

raf

TK5105.75 .S52 2005



0000010569

Broadband application for home electrical appliances /
Shahharrulrizad Mohamad Yunus.

**BROADBAND APPLICATION FOR HOME
ELECTRICAL APPLIANCES**

SHAHHARRULRIZAD BIN MOHAMAD YUNUS

MARCH 2005

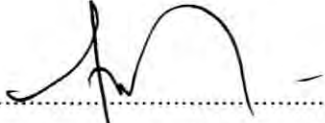
BROADBAND APPLICATION FOR HOME ELECTRICAL APPLIANCES

SHAHHARRULRIZAD BIN MOHAMAD YUNUS



**FACULTY OF ELECTRICAL ENGINEERING
KOLEJ UNIVERSITI TEKNIKAL KEBANGSAAN MALAYSIA
2005**

“I hereby admit have read this journal and in my opinion this journal is enough for the scope and quality in the purpose of degree bestowal in Bachelor of Electrical Engineering (Power Industry).”

Signature : 

Name of the Supervisor : En Mohamed Azmi Bin Said

Date : 10 - 3 - 2005

BROADBAND APPLICATION FOR HOME ELECTRICAL APPLIANCES

SHAHHARRULRIZAD BIN MOHAMAD YUNUS


This Report Is Submitted In Partial Fulfillment Of Requirement For
The Degree Of Bachelor In Electrical Engineering (Industry Power)

Faculty of Electrical Engineering
Kolej Universiti Teknikal Kebangsaan Malaysia

March 2005

“I declared that this thesis entitled ‘Broadband Application for Home Electrical Appliances’ is the result of my own except as cited in references. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any degree”

Signature

: 

Name of the Candidate

: Shahharrulrizad Bin Mohamad Yunus

Date

: 10 MARCH 2005

To most regretted my parents,
My family,
Lecturers and KUTKM staffing,
My friends,
For those have committed and sacrifice.

ABSTRACT

This project involves designing that is “Broadband Application for Home Electrical Appliances”. The main reason of this project is to design one software application which is combined with internet broadband facility which is used to control electrical circuit and electrical equipment at home. This project is developed using software and hardware. The software’s used is Visual Basic, Apacer Server and Assembler for micro-controller. For hardware, one micro-controller with complete circuit needs to control or execute the command which is passed through server. The entire project needs a personnel computer which is used as a server. This server will be connected directly to internet line all the time. Communication between server and micro-controller will be connected using serial port connection. After that from the micro-controller it will be connected to electrical wiring circuit using extra circuit (relay circuit). So that, micro-controller will receive command from the server and it will execute the command such as Switch “On” and Switch “OFF”. From this, the user will able to control electrical house equipments using internet line. For this project, the difficulty part is doing the interface between server and micro-controller. The serial port need to be programmed until it is able to receive and send data using Assembler 8051 and Visual Basic softwares. If this problem is solved, the project can be completed.

ABSTRAK

Projek ini melibatkan reka cipta iaitu “Aplikasi Jalur Lebar kepada Perkakasan Elektrik di Rumah”. Tujuan utama projek ini adalah untuk mereka cipta satu perisian yang dikombinasikan dengan kemudahan internet jalur lebar bagi mengawal litar elektrik dan juga perkakasan elektrik di rumah. Pembikinan melibatkan penggunaan perisian (software) dan perkakasan (hardware). Antara perisian yang digunakan ialah Visual Basic, Apacer Server dan Assembler bagi pengawal mikro (micro-controller). Untuk perkakasan pula, satu pengawal mikro dengan litar lengkap diperlukan bagi melaksanakan arahan yang disampaikan dari server. Secara keseluruhannya, projek ini memerlukan satu komputer peribadi yang akan digunakan sebagai server. Server ini pula akan disambungkan secara berterusan kepada talian internet di sepanjang masa. Komunikasi di antara server dan pengawal mikro disambung menggunakan sambungan serial port. Kemudian dari pengawal mikro ini pula, ia akan disambungkan pula kepada litar pendawaian elektrik menggunakan litar tambahan (relay circuit). Oleh hal yang demikian pengawal mikro akan menerima arahan dari server dan ia akan melaksanakan arahan tersebut seperti Switch “On” dan Switch “OFF”. Dengan ini pengguna mampu mengawal peralatan elektrik rumahnya secara kawalan jauh melalui talian internet. Bagi projek ini, kesukaran yang dihadapi adalah membuat antara muka di antara sever dan pengawal mikro. Serial port ini perlu diprogramkan sehingga ia mampu untuk menerima data menghantar data dengan menggunakan perisian Assembler 8051 dan juga Visual Basic. Sekiranya masalah ini dapat ditangani, penyiapan projek akan lebih lancar.

