Global edge telephonics:
1)mutex and semaphores difference
2)message queues and shared memory difference
3)what is the use of volatile keyword
4)how we will set particular bit in 32-bit address
5)explain null driver for character driver
6)why major number is required in driver
7)why device directory is required

l&t telephonic:
1) what is difference between linux kernel and embedde linux or android kernel
2) static variable
3) difference b/n calloc and malloc
4) if int *ptr its address is 2000 then increment ptr what is address?
5)how we find the loop in linked list?
6)what is difference between 32-bit and 64 bit?

Robort bosch:
1)static variable,volatile variable,?
2)what microcontroller used in pandaboard?
3)how the data is read/write done in register ?
4)write a block diagram from device to processor?
5)write a program for adding anode in single linked list?
6)write a program for strcopy using pointer program?
7)strngconcatination program?
8)how interaction is possible from controller to slave driver?
9)reverse linked list program?
10)write a program for reverse string program?
11)how to port android in panda board?
12)what is memory frame size in you project?
13)difference b/n const ptr and pointer to constant?
14)suppose str1=”rajamahender” and str2=”mah” then how searching is done whether str2 are in str1 or not?

Qualcomm:
1)without using sizeof operator how find size of integer?
2)explain project brefily?
3)Diff b/w mutex, semaphore ?
4)diff b/w tophalf & bottom half ?
5) wat is the exact use of bottomhalf ?
6)kernel interrupt handling ?
7)function pointer usage ?
8) i have a(), b(), c() functions..... so i called a() function in b() function, and i called b() function in c() function... print the stack pointer value of c() function ?
9) int i= 320;
char c;
c= (char)i;
printf("%d",c);
any errors ??
10)size of int without using sizeof() operator ?

Introduction and project details:

----

1) what is your work in your company?

A: Embedded system software development and testing

2) what is your current project?(i said broad bring up on omap)

A: Board bring up and poting on OMAP

3) what you had done in that project

A: GPIO validation. In uboot as well as kernel also

4) explain boot sequence of omap?

5) Explain i2c flow?

6) you kept development of UART driver? what u had developed its already there in your kernel source tree?

7) What kind of implementations that your company doing?

A: My company focusing on embedded technology and digital video solutions
porting open source on target platform, driver development and configuration
development of kernel based and user based software with focus on embedded linux

c questions:

----

first they will ask you how do you rate in c language out of 10

1) what are the compilation steps? explain

2) what is machine dependent and independent code? example

Ans:

The kernel is divided into two major parts: the machine-independent code and the machine-dependent code. The former is shared among the ports, is entirely in C, and contains the algorithms and data structures that do not depend on the architecture on which the kernel is running. The latter contains the routines and modules that are called by the machine-independent code. The assembly portion of the kernel is kept in one file, locore.s. The file pmap.c interfaces the kernel's machine-independent virtual memory code with the machine's memory mapping facilities. Finally, the miscellaneous machine-dependent code is appropriately placed in machdep.c.

3) why we use volatile?

Ans:

The volatile keyword is intended to prevent the compiler from applying any optimizations on objects that can change in ways that cannot be determined by the compiler.

example: Global variables modified by an interrupt service routine

The code reading data port must be declared as volatile in order to fetch latest data available at the port. Failing to declare variable as volatile, the compiler will optimize the code in such a way that it will read the port only once and keeps using the same value in a temporary register to speed up the program (speed optimization).

4) what is Static variable?
Ans:

static variables are those variables whose life time remains equal to the life time of the program.

There are 3 main uses for the static.

1. If you declare within a function:
It retains the value between function calls

2.If it is declared for a function name:
By default function is extern..so it will be visible from other files if the function declaration is as static..it is invisible for the outer files

3. Static for global variables:
By default we can use the global variables from outside files If it is static global..that variable is limited to with in the file.

5) write a program to print size of integer variable without using sizeof operator (implement using bitwise and pointers)?

6) write a program to find repeated character in a given string? code should be generic?

Ans:

```
#include<stdio.h>

int array[100];

main()
{
    int i, j, count = 0;
    char str[] = "aabbccddeeffgg";

    for( i = 0; str[i] != '\0'; i++)
    {
        if( array[i] == 0 )
        {
            array[i] = 1;
            count = 1;
            for( j = i + 1; str[j] != '\0'; j++)
            {
                if( str[i] == str[j] )
                {
                    count++;
                    array[j] = 1;
                }
            }
```

```
                        if(count > 1)
                                printf("%c", str[i]);
                }
        }
}
```

7) Implement string reverse program without using temporary buffer and strrev function?

Ans:

```
        #include<stdio.h>
        main()
        {
                char str[] = "ravindra";
                int i, len;
                len = strlen(str) -1;
                for( i=0; i < strlen(str)/2 ;i++)
                {
                        str[i] += str[len-i];
                        str[len - i] = str[i] - str[len -i];
                        str[i] = str[i] - str[len-i];
                }
                printf("%s\n",str);
        }
```

8) how do you transfer data from userspace to kernel space?

9) What is generic system call that is used to transfer data from user space to kernel space?

10) write program to find out middle node in circular linked list with minimum length of code?

11) How to know functions stack pointer address?

12) Implement bubble sort, quick sort, selection sort, merge sort?

13) write a Make file to compile 3 files and generate a executable with name hello
a
14) write a program to copy a string from source to destination without using strcpy?code should be optimised to only three lines?
ans: My answer is following

```
main()
{
        char str[]="aravind",dstr[100];
        char *sptr=str,*dptr=dstr;
        while(*dptr++ = *sptr++);
        printf("after copying is %s\n",dstr);
}
```

Linux internal
----

1) what is difference between mutex and semaphores

2) why we are using mutex and semaphore any way we are having conditional variables?

3) what is scheduler?

4) what are types of schedulers are there?

interviewer feedback:

why you are looking for a change?
1.how the device can interrupt the CPU
2.what we have to do In ISR
3.wht is the difference b/wSemaphore,Binary semphore,spin locks.
4.decalre constant Pointer and pointer to constant
5. why dont use Mutex in ISR s what are the reasons
6.Explain about I2C,SPI Protocals
7.can slave intaite the data transmision
8.can SPI is having Multi master or single master
9.static int a=30 in one file and static int a=30 in another file ,will u get an error or not
10.main concepts of Interrupts in lInux kernel

1. write a user defined function to strcpy()
2. what are the causes of stack crash
3. what is the diff b/w spinlock & mutex
4. explain about static int a, int a, const a, volatile a;
5. what is bitbanging

How to find out middle element of a single lnkedlist

2> How to findout sizeof(int) with out using sizeof function

3> implement sample char driver code

4> how to findout hw many 1's in a given integer with bitwise operators

Explain sample code in Project.

1. size of operator implement with own(bitwise operators)
2. strcpy (string)
3. sreekant how many repeated charescters are their
4. union

```
{
        int i=325;
        char ch=25;
}
sizof(union);
```

5.

```
union
{
        int i=325;
        char ch=(int *)i;
}
```

what is the ch value

6.

```
function1()
{
        function2 ()
        {
                 function3 ()
                 {
                 }
    }
}
```

what is the stack pointer

7. what is function pointer where we use it;

8. Macro where to use

9. compilation and build process

10. static and dynamic libraries

11. static and dynamic lybrares advantages and disadvantages

12. volatile where to use

i2C SLAVE ADDRESS 0x54 ; read – Write -

1> How to find out middle element of a single lnkedlist

2> How to findout sizeof(int) with out using sizeof function

3> implement sample char driver code

4> how to findout hw many 1's in a given integer with bitwise operatorsintf("\n");

1. count num of 1's & 0's in a num
2. can we kill a thread from another thread
3. if one process is in sleep state, another process is sending a signal to the sleeping process, how the sleeping will process the signal
4.delete a middle node in a SLL
5.what is the error

```
#include<stdio.h>
```

```
struct abc;
void main()
{
abc a;
return void;
}
```

6. what is the use of u-boot & wat is the protocol used to load os image by using uboot
7. how many signals are there in linux, wat is the size of signal queue
8. size of empty sructure

