



Designing and Simulation of Real Time PIN-Based Security System Using ARM LPC2138

Vineet Kumar Nandeshwar¹ Mr. Deepak Sharma²

M.E. Scholar¹ Associate Professor²

Electronics & Telecommunication Engineering, Chhatrapati Shivaji Institute of Technology, Durg,
Chhattisgarh, India

Corresponding author, E-mail address ¹vinu002@gmail.com, +919300991728,
²deepaksharma@csitdurg.in, +919907918508

Received Nov'28, 2019; received in revised form Jan'23, 2019; Accepted Jan' 23 2020

Abstract

Designing and Simulation of real time embedded system with LCD as display unit which is interfaced with ARM LPC2138 using a simulation software package; The LPC2138 is an ARM architecture-based microcontroller (an NXP product) to which an LCD is interfaced. The one - way or Simplex communication is established between the PC's Keyboard and LCD. The source code is written by using Embedded C language on Keil-μ vision4 development environment. The result is inferred from the simulation output.

Key Words: ARM7, LPC2138, UART, LCD, MCU, CPU, PIC, Point of Sale, VBAT & VREF.

Broad Area- Communication Engineering

Sub-Area- Embedded Systems .

1. Introduction

The exponential advancement in electronics industry led to revolutionize the field of embedded systems. ARM stands for Advanced RISC Machine. ARM processors are one such advancement in embedded stream that serve as a vital base for various modern electronic gazettes [6, 4]. One of such kind is LPC2138 microcontroller series based on 32-bit build-in ARM7TDMI-S architecture [1], [2], with emulation support, which combines it with high speed embedded flash memories ranging from 32k bytes to 512k bytes. Due to their tiny size and low power consumption these microcontrollers are ideally suited where miniaturization is a prime requirement, such as auto control and POS terminal. A vital advantageous point of present article is that it is based on “Proteus” Simulation

software, that makes ARM circuitry possible without even purchasing an actual Hardware Board for ARM.

1.1 ARM's core families

Introduction of RISC microprocessor for commercial purpose that is market leader in low power and cost-sensitive embedded applications is started. It features a simple architecture that inhibits little effectuations result in very low power consumption. The core base of ARM7TDMI is the fourth version of the ARM architecture. This effectuation uses a three-stage pipeline that includes a fetch - decode – execute arrangement. The core is majorly small, but has a high-performance processor and do have more than 70,000 transistors with low power consumption. This makes it most successful and

popular core available in the market. ARM cores use a 32-bit, Load-Store RISC architecture.

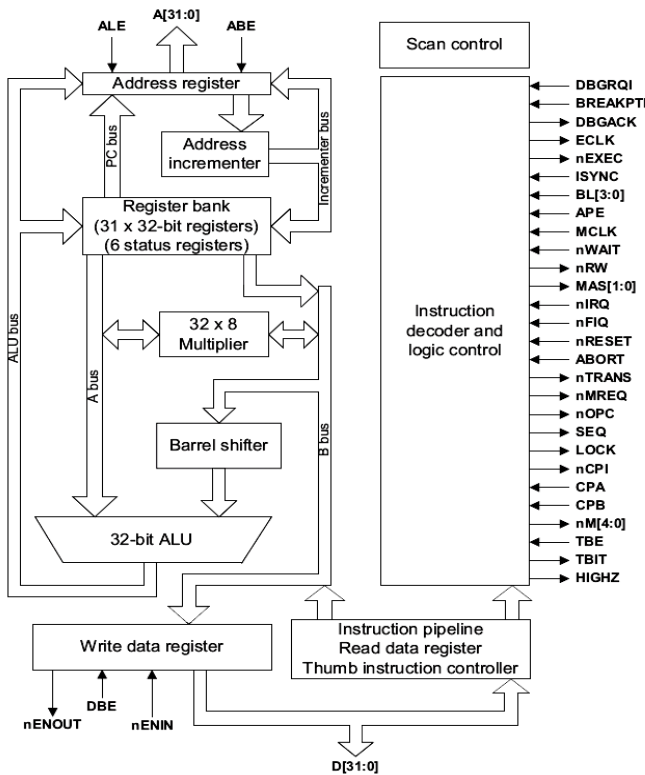


Figure 1. ARM7 Architecture (Standard ARM7 Architecture)

The core present here cannot manipulate directly the memory of the system. The data manipulation is done by loading registers with information that are located into the memory, which performs the data operation and stores the value back to memory. The processor contains total 37 registers. Meanwhile, this number is split into 7 different processor modes. These 7 processor modes are used to efficiently handle exceptions, such as interrupts, used to run user tasks and operating system. The processor core reserves some of the registers for specific usage, while most are available for general use. The processor core generally uses the reserved specific register R13 as a stack pointer (SP), R14 for link register (LR), R15 for program counter (PC), the current program status register (CPSR), and the saved program status register (SPSR).

