Wireless Hotel Ordering System

AJITH MEMANA AND FAISAL RASAK

Traditional ordering systems consist of waiters handing over the menu to customers and taking orders from them. The waiter's performance is defined by the speed at which the order gets processed, and it might get delayed at times.

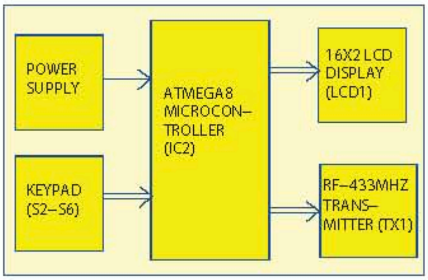

Fig. 1: Block diagram of the wireless hotel ordering system—transmitter side

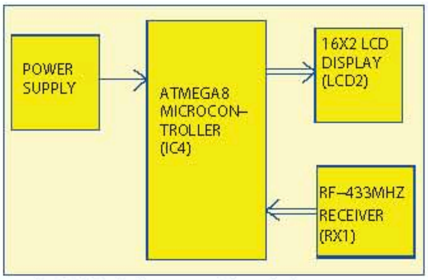

Fig. 2: Block diagram of the wireless hotel ordering system—receiver side

In the wireless ordering system, there is no waiter to take the order. Everything related to the waiter is done by the microcontroller (MCU) and a wireless transmitter. The customer can start ordering as soon as he or she is seated comfortably at the table. The delay caused by the waiter is eliminated in this system.

Circuit and working

The wireless hotel ordering system consists of a transmitter and a receiver section. The block diagram of the transmitter unit is shown in Fig. 1 and the receiver unit in Fig. 2.

The complete circuit diagram of the transmitter is shown in Fig. 3 and of the receiver is shown in Fig. 4.

Microcontroller. The heart of this circuit is an 8-bit AVR ATmega8 MCU that controls, stores and coordinates the activities of the system. The software program stored in the MCU controls the functions of the system.

AVR combines the most code-efficient architecture for C language and assembly programming with the ability to tune system parameters throughout its life cycle. ATmega8 provides the following features: 8k bytes of in-system programmable flash with read-while-write capabilities, 1k byte of SRAM, 23 general-purpose I/O lines, 32 general-purpose working registers, three flexible timers/counters with compare modes, internal and external interrupts, a serial programmable universal synchronous asynchronous receiver transmitter (USART), an SPI serial port and five software-selectable power saving modes.

The flash program memory can be reprogrammed in-system through an SPI serial interface by a conventional non-volatile memory programmer or an on-chip boot program running on AVR core. With AVR's USART you just need to write the data to one of the registers of USART and you are free to do other things, while USART is transmitting the bytes.

Test Points

Test point	Details
TP0, TP4	0V (GND)
TP1, TP5	5V
TP2-TP3	Train of pulse when S4 button is pressed
TP6-TP7	Train of pulse when S4 button is pressed

