Design of Bidirectional Security System
for Intravenous Drip Infusion with Hybrid Communication

Chin-Cheng Chen,[1,2] Fu-Hsien Chen,[3] Chun-Liang Hsu,[4*] Wei-Ching Chuang,[5]
Chun-Ying Lee,[5] Chun-Hsien Lee,[5] Chi-Ting Ho,[6] Yi Ma,[7] and Tung-Lung Wu[7**]

[1]Information and Engineering College, Jimei University, Xiamen 361021, China
[2]Department of Aeronautical Engineering, Chaoyang University of Technology, Taichung 413, Taiwan
[3]Department of Electronic Engineering, Lunghwa University of Science and Technology Taoyuan City 333, Taiwan
[4]Department of Electrical Engineering, St. John's University, New-Taipei City 251, Taiwan
[5]Institute of Electro-Optical and Materials Science, National Formosa University, Huwei 632, Yunlin, Taiwan
[6]Department of Mechanical Design Engineering, National Formosa University, Huwei 632, Yunlin, Taiwan
[7]School of Mathematics and Information Engineering, Longyan University, Longyan, Fujian, China

Automation systems are widely applied in many fields such as manufacturing, warehousing, machinery, power management, and smart living spaces. In this study, a bidirectional security system was designed and applied to intravenous (IV) infusion for wards in hospitals. The system comprised a high-resolution gravity sensor, a 2.4 GHz XBee wireless sensor network communication module, and a drip-quantity-detecting control box with a HOTEC Corp. MCU HT66F70A core microprocessor to achieve automated management in the medical field. Toward achieving sustainability in our medical environment and computer science technology, we adopted data transformation, an Internet of Things (IoT) wireless communication and sensor network, and an app to construct the bidirectional security system for IV infusion. The main aims of the proposed method in this study were as follows: (1) to invent a reliable method to detect the remaining quantity of a patient's drip infusion using a gravity sensor so as to prevent the dangerous condition of blood flowing backward or air intruding into the vein as well as to decrease the burden on nursing staff in wards, and (2) to avoid medical disputes due to the incorrect infusion from drips, the autorecognition of patients using scanning QR code technology was applied in the medical management system. The outcome of the study was proved successful and has won many important national invention contests.

1. Introduction

1.1 Study background and problems addressed

Many people have experienced taking care of ill family members. When the patient needs an infusion from an intravenous (IV) drip, the family taking care of the patients must carefully

---

monitor the IV drip lest it run out and cause blood to flow backward or air to enter the vein, causing danger to the patient. Although nursing staff in wards are responsible for monitoring IV drips in hospitals, this is a major burden on nurses. Nowadays, automation has been applied in many fields such as manufacturing, home appliances, warehouse management, public offices, and traffic management. In this study, we constructed a sustainable environment by implementing data transformation theory and the Internet of Things (IoT) for communication, which was combined with a high-resolution gravity sensor to detect the remaining quantity in the drip for application in hospital wards.

1.2   Study objectives and proposed method

In this study, we aim to utilize automation technology in the management of wards in a hospital by applying hybrid communication, a wireless sensor network, a human–machine interface, and an app to create a system for automatically detecting the condition of an IV drip. Not only is the remaining quantity of the drip ascertained but also whether it is the correct drip for the patient by comparing information periodically collected by the nurse with the information stored in the cloud server in advance. With this automation technology, we can not only decrease the burden on nurses in wards and thus increase their work efficiency but also decrease cases of medical negligence that result in medical disputes between hospital management and families of patients.

2.   Literature Review and Structural Concept of Design

2.1   Literature review

Automated management systems are interactive systems comprising humans, computer facilities, and managed objects. The core of these systems is information management. There have been many studies in this field. Mesquida and Mas studied integrated IT management from rules to organization issues.[1] IBM developed an intelligent management software program called Tivoli and proposed a new serial operation and maintenance system for IT.[2] An infrastructure as a service (IAAS) automation management system used cloud computing techniques to manage IT resources.[3] Another study focused on related techniques of automated management systems with the aim of producing green IT products.[4] In addition, L'hadi *et al.*[5] used a ZigBee wireless communication technique to build up a large-scale monitoring network with a hybrid topology and applied automated management techniques to the location management of a parking lot. In Ref. 6, the author used IT components or devices to explore monitoring techniques combined with wireless communication and supervisory control and data acquisition (SCADA) system data transmission, and visualized data analysis for efficient power management.

In Ref. 7, the authors applied wireless sensor network and wireless communication techniques to improve the efficiency of the power supply for traffic lights on a highway.[7] In Ref. 8, the authors designed a method of detecting water-leaked systems, which combined a

neural network and a wireless sensor network communication technique in which a humidity sensor system was designed around water pipes. The collected data was transmitted to the monitoring servo by RF and RS-485, and then the data was analyzed and shown on a screen to provide a real-time display of the water usage of a building. In Ref. 9, the author used an IoT technique to design a long-term environment monitoring system.[9] In Ref. 10, the authors integrated RFID and a wireless sensor network to develop a new structure and techniques at a low cost.[10] Reference 11 focused on wireless transmission to monitor patients' beds including urine detection and the flow amount.

In Ref. 12, calibration circuits and a frequency converter for CMOS for liquid level detection were explored and applied to detect the amount remaining in an IV drip.[12] In Ref. 13, the authors implemented a QR code of the human body for use in monitoring techniques and developed some related techniques.[13] Varadharajan *et al.* explored the application of microwave-reflectometer-based systems for the automatic control and real-time monitoring of the flow and level in IV medical infusions.[14] In medical and hospital environments, other types of devices based primarily on optical detection and counting infusion droplets are used. However, the proposed system in Ref. 14 is designed to avoid some of the typical shortcomings of these conventional methods, providing an effective means of automatically monitoring the instantaneous flow of IV medical solutions. To this end, the proposed system combines microwave time-domain reflectometry (TDR) measurement with a noninvasive sensing element (i.e., a strip electrode that is directly connected to the outer surface of an infusion bottle). Experimental results confirm that the solution flow can be controlled with acceptable accuracy by using a low-cost portable TDR device. Therefore, the proposed method can be considered as a promising control tool for in-hospital patient management and telemedicine procedure.[14] In Ref. 15, after the drug information was verified using a QR code, IV injection and flow detection were performed. Kim *et al.* argued that the flow rate of injected medication implemented by IV systems must be accurately monitored and meticulously controlled to prevent medical accidents. They designed and fabricated an ultrasonic flow (UF) meter with 1–3 piezoelectric composite transducers and tested it on a variety of flow rates reflecting those in medical injections. They also included a transducer wedge for angled beam propagation and an acoustic impedance matching layer in the design for transmission enhancement. To accurately measure the flow rate, the effect of the flow distribution inside the IV tube was taken into account. The developed UF successfully detected low flow rates (< 0.005 m/s) with a discrepancy of 1–2% compared with the reference rate of the infusion.[11] Rapid advances in personal health-care systems based on implantable and wearable medical devices promise to greatly improve the quality of diagnosis and treatment for a range of medical conditions.[16] However, increasing the programming ability and wireless connectivity of medical devices also provide opportunities for malicious attackers. Unfortunately, implantable/wearable medical devices have extreme size and power constraints and employ unique models, making it infeasible to simply borrow conventional security solutions such as cryptography. Zhang *et al.* proposed a general framework based on wireless channel monitoring and anomaly detection for securing medical devices. Our proposed system is based on a medical security monitor (MedMon) that monitors all the RF wireless communications to/from medical devices and

uses multilayered anomaly detection to identify potentially malicious transactions. Upon the detection of a malicious transaction, MedMon takes appropriate responses, including passive (notifying the user) and active (jamming the packets so that they do not reach the medical device) responses. A key benefit of MedMon is that it is applicable to existing medical devices in use by patients without hardware or software modifications. Consequently, it also has zero power overheads on these devices. Zhang *et al.*[15] demonstrated the feasibility of our proposal by developing a prototype insulin delivery system using off-the-shelf components (USRP software-defined radio). They evaluated its effectiveness under several attack scenarios. The results showed that MedMon can detect virtually all naive attacks and a large fraction of more sophisticated attacks, suggesting that it is an effective approach to enhancing the security of medical devices.[12]