TABLE OF CONTENTS

<u>Chapter</u>	<u>Topic</u>	<u>Page</u>
	Abstract	i
1	Introduction	
	1.1 Project Overview	1
	1.2 Objectives	
	1.2.1 To understand the software usage for combination purpose	2
	1.2.2 To understand network application like LAN and WAN	2
	1.2.3 To Develop Web Application System for User Requirement	3
	1.2.4 To understand TCP/IP protocol	3
	1.2.5 To Understand Micro-Controller Usage	3
	1.3 Scope	3
	1.4 Expected Result	4
2	8051 Overview	
	2.1 Introduction	5
	2.2 A Brief History of the 8051 Family	6
	2.3 8051 Architecture	7
	2.4 8051 Cross Assembler Overview	9
	2.4.1 Symbols	9
	2.4.2 Labels	11
	2.4.3 Assembler Directives	11
	2.5 8051 Instruction Mnemonics	12

2.6	Assembling and Running An 8051	13
2.7	Serial Port Control and Status Register (SCON)	14
2.8	Serial Port Modes	16
2.9	Timer 1 as Baud Rate Generator	17
2.10	The PCON Register	18
2.11	Types of Memory 8051	20
	2.11.1 Code Memory	21
	2.11.2 External RAM	21
	2.11.3 On-Chip Memory	22
	2.11.4 Register Banks	23
	2.11.5 Bit Memory	24
	2.11.6 Special Function Register (SFR) Memory	26
2.12	8255 General overview	27
	2.12.1 PIN Configuration	28
	2.12.2 Block diagram	29
	2.12.3 Control Word Register	30
3	Serial Communication	
3.1	Introduction	32
3.2	Interfacing the Serial / RS232 Port	32
3.3	RS-232C (EIA-232) Serial Interface	34
3.4	Serial Port's Registers (PC's)	36
3.5	Interfacing Devices to RS-232 Ports	38
	3.5.1 RS-232 Waveforms	38
	3.5.2 RS-232 Level Converters	39
4	Broadband	
4.1	Introduction	42
4.2	Understand to Broadband	42
4.3	Why Broadband?	43
4.4	The Advantage of Broadband	43

4.5	The Internet, Intranet and Extranet	45
4.6	TCP/IP Protocol	46
5	Electrical Wiring And Relay Circuit	
5.1	Introduction	48
5.2	Electrical Wiring Circuit	48
5.3	Relay	51
	Discussion	54
	Conclusion	55
	References	56

LIST OF FIGURE

<u>No</u>	<u>Title</u>	<u>Page</u>
2.2	Inside the 8051 Microcontroller Block Diagram	6
2.7	Serial Port Control Register	15
2.10	Power Control Register	19
2.11	Types of Memory 8051	20
2.11.3	Memory Map	22
2.12.1	Pin 8255 Configuration	28
2.12.2	Block diagram of the 8255	29
2.12.3.1	Mode definition of the 8255 control registers to configure the 8255	30
2.12.3.2	Bit definitions of the 8255 control register to modify single bits of port C	31
3.5.1.1	TTL/CMOS Serial Logic Waveform	38
3.5.1.2	RS-232 Logic Waveform	39
3.5.2.1	Pin outs for the MAX-232, RS-232 Driver/Receiver	40
3.5.2.2	Typical MAX-232 Circuit	40
5.2.1	Schematic Diagram Home Wiring Circuit	49
5.2.2	Schematic Lighting Circuit	50
5.2.3	Schematic Power Circuit	51
5.3.1	1 Way Relay Circuit	52
5.3.2	2 Ways Relay Circuit	52

LIST OF TABLE

<u>No</u>	<u>Title</u>	<u>Page</u>
2.7.1	Serial Port Control Register	15
2.7.2	Status Register	16
2.10	Power Control Register and Status Register	20
3.3.1	D Type 9 Pin and D Type 25 Pin Connectors	35
3.3.2	Pin Functions Summary	36
3.4.1	Standard Port Addresses	36
3.4.2	COM Port Addresses in the BIOS Data Area	37
3.4.3	Table of Registers	37

CHAPTER 1

INTRODUCTION

1.1 Project Overview

Nowadays, the broadband service application is widely used. Revolution for daily existence its given more benefits to consumers. The project of Broadband Application for Home Electrical Appliances becomes is one alternative to make sure of that the technology convenience is used to the fullest.

