Intelligent Biometric Access Control System with Environment Monitoring

Csaba-Zoltán Kertész, Petre Ogruțan, Iuliu Székely

Electronics&Computers Department
Transilvania University Braşov
csaba.kertesz@vega.unitbv.ro, ogrutan@vega.unitbv.ro, szekelyi@vega.unitbv.ro

Abstract

Biometric access control systems based on fingerprint recognition are becoming more and more popular as they provide a highly accurate personal identification, without the drawbacks of token based identification such losing or alienating the tokens. These systems however need great processing power for the recognition process and usually are employed in systems where a central computer is doing all the necessary processing and controls all the entrances to the building. In this work we propose an embedded fingerprint recognition system optimized for low power processing capable of independent work even with the loss of mains power. Without the central computer controlling all the security systems of the building, the access control system also monitors the other security systems like fire alarms or hazardous materials detectors through a dedicated CAN bus acting as necessary in danger situations.

1. Introduction

Intelligent buildings employ a large number of embedded systems including those responsible with the security of the building and the safety of the inhabitants. Such systems include access control systems (PIN code, RFID tag or biometric identification), effraction detection systems (motion sensors, heat detectors) and environment monitoring sensors (fire and smoke detectors, toxic or flammable gas transducers) [1].

Most of these systems are interconnected in a single network for increasing security. For example fire and smoke detectors are connected to a central computer that can alert the fire department [2].

Other systems interconnect the environment monitoring sensors, like gas detectors to a central computer, sometimes through a wireless GSM connection for higher reliability [3,4]. These systems however interconnect only partially the sensor systems, usually the access control system is not connected to the environment monitoring system.

However a more integrated system which connects access control systems to safety systems is advisable, as it is important to immediately know who is inside a building in danger situations.

Such systems are presented in [5] and [6]. These systems employ an ATmega16 microcontroller which collects all the signals form a fingerprint recognition systems (for access control) and from a series of CO, methane, propane and radon detectors (for environment monitoring) and sends these informations on a regular basis to a central computer for monitoring and storing in databases.

The system of monitoring some alarm signals from several sensors is fairly simple, but in case of a large building with many sensors can become rather complicated, and often many such systems are needed for different areas of the building. In this case only the central computer is aware of the situation in the whole building.

To overcome this issue we propose a system where all the sensors in building are connected to the same CAN network (offering a high robustness for such implementations), and a microcontroller based system is monitoring the entire communication on the network.

Another key aspect of the system is that the monitoring microcontroller is the very same as the one responsible for the fingerprint recognition. This microcontroller is fairly powerful because the high computational needs of the recognition algorithms, but stands in idle mode for most of the time as identifications are relatively short termed (below 1s) comparatively to the time of entering the building.

2. The Fingerprint Recognition System

The fingerprint recognition system is responsible for the identification of individuals entering the building. It employs a fingerprint sensor for acquiring an image of the fingers pressed against the surface of the sensor, a 32-bit microcontroller for processing the image and 2MB of memory for storing image templates of the authorized fingerprints.

When a finger is detected on the fingerprint sensor, a recognition process is started which generates a template from the fingerprint image which is stored to the templates stored in the system's memory. If a matching template is found the person is identified and signal is generated for opening the door. A log of the person's identity and time of entry is also created.

2.1. The Fingerprint Recognition Hardware

The system is based on a MB91F362GA type microcontroller [7] and a MBF200 fingerprint sensor [8], both from Fujitsu.

The sensor is connected to the microcontroller's external memory interface through an 8 bit parallel bus. The addressing is automatically generated by the microcontroller's external memory interface when accessing the corresponding addresses. The sensor is mapped to the address 0x10000000 (data register) and 0x10000001 (index register), having the Chip Select signal CS5. The block diagram of this connection is presented on figure 1.

The sensor used in the prototype was embedded within a MDFP200 development kit, having all the interface pins connected to a pin header. The sensor is also capable of driving several LED through its internal registers, which are also integrated to this board, and can offer information about the acceptance or rejection of the finger. An image of this development kit is presented on figure 2.

