាហរណ៍៖ _temperature=25\r\n សីតុណ្ហភាព 25 °C



- ADC គឺជាកន្លែងដែលអាចឱ្យយើងមានលទ្ធភាពបង្ហាញតម្លៃ Analog ដែលបម្លែងទៅជា Digital ពីប្រព័ន្ធ Microcontroller តាមរយៈខ្សែ Serial ឬ តាម Bluetooth និងអាចរក្សាទុកនូវទិន្នន័យទាំងអស់។
ឃ្លាបញ្ជាមានទំរង់៖

`_adc1=cmd\r\n`

`_adc2=cmd\r\n`

ពាក្យថា cmd វាជាតំរូវបស់ Digital ដែលបម្លែងពី Analog ។
ឧទាហរណ៍៖ _adc1=250\r\n



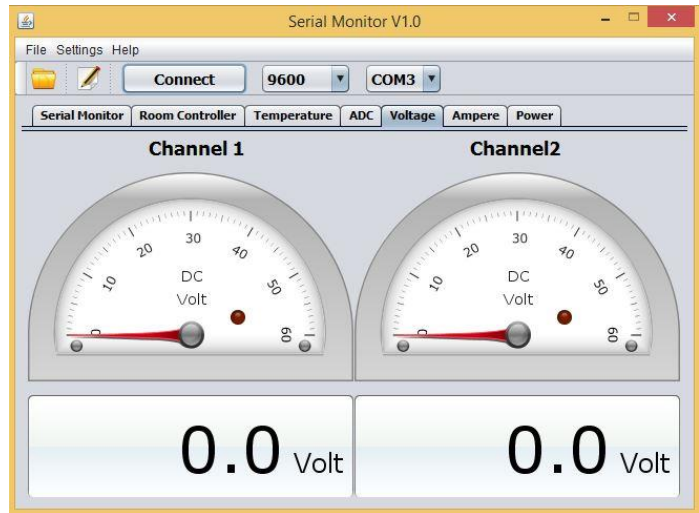
- Voltage គឺជាកន្លែងដែលអាចឱ្យយើងមានលទ្ធភាពបង្ហាញតម្លៃជាតង់ស្យុងអគ្គិសនីពីប្រព័ន្ធ Microcontroller តាមរយៈខ្សែ Serial ឬ តាម Bluetooth និងអាចរក្សាទុកនូវទិន្នន័យទាំងអស់។
ឃ្លាបញ្ជាមានទំរង់៖

```
_voltage1=cmd\r\n
```

```
_voltage2=cmd\r\n
```

ពាក្យថា cmd វាជាតម្លៃរបស់តង់ស្យុងអគ្គិសនី ។

ឧទាហរណ៍៖ _voltage1=25\r\n



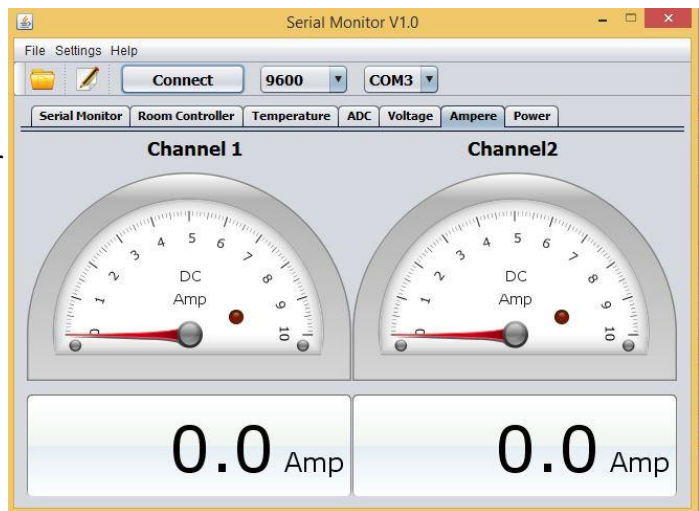
- Ampere គឺជាកន្លែងដែលអាចឱ្យយើងមានលទ្ធភាពបង្ហាញតម្លៃជាចរន្តអគ្គិសនីពីប្រព័ន្ធ Microcontroller តាមរយៈខ្សែ Serial ឬ តាម Bluetooth និងអាចរក្សាទុកនូវទិន្នន័យទាំងអស់។
ឃ្លាបញ្ជាមានទំរង់៖

```
_ampere1=cmd\r\n
```

```
_ampere2=cmd\r\n
```