1. Explain your projects detail?
2. How do get major number?
3. How IOCTL Works?
4. How malloc works? Explain?
5. What Fragmentation?
6. Memory barriers?
7. what is difference between user-space threads and kernel space threads?
8. what is difference between process and thread?
9. why do we require interrupt?
10. Explain how interrupt works?
11. Does interrupts disabled by default during execution of interrupt handler?
12. what we should not do in interrupt handler?
13. what is difference between top halves and bottom halves?
14. Difference between work queues and tasklets?
15. In which stage work queues are created?
16. what is difference between kmalloc and vmalloc?
17. what is difference between physically continous and virtually continous
18. In RAM i'm having 2Mb memory space but from driver i'm allocating 4mb size with
19. GFP_KERNEL and GFP_ATOMIC which call fails? Explain?
20. What is DMA? How it works?
21. How do you initiate DMA transfer?
22. On which addresses DMA works?
23. If any contents are of userspace data are cached what you should do in your DMA transfer, if you want transfer that data to device?
24. What is difference between spinlock and mutex?
25. In Uniprocessor system, if two processes accessing critical section how do you implement synchronization?
26. In Uniprocessor system, if a processes accessing critical section in between a interrupt arises the interrupt handler also accessing critical section how do you avoid this race condition?
27. What is the worst case to add a node to Binary search Tree?
28. What is the Depth of Binary search Tree if it has 1000 nodes?
29. What is the worst case for bubble sort?

C questions:

1. Take a sentense and insert a one string into sentense
2. check even bit positions in number, if 0 set to '1'.
3.Storage classes and their scope,default values
4.count set bits of a number
5. Reverse a single linked list
6. Implement a binary search tree
7.Insert a node in Double linked list
8.what is a volatile and where we use
9.write a program to reverse a string without using strrev.
10. Difference between union and structure
11.write a program to merge sort
12.Reverse a double lined list

OS concepts :

13. Tell me about scheduling algorithms and explain
14. Diff b/w mutex and semaphore
15. what is bottom half and what we can do in that
16. where we use spinlocks, why
17. Charcter driver template
18. what is Fragmentation
19. what is RTOS,where we use
20. Difference between shared memory and message queus
21. How to read data using I2C(programming)
22. which is faster I2C or SPI
23. what is u-boot
and mostly on Project related questions,,,,,,,,,

Must and should have to prepare project well ....

Telephonic:

1. If you are having some files a.c,b.c,c.c etc, and you are having a variable int n; how do you declare the integer variable, to access the integer variable in all files ?
2. On 32bit machine if you are having int *ptr, char *ch, then what would be size of ptr and ch ?
3.
```
int a[10];
int *ptr = NULL;
then the statements    1) ptr=a;
                       2) a=ptr;
are valid or not ?
```
4. What is meant by Modulation And what is the need of modulation ?
5. In an integer int n; how can you count the number of bits that are set ?
6. An integer int n; is of 2bytes and having a value 0x1122, then how can you make it to 0x2211 ?
7. Explain the Android boot flow & Architecture ?
8. How do u switch from user level to kernel level ?
9. What are runlevels in Linux ? And what is the difference in between them ? And how do you switch in between runlevels ? And whar is the normal user runlevel ?
10. what are the debuggers you are familier with ? Are you comfortable with GIT ,ADB tools and also with ARM architecture ?
11. what are the different types IPC mechanism's are there ?
12. What is the RAM size of the board Board you handled ?

F2f:

1. How can you calculate the Angle Alpha shown in the pentagon ? Hints : Sum of all the Angles in a triangle are equal to 180 Degrees, all the sides of pentagaon are equal. The Centre is in equi distance from all the edges.

2. how to delete a given N th node in a single linked list ?
3. Draw the memory layout of a process adress space ?
4. If you have an expression like 10+20-30/40*50. How this expression will store in stack how this expression is given to the cpu from stack ?
5. If you have an integer of 16bit int n; with value 0x1122, then write a preprocessor command (Macro) for exchanging the bytes to get the number as 0x2211 ?
6. How can you view a process Stack and Heap ?

Aravind

1. Explain your projects detail?
2. How do get major number?
3. How IOCTL Works?
4. How malloc works? Explain?
5. What Fragmentation?
6. Memory barriers?
7. what is difference between user-space threads and kernel space threads?
8. what is difference between process and thread?
9. why do we require interrupt?
10. Explain how interrupt works?
11. Does interrupts disabled by default during execution of interrupt handler?
12. what we should not do in interrupt handler?
13. what is difference between top halves and bottom halves?
14. Difference between work queues and tasklets?
15. In which stage work queues are created?
16. what is difference between kmalloc and vmalloc?
17. what is difference between physically continous and virtually continous
18. In RAM i’m having 2Mb memory space but from driver i'm allocating 4mb size with
19. GFP_KERNEL and GFP_ATOMIC which call fails? Explain?
20. What is DMA? How it works?
21. How do you initiate DMA transfer?
22. On which addresses DMA works?
23. If any contents are of userspace data are cached what you should do in your DMA transfer, if you want transfer that data to device?
24. What is difference between spinlock and mutex?
25. In Uniprocessor system, if two processes accessing critical section how do you implement synchronization?
26. In Uniprocessor system, if a processes accessing critical section in between a interrupt arises the interrupt handler also accessing critical section how do you avoid this race condition?
27. What is the worst case to add a node to Binary search Tree?
28. What is the Depth of Binary search Tree if it has 1000 nodes?
29. What is the worst case for bubble sort?