Figure 2. The MDFP200 development kit

When a finger is pressed on the surface of the sensor a finger present interrupt is generated (the corresponding thresholds are set in the initialization routines – after which the sensor enters to sleep mode and has no current consumption). This interrupt signal is connected to an external interrupt of the microcontroller. In the interrupt service routine the microcontroller initiates the acquiring process, and reads in the fingerprint image pixel by pixel. The analog-digital converter is embedded into the fingerprint sensor.

Unfortunately the end of conversion flag for each pixel is stored into a special flags register inside the fingerprint sensor which makes DMA reading impossible. For this reason the microcontroller must continuously work during the reading process. However this reading process can be done in the main routine of the microcontroller, and other interrupts served meanwhile, as the reading process is not time critical.

2.2. The Fingerprint Recognition Software

The fingerprint recognition is done by a minutia matching algorithm, which extracts key features from

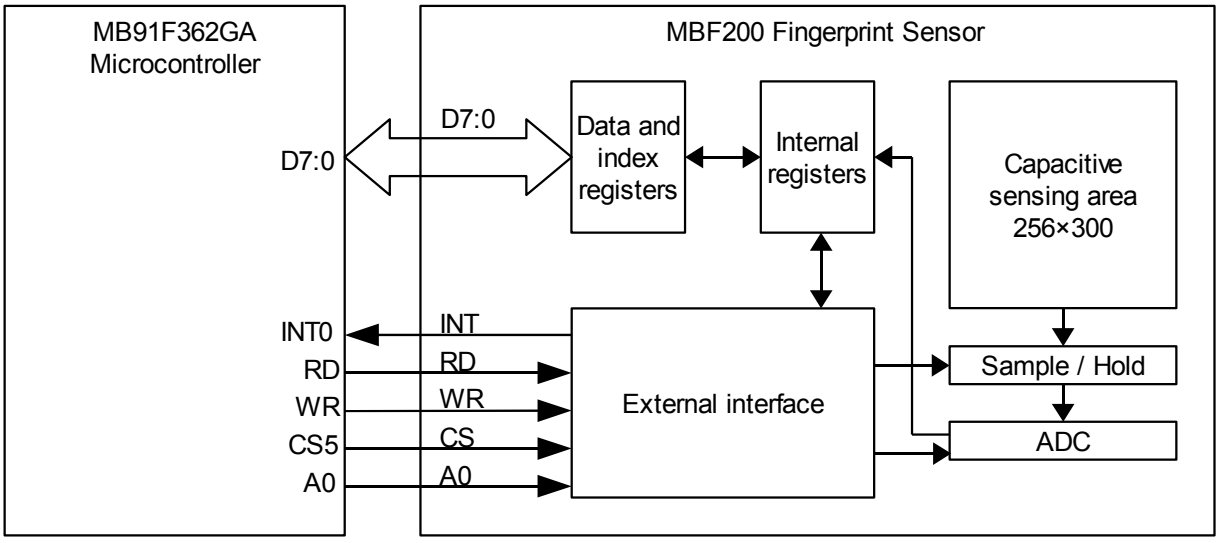

Figure 1. Block diagram of the fingerprint sensor - microcontroller connection

the fingerprint such as ridge endings and bifurcations and stores the positions of these minutiae in a fingerprint template. This template is compared with the templates stored in the memory and a matching score is generated. If this score is above a certain threshold the identity is confirmed [9].

The block diagram of this algorithm is presented on figure 3.

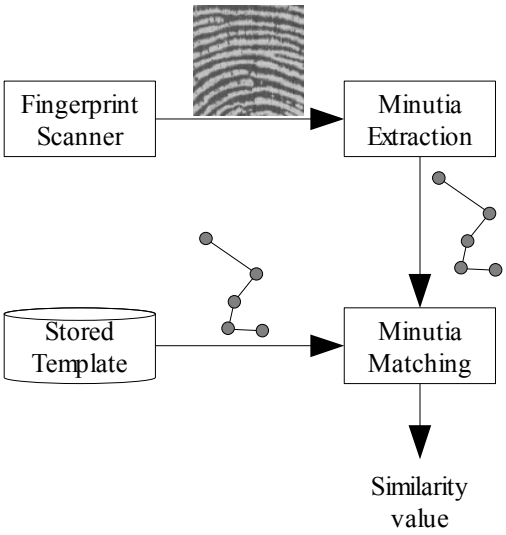

Figure 3. Fingerprint recognition by minutia matching

Because of the lack of a floating point unit in the microcontroller, minutiae extraction is done on a binary skeleton of the fingerprint image, which needs only logic operations.

