



Automotive & Embedded Info

Never Forget Basics Whether its Life or Anything Else ... Basics are Cores. While seeing a Tree how can we forget Seed...

C, Embedded & Automotive Interview Questions

1. Why pre is faster than post?

2. Consider the two statements below and point out which one is preferred and why?

```
#define B struct A*
```

```
Typedef struct A* C
```

The typedef is preferred. Both statements declare pointer to struct A to something else and in one glance both looks fine. But there is one issue with the define statement. Consider a situation where we want to declare p1 and p2 as pointer to struct A. We can do this by:

```
C p1,p2;
```

But doing – B p1, p2, it will be expanded to struct A * p1, p2. It means that p1 is a pointer to struct A but p2 is a variable of struct A and not a pointer.

This is compiler –and –OS specific.

Typically a fixed size of **stack** is allocated for every process by the OS which is platform-specific. There's no limit on heap size, program usually have all of the available virtual address space.

4. Size of Function within stack?

Size of function = Size of all local variable which has declared in function + Size of those global variables which has used in function + Size of all its parameter+ Size of returned value if it is an address.

```
int x=2;

int* calculate

void main(){

int *p,a=0,b=5;

p=calculate(a,b);

printf("%d",*p);

}

int * calculate(int a,int b){

static int z;

z=a+x+b;

return &z;
```

```
}

```

Size of calculate function = 2 (global variable x) + 2 (local variable z) + 2*2 (parameter a, b) + 2 (returning address of z) = 10 bytes

5. Memory representation of char, int and float?

Unsigned int of 8 bit (0 to 255) → i = 0xFF;

Signed int of 8bit (-128 to +127) → first bit shows magnitude (0 for +ve and 1 for -ve) and remaining bits shows value. If i = 0xFF; 1 111111, then decimal value is → -1 and i = 0x80; 1 0000000, then decimal value is → -128

Float → first bit shows magnitude (0 for +ve and 1 for -ve), other 7 bits shows exponent and remaining 24 bits shows mantissa.

Ex: – Any Number Memory representation x.y = S E and M. To calculate number from memory representation = $S * M * 2^E$

6. How to find the controller is 8 bit or 16 bit or 32 bit using C program?

You can't do that using a C program (or any other language and even assembler).

First of all, note that 8 bit, 16 bit etc refers to the CPU *data bus*, which is not necessarily of the same width as the CPU *address bus*.

Although there are no guarantees about the default type sizes in the C standard, generally the size of int corresponds to the width of the data bus. Except this is not true for 8 bit processors.

Similarly, the size of a pointer corresponds to the size of the address bus, but the address bus is not necessarily of the same width as the data bus

As a rule of thumb (no standard exists):

Data bus	Addr bus	int	long	pointer
8	16	16	32	16
16	16	16	32	16
32	32	32	32	32
64	64	32	32/64	64

7. How the size of int is decided? Is it true that the size of int will depend on the processor? For 32-bit machine, it will be 32 bits and for 16-bit it's 16.

It depends on the compiler. For eg : Try an old turbo C compiler & it would give the size of 16 bits for an int cause the word size (The size the processor could address with least effort) at the time of writing the compiler was 16.

8. How to prevent same header file getting include multiple times?

```
#ifndef __HDRFILE__H__
```

```
#define __HDRFILE__H
```

```
#endif
```

9. How to call a function before main() function?

To call a function pragma startup directive should be used. Pragma startup can be used like this –

```
#pragma startup fun
```

```
void fun
```

```
{
```

```
....
```

```
}
```

```
main()
```

```
{
```

```
...
```

```
}
```

But this pragma directive is compiler dependent. Gcc does not support this. So, it will ignore the start-up directive and will produce no error. But the output in that case will be

10. Can a program be compiled without main() function?

Yes, it can be but cannot be executed, as the execution requires main() function definition.

11. Can a variable be both volatile & constant?

Yes the const modifier means that this code cannot change the value of variable but that does not mean that the value cannot change by means outside this code. Ex. Status Register.

12. Why global variables are evils?

1. Since every function can access to these so it becomes increasingly hard to figure out which function actually read and write to these variable.
2. Non locality issue
3. Con-currency issue
4. Name space pollution
5. Testing and confinement.

13. How to optimize loop Operation?

Loop variable should be local and it has a register storage class.

14. What will be the output of following code?

```
int i=0;Printf("%d %d %d",i,++i,i++); o/pà2 2 0.
```

```
int j=(1,2,3);o/pà3.
```

15. Stack Push and Pop Function?

<pre>#define MAX 5 int top, status; /*PUSH FUNCTION*/ void push (int stack[], int item){ if (top == (MAX-1)) status = 0; else { status = 1; ++top; stack[top] = item; } }</pre>	<pre>/*POP FUNCTION*/ int pop (int stack[]){ int ret; if (top == -1) { ret = 0; status = 0; } else { status = 1; ret = stack[top]; --top; } return ret; }</pre>
---	---

16. What is Modular Programming?

Dividing the program in to sub programs (modules/function) to achieve the given task is modular approach. More generic functions definition gives the ability to re-use the functions, such as built-in library functions.

