DEVELOPMENT OF ARM-BASED APPLICATION SYSTEM

ENGR. ZARINA BINTI MOHD NOH
SANI IRWAN BIN SALIM
ENGR. NORHIDAYAH BINTI MOHAMAD YATIM
ENGR. NUR ALISA BINTI ALI
ENGR. MUZALIFAH BINTI MOHD SAID

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

3 International Conference on Engineering and ICT (ICEI2012)
Melaka, Malaysia
4 – 6 April 2012

DEVELOPMENT OF AN ARM-BASED APPLICATION SYSTEM

Mohd. N. Zarina[a*], Salim S. Irwan[b], Mohamad Y. Norhidayah[c], Ali N. Alisa[d] and Mohd. S. Muzalifah[e]

Fakulti Kej. Elektronik & Kej. Komputer (FKEKK), Universiti Teknikal Malaysia Melaka (UTeM), Melaka, Malaysia.
[a]Email: zarina.noh@utem.edu.my, [b]Email: sani@utem.edu.my, [c]Email: norhidayahm@utem.edu.my,
[d]Email: alisa@utem.edu.my, [e]Email: muzalifah@utem.edu.my

***Abstract*—The aim of this paper is to expose the development process and software involved in realizing an ARM-based application system. The application system consists of a cruise algorithm inten ded to be used in an autonomous robot prototype, which is developed with the help of *Flowcode* software that utilizes flowcharts as its design entry. The flowchart is then configured to be tested for real-world application over *E-blocks* board integrated with an ARM-based microcontroller chip from Atmel, AT91SAM7S128. It is hoped that the development process shared in this paper may be benefitted for researchers who wishes to start developing an ARM-based system for further study or other purpose in one way or another.**

Keywords — ARM-based processor; AT91SAM7S128; E-blocks; Flowcode.

I. INTRODUCTION

ARM7 was introduced somewhere in 1993, but it lacked some features, which was then covered by its successor, ARM7TDMI [1]. Several properties of ARM7TDMI has made it as one of ARM's best-selling processors – small die area, very low power and rich instruction set [1, 2]. It has von-Neumann architecture and uses RISC principle. Each registers can store 32-bits data (data lines are also 32-bits wide with 32-bit address lines) and the processor can have up to 4Gbytes memory locations connected to it [2]. ARM system uses memory-mapped I/O (Input/Output) properties in which the programmers need to know the exact location of its RAM (or ROM) and I/O locations before writing the program. The registers are not all accessible at the same time. Depending on the mode (ARM state), only 16 general-purpose and one or two status registers are available at one time. There are a total of 7 modes of operation for ARM7TDMI processor; user, system, fast interrupt, interrupt, supervisor, abort and undefined [2]. Each of these modes defines the operation executed by the ARM program.

The properties of ARM7TDMI are introduced in this paper because the application system developed in this research is utilizing the ARM7TDMI processor core. The low-end application system is developed because the main focus of this research is only to explore the development process involved in realizing an ARM-based application system, but not concentrating on the application itself.

II. ARM PROCESSOR DEVELOPMENT TOOLS

Any development tools for ARM processors can be used to write the code for an ARM application system provided that the development package includes simple instruction set-up instructions with an assembler, programmer and supported simulator.

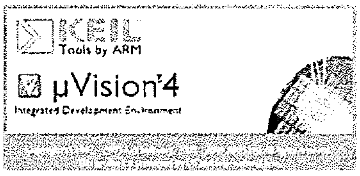

Figure 1. Startup interface of *Keil uVision4* software

One of the evaluation versions of ARM development tools is *Keil uVision4* which can be downloaded from the website of Keil Elektonik GmbH [3]. This evaluation version includes simulator and simple methods of loading programs into the target hardware for real-time execution monitoring. The startup interface of the software is as shown in Fig. 1. The program code may be written in C-language and need to be assembled (to its binary file) in order to be loaded into the target device. Users have to develop their application circuits in order to validate their system by using this development tools. The target device varies according to the users' requirement.

Other ARM processor development tool that might be useful for ARM-based system prototype is *mbed NXP LPC1768* microcontroller system which can be accessed via its website [4]. This microcontroller chip utilizes ARM Cortex-M3 core in its system. Fig. 2 shows the *mbed NXP LPC1768* chip. It can be powered up via the USB port on the computer or by using an external DC source power.

The use of *mbed NXP LPC1768* allows users to prototype their system easily as the tools enable users to perform 'drag-and-drop' programming via the USB lead (connected to the *mbed NXP LPC1768*) chip. As long as the circuits have been developed in accordance to the NXP LPC1768 chip, the assembled program can be tested just by saving the program into the chip's space. The resources are stored online and users need to have access to its website to write the coding and assemble the program. An advantage to this development tools is that, users do not need to install its software (in the computer) to write the coding as the resources are all online. It provides mobility to the users besides eliminate the need for extra space for software installation.

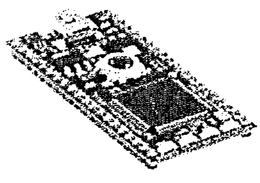

Figure 2. NXP LPC1768 microcontroller chip [4]

IV. PROTOTYPE DEVELOPMENT

The hardware of the prototype is realized via the use of E-blocks board, which was distributed by Matrix Multimedia Ltd [6]. There are two kinds of E-blocks board; upstream or downstream board [7]. The upstream board controls the flow of process in the system while the downstream board executes the process given by the upstream board. The upstream and downstream board is connected via the D-type connector. An example of the upstream board is the device programmer, while the downstream boards are usually the I/O circuitry modules such as LEDs, switches, LCD, 7-segment display, keypad or RS-232 board [7]. The use of E-blocks board in this research saves time, where the users does not have to build the application circuit from scratch, but rather used the available board that suits their application system.