However the fingerprint images captured by the capacitive sensor is rather noisy and presents some distortions, which inhibit the direct extraction of the skeleton. Because of this the fingerprint image must be enhanced.

For enhancing the fingerprint image a filter must be used which takes into account the properties of fingerprint image, the most important of which is the presence of highly parallel ridgelines with given orientation and frequency. Best enhancement results are offered by the even-symmetric Gabor filter tuned to the local orientation and frequency [10].

The main steps of this method is presented on figure 4. The normalization is necessary to bring all images to the same base. Image information is sensor dependent so the image parameters must be estimated only once for the sensor used.

The orientation and frequency estimation is done by calculating the gradients of the image, from which the orientation can be determined either by using trigonometric functions, or as an optimized version for microcontroller processing by estimating the orientation as one of 8 evenly distributed (22.5° apart) orientations with the method detailed in [11].

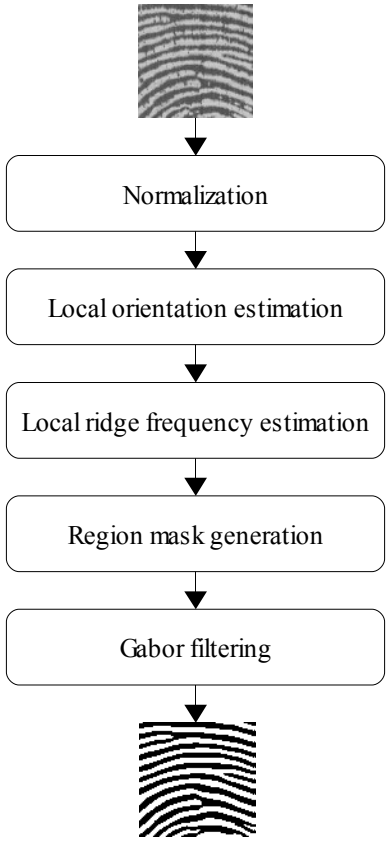

Figure 4. Gabor filtering method

The Gabor filter is defined as

$$G(x,y,\theta,f)=\exp\left\{-\frac{1}{2}\left(\frac{x_\theta^2}{\sigma_x^2}+\frac{y_\theta^2}{\sigma_y^2}\right)\right\}\cdot\cos(2\pi f x_\theta)$$

$$x_\theta = x\sin\theta + y\cos\theta$$

$$y_\theta = y\sin\theta - x\cos\theta$$

where θ is the local orientation, f is the local frequency, σ_x and σ_y are the standard deviations of the Gaussian envelope experimentally determined for the given sensor and x and y are the indices in the filter kernel ranging $-\frac{W}{2},+\frac{W}{2}$, W being the kernel size. Because of the fixed orientations and frequencies estimated the kernel of the filter can be determined at compile time offering good execution time on a microcontroller.

The entire minutia extraction process can be done around 0.8-0.9 seconds using the MB91F362GA microcontroller at full speed (64MHz).

At the end of the minutia extraction process a list of the coordinates of minutiae is resulted. This list is stored in templates and can be compared to other templates. Due to the random nature of the fingerprint acquiring process, these points cannot be matched on an absolute coordinate system, but rather minutiae pairs with their relative distance is used.

Because of this randomness, the matching process can take varying time for execution but usually is in the order of 100ms for each compare. Obviously total matching time is linearly dependent of the number of stored templates. To speed up matching process in case of a large number of authorized persons, a claim-and-verify process can be employed in which case every person must make first a claim about their identity, which is than verified by matching the acquired image to the template of the claimed person.