This broadband applications for electrical home appliances has becomes a new season for convenience it will be introduced to the consumer especially for the consumer which always does not return back home because occupational arrangement and as being likely corporate people and it is applicable utility for consumers who emphasize security at home place.

This project constructed by usage of the software and hardware. The example of softwares used is Visual Basic, Micro-controller Assembler (ASM51) and Apacer Server (to create the client server). For hardware, this project needs one complete micro-controller circuit, relay and house wiring (power circuit and lighting circuit).

Visual Basic software will be used to build one program for interface between computers (server) to micro-controller. Connection from server to micro-controller is made in serial that is by using serial port on the server. High possibility, that this part is a difficult section and is a problem to this project. For micro-controller, it has its program that is dependable to the assembler which suits with micro-controller chip (8051). For Apacer Server software, it is used as Host to Client (connection to the internet).

For hardware, micro-controller complete circuit will be used for building one micro-controller which operates as enforcement. Micro-controller will be receiving

input from the Server that is command as “Switch On (1) or Switch OFF (0)” and after that micro-controller will release output (0 or 1). “0” means voltage= 0volt while “1”means voltage= 5 volt. This voltage will switch on the transistor which is connected to the micro-controller.

From the micro-controller, it will be connected to relay. Relay has the same functions as the switch. From the relay, it will connect to the house wiring like the lighting circuit and power circuit. So the electrical equipments are able to be controlled automatically and manually.

1.2 Objectives

1.2.1 To understand the software usage for combination purpose.

The software is used for programmed the IC (EEPROM) to control the circuit. The program is programmed in ASM file. This file programmed using by ASM51.exe. Software Visual Basic is used for create a new program for window switch “ON” and switch “OFF” which we put it on server. For Apacer Server is used to create personal server which this server we must be ON it at all the time. The software of Php, MySql and Dreamweaver FX is used to write the database and create a new personal website.

1.2.2 To understand network application like LAN and WAN.

This project can help to understanding the applications of the Network and System Administrators, how the flow through the WAN/LAN infrastructure. These ways can perform metrics about network traffic, applications, latency and response time.

1.2.3 To Develop Web Application System for User Requirement

The function of the web application system is to capture the incoming data, to process the data and to produce an analyzed data. The data will be sent to server. The server will be identify the data and after that its will be sent to serial port at COM 1 and after that the serial port will be sent to micro-controller. The micro-controller will be operate the instructive or command as switch "ON" or switch "OFF". Web application system also has the networking function, which allows users from other location to share the data through the network or the Internet.

1.2.4 To understand TCP/IP protocol.

The name TCP/IP refers to a suite of data communication protocols. The name is misleading because TCP and IP are only two of dozens of protocols that compose the suite. Its name comes from two of the more important protocols in the suite: the Transmission Control Protocol (TCP) and the Internet Protocol (IP).

1.2.5 To Understand Micro-Controller Usage

The micro-controller is used for control whole the system. The micro-controller will be received the data from the server and after that the micro-controller will be to operate the command to control the wiring circuit.

1.3 Scope

This project concentrates on functional of Micro-controller. First target is, Micro-controller must be success fully running and it will be programmed using by

the microchip 8051 Assembler (ASM51.exe) to control the system. The focus of this project will be control the wiring circuit using external source. Second part, the scope will be focus to interface between Server with Micro-controller using the serial port to processing data, classifying data and analyzing data and transmit the data. The final scope is the server will be communicating to the website. So at the final scope, the project can be use the Internet to control the electrical wiring circuit.

1.4 Expected Result

The expected result is, the project can be accomplish the whole of scope that given that is we can be control the home electrical appliances using the internet broadband connection.

CHAPTER 2

8051 OVERVIEW

2.1 Introduction

The 8051 series of microcontrollers are highly integrated single chip microcomputers with an 8-bit CPU, memory, interrupt controller, timers, serial I/O and digital I/O on a single piece of silicon.