ពាក្យថា cmd វាជាតម្លៃរបស់ចរន្តអគ្គិសនី។

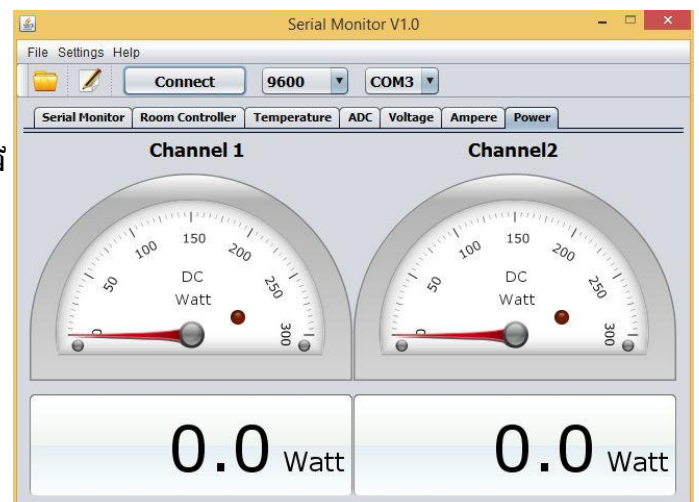
ឧទាហរណ៍៖ _ampere1=5\r\n



- Power គឺជាកន្លែងដែលអាចឱ្យយើងមានលទ្ធភាពបង្ហាញតម្លៃជាអានុភាពអគ្គិសនី និងអាចរក្សាទុកនូវទិន្នន័យទាំងអស់។
ឃ្លាបញ្ជាមានទំរង់៖

```
_power1=cmd\r\n
```

```
_power2=cmd\r\n
```



Example Code use PIC16F887, Bluetooth HC-07 Baud rate 9600

I will send data ADC channel 0 form microcontroller to PC.

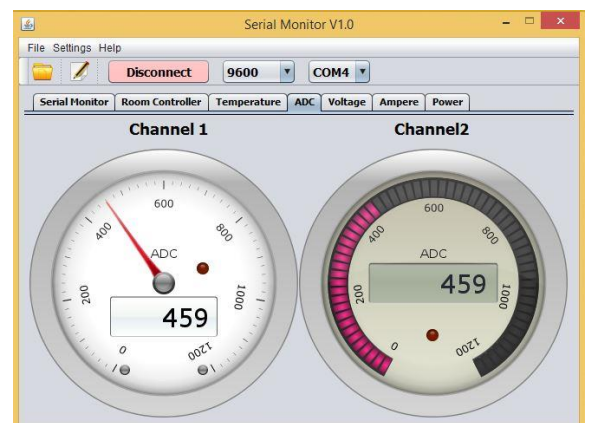
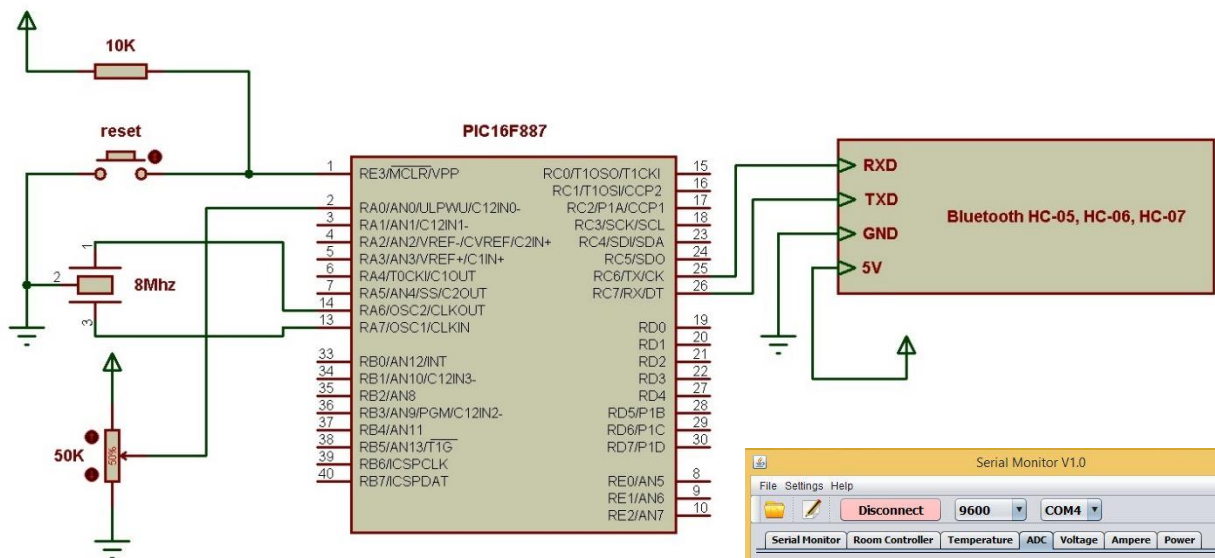
The command is “_adc1=data ADC\r\n”

