# Security System Based on CAN Bus in Accelerator Control

Cui Yonggang
Department of Engineering Physics, Tsinghua University
Beijing 100084, China
Email: cuirr@thinker.ep.tsinghua.edu.cn

**Abstract**

Controller Area Network (CAN) is widely used in industrial control. In this paper, as an item of R&D of Shanghai Synchronous Radiation Facility (SSRF), a PC board which works as the interface between the PC-bus and CAN bus has been designed. CAN control module, a kind of electronic-magnetic lock, has been developed successfully. The software, written in C++ and assemble language, gives a friendly user interface. Under control of this software the PC board working together with a few sets of control modules installed on the doors, makes up a security system. The system can be used to control all the entrance for the accelerator hall to ensure accelerator radiation safety. The system has been tested in the laboratory environment. The PC machine responds to the 'open door' requirements from the doors in time and records the 'open door' events on the disk. It can also lock all the doors when the accelerator is running. The number of entrance in the system can extend to 16 under the current design. High performance of CAN bus can be evaluated with the system.

## 1 Introduction

The Controller Area Network (CAN) was developed by Bosch and Intel for automobile communication as a serial bus. It is a serial network that efficiently supports distributed real-time control with a very high level of data integrity. It provides high-speed but low cost multiplex wiring between nodes, automatic bus arbitration, and error definition. Because of these advantages, it is a right selection in real-time control, such as accelerator control.

In this paper, as the R&D of Shanghai Synchronous Radiation Facility (SSRF), a security system based on the CAN bus has been built to test the performance of the CAN bus. In such a system, any person can only enter the controlled area with a registered magnetic card. The system can control all the doors connected to the CAN bus, respond to the In/Out requirements from the doors in time and save the information of In/Out events on the disk of PC machine. The security system is also one part of the interlock system for the accelerator. When the accelerator is going to run, the disable signals are sent to every door's control module through CAN bus.

## 2 Hardware description

The security system consists of a PC with a CAN network adapter used as the control host, and several magnetic locks with control modules embedded in every lock. The block diagram of the whole system can be seen in figure 1. As one part of the whole accelerator control system the PC can also join the LAN net with TCP/IP or other protocols to communicate with other control hosts.

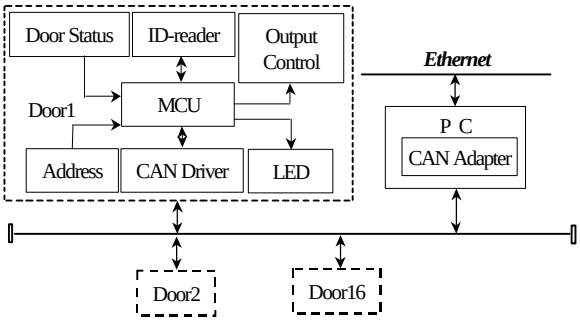


Fig. 1 Topology Diagram of Security System

### *2.1 PC computer and CAN Bus adapter*

The PC computer is the system core of control and database management. It has a CAN bus adapter in it which is designed as the interface between the PC's ISA bus and CAN bus.

On this board, there are two driving circuits for selection. The first one uses a 82C200 from PHYLIPS as the peripheral IC and 82C250 of PHYLIPS as the CAN bus transceiver IC. Between them, an optical isolator is used to reduce the interfere between two ICs. In another circuits, discrete components are employed to replace the transceiver IC and optical isolator. This network is simple and cheap. The same driving circuits are used in the control modules on locks.

### *2.2 Control module in magnetic lock*

The control module in the magnetic lock is the sensor and actor of the whole system. It reads out the ID of magnetic card and sends In/Out requisitions to PC machine. It also executes the command sent by PC machine.

The main chip of the control module is a MC68HC05X4 Microcontroller Unit (MCU) of MOTOROLA, which has the CAN control module, MCAN, embedded in it. Around the MCU, there are CAN driver module, ID-reader module, and output control module. The CAN driver module uses discrete components mentioned previously. The ID-reader module has a magnetic card reader to read out the card ID. The output control module controls the mechanical lock-up mechanism through a relay.

There are also some other circuit on the control module board, such as door status module, address module, and LED module. Door status module gives out the current Open/Close status of the door. Address module offers a address to identify a specified lock. LED module uses LED and bells to give users information prompt.

## 3 Software of the system

Software of the system has two parts. One is the program working with the control module and the other one is the control software running on the PC computer.

### *3.1 Program of MCU*

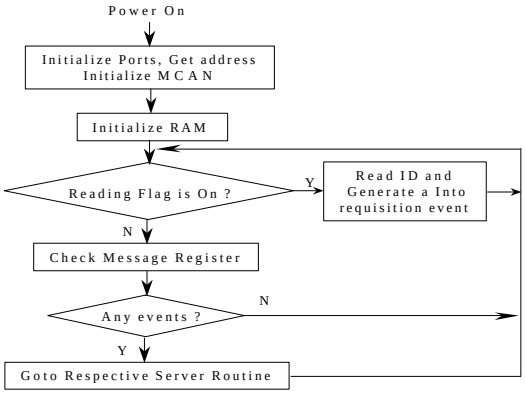


Fig. 2 Flow chart of MCU program

The software of the MCU is an event-driven program written in assemble language. The events driving the program can be the In/Out requisitions on a lock, the events of received message on CAN bus, or the requisitions events of execute commands. The flow chart of main routine of this program is shown in figure 2.