All members of the 8051 series of microcontrollers share a common architecture. They all have the same instruction set, addressing modes, addressing range and memory spaces. The primary differences between different 8051 based products are the amount of memory on chip, the amount and types of I/O and peripheral functions and the component's technology.

2.2 A Brief History of the 8051 Family

In 1981, Intel Corporation introduced an 8-bit microcontroller called the 8051. This microcontroller had 128 bytes of RAM, 4K bytes of on-chip ROM, two timers, one serial port, and four ports (each 8-bits wide) all on a single chip. At the time it was also referred to as a "system on chip". The 8051 is an 8-bit processor, meaning that the CPU can work on only 8-bits of data at a time. Data larger than 8 bits has to be broken into 8-bits pieces to be processed by the CPU. The 8051 has a total of four I/O ports, each 8 bit wide. See Figure 2.2. Although the 8051 can have a maximum of 64K bytes of on-chip ROM, many manufacturers have put only 4K bytes on the chip.

The 8051 became widely popular after Intel allowed other manufacturers to make and market any flavor of the 8051 they please with the condition that they remain code-compatible with the 8051. This has led to many versions of the 8051 with different speeds and amounts of on-chip ROM marketed by more than half a dozen although there are different flavors of the 8051 in terms of speed and amount of on-chip ROM, they are all compatible with the original 8051 as far as the instructions are concerned. The 8051 in the original member of the 8051 family, Intel refers to it as MCS-51.