17. String and Array related program?

18. What is Dijkstra's Algorithm?

The **#pragma pack** directive modifies the current alignment rule for only the members of structures whose declarations follow the directive. It does not affect the alignment of the structure directly, but by affecting the alignment of the members of the structure, it may affect the alignment of the overall structure according to the alignment rule.

The **#pragma pack** directive cannot increase the alignment of a member, but rather can decrease the alignment. For example, for a member with data type of integer (int), a **#pragma pack(2)** directive would cause that member to be packed in the structure on a 2-byte boundary, while a **#pragma pack(4)** directive would have no effect.

The **#pragma pack** directive is stack based. All pack values are pushed onto a stack as the source code is parsed. The value at the top of the current pragma pack stack is the value used to pack members of all subsequent structures within the scope of the current alignment rule.

In the code shown below, the structure s__t2 will have its members packed to 1-byte, but structure s__t1 will not be affected. This is because the declaration for s__t1 began before the pragma directive. However, s__t2 is affected because its declaration began after the pragma directive.

```
struct s__t1 {  
  
    char a;  
  
    int b;  
  
    #pragma pack(1)  
  
    struct s__t2 {  
  
        char x;
```

```

int y;

} S2;

char c;

int d;

} S1;

```

19. C small questions?

Escape Sequence	
%%	To print a string containing '%' in printf. <pre>printf("He got 90%% marks in math");</pre>
%n	<pre>Int c; printf("Hello%n world ", &c); printf("%d", c);</pre> O/P → Hello 0
Return value of printf	On successful return 1 or number of characters printed
Return value of scanf	No of input item.
printf(%d) output	Garbage value
malloc(sizeof(o))	Return either "a null pointer or a unique pointer that can be successfully passed to free()". But it should not dereference.
Override a macro	#ifdef A #undef A #endif #define A 10

C = a, b	C = a; comma has least precedence
Loop count L-H or H-L	Always use H-L
Seconds in a year	#define SECONDS_IN_A_YEAR(60UL*60UL*24UL*365UL)
BODMAS	Brackets, order power of, divide, multiply, addition and subtraction
#error	to throw an error message during compile time
Int x=2; int y=3; z=0;	z= x+++y; printf(“%d %d”, z, x++y); O\Pa 5 and 6
#pragma warn (+/-/.) xxx	+ ON, – OFF, . Toggle. Ex: – #pragma warn -rvl
Rename function	Using typedef

20.C small codes: –

Mul of 2 num without using * and loop	
<pre>int multiply(int x, int y) { if(y == 0) return 0; if(y > 0) return (x + multiply(x, y-1)); if(y < 0) return (-x – multiply(-x,</pre>	

```
}  
  
int main()  
  
{  
  
printf("\n %d", multiply(8, -6));  
  
getchar();  
  
return 0;  
  
}
```

21. Why do we prefer Active Low Signal?

The basic idea as is to keep a low impedance path to system ground on any selected device, ensuring that noise did not deselect it at a critical time.

Using HI levels to select means that if a power supply glitch happens, the CS is pulled low with the power supply.

22. Size of integer depends on?

It is depends on the primary compiler. If you using turbo c means the integer size is 2 bytes. else you are using the GNU gcc compiler means the integer size is 4 bytes. It is depends on only implementation in C compiler.

23.Can devices be added & removed while the system is running in I2C?

YES.

24. How many numbers of devices can be connected to I2C bus?

Maximum bus capacitance should not be exceeding capacitance of 400pf. According to the node address, it is 8 bit so the no of device connected to bus $2^8=128$.

25. What is the address in i2c bus?

Each device connected to the bus is software addressable by a unique address (hardware address provided by IC manufacturer).

26. Is it possible to have multiple master in i2c bus?

Yes

27. In write transaction, the master monitor the last ACK and issue stop condition? T/F

In read transaction slave sends data byte to master and master sends acknowledge bit to slave and in write transaction master sends data byte to slave and slave sends acknowledge bit (slave hold the SDA line low after receiving last bit of byte till master release the SCL line).

28. In read transaction, the master does not ACK last byte it receive and issue stop condition? T/F

Yes master acknowledge the last byte then issue stop condition.

29. How will the master indicate it is either address/data? How it will intimate to the slave that master is going to either read/write?

First every i2c device has its register for read and write and each register has its address. to read/write data master has to transmit that register address from where master wants to read/write. Second to read and write master sends 1 or 0 respectively with slave address to the slave on i2c bus.

30. Number of devices can be connected to I2C bus.

31. Is it possible to have multiple master in SPI bus?

As a protocol it does not support but if we want then we can make it as per our application requirement.

