Section #02

(Debug Commands)

This section is bound to discuss the following elements:

- Revision of hexadecimal and binary numbering systems.
- What is the "main memory" and its addresses?
- What are "CPU Registers"?
- Debug:
  - What is debug, advantages, and disadvantages.
  - Debug commands.
  - How to use debug to write, save, load, and show machine code for programs written in assembly language.

1) Hexadecimal and binary numbering systems:

- The hexadecimal (hex) numbering system is used in programmable controllers because a word of data consists of 16 data bits, or two 8-bit bytes.
- The hexadecimal system is a base 16 system, with A to F used to represent decimal numbers 10 to 15 (Table 2.1).
- The hexadecimal numbering system allows the status of many binary bits to be represented in a small space, such as on a computer screen.

Hexadecimal	Binary	Decimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
A	1010	10
B	1011	11
C	1100	12
D	1101	13
E	1110	14
F	1111	15

Table 2.1. Hexadecimal and binary numbering systems.

- Hexadecimal numbers can easily be converted to binary numbers. Conversion is accomplished by writing the 4-bit binary equivalent of the hex digit for each position, as illustrated in Figure 2.1.

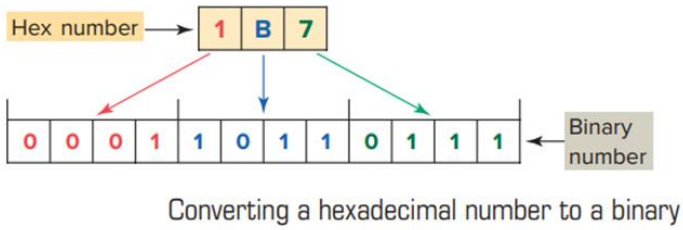


Figure 2.1 Converting a hexadecimal number to binary.

- Example of adding two binary numbers:

$$\begin{array}{rr} {\scriptstyle 0\,1\,1\,1} & \\ 00111 & 7 \\ 10101 & 21 \\ \hline 11100 & =28 \end{array}$$

- Example of adding two hexadecimal numbers:

$$\begin{array}{rcccccccc} & & & & {\scriptstyle 1} & {\scriptstyle 1} & & {\scriptstyle 1} & \\ & 9 & C & 3 & 7 & 2 & 8 & 6 & 5 \\ + & 1 & 3 & 9 & 5 & E & 8 & 4 & B \\ \hline & A & F & C & D & 1 & 0 & B & 0 \end{array}$$

2) What is the "main memory" and its addresses?

- The main memory in a computer is called Random Access Memory (RAM).
- This is the part of the computer that stores operating system software, software applications and other information for the central processing unit (CPU) to have fast and direct access when needed to perform tasks.
- It is called "random access" because the CPU can go directly to any section of main memory and does not have go about the process in a sequential order.

- Real memory addressing:
  - Memory is divided into segments as shown in Figure 2.2.
  - Each segment in memory is 64 KB.
  - Each segment number in debug is four hexadecimal numbers.
  - Each offset address in debug is four hexadecimal numbers.
  - Actual address = segment address + offset address as shown in Figure 2.3
  - The CPU deals with these segments through segment registers.

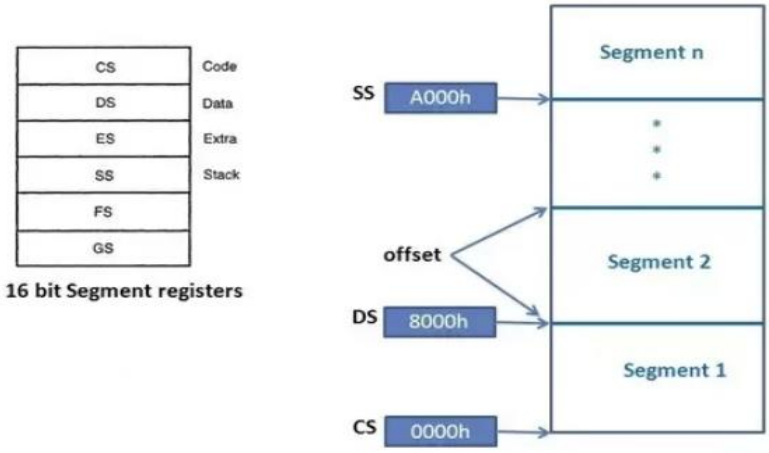


Figure 2.2 Real memory addressing.

