IJSES
International Journal of Scientific Engineering and Science
Volume 5, Issue 10, pp. 38-42, 2021. ISSN (Online): 2456-7361

Futsal Field Automatic Alarm Using the Microcontroller at Mega 8535

Jamal Abdullatif, S.Kom, MT, AM.d AP
Email address: jamallatif17(at)gmail.com

Abstract— *Now it is growing, sport in the eyes of the community that makes many people try, namely renting out the Futsal field. They are increasingly competing to make their Futsal field business more attractive and more modern Therefore we need a system that aims to set automatic alarms on the futsal field so that there is no need to remind players or visitors and futsal field users by pressing the alarm manually. Because until now there are several places that still depend on regular hours and sometimes always keep the time so as not to miss the time ordered from the customer, therefore we have a goal to build an alarm that automatically uses the ATMega 8535 microcontroller. In this final project we designing and making an Automatic Alarm device Using the ATMega 8535 Microcontroller. Where this tool functions by setting an automatic alarm by using a timer. The hardware manufacture in the form of a mechanical conveyor that uses ATMega 8535. While the ATMega 8535 Microcontroller as a control will sound an automatic alarm. Which will not depend on the controlled time manually, but a device designed using an ATMega 8535 microcontroller which will automatically support the futsal field*.

Keywords— *futsal field automatic alarm*.

I. BACKGROUND

Microcontroller makes a very popular controller base and has many significant benefits for machine language programming, and can regulate output and input by adjusting the program to be created. This final project will discuss how to design a relay, buzzer control system application with a microcontroller. The purpose of this final project is to design an alarm system based on the ATMega 8535 microcontroller. The computer and microcontroller are equipped with a means for alarms on the computer and a microcontroller has also been installed, which will act as input output on the computer to run the microcontroller which will act as the brain of the alarm design that communicates. data from the computer to the microcontroller will leak time to the alarm. Meanwhile, the microcontroller has installed the TxD pin for transceiver (sending) data serially and the RxD pin for serially received data. Until now, the existing facilities are serial data transfer from computer to microcontroller only. On the other hand, there is no such thing as a microcontroller to a computer. It is useful to control objects remotely by using. Microcontroller relay, buzzer and other component tools that will support to run the tool that is built later.

II. PROBLEM FORMULATION

The problems that will be raised in this Final Project are:
1. How to make a hardware driver on a data communication system between a computer and a microcontroller to build an alarm.
2. How to make software driver on a microcontroller system on alarm

III. FINAL PROJECT OBJECTIVE

The objectives of this Final Project are;
1. The purpose of this final project is to create a control system application based on a microcontroller which is equipped with feedback between computers with a microcontroller and Running LED Display via LCD.
2. This application functions to set alarms automatically and displays rental times to customers. In planning and making the "alarm With ATMega 8535 Microcontroller" the author needs to provide problem limits, namely:
3. Programming using CodeVision C language.
4. The alarm will turn on when the time that has been rented runs out
5. The working principle of the ATMega8535 microcontroller as the overall controller of the operation of the tool used.
6. In the manufacture of this tool using the ATmega8535 Microcontroller as a processing unit and port controller which functions as a control to the Buzzer and Running LED. The minimum system utilization of the ATmega 8535 Microcontroller is because this Microcontroller has features that are sufficient to meet all the needs of this system.

IV. THEORY EXPLANATION

1 Planning and manufacture of hardware (hardware)

a. Providing the AT Mega 8535 Microcontroller as a means to make hardware (hardware).
b. Create a circuit in the communication system between several components that will be used to build an alarm.
c. Providing a computer (PC) as a means for making software.

2 Circuit Running Led

Running LED is an electronic circuit that is often used as a decoration. Whether it's as decoration in private homes or in commercial places such as restaurants, parks, cafes and other places that are considered more suitable and attractive if added with glittering decorative lights. Walking lights have also been used as city decorations at night. So that the city looks more beautiful and attractive. Basically all series of decorative lights use the same working principle as running lights, namely utilizing alternating output conditions or shift registers so that with these conditions various combinations of lights can be

International Journal of Scientific Engineering and Science
Volume 5, Issue 10, pp. 38-42, 2021. ISSN (Online): 2456-7361

made. Moreover, if the combination can be matched with the appropriate color arrangement, so that it will create a beauty that is pleasing to the eye. To make a series of decorative lights is actually quite easy. Because you don't need to exhaust your mind to do a work analysis on the circuit and also don't need to make adjustments to the circuit to get maximum results. All you need is a clock signal generating circuit and an alternating shift register or output generating circuit. You can use any circuit as a clock signal generator, such as a transistor oscillator circuit or an astable IC 555 circuit. Then to get an output that has rotating logic we can use IC 4017 which is often known as the Johnson Counter and is most often used in running light circuits. IC 4017 has 10 outputs which are counted in turn, starting from O0 (pin 3) to O9 (pin 11). You can see a description of the IC 4017 pin along with the truth table in the image below.

