USING SMS MESSAGES FOR THE TRANSMISSION OF DIGITAL AND ANALOGUE SIGNALS ACQUIRED BY S87C552 MICROCONTROLLER

Petruț DUMA

Faculty of Electronics and Telecomunications,"Gh.Asachi" Technical University Iaşi, Bd. Carol I, no.11, Iaşi, 700506, Romania, e-mail: pduma@etc.tuiasi.ro

Abstract 4 **The paper describes the hardware structure of a transmission system for digital and analog signals that uses SMS messages and is based on an application system equipped with a S87C552 microcontroller and on a SIEMENS M55 mobile phone with modem. The command program performs an initialization, peripheral circuits programming sequence, digital and analog signals acquisition and testing, communications setting with the mobile phone, data packing, called number packing, setting the other fields in order to create the SMS message according to "Protocol Data Unit". If the digital/analog signals have values that trigger at least one alarm, the microcontroller uses AT commands to send, through its serial interface, a SMS message with these values to the mobile phone.**

Keywords: *mobile phone, SMS message, protocol data unit, application system, S87C552 microcontroller, digital and analog signal acquisition.*

1. INTRODUCTION

In order to transmit a small volume of digital and analog signals from a microcontroller driven application system to another, standard SMS messages used in GSM mobile phone systems can be used. The basic structure of a SMS transmission system controlled by an application system equipped with a S87C552 microcontroller is shown in fig.1.

The system consists of an interface for digital and analog signals acquisition (I-D&ASA) from the external medium, the application system equipped with S87C552 microcontroller (AS-µC), a RS232 interface used for communication between the application system and the GSM mobile phone (I-RS232), and the mobile phone itself (MP). The application uses a SIEMENS M55 mobile phone that includes a modem and performs serial asynchronous SMS message transmission/reception in PDU (Protocol Data Unit) mode. The application system communicates with the mobile phone using AT commands and an implementation of the "Protocol Data Unit".

2. THE STRUCTURE OF SMS MESSAGES TRANSMITTED IN PDU MODE

The SMS message according to PDU mode consists of a sequence of hexadecimal values that include the service centre number, followed by a chain of packets used by the transport layers of the SMS known as the transport protocol data unit – TPDU. This protocol includes various fields that are added to the message itself. Most data from these fields are packed in different manners in order to insure the transmission of SMS messages. The structure of a SMS message transmitted in PDU mode is presented in fig. 2.

The meanings of the acronyms used are:

-SCA service centre address, i.e. the number of the SMS service centre from the operator's network. It is not required by some mobile phones. It can have a hexadecimal value between 00 or 01 up to 80 that designates "unknown", situation when the phone will use the number stored in its settings;

-ID the first byte of the SMS – subunit message, TPDU type identifier. It consists of SMS identifiers and flags that require a status report or the presence of field VP;

-MR reference message. Any value set between 00 and 255 allows the phone to define the message reference;

-DA destination address, the telephone number of the called subscriber;

-PID protocol identifier that states the nature of the data transported;