3. The CAN Interface

The MB91F362GB microcontroller has a built-in 4 channel CAN controller conforming to [12] Part A and B. Each channels have 16 message buffers with separate acceptance filters. Every buffer can programmed as either to transmit messages or to receive messages with a given range of identifiers. At transmission the CAN identifier, the DLC and the data bytes must be specified. At reception the actual identifier (which must be within the specified range), the DLC and the data bytes can be read from the associated registers.

With this interface the MB91F362GA microcontroller can be turned into an independent monitor of the CAN bus [13], which is able of receiving any kind of message.

The microcontroller is connected to the CAN bus through a driver (PCA82C250) and a standardized DSUB9 connector as presented on figure 5.

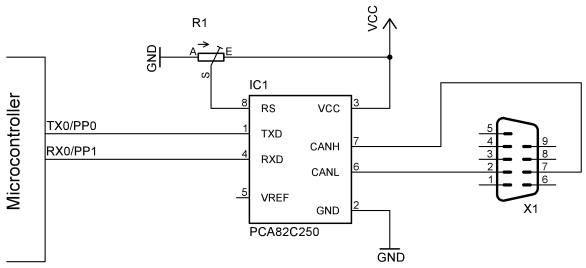

Figure 5. CAN transceiver

The DSUB9 connector has the following pinout (standardized for CAN based applications):

1. Not Connected
2. CAN_L (bus line)
3. CAN_GND (common ground)
4. Not Connected
5. CAN_SHLD (cable shielding)
6. CAN_GND (common ground)
7. CAN_H (bus line)
8. Not Connected
9. CAN_V+ (external power supply – optional)

The microcontroller can be turned into pure monitoring mode, in which case the transmitter is disconnected from the CAN bus, not interfering with the messages passed. If this mode is not used every message will be acknowledged which can cause unexpected behavior in case of some systems are relying on these acknowledgments.

However for a simple environment monitoring this option is not needed, especially if the sensors are only sending data on request. For this purpose 2 channels of the CAN interface in the microcontroller is used. One of them listens to all messages on the bus without interfering, allowing the integration of existing CAN based solutions. The other channel is used for periodically checking the sensors. A limited range of IDs are used for the sensors, and the microcontroller generates data request messages (feature of the CAN protocol) from these IDs.

4. The Prototype

The block diagram of the prototype is presented on figure 6.

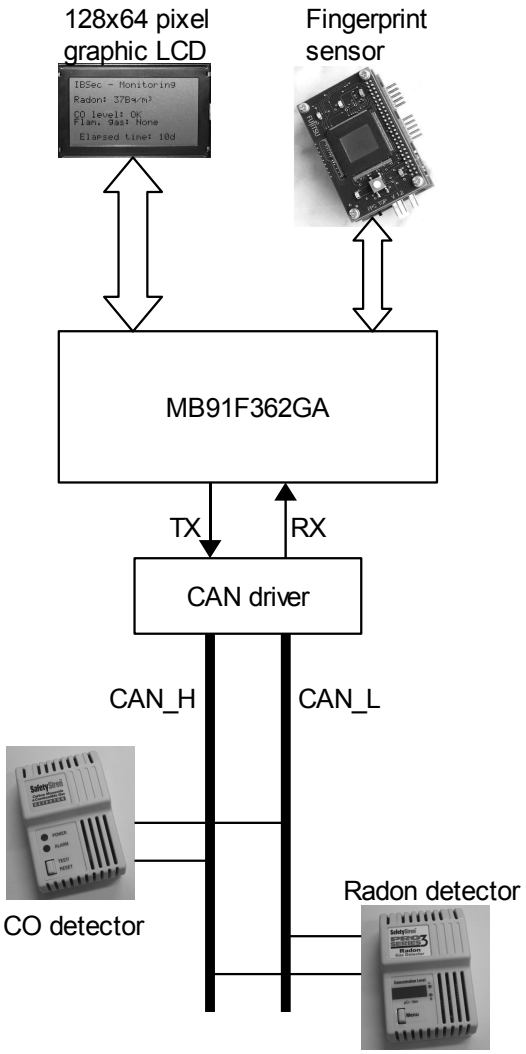

Figure 6. Block diagram of the proposed system

The prototype was built around a MB91360 Starterkit development board containing the MB91F362GA microcontroller, the SDRAM memory and the CAN driver. The external memory interface is present on a pin header which is connected with a ribbon cable to the fingerprint sensor board. Other pins of the microcontroller is connected to pinheaders, from which the K and L ports were used to connect a 128x64 pixel graphic LCD as a user interface.

