



UNIVERSIDADE FEDERAL RURAL DO SEMI-ÁRIDO
CENTRO MULTIDISCIPLINAR DE PAU DOS FERROS
BACHARELADO EM ENGENHARIA DE COMPUTAÇÃO

Rafael Vieira Abrantes

Relatório de Estágio Supervisionado na Empresa Brisanet em Pereiro - CE

Pau dos Ferros-RN

2019

Rafael Vieira Abrantes

**Relatório de Estágio Supervisionado
na Empresa Brisanet em Pereiro -
CE**

Relatório apresentado a Universidade Federal Rural do Semi-Árido – UFRSA, *Campus* Pau dos Ferros, para a obtenção do título de Bacharel em Engenharia de Computação.

Orientador: Prof. Dr. Ernano Arrais Junior.

Pau dos Ferros-RN

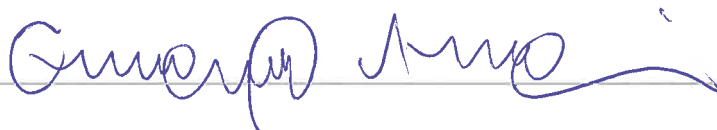
2019

Relatório de Estágio Supervisionado

Relatório apresentado a Universidade Federal Rural do Semi-Árido – UFERSA, Campus Pau dos Ferros, para a obtenção do título de Bacharel em Engenharia de Computação.

APROVADO EM 14/03/2019

BANCA EXAMINADORA



Prof. Dr. Ernano Arrais Junior (Orientador)



Prof. Dr. Francisco Carlos Gurgel da Silva Segunda (1º Membro)



Prof. Dr. Rodrigo Soares Semente (2º Membro)

Resumo

O estágio supervisionado consegue aproximar o aluno do mercado de trabalho, ocasionando maior experiência e observação da união entre a teoria aprendida na universidade e a prática na empresa. O estágio foi realizado na empresa Brisanet situada na cidade de Pereiro – RN, entre os dias 8 de outubro e 16 de novembro de 2018. Foram realizados projetos de automação no controle de vazão, solucionar problemas com tamanho de dados a serem transmitidos, análises de sinais de rádio e GPS, com isso, o estágio foi essencial para compreender como realmente funciona o trabalho dentro de uma empresa.

Palavras-chave: Estágio; Brisanet; automação; mercado de trabalho.

Sumário

1. Introdução.....	6
2. O campo de estágio	7
3. Atividades realizadas.....	8
3.1. Sensor de Vazão.....	8
3.1.1. Primeira Parte	9
3.1.2. Segunda Parte	10
3.1.3. Terceira Parte.....	11
3.2. Transmissão de dados (MQTT)	11
3.3. Transmissor de ondas de rádio.....	12
3.4. GPS	16
4. Relação teoria-prática.....	18
5. Considerações finais.....	19
6. Referências	20
7. Anexos.....	21

1. Introdução

O estágio supervisionado é de suma importância para a capacitação e aprendizado do discente no mercado de trabalho, fazendo que o aluno aprenda a utilizar os ensinamentos da universidade e posteriormente implementá-las no mercado.

O estágio foi realizado na empresa Brisagnet, situada na cidade de Pereiro – CE, na rodovia CE-138, na divisa entre os estados do Ceará e do Rio Grande do Norte. No estágio foram desenvolvidas algumas etapas de um projeto da empresa. O supervisor foi Pedro Guilherme Lopes e o orientador foi o Professor Dr. Ernano Arrais Junior.

O estágio na empresa se deu entre os dias 8/10/2018 e 16/11/2018, no período das 8 horas até as 15 horas, contabilizando as 160 horas necessárias para a realização do estágio supervisionado, conforme estabelecido no Projeto Pedagógico do Curso.

2. O campo de estágio

A Brisagnet foi criada em 1998, sendo uma empresa voltada para a área de telecomunicações, oferecendo serviços de internet, telefone fixo e móvel e servidor de televisão por assinatura, sendo localizada da cidade de Pereiro – CE. A empresa já atende mais de 200 cidades espalhadas nos estados do Ceará, Paraíba, Pernambuco e Rio Grande do Norte.

A empresa realiza trabalhos paralelos na área de automação para projetos internos e paralelos do foco principal. Na Figura 1 tem-se uma imagem do prédio sede da empresa.

Figura 1: Sede da empresa Brisagnet.



Fonte: Brisagnet.

O prédio da Brisagnet contém 2 andares. No térreo é localizado o laboratório de eletrônica contendo todos equipamentos necessários para a realização do estágio: osciloscópio, fonte de tensão, multímetro, amperímetro, ferro de solda, sensores, componentes eletrônicos (resistores, capacitores, etc.) e impressora 3D.

3. Atividades realizadas

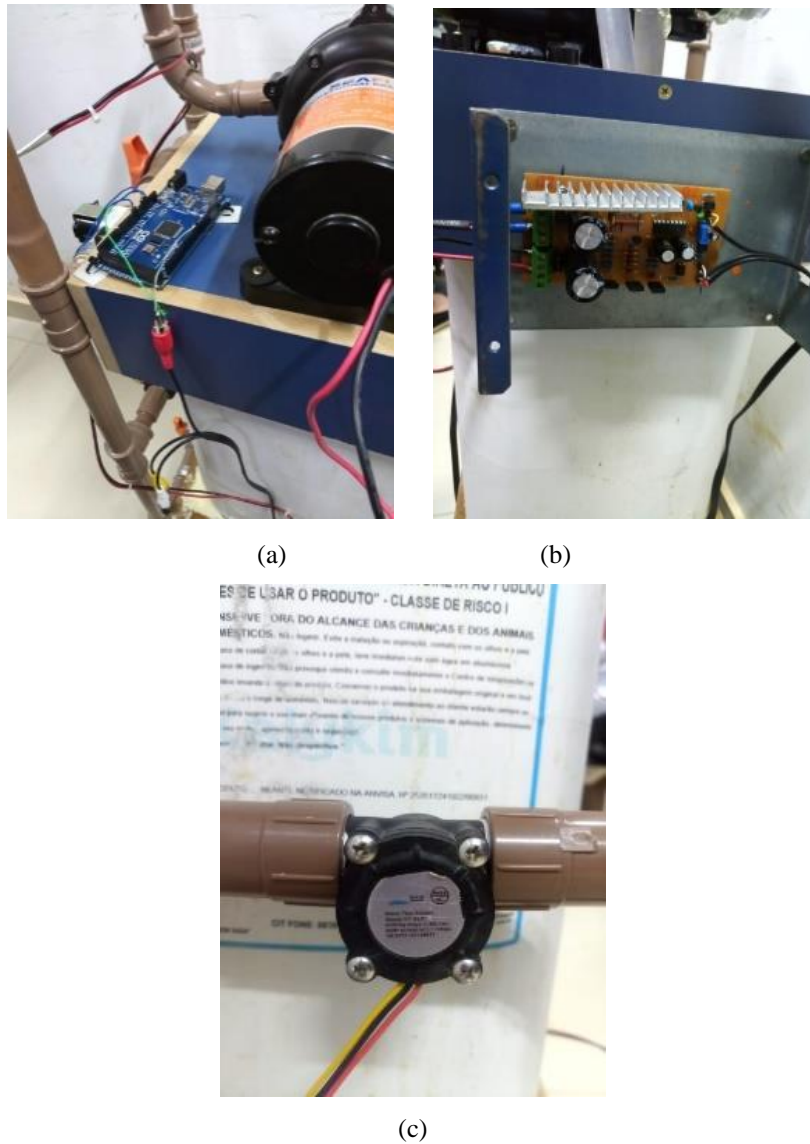
Foram realizadas 4 atividades distintas, sendo elas:

