

# A Multi Core Data Transmission and Management Method of the Wireless MIMO Network

Changjian Deng (✉ [chenglidcj@cuit.edu.cn](mailto:chenglidcj@cuit.edu.cn))

Chengdu University of Information Technology <https://orcid.org/0000-0002-6471-8623>

---

## Research Article

**Keywords:** MIMO, multi core, wireless networks, data transmission and management

**Posted Date:** June 4th, 2021

**DOI:** <https://doi.org/10.21203/rs.3.rs-486996/v1>

**License:**  This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

---

# A multi core data transmission and management method of the wireless MIMO network

Changjian Deng

Received: date / Accepted: date

**Abstract** With the development of vehicle and mobile robot wireless technology, there are lots of data may transmit simultaneously. So, the paper proposes a multi core data transmission and network management method. In this method, the MIMO multi core system works in parallel mode; and it has multi agent feature in network operation, meanwhile it has feature of sub-graph decomposition characteristics also in MIMO topology of network. Based on the mathematical model of MIMO multi core system, the paper designs two prototype of network gateway, one is made of five CC2530 , a CAN module, and it is a demo of a vehicles gateway; the another is CC1320 multicore gateway system. The simulations and experiments show the feasibility and effective of this wireless multi core data transmission and management method.

**Keywords** MIMO · multi core · wireless networks · data transmission and management

## 1 Introduction

In many applications, the embed communication and data processing are very important. There are two main method, one is using more powerful chip, the another is using multi core and MIMO technology.

For example, the FANET (Kakamoukas, 2020) allows the collaboration and co-ordination of multiple UAVs, then it promotes that the system has more capabilities. For example, The FANET can be deployed in six different applications: Crop Scouting, Crop Surveying and Mapping, Crop Insurance, Cultivation Planning and Management, Application of Chemicals, and Geofencing. In every UAV, there are many communication tasks: the remote control task, the Graphical transmission, the FANET, and so on. It is a MIMO communication system.

---

Changjian Deng  
Chengdu, China  
Tel.: +13551257136  
Fax: +85966915  
E-mail: chenglidcj@cuit.edu.cn

In the communication architectures of FANET, the (Emadi, 2020) concludes three different kinds of communication architectures: 1) the single UAV Ad-hoc network; 2) Multi-Group UAV Ad hoc Network; 3) Multi-Layer UAV Ad hoc Network. Their tasks are more complicate and uncertain. The need of using powerful chirp or using multi core technology are very urgent. Their performance parameters are average end-to-end delay, throughput, packet delivery ratio (Santiago, 2018), and routing overhead (Leonov et al., 2018), (?). And these parameters can be used to analyze the effective of routing protocols (for example: OLSR, DSDV, DSR, TORA, ZRP) and a task schedule (Chowdhuri, 2018), (Fayez et al., 2018), (Fayez et al., 2018).

In Mobile Robotic Networks, there are more and more coordinated tasks, and they need reliable communications (LeNy, 2013). There are many internal attacks, they all need evaluate and detect. The need of using powerful chirp and using multi core technology are also urgent (Barannik, 2018). (Liu, 2021) proposes learning control schemes in mobile robot networks. (Petitti, et al., 2018) shows a distributed map building approach, the complicates of communication and tasks are common in mobile robotic networks. (Boldrer, 2021) shows a demo of multi task using multi agents and graph topology based on Minimum Spanning Tree (MST).

In VANET, (Moni, 2021) shows that VANETs may find parking places, updating inbuilt vehicle navigation systems, and so on (Peng, 2018). There are some comparative studies of the following VANET simulators: SUMO, NS2, OMENET, and so on (Ying, 2017), (Manvi, 2017). Eze et al. Eze et al. discuss the security of communicating vehicles (Eckhoff, 2018). The need of using powerful chirp or using multi core technology (Peng, 2018), and MIMO communication are also urgent.

Although there are some researches on multi core or MIMO system, the system research of multi core combining with many SISO to form a MIMO system is less. The paper focuses on a wireless multi core data transmission and management method of MIMO network. The contributions of paper are 1) construct the mathematic and technology model of this MIMO multi core system; 2) designs two prototype of network gateway, one is five CC2530 and CAN module to present a vehicle gateway demo, the other is CC1320 series multicore system.

The rest of paper is arranged as below: Section II is methodology of MIMO multi core system. Section III is simulations and experiments. The Section IV is conclusion.

## 2 Methodology

### 2.1 Mathematical or engineer model

In the context of connectivity of FANET, MANET or VANET, some new theories are used for analysis the graph related problem, for example, Menger's Theorem and Connectivity, identification of cyber communities, and so on. There are some concepts:

**Node degree:** the number of edges that enter or exit a node. In unidirectional networks, the sum of incident and outgoing edges is called node degree.

**Network diameter:** the longest distance between any two nodes in the network, that is, the maximum number of paths.

Bisection width: the minimum number of edges that must be removed for each half of a bisection network.

Bisection bandwidth: the maximum information bit (or byte) passing through all connections on the smallest bisection plane in a second.

### 2.1.1 Parallel operation model of MIMO multi core system

In order to operate parallelly, the design of this system considers the following characteristics:

- (1) Separating the work into discrete parts;
- (2) Execute multiple program instructions at any time and in time;
- (3) The time-consuming of solving problems with single computing resource firstly.

Like parrel computing, it can be divided into data parallel and task parallel. Here we use MIMD (Multiple Instruction Stream Multiple Data Stream). MIMD machines can be divided into five common categories: parallel vector processor (PVP), symmetric multiprocessor (SMP), massively parallel processor (MPP), workstation cluster (cow), distributed shared memory processor (DSM).

To comparing our method with single core network, we use Gustafson's law. It is proposed on the basis of Amdahl's law. It defines the acceleration ratio

Speedup ratio = performance before improvement / performance after improvement

=The time of executing a task before.. / the time of executing a task after adopting improves

$$W' = n * W_{serial-part} + (1 - N) * p * W_{parallel-part}$$

Where W is the time to complete the task when p processors are serial + parallel, W ' is the time  $T_1$  for one processor to complete the task, P is processor acceleration factors

$$S = T1/T2 = W'/W = n + (1 - n) * p$$

N is the serial percentage task.

### 2.1.2 multi agent wireless network

This wireless sensor network is designed to deploy in MANET, VANET , FANET, mobile robot system, and so on. In these systems, the multi-agent system shows its advantage though its communication, cooperation, mutual solution, coordination, scheduling, management and control among the agents. Especially it has strong robustness and reliability of learning and reasoning .

Method A):slicing algorithm

As mentioned above, we need to divide these data according to certain dimensions. According to the requirements, user ID is the best. Since there is no state association between users, there is no need for transactional and secondary iterative calculation. We use a simple hash module to divide ID.

$$f(memberid)\%5 = ServerN$$

Method B):message queuing

This approach introduces a third party, message queuing. It first uses a separate program to push the user information to the message queue, and then each machine cancels the queue. So there are three roles:

1. Push message, referred to as master.
2. Message queue, take rabbit mq as an example.
3. Each handler, referred to as worker or slave

Method C): distributed computing

1. Apache storm is a stream processing framework that focuses on very low latency and is the best choice for workloads that require near real-time processing. This technology can handle a very large amount of data and provide results with very low latency.
2. MapReduce's computing model is very simple, its idea is "divide and conquer", mapper is responsible for "divide", that is, the complex large task is divided into several small tasks to process, there is no dependency between each other, so that it can be distributed to multiple computing nodes to achieve a high degree of parallel computing capability; reducer is responsible for the summary and output of the results of the map phase.

### 2.1.3 sub-graph decomposition method

In physics and logic layer, we should find the minimal path cost, task and time consuming to undertake transmitting and store work. For example, we should find the minimal hop or distance of transforming, the minimal transmitting cost, and these optimizations based on the topology of network. The foundation to solve these problems mainly come from graph theory:

1. connected graph is based on the concept of connectivity. In an undirected graph  $G$ , if there is a path from vertex  $I$  to vertex  $J$  connected, then  $I$  and  $J$  are said to be connected. If  $G$  is a digraph, then all edges in the path connecting  $I$  and  $J$  must be in the same direction. If any two points in a graph are connected, then a graph is called a connected graph. If this graph is a directed graph, it is called a strongly connected graph.
2. Connected component: a polar connected subgraph of an undirected graph  $G$  is called a connected component (or connected branch) of  $G$ . A connected graph has only one connected component, that is, itself; an unconnected undirected graph has multiple connected components.
3. Strongly connected graph: a directed graph  $G = (V, e)$  is said to be strongly connected if there are paths from  $X$  to  $y$  and from  $y$  to  $X$  for any two different vertices  $X$  and  $Y$  in  $V$ . Accordingly, we have the concept of strongly connected components. A strongly connected graph has only one strongly connected component, which is itself; a non strongly connected digraph has multiple strongly connected components.
4. Unidirectional connected graph: Let  $G = \langle V, E \rangle$  be a directed graph. If  $u \rightarrow V$  means that  $G$  contains at most one simple path from  $u$  to  $V$ , then  $G$  is simply connected.

5. Weakly connected graph: replace all directed edges of a directed graph with undirected edges, and the resulting graph is called the base graph of the original graph. If the base graph of a digraph is connected, then the digraph is weakly connected.