Wireless body area networks (WBANs) provide tremendous opportunities for remote health monitoring in the future. However, WBAN health-monitoring systems encounter a number of challenges in their realization including efficient information extraction, dynamically fine tuning the monitoring process according to the quality of data, and translating the high-level requirements of medical officers to low-level sensor reconfiguration.[13] Varadharajan *et al.* discussed some of the requirements for using a technology for healthcare applications and proposed techniques for the secure monitoring of patients with wandering behavior in a hospital or elderly care environment. One of the aims of our work in this study is to use a technology for the secure monitoring of patients with wandering behavior to keep them away from danger and detect whether the behavior of the patient violates the policies of the hospital or the privacy of other patients. Their approach makes use of software-defined networking (SDN), wireless LAN (WLAN), and wearable devices for patients. Our approach incurs a low cost since a WLAN is widely deployed. However, there are some challenges in using WLAN for monitoring patients with dementia since it is primarily used for accessing the Internet and its open nature makes it vulnerable to different types of security attacks. Hence, we use SDN to solve some of these problems and prioritize monitoring services. We have developed a security application for an SDN controller that can be used to enforce fine granular policies for communication between the hosts, realize the real-time location tracking of patients, and deal with attacks on hospital networks.[14] Zhang *et al.* predicted that IoT, especially smart wearable devices, will play an important role in improving the quality of medical care, providing convenience for patients and improving the management of hospitals.[15] However, owing to the limitations of communication protocols, no unified architecture that can connect all intelligent objects in smart hospitals exists, although this will be made possible by the emergence of narrowband IoT (NB-IoT). In light of this, we propose an architecture to connect intelligent objects in smart hospitals based on NB-IoT and introduce edge computing to deal with the requirement of latency in the medical process. As a case study, Zhang *et al.* developed an infusion monitoring system to monitor the real-time drop rate and volume of the remaining drug during IV infusion. Finally, they gave some challenges and future directions for building a smart hospital by connecting intelligent objects.[15] Elfaramawy *et al.* designed a low-power wireless respiratory monitoring system with cough detection to measure the breathing rate and the frequency of coughing. This system uses wearable wireless multimodal patch sensors designed using off-the-

shelf components. These wearable sensors use a low-power nine-axis inertial measurement unit to quantify respiratory movement and a microphone to record audio signals. Data processing and fusion algorithms are used to calculate the respiratory frequency and coughing events.[16]

From the related literature above, we acquired the following important viewpoints:

(1) A remote monitoring system includes an end device to detect the data required by the system.

(2) An intelligent monitoring system must be equipped with a reliable communication system.

(3) A powerful human–machine interface design of a servo computer can support the efficient processing of data and messages.

To realize a reliable IV drip infusion detecting system, it is necessary to consider (1) the design of the end device with the function to detect the remaining quantity in the drip and transmit the data to the servo computer, (2) the communication method used to precisely transmit the data from the end device to the servo computer, (3) the design of an interface of the main servo computer in the management center far from the wards, which has powerful data processing capabilities, and (4) the use of a modern technology to ensure the reliability and safety of the drip infusion system.

2.2 Concept of structure design

We address the challenges in the previous subsection by proposing an architecture that allows virtual groups to be formed between devices of patients, nurses, and doctors to enable the remote analysis of WBAN data. Group formation and modification are performed with respect to patients' conditions and medical officers' requirements, which can easily be adjusted through high-level policies. We also propose a new metric called quality of health monitoring (QHM), which allows medical officers to provide feedback on the quality of the WBAN data received. The accumulated WBAN data are transmitted to virtual group members through an underlying environmental sensor network. The proposed approach is evaluated through a series of simulations. On the basis of the literature review, we can summarize that to minimize the time taken, cost, the burden on monitoring staff, and dangerous factors in the environment, the best solution for a long-term monitoring system is a wireless sensor network and related communication techniques. Therefore, in this work, we applied a wireless sensor network and related communication techniques to the IV drip monitoring of patients in the wards of a hospital to solve the related data transmission problem. Finally, an architecture is proposed, as shown in Fig. 1, which allows virtual groups to be formed between devices of patients, nurses, and doctors to enable the remote analysis of WBAN data.

3. Design Procedures and Hardware & Software Techniques

The system has three parts described as follows:

(1) A drip-detecting control box, which is used to detect the remaining quantity of the drip. When the remaining quantity is less than a specific quantity, the control box sends a warning to the nursing center.
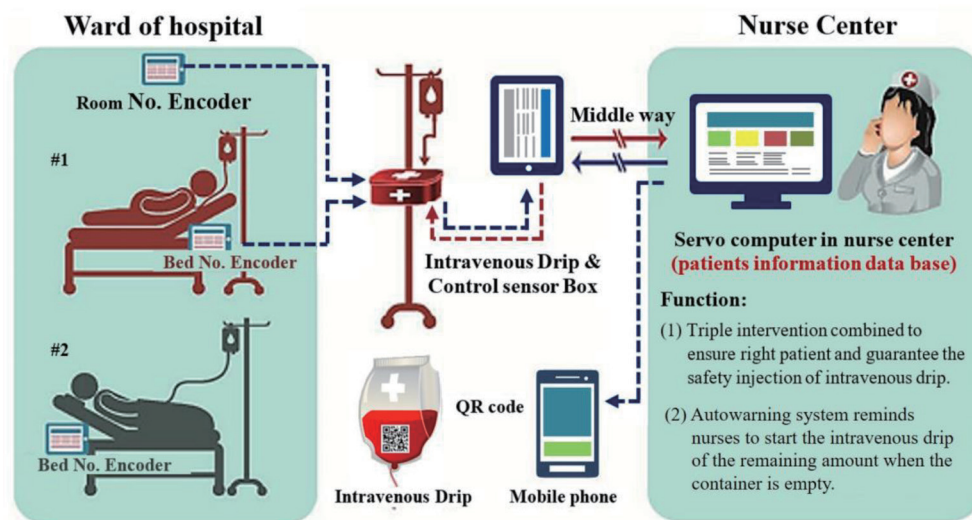
Fig. 1.　　(Color online) Structure of secure IV drip system.

(2) When the nursing center receives the warning requests from the control box, a related servo screen indicates the condition and requests both the nursing staff and the patient's family to change the drip.

(3) When a nurse goes to inject the patient or change the drip, he or she will compare the related information by using a mobile phone app to scan the QR code on the patient's wrist, and send it to the nursing center management servo. If the result is correct, the nurse can proceed with the next step of the work. The corresponding screens on the mobile phone are shown in Figs. 2–4.

