Lab 3
Realization of An Automatic Ticket Issuer
Out: 12/24/2002
Due: 1/14/2003

1. Objectives
This lab is to implement the automatic ticket issuer using an ARM Evaluator-7T board. The purpose of this lab is to make students familiar with one of the well-known hardware platforms for embedded system design.

2. Problem Descriptions
You are asked to modify the automatic ticket issuer developed in Lab 2 such that the code can be run on an ARM Evaluator-7T board. The specifications and requirements for inputs and outputs are basically the same as that described in Lab 2. However, the manner for inputs and outputs is different because the ARM Evaluator-7T board has limited input and output capabilities. The following is the manner for inputting and outputting:

Inputting:
The primary input device on the ARM Evaluator-7T board is a 4-bit DIP switch that can represent a number from 0 to 15. Therefore, pressing a button on the automatic ticket issuer means that setting up the switch and then pressing the user interrupt switch to notify the machine about the readiness of an input. Because a button is represented by a pair of (X,Y) as described in Lab 2, pressing a button means to input two numbers using the DIP switch. The following is a legal input sequence:

(1,2), (2,1), (3,3), (4,4), (5,1), (6,10), (6,10), (6,10), (6,10).

It means that the customer would like to buy a round-trip ticket from 松山 to 台北 with full-price and have deposited 40 dollars. You may notice that we can not input a 20- and 50-dollar coin in this way. So, here we use a 11 to represent a 20-dollar coin and a 12 to represent a 50-dollar coin. Other inputs are illegal. Your program still has to check an illegal input. When there is an illegal input, the input pair is ignored.
Outputting:

The primary output devices on the board are a seven-segment display and 4 LEDs. So, for outputting we use the seven-segment display to show the output choice and the LEDs to show the content. Let (SD, LD) be a pair of numbers to denote the contents of the seven-segment display and LEDs. Then, an output is represented by a sequence of (SD, LD) pairs. For example, in response to the input given by the above example, the output will be

\((1,1),(1,1),(1,1),(2,2),(2,1),(2,3),(2,4)\).

It means that a customer has got a round-trip ticket from 松山 to 台北 and got back 4 one-dollar coins. When outputting a pair of number, the LEDs and seven-segment display should first be cleared and then display the pair of numbers after waiting for a sufficient time interval.

To show the status of a customer’s inputs on the LEDs and seven-segment display, we add a new type of input button. This type of input button is designated as the 8th input selection category. When an input pair of (8,8) occurs, the system knows that a customer want to know which input selection category has not yet been selected. So, in response to such an input of (8,8), the output should be a sequence of (X,Y) pairs, where X is 4 and Y denotes an input selection category. For example, the output response may look like (4,1),(4,3),(4,5) which means that a customer has not processed the input selection categories 1, 3, and 5. Pressing this button is optional, i.e., it is not necessary for a customer to press this button during a transaction to get his/her tickets.

One more change is with the LED output in Lab 2 for showing the status of the machine. The change is as follows. When the system is ready for starting a new transaction, the four LEDs on the board is first turned on. And once the transaction is started, they should be turned off. If there is a malfunctioning, the four LEDs are flashing in every two seconds.

As regards to time-out, you have to follow the specifications and requirements presented in Lab 2.

▲ Tasks to be done:
Modify the programs PC and PD written in Lab 2 according to the changes made in this lab such that the programs can be executed on the ARM Evaluator-7T board.
3. Evaluation
You have to hand in a report that summarizes your results and points out the specific features of your software implementation. The report should consist of the following main sections:
1. Cover page
2. Introduction
3. Method
4. Results
5. Discussion

The name of your report is designated as \textit{report\_your\_student\_ID.doc}, where \textit{your\_student\_ID} represents your student ID number. Name each of the two programs as \textit{PX\_your\_student\_ID.XXX}, where \textit{PX} is either PC or PD; \textit{XXX} indicates the file extension. You have to \textit{ftp} both the two programs and the report to the directory called \textit{LAB3} in our website for this course, respectively. You also have to hand in a copy of the report by the due date.
Your score will be computed based on the quality of the report and the correctness of your programs. Your programs will be tested on a board by a teaching assistant. Time will be scheduled for this purpose.

4. Materials
You probably need to make reference to the user guide of the ARM Evaluator-7T board. This user guide can be downloaded from the ARM’s web site.

5. Notes for 3C Lab
Because there are only 25 ARM Evaluator-7T boards, this will limit the number of students doing their projects in the 3C lab. When you want to use the lab, you must come to the lab during the open hours and borrow a board from a teaching assistant who is in charge of the lab during those hours. The boards will be lending out on the basis of first-come-first-serve.
After using the board, you should return the board to the teaching assistant. \textit{You should not pass the board to another student without returning the board first and should not take the board away from the 3C lab}. If you violate this rule, you will be responsible for buying a new board and give it back to the lab if the board is damaged or lost. In addition, you will get 10% less the score you should have obtained. \textit{You should be as careful as possible in handling the board}. 