Figure 2.3 Memory address as shown in debug.

3) What are "CPU Registers"?

- Register is one of a small set of data-holding places that are part of the computer processor.
- It is used to store information temporarily in the CPU, and it is a quickly accessible location available to a computer's processor.

- Registers usually consist of a small amount of fast storage (16-bits).
- As shown in Table 2.2, general-purpose registers can be accessed as 16-bit or 8-bit registers; all other registers can be accessed only as the full 16 bits.
- The data always fill the low 8-bits of a register then if still needed use some or all the high 8-bits.

Category	Bits	Register
General	16	AX, BX, CX, DX
	8	AH, AL, BH, BL, CH, CL, DH, DL
Pointer	16	SP (Stack Pointer), BP (Base Pointer).
Index	16	SI (Source Index), DL (Destination Index)
Segment	16	CS (Code segment), DS (Data Segment) SS (Stack Segment), ES (Extra Segment).
Instruction	16	IP (Instruction Pointer).
Flag	16	FR (Flag Register).

Table2.2 Processor Registers

4) What is debug, advantages, and disadvantages.

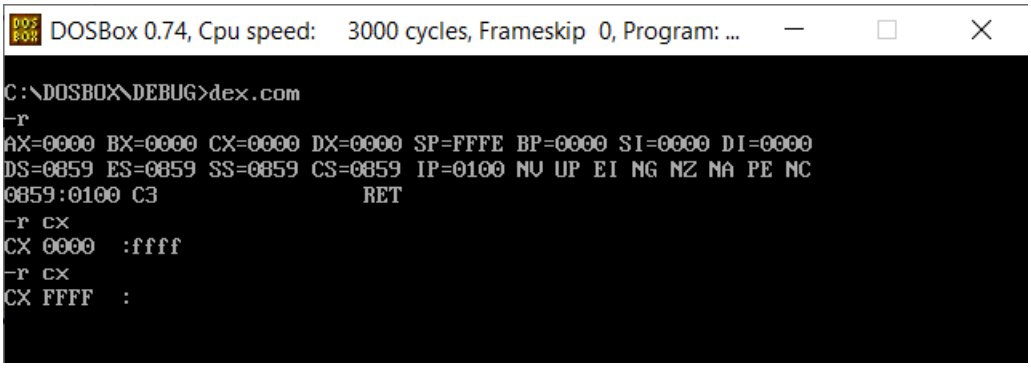
- Debug is a DOS command that works directly with memory and processor registers.
- used for programming in assembly language.
- Only files ending in ".com" can be created.
- The size cannot be larger than 64Kb.
- You can see all debug commands by typing? in DOS.
- All numbers are in hexadecimal.
- can use upper and lower case.
- Ctrl-c stops any command.

5) Debug Commands:

MS-DOS Debug Commands		
assemble	A	[address]
compare	C	range address
dump	D	[range]
enter	E	address [list]
fill	F	range list
go	G	[=address] [addresses]
hex	H	value1 value2
input	I	port
load	L	[address] [drive] [firstsector] [number]
move	M	range address
name	N	[pathname] [arglist]
output	O	port byte
proceed	P	[=address] [number]
quit	Q	
register	R	[register]
search	S	range list
trace	T	[=address] [number]
unassemble	U	[range]
write	W	[address] [drive] [firstsector] [number]

1. R (Register)

- The **R** command displays and modifies the register contents.
- ***Syntax is → R registerName***
- **R CX→** only displays the contents of the CX register, followed by:
- You can enter the hexadecimal number to change the content of the cx register.
- Pad the left with zero if the hex is less than four digits.
- You cannot insert a digit less than 0000 and larger than FFFF.