In this study, we implemented a completely new method to detect the remaining amount of drip with a load-cell sensor (Fig. 5) and transfer the weight into a drip volume after an A/D conversion process. The related procedures are as follows:

(1) First, we used a load cell to sense the weight of a full bottle of a drip. The sensed weight was transferred by a tiny electric voltage (about 0.00x mv) that could not be converted by the A/D converter in a HOTEC microprocessor. Therefore, it was necessary to design an amplifier circuit to enlarge the voltage range to 0–3.3 V in accordance with the HOTEC data sheet. Then, the enlarged voltage of the drip weight was sent to the A/D converter in the HOTEC microprocessor.

(2) Secondly, the A/D converter converted the analog voltage to a digital value that was an output of binary data; the first converted data was equal to the full-bottle drip weight, and we set up the variable MAXCELL in a system C-language operating program. We then poured away the drip liquid in the bottle to carry out another A/D conversion for the empty bottle, and we obtained the converted data of the empty bottle, then set up the MINCELL variable.

(3) In the initialization stage of the program, we calculated the conversion range of the drip weight excluding the bottle weight, which was RangeCell = MAXCELL − MINCELL, so

Fig. 2. (Color online) QR code scanning interface on mobile phone (before scanning).

Fig. 3. (Color online) QR code scanning interface on mobile phone (after scanning).
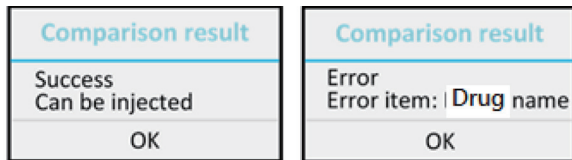

Fig. 4. (Color online) Echo message from servo computer after message comparison.
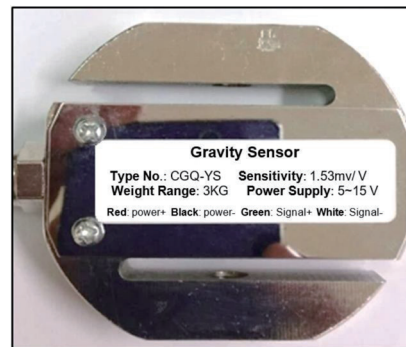

Fig. 5. (Color online) Load-cell gravity sensor.

that, for example, if the amount of drip liquid changes from 0 to 500 g, the corresponding converted data range can be from the chip A/D converter.

(4) Then, we input the analog voltage of 3.3 V to obtain the corresponding maximum weight (for input voltages from 0 to 3.3 V, indicating the analog input for the A/D converter as well as the weight range from minimum to maximum). We obtained a maximum weight of 2408 g, which was stored as the variable MAXg in the program.

(5) Then, we obtained the digital value change per g (stored as the variable Ming in the program), which was calculated as Ming = MAXg/RangeCell. This also meant that we could detect the A/D-converted digital data. Because we want to increase the accuracy of the detected weight of the object on the load cell, we adopted the average value of MAXg obtained by sampling 16 times, which was the variable ADVAGE in the program, allowing

us to obtain the equivalent weight (stored as the variable REALCELL) of the detected object on the load cell. The formula was REALCELL = ((ADVAGE−MINCELL) × Ming). The program used to calculate the weight of the detected object was as follows.

```
//=======================================
Weight calculation
//=======================================
RangeCell = (MAXCELL−MINCELL);
Ming = (MAXg/RangeCell)
//=======================================
//ADRAVG has been done 16 times averaged
REALCELL = ((ADVAGE−MINCELL) × Ming);
```

Part of the A/D converter program is shown as Fig. 6.

3.1 Hardware design of drip-detecting control box

In this study, we implemented a HOTEC MCU HT66F70A chip as the main control chip (Fig. 7). We used its 12-bit enhanced A/D converter function to receive the analog signal from the load cell, and then convert it into a digital signal. Because there is no UART peripheral device


```
DEFINE_ISR(ISR_TM0,0x14)
{
    CTM_Count1--;
    _clrwdt1();
    switch(CTM_Count1)
    {
            case 2345:
                        ADR1=ADR;
                        ADRTotal=ADRTotal+ADR1;
                        ADR1_LByte=(u8)ADR1;
                        ADR1_HByte=(u16)ADR1>>8;
                         clrwdt1();
                        break;
            case 2190:
                        ADR2=ADR;
                        ADRTotal=ADRTotal+ADR2;
                        ADR2_LByte=(u8)ADR2;
                        ADR2_HByte=(u16)ADR2>>8;
                        _clrwdt1();
                        break;
            case 2035:
                        ADR3=ADR;
                        ADRTotal=ADRTotal+ADR3;
                        ADR3_LByte=(u8)ADR3;
                        ADR3_HByte=(u16)ADR3>>8;
                        _clrwdt1();
                        break;
            case 1880:
                        ADR4=ADR;
                        ADRTotal=ADRTotal+ADR4;
                        ADR4_LByte=(u8)ADR4;
                        ADR4_HByte=(u16)ADR4>>8;
                        _clrwdt1();
                        break;
```
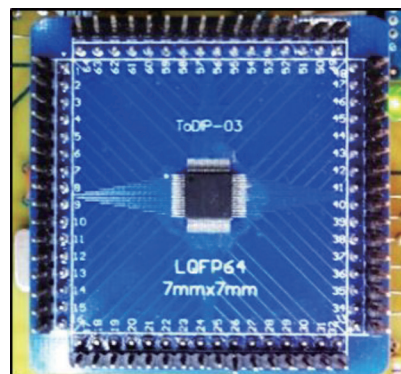
Fig. 6.   A/D conversion program.

Fig. 7.   (Color online) HOTEC MCU HT66F70A chip.

in the HT66F70A chip, we need to use I/O PC3, PC4 to simulate UART Tx and Rx and send information to the nursing center management servo with ZigBee. The simulated step was in the program used to define Tx and Rx as transmit and receive data pins, respectively, and then we defined the transmit speed through the system frequency and the transmit constant of the computer UART using the formula.

UART TRANSMIT SPEED CONSTANT
= {SYSTEM FREQUENCY/ (UART TRANSMIT SPEED × 12)} − 3.

Through HT66F70A I/O PC3, PC4, we simulate UART Tx and Rx and send information to the nursing servo with ZigBee. The related part of the program is shown in Fig. 8.

3.2 Hardware design of wireless communication with ZigBee

In practical application in wards, the information transmission should adopt wireless communication instead of cables for aesthetic reasons. Therefore, we designed a wireless communication control board by combining XBee-PRO S2B 89C2051 and ZigBee Modules as shown in Fig. 9.