In this research, the E-blocks board used in the prototype development consists of the ARM programmer board (EB031), ARM daughter board (EB034), LEDs board (EB004) and Motor Driver board (EB022) [8-10].

Fig. 8 shows the developed prototype which comprises of 4-tiers E-blocks board. The EB004 block reflects the Collect Module block in Fig. 4, where the light movement of 8 LEDs imitates the act of collecting thing by the board. The EB022 reflect the Cruise Module, in which it is connected to a pair of Pololu wheels (42x19mm dimensions). The board allows users to drive two motors simultaneously and permits independent PWM control for each motor. An external DC power source of minimum 5V is needed to power up the board and another power source is needed to drive the motors (3.3-24V range) [10].

The ARM-based processor is integrated on the AT91SAM7S128 microcontroller chip which is housed on the EB034 board. The AT91SAM7S128 microcontroller chip utilizes ARM7TDMI core in its system. This board is detachable and users may connect this chip to other prototype circuit board [8]. The EB034 board is used with the EB031 programmer board, in order to load the prototype's program into the microcontroller chip. The EB031 board comes with 5 serial ports (D-connectors) which allows users to connect the downstream boards [8]. In programming phase, the board may be powered up by the USB port connected to it. In testing phase, an external regulated DC power source of 6-9V is needed to power up the EB031 board.

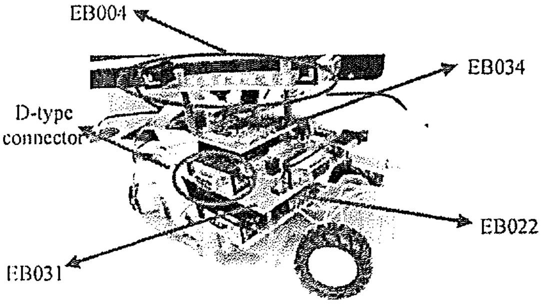

Figure 8. Prototype developed

V. DISCUSSION

The application system in this research had been successfully developed with the use of E-blocks board and *Flowcode* software. The combination of these two tools is useful when the main concern in the system's development is the realization of a working and functioning prototype. Even though the use of *Flowcode* software permits users to enter the flowchart of the system operation in the software, certain properties of the *Flowcode* software need to be familiarized by the user before the software can be fully utilized in the design. Same goes for the E-blocks board where users need to browse through the datasheet before applying the board in the prototype development. Prior knowledge to electronic circuit is needed to speed up the understanding process, before the E-blocks board can be connected as a working prototype.

VI. CONCLUSION

In conclusion, the development of an ARM-based application system can be done with the use of E-blocks board and *Flowcode* software. The combination of these tools has an added advantage in the design of a system prototype.

The use of flowchart in the design entry level saves times and eliminates the need to debug the programming code. This reduces error and saves time. In a project where the realization of a working prototype is important, the use of flowchart is the best option as it focuses more on the design requirement.

The E-blocks board applied in this ARM-based application system help ease development process as the board is recognized by the *Flowcode* software (flowchart developed can be loaded directly to the target board) and peripherals used in the design are easily connected to one another (via the D-type connector on the serial ports). The only constraint is whether the available E-blocks board matches the design requirement of the intended system. When the developed system is quite new to the developers (such as an ARM-based processor environment), the use of E-blocks board saves time, in a way that the related circuitry had already been defined, just like a predefined function already declared in system and ready to be used in a program.

Hence, researchers may opt for the combination of these two tools in the development of an ARM-based application system, especially for beginners who wishes to explore and get familiar with an ARM-based platform environment.

ACKNOWLEDGMENT

The authors would like to address the greatest appreciation to everyone involved in the success of this research especially those in Fakulti Kej. Elektronik & Kej. Komputer (FKEKK), Universiti Teknikal Malaysia Melaka (UTeM). The continuous support and opportunities given in completing this research is an invaluable experience to us.

REFERENCES

[1] W. Hohl. *ARM Assembly Language – Fundamentals and Techniques*, US: CRC Press, 2009, pp. 2–10.

[2] ARM Limited, *ARM7TDMI Technical Reference Manual: ARM DDI 0210C*, 2004.

[3] ARM Development Tools, ARM Ltd and ARM Germany GmbH (Accessed 2011). Citing Internet sources URL http://www keil.com/arm/

[4] NXP LPC1768 mbed Microcontroller (Accessed 2011). Citing Internet sources URL http://mbed.org/nxp/lpc1768/

[5] *Flowcode* by Matrix Multimedia Ltd. (Accessed 2011). Citing Internet sources URL http://www.matrixmultimedia.com/flowcode.php

[6] *E-blocks Board* by Matrix Multimedia Ltd. (Accessed 2011). Citing Internet sources URL http://www.matrixmultimedia.com/eblocks.php

[7] Matrix Multimedia Ltd., *E-blocks User Guide: EB355-80-V4*, 2007.

[8] Matrix Multimedia Ltd., *ARM programmer and daughter board (EB185-00-1)*, Technical Datasheet (Document Code: EB185-30-1), 2008.

[9] Matrix Multimedia Ltd., *LED board (EB004-00-2)*, Technical Datasheet (Document Code: EB004-30-2), 2005.

[10] Matrix Multimedia Ltd., *Motor driver board (EB022-00-1)*, Technical Datasheet (Document Code: EB022-30-1), 2006.