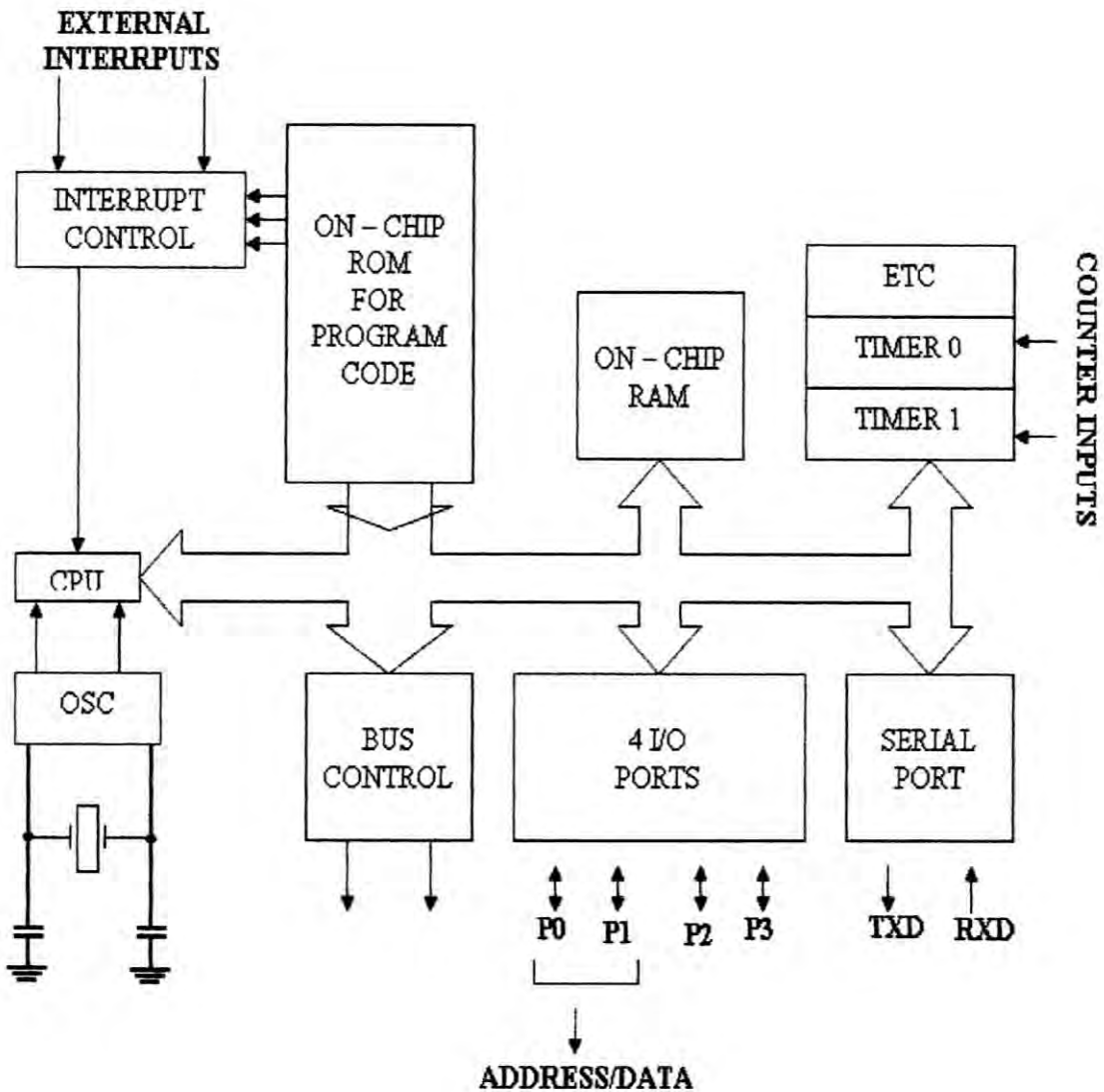


Figure 2.2: Inside the 8051 Microcontroller Block Diagram

2.3 8051 Architecture

The 8051 is an 8-bit machine. Its memory is organized in bytes and practically all its instructions deal with byte quantities. It uses an Accumulator as the primary register for instruction results. Other operands can be accessed using one of the four different addressing modes available: register implicit, direct, indirect or immediate. Operands reside in one of the five memory spaces of the 8051. The five memory spaces of the 8051 are: Program Memory, External Data Memory, Internal Data Memory, Special Function Registers and Bit Memory. The Program Memory space contains all the instructions, immediate data and constant tables and strings. It is principally addressed by the 16-bit Program Counter (PC), but it can also be accessed by a few instructions using the 16-bit Data Pointer (DPTR). The maximum size of the Program Memory space is 64K bytes. Several 8051 family members integrate on-chip some amount of either masked programmed ROM or EPROM as part of this memory space.

The External Data Memory space contains all the variables, buffers and data structures that can't fit on-chip. It is principally addressed by the 16-bit Data Pointer (DPTR), although the first two general purpose registers (R0, R1) of the currently selected register bank can access a 256-byte bank of External Data Memory. The maximum size of the External Data Memory space is 64Kbytes. External data memory can only be accessed using the indirect addressing mode with the DPTR, R0 or R1.

The Internal Data Memory space is functionally the most important data memory space. In it resides up to four banks of general purpose registers, the program stack, 128 bits of the 256-bit memory, and all the variables and data structures that are operated on directly by the program. The maximum size of the Internal Data Memory space is 256-bytes. However, different 8051 family members integrate different amounts of this memory space on chip. The register implicit, indirect and direct addressing modes can be used in different parts of the Internal Data Memory space. The Special Function Register space contains all the on-chip peripheral I/O registers as well as particular registers that need program access.

These registers include the Stack Pointer, the PSW and the Accumulator. The maximum number of Special Function Registers (SFRs) is 128, though the actual number on a particular 8051 family member depends on the number and type of peripheral functions integrated on-chip. The SFRs all have addresses greater than 127 and overlap the address space of the upper 128 bytes of the Internal Data Memory space. The two memory spaces are differentiated by addressing mode. The SFRs can only be accessed using the Direct addressing mode while the upper 128 bytes of the Internal Data Memory (if integrated on-chip) can only be accessed using the Indirect addressing mode.

The Bit Memory space is used for storing bit variables and flags. There are specific instructions in the 8051 that operate only in the Bit Memory space. The maximum size of the Bit Memory space is 256-bits. 128 of the bits overlap with 16-bytes of the Internal Data Memory space and 128 of the bits overlap with 16 Special Function Registers. Bits can only be accessed using the bit instructions and the direct addressing mode. The 8051 has a fairly complete set of arithmetic and logical instructions. It includes an 8X8 multiply and an 8/8 divides. The 8051 is particularly good at processing bits (sometimes called Boolean Processing). Using the Carry Flag in the PSW as a single bit accumulator, the 8051 can move and do logical operations between the Bit Memory space and the Carry Flag. Bits in the Bit Memory space can also be used as general purpose flags for the test bit and jump instructions. Except for the MOVE instruction, the 8051 instructions can only operate on either the Internal Data Memory space or the Special Function Registers. The MOVE instruction operates in all memory spaces, including the External Memory space and Program Memory space. Program control instructions include the usual unconditional calls and jumps as well as conditional relative jumps based on the Carry Flag; the Accumulator's zero state, and the state of any bit in the Bit Memory space. Also available is a Compare and Jump if Not Equal instruction and a Decrement Counter and Jump if Not Zero loop instruction.