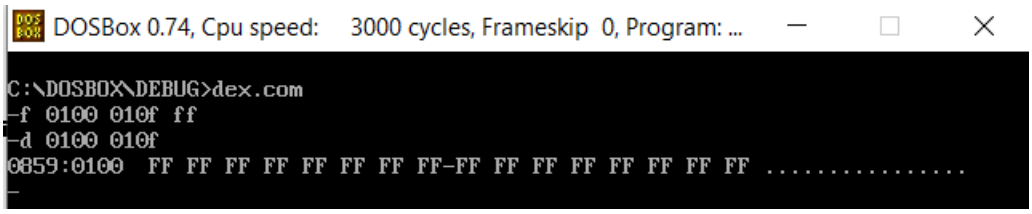


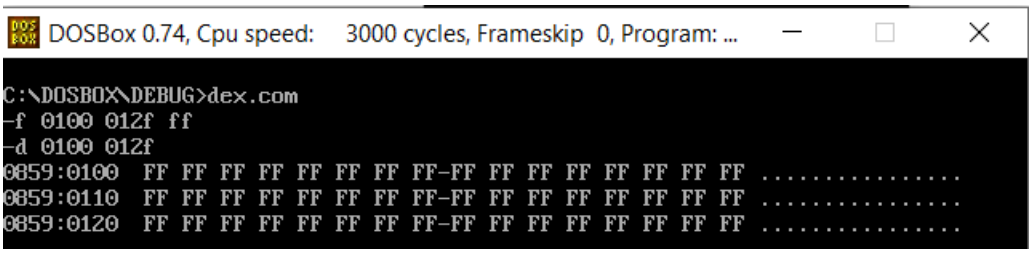
2. F (Fill)

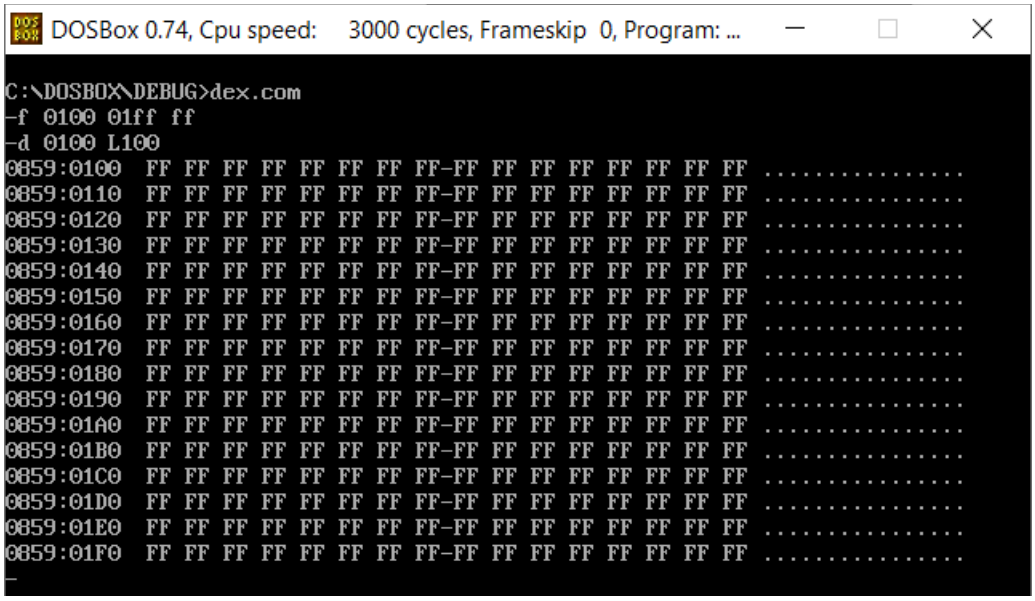
- Filling block of memory with data.
- ***Syntax is → F StartAddress EndAddress Data***
- F 0100 010f ff → filling 16 bytes of memory with ff
- F 0100 012f ff → filling 3 x 16 = 48 bytes of memory with ff
- F 0100 01ff ff → filling 16 x 16 =256 bytes of memory with ff

3. D (Dumb)

- Display the contents of an area of memory.
- ***Syntax is → D StartAddress EndAddress***
- ***Syntax is → D StartAddress***
- ***Syntax is → D StartAddress length***

