MICROPROCESSOR BASED SYSTEM TO ESTABLISH INTELLEGENT DRIVERS

Mazin M. Ali / Msc. Electronic Engineering
Al-Qadissyia University – College of Engineering

Abstract

A system of sixteen programmable drivers has been established around an 8085 microprocessor, where each driving path is independent and can be programmed to energize the driven element at the desired timing for the desired period of time . An input programming circuit of thirty programming steps is built to enter the driving sequence for selected driver that specified by the driver selector .

نظام مرتكز على المعالج الدقيق لبناء مسوقات ذكية

مازن مكي علي / ماجستير هندسة الكترونية
جامعة القادسية – كلية الهندسة

الخلاصة

نظام مرتكز على المعالج الدقيق ٨٠٨٥ ذات ستة عشر مسوقات (مسارات) قابلة للبرمجة تم التوصل لبنائه ماديا" وبرمجيا" ، حيث أن كل مسار يكون مستقلا" بأدائه وقابل للبرمجة بما يؤهله لسوق العنصر المطلوب السيطرة الزمنية على فترات اشتغاله عند أي لحظة من الزمن ولأي فترة منها . كما أن دائرة إدخال برمجي ذات ثلاثون خطوة مبرمجة قد تم بنائها للتمكن من ادخال تتابع السوق المطلوب لعنصر السوق المختار .

Introduction

As the demand on electronic equipments increases , many branches of our modern life becomes more dependents on these equipments . And as those increasingly complicate then the necessity for proper timing of operation becomes more insisted such that the operation of equipment can be controlled in certain time , for proper time of period and repetitive many time as required .

The first control problem faced human in industrial life was in 16^{th} century, where the engineers at that time were in front of the challenge how to control the speed of engine without human intervention [1]. Since that time, the control styles improved with improvement in industrial process reaching to 1950s, where the control techniques became more advance and more complicated with the advent of digital computer to establish to automatic control systems [2]. Estimations state that with conventional machine tools, only about 20% percentage of time is spent removing materials while with electronic controlling the time gets improvement to 80% [3].

Many studies had been achieved in field of programmable drivers, so many products had been manufactured to meet this aim, like the DPC50501 which is a programmable driver to control operation of servo motor [4], and the LM5105 which is a high voltage driver with programmable dead time introduced by National Semiconductor to drive the N-channel MOSFET [5]. All of those programmable drivers and others are of a single programmable output, so the are able to control a single element at a moment.

In contrast, the field programmable gate arrays (FPGA) can give controlling on performance of many elements at the same time, but it suffering from the enter connect delay and this challenge is crucial since most FPGA implementations use many long wires [6]. The modeling introduced by Edmund [6] just improved the performance of FPGA by 19% to 40%, but the problem of delay still stands.

The concept of intelligent control was firstly introduced nearly more than two decades ago by Fu and G, saridis. Despite its significance and applicability to various processes, the control community had not paid substantial attention to such an approach [7]. Nazim and Suliaman found the intelligent control system very soft and acceptable when they tried to use it in hybrid mobile robot peed control and it provided successful utilization in field of hybrid vehicle recently immerged in automotive industry [8].

Here is a new approach to introduce an intelligent system can control many drivers at any time, for any period and for any repetitive sequence. The approach is built around the 8085 microprocessor, which can replaced by any other processor if it is required to improve the speed of processing or to handle more of programmable drivers. This will introduce an intelligent system to give very flexible and reliable controlling on timing of operation of any electrical equipment or system like machines, industrial process, traffic lights … etc.

In contrast with PLC system, where the user need a certain level of education or may need a training course to be familiar with using of the PLC in addition to programming it by ladder language [9], the suggested approach is very simple to deal with.

Theory

Any programmable system, such as a microprocessor or computer, should has a general purpose processing unit in addition to other electronics circuits to input data or signals from external circuit, move or transfer data within system, store data before, during or after processing and finally output the processed data to external circuits or systems [10].

Each of these operations can be performed by an appropriate configuration of combinational and sequential logic circuits. All of that circuits will form the body of system which must be supported by sequence of instruction codes which will define the operations necessary to implement a particular processing function. So this type of system can be considered to be composed of two parts, the hardware and software which defines the application function and must be written carefully to lead the hardware to achieve this function [11].

The hardware is controlled by software which is built by using the instructions set, of processor in use, in proper sequence [10]. Each instruction requires T states of clock signal to be executed, so each instruction spends time equal to (required T states) x (period T of frequency of crystal) x 2 . The total time of program execution is the time required to execute all instructions within program [12].

Practical design

In design of microprocessor based systems, it is essential to consider both hardware and software simultaneously. Both are interrelated and each will have impact on the other [11]. So the overall design of system includes two related parts , the hardware approach and software approach .

Software approach

The software represent the soul of body , hardware , where Gongare [13] estimate that software may represent 70% of total design . Here four different programs are required to support the hardware design , they are :

i.–Input program , it is necessary to manage receiving of programming data by the processor unit and store them in RAM , these data are of serial form and represent the manner by which the output drivers will be energized when run program has been run .During execution of this program , all ports of I/O unit is in input mode . This program is activated by high logic on RST5.5 interrupt input in processor .**Figure (1)** shows the flow chart for this program .

ii.– Check program , it is used to check the programming status of any diver , stored in RAM , individually . During execution of this program , all ports are in input mode but P_{CL} is output and P_{C0} is connected to serial in of input programming unit while P_{C1} is connected to the clock input of the unit to provide the clock signal generated through programming . This program is activated by high logic on P_{C3} . Flow chart given in **Figure (2)** illustrates this program .

iii.– Run program , this program transfer the programming conditions of various drivers , stored in RAM , to output ports of I/O unit to be used to energize the final drivers . During execution of this program , ports A and B are in output mode while P_{CH} is input to specify which driving program , of sixteen ones , is chosen via the thumb switch for execution . This program is illustrated by the flow chart given in figure . This program is activated by high logic on RST6.5 interrupt input in processor .**Figure (3)** shows the flow chart for this program .

iv.– Monitor program , this program is running in idle condition of system waiting for the next job to be done . In this program all ports are in input mode and the interrupt inputs are unmasked .This program also include the cancel facility where a specific column within a

specific program can be canceled , or specific program is canceled in complete , or all programs are canceled , these cancellation programs are activated by high logic on P_{C0} , P_{C1} , and P_{C2} respectively .**Figure (4)** gives flow chart to this program .

2 Hardware approach

This approach consists of many circuits built around an 8085 microprocessor, it includes:

a) The 8085Ic as the central processing unit .

b) The 2764Ic as EPROM to store various programs necessary to operate the system .

c) The 6116Ic as RAM to store the programs which represent the required styles of operating the output driving elements .

d) The 8255Ic as the I/O unit to provide the necessary communication between processor and other units within the system .

e) Input programming unit , also called the serial input data unit , which consists of six integrated circuits type 7496 to act as parallel IN serial OUT shift register , to provide thirty bits programmable shift register .

f) Driver selection unit , which consists of three other 7496ICs connected to each other to act as ring shift register to indicate the driver that to be programmed , deleted or checked .

g) Final output units , which responds to operating signals provided by the processor to energize the output drivers , which are represented here by a set of relays .

h) Rotary switch , this is a thumb switch of sixteen positions acts as decimal – to – 4bits binary converter .It is used to get sixteen different addresses in order to partition the RAM into sixteen equal parts each is used to store different driving programm .

Connection of those parts to each other, forming the overall hardware, which is shown below in **Figure (5).**

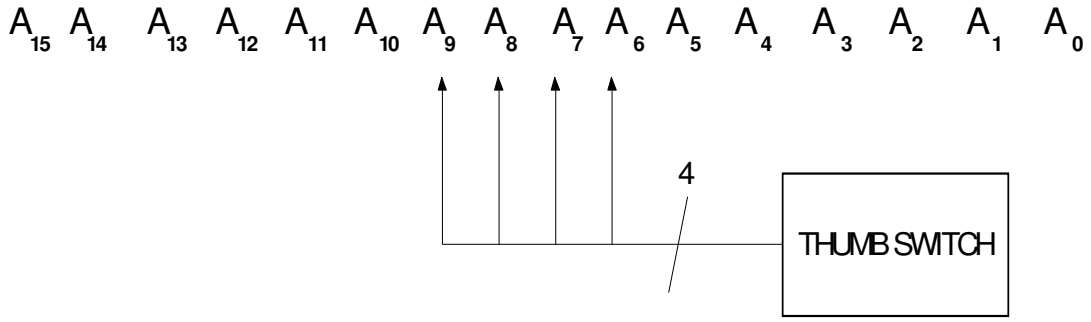
Memory map :

In microprocessor based system, built around the 8085μP , more than 65000 memory locations can be addressed [13] . To meet requirements of this approach , the following memory map is suggested .

- The addresses (0000 – to – 03FF) are specified to addressing of EPROM where the programs of software design are stored .

- The addresses (0400 – to – 07FF) are specified to addressing of RAM where this memory space is farther divided into sixteen partition , each is used to store one of the sixteen input programs . Here , the partition is performed by inserting the 4bits digital form obtained from the thumb switch into the addressing format to occupy the positions of A_6 – to – A_9 (address modification) as illustrated below .