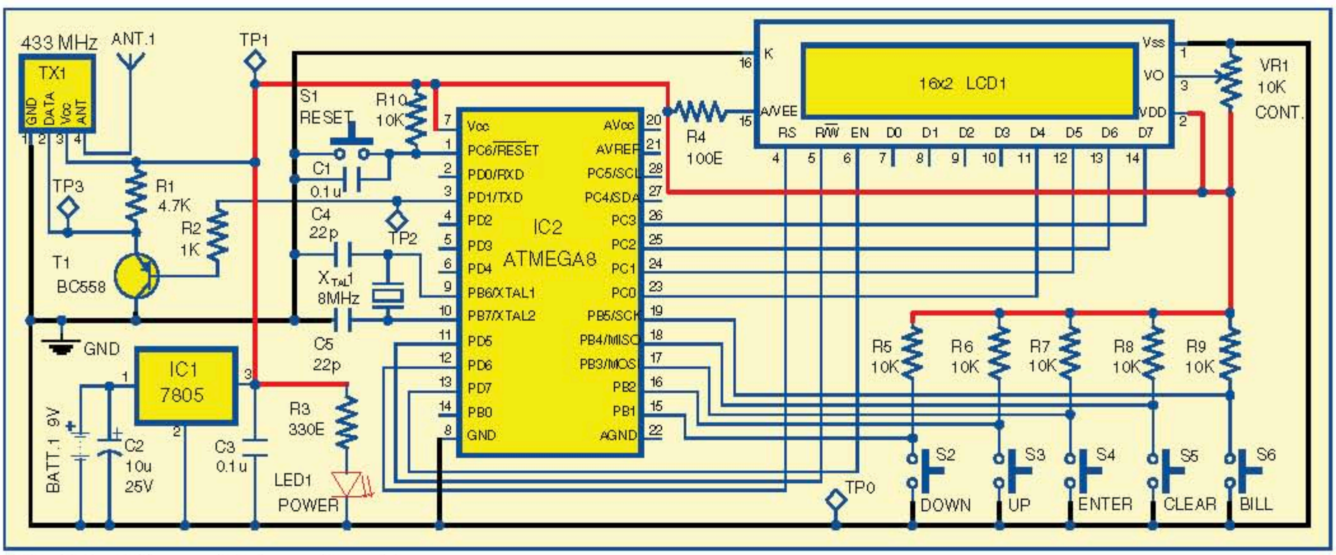

Fig. 3: Circuit diagram of the transmitter section

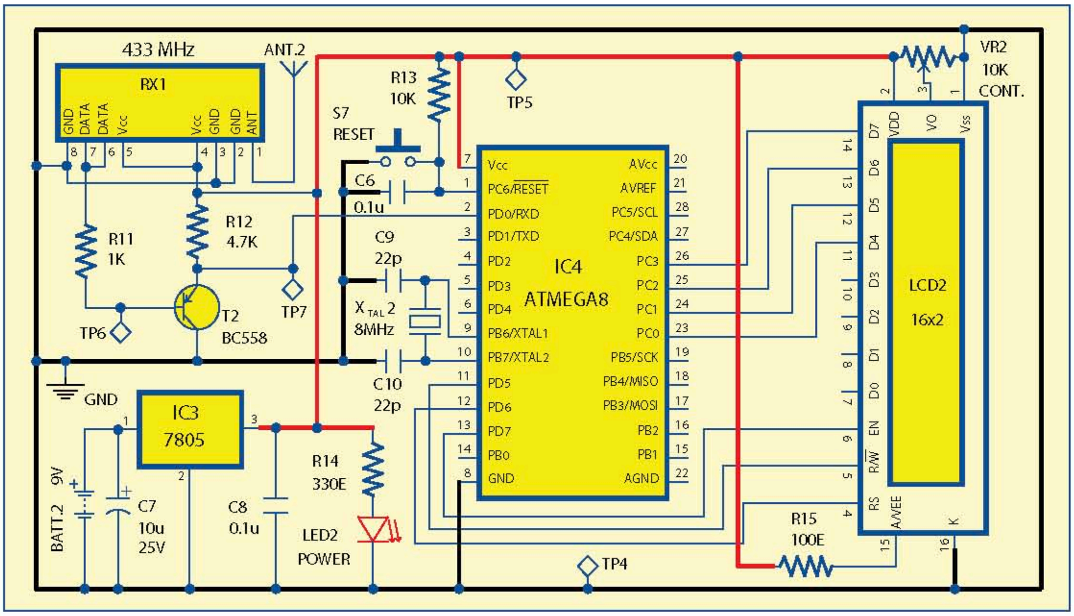

Fig. 4: Circuit diagram of the receiver section

PARTS LIST	
Semiconductors:	
IC1, IC3	- 7805, 5V regulator
IC2, IC4	- ATmega8, CMOS 8-bit microcontroller
T1, T2	- BC558 pnp transistor
LCD1, LCD2	- 16×2 LCD display
LED1, LED2	- 5mm LED
Resistors (all 1/4-watt, ±5% carbon):	
R1, R12	- 4.7-kilo-ohm
R2, R11	- 1-kilo-ohm
R3, R14	- 330-ohm
R4, R15	- 100-ohm
R5-R9, R10, R13	- 10-kilo-ohm
VR1, VR2	- 10-kilo-ohm preset
Capacitors:	
C1, C3, C6, C8	- 0.1µF ceramic
C2, C7	- 10µF, 25V electrolytic
C4-C5, C9, C10	- 22pF ceramic
Miscellaneous:	
TX1	- 433MHz transmitter module
RX1	- 433MHz receiver module
BATT.1, BATT.2	- 9V battery
S1-S7	- Tactile switch
$X_{TAL}1$, $X_{TAL}2$	- 8MHz crystal oscillator
ANT.1, ANT.2	- 17cm single-wire antenna

Keypad interface. The keypad consists of five tactile switches (S2 through S6), where each switch is assigned a particular function for up, down, enter, clear and bill functions in the transmitter unit. There is one reset switch each in transmitter unit and receiver unit, which is used if/when there is a problem in the LCD display.

LCD display module. The 16x2 LCD display is used for the user interface. The LCD is directly controlled by the MCU. The LCD used is a 16x2 (16 characters by two lines) format type, driven by Hitachi HD44780 or an equivalent driver controller. The LCD module can display all alphabets and symbols as per ASCII format, and can store about eight custom characters.

The HD44780 has two 8-bit registers, an instruction register (IR) and a data register (DR). The IR stores instruction codes such as display clear and cursor shift, and address information for display data RAM (DD RAM) and character generator RAM (CG RAM). The IR can be written from the MCU but not read by it.

The DR temporarily stores data to be written into DD RAM or CG RAM and data to be read out from DD RAM or CG RAM. Data written into the DR from the MCU is automatically written into DD RAM or CG RAM by internal operation.

The DR is also used for data storage when reading from DD RAM or CG RAM. When address information is written into the IR, data is read into the DR from DD RAM or CG RAM by internal operation. Data transfer to the MCU is then completed by the MCU reading the DR.

RF transmitter module. The RF433MHz transmitter module is for ASK modulation using an RF transmitter tuned to 433MHz frequency. It features an in-built crystal oscillator and can be used to send bit-wise transmission. Orders given by the customer are encoded and sent wirelessly through this transmitter. The power supply can vary from 4V to 12V, depending on the range and power required. Data pin of the transmitter module is connected to TXD pin of IC2 through transistor T1 and resistor R2. An antenna of 15cm to 20 cm length is desirable for long-range communication.

RF receiver module. The RF433MHz receiver module can work in sync with RF433MHz transmitter. It receives signals sent by the transmitter. Receiver data pins are connected to RXD pin of IC4 through transistor T2. The receiver only needs 5V supply and an external antenna. This module is ideal for short-range remote-controlled applications, where cost is the primary concern.

Power supply. The system requires a regulated voltage of +5V for the MCU, LCD and RF modules. The battery voltage is converted to 5V, required for both transmitter and receiver units using IC 7805 regulator.

Capacitors are used in the input and output of IC 7805 to have more stability and to bypass transient signals. An LED in series with a current-limiting resistor acts as a power-on indicator.

Clock inputs. Crystal oscillators $X_{TAL}1$ and $X_{TAL}2$ are used as external clock sources in this project. Either a quartz crystal or a ceramic resonator may be used. For resonators, the maximum frequency is 8MHz if CKOPT fuse bit is unprogrammed, and 16MHz

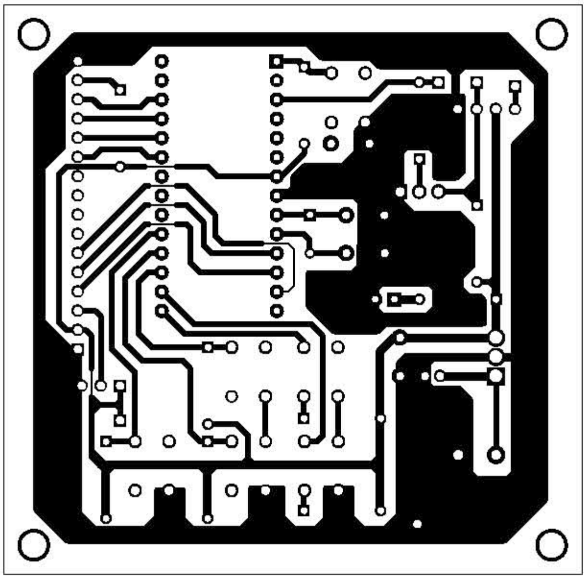

Fig. 5: Actual-size PCB of the transmitter unit

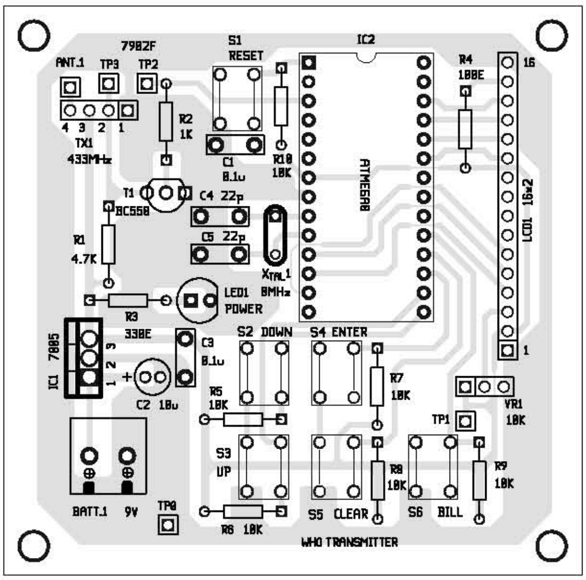

Fig. 6: Component layout of the PCB of the transmitter unit

if CKOPT fuse bit is programmed. Capacitor values for C4 and C5 in transmitter and C9 and C10 in receiver should always be equal for both crystals and resonators. The optimal value of capacitors depends on the crystal or resonator in use, amount of stray capacitance and electromagnetic noise of the environment.

Reset inputs. ATmega8 has four sources of reset, namely, power-on reset, external reset, watchdog reset and brown-out reset. Switch S1 is used as an external reset to reset MCU IC2, when a low voltage level is present on the reset pin for longer than the minimum pulse length as defined in the datasheet. Similarly, switch S7 is used as an external reset for IC4.

The transmitter unit will be energised as soon as the power supply is applied and the MCU will be reset due to the action of power-on reset circuit formed by capacitor C1 and resistor R10. The MCU will initially be in idle state and the LCD back light will be in low brightness in order to save power.

Welcome (in Malayalam) message language will be displayed in the idle state. In this state, the system checks the status of switches S2 through S6 connected to port B of ATmega8. When the user presses any switch, the system breaks from the idle state and jumps to the menu directly.

Now, the program is in continuous main loop. Since the LCD display contains only two lines, it can display only two messages at a time. The current selected item is highlighted by enclosing it within > and < symbols. The LCD display is updated only when a key is pressed, thereby avoiding updates in the loop.

By pressing Up or Down switch, the user can go to Previous or Next menu. The quantity of selected item can also be set using Up and Down. The price of the selected item is also displayed in the second line of the display. If the user wishes to cancel the order, he can press Clear, and "Cancelled.." will be displayed on the LCD and the program will jump back to the main menu.

Pressing Enter will confirm the order, and "Sending.." will be displayed on the LCD and a code will be generated internally. This code will contain the item code and the number of items ordered. This code is transmitted serially using USART communication technique in AVR. After sending, "Sent" is displayed on the LCD and the program jumps to the main menu.

By pressing Up or Down, the user can review the order, including items ordered and the quantity. After ordering the required items, if the user wishes to view and verify the bill, he or she can press Bill and it will show the total amount in rupees.

Game. While the user waits for food, he or she can play Shooter game by pressing Game. The game features three custom characters and Enter shoots enemies.

Software program

Programming of the AVR is done using embedded C language. It is similar to C language but includes all functionalities of C as well as access to AVR pins, peripherals and controls. The Cross compiler used here is AVR-gcc, which is an open source compiler.

Hex codes generated are burnt into the MCUs for transmitter and receiver units. Working of the program is explained as comments in transmitter and receiver source codes.

The program jumps to the main function where the object code actually starts. Enter key causes the program to jump to the sub-loop of ordering. Pressing Enter will confirm the order and the entered quantity is stored in the item quantity variable in the program. The key de-bouncing problem is eliminated by the software by adding some delays for every key pressed.

Construction and testing

An actual-size, single-side PCB of the transmitter unit is shown in Fig. 5 and its component layout in Fig. 6. An actual-size, single-side PCB of the receiver unit is shown in Fig. 7 and its component layout in Fig. 8.