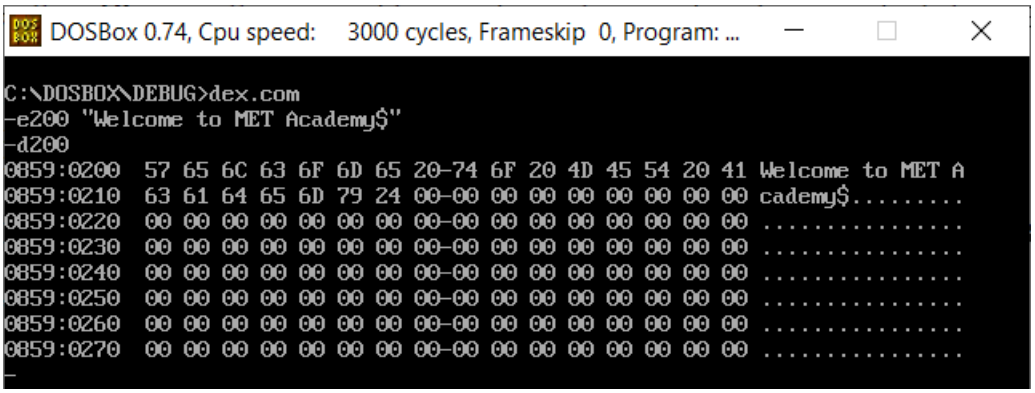




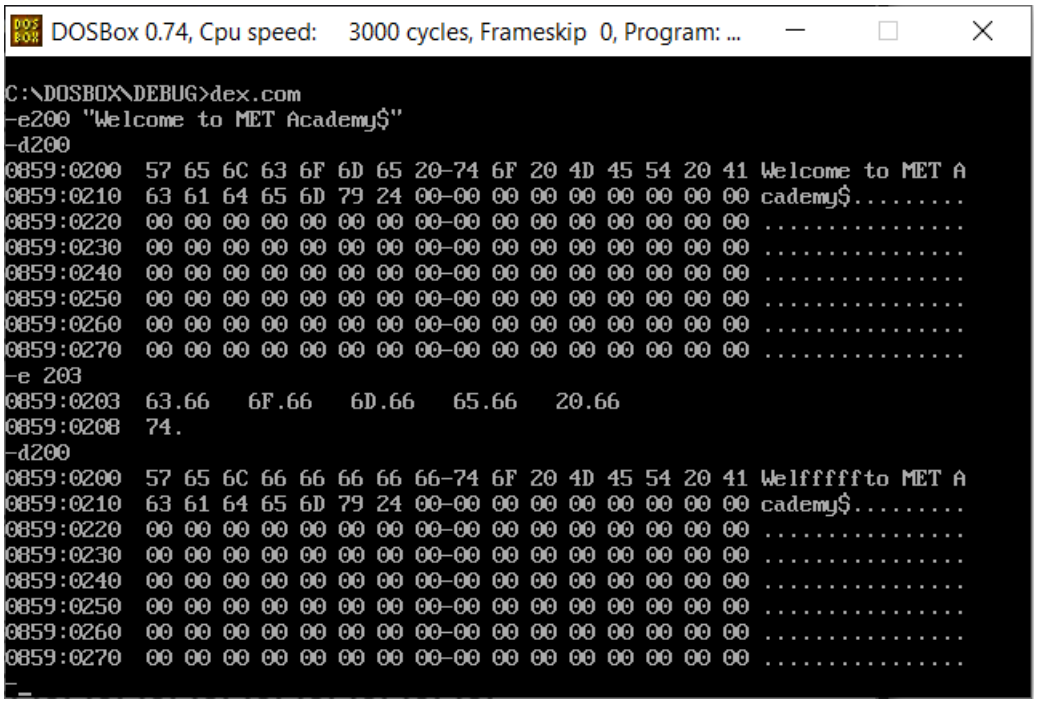


4. E (Enter)

- The E command is used to enter data directly into memory locations.
- Single(') or double(") quote marks are acceptable for entering ASCII data.
- ***Syntax is → EAddress "ASCII data$"***

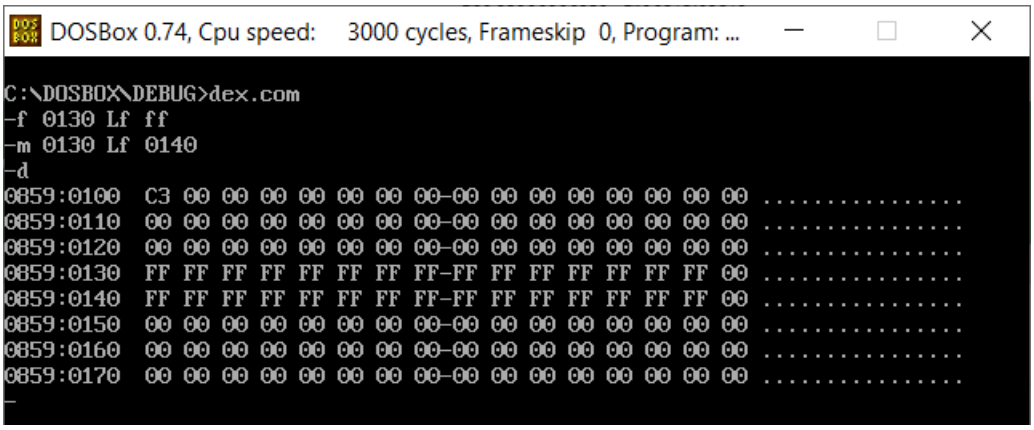


- You can alter data with E command.



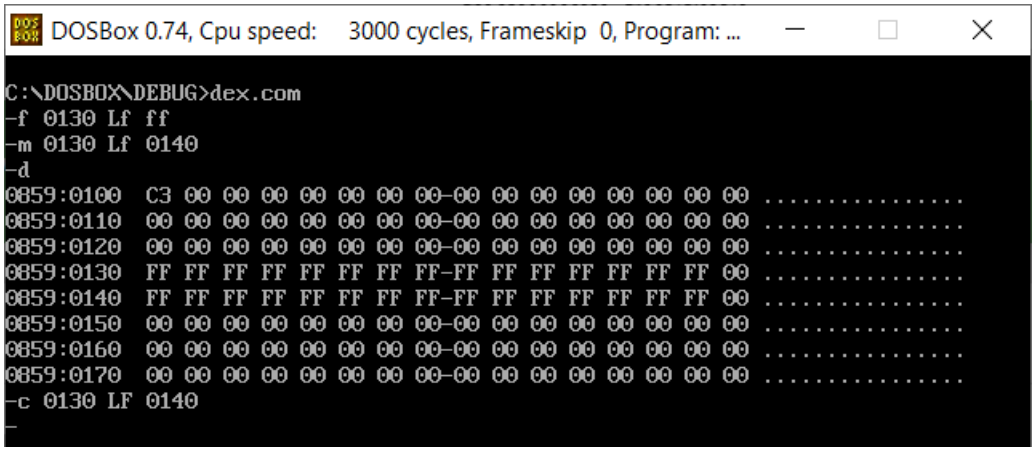
5. M (Move)

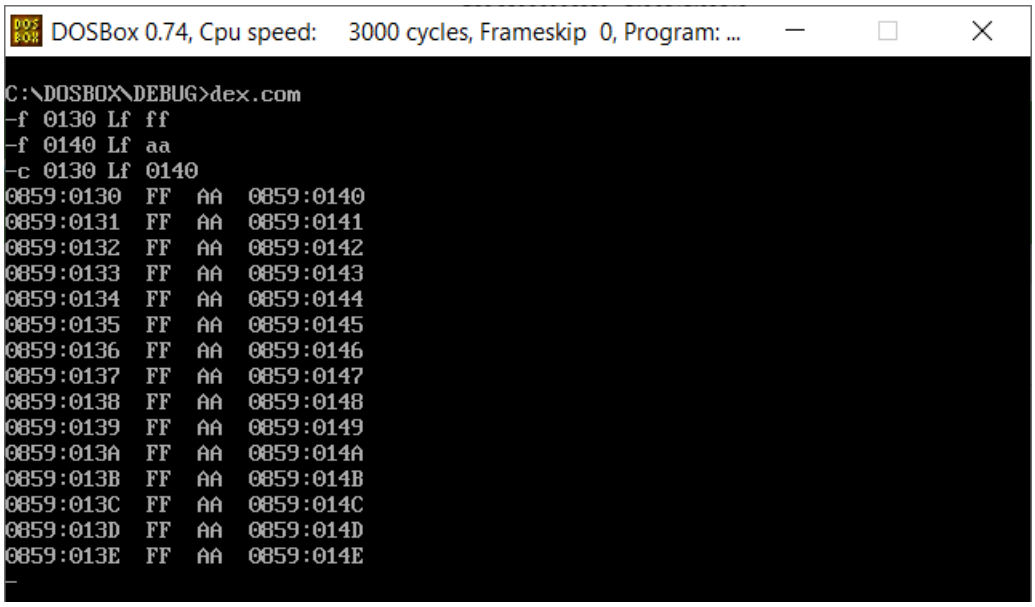
- The M command is used to move or copy data from one location to another.
- ***Syntax is → M [StartA EndA DestinationA].***
- M 0130 Lf 0140 → Copy data in address (0130 : 013f) to (0140:014f)



6. C Compare

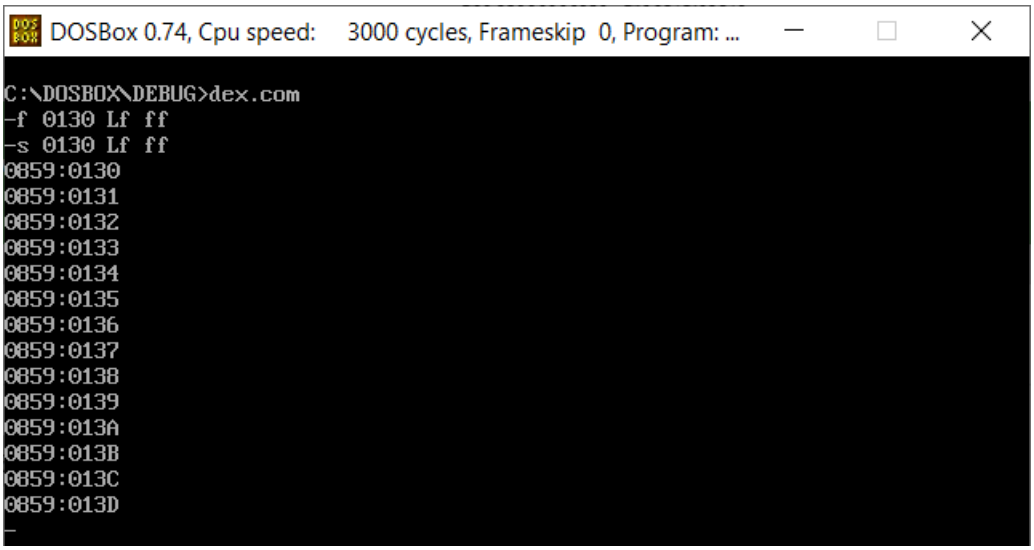
- The C command is used to check two areas of memory and display bytes that contain different data.
- If two area are identical, debug replay with –
- ***Syntax is → C [StartA EndA StartCompareA].***
- C 0130 L5 0140 → This return – (There is no difference).





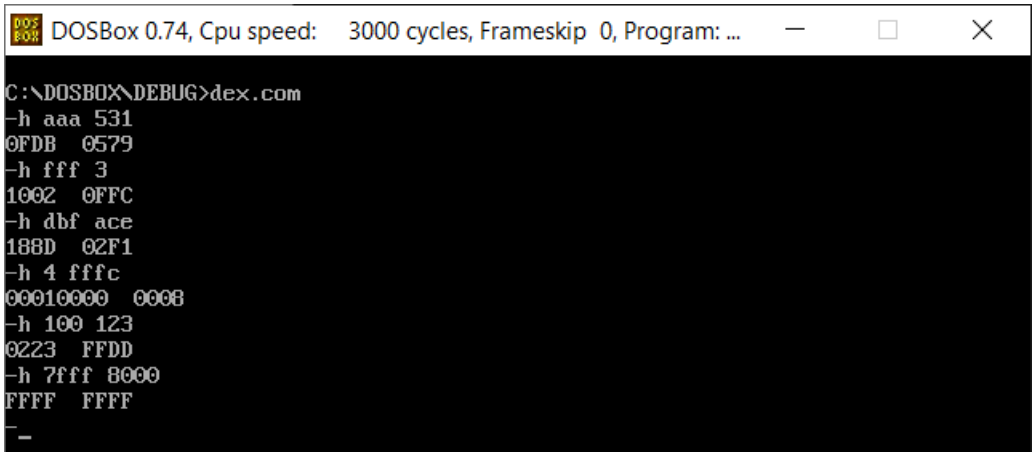
7. S (Search)

- The S command is used to search a block of data for a specific value.
- ***Syntax is → S[StartA EndA Value]***
- S 0130 Lf ff → returns with an address that contains ff value.



8. H (Hex)

- The H command is used to add and subtract two hexadecimal numbers.
- Show sum first, then difference.



6) How to use debug to write, save, load, and show machine code for programs written in assembly language. (Section No.3)

Scan the QR code to download debug64-bit V1.11.