The address modification is achieved by software where the monitor program read the status of thumb switch on port P_{CH} and load it in the corresponding bits positions in program counter forming the address of program to deal with, loading, running, reading or deleting .



- The other addresses are not used and can be used for more improvement in future .

Results:

1) The performance of system design was firstly checked using the software (protues6), where it is found work properly.
2) According to the given hardware and software design, a practical model was achieved, while by the monitoring to the performance of this practical model, under many tests, it is found that the system is of very high reliability and accurate in energizing period of excitation.
3) The minimum time of execution of run program is estimated theoretically 200 microsecond while there is no upper limit for maximum time.
4) For the used crystal of 3.175MHz and the delay time specified in delay loop in the flow chart of run program, the total time of execution of each driven program is 15 second, divided into thirty steps each of 0.5 second.
5) The driven elements were properly energized individually and then many driven elements were energized at the same time, it is found that the process was done properly without any confusion or illegal operations.
6) After 15 second of operation, the system back to idle status entering the monitor program waiting for the next command.

Conclusions:

1) The system is accurate in timing and of high reliability.

2) The design of system is flexible, where the speed of execution of driven programs and so the time of each step can be changed through hardware by replacing the clock crystal with other of different frequency, or through software by varying the period of delay loop specified by flow chart of run program.
3) If there was an expected bad effect on the legal operation of system resultant from inductive loads that may be connected to the programmable drivers, then problem can be avoided by using photo coupler as interface between system and driven elements rather than the relays, where these couplers will provide good optical isolation from inductive loads.
4) It is possible to end execution of driven program by specifying one of drivers for this purpose, so that the processor can recognize energizing of that driver and route the execution toward monitor program.
5) For repetitive operation to certain program, another driver can be specified so that energizing of that driver will inform the processor to repeat execution of the worked program.
6) The system reflects very well the benefits of open loop control system, in spite of , it can be modified to serve the applications of closed loop control systems.

References:

1. Roland S. Burns , Advance Control Engineering , Butterworth-Heinemann , Oxford , 2001.

2. Frances H. Raven , Automatic control Engineering , 3rd edition , McGraw-Hill , 1978 .

3. Steve Krar & Arthur Gill , Computer Numerical Control Programming Basics , Industrial press , New York , 1999 .

4. DPC50501 programmable driver , internet , www.anaheimautomation.com .

5. LM5105 100v Half Bridge Gate Driver with programmable Dead-Time , internet , www.national.com .

6. Edmund Lee , Interconnect driver design for long wires in field-programmable gate array , Msc thesis , University of Toronto , 2003 .

7. Ali Zilouchian & Mo Jamshidi , Intelligent Control Systems using soft computing methodologies , CRC press , New York , 2oo1 .

8. Nazim Mir-Nasiri & Sulaiman Hussaini , New intelligent transmission concept for hybrid mobile robot speed control , International Journal of advanced robotics systems , Vol.2 No.3 , 2005 .

9. L. A. Bryan & E. A. Bryan , Programmable controllers theory and implementation , second edition , Industrial Text Company publishing , USA , 1997 .

10. M. A. Laughton & D. J. Warme , Electrical engineer's reference book , sixteenth edition , Newnes , Great Britain , 2003 .

11. Ramesh S. Gaonkar , The Z80 microprocessor architecture programming and design , Merrill Publishing , USA , 1988 .

12. Claude A. Wiatrowski & Charles H. House , Logic Circuits and microcomputer systems , 3rd edition , Mc-Grew-Hill , 1984 .

13. Ramesh S. Gaonkar , Microprocessor architecture programming and applications , forth edition , Prentic-Hall , USA , 1999 .

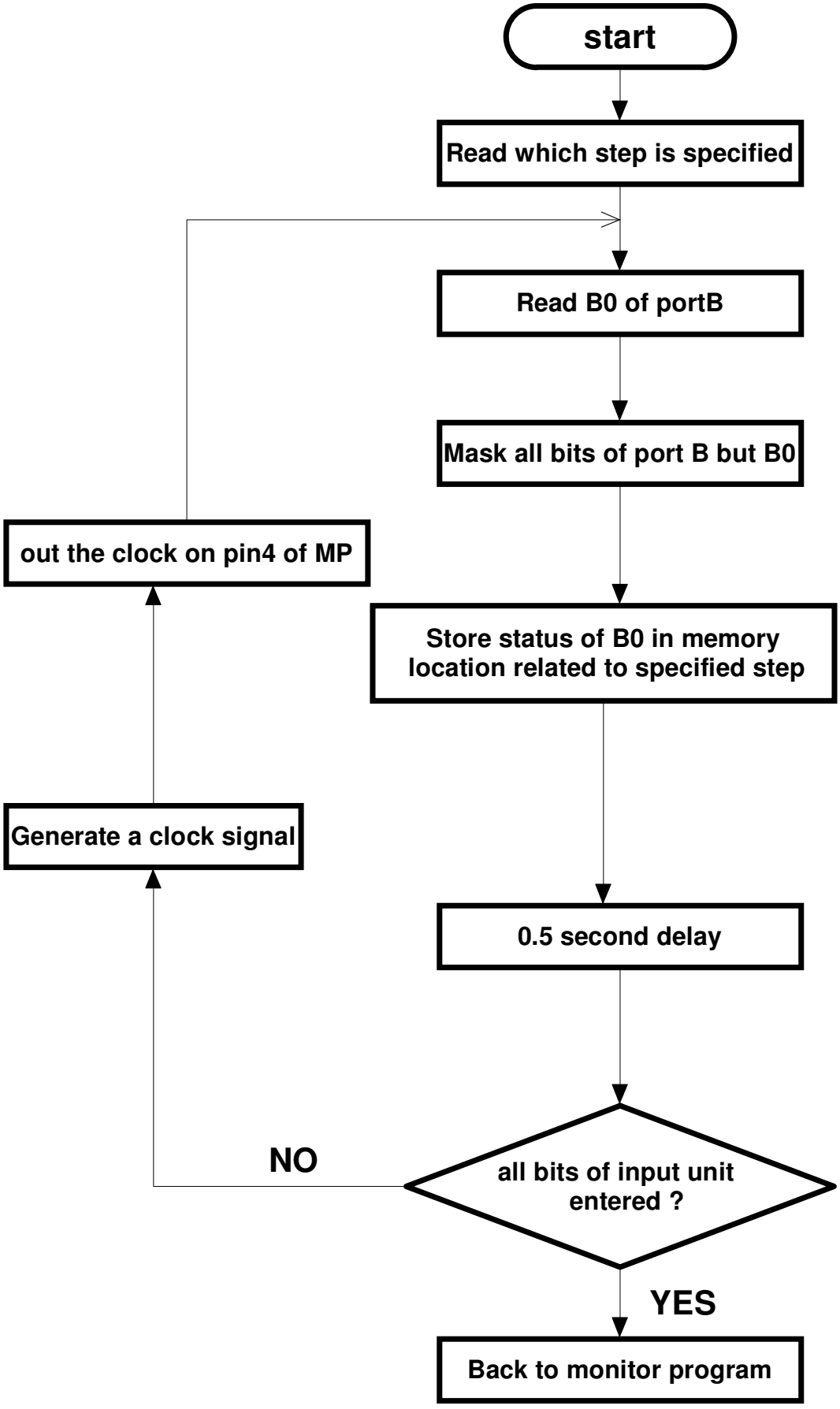
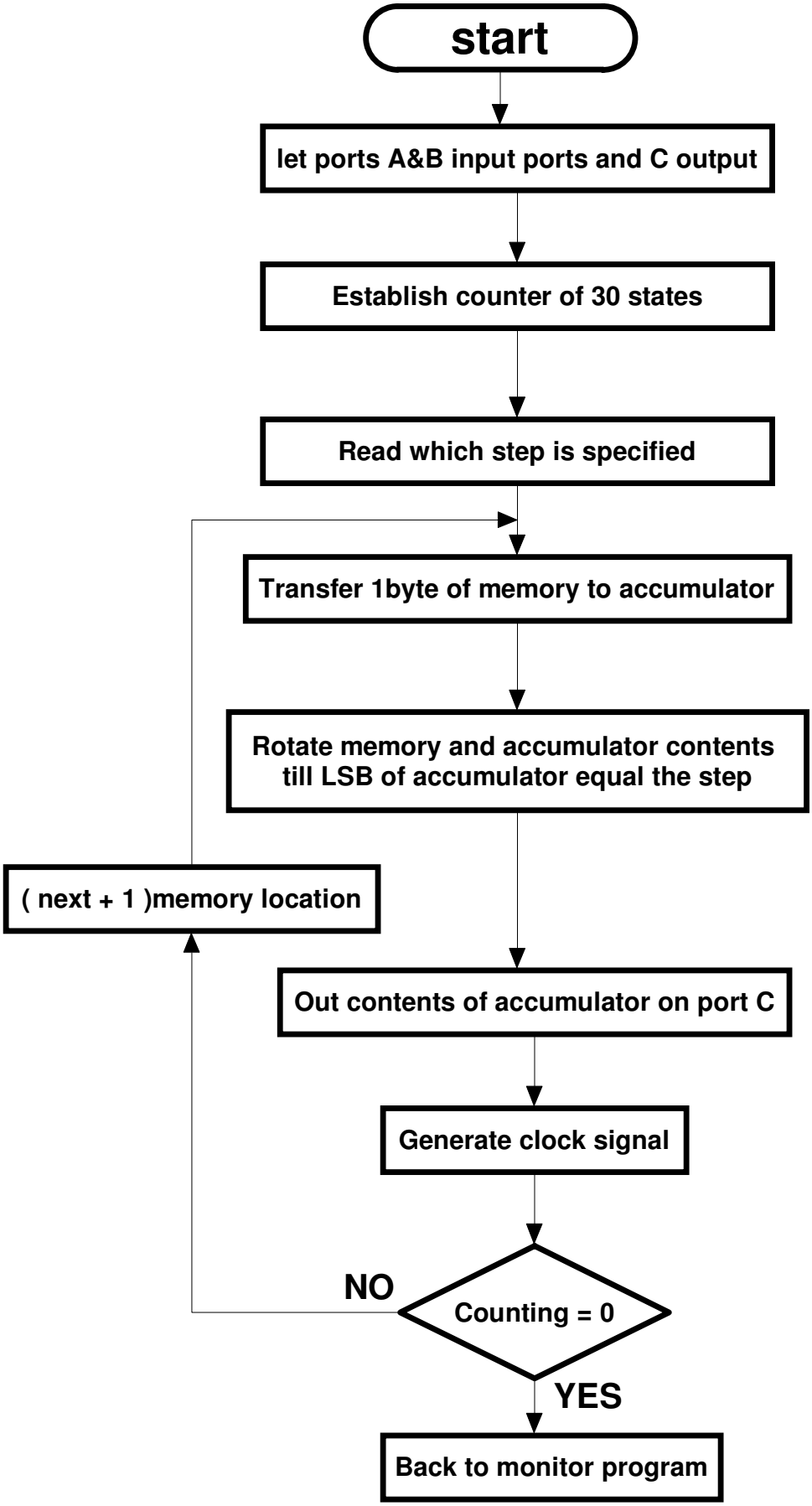
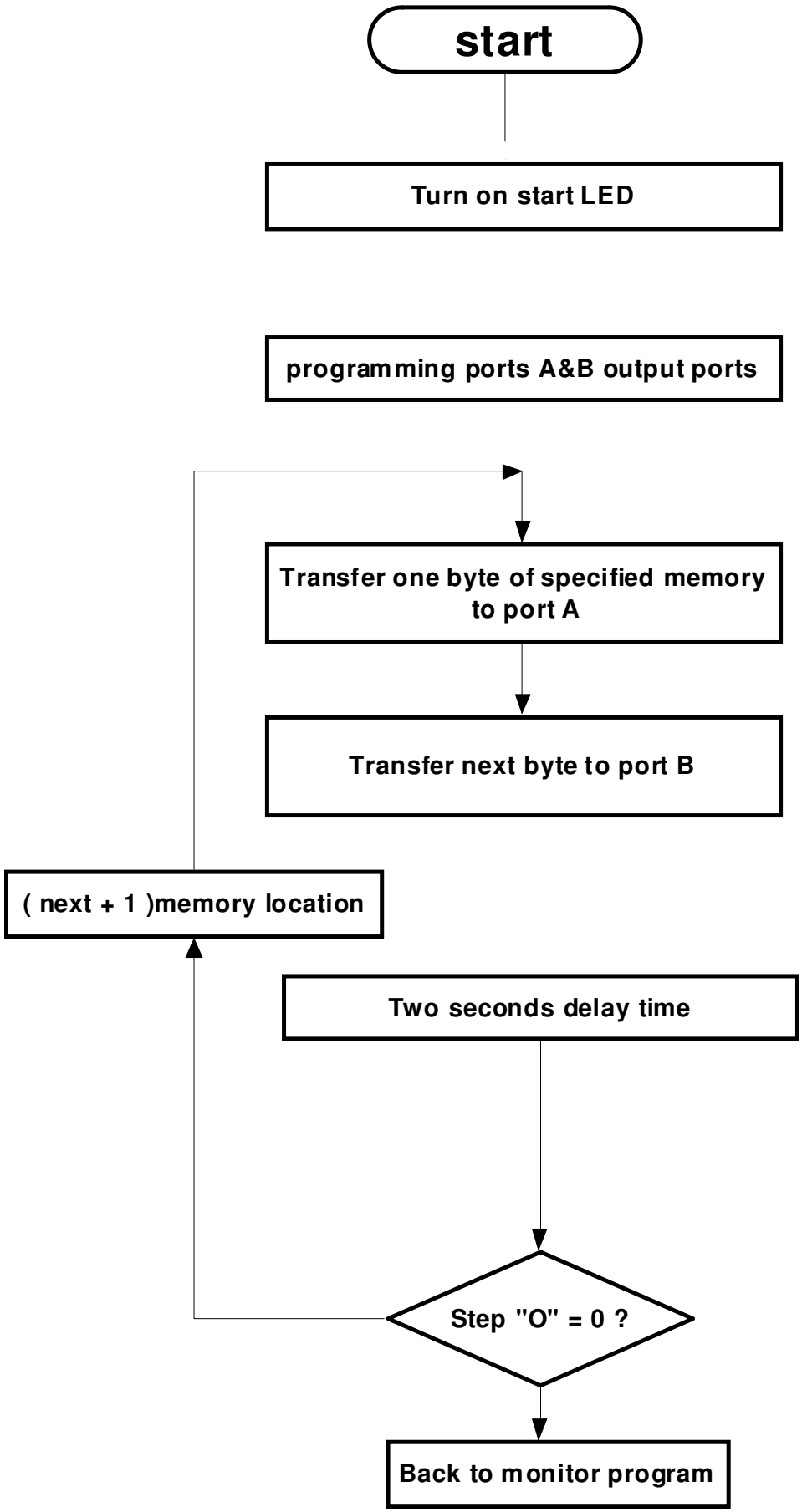


Figure (1) - input program



Figure(2) - check program



Figur(3) - run program

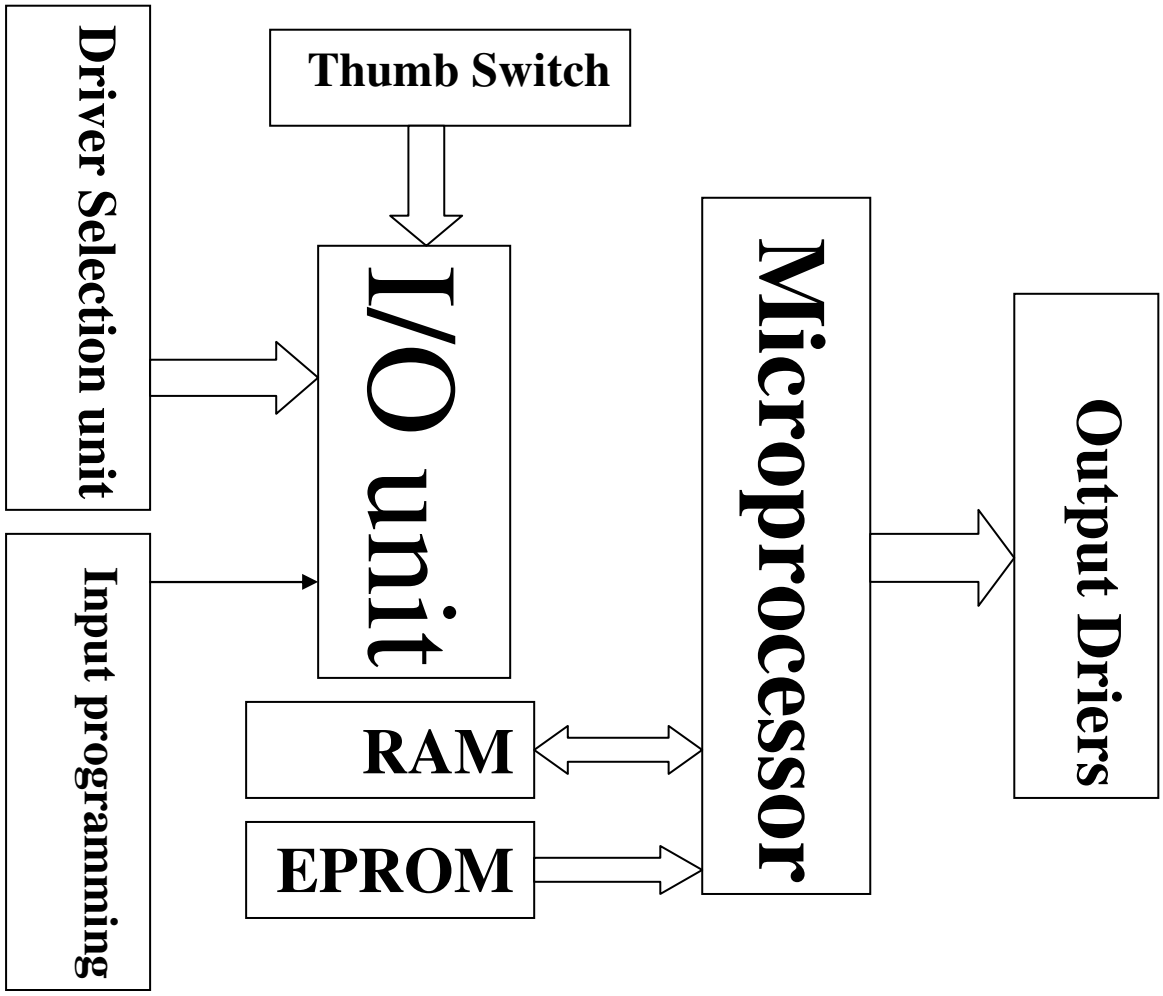
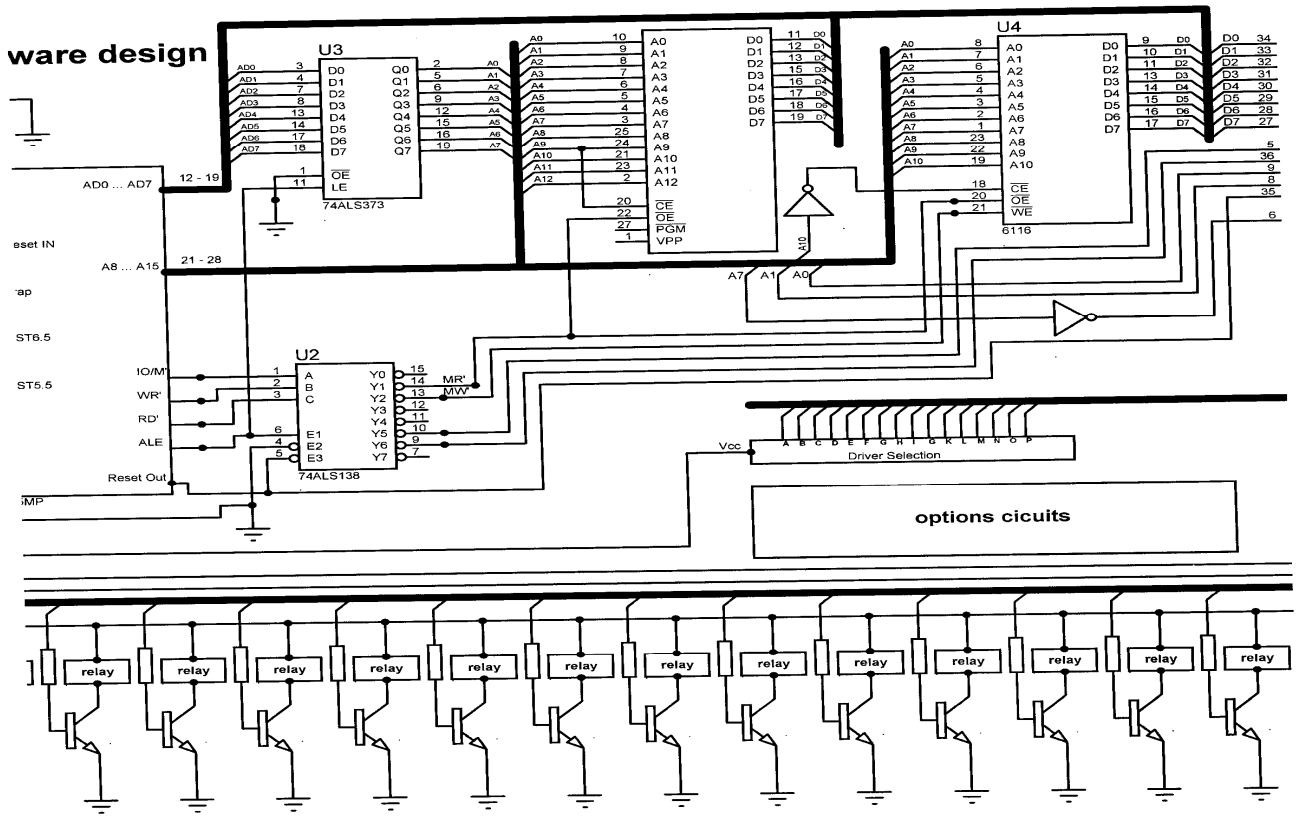
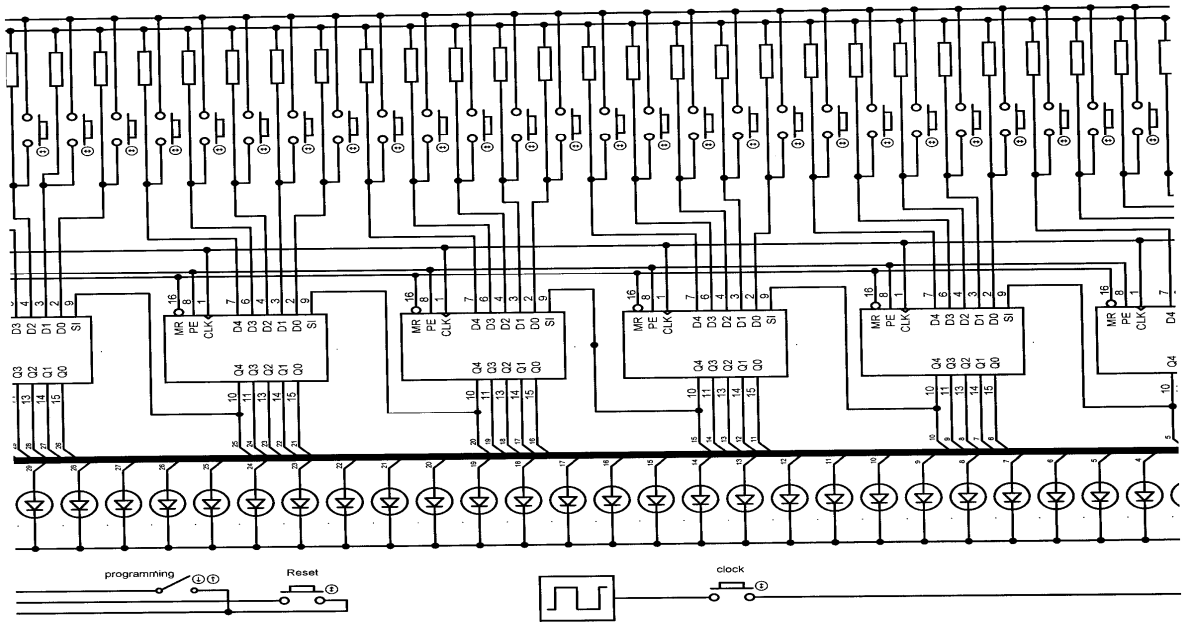