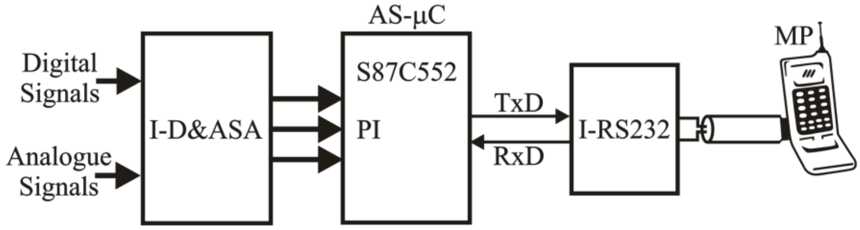

Figure 1: Structure of a SMS transmission system

SCA	ID	MR	DA	PID	DCS	VP	UDL	UD
	SMS – SUBMIT TPDU							
PDU								

Figure 2: Structure of a SMS message

-DCS data coding scheme that sets the format of the transported data and their place of storage;
-VP validity period - is the period during which the sent messages are kept by the service centre of the network operator;
-UDL user data length (message length);
-UD user data, i.e. the data message adequately packed.
The telephone number packing method for the service centre address SCA and the destination address consists in creating a field with the following structure: the telephone number length assumed as the field's length in bytes, including TA; the address type byte (TA) containing the telephone number format; the telephone number divided into pairs of digits for each byte, the digits switched inside each byte. If the number of digits of a telephone number is odd, the last pair is completed with hexadecimal digit F.
Packing the user data messages consists of coding the 7-bit data into bytes. A SMS message sent by a mobile phone has, according to ETSI specifications, a length of no more than 140 bytes. The GSM alphabet used for SMS messages has 7 bits for each character. Therefore, using message packing, 140*8/7=160 characters can be sent. Packing the user messages is performed as follows:
- the first byte includes the first 7-bit character stored in the least significant bits, while the most significant bit includes the least significant bit of the second character;
- the second byte stores the remainder of the second character into the least significant 6 bits, the most significant two bits being used for storing the two least significant bits of the third character;
- accordingly, the third byte includes the remainder of the third character stored in the least significant 5 bits, while in the most significant three bits are stored the 3 least significant bits of the fourth character and so on.
If the last byte of the user data message has any unused bits, they are reset. An 8 character message packed in this manner will have a length of 7 bytes.

3. THE HARDWARE AND SOFTWARE STRUCTURE OF THE SYSTEM

The hardware structure consists of an application system equipped with S87C552 microcontroller that acquires digital and analog signals and communicates through the serial interface with a SIEMENS M55 mobile phone.
The application system is equipped with S87C552 microcontroller and has the structure shown in fig.3.

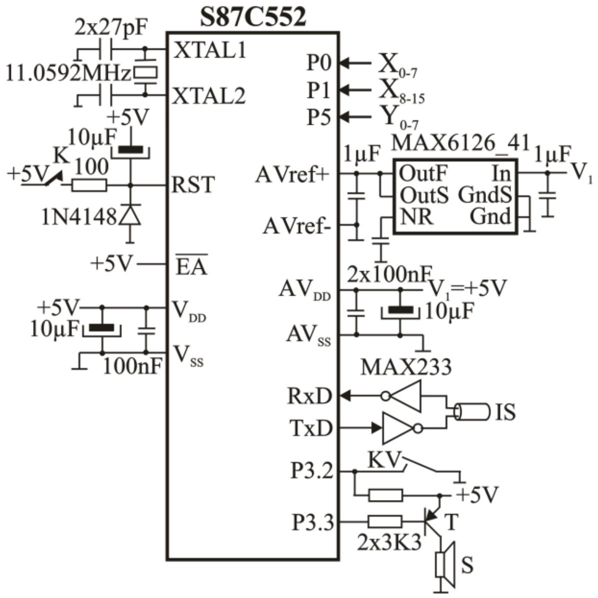

Figure 3: Application system

The used microcontroller includes on a single chip the following features and resources: microprocessor optimized for command and control applications; EPROM internal program memory of 8 Kbytes (allows the selection of 64 Kbytes external program memory space); 256 bytes RAM internal data memory; memory space for special function registers (SFR); allows the selection of 64 Kbytes external data memory space; five 8-bit parallel input/output ports (P0 - P4) plus one 8-bit parallel input port (P5) shared with analog inputs; 10-bit analog to digital converter (ADC) with eight multiplexed analog inputs; two 16-bit timer/event counters (T0, T1); 16-bit timer/ event counter (T2) coupled to four capture registers and three compare registers; programmable Watchdog timer (T3); two 8-bit resolution, pulse width modulated output channels (PWM); serial asynchronous data communications unit (UART); bus serial input/output port with byte oriented master and slave functions (I^2C); interrupt system, with fifteen sources on two priority levels; two software selectable modes of power reduction (idle mode and power-down modes); internal clock oscillator; etc.
The application system includes, besides the microcontroller chip, a 11.0592 MHz quartz crystal and two capacitors for the clock generator, a 10 µF capacitor, a 1N4148 diode and switch K for initialization and de-coupling capacitors (100nF,10µF).

The command program of the application system is placed in the internal program memory of the microcontroller which requires $\overline{EA}$ pin to be connected at +5V.
Input RxD used for data reception and output TxD used for serial data transmission have standard logical levels. A RS232 interface for level conversion must be used to communicate with the mobile phone. In this respect, a MAX233 circuit is used that includes both the line receiver and transmitter buffers on a single chip.
A speaker (0.25 W) is connected to the line P3.3 through a T transistor in order to audio signalize the transmission of the SMS messages, the digital or analog signals changes, eventual communication errors etc.
Input digital signals X_i, i=0÷15, are acquired from limiting circuits, sensors, transducers etc. These digital values are delivered to the microcontroller's input ports P0 and P1.
Input analog signals Yj, j=0÷7, are acquired from sensors, transducers etc. These analog values are delivered to the microcontroller's input port P5. These port's lines are internally connected to an eight-input analog multiplexer with the output line connected to a 10-bit successive-approximation analog-to-digital converter. The multiplexer and the converter are both controlled by software, managing registers ADCON and ADCH.
The reference voltage for the converter is obtained from the integrated circuit MX6126-41, a ultra high precision and ultra low noise voltage regulator of 4.096V.
The analog signals to be measured can vary between A_{Vref-}=0 and A_{Vref+}=4.096V. In order to acquire signals of higher voltage than the reference voltage, it can be used a resistive divider that translates the voltage in the required range, while for signals with both polarities, the signal can be added a DC component followed by a division.
After the application system is powered on and the microcontroller is initialized, the program sequence that initializes the application is run. This sequence consists of initializing the system and communication variables, programming the serial interface and counter T1 that sets the baud rate, checking the communication with the mobile phone, selecting the mobile phone memory, erasing received SMS messages etc.
Then, the software starts a main program loop that performs the acquisition of the digital and analog signals, checks if the digital signals have the pre-defined values and if the analog signals are within the required range.
The alteration of the digital signals triggers the execution of a software parasite removal program sequence that eventually saves the stable digital values of the inputs into the internal data memory. The alteration of the analog signals also triggers the execution of a program sequence that checks if the signals are maintained outside the defined boundaries for a specified amount of time, then saves the analog inputs, in digital form, into the internal data memory. After the sequence is concluded, an adequate warning tone is transmitted, the digital/analog signals are checked individually or in groups and the main program loop is resumed.
If, after checking the digital/analog signals, at least one alarm value is detected, the system runs a program sequence that transmits an alarm tone, followed by the transmission of a SMS message containing all the alarms detected, then the main program loop is resumed.
The values of the digital signals can be transmitted binary or in hexadecimal, while the values of the analog signals are transmitted in decimal and ponderated with the adequate constants in order for the result to be expressed in mV. The values for both signal categories are converted into ASCII codes and merged with relevant short text messages, in order to be made easy to understand at destination. Transmitting a SMS message consists of data packaging, destination phone number packaging, setting the other fields and writing the SMS message in PDU format into the internal data memory of the microcontroller.
Using the AT command for sending SMS messages, the hexadecimal digits of the message are converted to 7-bit characters according to the default alphabet and then transmitted serially asynchronously to the mobile phone using a baud rate of 9600 bps.
The syntax of the AT command for modem SMS transmission is:
AT+CMGS=<lng><CR><PDU><ctrl+z>
where: <lng> - the length of the PDU message; <CR> - command separator; <PDU> - the sequence of hexadecimal values of the SMS message in PDU format; <ctrl+z> - command terminator.
If the message is transmitted correctly, the phone replies:
+ CMGS=<mr>
where <mr> is the reference message number.
If the message is transmitted wrongfully, the phone replies:
+CMGS ERROR

4. CONCLUSIONS

The transmission system for digital and analog signals using SMS messages on a mobile phone has a minimal hardware structure, it has been built in practice and can be used in various applications required to send a small amount of data.
Figure 4 presents the system that was built in practice, consisting of a development system equipped with microcontroller S80C552, the board used for the

acquisition of 16 digital signals provided by switches and 8 analog signals collected from multi-turn presets connected through (user controlled) repeaters, and the mobile phone SIEMENS-M55. The system acquires continuously data that are sent automatically through SMS messages when not within the user-defined range.

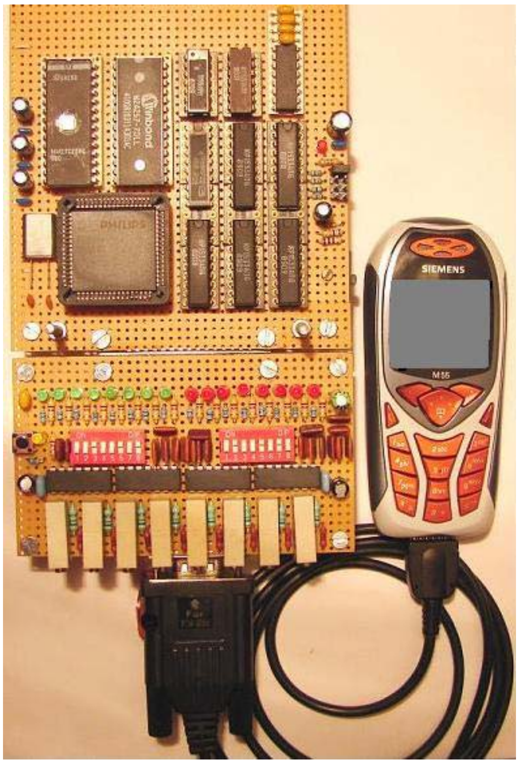

Figure 4: Application system

If many digital signals must be transmitted and the microcontroller's ports are not sufficient, additional input parallel ports can be used to acquire these digital signals. These ports are implemented with tri-state buffers and selected by a decoder. If multiple analog signals must be transmitted, they can be acquired using 8-line or 16-line analog multiplexers, connected to port P5.

For signals acquired from an external medium with power stages, high amperage or high voltage, galvanic isolation must be used.

The command program is written in assembly language, requires a program memory area of some 3 Kbytes and may be easily extended for other applications.

If needed, the recipient users of the SMS messages can use a specific SMS message to require a complete report that includes all the digital and analog signals.

Since the transmission of SMS messages in GSM networks at peak hours may be delayed, this might present as a downside in certain applications of this system.

The system described represents a preliminary stage for designing acquisition systems for signals originating from sensors placed in home and office buildings, including unauthorized entry sensors such as proximity, broken windows, door/window use etc., sensors to detect gas, water, fire, carbon dioxide etc., sensors for enclosed space surveillance and also for monitoring temperature, humidity and atmospheric pressure in such areas.

References

[1] O. Căpățînă, *Proiectarea cu microcalculatoare integrate*, Ed. Dacia, Cluj Napoca, 1992.

[2] P. Duma, *Microcontrolerul I8051. Aplicații*. Ed. Tehnopress, Iaşi, 2004.

[3] J.K. Hintz, D. Tabak, *Microcontrollers. Arhitecture, Implementation and Programming.* McGaw Hill, N.Y., 1993.

[4] B.J. Peatmann, *Design with Microcontrollers*. McGraw Hill, N.Y., 1998.

[5] D. Somnea, T., Vlăduț, *Programarea în Assembler*, Ed. Tehnică, Bucureşti, 1992.

[6] xxx *SMS Format*.

[7] xxx *SIEMENS,* AT Command Set for Siemens Mobile Phone and Modems, 2003.

[8] xxx *SIEMENS,* M55 Mobile Phone.

[9] xxx *MAXIM, RS232 Drivers/Receivers*. Data Sheet, 2001.

[10] xxx *MAXIM, Ultra High Precision, Ultra Low Noise, Series Voltage Reference,* Data Book, 2004.

[11] xxx *PHILIPS, S87C552 Microcontroller*. Data Sheet, 2002.

[12] xxx *TEXAS INSTRUMENTS,* Data Book, 2002.