32. How will master convey that it is stopping transmission of data?

The master stop toggling its clock and then by deselected SS pin of slave.

33. How to select slave in SPI bus?

Via SS pin.

34. Can devices be added & removed while the system is running (hot swapping) in SPI?

Yes.

35. Is it possible to connect slave in daisy chain? What is the role of shift register in master and 7? Slave in SPI?

Role of shift register in SPI to transfer of data bit by bit from master to slave with clock.

36. Duplex communication is possible by simultaneously using MOSI & MISO during each SPI clock cycle? T/f –

Yes.

37. What are the function of can transceiver?

The transceiver provides differential transmit capability to the bus and differential receive capability to the can controller. Transceiver provides an advanced interface between the protocol controller and the physical bus in a

controller area network (can) node.

Typically, each node in a can system must have a device to convert the digital signals generated by a can controller to signals suitable for transmission over the bus cabling (differential output). It also provides a buffer between the can controller and the high-voltage spikes that can be generated on the can bus by outside sources (emi, esd, electrical transients, etc.).

The can transceiver is a device which detects the signal levels that are used on the can bus to the logical signal levels recognized by a microcontroller.

38. Can you have two transmitters using the same exact header field?

No – that would produce a bus conflict. Unless you have middleware that ensures only one node can transmit at a time.

39. Why can is having 120 resistor at each end?

To minimize the reflection reference, to reduce noise. To ensure that reflection does not cause communication failure the transmission line must be terminated.

40. Why can is message oriented protocol? Can logic what it follows?

In can protocol arbitration is done using identifier field of the messages send by nodes on can bus. Wired and logic.

41. What happen when two can node is sending same identifier at the same time?

Two nodes on the network are not allowed to send messages with the same id. If two nodes try to send a message with the same id at the same time arbitration will not work. Instead, one of the transmitting nodes will detect that his message is distorted outside of the arbitration field. The nodes will then use the error handling of can, which in this case ultimately will lead to one of the transmitting node being switched off (bus-off mode).

42. What is the speed of CAN? 1 mbit/sec upto 1 mtr and 50kbit/sec upto 1600 mtr. If cable length increases there will be decrease in speed due to cable RLC on the cable.

43. What is the use of SRR (substitute remote request) bit? In CAN extended frame this bit comes in picture. This bit always keep recessive in order to allow a standard frame higher priority than the extended frame.

In standard frame after 11 bit identifier RTR bit comes that is always dominant except in remote frame but in extended frame after 11 bit identifier SRR bit comes that is always recessive so that standard frame message has always priority than extended frame message. Can bus works on and wired logic and dominant one will always wins bus access.

44. What is difference between standard CAN and extended CAN?

Number of identifiers can be accommodated for standard frame are 2^{11} . Number of identifiers more compare to base frame, for extended frame are 2^{29} . Ide bit – 1 for extended frame.

Ide bit – 0 for standard frame.

45. What is the difference between rs-485 and can?

Rs-485 is not a protocol. Its only provides basic rule & physical link or media for communication.

Communication frame structure, the scheme for addressing other nodes, the mechanisms to avoid data collisions and other tasks have to be implement by the developer in the form of a protocol software.

Can not only provides the physical media for the communication, it also provides all other mechanisms necessary for addressing data packets (messages), avoiding data collisions, detecting failures in the transmitted data, automatic repetition of disturbed messages and ensuring data consistency over all nodes in a network. Furthermore, can specifies

	the structure of the data frame, with message identifier, data and control byte.
In rs-485 communication is carried out on the basis of by individual character which are written or read via an 8-bit register. The structure telegram must be implemented in software.	Can functions with a preset telegram format in addition to 0-8 data byte, also contain the control information for addressing and data consistency? The user has to transmit message identifier and the payload and rest is added by a so called can controller.
Supported bus arbitration principles- master/slave or token ring.	Due to message wise arbitration, can enables multi master operation which can be derived from that like master/salve or token ring.
Detection of data collision – Not implemented	Detection of data collision- CSMA/CD. Carrier sense multiple access / collision resolution.
No error management.	Can has very advanced error management.
Data limit- transmission of bulk of data.	Data- limited 8 byte at a time.
Bus length- 1200 mtr at 100kbit/sec.	Bus length- 1600 mtr at 50 kb/sec
Transmission rate- 10 mbit/sec up to 12 mtr.	Transmission rate- 1mbit/sec up to 1 mtr.
Bus transceiver cost- 70 euro	Bus transceiver cost- 35 euro

46. In SPI and I2C which one is when to use?

I2C	SPI
When no of i/o are less.	When no of i/o are not the issue.
Only half duplex is necessary.	Full duplex is necessary
High speed communication is not needed.	High speed communication is necessary.
Can work as a Multi master.	Only one master..

Advertisements



REPORT THIS AD



REPORT THIS AD

Share this:



Twitter



Facebook

Like

Be the first to like this.

Automotive & Embedded Info / Powered by WordPress.com.