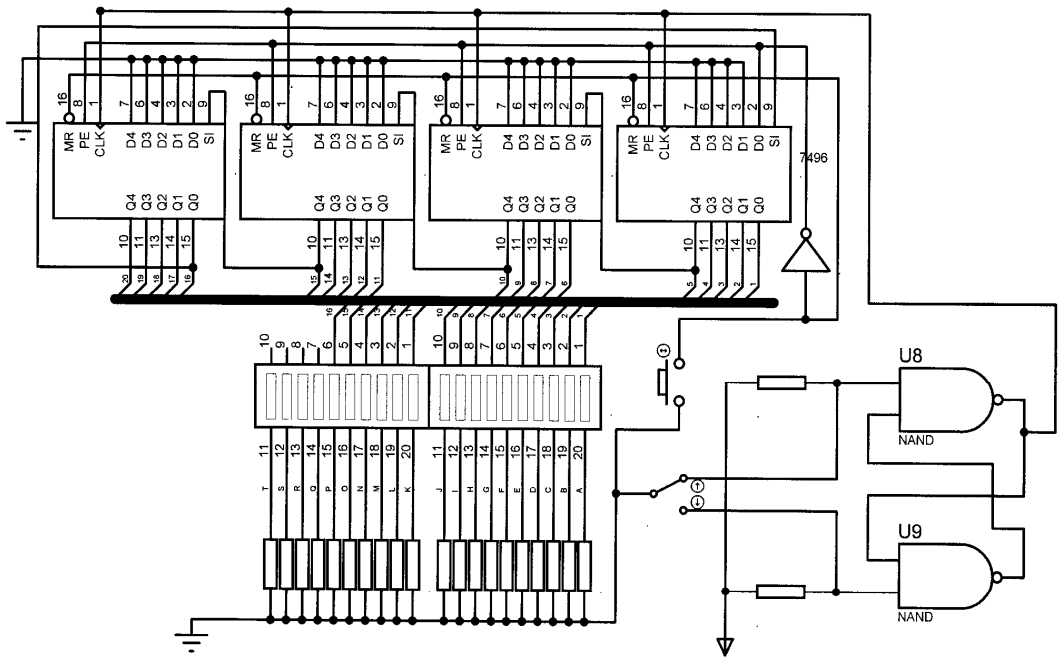
Figure (5) Basic construction of hardware approach



Details of main circuit



Input unit



Driver Selection unit