- Controle da bomba hidráulica de corrente contínua com o sensor de vazão. O projeto foi dividido em 3 etapas:
 - Primeira etapa: Funcionamento do sensor de vazão;
 - Segunda etapa: Controle da bomba hidráulica com o sensor de vazão;
 - Terceira etapa: Funcionamento do controle automático;
- Aumento da taxa de transmissão de dados, utilizando o esp8266;
- Teste de distância do sinal de rádio frequência (RF) utilizando o HC-12;
- Teste do GPS em movimento;

3.1. Sensor de Vazão

Para testar o sistema foi projetado um equipamento de teste, conforme apresentado na Figura 2. O equipamento contém uma bomba hidráulica de corrente contínua (DC) para bombear a água no encanamento, um sensor de vazão e um circuito que limita tensão e corrente, sendo o controle efetuado através de um microcontrolador Arduino.

Figura 2: (a) Sistema de vazão com o arduino e o motor DC, (b) Circuito para o controle do motor DC e (c) sensor de vazão.



Fonte: O autor.

3.1.1. Primeira Etapa

Na primeira parte foi realizado o teste do funcionamento do sensor de vazão utilizando o Arduino. Foi pesquisado o *datasheet* do sensor de vazão e sobre o modo de operação do sensor, com isso, foi implementado um programa para a leitura da vazão conforme apresentado na Figura 3. No programa foi retirado a frequência que o sensor de vazão gera, e depois colocado na fórmula da vazão obtida de acordo com o *datasheet* do sensor.

Figura 3: Parte do código da leitura de vazão.

```

double frequencia; //CALCULO PARA FRENQUENCIA DA VAZAO REAL
double vazao; //CALCULO DA VAZAO COM A EQUAÇÃO DO DATASHEET DO SENSOR

double pulsos_vazao; //RETIRADA DOS PULSOS DE CLOCK DO SENSOR
double valor_pulsos;

pulsos_vazao = 0; //INICIAR A CONTAGEM DOS PULSOS DO SENSOR COM 0
pulsos_vazao = pulseIn(sensor, LOW); //CALCULO DA VAZAO EM MICROSSEGUNDOS
valor_pulsos = pulsos_vazao * 0.000001; //TRANSFORMANDO PARA SEGUNDOS
frequencia = pow(valor_pulsos * 2, -1); //CALCULO DA FREQUANCIA COM SOMENTE A BORDA DE SUBIDA

vazao = (frequencia * 0.1212) + 0.0535; //EQUAÇÃO OBTIDA A PARTIR DO DATASHEET DO SENSOR

```

Fonte: O autor.

O resultado código para a leitura de vazão ocorreu conforme o esperado e com o valor máximo de 10 L/min, com isso, foi realizado a próxima etapa.

3.1.2. Segunda Parte

A segunda parte utilizando o sensor de vazão foi para controlar o motor conforme determinado pelo usuário. Para implementar o código foi sugerido a utilização do controle Proporcional, Integral e Derivativo (PID).

O PID é utilizado para aplicação que necessita de mais precisão. O controle é constituído em três partes:

- A primeira parte chamada de Proporcional (P) ou ganho, por exemplo, utilizando dois carros um atrás do outro e deseja que o carro atrás se aproxime a uma distância x (*setpoint*) do carro à sua frente, então, o carro acelera proporcionalmente até chegar próximo ao outro carro, porém, ele desacelera e distância do outro carro e logo após acelera de novo tentando mantém a distância desejada (SILVEIRA, 2017).
- A segunda parte é a Integral (I). Utilizando o mesmo exemplo, o carro agora tenta de forma mais precisa e suave realizar a aproximação e manter a distância desejada (SILVEIRA, 2017).
- A terceira parte é a Derivada (D) é utilizada para a eliminação do erro acumulado na integral (SILVEIRA, 2017).

Na Figura 4, a parte principal da classe gerada do controle do PID para o controle da vazão, no PID existe três pontos chaves, primeiro a proporcionalidade, segundo a integral e terceiro o derivativo (SILVEIRA, 2018).

Figura 4: Código do controle da vazão.

```

double process()
{
    //implementação P I D
    erro = setPoint - sample;
    double deltaTime = ((millis() - lastProcess) / 1000) * 60;
    lastProcess = millis();

    //p
    P = erro * KP;

    //I
    I += (erro * KI) * deltaTime;

    //D
    D = (lastSample - sample) * KD / deltaTime;

    // soma tudo

    pid = P + I + D;

    // Serial.println("\nValor do PID = ");
    // Serial.println(pid,6);
    // Serial.println("\n");
    return pid; //RESULTADO DO PID PARA O SISTEMA DE CONTROLE
}

```

Fonte: O autor.

Logo após foram realizados os testes e conforme o usuário foi determinando o valor da vazão desejado, o sistema gerou um valor PWM para o circuito e conseguia controlar o motor e conseqüentemente a vazão. Com o funcionamento do PID, foi ajustado os valores constantes do constante proporcional (KP), constante integral (KI) e constante derivativa (KD) para melhorar a precisão do PID. Na acima apresenta uma parte do PID com as constantes.

3.1.3. Terceira Parte

A terceira parte do controle de vazão foi retirar a ação do usuário determinar o valor da vazão, sendo assim, o sistema iria fazer o PWM (Modulação Por Largura de Pulso) variável até que a vazão se manter constante ou não variar muito, com isso, o valor da vazão constante seria o do controle do PID fazendo o sistema controlar a vazão.

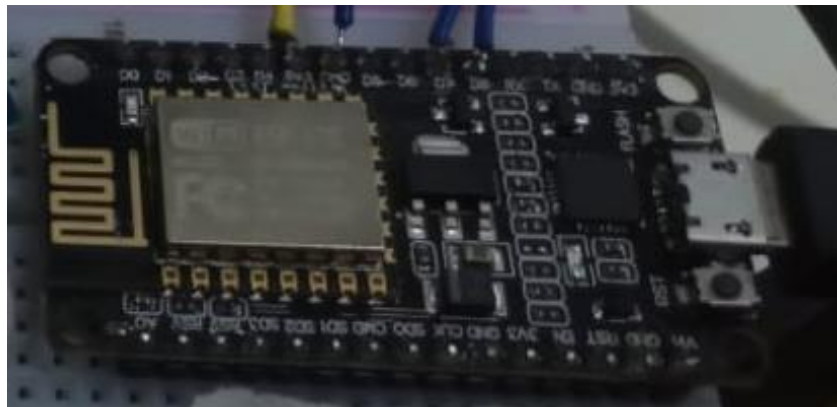
Adicionando no código realizado na segunda parte obteve resultado satisfatório, porém somente ocorria quando era manipulada a válvula do encanamento, com isso, foi discutindo com o supervisor o ocorrido e concluiu-se que precisaria do sensor de pressão com o de vazão para o sistema funcionar nas especificações do projeto, sendo assim, o projeto do controle de vazão ficou suspenso até a espera do sensor de pressão.

3.2. Transmissão de dados

Após o término do projeto com o sensor de vazão, outra tarefa durante o estágio foi proposta para execução: tentar usar o ESP8266 para trabalhar com o protocolo MQTT (*Message Queue Telemetry Transport*) e tentar possibilitar a transmissão maior dos dados.

Primeiro foi realizar o entendimento do ESP8266 e sobre o MQTT, (na Figura 5 mostra o ESP8266).

Figura 5: ESP8266.



Fonte: O autor.

O MQTT é muito utilizado para dispositivos de internet das coisas (IoT). O modelo de publicação e assinatura do MQTT é definido em três etapas:

- Primeiro: conexão do cliente ao broker, sendo que, o broker é um servidor que recebe as mensagens dos clientes, logo após a conexão, roteia as mensagens para os clientes de destino. Com a conexão o cliente assina o “tópico” de mensagem no broker (YUAN, 2017).
- Segundo: as mensagens do cliente são publicadas em um tópico, enviado tudo ao broker (YUAN, 2017).
- Terceiro: O broker encaminha todas as mensagens aos clientes que assinam o tópico (YUAN, 2017).

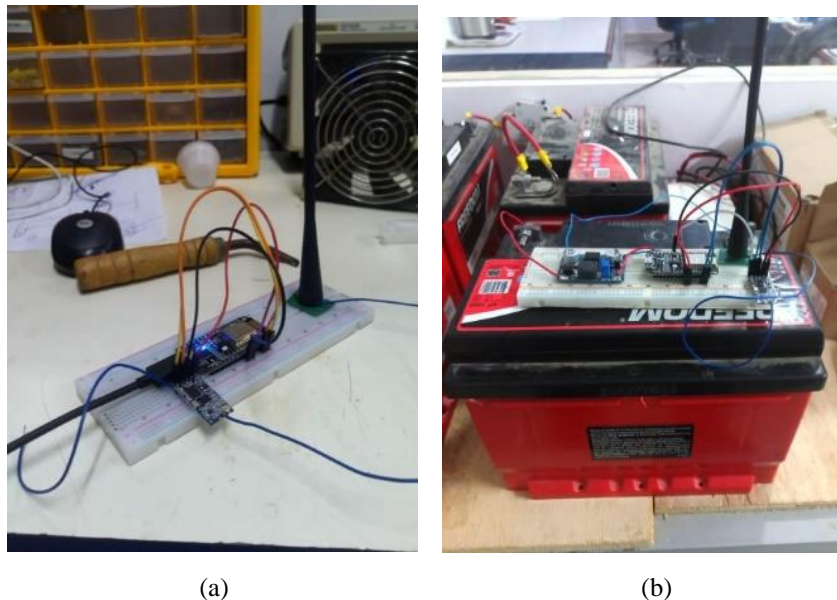
Com algumas pesquisas, foi encontrado algumas possibilidades aumento do tamanho da transmissão dos dados. Utilizando a interface de programação do Arduino, foi modificada a biblioteca do MQTT, porém, ao modificar a biblioteca foi encontrado diversos erros e ao corrigir os erros, aparecia mais e mais, com isso, foi decidido a suspensão do projeto e o supervisor decidir pular para o próximo projeto.

3.3. Transmissor de ondas de rádio

O terceiro projeto foi para analisar o alcance de ondas de rádio em volta da empresa utilizando dois ESP8266 e dois HC-12. Assim, foi criado um código para cada ESP8266, nomeado de ESP 7 e ESP 8. O código do ESP 7 mandava uma mensagem para o ESP 8 utilizando o HC-12, então, o ESP 8 recebia e mandava uma mensagem de confirmação para o ESP 7 e o ESP 7 confirmava o recebimento ativando um led.

O circuito com os dois ESP8266 é mostrado na Figura 6 (a) e (b) com a segunda antena utilizada no HC-12, a primeira foi a padrão.

Figura 6: (a) ESP 7 e (b) ESP 8.



Fonte: O autor.

No primeiro teste foi utilizado a antena padrão e a frequência padrão do HC-12 de 433 MHz, e foi traçado no mapa o alcance em volta da empresa. Foram feitos três pontos fixos, o primeiro dentro do laboratório de eletrônica, segundo foi na porta da frente da Brisanet e o terceiro ponto foi na frente da empresa Nossa Fruta.

Na Figura 7 (a), o primeiro ponto: obteve-se a distância máxima de 69 m, devido existirem diversas atenuações no local devido a paredes e containers; na Figura 7(b), a distância máxima de 108 m, uma vez que neste teste a quantidade de atenuações era menor; na Figura 7 (c) obteve a distância máxima de 233 m, sendo o espaço com número muito menor de atenuações.

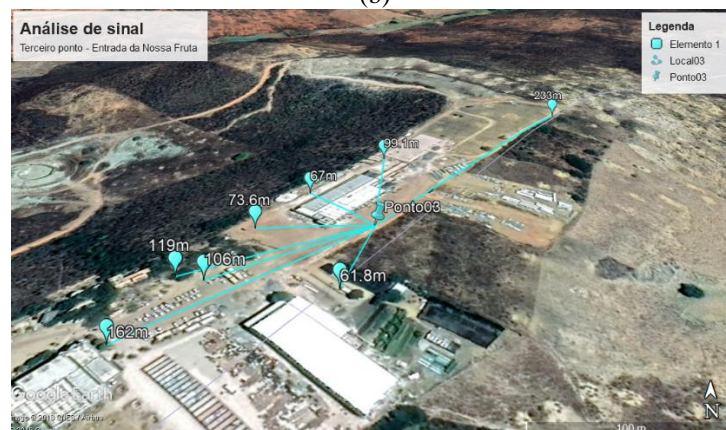
Figura 7: Análise do alcance, (a) dentro do laboratório de eletrônica, (b) porta da frente da Brisanet e (c) de frente da Nossa Fruta.



(a)



(b)



(c)

Fonte: O autor.

Logo após foi substituído a antena padrão pela a da Figura 7 e obteve-se resultados melhores. Na Figura 8 (a) obteve 74,2 m, na Figura 8 (b) foi de 220m e na Figura 8 (c) de 268m devido ao menor número de atenuações.

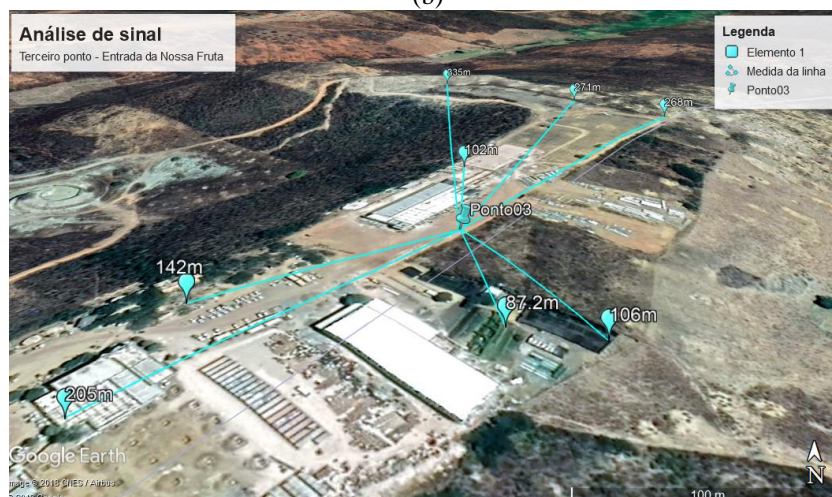
Figura 8: Análise do sinal com a segunda antena, (a) primeiro ponto, (b) segundo ponto e (c) terceiro ponto.



(a)



(b)



(c)

Fonte: O autor.

Todos os dados obtidos com a segunda antena foram melhores com que a antena padrão, com isso, a distância com a segunda antena foi satisfatória para utilização do projeto da empresa.

Na Tabela 1 é apresentado todas as distâncias encontras com as duas antenas, no lado esquerdo são os dados com a antena padrão e no direito são os dados com a outra antena. Na Tabela 1 pode observar que as distâncias com a antena padrão obteve uma distância menor.

Tabela 1: Distância do sinal.

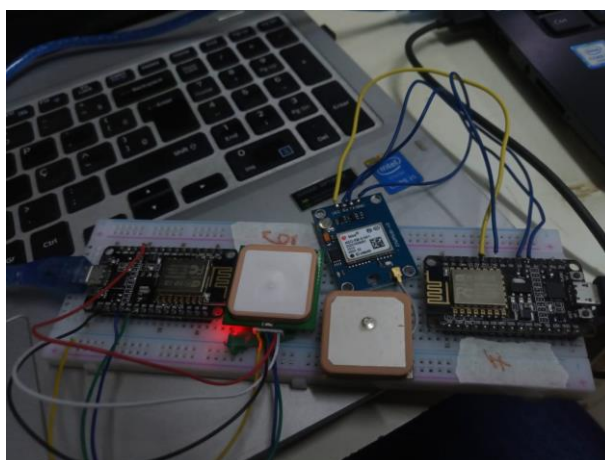
Dados	Ponto 1 (m)		Ponto 2 (m)		Ponto 3 (m)	
1	58,8	74,2	52,8	74	162	205
2	51,7	52,2	78,7	125	119	142
3	37,8	59,8	108	220	106	87,2
4	30,04	62,6	98,3	182	73,6	106
5	69,3	82,9			61,8	102
6		59,1			67	335
7					99,1	271
8					233	268

Fonte: O autor.

3.4. GPS

O último projeto trabalhado na empresa foi o teste do GPS em movimento. Com suspeitas do mal funcionamento do equipamento o supervisor decidiu testa-lo em movimento, então, utilizamos dois GPS e dois ESP8266 e foram testados em movimento entre Pereiro – CE e Pau dos Ferros – RN. Na Figura 9 apresenta os dois GPS e os dois ESP8266 testados.

Figura 9: Dois ESP8266 com o GPS.

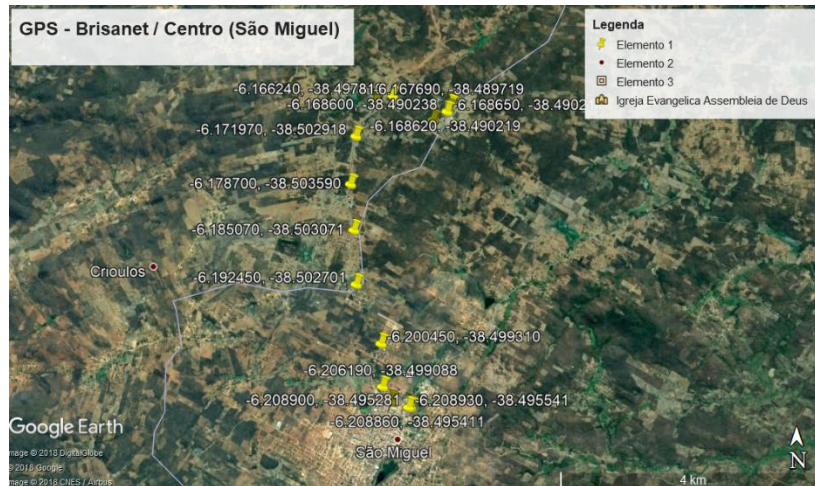


Fonte: O autor.

No primeiro teste foi realizado somente com o GPS com suspeitas de problemas.

No percurso determinado o GPS parou de funcionar na saída da cidade de São Miguel – RN, conforme apresenta a Figura 10.

Figura 10: Teste com o primeiro GPS.



Fonte: O autor.

Em seguida foi utilizado o segundo GPS que realizou todo o percurso sem apresentar problemas, sendo que não foi necessário traçar no mapa o percurso, com isso, para os próximos projetos da empresa foi descartado o primeiro GPS.

4. Relação teoria-prática

Durante o período do estágio foram encontradas diversas dificuldades, principalmente porque nem foram cursadas todas as disciplinas do curso, por exemplo, a utilização do PID do sistema do controle da vazão, com isso, foi realizado um estudo bibliográfico para implementação da proposta.

Algumas disciplinas foram essenciais para realizar alguns projetos, como as disciplinas de circuitos elétricos, eletrônica analógica e circuitos digitais. Para implementação do código no Arduino foi utilizada toda a base da disciplina de sistemas digitais.

A compreensão do protocolo MQTT foi facilidade por relação parecidas com alguns conteúdos da disciplina de redes de computadores.

A criação dos códigos de cada projeto utilizou conceitos aprendidos em algoritmos de programação, estrutura de dados e paradigmas de programação.

5. Considerações finais

O estágio supervisionado foi de grande importância para o aprimoramento do aprendizado e de como se comportar em uma empresa, aproveitando o máximo da experiência, sendo essencial para a formação profissional e pessoal, além de realizar os projetos foi também importante como trabalhar em equipe e o relacionamento com outros colaboradores de diversas áreas.

Os projetos realizados com exceção do aumento do tamanho de dados obtivam êxitos. Foi encontrado dificuldade em trabalhar com equipamentos sem o conhecimento de imediato, mas com algumas pesquisas foram sanadas.

Como o estágio supervisionado foi realizado antes do período de conclusão do curso, algumas disciplinas essenciais não foram assistidas, dificultando a implementação de algumas atividades, e isso levou a um estudo reforçado e proporcionou a curiosidade em conhecimentos novos e diferenciados.

6. Referências

BRISANET. **CONHEÇA A HISTÓRIA DA BRISANET**. 2018. Disponível em: <<https://blog.brisanet.com.br/index.php/2018/11/01/conheca-a-historia-da-brisanet/>>.

Acesso em: 11 jan. 2019.

DEITEL, Paul J.; DEITEL, Havey. **C: como programar**. 6.ed. São Paulo: Pearson Prentice Hall, 2011.

DEITEL, Paul; DEITEL, Harvey. **C++ how to program: late objects version**. 7.ed. New Jersey: Pearson, 2011. 926p. ISBN: 9780132165419.

JOHNSON, David E; HILBURN, John L; JOHNSON, Johnny R. **Fundamentos de análise de circuitos elétricos**. Rio de Janeiro: LTC, 2012. 539p. ISBN: 9788521612384.

NISE, Norman S. **Engenharia de sistemas de controle**. Rio de Janeiro: LTC, 2012. 659p. ISBN: 978-85216-2135-5.

OGATA, Katsuhiko. **Engenharia de controle moderno**. São Paulo: Pearson Prentice Hall, 2010. 800 p. ISBN: 9788576058106.

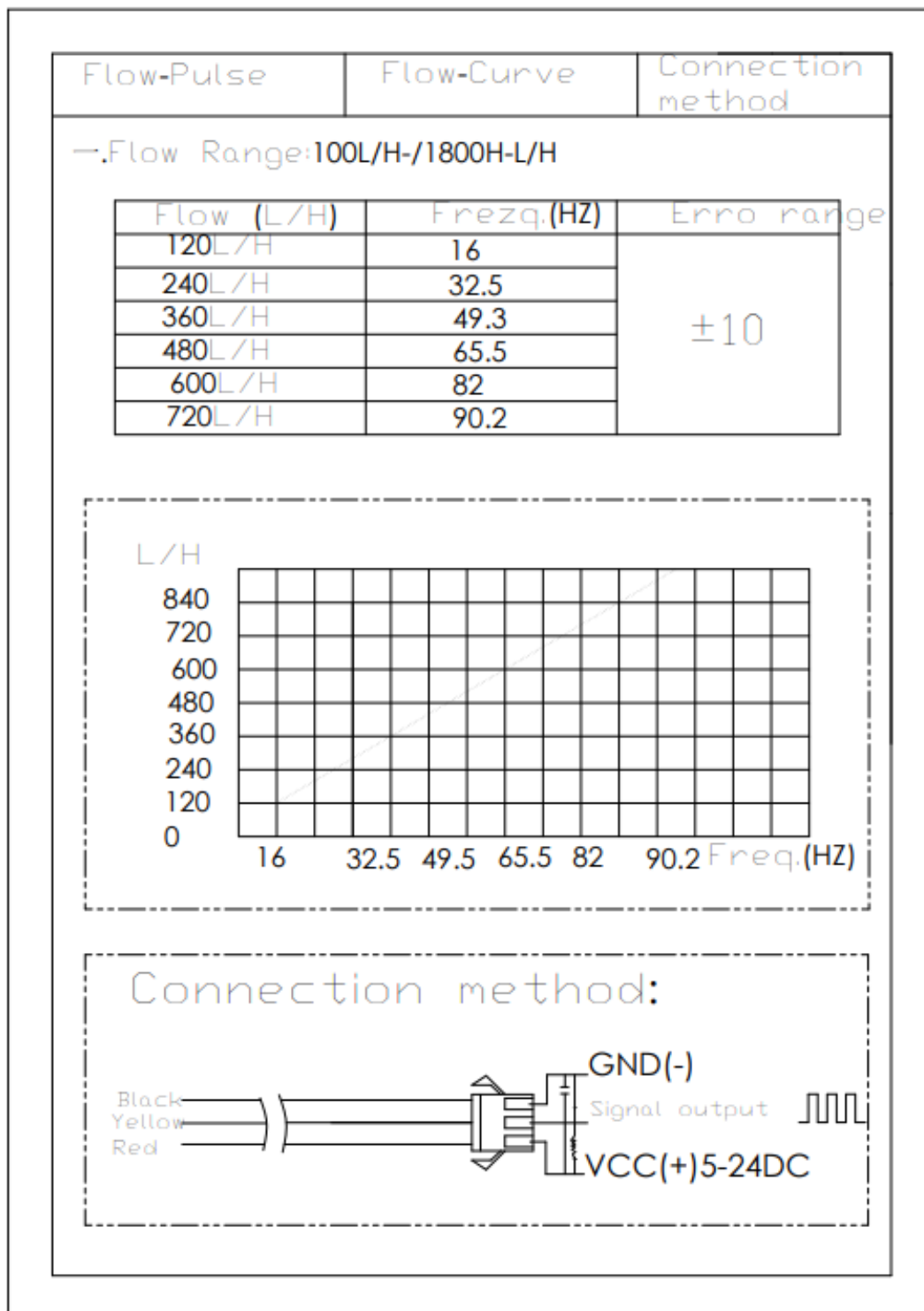
SEDRA, Adel S; SMITH, Kenneth C. **Microeletrônica**. 5.ed. São Paulo: Pearson, 2007. 847p. ISBN: 9788576050223.

SILVEIRA, Cristiano Bertulucci. **O Controle PID de Forma Simples e Descomplicada**. 2017. Disponível em: <<https://www.citisystems.com.br/controle-pid/>>. Acesso em: 1 nov. 2018.

VAHID, F. **Sistemas digitais: projeto, otimização e HDLS**. Porto Alegre: Bookman, 2008. 560p;

YUAN, Michael. **Conhecendo o MQTT**. 2017. Disponível em: <<https://www.ibm.com/developerworks/br/library/iot-mqtt-why-good-for-iot/index.html>>. Acesso em: 10 mar. 2019.

7. Anexo A – *datasheet* sensor de vazão (YF-S201)



YIFA the plastics Ltd Product Introduction

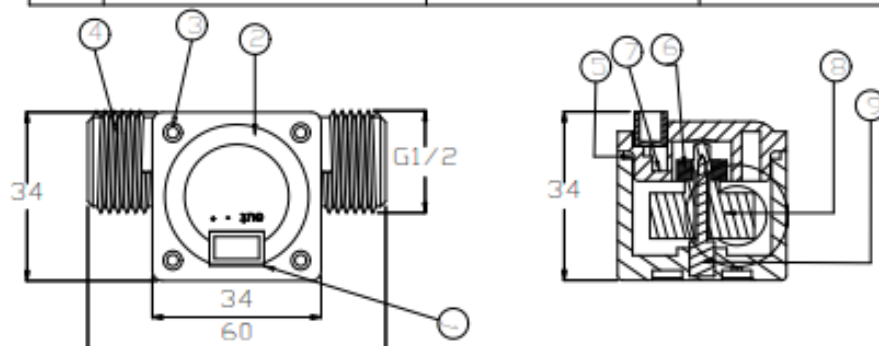
- 1.Modle:YF-21
- 2.Product Name:Hall sensor
- 3.Flow Range: 1-30L/MIN
- 4.(1)Connection Method



- (2)Voltage Range 3.5-24VDC, Pulse Characteristic:F=7Q(L/MIN).
- (3)Extent of error:±5%.
- (4)Flow-Pulse
 - 2L/MIN=16HZ 4L/MIN=32.5HZ 6L/MIN=49.3HZ
 - 8L/MIN=65.5HZ 10L/MIN=82HZ

5.Bom

No.	Item	Material	Qty.
1	Connection wire		1
2	Bonnet	PA	1
3	Screw		4
4	Valve body	PA	1
5	Leak press valve		1
6	Magnet		1
7	Hall		1
8	Impeller	POM	1
9	Rustless steel axis	SUS304	1
10			
11			



Product Features

- Long-distance wireless transmission (1,000m in open space/ baud rate 5,000bps in the air)
- Working frequency range (433.4-473.0MHz, up to 100 communication channels)
- Maximum 100mW (20dBm) transmitting power (8 gears of power can be set)
- Three working modes, adapting to different application situations
- Built-in MCU, performing communication with external device through serial port
- The number of bytes transmitted unlimited to one time
- Update software version through serial port

Product Introduction

HC-12 wireless serial port communication module is a new-generation multichannel embedded wireless data transmission module. Its wireless working frequency band is 433.4-473.0MHz, multiple channels can be set, with the stepping of 400 KHz, and there are totally 100 channels. The maximum transmitting power of module is 100mW (20dBm), the receiving sensitivity is -117dBm at baud rate of 5,000bps in the air, and the communication distance is 1,000m in open space.

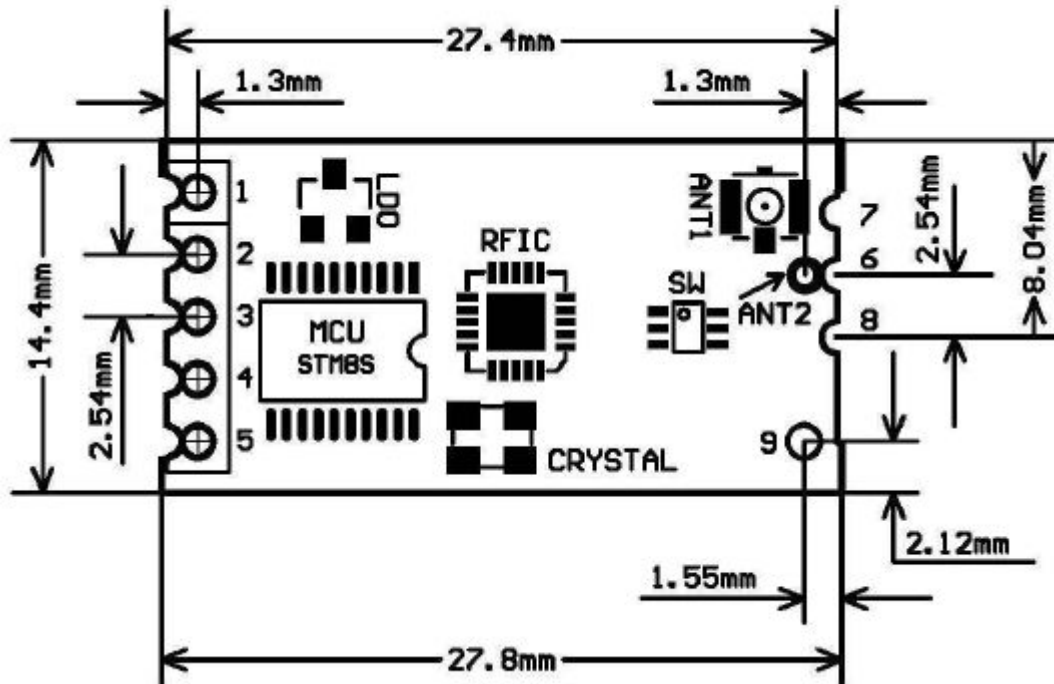
The module is encapsulated with stamp hole, can adopt patch welding, and its dimension is 27.8mm × 14.4mm × 4mm (including antenna cap, excluding spring antenna), so it is very convenient for customers to go into application system. There is a PCB antenna pedestal ANT1 on the module, and user can use external antenna of 433M frequency band through coaxial cable; there is also an antenna solder eye ANT2 in the module, and it is convenient for user to weld spring antenna. User could select one of these antennas according to use requirements.

There is MCU inside the module, and user don't need to program the module separately, and all transparent transmission mode is only responsible for receiving and sending serial port data, so it is convenient to use. The module adopts multiple serial port transparent transmission modes, and user could select them by AT command according to use requirements. The average working current of three modes FU1, FU2 and FU3 in idle state is 80µa, 3.6mA and 16mA respectively, and the maximum working current is 100mA (in transmitting state).

Product Configuration

Standard configuration of HC-12 module only contains one 433MHz-frequency-band wireless communication module with IPEX20279-001E-03 standard RF socket. The optional accessories are 433MHz-frequency-band spring antenna, IPEX-to-BNC coaxial cable and matching 433MHz-frequency-band omni-directional rubber antenna of BNC connector. User could purchase them according to use requirements.