In this study, the communication format of the ZigBee sensor network followed the XBee inner communication protocol, and separate protocols were designed between the mobile phone and the servo. The protocols of the communication system are shown as follows.

```
//===================================================
//      UART      INITIAL
//=====================================
//      _cp0c=0x08;
//      _cp1c=0x08;
    TX_PC2Direction=0;                  //Config PC.2 as Output
    RX_PC3Direction=1;                  //Config PC.3 as Input
    TX  data=1;                         //TX Pin Keep High
    BaudRate=39;                        //9600

    ZigbeeREG_Const=0;

    TransmitData[0] = 0x7E;             //0X7E
    TransmitData[1] = 0x00;             //0
    TransmitData[2] = 0x15;             //15
    TransmitData[3] = 0x10;             //10
    TransmitData[4] = 0x00;             //0
    TransmitData[5] = 0x00;             //MAC Address 1
    TransmitData[6] = 0x00;             //MAC Address 2
    TransmitData[7] = 0x00;             //MAC Address 3
    TransmitData[8] = 0x00;             //MAC Address 4
    TransmitData[9] = 0x00;             //MAC Address 5
    TransmitData[10] = 0x00;            //MAC Address 6
    TransmitData[11] = 0xFF;            //MAC Address 7
    TransmitData[12] = 0xFF;            //MAC Address 8
    TransmitData[13] = 0xFF;            //Address 1
    TransmitData[14] = 0xFE;            //Address 2
    TransmitData[15] = 0x00;            //0
    TransmitData[16] = 0x00;            //0
    TransmitData[17] = 0x00;            //0
    TransmitData[18] = 0x00;            //0
    TransmitData[19] = BedNumber;       //Device Number
    TransmitData[20] = CELL_LByte;      //CC Low  Byte
    TransmitData[21] = CELL_HByte;      //CC High Byte
    TransmitData[22] = ADRAVG_LByte;    //ADC Low  Byte
    TransmitData[23] = ADRAVG_HByte;    //ADC High Byte
    TransmitData[24] = 0x00;
//=============================
```

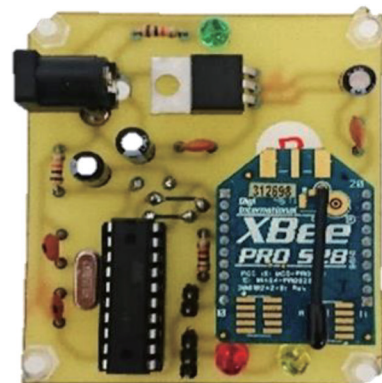Fig. 8. Part of program for using I/O pins to simulate UART.

Fig. 9. (Color online) Wireless communication module on drip-detecting control box.

(1) Protocol between servo and middle way

The protocol content includes the MAC address, start bit, data length, package type, communication number, echo, registration number, gender, name, and bed number as shown in Fig. 10.

(2) Protocol between drip-detecting control box and middle way

The protocol content includes the XBee MAC address, the condition of the drip, the type of drip, the device name, the data of the drip, and the data for verification as shown in Fig. 11.

(3) Protocol between mobile phone and servo

The protocol content includes the patients' social security number, bed number, drip number, drip-detecting control box number, and comparison data as shown in Fig. 12.

The hardware design of the drip-detecting control box is schematically shown as Fig. 13. A flowchart of the software design is shown in Fig. 14.
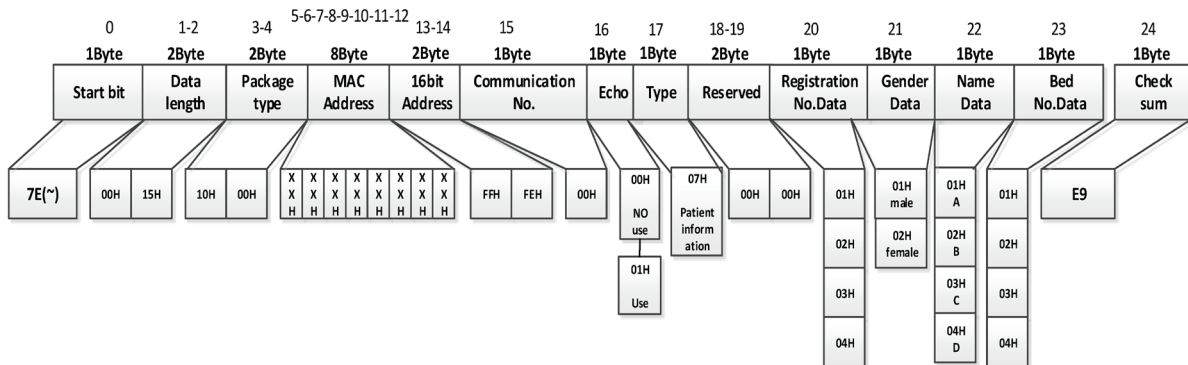
Fig. 10.   Protocol between servo and middle way.
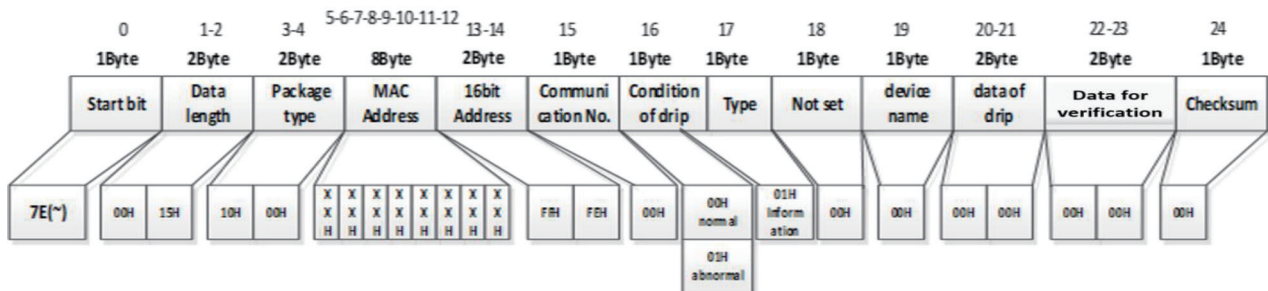
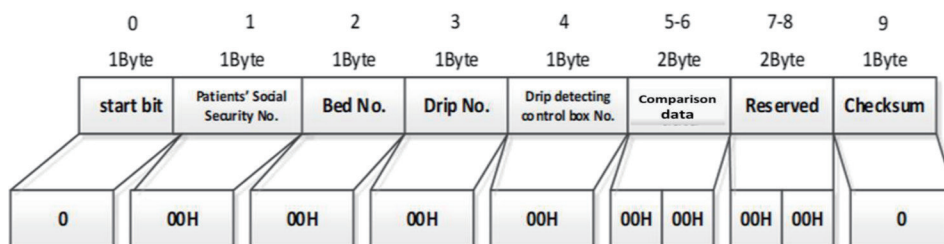Fig. 11.   Protocol between drip-detecting control box and middle way.

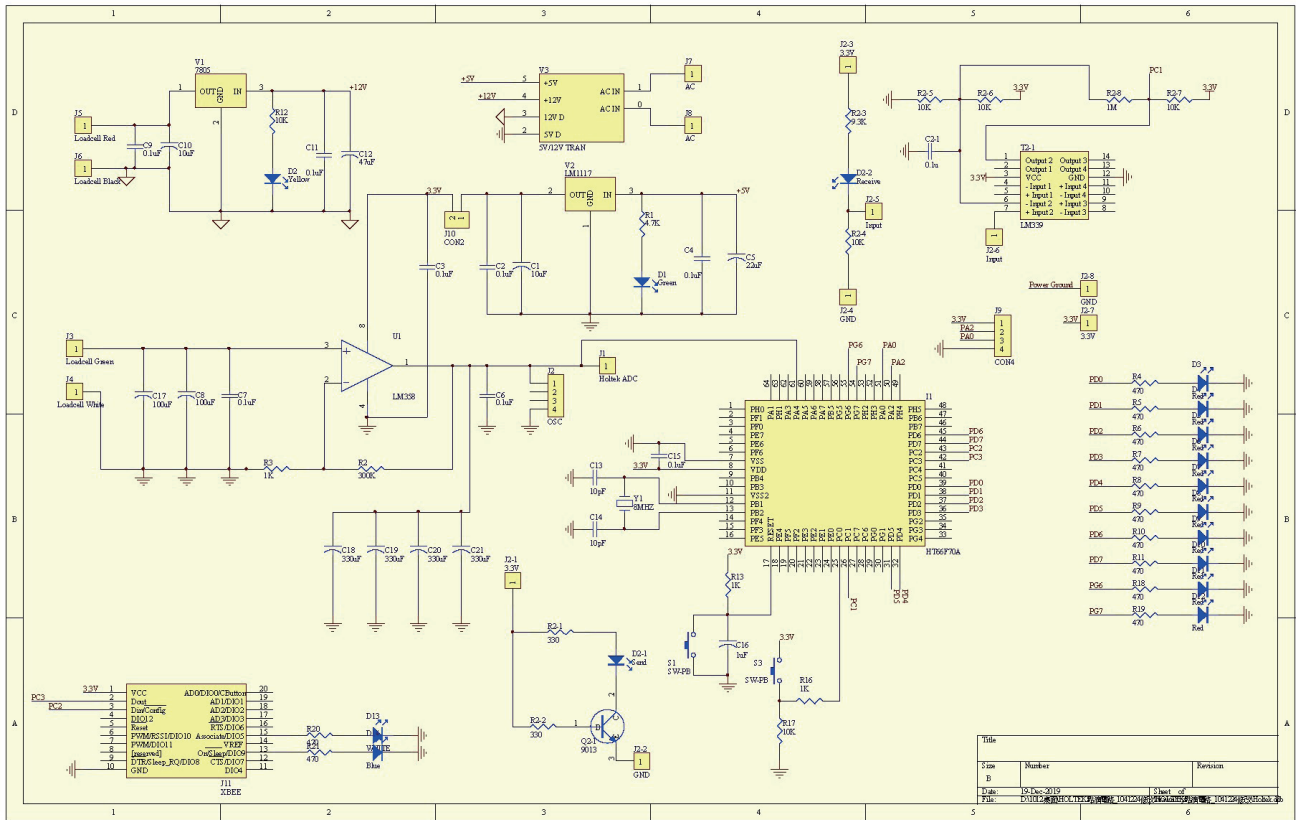Fig. 12.   Protocol between mobile phone and servo.

Fig. 13.   (Color online) Schematic of drip-detecting control box hardware design.
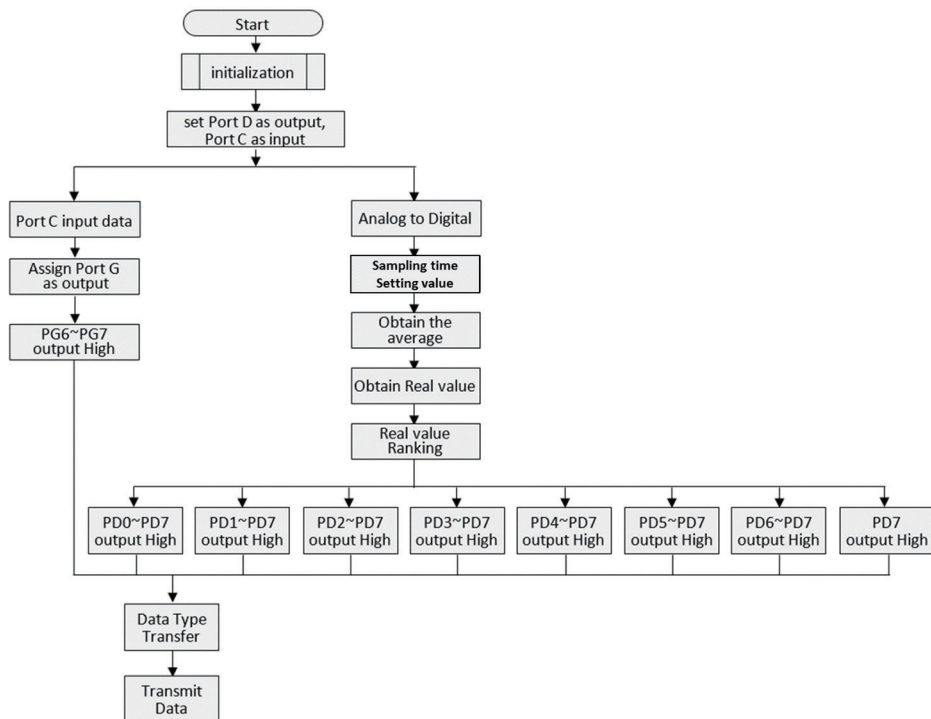

Fig. 14.   Firmware flowchart of MCU HT66F70A chip.

3.3 System software design of servo and middle way

We first need to collect patient information after the patient sees the doctor. This information includes name, gender, birth, name of illness, social security ID number, reserved number, ward room number, bed number, name of IV drip, and prescription by an assigned doctor. All the data are entered in the database of the servo computer of the nursing center and incorporated in a QR code, which is pasted on the drip, the patient's wrist ring, the ward's wall, the bed of the patient, and the drip-detecting control box. The ward and bed numbers are encoded in a X-Bee hardware module and sent to the servo computer through the middle way before the nurse injects the IV drip. To prevent incorrect injection, the information is converted to a QR code using the system software that we designed, which is pasted on the patient's wrist ring as well as IV drip. The nurse uses a smart phone to scan the QR code before performing the injection. After scanning, the information is sent to the nursing center through the designed app in the smart phone for comparison with the data in the servo computer database. If the data is correct, the servo will send a message verifying that this is the correct patient, and the injection can proceed. During the injection process, the remaining quantity of the IV drip is detected by the IV drip control sensor box and the nursing staff is notified if the IV drip is running out and a new drip is required.

The detection system was designed with a load cell to sense the weight of the drip, and the weight is transformed into a tiny voltage signal. Since the remaining quantity of the drip needs to be measured as a volume in the unit of cubic centimeters (cc), a microprocessor is required (here, we used HOTEC Corp. MCU HT66F70A) to convert the electric voltage into digital data using an A/D converter and judged by the CPU. The detected remaining quantity of the drip is transmitted to the SOC middle way by X-Bee wireless sensor network modules, then the SOC middle way (Fig. 15) sends the message to the nursing center management servo computer (Fig. 16) through the Internet. After the patient sees the doctor, all the information of the patient is stored into the management servo computer in the nursing center (Fig. 17). If the patient needs to stay in the hospital and be injected with an IV drip, he or she is led to a ward room and bed by a nurse The operating interface of the software for scanning the QR Code is shown in
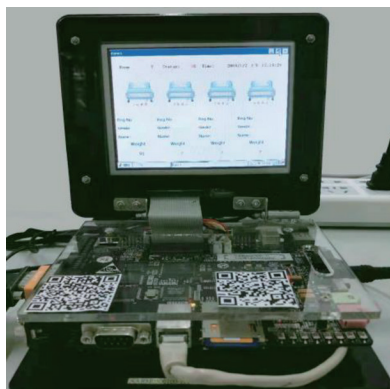

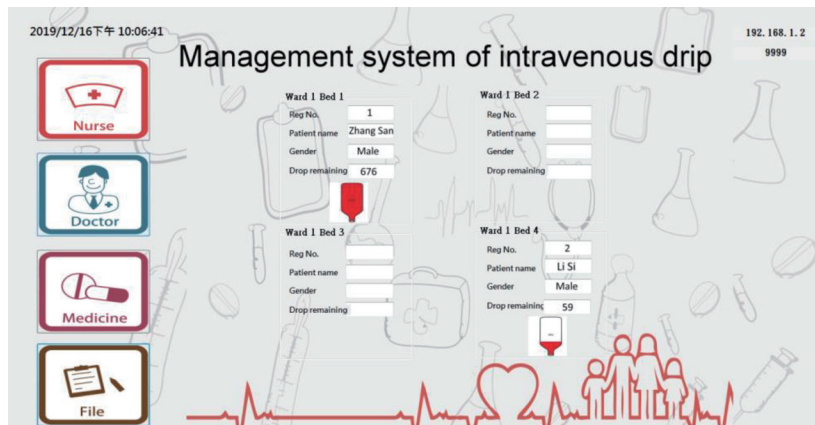Fig. 15.   (Color online) SOC middle way.