This code use for MikroC Pro compiler.

```
char uart_rd;
int adc;
char buffer[14];                                     // _adc1=0000\r\n

void main() {
    ANSEL = 1;
    ANSELH = 0;
    TRISA = 0X01;
    UART1_Init(9600);                                // Initialize UART module at 9600 bps

    while (1) {
        adc = ADC_Read(0);                           // Read ADC channel 0
        sprintf(buffer, "_adc1=%4d\r\n", (int)(adc)); // buffer= "_adc1=data ADC"
        UART1_Write_Text(buffer);                     // send buffer to PC
        delay_ms(50);
    }
}
```



Example Code use PIC16F887, Bluetooth HC-07 Baud rate 9600

```

char uart_rd;
char buffer[13];                // _load=000\r\n

void main() {
    ANSEL = 0;                  // Configure AN pins as digital
    ANSELH = 0;

    TRISD = 0;
    PORTD = 0;

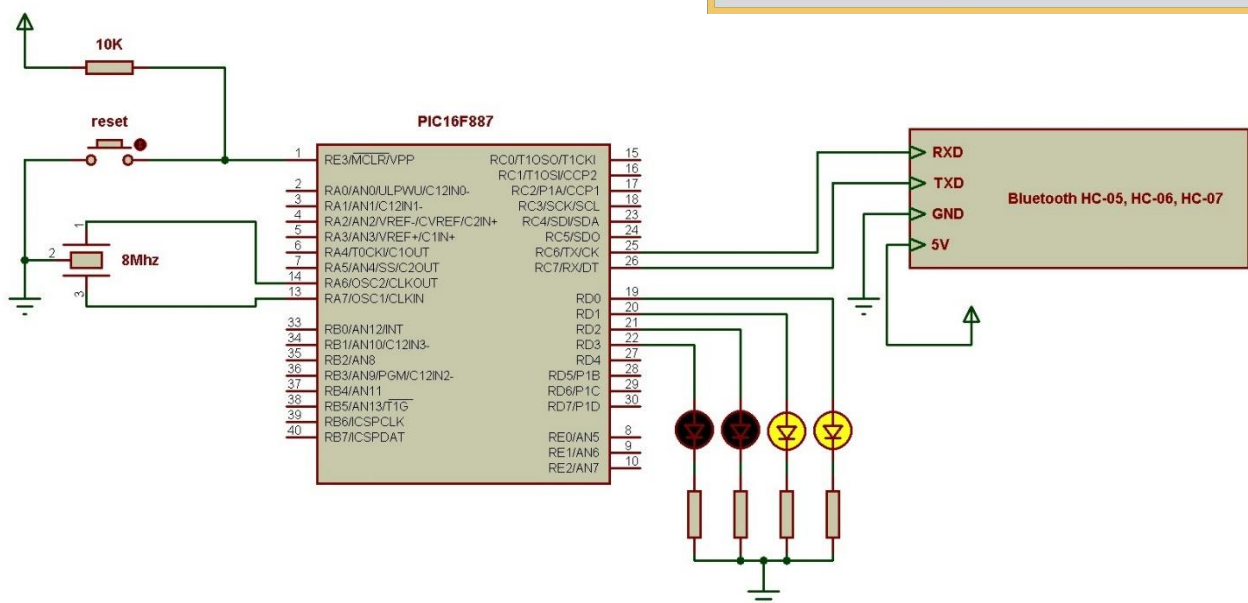
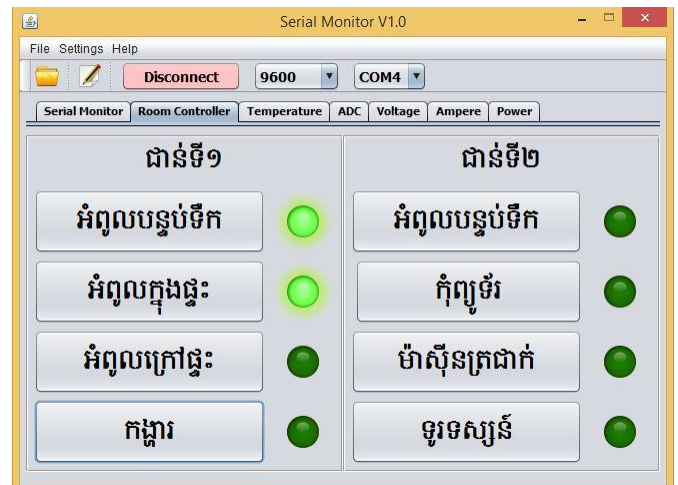
    UART1_Init(9600);           // Initialize UART module at 9600 bps

    while (1) {                 // Endless loop