For example, if the adjacency table is used to store the information of the net, then a field for storing the weights should be added to the node. The number of nodes in the single chain table of each vertex is the outgoing degree of the vertex. Whether it is storage graph or network, it is necessary to set a header node in front of each single linked list. The first field data of these header nodes is used to store the number I of node VI, and the second field firstarc is used to point to the first node in the linked list. Path length is the number of edges or arcs on the path. If the first vertex is the same as the last vertex, then the path is a loop. The vertices in the path do not appear repeatedly, the path is called a simple path.

**Definition 1** Menger's Theorem: (vertex version) Let  $x$  and  $y$  be vertices of a graph  $G$  and  $xy \notin G$ . Then the minimum size of an  $x$ - $y$  cut equals the maximum number of pairwise internally disjoint  $x$ - $y$  paths.

**Definition 2** Menger's Theorem: (edge version) Let  $x$  and  $y$  be vertices of a graph  $G$ . Then the minimum size of an  $x$ ,  $y$ -disconnecting set of edges equals the maximum number of pairwise edge-disjoint  $x$ - $y$  paths. Then the minimum size of an  $x$ - $y$  cut equals the maximum number of pairwise internally disjoint  $x$ - $y$  paths.

**Definition 3** A flow assigns a value to each edge  $e$ . We use  $f_e$  to denote the total flow on edges leaving a vertex and  $g_e$  to denote the total flow on edges entering a vertex. A flow is feasible if it satisfies the capacity constraints for each edge and the conservation constraints for each node.

**Definition 4** In a network a source / sink cut is a partition of the edge set into one set containing the source, but not the sink, and the remaining edges form the set.

**Theorem 1** In a network a source / sink cut is a partition of the edge set into one set containing the source, but not the sink, and the remaining edges form the set.

#### 2.1.4 some estimation models to estimate parameter of network

1. Scalability: one of the main challenges faced when designing an optimal routing protocol is the adaptation of scalability, where changing number of nodes has a strong effect on adaptation to topology change and load coming from packets.
2. Speed: all routing protocols have their unique features and most of them suffer when there is a rapid speed change.
3. Mobility: routing protocols depend on the mobility as well, obtaining a high mobility feature in routing protocols is a challenge.
4. Delivery Ratio: The ratio of number of messages that have been successfully delivered to the destination nodes to the number of generated messages from the source nodes. The delivery ratio is defined as

$$\text{DeliveryRatio} = m/n$$

where  $m$  is the number of packets received, and  $n$  is the number of packets sent.

5. Delivery Latency: The average time duration from message generation unto the successful message reception. It is defined as

$$DeliveryLatency = \sum_{i=1}^m T_i$$

where  $T_i$  is information transmission delay. 3) Overhead Ratio: The ratio of the total number of messages created by source nodes to the total number of messages forwarded by all nodes. It is calculated by

$$OverheadRatio = (h - \sum_{i=1}^m) / (\sum_{i=1}^m mF_i)$$

where  $h$  is the number of packets, and  $F_i$  is the number of successfully packets.

Since the symmetric kernel matrix of MIMO nonlinear system is uniquely determined, we can identify the nonlinear system by identifying the kernel matrix. There are two main method.

- 1) Random response method

For the  $n$ -order homogeneous Volterra nonlinear system, a group of uncorrelated real stationary Gaussian white noise with mean value of zero and intensity of  $a$  is used as the input signal of the system. By measuring its output signal, the  $n$ -order Volterra kernel can be calculated.

Theorem: suppose that the input signal of  $n$ -order homogeneous MIMO Volterra nonlinear system is a group of uncorrelated real stationary Gaussian white noise with zero mean and intensity  $a$ , then the  $n$ -order Volterra kernel matrix of the system can be expressed as:

$$H_n(t_1, t_2, \dots, t_n) = (h_{n,i,j}(t_1, t_2, \dots, t_n))_{i=n, j=m^2} \quad (1)$$

Where  $I$  and  $j$  are row and column marks of matrix respectively, The elements of Volterra kernel matrix are not equal to each other.

- 2) Impulse response method

For the  $n$ -order homogeneous Volterra nonlinear system, suppose that the  $M$  input signal  $x(t)$  is  $x_i(t) = \delta(t)$ ,  $x_j(T) = 0, j \neq i$ , then

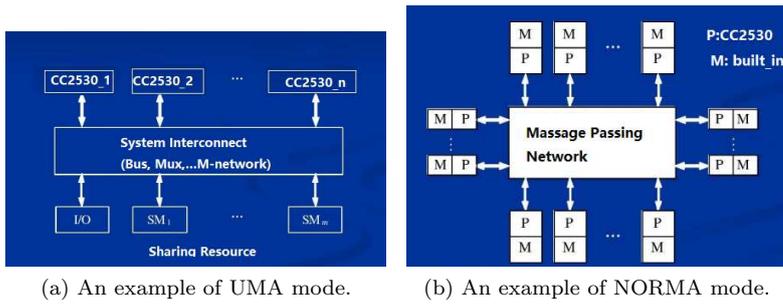
$$y_i(t) = h_{n,j,i-1,(m^{n-1}+\dots+m+1)}(t, t, \dots, t) \quad (2)$$

Suppose that the input signal  $x(T)$  may have two, three or up to  $m$  component, the component is  $\delta(t)$ , and the other components are 0, then  $h_{n,j,i}(t, t, \dots, t)$  can be calculated. By taking  $m$  input signal,  $x(t)$  is

$$x_i(t) = \delta(t) + \delta(t - T_{i1}) + \dots + \delta(t - T_{ip}) \quad (3)$$

Then get  $h_{n,j,i}(t_1, t_2, \dots, t_n)$ . If for a general  $n$ -order Volterra nonlinear system, the input signal  $x(T)$  can be taken as

$$x_i(t) = \sum_{j=1}^{pi} k_i \delta(t - T_j) \quad (4)$$



**Fig. 1** Comparing two different mode Example.

Where  $I, p_i = 1, 2, \dots, m$ ;  $k_j$  is a non negative integer;  $T_j$  is a non negative number which is not equal to each other. By adjusting  $k_j$ , their Volterra kernel matrix can be identified

Note: this method has theoretical value. In practical application, it is difficult to measure because the output signal is transient response

## 2.2 methodology of technology

### 2.2.1 parrell mode design

In the technique design, paper uses SPI or IIC in-board bus to connect all processors. And NO-Remote Memory Access(NORMA) mode and UMA mode is analyzed.

### 2.2.2 multi agent networking design

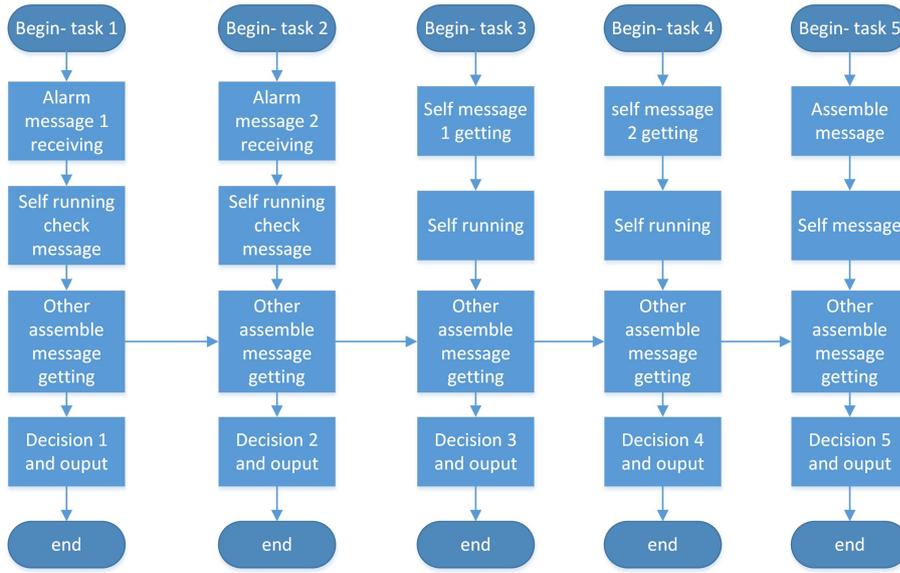
As multi core system, distributed control and measurement system is suitable, and for its autonomy and self-organization properties, multi agent system should be designed.

Multi agent system should owing to the below rules:

- (1) design every independence and autonomy agent.
- (2) distributed application design, consider its modularity, easy expansibility, flexible and simple design.
- (3) constructs multi-level and diversified agents according to the object-oriented method
- (4) uses information and construction integration technology to integrate a MAS subsystem.

To simplify this problem, the first way is computing task splitting. If the data can be split horizontally and distributed to five agents, each agent will only calculate one fifth of its own data.

As mentioned above, we need to divide these data according to certain dimensions. According to the requirements, user ID is the best. Since there is no state association between users, there is no need for transactional and secondary iterative calculation. We use a simple hash module to divide ID.



(a) MIMO scheduling design.

**Fig. 2** MIMO scheduling design.

The general peer-to-peer intelligent agents approach permits to model systems that represent environment in which independent autonomous elements have to communicate and collaborate in order to achieve the desired results:

### 2.2.3 MIMO task scheduling design

The MIMO scheduling design is shown in Fig.2.

All the tasks are dispatched to different nodes. For example, in the multi mobile robot system, their motion is defined by the control light, other predefine path, their moving information, system information, and circumstance information.

In our demo system, every different task is undertaken by same kind of core (for example, CC2530), five different tasks use five different cores. All the cores run their own program .

Some task should be cooperated , for example, the clock time-synchronization , the memory or test data are in share. Normal time-synchronizition method is a beacon in the bus or a hardware interrupt signal. The memory or test data sharing normal use sharing memory method.