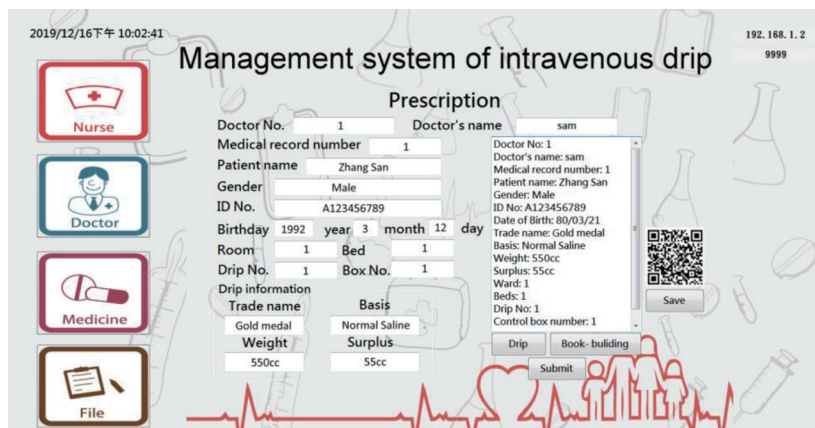Fig. 16.   (Color online) Servo computer in nursing center.

Fig. 17.   (Color online) Patient's information interface on servo screen.

Figs. 2 and 3, and the messages giving the comparison results are shown in Fig. 4. The method of detecting the drip weight is shown in Fig. 18. The whole system of the IV drip is shown in Fig. 19.

According to the above description, there are some requirements of the system software to accomplish the desired functions. The main procedures of the designed servo computer system software are shown in the flowchart in Fig. 20, and the software of the SOC middle way is shown in Fig. 21. The drip weight detection method is shown in Fig. 18. The whole IV drip system is shown in Fig. 19.

The designed VB.NET program of the nursing center management servo is described hereinafter. We need to initialize the working environment of the servo computer and the related variables and parameters.

(1) Configure the communication socket used in VB.NET.

(2) Set the monitoring thread Listen () so as to receive the data transmitted from the SOC middle way or the nurse's mobile phone.
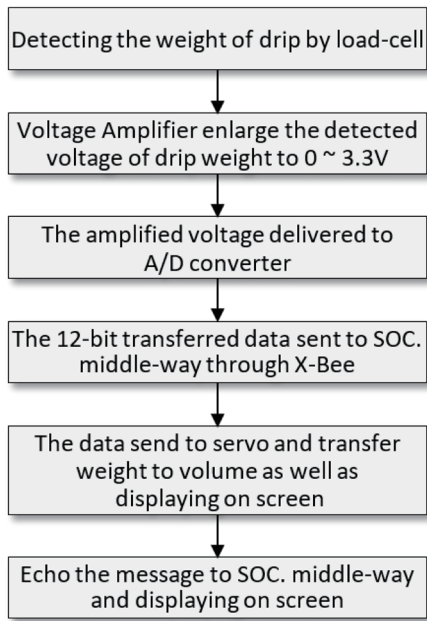
Fig. 18.   Methodology of drip weight detection.
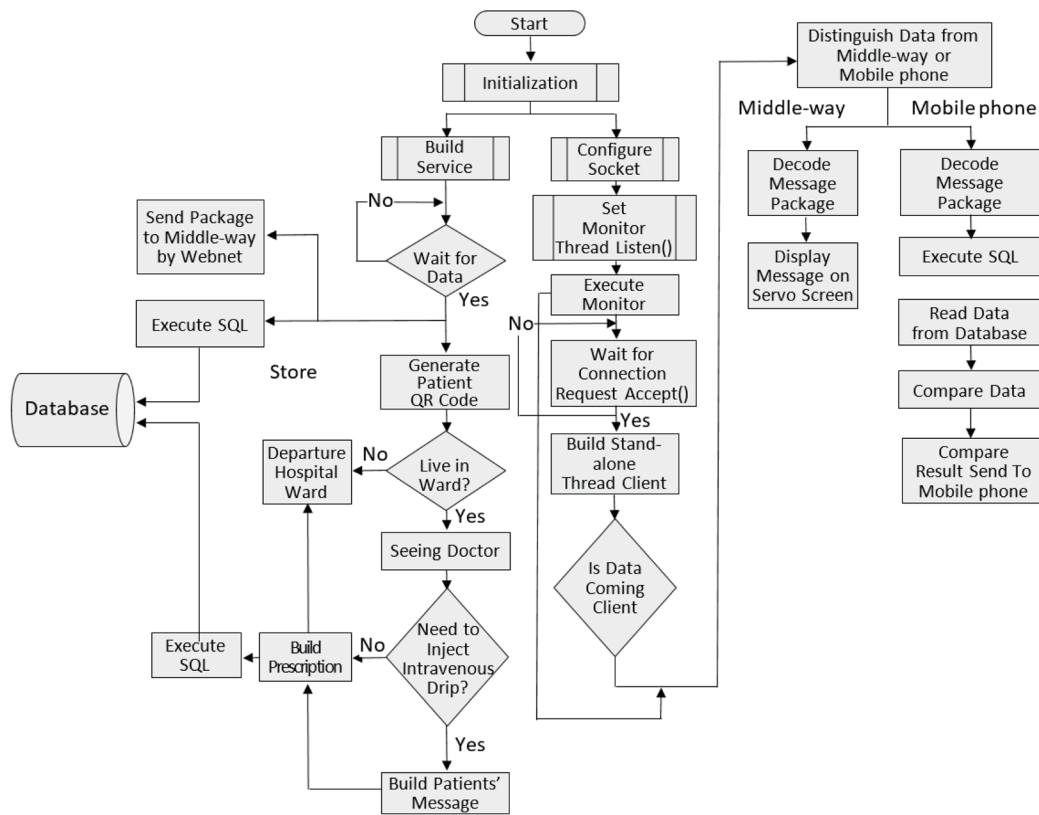

Fig. 19.   (Color online) Whole IV drip system.

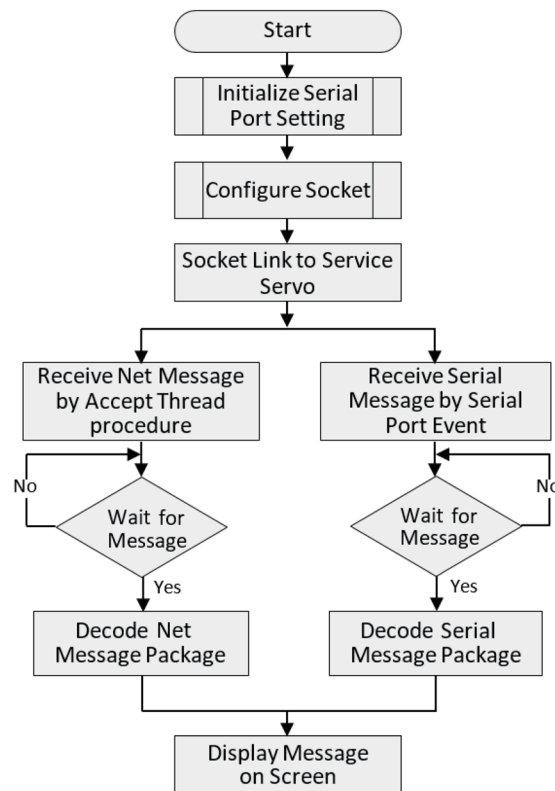Fig. 20.   Flowchart of servo computer program software with VB.NET.

Fig. 21.   Flowchart of SOC middle-way program software with VB.NET.

(3) Execute the monitoring process and wait for the connection request. If connection is requested, build up the stand-alone thread Client Socket ().

(4) Judge whether the data are coming from the Client Socket; if not, keep monitoring. If yes, distinguish whether the data are coming from the middle way or mobile phone.

(5) If the data are coming from the mobile phone, decode the message package, execute SQL, read the data from the database, compare the two kinds of data, then send the comparison result back to the nurse's mobile phone.

(6) If the data are coming from the middle way, decode the message packages, then display the related message on the servo-computer screen as well as on the screen of the SOC middle way.

(7) Next, build a service database including the patient's name, gender, birth, name of illness, social security ID number, reserved number, ward room number, bed number, and name of IV drip.

(8) Wait for the transmitted data until they arrive. Then, send the response message back to the middle way, execute SQL, and generate the patient's QR code.

(9) Does the patient need to stay in the hospital ward? If yes, does the patient need an IV drip? If yes, compile the patient's message and generate the QR code; if not, compile the prescription and prepare for departure from the hospital.

The related design steps are described as follows:

(1) Initialize the serial port of the middle way that receives the data sent from the X-Bee wireless communication module on the drip control box.
(2) Configure the socket of the communication software device.
(3) Link the socket with the distant servo computer in the nursing center.
(4) Receive the serial data from the web net by the accept thread procedure (in VB.NET software). Then, decode the message package (right branch of the flowchart in Fig. 21).
(5) Receive the serial data by serial port event (in VB.NET software). Then, decode the message package (left branch of the flowchart in Fig. 21).
(6) After decoding the message package, all the messages will be displayed on the screen of the middle way.

3.4 Several key steps in design of servo, middle way, and drip-detecting control box

In addition, the program has some important key steps. Because the servo computer needs to connect with the middle way and the mobile phones of nurses, the main servo computer must know its own IP and provide it to the connect web-net. Thus, at the beginning of the program, the main servo must obtain its own IP from the setting value in the system. The related VB.NET part of the program is shown in Fig. 22.

There may be several patients in the wards who need to be injected with an IV drip. In this case, the remaining quantity in all the drips must be detected and the data are sent to the servo through the middle way. In the case of many drip control boxes, the communication may slow down, decreasing the web-net transmission efficiency. The solution to this problem is to design a thread function in the VB.NET program as shown in Fig. 23.

As the servo and each middle way are connected, all the middle ways were clients for the servo computer, so the servo must initialize the network for the clients. The related part of the program for client initialization is shown in Fig. 24.

That part of the program is used to judge the data received from the XBee modules on the drip-detecting control box, which would be collected by the middle way through a serial port as shown in Fig. 25. The data are then sent to the servo. After sending the data to the servo, the data are decoded according to the message package and the servo takes appropriate action. The related program is shown as Fig. 26.

```
Function MyIP() As String
    Dim ip() As IPAddress = Dns.GetHostEntry(Dns.GetHostName).AddressList
    For Each it As IPAddress In ip
      If it.AddressFamily = AddressFamily.InterNetwork Then
        Return it.ToString
      End If
    Next
    Return ""
End Function
```

Fig. 22. (Color online) Part of program for obtaining the IP of the servo computer.

```
Private Sub ServerSub()
    Dim EP As New IPEndPoint(IPAddress.Parse(TextBox1.Text), TextBox2.Text)
    Server = New TcpListener(EP)
    Server.Start(10)
    Do While True
        If gateway = 0 Then
            Client = Server.AcceptSocket
            Th_Clt = New Thread(AddressOf Listen)
            Th_Clt.IsBackground = True
            Th_Clt.Start()
            gateway += 1
        Else
            Client1 = Server.AcceptSocket
            Th_Clt1 = New Thread(AddressOf Listen)
            Th_Clt1.IsBackground = True
            Th_Clt1.Start()
        End If
    Loop
End Sub
```

Fig. 23.   (Color online) Part of program for designing a thread function.

```
Try
    With SerialPort1
        .BaudRate = 9600
        .Parity = IO.Ports.Parity.None
        .DataBits = 8
        .StopBits = IO.Ports.StopBits.One
    End With
    SerialPort1.Open()
Catch ex As Exception
End Try
```

Fig. 24.   (Color online) Part of program for initializing serial port of middle way.

```
mySocket = New Socket(AddressFamily.InterNetwork, SocketType.Stream,
ProtocolType.Tcp)
mySocket.Connect(New IPEndPoint(IPAddress.Parse(TextBox_ServerIP.Text), 9999))
AcceptThread.Start()
send("0")
```

Fig. 25.   (Color online) Part of program for initializing client network.

3.5   Validation of system

After the system was designed, it was implemented in Taiwan Cheng-Shing Hospital to verify its operation. Its operation was proved to be very successful. On the right of Fig. 27, we can see the detection of a drip almost running out of liquid, as also indicated on the drip control box with a red LED (left figure). In contrast, the diagram of the drip on the right of Fig. 28 shows a drip full of liquid. All the information was displayed on the servo screen as well as the middle-way screen.

```vb
Dim Buf(24) As Byte
    Dim theBytes(2) As Byte
    Dim CChl As Int16
    Const LEFT_BYTE = 1
    Const RIGHT_BYTE = 0
    Timer1.Enabled = False
    If SerialPort1.BytesToRead > 0 Then
        SerialPort1.Read(Buf, 0, 25)
        '=======================================================================
        If Buf(0) = 126 And Buf(17) = 0 Then
            a2 = Buf(20)
            a3 = Buf(21)
            theBytes(LEFT_BYTE) = a3
            theBytes(RIGHT_BYTE) = a2
            CChl = BitConverter.ToInt16(theBytes, 0)
            '=======================================================================
            If Buf(0) = 126 And Buf(19) = 1 Then
                Label33.Show()
                Label33.Text = CChl
                SetNetSendData1()
                mySocket.Send(NetSendData)
            ElseIf Buf(0) = 126 And Buf(19) = 2 Then
                Label35.Show()
                Label35.Text = CChl
                SetNetSendData2()
                mySocket.Send(NetSendData)
                ElseIf Buf(0) = 126 And Buf(19) = 3 Then
                Label37.Show()
                Label37.Text = CChl
                SetNetSendData3()
                mySocket.Send(NetSendData)
            ElseIf Buf(0) = 126 And Buf(19) = 4 Then
                Label39.Show()
                Label39.Text = CChl
                SetNetSendData4()
                mySocket.Send(NetSendData)
            End If
        End If
    End If
```

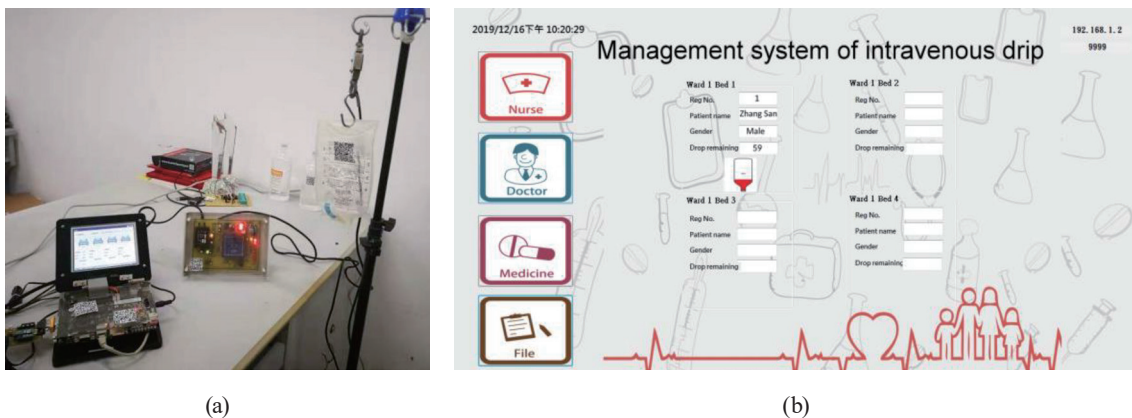Fig. 26.　(Color online) Program code of serial port receive and judge for transmitted data from XBee.

(a)　　　　　　　　　　　　　　　　　　　　(b)

Fig. 27.　(Color online) Detection of drip running out of liquid by control box (a) as shown on servo screen (b).

(a)   (b)
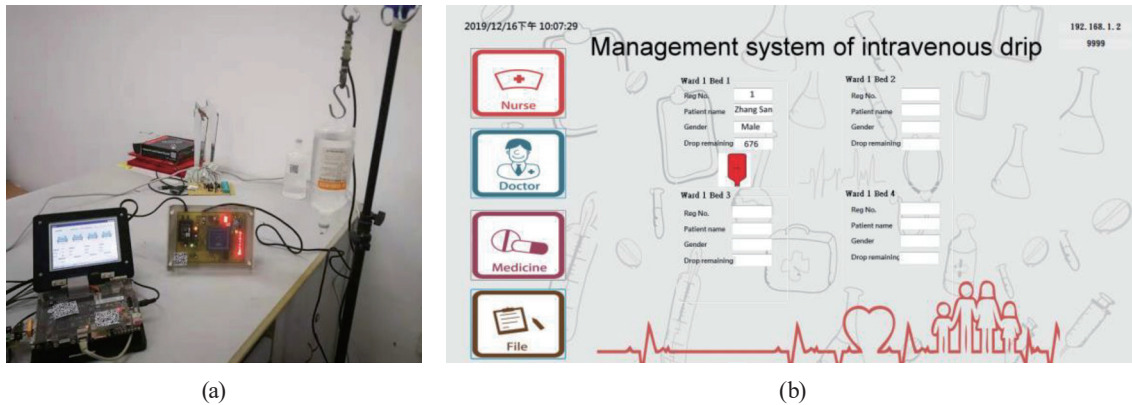
Fig. 28.   (Color online) State of control box (a) and servo screen (b) for full drip.

4.   Conclusions

The outcome of the monitoring of IV drips was proved to be successful, and the validation of the system effect is shown in Figs. 27 and 28. The LED lights on the drip control box indicated the ratio of drip quantity, and the related detected data of the drip and patient were also shown on the screen of the servo computer in the management center. Figure 27 indicates that the drip was nearly running out, whereas Fig. 28 indicates one full quantity drip.

In this study, we have created a novelty by using a serial of QR code, which is generated by the system after patients see a doctor, and this message is saved in the servo computer. When the nurse tries to inject a patient, she needs to compare the message via the designed app in the smart phone with the servo used to identify whether the patient is the correct one before injecting the IV drip. We implemented the smart system in ward management, not only to detect the remaining quantity of the IV drip to improve the work efficiency of the nursing staff but also to provide a protection system for preventing wrong injection that may cause medical disputes. After a real application and operation, the drip weight detection controller proved to be successful and stable. With the development of the X-Bee module in this study, the whole system could be applied to other remote-control systems to carry out data collection and monitoring, and even implementation in many other industry control systems.

References

1   A. L. Mesquida and A. Mas: Comput. Stand. Interfaces **37** (2015) 80. https://doi.org/10.1016/j.csi.2014.06.005
2   F. S. Shi, Y. Zhou, and P. Shi: Appl. Mech. Mater. **599** (2014) 2215. https://doi.org/10.4028/www.scientific.net/AMM.599-601.2215

3   H. Takahashi and T. Kushida: Int. J. Distributed Sens. Netw. **10** (2014) 1. https://doi.org/10.1155/2014/931968

4   S. O. Shim: Adv. Mater. Res. **712** (2013) 3187. https://doi.org/10.4028/www.scientific.net/AMR.712-715.3187

5   I. L'hadi, M. Rifai, and Y. S. Alj: Proc. 2014 5th Int. Conf. Information and Communication Systems (IEEE, 2014) 1-6. https://doi.org/10.1109/IACS.2014.6841980

6   M. T. Lazarescu: IEEE J. Emerging Selected Topics Circuits Syst. **3** (2013) 45. https://doi.org/10.1109/JETCAS.2013.2243032

7   F. M. Al-Turjman, A. E. Al-Fagih, and H. S. Hassanein: Proc. 2012 Int. Conf. Computing, Networking and Communications (IEEE, 2012) 835–839. https://doi.org/10.1109/ICCNC.2012.6167542

8   C. T. Chiang and P. C. Tsai: Proc. 2014 IEEE Int. Conf. Mechatronics and Automation (IEEE, 2014) 342–346. https://doi.org/10.1109/ICMA.2014.6885720

9   Y. Murata and K. Yoshida: Proc. 6th Int. Conf. Health, Telemedicine, and Social Medicine (eTELEMED, 2014) 191–196.

10 A. Cataldo, G. Cannazza, N. Giaquinto, A. Trotta, and G. Andria: IEEE Trans. Instrumentation Meas. **61** (2016) 1866. https://doi.org/10.1109/TIM.2012.2192346

11 T. Kim, J. Kim, and X. Jiang: IEEE Sens. J. **17** (2017) 5741. https://doi.org/10.1109/JSEN.2017.2727340

12 M. Zhang, A. Raghunathan, and N. K. Jha: IEEE Trans. Biomed. Circuits Syst. **7** (2013) 871. https://doi.org/10.1109/TBCAS.2013.2245664

13 S. Ivanov, C. Foley, S. Balasubramaniam, and D. Botvich: IEEE Trans. Biomed. Eng. **59** (2012) 3238. https://doi.org/10.1109/TBME.2012.2208110

14 V. Varadharajan, U. Tupakula, and K. Karmakar: IEEE Access **6** (2017) 11523. https://doi.org/10.1109/ACCESS.2017.2773647

15 H. Zhang, J. Li, B. Wen. Y. Xun, J. Liu, and B. Gosselin: IEEE IoT J. **5** (2018) 1550. https://doi.org/10.1109/JIOT.2018.2792423

16 T. Elfaramawy, C. L. Fall, S. Arab, M. Morissette, F. Lellouche, and B. Gosselin: IEEE Sens. J. **19** (2019) 650. https://doi.org/10.1109/JSEN.2018.2877617