### *3.2 Software on PC*

On the PC machine, control software runs under Windows 3.1. The software includes an application written in MS VC++ for control and database management and a device driver written in 80X86 assemble language with DDK. The structure is described in figure 3.

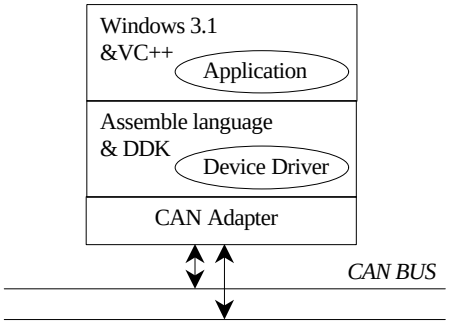


Fig. 3 Logical Structure of PC Software

#### *3.2.1 Device driver*

The Device Driver offers a DLL library to Windows 3.1. This library includes the functions, which are needed to access the CAN adapter, described as following.

```
void ReadCanData(char FAR * DataBuf); //fetch data from
        the message queue defined in device driver
void WriteCanData(char FAR * DataBuf); //write data to
        transmit buffer
void TransmitData(); //send out the data onto the CAN bus
void Initialize();//initialize the CAN adapter
```

#### *3.2.2 Application based on Windows 3.1*

The application based on Windows 3.1 has the functions to monitor the work of the whole system, including the judgment of a valid card number, the registration of each In/Out information, retrieve of history information, the database management, and so on. The application is written in MS VC++. With OOP programming, new data structure and classes for database management are defined.

```
struct RegisteredCard { //Structure to define a magnetic
            card
      char CardID[MAXCARDIDLEN]; //Card ID
      char Name[MAXNAME+1]; //Owner's name of this
            card
      struct RegisteredCard * NextCard; //Pointer of next
            card
};
struct Record { //Structure to define a piece of information
      char Name[MAXNAMELEN+1]; //Name
      char DataTime[20]; //Time
      int   Room; //Where
      char CardID[MAXCARDIDLEN+1]; //Card ID
      char    Comment[MAXEVENTCOMMENTLEN+1];
            //Some comment about it
};
class CardLinkList { //Class to define database of
            registered card
    int   ManyCard; //Number of registered card
    struct Card * Room[MAXROOMNUM]; //To store
            information of card registered on every lock
public:
    …
    BOOL Search(char * lpCardID, struct Record *
            NewRecord, int Belongto = 0); //Search a ID in
            database
    void AddCard(.);
    void DelCard(.);
    …
};
class RecordQueue { //Class to define information queue,
            each node is a piece of information
```

```
    struct Record Event[MAXEVENTNUM]; //Queue
    int   Header, Tail; //Header and tail pointer of queue
public:
    BOOL Empty(); //Queue is empty or not
    BOOL Full(); //Queue is full or nor
    void Initialize(); //Initialize the queue
    …
    void GetARecord(struct Record * pointer); //Fetch a
         piece of information from queue
    void AddRecord( struct Record AnEvent); //add a
         piece of information into queue
    …
};
```


Fig. 4 Window of Application

Based on these structures and classes, the application is built. It has an interface of Windows style, which is easy to use. Figure 4 shows the main window.

## 4 Conclusion

The security system is tested in the laboratory environment with several locks connected to the CAN bus. Some advantages of CAN bus can be seen here.

First is automatic bus arbitration. In such a multimaster system, several nodes may send messages at same time. According to the priority of the message, CAN bus arbitrates them automatically. In the experiment, when the requisitions occur on different doors at same time, the PC machine can immediately answer them one by one. No delay time can be felt.

Secondly, CAN bus has a high speed and low cost. In the experiment, the physical medium is unshielded twist ware (UTP). The transmission rate can be 667 bits/s with 40m long ware, limited by the frequency of 8MHz crystal on the CAN adapter. Theoretically, this rate can be 1Mbits/s high within such a distance.

Thirdly, like the Ethernet, CAN bus is a network to broadcast message, not a peer-to-peer network. In the experiment, the PC machine can send a instruction to lock every door in the system. Because of this, the control machine does not have to pole other nodes one by one. It is very useful in changing parameters in different parts of a system synchronously, such as to increase the magnetic fields in every part of store ring.

In addition, according to the driving capability of driving circuit, the number of nodes in such a system can extend to 16 currently. In order to add more nodes, transistors must be used.

This project only demonstrate the CAN bus. We expect it to find its right position in system control of SSRF.

## Reference

[1] MC68HC05 ADVANCE INFORMATION, Motorola Inc.,1992

[2] Techniche Informationen, DESY, 900927