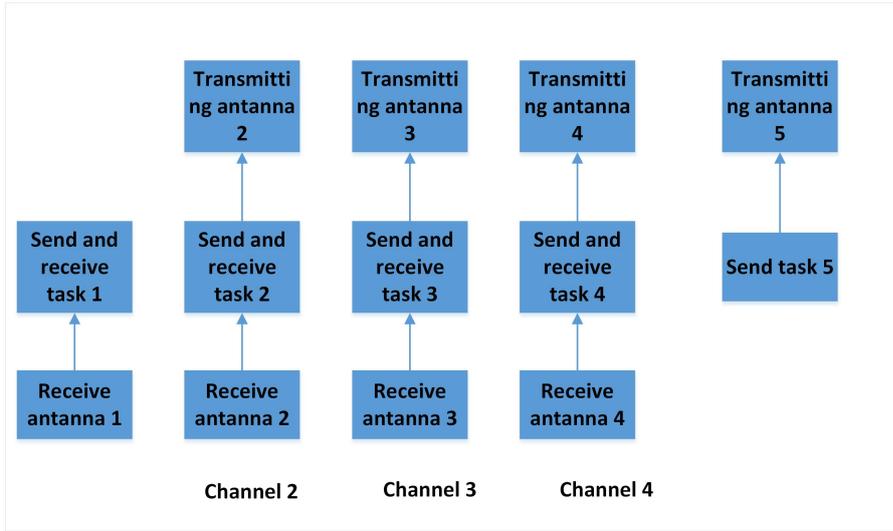
### 2.2.4 Multi source routing method in MIMO system

The MIMO communication system is shown in Fig.3.

In this system, there are two different part, one is using different channel ,so, there are smaller interface to another task. the another is every task having their different network forming method.

The multi core data transmission arithmetic is shown table.1

The multi core data store arithmetic is shown table.2



(a) MIMO communication system

**Fig. 3** MIMO communication system.**Table 1** The multi core data transmission arithmetic.

- 
- step 1) every task dispatch one different channel
  - step 2) one kind of sending or receiving task is undertaken by one kind of cell
  - step 3) one kind of cell form one kind of networks
  - step 4) the multi core undertake their task parrell
- 

**Table 2** The multi core data store arithmetic.

- 
- step 1) every kind of parameter dispatch memory cell array in one kind of core
  - step 2) system parameter dispatch memory cell in bottom broad
  - step 3) every core use IIC bus or other bus getting system parameter value
  - step 4) the memory cell use FIFO update the memory
- 

### 3 Simulation and Experiment

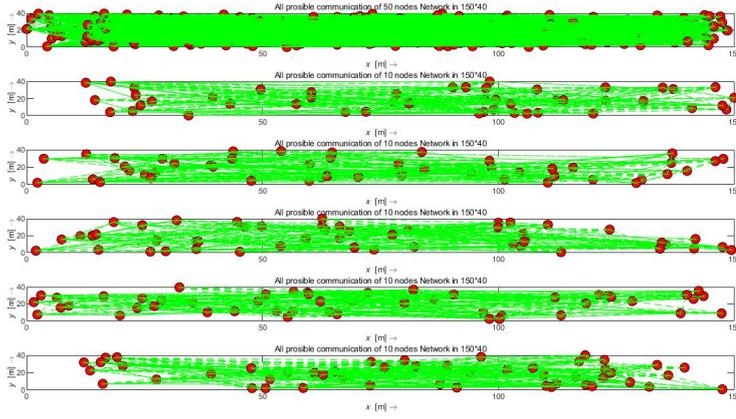
#### 3.1 Simulation

##### 3.1.1 communication resources comparison

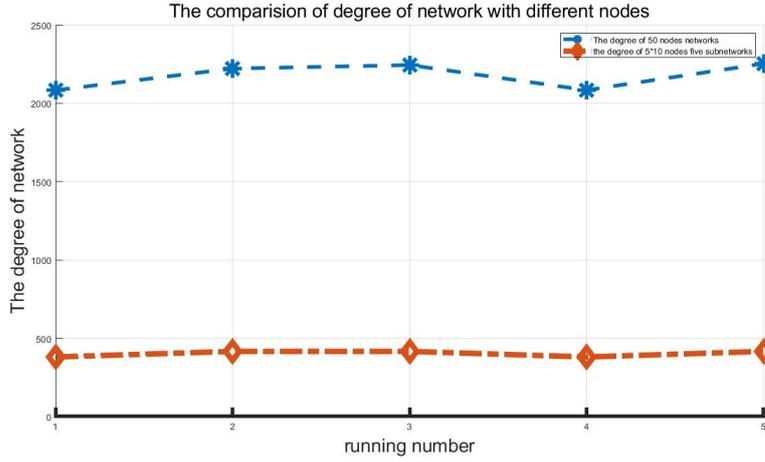
In 40\*150m area, there are two different method. First, we use multi core MIMO communication in the paper, the another we discuss the normal communication.

Firstly, we use 50 nodes random that are deployed in this area. Without general, no matter what method we used, the resources that be used in one same network are the function of the connected nodes(or the degree of network).

$$Resources_{50network} = F(Degree_{50}) \quad (5)$$



(a) the connection status of different nodes random network

**Fig. 4** the connection status of random network.

(a) The comparison of the degrees of random network

**Fig. 5** The comparison of the degrees of random network .

So, there are different when we use five subnetworks to replace one 50 nodes network(the paper used).

$$Resources_{5subnetworks \times 10nodes} = 5 \times F(Degree_{10}) \quad (6)$$

As we simulations, in a random network , the communication resources multi core MIMO communication occupied is 4-5 times than the communication resources that normal one networks used.

The comparison of the connection status of random network is shown in Fig.4. The comparison of the degrees of random network is shown in Fig.5.

**Table 3** The communication parameters of 50 nodes network.

parameters	multi core network	single network
The averal transmitting delay	12*2 ms	12*2+12*2+40*2
The maximum transmitting thoughout of every node	3125 bps	625bps
The robust and stability	t	t/5

**Table 4** The communication interface tests

parameters	2 in different channel	4 in different channe
Receive rate	1.95-2.01	1.90-2.00
The erroe rate	less than 0.01	less than 0.01

### 3.1.2 The communication performance comparision

In application,the averal transmitting delay, max transmitting thoughout, the robust and stability are mainly concerned parameters.

The simualtion network is same as above. For the transmitting delay limit of some application is small, so here set the maximum number of hop is two.

The averal tranmitting delay is proportionate to the data processing time  $t_{dp}$  in the every node, is proportionate to data transmitting time (include retransmitting)  $t_{st}$  in the connect edge in the network, and is proportionate to time of forming  $t_{nf}$  and managing network  $t_{nm}$ .The data processing time ,transmitting time is almost same in every cole. the network forming and managing time is proportionate to the degree of network  $De$ .if the node number is n,the hop number is m, then,

$$T_{delay} \simeq m \times t_{dp} + t_{st} * De/n + (t_{nf} + t_{nm}) * De/n \quad (7)$$

The maximum transmitting thoughout  $Thom$  is inversely proportional to nodes number in the networks.If the transmitting rate is R,then,

$$Thou_{max} \sim R/n \quad (8)$$

and it is reasonable the robust and stability is inversely proportional to nodes number or the degree of network.Comparing the multi core MIMO communciation with ordinary network is shown in Table 3.

Proposed the transmitting data is 40bytes in maximum delay time caculation,the transmitting speed is 250k bps, the maximum hop is two, every hop is half of number, the transmitted data processing time is almost  $2 * 40 * 8 / 250 = 2.56$  ms ,the whole data processing time is two times of the transmitted data processing time is almost 6ms.

The simulation of vehicle wireless work is shown in Fig.6 .

## 3.2 Experiment

### 1) Communication test

First communciation test is the receive rate with two to four nodes in different channel when a node is transmitting to coordinator. The software test form is designed by VC++. It is shown in Fig.7.

```

In [2]: cd ~/omnetpp/engines/veins-veins-5.0
~/omnetpp/engines/veins-veins-5.0
~/omnetpp/engines/veins-veins-5.0$ cd example/
~/omnetpp/engines/veins-veins-5.0/example$ cd veins-veins-5.0/
~/omnetpp/engines/veins-veins-5.0/example/veins-veins-5.0$ cd examples/
~/omnetpp/engines/veins-veins-5.0/example/veins-veins-5.0/examples$ cd veins/
~/omnetpp/engines/veins-veins-5.0/example/veins-veins-5.0/examples/veins$ d:/Program/Eclipse/Sumo/bin/sumo.exe -c erlange
n.sumo.cfg
loading configuration ... done.
~/omnetpp/engines/veins-veins-5.0/examples/veins$ cd ..
~/omnetpp/engines/veins-veins-5.0/examples$ cd ..
~/omnetpp/engines/veins-veins-5.0/examples$ cd ..
~/omnetpp/engines/veins-veins-5.0$ dir
configure doc examples images print-veins-version src sumo-launchd.py
COPYING doxy.cfg format-code.sh Makefile README.txt subprojects
~/omnetpp/engines/veins-veins-5.0$ sumo-launchd.py -vv -c D:/Program/Eclipse/Sumo/bin/sumo.exe
logging to c:/users/chenj-1/AppData/Local/Temp/sumo-launchd.log
listening on port 9999
connection from 127.0.0.1 on port 9999
Handling connection from 127.0.0.1 on port 9999
Got Traci message of length 2
Got Traci command of length 1
Got Traci command 0x0
Got Traci command 0x0
Got Traci message of length 302
Got Traci command of length 297
Got Traci command 0x0
Got Traci command 0x0
Got CMD_FILE_SEND for "sumo-launchd_launch.xml"
Got CMD_FILE_SEND with data "c:\launch
  <copy file="erlangen.net.xml"/>
  <copy file="erlangen.rou.xml"/>
  <copy file="erlangen.poly.xml"/>

```

(a) The simulation of vehicle wireless work

**Fig. 6** The simulation of vehicle wireless work.

The paper realizes five cc2530 and one CAN communication gateway to test the performance. Its diagram is shown in Fig.8

The serial communication uses ack to acknowledge the safe receiving the message that is coming from wireless network, for example, the CC2530 chip. Then, the multi core CC2530 operated parallelly. They execute their program instruction independently. They form different networks. Every subnetwork has its own main task.