The SPSR and the CPSR contain the status and control bits specific to the properties in which the processor is operating. These properties delimitate the operating mode, like an ALU status flag, interrupt flags & operating state, i.e. in 32-bit ARM or in 16-bit Thumb.

1.2 Migrating from 8051 to ARM

So as long as the 8051 is powerful enough for system needs then we shouldn't move. But if the user feels like he needs a change he should move to ARM based architectures for a number of reasons.

- A. They are dropping prices rapidly.
- B. They are FULLY 32-bit cores; 8-bit MCU's just don't cut it sometimes. For example, a DVD player from 2002 has an 8052 MCU embedded in it as a front end (remote control reading, On Screen Display etc.). But newer DVD players from what have been seen are either using proprietary cores or ARM based cores.

Budget wireless chipset and routers use ARM7 and ARM9 chips (probably to do the WEP/WPA encryption). Well the Wi-Fi chipset in the PSP uses an ARM9 based CPU.

Supported by freeware GNU compilers (for example gcc), makes development painless & if user wants to code in assembly, can bring free assembler.

Tons of very cheap Devkits available (that operated close to 60Mhz, a 60Mhz ARM are a very powerful chip, definitely could do mp3 decoding) and of course, the iPod nano (and probably all iPods) uses arm as a main CPU.

ARM is rapidly gaining ground and with their price going downwards every year, soon they will

replace most 8-bit controllers in anything but mundane tasks.

Generally, when it comes to MCU's: PIC - Only use for mundane tasks (keyboard handling, scanning of image, etc.) as the single cycle architecture makes this stuff much easier.

8051 is used for more complex tasks; still it is limited because of the limited stack space. Handles complicated tasks much more elegantly than peak.

ARM7/9 - Use for complicated tasks that require a lot of code (Video players, MP3 players, etc.). Handles complex tasks very elegantly and efficiently.

ARM executes almost all the instruction in only one cycle, whereas 8051 micro controller takes more than one cycle in almost all the instruction except register transfer. Example conditional jump takes three cycles for execution ex: DJNZ in 8051 conditional jump takes 1 cycle for execution and in ARM the example for this type of execution is BNEQ instruction.

ARM executes almost all the instruction in only one cycle, whereas 8051 micro controller takes more than one cycle in almost all the instruction except register transfer. Example conditional jump takes three cycles for execution ex: DJNZ in 8051 conditional jump takes 1 cycle for execution and in ARM the example for this type of execution is BNEQ instruction.

An ARM is a RISC based architecture, while 8051 is a CISC based architecture but having a smaller number of instructions as compared to ARM which is RISC.

ARM is based on load, store architecture, i.e. data processing instruction cannot access memory

directly; data has to be stored in a register before processing. Whereas in 8051 can access memory directly.

ARM have conditional data processing instruction were as 8051 does not.

ARM is compact in size. For ARM ADC is in built in LPC2138 whereas for microcontroller it is placed externally.

3.System Design

The Design of the system consists of an ARM LPC2138 controller consist of ARM7TDMI-S architecture as the main or the heart of the system. An LCD LM016L is interfaced; it is a 16x2 alphanumeric display LCD. Finally, a virtual terminal is selected in the result. The block diagram showed next, followed by a flow chart elaborate the design.

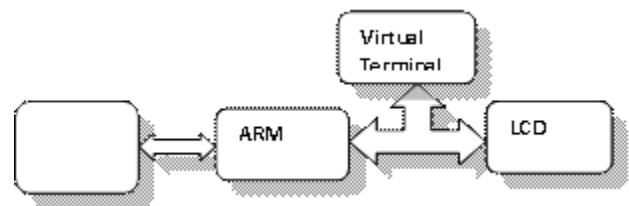


Figure 2. System Design

3.1 Block Diagram Explanation

The system design is prepared in 'Proteus' simulation software version 8. The ARM LPC2138 is interfaced with the LCD, and with the virtual terminal. A string of characters, for example "1234" is sent to the ARM, the controller from its UART pin receives the data; compares it with the database; if the data matches it sends an acknowledgement to the LCD and simultaneously to Virtual terminal that "password match" otherwise acknowledges "wrong password".

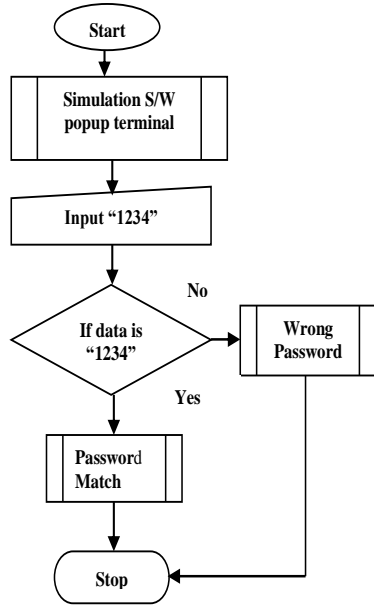


Figure 3. Software Flow Chart

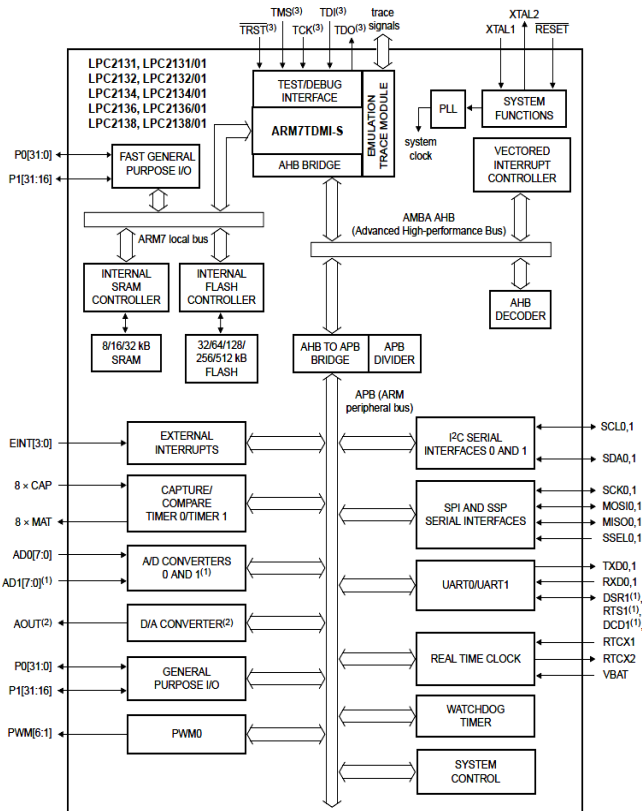


Figure 4. System Hardware Design (Proteus V.8)

3.1.1 Simulated Hardware Design

The simulated circuitry of LPC2138 (an ARMv7 based microcontroller by NXP) and the LCD LM016L is shown next; in figure 4. It has been designed using ‘Proteus’ version8 simulation software. The simulation package contains various types of IC’s with processors and controllers along with various inbuilt peripheral and driver IC’s. For the present research, one of its Controller i.e. LPC2138 is selected from the device database of software package. Its internal architecture is based on ARM7TDMI architecture. The VBAT and VREF are provided with +3.3v supply & VSS are kept grounded of LPC2138. The general purpose in-out pins are interfaced with the LCD module [6]. Finally, the virtual terminal in the ‘‘Proteus’’ software is attached with the RX and TX pin of the controller.

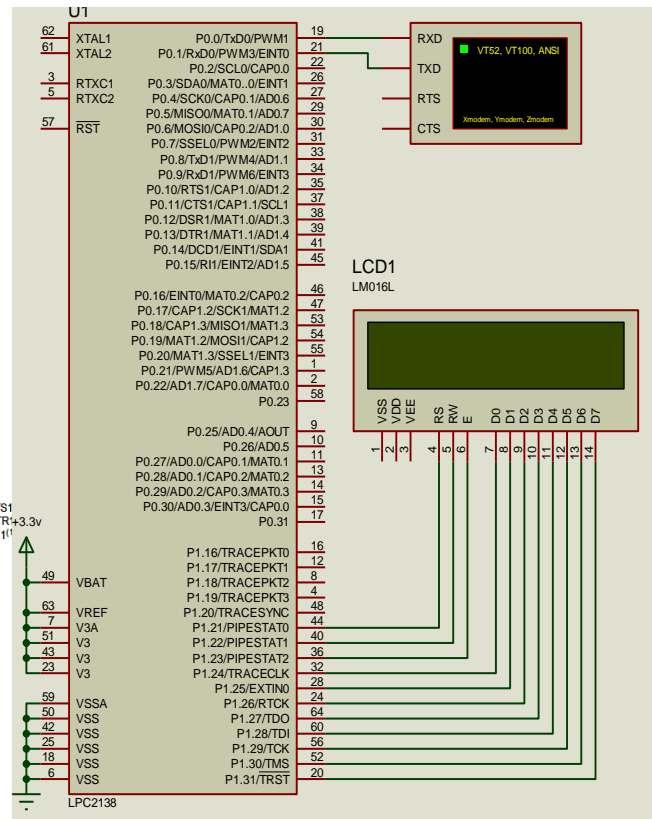


Figure 5. Block Diagram of LPC2138 (Courtesy: NXP)

4. Firmware development

The firmware for design, implementation of the embedded system is organized by using Embedded C language. The IDE used for software development is Keil-μ vision4 development environment. The oscillator frequency for ARM in the source code is defined as 12 MHz, the general purpose in – out pins (GPIOs) [2] are initialized and configured the pins of port 0 & port1 are directed as output pins. UART 0 [20] is initialized for transmission and reception of data from PC to ARM & then to the LCD display unit.

5. Result

The output of the system designed is visible clearly in fig.6. It can be inferred that the Simplex communication is established between ARM and LCD and to the virtual terminal. The keyboard typed password is compared with a database (i.e. “1234”) in ARM, The inbuilt 512kB of flash memory stores the data; for the process it is taken as “1234”, if this matches an acknowledgement is sent simultaneously to both; virtual terminal & to LCD, that “password match”. Similarly, if there is a wrong entry through keyboard, then it acknowledges “wrong password” at the virtual terminal [16] and to the LCD at the same moment. This can be justified by the figure 6, 7 & 8 taken at different moment as the clip mentioned here is screened during the running process of “Proteus” application software. By a close verification during running process, the words shown and cursor moment on the LCD and the virtual terminal remains approximately same for a particular moment.

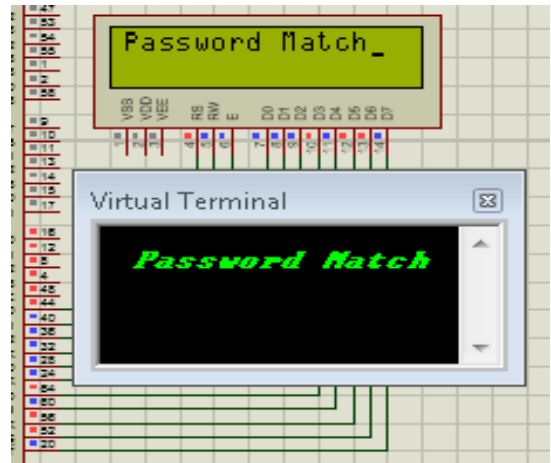


Figure 6. Simulated result output (for “password match”)



Figure 7. Simulated result output (for “wrong password”)

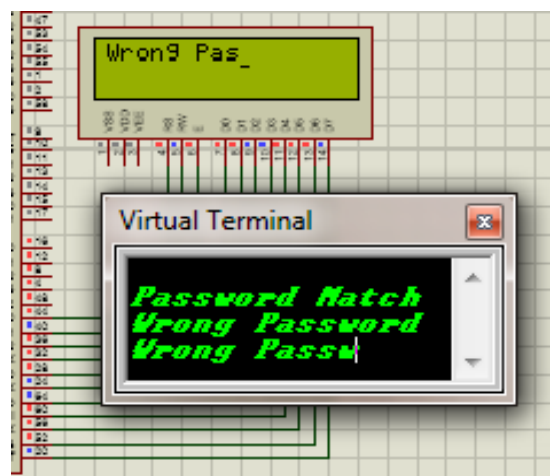


Figure 8. Simulated result output (At a particular instant)

Conclusion

It can be concluded from above heading under the result section that a Simplex communication is established between the ARM Controller and the LCD display unit; that can be inferred from the snapshot, both displays get a particular acknowledgement simultaneously. This makes the system more secure as the system verifies the PIN details. The results found its significance in a wider range of the gateway, point of sale, automation and industrial application protocol systems.

Conflict of interest

The author declares no conflict of interest.

References

- [1]. Indersain, N. S., & Singh, D. (2013). Design and Implementation of μ c/Os II Based Embedded System Using ARM Controller. *International Journal of Engineering and Technical Research (IJETR)*, Volume-1, Issue-2, April 2013.PP.1-4
- [2]. Wen, Z., & Meng, J. (2011, May). Design of vehicle positioning system based on ARM. In *2011 International Conference on Business Management and Electronic Information* (Vol. 4, pp. 395-397). IEEE.
- [3]. Bazzaz, M., Salehi, M., & Ejlali, A. (2013). An accurate instruction-level energy estimation model and tool for embedded systems. *IEEE transactions on instrumentation and measurement*, 62(7), 1927-1934.
- [4]. Zhou, L., Zhu, Y., Chen, A., & Liu, C. (2013). A simulation platform for ZigBee-UMTS hybrid networks. *IEEE communications letters*, 17(2), 293-296.
- [5]. Wang, R., & Yang, S. (2004). The design of a rapid prototype platform for ARM based embedded system. *IEEE Transactions on Consumer Electronics*, 50(2), 746-751.
- [6]. López-García, M., Ramos-Lara, R., Miguel-Hurtado, O., & Cantó-Navarro, E. (2013). Embedded system for biometric online signature verification. *IEEE Transactions on industrial informatics*, 10(1), 491-501.
- [7]. Chang, N., Kim, K., & Lee, H. G. (2002). Cycle-accurate energy measurement and characterization with a case study of the ARM7TDMI [microprocessors]. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 10(2), 146-154.
- [8]. Martínez-Espronedá, M., Martínez, I., Serrano, L. Led, S., Trigo, J.D., Marzo, A., Escayola, J. and García, J. 2011. Implementation Methodology for Interoperable Personal Health Devices With Low-Voltage Low-Power Constraints, *IEEE transactions on information technology in biomedicine*, 15(3), May, 398-408.
- [9]. Wehmeyer, L., Jain, M. K., Steinke, S., Marwedel, P., & Balakrishnan, M. (2001). Analysis of the influence of register file size on energy consumption, code size, and execution time. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 20(11), 1329-1337.
- [10]. Jacquet, D., Hasbani, F., Flatresse, P., Wilson, R., Arnaud, F., Cesana, G., ... & Grover, C. (2014). A 3 GHz dual core processor ARM cortex TM-A9 in 28 nm UTBB FD-SOI CMOS with ultra-wide voltage range and energy efficiency optimization. *IEEE Journal of Solid-State Circuits*, 49(4), 812-826.

- [11]. Manivannan, M., & Kumaresan, N. (2011, March). Design of on-line interactive data acquisition and control system for embedded real time applications. In *2011 International Conference on Emerging Trends in Electrical and Computer Technology* (pp. 551-556). IEEE.
- [12]. Faundez-Zanuy, M., Espinosa-Duro, V., & Ortega, J. A. (2005). A low-cost webcam & personal computer opens doors. *IEEE Aerospace and Electronic Systems Magazine*, 20(11), 23-26.
- [13]. Chen, S. L., Lee, H. Y., Chen, C. A., Huang, H. Y., & Luo, C. H. (2009). Wireless body sensor network with adaptive low-power design for biometrics and healthcare applications. *IEEE Systems Journal*, 3(4), 398-409.
- [14]. Lee, H., Lee, S. H., Kim, T., & Bahn, H. (2008). Secure user identification for consumer electronics devices. *IEEE transactions on Consumer Electronics*, 54(4), 1798-1802.
- [15]. Kukula, E. P., Sutton, M. J., & Elliott, S. J. (2010). The human-biometric-sensor interaction evaluation method: Biometric performance and usability measurements. *IEEE Transactions on Instrumentation and Measurement*, 59(4), 784-791.
- [16]. Wei, H., Chen, Y., Tan, J., & Wang, T. (2010). Sambot: A self-assembly modular robot system. *IEEE/ASME Transactions on Mechatronics*, 16(4), 745-757.
- [17]. Benini, L., Macchiarulo, L., Macii, A., & Poncino, M. (2002). Layout-driven memory synthesis for embedded systems-on-chip. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 10(2), 96-105.
- [18]. Goodacre, J., & Sloss, A. N. (2005). Parallelism and the ARM instruction set architecture. *Computer*, 38(7), 42-50.
- [19]. Bonny, T., and Henkel, J., (2008). Efficient Code Compression for Embedded Processors, *IEEE transactions on very large scale integration (vlsi) systems*, 16(12), December, 1696-1706.
- [20]. Yang, C. L., Tseng, H. W., Ho, C. C., & Wu, J. L. (2005). Software-controlled cache architecture for energy efficiency. *IEEE Transactions on Circuits and Systems for Video Technology*, 15(5), 634-644.