For practical experiments a CO level and a radon level monitor were simulated using a Fujitsu Dice-Kit development board. This board employs a MB90F352 microcontroller which contains CAN interface. The transmitted values were set by a potentiometer and were sent every minute.

5. Conclusions

The presented system offers a higher integration of the security and safety related systems inside intelligent buildings. The systems allows the monitoring of both persons entering the building and the danger factors in the environment and thus a better situation handling in case of danger.

By using a microcontroller, which is already present in the system and most of the time is running idle (the on used for fingerprint recognition) as the central part of the system instead of a computer both the costs and the power consumption of the system can be greatly reduced.

Acknowledgments

This work has been conducted at Transilvania University of Brasov within the frame of a CEEX contract: Remote Monitoring and Control of Intelligent Buildings coordinated by prof. dr. ing Aurel Vlaicu, Technical University of Cluj-Napoca, and local coordinator prof. dr. ing. Mihai Romanca.

References

[1] Gossmann, O., Meixner, H., *Sensors Applications vol.2 – Sensors in Intelligent Buildings*, Wiley-VCH, 2001

[2] Luo, R.C., Lin, S.Y, Su, K.L., *A multiagent multisensor based security system for intelligent building*, in Proceedings of Multisensor Fusion and Integration for Intelligent Systems, 2003, pp. 311-316

[3] Chien, T.L., Su, K.L., Guo, J.H., *Design a GSM Based Distributed Security System for Intelligent Building*, in Proceedings of 1st International Conference on Positioning Technology, Hamamatsu, Japan, June 2004

[4] Stoimenov, L., Rancic, D., *Knowledge-Based Components of the Fireguard – an Intelligent GIS for Fire Department Services*, in Proceedings of IASTED International Conference on Parallel and Distributed Computing and Systems, Chicago, 1996, pp. 540-543

[5] Kertész, Cs.Z., Ogruţan, P., *Intelligen Building Security with Alternate Communication Path*, Proceedings of 13th International Symposium for Design and Technology of Electronic Packaging, Baia Mare, 2007, pp. 142-145

[6] Ogruţan, P., Romanca, M., Kertész, Cs.Z., *A Multisensor GPRS-based Security System for Intelligent Building*, Acta Technica Napocensis, vol. 48, nr. 3, Cluj Napoca, 2007, pp. 45-48

[7] ***, *FR50 32-bit Microcontroller MB91360 Series Hardware Manual*, Fujitsu Limited, April 2003

[8] * * *, *MBF200 Fingerprint Sensor Embedded Development Kit MDFP200 – User Guide*, Fujitsu ltd., March, 2004

[9] Maltoni, D., Maio, D., Jain, A., Prabhakar, S., "Handbook of Fingerprint Recognition", 2002

[10] Hong, L., Jain, A.K., Pankanti, S., Bolle, R., "Fingerprint Enhancement", IEEE Workshop on Applications of Computer Vision, Sanratosa, FL, 1996

[11] Kertész, Cs.Z., *Speed-optimized Fingerprint Image Enhancement for Embedded Systems*, Proceedings of 11th International Conference on Optimization of Electrical and Electronic Equipment, vol IV., Braşov, 2008, pp. 75-79

[12] ***, *CAN Specification Version 2.0*, Robert Bosch GmbH, 1991

[13] Kertész, Cs.Z., Gerigan C., *CAN Bus Analyzer with General Purpose Microcontroller*, Proceedings of 10th International Conference on Optimisation of Electrical and Electronic Equipment, vol IV., Braşov, 2006, pp. 61-64