Product Dimension



Definition of Pins

HC-12 module can adopt patch welding, or weld 2.54mm-spacing pin header, and directly insert it onto user's PCB. The module totally has nine pins and one RF antenna pedestal ANT1, and their definitions are as shown in the table below:

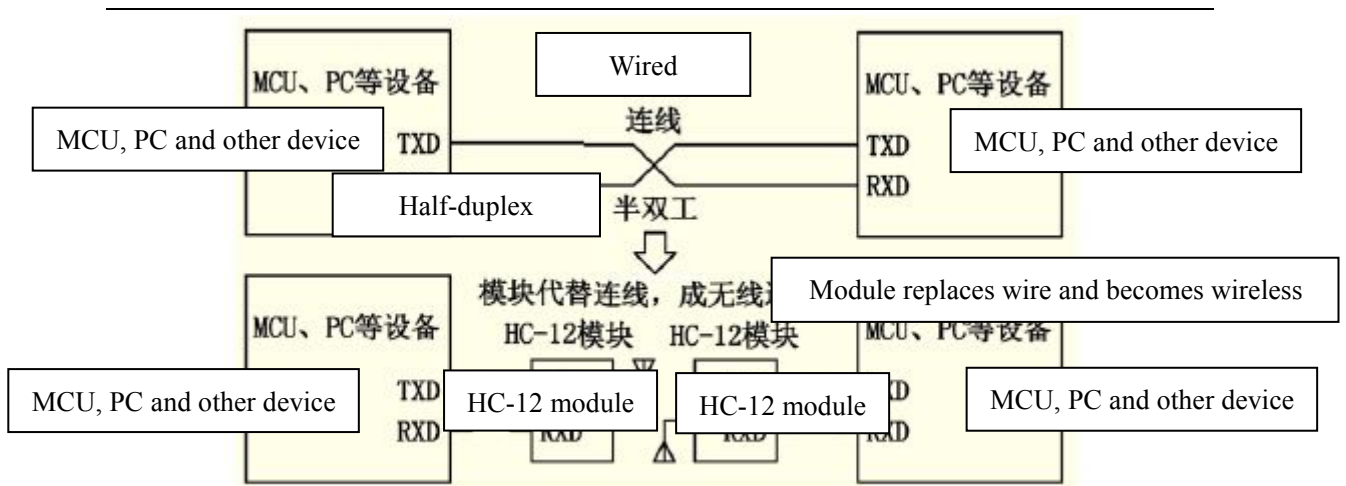
Pin	Definition	I/O direction	Note
1	VCC		Power supply input, DC3.2V-5.5V, with load capacity not less than 200mA. (Note: If the module is working in transmitting state for a long time, it is suggested that one 1N4007 diode should be connected in series when the power voltage is greater than 4.5V, to avoid heating of built-in LDO of module.)
2	GND		Common ground

3	RXD	Input, weak pull-up	URAT input port, TTL level; 1k resistance has been connected in series inside
4	TXD	Output	URAT output port, TTL level; 1k resistance has been connected in series inside
5	SET	Input, internal 10k pull-up resistance	Parameter setting control pin, valid for low level; 1k resistance has been connected in series inside
6	ANT	Input/output	433MHz antenna pin
7	GND		Common ground
8	GND		Common ground
9	NC		No connection, used in fixing, compatible with HC-11 module pin position
ANT1	ANT	Input/output	IPEX20279-001E-03 antenna socket
ANT2	ANT	Input/output	433MHz spring antenna solder eye

Pins 1-6 have two bonding pads respectively, and the outer half-hole bonding pad is used in patch welding. When the inner bonding pad ANT2 of Pin 6 is used in patch welding, the spring antenna can be welded with hands. The inner round-hole bonding pads of Pins 1-5 are used to weld 2.54mm-spacing pin header, and can be directly inserted onto user's PCB socket.

Wireless serial port transparent transmission

(1) Simple introduction of working principle



As shown in the above figure, HC-12 module is used in physical wiring when replacing half duplex communication. The left device sends serial port data to module, and after RXD port of left module receives the serial port data, it will automatically send the data into the air via radio wave. The right module can automatically receive the data, and restore, from TXD, the serial port data originally sent by the left device. It is the same from right to left. Only half duplex state is available between modules, and they cannot receive and send data at the same time.

(2) Serial port transparent transmission

HC-12 module has three serial port transparent transmission modes, expressed with FU1, FU2 and FU3 respectively. In the use, all modes are only responsible for receiving and sending serial port data rather than wireless transmission. The default working mode of system is in FU3 full-speed mode, and in this mode, the baud rate in this air can be automatically adjusted according to baud rate of serial port, and the communication distance will be the farthest at the low baud rate. Different modes cannot transmit data to each other, and user could select the optimal mode according to practical circumstances.

The modules are usually used in pairs, and transmit data by means of half duplex. Meanwhile, the transparent transmission mode, serial port baud rate, and wireless communication channel of two paired modules shall be set to be the same. The default setting is FU3, 9,600bps (8-dibit data, no check, one stop bit), CH001 (433.4MHz).

Use the number of bytes continuously sent to serial port of module unlimited to one time. However, considering ambient interference and other factors, if thousands of data size is sent continuously at a time, some number of bytes may be lost. Therefore, the upper computer shall have response and resending mechanism, to avoid information loss.

(3) Three serial port transparent transmission modes

When HC-12 module leaves the factory, its default serial port transparent transmission mode is FU3. Then the module is in full-speed state, and the idle current is about

16mA. In this mode, the module can automatically adjust the baud rate of wireless transmission in the air according to serial port baud rate, and the corresponding relationship is as shown in the table below:

Serial port baud rate	1,200bps	2,400bps	4,800bps	9,600bps	19,200bps	38,400bps	57,600bps	115,200bps
Baud rate in the air	5,000bps		15,000bps		58,000bps		236,000bps	

To get the farthest communication distance, the serial port baud rate can be set to be low. For short-time transmission of mass data, set the serial port baud rate to be high, but the communication distance will be reduced accordingly.

The receiving sensitivity of module at different baud rates in the air is as shown in the table below:

Baud rate in the air	5,000bps	15,000bps	58,000bps	236,000bps
Wireless receiving sensitivity	-117dBm	-112dBm	-107dBm	-100dBm

Generally, every time the receiving sensitivity is reduced by 6dB, the communication distance will be reduced by half.

When “SET” pin of module is at low level, the serial port transparent transmission mode can be set through AT command (see the introduction in the following chapter for details).

FU1 mode is relatively power saving mode, and the idle working current of this mode is about 3.6mA. In this mode, the module can also set eight types of serial port baud rate as shown in the above table, but the baud rate in the air is uniform, 250,000bps.

FU2 mode is power saving mode, and the idle working current of this mode is about 80μA. In this mode, the module only supports baud rates of 1,200bps, 2,400bps and 4,800bps, and the baud rate in the air is uniform, 250,000bps. If the module is set to be other serial port baud rate, the module cannot conduct communication normally.

Meanwhile, when the module is set to be FU2 mode in FU1 and FU3 mode, the baud rate exceeding 4,800bps will be automatically reduced to be 4,800bps. In FU2 mode, the sending time interval of data package cannot be too short; otherwise, the data will be lost. It is suggested that the sending time interval of data package should not be less than 1sec.

The following gives some characteristics reference values of various modes:

Mode	FU1	FU2	FU3	Remark
Idle current	3.6mA	80 μ A	16mA	Average value
Transmission time delay	15-25mS	500mS	4-80mS	Sending one byte
Loopback test time delay 1	31mS			Serial port baud rate 9,600, sending one byte
Loopback test time delay 2	31mS			Serial port baud rate 9,600, sending ten bytes

Note: Loopback test time delay means the duration from the time of, after conducting short circuit on TX and RX pins of one module and sending serial port data to the other module, starting to send serial port data to the other module to the time that the returned data appear at TX pin of the other module.

Module Parameter Setting AT Command

AT command is used to set the module parameters and switch the module functions, and after setting, it will be valid only after exiting from setting state. Meanwhile, modification of parameters and functions will not be lost in case of power failure.

(1) Command mode entering

The first way to enter: in normal use (energized), put Pin 5 “SET” in low level;

The second way to enter: disconnect power supply, first put Pin 5 “Set” in low level, and then energize it;

Either of the above two ways can make the module enter AT command mode; release it (not put pin “SET” in low level), and exit from the command mode. If the module function is changed after exiting from command mode, it will be switched to corresponding functional status.

In the second way, the module enters AT in the serial port format of 9,600, N, 1 constantly.

(2) Command instruction

①. AT

Test command.

e.g.:

Send “AT” command to module, and the module returns “OK”.

②AT+Bxxxx

Change the serial port baud rate. The baud rate can be set to be 1,200bps, 2,400bps, 4,800bps, 9,600bps, 19,200bps, 38,400bps, 57,600bps, and 115,200bps. The default value is 9,600bps.

e.g.: To set serial port baud rate of module to be 19,200bps, first send “AT+B19200” command to module, and the module returns “OK+B19200”.

③AT+Cxxxx

Change wireless communication channel, optional from 001 to 127 (for the wireless channel exceeding 100, the communication distance cannot be ensured). The default value of wireless channel is 001, and the working frequency is 433.4MHz. The channel stepping is 400KHz, and the working frequency of Channel 100 is 473.0MHz.

e.g.:

To set the module to work at Channel 21, first send “AT+C021” command to the module, and the module returns “COK+C021”. After exiting from the command mode, the module will work at Channel 21, and the working frequency is 441.4MHz.

Note: As the wireless receiving sensitivity of HC-12 module is relatively high, when the serial port baud rate is greater than 9,600bps, five adjacent channels shall be staggered to use. When the serial port baud rate is not greater than 9,600bps, in short-distance (within 10m) communication, also five adjacent channels shall be staggered to use.

④AT+FUx

Change serial port transparent transmission mode of module and three modes are available, namely, FU1, FU2 and FU3. The default mode of module is FU3, and only when serial port transparent transmission mode of two modules is set to be the same, can normal communication be available. For detailed introduction, please see the above “wireless serial port transparent transmission”.

e.g.:

Send “AT+FU1” to module, and the module returns “AT+OK”.

⑤AT+Px

Set transmitting power of module, x is optional from 1 to 8, and the corresponding transmitting power of module is as shown below:

x value	1	2	3	4	5	6	7	8
Transmitting power of module (dBm)	-1	2	5	8	11	14	17	20

The default value is 8, and the higher the transmitting power is, the farther the communication distance is. When the transmitting power level is set to be 1, the transmitting power is the minimum. Generally speaking, every time the transmitting power is reduced by 6dB, the communication distance will be reduced by half.

e.g.:

Send “AT+P5” command to module, and the module returns “OK+P5”. After exiting from the command code, the transmitting power of module is +11dBm.

⑥AT+Ry

Obtain single parameter of module, y is any letter among B, C, F and P, respectively representing: baud rate, communication channel, serial port transparent transmission mode, and transmitting power.

Example 1:

Send “AT+RB” to module, and if the module returns “OK+B9600”, it is inquired that the serial port baud rate of module is 9,600bps.

Example 2:

Send “AT+RC” command to module, and if the module returns “OK+RC001”, it is inquired that the communication channel of module is 001.

Example 3:

Send “AT+RF” command to module, and if the module returns “OK+FU3”, it is inquired that the module is working in serial port transparent transmission mode 3.

Example 4:

Send “AT+RP” command to module, and if the module returns “OK+RP: +20dBm”, it is inquired that the transmitting power of module is +20dBm.

⑦AT+RX

Obtain all parameters of module. Return serial port transparent transmission mode, serial port baud rate, communication channel, and transmitting power in order.

e.g.:

Send “AT+RX” command to module, and the module returns “OK+FU3\r\n OK+B9600\r\n OK+C001\r\n OK+RP: +20dBm\r\n”. (“\r\n” means return\nnewline)

⑧AT+Uxxx

Set data bits, check bit and stop bit of serial port communication. For check bit, N means no check, O means odd check, and E means even check. For stop bit, 1 means one stop bit, 2 means two stop bits, and 3 means 1.5 stop bits.

e.g.:

To send serial port format to be eight data bits, odd check, and one stop bit, please Send “AT+U8O1” to module, and the module returns “OK+U8O1”.

⑨AT+V

Inquire firmware version information of module.

e.g.:

Send “AT+V” command to module, and the module returns “HC-12_V1.1”.

⑩AT+SLEEP

After receiving the command, the module enters sleep mode after exiting from AT, the working current is about 22 μ A, and this mode doesn't allow serial port data transmission. Then enter AT setting state again, and the module will exit from sleep mode automatically.

e.g.:

When wireless data transmission is not needed, to save power, send "AT+SLEEP" command to module, and the module returns "OK+SLEEP".

(1)AT+DEFAULT

Set serial port baud rate, communication channel, and serial port transparent transmission mode to be default value.

e.g.:

Send "AT+DEFAULT" to module, and the module returns "OK+DEFAULT", and the default value is restored. The serial port baud rate is 9,600bps, communication channel is C001, and serial port transparent transmission mode is FU3.

(2)AT+UPDATE

Put the module in the status of waiting for software update.

After sending the command, the module will not respond to command any more, until it is re-energized.

After sending the command, please close the serial port assistant, and turn on HC-1X updater to update the software. For detailed operating method, please refer to the following "software update" introduction.