        if (UART1_Data_Ready()) { // If data is received,
            uart_rd = UART1_Read(); // read the received data,

            if(uart_rd == 'a')
                PORTD.F0 = ~PORTD.F0;
            if(uart_rd == 'b')
                PORTD.F1 = ~PORTD.F1;
            if(uart_rd == 'c')
                PORTD.F2 = ~PORTD.F2;
            if(uart_rd == 'd')
                PORTD.F3 = ~PORTD.F3;

            sprintf(buffer, "_load=%3d\r\n", (int)(PORTD));
            UART1_Write_Text(buffer);
            delay_ms(20);
        }
    }
}

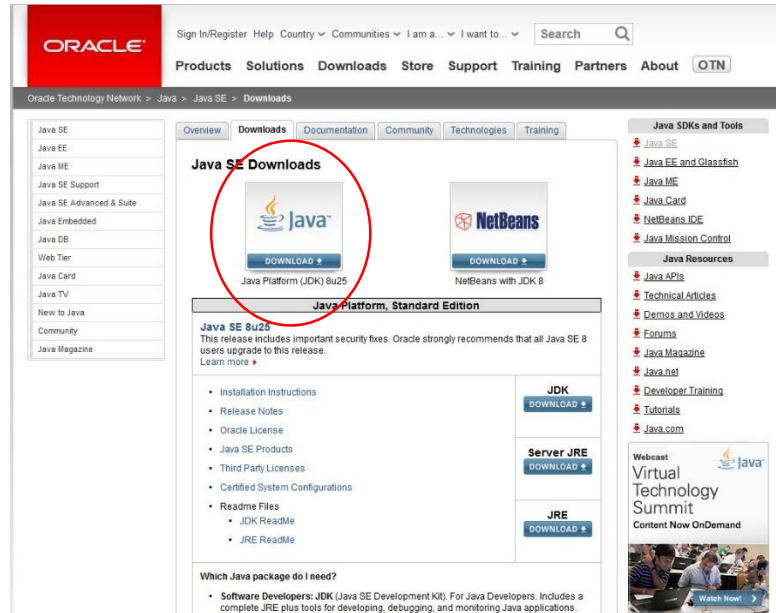
```



Serial Monitor V1.0 អាចដំណើរលើប្រព័ន្ធ MAC OS (Apple Computer) និង Window ។

- MAC OS (Apple Computer) អាចប្រើ Serial Monitor V1.0 បានដោយសារ MAC OS វាមាន JAVA JDK ស្រាប់។
- Window អាចប្រើ Serial Monitor V1.0 បានលុះត្រាតែយើងត្រូវដំឡើង JAVA JDK សិន។

ទាញយក JAVA JDK <http://www.oracle.com/technetwork/java/javase/downloads/index.html>



- ពេលដំឡើង JAVA JDK ហើយយើងអាចប្រើ Serial Monitor V1.0 ដោយបើកវាជាមួយ Java(TM) Platform SE binary ។