These MCUs can work in slicing algorithm, message queuing and distributed computing. The vehicle wireless CAN gateway hardware demo is shown in Fig.10.

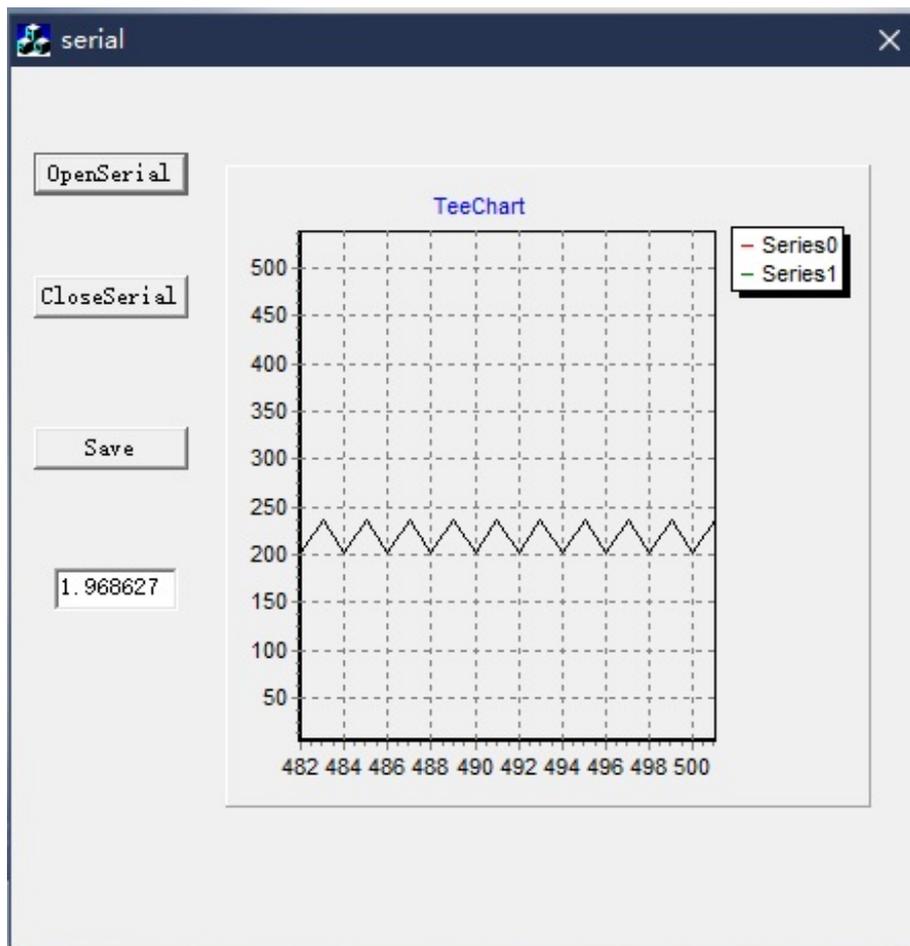
Another CC13xx gateway is shown in Figure.11.

## 4 Conclusion

The paper solves the problem of the large amount of data transmitting in the wireless network. And the paper proposes the multi core and MIMO data transmission and network management method.

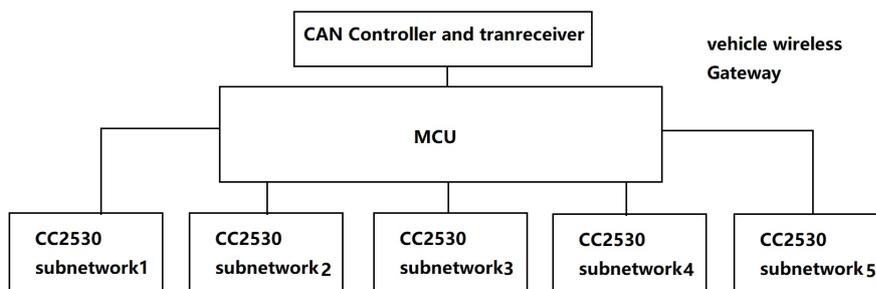
To suit for uncertain applications, the MIMO multi core system works in parallel mode is considered, and to make them work together effectively, the multi agent operation mode is deployed in this system.

Meanwhile, to simplify the topology, the sub-graph decomposition method is used in this MIMO wireless network. Based on the mathematical model of MIMO multi core system, the paper designs two prototype network gateways, one is five CC2530 and CAN modules to present a vehicle gateway demo, the other is CC1320 series multi-core system. The simulations and experiments show the feasibility and effectiveness of this wireless multi core data transmission and management method.



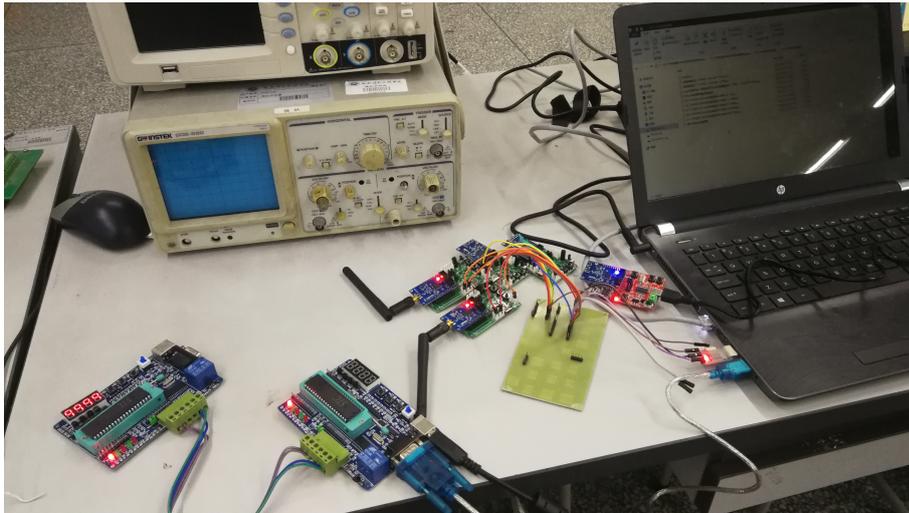
(a) The software test form

Fig. 7 The software test form.



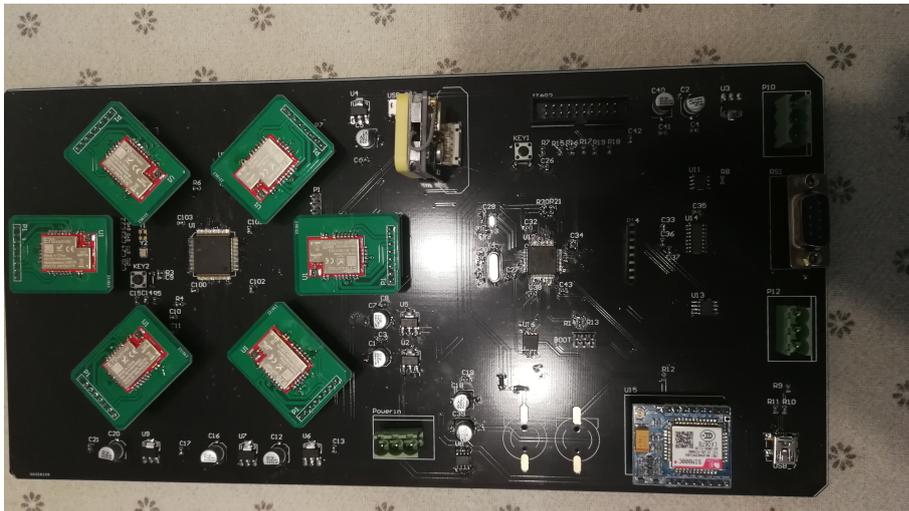
(a) vehicle wireless CAN gateway

Fig. 8 vehicle wireless CAN gateway



(a) The vehicle wireless CAN gateway hardware demo

**Fig. 9** The vehicle wireless CAN gateway hardware demo



(a) CC13xx gateway

**Fig. 10** The CC13xx gateway demo

## 5 Declarations

### 5.1 Funding

The paper is supported by 2019RYJ03 Open fund project of Sichuan Key Laboratory of artificial intelligence.

## 5.2 Conflicts of interest

The paper has no conflicts of interest.

## 5.3 Availability of data and material

Not applicable.

## 5.4 Code availability

Not applicable.

## References

- Kakamoukas, G. A. , Sarigiannidis, P. G. , & Economides, A. A. . (2020). *Fanets in agriculture - a routing protocol survey*. Internet of Things.
- Al-Emadi, Sara & Al-Mohannadi, Aisha. (2020). *Towards Enhancement of Network Communication Architectures and Routing Protocols for FANETs: A Survey*. 10.1109/CommNet49926.2020.9199627.
- A. Garcia-Santiago, J. Castaneda-Camacho, J. F. GuerreroCastellanos, and G. Mino-Aguilar(2018).*Evaluation of AODV and DSDV routing protocols for a FANET: Further results towards robotic vehicle networks*. in 2018 IEEE 9th Latin American Symposium on Circuits Systems (LASCAS), Puerto Vallarta,Mexico, 2018, pp. 1–4.
- A. V. Leonov and G. A. Litvinov, *Considering AODV and OLSR Routing Protocols to Traffic Monitoring Scenario in FANET Formed by Mini-UAVs*. 2018 XIV International Scientific-Technical Conference on Actual Problems of Electronics Instrument Engineering (APEIE), Novosibirsk, Russia, Oct. 2018, pp. 229–237. DOI: 10.1109/APEIE.2018.8545667.
- N. Sharma and S. Ali, *Study of Routing Protocols in MANET-A Review*. in 2019 6th International Conference on Computing for Sustainable Global Development (INDIACom),New Delhi, India, 2019, pp. 1245–1249.
- S. Chowdhuri, S. Banerjee, and S. S. Chaudhuri, *Implementation of a novel Minimum Energy Consume Secure RoutingProtocol for reliable and energy efficient rescue operation of FANET*. in 2018 IEEE Applied Signal Processing Conference (ASPCON), 2018, pp. 129–133.
- F. A. Fayez, M. Hammoudeh, B. Adebisi, and K. N. A.Sattar,Assessing the effectiveness of flying ad hoc networks for international border surveillance,*International Journal of Distributed Sensor Networks*, 1:7, 2019. DOI: 10. 1177/1550147719860406. Available: <https://doi.org/10.1177/1550147719860406>.
- Y. Liu, Z. Liu, J. Shi, G. Wu, and C. Chen, Optimization of Base Location and Patrol Routes for Unmanned Aerial Vehicles in Border Intelligence, Surveillance and Reconnaissance, *Journal of Advanced Transportation*, Jan. 2019. DOI: 10.20944/preprints201809.0487.v1

- Le Ny, Jerome & Pappas, George. (2013). Adaptive Deployment of Mobile Robotic Networks. *Automatic Control, IEEE Transactions on*. 58: 654-666. 10.1109/TAC.2012.2215512.
- Barannik, Elena & Basan, Alexander & Makarevich, Oleg. (2018). *Evaluating and Detecting Internal Attacks in a Mobile Robotic Network*. 516-5168. 10.1109/CyberC.2018.00102.
- Liu, Yen-Chen & Lin, Tsen-Chang & Lin, Mu-Tai. (2021). Indirect/Direct Learning Coverage Control for Wireless Sensor and Mobile Robot Networks. *IEEE Transactions on Control Systems Technology*. PP. 1-16. 10.1109/TCST.2021.3061513.
- Petitti, Antonio & Di Paola, Donato & Colella, Roberto & Milella, Annalisa & Stella, Ettore & Coratelli, Antonio & Naso, David. (2018). *A Distributed Map Building Approach for Mobile Robotic Networks*. 116-121. 10.1109/COASE.2018.8560499.
- Boldrer, Manuel & Bevilacqua, Paolo & Palopoli, Luigi & Fontanelli, Daniele. (2021). Graph Connectivity Control of a Mobile Robot Network With Mixed Dynamic Multi-Tasks. *IEEE Robotics and Automation Letters*. PP. 1-1. 10.1109/LRA.2021.3061072.
- Moni, Shafika & Manivannan, D.. (2021). *Privacy-preserving and Authentication in VANETs*.
- Peng, H., Liang, L., Shen, X., and Li, G. Y. 2018. Vehicular communications: A network layer perspective. *IEEE Transactions on Vehicular Technology*.
- Ying, B. and Nayak, A. 2017. Anonymous and lightweight authentication for secure vehicular networks. *IEEE Transactions on Vehicular Technology*. 66:12, 10626–10636.
- Manvi, S. S. and Tangade, S. 2017. A survey on authentication schemes in VANETs for secured communication. *Vehicular Communications*,9, 19–30
- Eckhoff, D. and Sommer, C. 2018. Readjusting the privacy goals in vehicular ad hoc networks: a safety preserving solution using non overlapping time slotted pseudonym pools. *Computer Communications*.122, 118–128.

# Figures

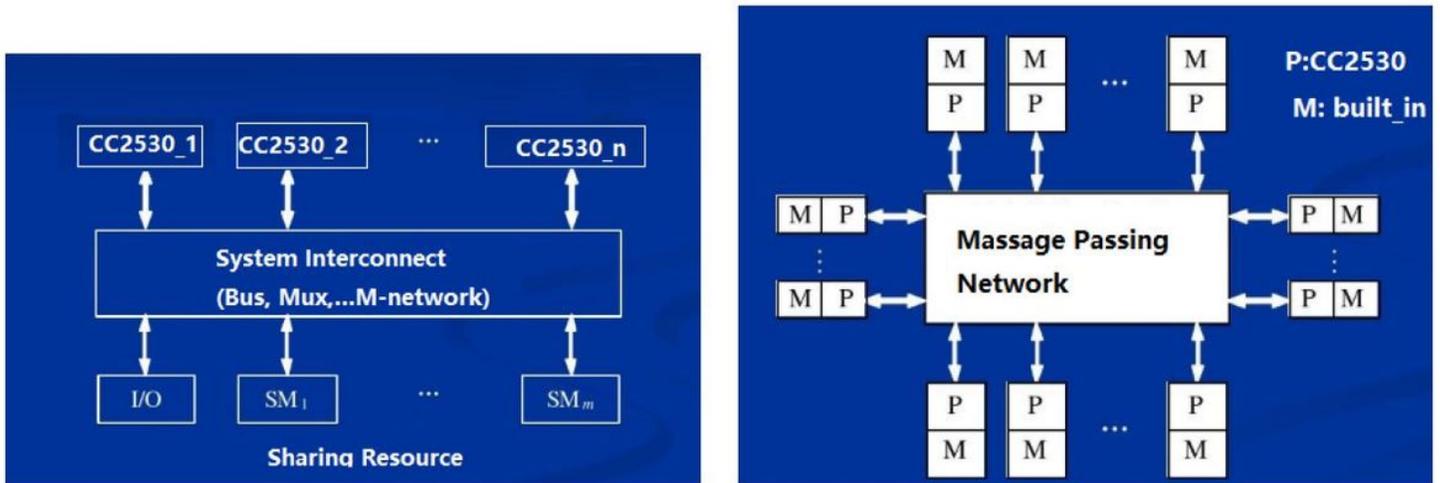


Figure 1

Comparing two different mode Example.

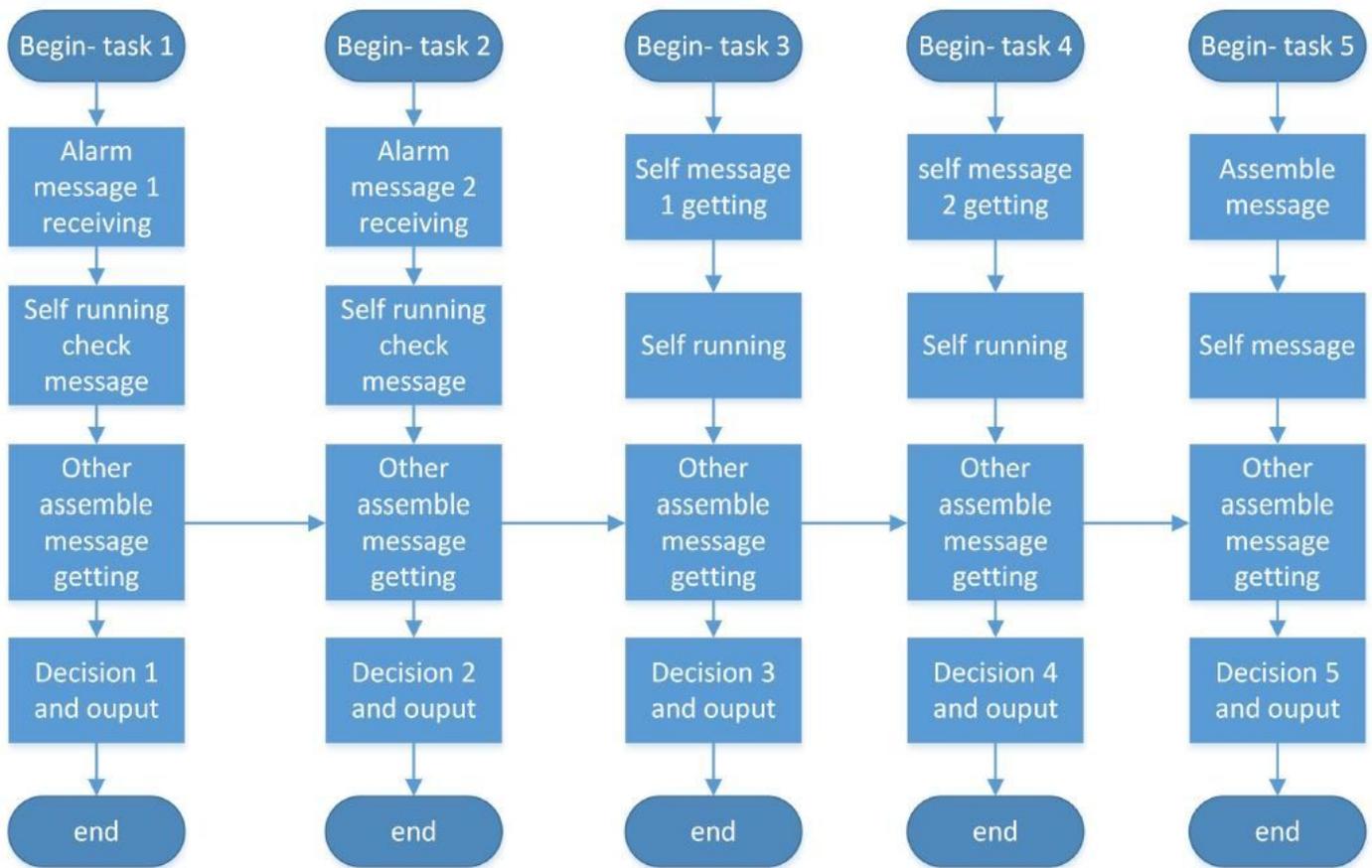
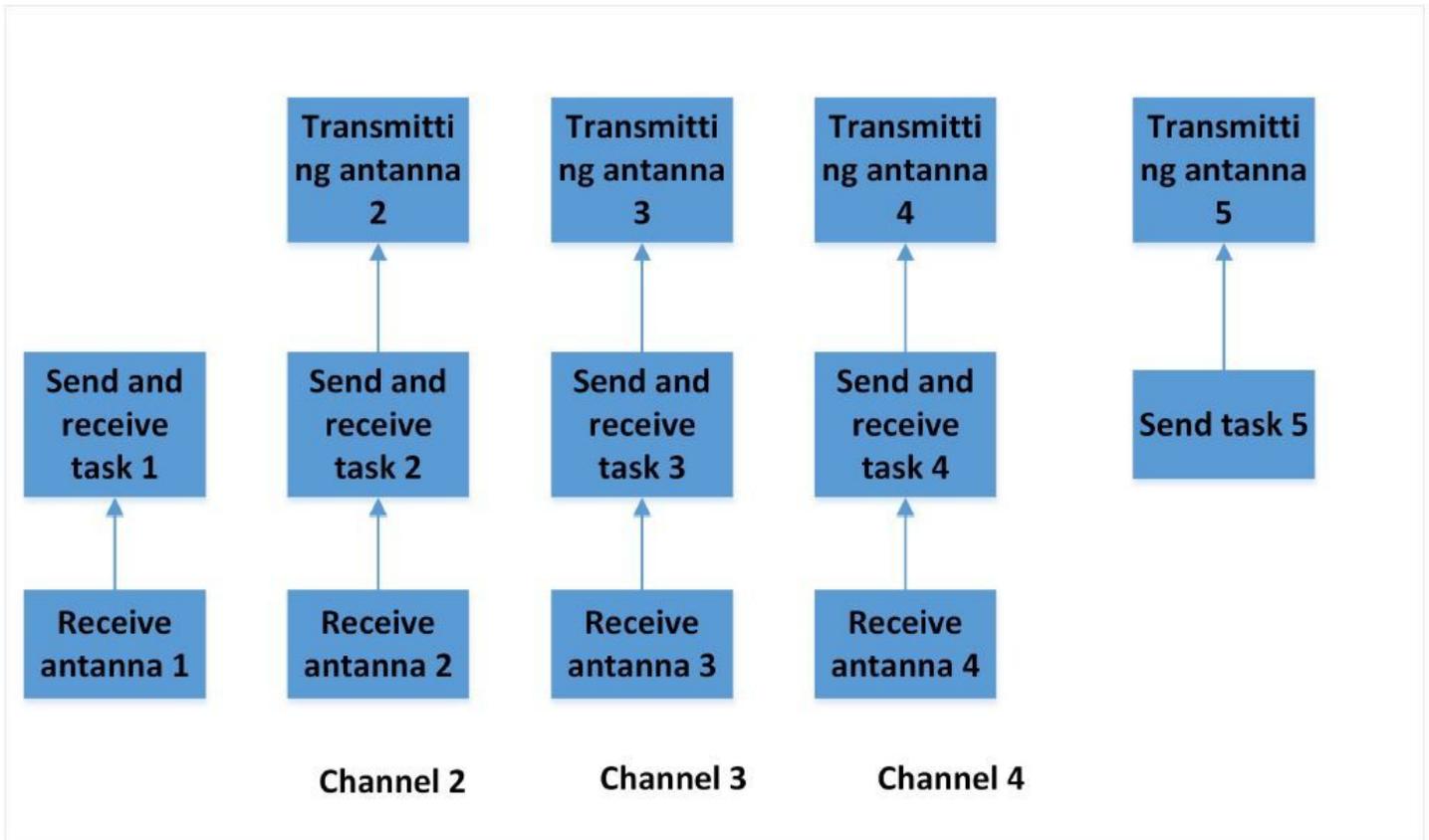


Figure 2

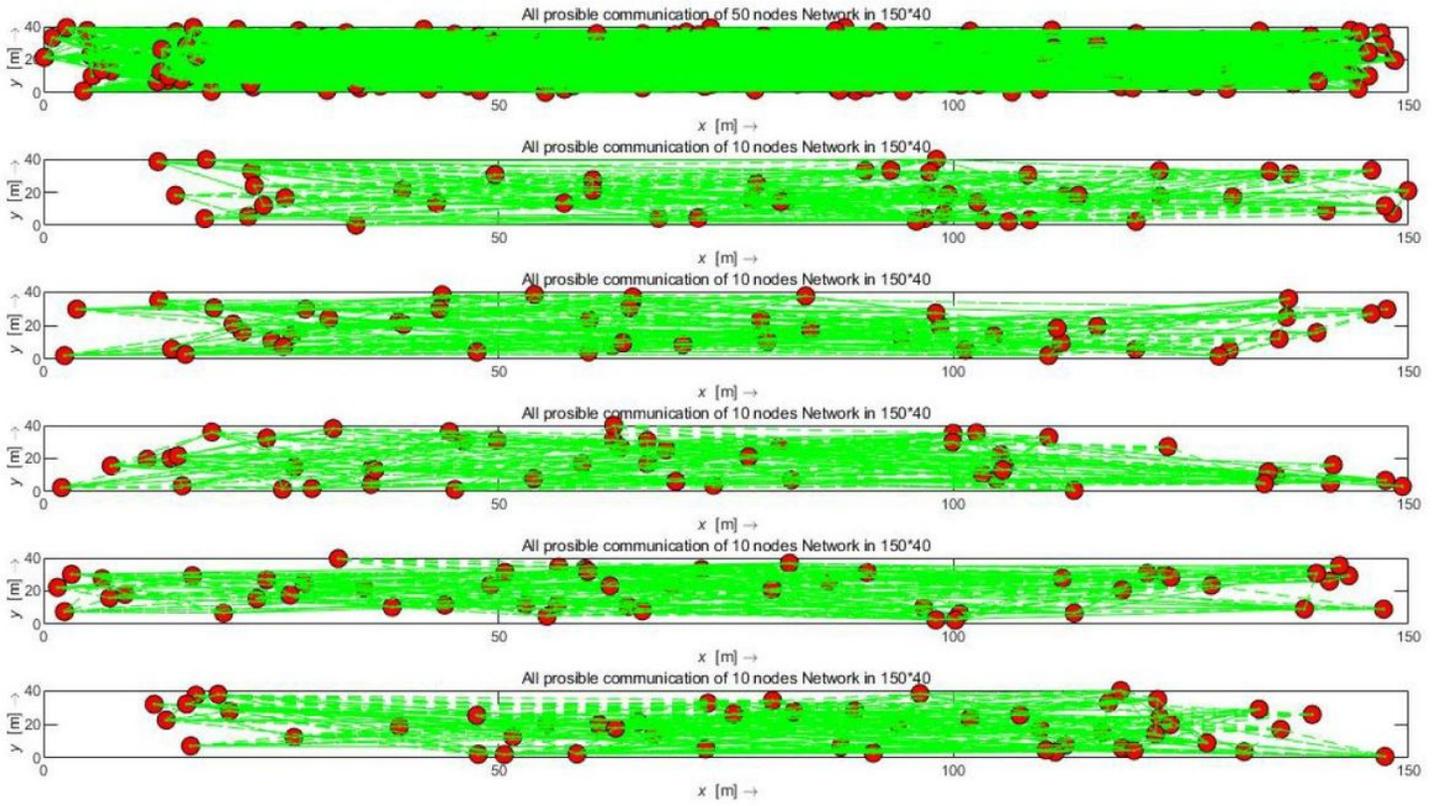
MIMO scheduling design.



(a) MIMO communication system

Figure 3

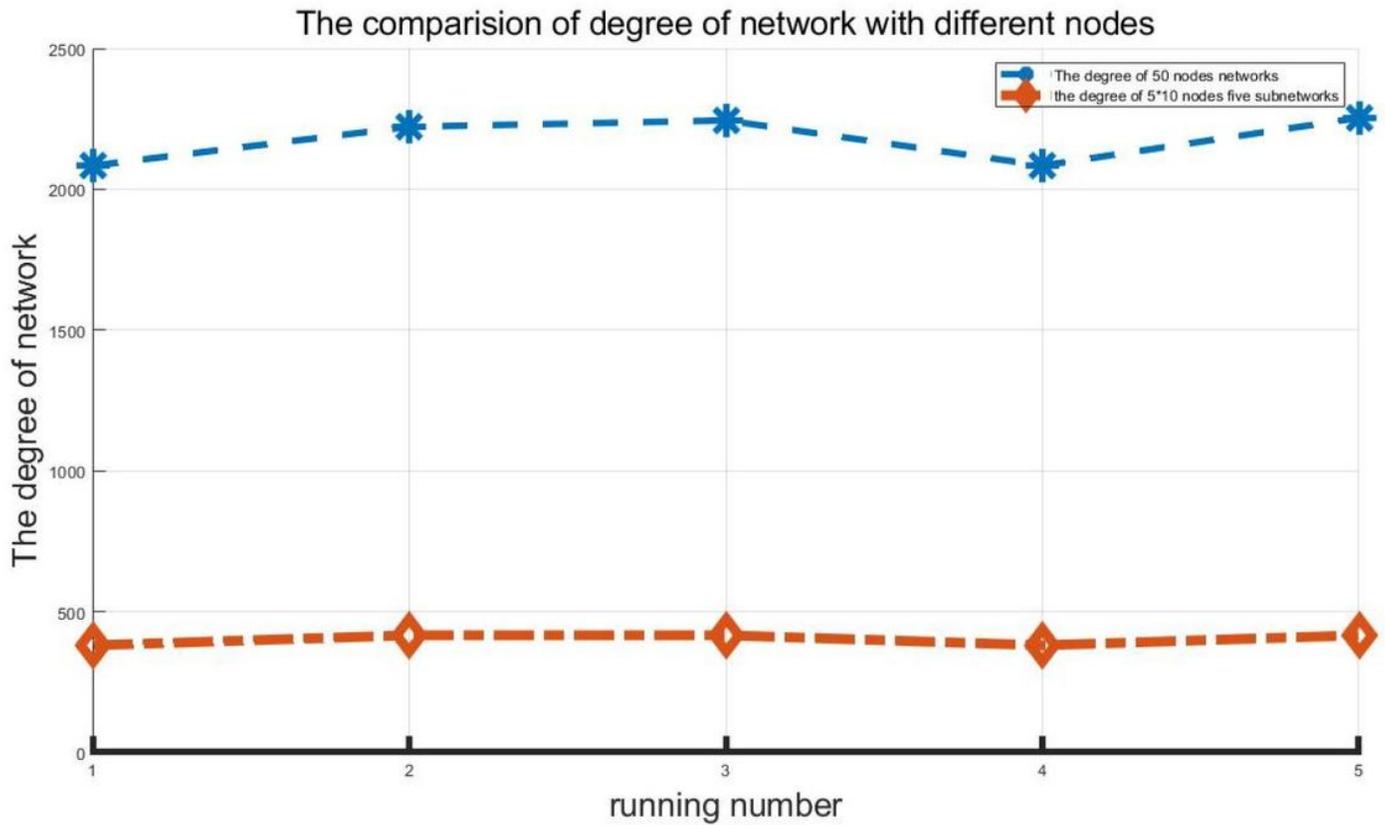
MIMO communication system.



(a) the connection status of different nodes random network

**Figure 4**

The connection status of random network.



(a) The comparison of the degrees of random network

Figure 5

The comparison of the degrees of random network .

```

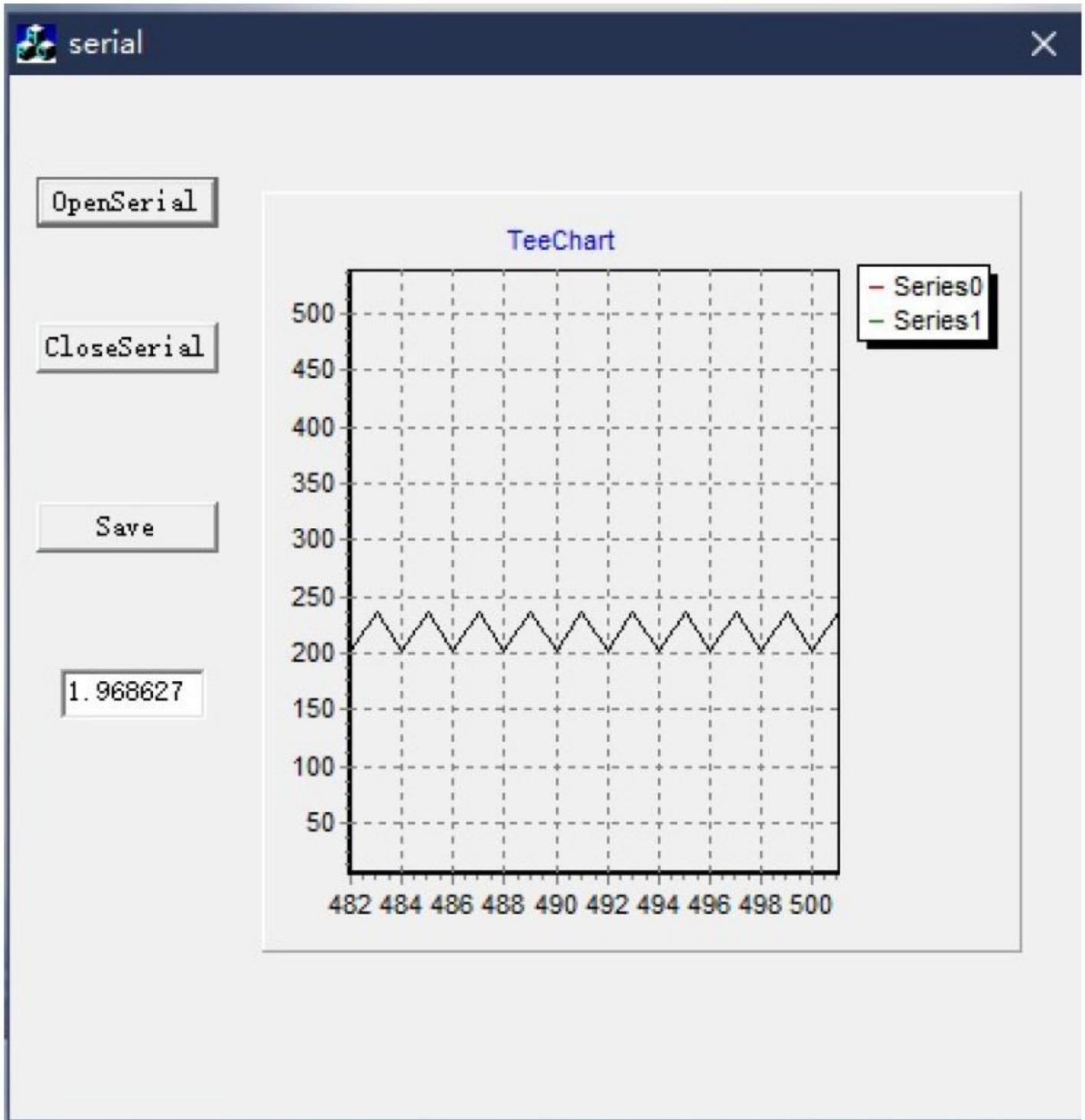
/c/ti/omnetpp/example/veins-veins-5.0
/c/ti/omnetpp$ cd example/
/c/ti/omnetpp/example$ cd veins-veins-5.0/
/c/ti/omnetpp/example/veins-veins-5.0$ cd examples/
/c/ti/omnetpp/example/veins-veins-5.0/examples$ cd veins/
/c/ti/omnetpp/example/veins-veins-5.0/examples/veins$ d:/Program/Eclipse/Sumo/bin/sumo.exe -c erlange
n.sumo.cfg
Loading configuration ... done.
/c/ti/omnetpp/example/veins-veins-5.0/examples/veins$ cd ..
/c/ti/omnetpp/example/veins-veins-5.0/examples$ cd ..
/c/ti/omnetpp/example/veins-veins-5.0$ dir
configure doc examples images print-veins-version src sumo-launchd.py
COPYING doxy.cfg format-code.sh Makefile README.txt subprojects
/c/ti/omnetpp/example/veins-veins-5.0$ sumo-launchd.py -vv -c D:/Program/Eclipse/Sumo/bin/sumo.exe
Logging to c:/users/chengl-1/appdata/local/temp/sumo-launchd.log
Listening on port 9999
Connection from 127.0.0.1 on port 55578
Handling connection from 127.0.0.1 on port 55578
Got TraCI message of length 2
Got TraCI command of length 1
Got TraCI command 0x0
Got CMD_GETVERSION
Got TraCI message of length 302
Got TraCI command of length 297
Got TraCI command 0x75
Got CMD_FILE_SEND for "sumo-launchd.launch.xml"
Got CMD_FILE_SEND with data "<launch>
<copy file="erlangen.net.xml"/>
<copy file="erlangen.rou.xml"/>
<copy file="erlangen.poly.xml"/>

```

(a) The simulation of vehicle wireless work

Figure 6

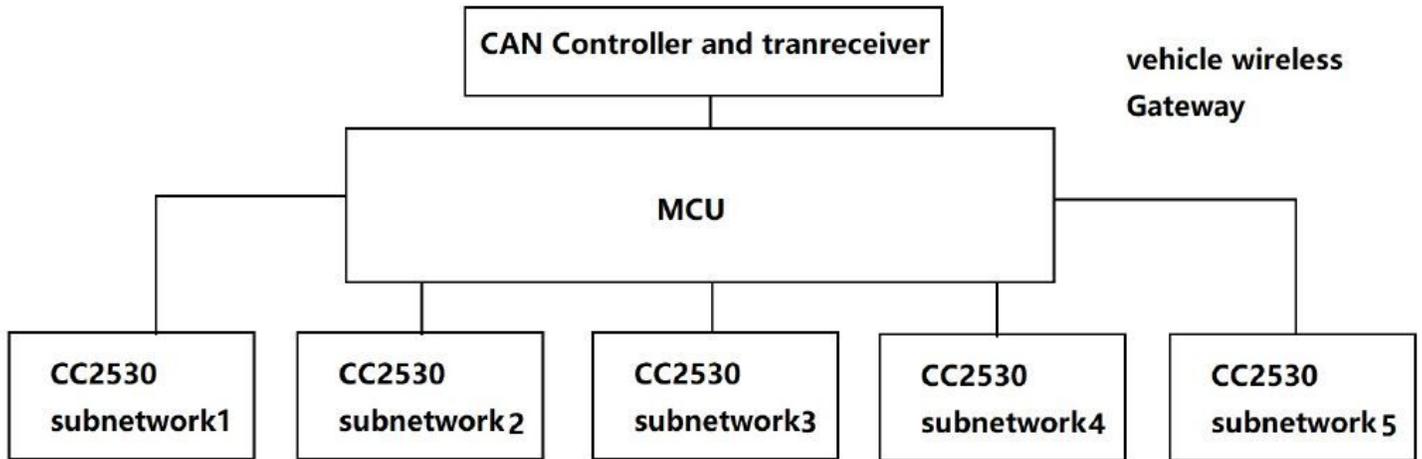
The simulation of vehicle wireless work.



(a) The software test form

Figure 7

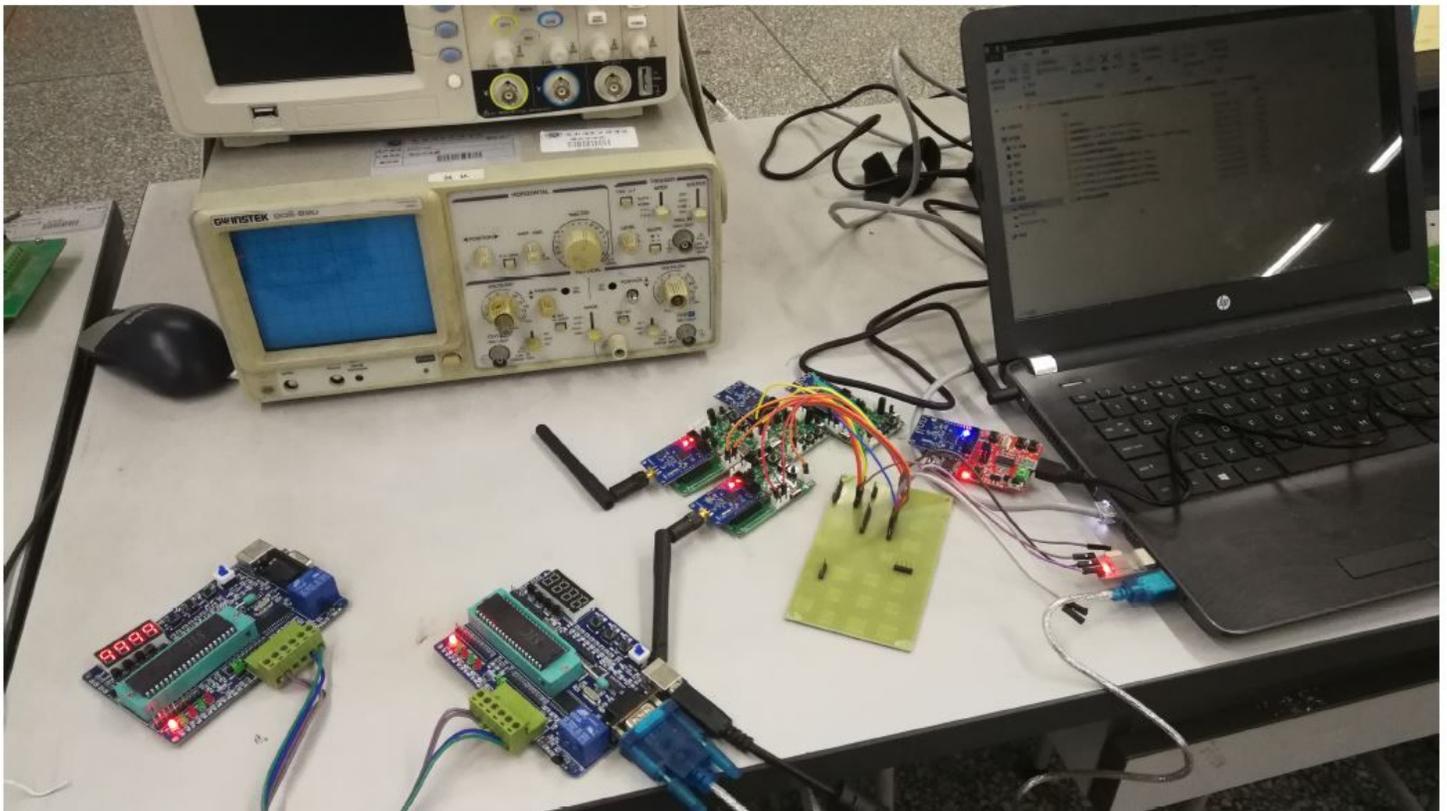
The software test form.



(a) vehicle wireless CAN gateway

Figure 8

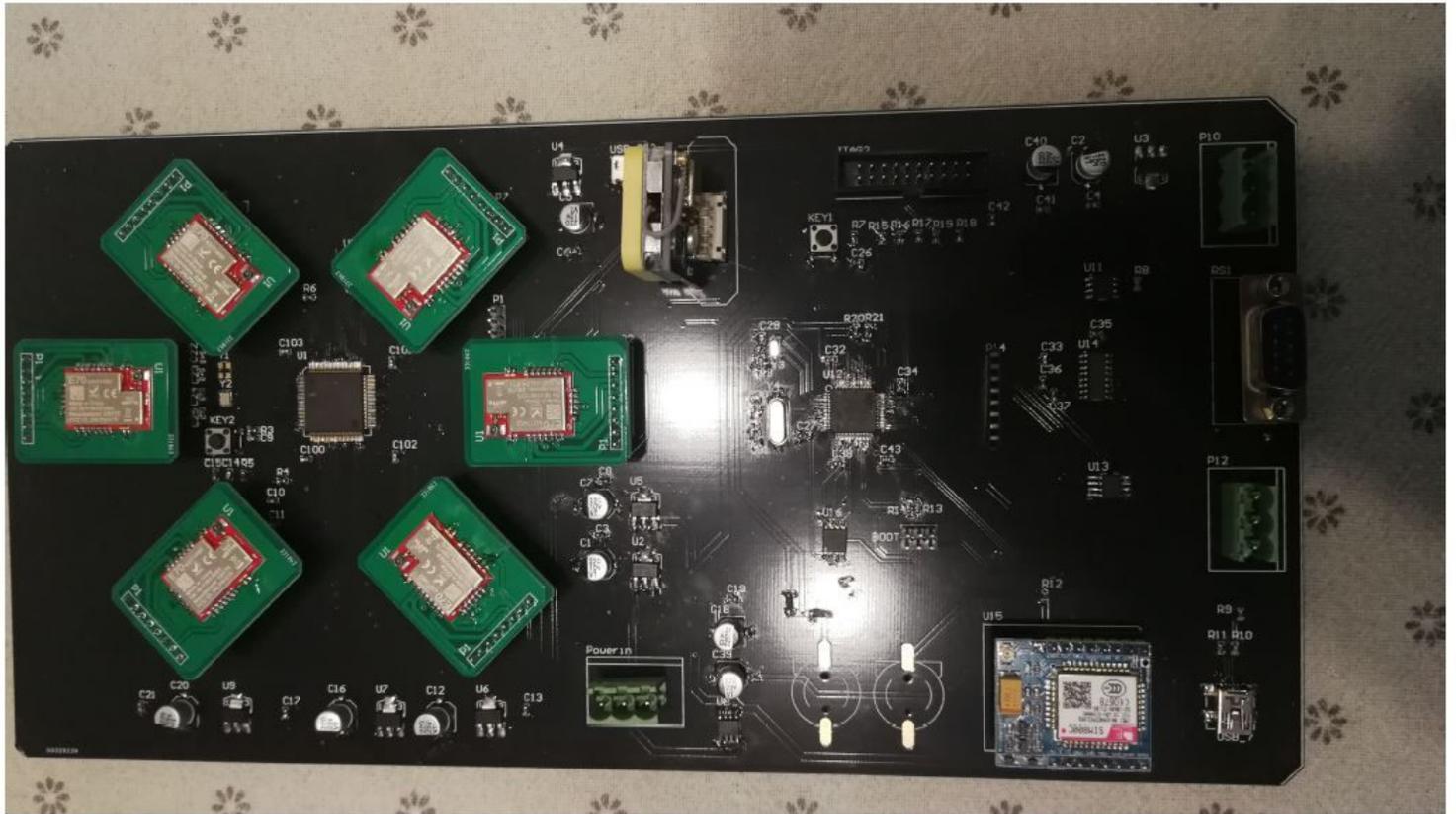
vehicle wireless CAN gateway



(a) The vehicle wireless CAN gateway hardware demo

Figure 9

The vehicle wireless CAN gateway hardware demo



(a) CC13xx gateway

Figure 10

The CC13xx gateway demo

## Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- [AuthorBiography.docx](#)
- [4](#)
- [example.eps](#)
- [history.txt](#)
- [readme.txt](#)
- [rotating.sty](#)
- [template.tex](#)