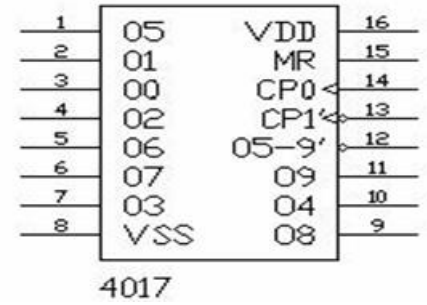

Figure 1. Truth Table

The series of running lights above uses ten LEDs as output indicators. The flashing speed of the lamp is determined by the values of R1, C1 and VR1. The greater the value of the three components, the longer the time period and vice versa. The output of IC 4017 has a very limited current supply so we must be able to use transistors, SCRs or relays.

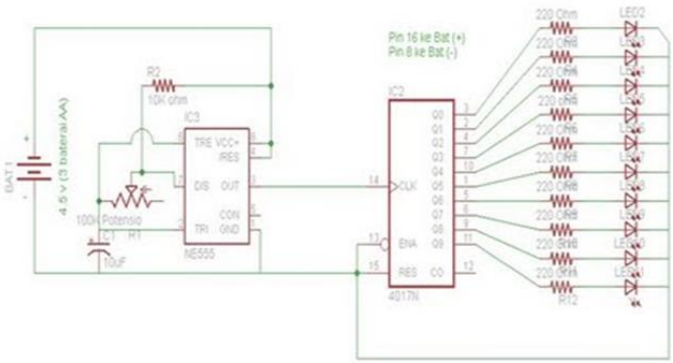
Figure 2. Pin pins in IC

1. Rotate the potentiometer to change the sliding speed of the LED.
2. The animation of the LEDs can be arranged in such a way that they form the desired movement, for example, by placing the LED array 1-10-2-9-3-8-4-7-5-6, a "cyclops" effect or back and forth will be obtained.
3. IC 555 & 4017 can be given input up to 12V, so it can turn on the LEDs that are installed in series as much as 5 pieces. Try to be creative by adding LEDs in each output leg and try to combine it with a combination of colorful LEDs.
4 If you use more LEDs, you can use a common transistor such as BD139 or MJ31, as shown in the picture. Adjust the input voltage until the LED is bright. Basically if the LEDs are in series then the required input voltage follows the formula.

$$V_{in} = V_{forward} * n_{led}$$

In general, LEDs have a V forward of 2.4 V, so if you use 5 LEDs it takes about 12 Volts. If you use more than 5 LEDs (such as 20 LEDs) then 5 LEDs are arranged in series and then paralleled with 3 other series circuits. Because the current from 4017 is quite small, a transistor is needed to carry a larger current. quoted from led.

Make the hardware with 12 leds where the first 8 leds are connected to data pins (2-9) and the last 4 leds are connected to control pins (1,14,16,17) add a resistor and a cathode system connection.
2. Create a new project then add 12 shapes for the led simulation, 1 label for the title then 5 command buttons and 4 timers.
3. Then the design becomes like the following picture

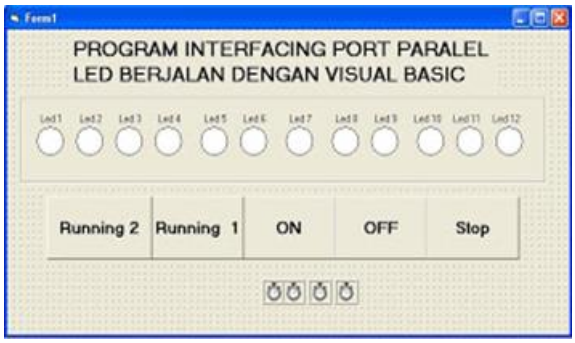

Figure 3. LED Parallel Port Interfacing Program running with Visual Basic

4. Set the properties of each component such as for the shape backstyle change to 1-opaque then backcolornya select white color then for the timer all intervals give a value of 100 and in a state of false.
5. Save the project and Form in the same folder and then copy the input32.dll in the folder where it is stored.

3 Buzzer

Figure 4. Buzzer

Buzzer is a device that can convert electrical signals into sound signals. In general, buzzers are used for alarms, because their use is quite easy, namely by providing an input voltage, the buzzer will make a sound. The sound frequency emitted by the buzzer is between 1-5 KHz.

International Journal of Scientific Engineering and Science
Volume 5, Issue 10, pp. 38-42, 2021. ISSN (Online): 2456-7361

4 Relay

Relay is an electronic component (circuit) which is electronic and simple and composed of a switch, coil, and iron shaft. The use of this relay in electronic devices is very much. Especially in devices that are electronic or automatic. For example on television, lights and others.

Figure 5. Relay

The way this component works begins when an electric current flows through the coil, then creates a magnetic field that can change the position of the switch in the relay. So it can produce a larger electric current. That's where the virtue of this simple component is that with its minimal form it can produce a larger current.

The use of relays in electronic devices has the following advantages:

1. Can control the desired electric current and voltage.
2. Can maximize the amount of electric voltage until it reaches its maximum limit.
3. Can use both switches and coils more than one, according to their needs.

5 Microcontroller

Microcontroller is a computer system that all or most of its elements are packaged in one IC chip, so it is often called a single chip microcomputer. Microcontroller is a microprocessor system in which there is already a CPU, ROM, RAM, I / O, Clock and other internal equipment that are interconnected and well organized (observed) by the manufacturer and packaged in a ready-to-use chip.

6 ATMega 8535

The ATMega8535 is a low power 8bit CMOS microcontroller based on the RISC architecture. Instructions are executed in one clock cycle, ATMega8535 has a throughput of close to 1 MIPS per MHz, this makes ATMega8535 able to work at high speeds even with low power usage. The ATmega8535 microcontroller has several features or specifications that make it an effective controller solution for various purposes. These features include:

1. 32 I/O channels, consisting of Ports A, B, C and D
2. ADC (Analog to Digital Converter)
3. Three Timers/Counters with comparison capabilities
4. CPU consisting of 32 registers
5. Watchdog Timer with built-in oscillator
6. 512 bytes of SRAM
7. Flash memory of 8kb with the ability to read while write
8. Internal and External Interrupt Units
9. SPI interface port to download program to flash
10. EEPROM of 512 bytes which can be programmed during operation
11. Analog comparator interface
12. USART port for serial communication.

7 Definition of Microcontroller

Microcontroller is a computer system that all or most of its elements are packaged in one IC chip, so it is often called a microcontroller single-chip microcomputer. Furthermore, a microcontroller is a computer system that has one or several very specific tasks, in contrast to a PC (Personal Computer) which has various functions. Another difference is the very different ratio of RAM and ROM between computers and microcontrollers. Microcontroller is a microprocessor system in which there is already a CPU, ROM, RAM, I / O, Clock and other internal equipment that are interconnected and well organized (observed) by the manufacturer and packaged in a ready-to-use chip. So we just need to program the contents of the ROM according to the manufacturer's usage rules, according to Winoto (2008:3). The technology used in the AVR microcontroller is different from the MCS-51 series microcontroller. The AVR has RISC (Reduced Instruction Set Computer) technology, while the MCS-51 series has CISC (Complex Instruction Set Computer) technology. AVR microcontrollers can be grouped into four classes, namely the ATtiny family, AT90Sxx family, ATMega family, and AT89RFxx family. Basically what distinguishes each class is memory, peripheral completeness and additional functions it has. The following is a more complete explanation of the ATMega8535 Microcontroller:

8 ATMega8535. Pin Configuration

ATMega AVR microcontroller has 40 pins with 32 pins of which are used as parallel ports. One parallel port consists of 8 pins, so the number of ports on the microcontroller is 4 ports, namely port A, port B, port C and port D. For example, port A has pins between port A.0 to port A.7, thus then for port B, port C, port D. The microcontroller pin diagram can be seen in the following figure:

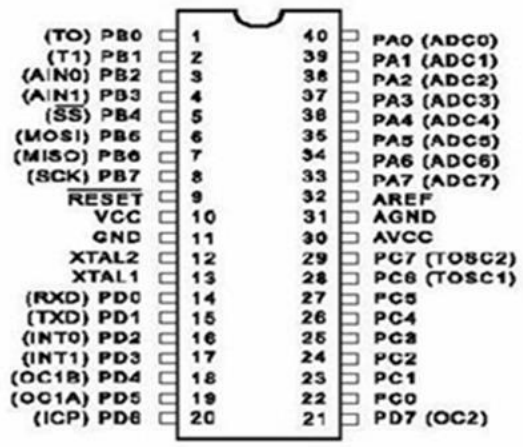

Figure 6. ATMEGA 8535. PIN DIAGRAM

The following is an explanation table regarding the pins contained in the ATMega8535 microcontroller:

Table Explanation of pins on the ATMega8535. Microcontroller

International Journal of Scientific Engineering and Science
Volume 5, Issue 10, pp. 38-42, 2021. ISSN (Online): 2456-7361

TABLE 1. Pin Microcontroller ATMega 8535

Vcc	Supply voltage (5 volt)
GND	Ground
RESET	Low level reset input, on this pin for more than the minimum pulse length will result in a reset even though the clock is running. RST on pin 9 is a reset of the AVR. If this pin is given a low input for at least 2 machine cycles, the system will be reset
XTAL 1	Inverting oscillator amplifier input and input to the internal clock operating circuit
XTAL 2	The output of the inverting oscillator amplifier
Avcc	Supply voltage pins for port A and ADC. This pin must be connected to Vcc even if the ADC is not used, it must be connected to Vcc through a low pass filter
Aref	analog voltage reference pin for ADC
AGND	pin for analog ground. Connect this pin to GND, unless the board has a separate analog ground

The following is an explanation of the ATMega8535 microcontroller pins according to their respective ports:

1. *Port **A***

Pin 33 to pin 40 are the pins of port A. It is an 8-bit directional I/O port. Each pin can provide an internal pull-up resistor (adjustable per bit). The port A output buffer can provide 20 mA of current and can control the LED display directly. Data Direction Register port A (DDR) must be set before port A is used. The DDRA bits are set to 0 if you want to use the adjusted port A pins as input, or 1 if you want to use the output. In addition, the pins on port A also have special alternative functions as can be seen in the table.

TABLE 2. Table Explanation of pins on port A

Pin	**Information**
PA.7	ADC7 (ADC *Input Channel* 7)
PA.6	ADC6 (ADC *Input Channel* 6)
PA.5	ADC7 (ADC *Input Channel* 5)
PA.5	ADC4 (ADC *Input Channel* 4)
PA.3	ADC3 (ADC *Input Channel* 3)
PA.2	ADC2 (ADC *Input Channel* 2)
PA.1	ADC1 (ADC *Input Channel* 1)
PA.0	ADC0 (ADC *Input Channel* 0)

2. Port B

Pin 1 to pin 8 is the pin of port B. It is an 8 bit directional I/O port. Each pin can provide an internal pull-up resistor (adjustable per bit). The output buffer port B can provide 20 mA of current and can control the LED display directly. Data Direction Register port B (DDRB) must be set before port B is used. DDRB bits are set to 0 if you want to function the adjusted port B pins as input, or 1 if you want to use the output. In addition, port B pins also have special alternative functions as can be seen in the table:

TABLE 3. Table Explanation of pins on port B

Pin	**Information**
PB.7	SCK (SPI *Bus Serial Clock*)
PB.6	VISO (SPI *Bus Master Input/Slave Output*)
PB.5	VOSI (SPI *Bus Master Output/Slave Input*)
PB.4	SS (SPI *Slave Select Input*)
PB.3	AIN1 (*Analog Comparator Negative Input*) OCC (*Timer/Counter0 Output Compare Match Output*)
PB.2	AIN0 (*Analog Comparator Positive Input*) INT2 (*External Interrupt2 Input*)
PB.1	T1 (*Timer/Counter1 External Counter Input*)
PB.0	T0 (*Timer/Counter0 External Counter Input*) XCK (JSART *External Clock Input/Output*)

3. Port C

Pin 22 to pin 29 are the pins of port C. Port C itself is an input or output port. Each pin can provide an internal pull-up resistor (adjustable per bit). The port C output buffer can provide 20 mA of current and can control the LED display directly. Data Direction Register port C (DDRC) must be set before port C is used. DDRC bits are set to 0 if you want to use the adjusted port C pins as input, or 1 if you want to use the output. In addition, port D pins also have special alternative functions as can be seen in table:

TABLE 4. Table Explanation of pins on port C

Pin	**Information**
PC.7	TOSC2 (*Timer Oscillator Pin* 2)
PC.6	TOSC1 (*Timer Oscillator Pin* 1)
PC.1	SDA (*Two-Wire Serial Bus Data Input/Output Line*)
PC.0	SCL (*Two-Wire Serial Bus Clock Line*)

In the block diagram ATMega8535 depicted 32 general purpose Working registers that are connected directly to the Arithmetic Logic Unit (ALU). This allows two different registers to be accessed in one clock cycle

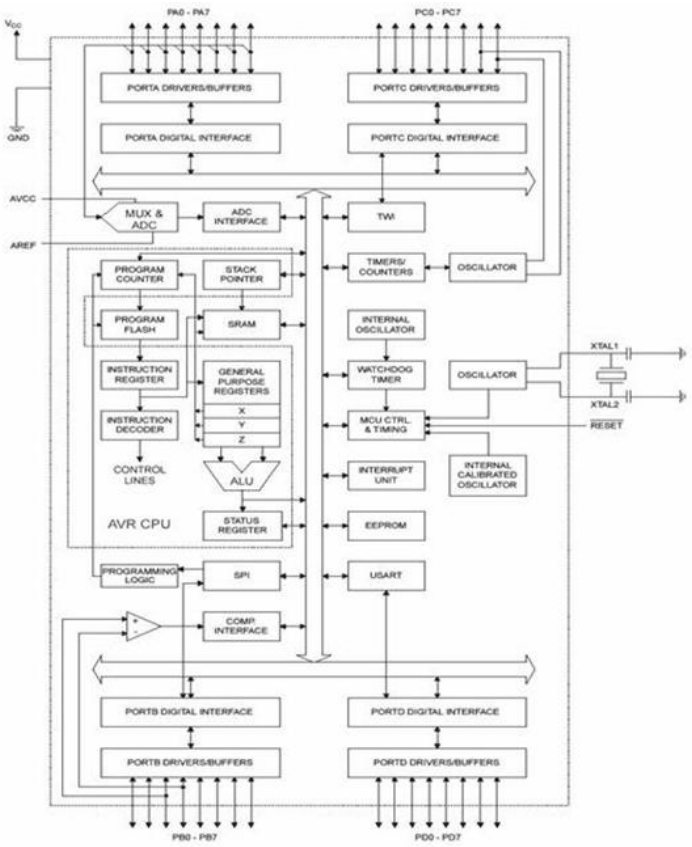

Figure 7. Microcontroller block diagram at mega 8535

V. RESEARCH METHOD PLANNING

The software design that is loaded must be integrated with all supporting devices used in this tool and can be connected to each other according to its function. And can process any given input.

International Journal of Scientific Engineering and Science
Volume 5, Issue 10, pp. 38-42, 2021. ISSN (Online): 2456-7361

1 Interface Visual Basic 6.0

Making application programs using Visual Basic is done by making the application display on the form, then given a script program in the required components. The form is composed of components that are in the [Toolbox], and each component that is used must have its properties set through the [Property] window. Menus are basically standard operations in the Windows operating system, such as creating a new form, creating a new project, opening a project and saving a project. In addition, there are facilities for using visual basic on the menu. For more details Visual Basic provides a very complete and detailed help in MSDN.

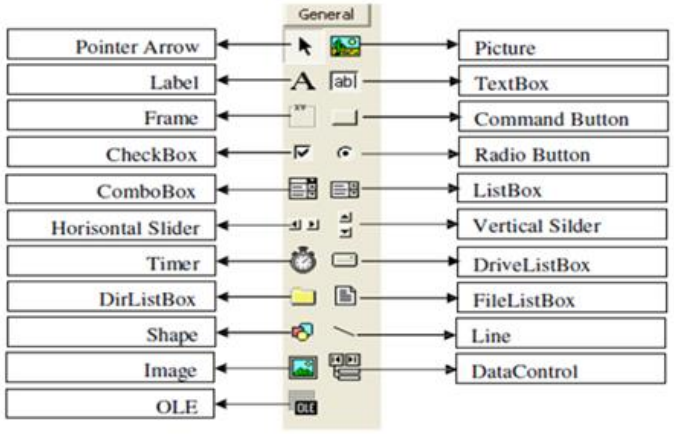

Figure 8. Visual Basic 6.0

2 Basic Concepts of Programming in Visual Basic 6.0

The basic concept of programming Visual Basic 6.0, is making a form by following the programming rules of Property, Method and Event. This means:

(1) Property: Each component in Visual Basic programming can be set its properties according to application needs. The property that should not be forgotten on each component is "Name", which means the name of the variable (component) that will be used in scripting. The “Name” property can only be set through the Property window, while the other property values can be set via scripts such as

Command1.Caption=”Play”
Text1.Text=”Visual Basic”
Label1.Visible=False
Timer1.Enable=True

(2) Method: That the running of the program can be adjusted according to the application by using a programming method that is set as the action of each component. This method is a place to express the programming logic of making an application program.

(3) Events: Each component can act through events, such as the click event on the command button written in the Command1_Click script screen, or the Mouse Down event on the picture written with Picture1_MouseDown. Setting events in each component that will execute all methods created.

3 Program Planning in Microcontroller Systems

One of the main programs as well as the basis for controlling this device is a microcontroller whose function is to process inputs through switches and computers to run other supporting devices. For programming, the microcontroller must be designed as well as possible by taking into account the features available on each supporting device so that all programs can run properly. For more details see the flowchart image.

Block Diagram:

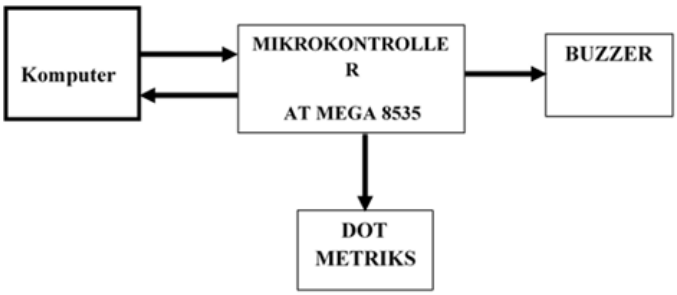

Figure 9. Block Diagram

4 How Block Diagrams Work

The computer sends the time code to the AT Mega 8535 micro controller. The timer is displayed via the LCD and visuals on the monitor and when the time is set, the micro controller will send an alarm or buzzer indicating the time has run out then the alarm is turned off manually and to calculate the time counter on the micro controller AT Mega 8535 returns from scratch or, from scratch.

5 Flow Chart

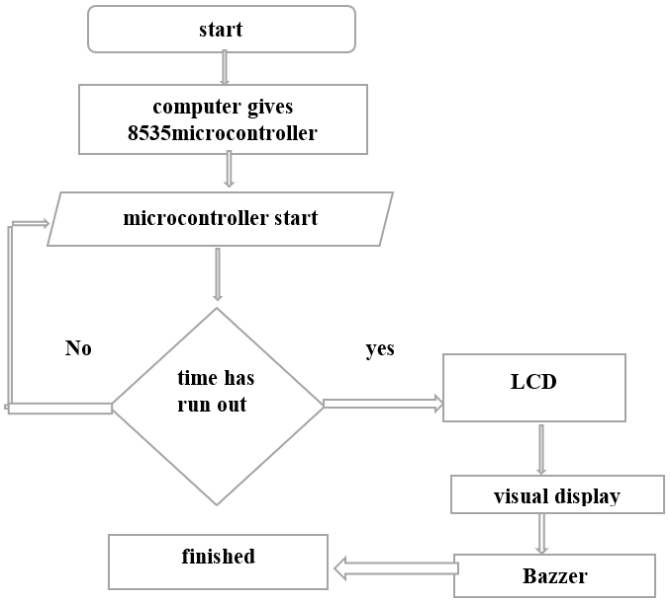

Figure 10. Flow Chart

REFERENCES

(1) http://beskem.net/pengertian-mikrokontroler-atmega8535/
Pccontrol, RS232 / RS-485 Programming w/ Visual Basic 6, http://www. Programming RS232 _ RS-485 with Visual Basic 6 « Controlling the equipment with a Computer.html,
http://aisi555.blogspot.com/2011/07/practice-led-walking-with-ic-4017.html
http://the-overgameonline.blogspot.com/2012/05/buatprogram-port-parallel-led.html