2.4 8051 Cross Assembler Overview

The 8051 Cross Assembler takes an assembly language source file created with a text editor and translates it into a machine language object file. This translation process is done in two passes over the source file. During the first pass, the Cross Assembler builds a symbol table from the symbols and labels used in the source file. It's during the second pass that the Cross Assembler actually translates the source file into the machine language object file. It is also during the second pass that the listing is generated. The following is a discussion of the syntax required by the Cross Assembler to generate error free assemblies.

2.4.1 Symbols

Symbols are alphanumeric representations of numeric constants, addresses, macros, etc. The legal character set for symbols is the set of letters, both upper and lower case (A..Z,a..z), the set of decimal numbers (0..9) and the special characters, question mark (?) and underscore (_). To ensure that the Cross Assembler can distinguish between a symbol and a number, all symbols must start with either a letter or special character (? or _). The following are examples of legal symbols:

```
PI  
Serial_Port_Buffer  
LOC_4096
```

In using a symbol, the Cross Assembler converts all letters to upper case. As a result, the Cross Assembler makes no distinction between upper and lower case letters. For example, the following two symbols would be seen as the same symbol by the Cross Assembler:

Serial_Port_Buffer

SERIAL_PORT_BUFFER

Symbols can be defined only once. Symbols can be up to 255 characters in length, though only the first 32 are significant. Therefore, for symbols to be unique, they must have a unique character pattern within the first 32 characters. In the following example, the first two symbols would be seen by the Cross Assembler as duplicate symbols, while the third and fourth symbols are unique.

```
BEGINNING_ADDRESS_OF_CONSTANT_TABLE_1
BEGINNING_ADDRESS_OF_CONSTANT_TABLE_2
CONSTANT_TABLE_1_BEGINNING_ADDRESS
CONSTANT_TABLE_2_BEGINNING_ADDRESS
```

There are certain symbols that are reserved and can't be defined by the user. These reserved symbols are listed in Appendix C and include the assembler directives, the 8051 instruction mnemonics, implicit operand symbols, and the following assembly time operators that have alphanumeric symbols: EQ, NE, GT, GE, LT, LE, HIGH, LOW, MOD, SHR, SHL, NOT, AND, OR and XOR.

The reserved implicit operands include the symbols A, AB, C, DPTR, PC, R0, R1, R2, R3, R4, R5, R6, R7, AR0, AR1, AR2, AR3, AR4, AR5, AR6 and AR7. These symbols are used primarily as instruction operands. Except for AB, C, DPTR or PC, these symbols can also be used to define other symbols. The following are examples of illegal symbols with an explanation of why they are illegal:

1ST_VARIABLE	(Symbols can not start with a number.)
ALPHA#	(Illegal character "#" in symbol.)
MOV	(8051 instruction mnemonic)
LOW	(Assembly operator)
DATA	(Assembly directive)

2.4.2 Labels

Labels are special cases of symbols. Labels are used only before statements that have physical addresses associated with them. Examples of such statements are assembly language instructions, data storage directives (DB and DW), and data reservation directives (DS and DBIT). Labels must follow all the rules of symbol creation with the additional requirement that they be followed by a colon. The following are legal examples of label uses:

TABLE_OF_CONTROL_CONSTANTS:

DB 0,1,2,3,4,5 (Data storage)

MESSAGE:

DB 'HELP' (Data storage)

VARIABLES:

DS 10 (Data reservation)

BIT_VARIABLES:

DBIT 16 (Data reservation)

START:

MOV A, #23 (Assembly language instruction)

2.4.3 Assembler Directives

Assembler directives are used to define symbols, reserve memory space, store values in program memory and switch between different memory spaces. There are also directives that set the location counter for the active segment and identify the end of the source file.

Only one directive per line is allowed, however comments may be included. The following are examples of assembler directives: