Mini Device for Environmental Forensics

CALIN CIUFUDEAN, PINZARIU CIPRIAN
Faculty of Electrical Engineering and Computer Science
"Stefan cel Mare" University of Suceava
9 University str., 720225, Suceava
ROMANIA
http://www.eed.usv.ro

Abstract: - Comprehensive, accurate, costless and timely information about climate conditions are critical to sound decision making the evolution of life on Earth.
Our project focuses on the capability to collect, synthesize, interpret and manage complex information about environmental and human-made contaminants and the condition of the environment.

Key-Words: - Environment protection, data acquisition, microcontroller, remote controlled device.

1 Introduction

It is well-known that nowadays climate changes concern humanity and ultimately determine the evolution of life on Earth. The capacity to collect, synthesize, interpret, manage, disseminate and understand complex information about environmental and human-made contaminants and the condition of the environment for both the academic world and the community. Therefore, we consider that comprehensive, accurate, costless and timely information about climate conditions are critical to sound decision making.

Our project focuses on the capability to collect, synthesize, interpret and manage complex information about environmental and human-made contaminants and the condition of the environment. We believe that effectively managing and sharing this information within the decision staff at all levels of government and industry will contribute to the capability to detect, prevent, protect from and recover from environment perturbations due to natural or human-made contaminant factors.

2 A Device for Environmental Forensics

This paper focuses on one of our projects entitled "Mini Device for Environmental Forensics (MiDEF)" concerns an automated supervising system for the air and soil temperature, humidity, PH, the diurnal and nocturnal lighting, the force and direction of the wind.

The basic scheme of MiDEF is shown in figure 1.

MiDEF supervises 6 sensors: respectively, the measuring of temperature, PH, light intensity, intensity and direction of the wind. Our device memorizes, in the EEPROM (see figure 1), the values collected from the sensors and interpreted by the microcontroller unit (MCU). Due to the fact that most of the time we have to deal with slow processes the sensors are scanned at a programmable time, between 1 minute and 6 hours. The sample time is chosen according to the monitored period (e.g., for a few days or a few weeks).

The EEPROM memorizes the exact date and hour when the sensor's capture begins, so thanks to Real Time Clock (RTC), the MiDEF will mark exactly the EEPROM's locations (which will be left intentionally blank) corresponding with a possible period of pause, due to a device fault or to a long absence of electric supply. After the fault if repaired or the electric supply is reconnected, the MiDEF continues its normal activity, the data collected from sensors being memorized by EEPROM in the first location after the blank zone.

We notice that our device (MiDEF) is supplied by a battery that ensures autonomy for 48 hours and the RTC is supplied by a one-year back-up 3V Li battery. The most significant electricity consumption for MiDEF appears when the 7 segment display is turned on and the data are transferred to a PC.

MiDEF has two working modes:

- Active, when all the functions are available (including the PC communication, the sensors collect data and the interpreted values are shown on the 7 segment display)
- Sleep, when most of the functions are disabled, apart from the RTC, Power Control block and the MCU which is in Sleep mode.

When the supply is turned on, MiDEF will work in the Active mode. Before shutting down the MiDEF, an algorithm for collecting data from sensors was planned. The algorithm will be respected after the device restarts. For example, if MiDEF is programmed to collect data every hour, the MiDEF will do so even when it is started at any given time past the scheduled hour.

There are a few events which determine the device to switch from Sleep mode to Active mode:

- It is the time to collect data;
- A signal fro the Power Control block informs MCU that the device was switched on back-up supply and the event is noted in a log;
- A signal from the Power Control block informs MCU that the device was switched from back-up supply to normal supply;
- The On/Stand by button was pushed (e.g., we wants to start transferring data from MiDEF to a PC or we want to display the collected data on the 7 segment display of the MiDEF).

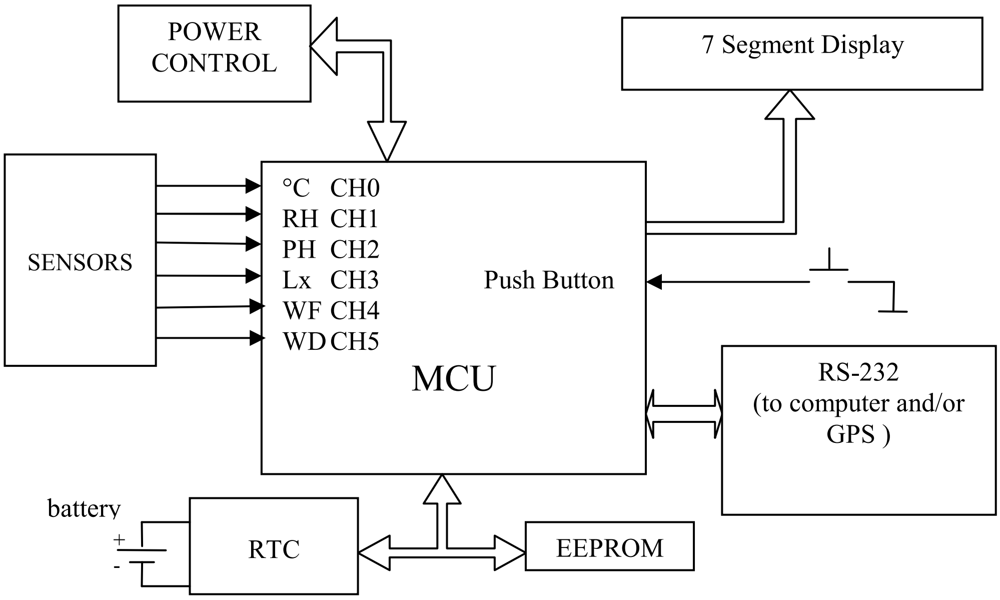

Fig. 1 – Schematic representation of MiDEF

There are also a few events which determine the device to switch from Active mode to Sleep mode:
- A data acquisition is completed;
- An event from the Power Control block was processed by MCU;
- The PC sent the command Sleep to MiDEF (this command is useful when we want to disconnect MiDEF from the PC, the RS-232 interface is off, so we may unplug the cable safely)
- The On/Stand by button was pushed (e.g. we want to stop the data transfer from MiDEF to the PC or we want to stop displaying data on the 7 segment display of MiDEF).

Our software allows a PC to program (when connected to MiDEF) the data sampling frequency and also allows us to verify the functionality of MiDEF (for example setting on or off the 7 segment display). We mention that the sampling frequency cannot be changed during the data capturing time because this operation will determine the overwriting of data in EEPROM, starting from Offset 0. A data acquisition is made at the scheduled hour as follows:
For 16 seconds we have 16 measurements and at in the end the average value of these 16 interpreted values is memorized (e.g. if we measure the temperature, we memorize the average value in Celsius degrees, not the average value in voltage received from the sensor). The very first measurement is made after one second (considering the moment we start the data acquisition) in order to allow the sensors to stabilize the measured data. This is useful especially when MiDEF is in Sleep mode and it follows the data acquisition process. MiDEF will automatically enter the Sleep mode after average data are stored in the EEPROM.
If the data collecting period was exceeded and the EEPROM is full, only the latest sample will be over written. The supply of device is flexible, with one of the following two modes: $8V_{a.c.}$– $16V_{a.c.}$ (typical transformer $220V_{a.c.}$ – $12V_{a.c.}$, 500mA); $8V_{d.c.}$– $20V_{d.c.}$ (typical $12V_{d.c.}$); the battery voltage for RTC: 2,5 – $3{,}5V_{d.c.}$ (typical $3V_{d.c.}$).

3 About MiDEF's Hardware

The brain of our device (MiDEF) is a microcontroller PIC16F877. On the I2C bus the EEPROM 24C64 and

the RTC DS1307 are connected. The serial interface is MAX 232. Two of the eight analogical channels of PIC 16F877 are used for the voltage references (V_{REF1}= 2.00V and V_{REF2}=4.55V). Thus, on the 256 divisions between the two references there will be 1^0 Centigrade, by using only 8 bits of the Analog-Digital Converter (ADC) of PIC16F8777. The supply voltage is stabilized by a LM7805 Texas Instruments circuit, which can be successfully used in difficult working conditions (large humidity and temperature variations). The 2V and 4.55V voltage references are obtained by means of a resistive voltage divisor whose thermo drift is perceived in the least significant of the 10 bits of the ADC (only the most significant 8 bits are used, in order to avoid the noise induced by the sensors). A transistor is polarized to open at the lowest supply voltage on the input of the LM7805 voltage stabilizer. This transistor will determine the interruptions of the MCU. One communicates serially with the display device separately from RS232 and I2C.
The working frequency of the MCU is 20 MHz. The serial port is configured as follows: 9600 Baud rate, 8 bit dates, 1 bit stop, no parity, no flow control (only RX, TX, GND), clock on I2C bus is 100KHz.

4 About MiDEF's Software

The soft we built for MiDEF is a small kernel, capable of reacting in real time at external or internal events. After the initialization one enters an infinite loop in which one waits for the signalizing of a flag in the flag register. After exiting a branch one restarts testing the flags in order to consider the priorities when dealing with events. In the meantime, the interruptions are active. The flags are set at the end of a complete operation. More interruptions can be generated until an operation is completed.
Example: receiving a block of bytes from the PC, writing a block of bytes in the EEPROM, processing the 16 measurements. PIC16X is a microcontroller with Arithmetic Logical Unit (ALU) on 8 bits. As we need to calculate the voltages of the analogical inputs and the sampling frequency in seconds, the software is able to operate on 32 bits (e.g., operations such as sums, multiplications and byte shifting). If the multiplication were implemented with repetitive sums it would take too much time. Having sufficient memory, for multiplication operations a cache which simplifies elementary operations is used. Thus, a multiplication of 32 bits at worst, only takes 400 µs at a working frequency of 20 MHz. These software improvements are not necessary when using temperature sensors.
The RS232 communication protocol is conceived as to validate the information transmitted between MCU and the PC. A command sent by the PC is denied by the MCU and sent back to the PC. The computer software checks if the command received by the microcontroller is correct and sends a confirmation to continue processing the command. Each block of bytes is accompanied by CRC 8 byte and an ACK byte sent by the receiver of the block. CRC8 covers a block of 16 bytes. The resources of PIC16F877 are partially spared, thus allowing the upgrading of the system for GPS communications. The code does not exceed 2 KB. The microcontroller software was written in assembler MPLAB. Certain typical operations/modules were tested in an MPLAB simulator. The serial communication test models were made in VISUAL C++. A HyperTerminal was first used for sending commands to MiDEF.

5 Conclusion

We mention that MiDEF allows the storing of complex data taken from the environment and loading them into a computer (e.g. PC, laptop). Our system also has the capability to transmit these data wireless to a server which supervises several MiDEFs displayed on an area with no limit in diameter because the data are transmitted through satellite (GPS). MiDEF offers the possibility to collect data in real time at high accuracy and lower price than when conventional measurement devices are used. One may also notice the mobility of our MiDEF because it weighs only 200 g and can be easily moved whenever wanted in order to cover a larger area for environmental measurements.
The future work of our team will focus on adapting a mobile remote controlled robot which will guide MiDEF in the desired area. Due to the fact that MiDEF can transmit data taken from the environment by satellite the robot will also be driven by a satellite with a GPS device.
Below are two photos, first represents the actual stage of our device (MiDEF), the second shows a graphic witch recorded two temperature recorded inside (CH1) / outside (CH0) our laboratory for 20 hours period with a sampling rate of 15 minutes.

References:

[1] C. Filote, C. Ciufudean, s.a., *Impact of European Aquis on Education in Enviromental Management,* "Steafn cel Mare"University Publishing House, 2006
[2] G. Kavaiya, *Low-power Embedded Systems*, Microchip Technology Inc. Publishing House, 2001.
[4]http://support2.microchip.com/KBSearch/KB_Ticket.aspx?ID=Tt6UJ9A000Z33
[4] J. Singh, *Operations Research*, Penguin Books, Midlessex, England, 1999.
[5] S.Feenstra, Use of Logarithmic-Scale Correlation Plots to Represent Contaminant Ratios for Evaluation of

Subsurface Environmental Data, *Environmental Forensics,* vol.7.,pp.175-185, 2006.

[6] Z.Wang, s.a, Forensic Fingerprinting of Biomarkers, *Environmental Forensics,* vol.7.,pp.105-1465, 2006.