You can also assemble the circuit on a general-purpose PCB. Before mounting the MCUs on the PCBs, burn the respective codes into the MCUs using a suitable programmer board. After mounting all components on the transmitter and receiver PCBs, switch on the power supplies on the respective units. LCD1 in the transmitter unit will display Welcome (Fig. 9) and LCD2 in the receiver unit will display Waiting 4 Orders (Fig. 10). Presets VR1 and VR2 are used for contrast control of the LCD displays. If there is any display problem, press reset switch momentarily and check the display again. For further troubleshooting the circuit, check voltages at various test points as given in the table.

Ordering procedure

1. Assuming the transmitter unit is installed near the table where the customer is seated, when he or she presses a key, the menu appears. The user can browse items/menu using Up/Down keys. A typical menu is shown in Fig. 11.

2. If the user wants a particular item, he or she can select that item by pressing Enter, along with the quantity and again press Enter to confirm the order.

3. If the user wants to cancel the order, he or she can press Clear/Cancel.

4. After confirming the order, the item and quantity are sent to the receiver through RF wireless communication. The item and quantity are displayed on the LCD in the receiver unit. The receiver can be installed in the kitchen or in the room where the hotel manager sits.

5. The bill amount is calculated instantly after every order. The user can view the bill and verify at any time by pressing Bill.

6. While the user is waiting, he or she can play the game by pressing Game. The game is shown in Fig. 12.

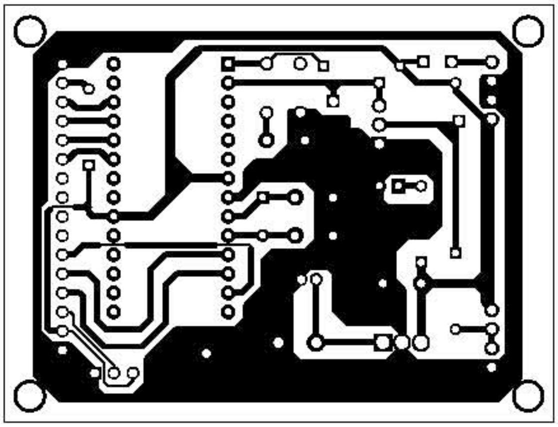

Fig. 7: Actual-size PCB of the receiver unit

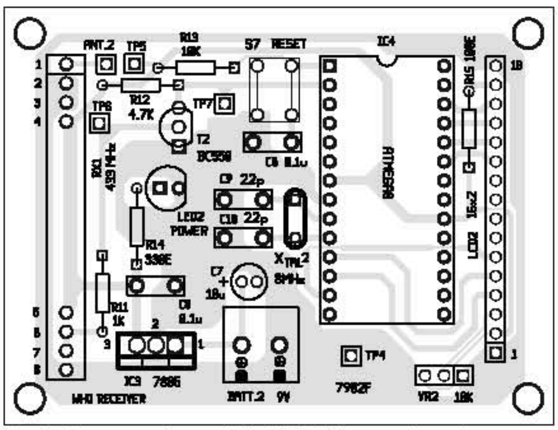

Fig. 8: Component layout of the PCB of the receiver unit

Future scope

The cost-effectiveness and scope for extensive expansion proves to be effective not only for small and medium restaurants but also for premium and luxury hotels.

1. This technology can be coupled with RFID tags (like ID cards used in colleges or offices) for advance ordering of food items in college and office canteens.

2. Data analysis can be done by connecting to an external data processing machine, for example, a regular personal computer.

3. Extra voice channel transmission for special interaction can be done with the chef.

4. Automated serving system (pneumatic/robotic) coupled with this system can be implemented for a completely automated hotel. ●

Fig. 9: Welcome message in the transmitter unit

Fig. 10: Message in the receiver unit

Fig. 11: Typical menu on the LCD

Fig. 12: The game

EFY Note

The source codes of this project are included in this month's EFY DVD and are also available for free download at *source.efymag.com*

Ajith Memana is B.Tech (electronics and communication) from FISAT, Mookkannoor/MG University. He is an Android developer at Qburst Technologies, Infopark, Kochi

Faisal Rasak is also B.Tech (electronics and acmmunication) from FISAT, Mookkannoor/MG University. He is a game engineer at Tuttifrutti Games, Infopark, Kochi