Explain Display framework in android system?

How to works copy_to_user and copy_from_user function?

How to mapping frame buffer? gralloc

How to control interrupt context in linux?

Why spin locks implemented in Uniprocessor system (msm8625)?

What is platform register?

What is the probe datastructure in linux?

how to map volatile variable using cache memory?

What is the diff b/w kmalloc and vmalloc?

what are the flags in kmalloc?

1. How to identifuy boot loader kernel image.
2. Explain character driver.
3. file formats ; jffs- run time changes support; ext - doesn't support
4. kernel crash
5. Log Analysis in Android
6. Wait queue in Linux Kernel
7. never goes in sleep mode.
8. spin locks

Date : 26/April/2013

1. Tell me about yourself ?
2. what is embedded system ? Give a block diagram ?
3. what is the architecture of your processor (based on your block diagram) ?
4. what is Bus hang ?
5. what is board bringup ?
6. what are different types of RAM's and what is the main difference between each ?
7. what are the Operating systems functionalities after the board has finally booted ?

a. process management

b. Memory management

c. scheduling

d. MMU

e. process state diagram

8. what is semaphore and mutex ?
9. what is binary semaphore and counting semaphore ?
10. what is FUTEX?
11. what is priority inversion ?
12. what is priority inheritance?
13. How do you debug a source code (In general) ?
14. Do you have any idea on TRACE32 ?

15. what are storage classes, what variables are stored in it and what are its scope ?
16. what are register variables and what is its use ?
17. what are different data types in C ?
18. what are different segments in a program ?
19. what is Macro ? what is its use ?
20. what is preprocessor?
21. what are different preprocessor directives ?
22. what is difference between Macro and functions ?
23. How to delete a particular node from a single linked list ?
24. How to merge two single linked lists ?

Date :26/April/2013

1. Programm to add node in double link list
2. heap memory allocated and got pointer, how to know the allocated size.
3. other that local variable what are will go to stack
4. How to analyze stack overflow.
5. How to analyze data corruption in heap memory due to external module.
6. Difference between Binary semaphores and Mutex
7. IPC mechanizms
8. memory layout of a programm
9. Memory Management Unit
10. Trace32

Date : 26/April/2013

1. Where you done your B. Tech? What is your company name?
2. Master and multi slave communication?

a.What happens If slave stuck while receiving the data?

b.What happens if master got stuck?

3. What is importance of clock?
4. How to resolve when both Master and slave in system(ie watchdog timer)?
5. What is Little endian and Big endian?
6. Write program to convert value from Little endian to Big endian and vice versa?

And also (from:0x12345678 to 0x78563412)

7. What is GPIO? Purpose and applications?
8. What is your role in your project?
9. What is volatile ? And its storage location ?
10. Memory layout of c program?
11. Explain boot sequence ?
12. How does a processor understand the breakpoint condition while debugging ?
13. Different assignments of breakpoints ?
14. How to debug ARM code ? (ie trace32)

15. Difference between malloc() and calloc() ?

16. Explain about Producer – Consumer problem Using semaphore?

Date : 29/April/2013 Interviewer : Kishore Thirupathi

1. What is an O.S. ? Tell me briefly about the functions of O.S.

2. Explain different scheduling algorithms.

3. In Round Robin algorithm,suppose every process is given a time slot of 10ms and if a process is done with its job before the allocated time slot,what will be the real time scenario? Will the process wait till time slot is completed or the CPU time will be allocated to a diffrent process?

4. In Shortest Job First algorithm,how will the kernel be able to know the time required by a process for its execution?

5. Explain IPC mechanisms? Why are they needed in real time systems?

6. In Shared memory IPC technique,what will happen if a process having access to share memory terminate abruptly?

7. In Socket Communication, listen call tells the number of clients a server can process,say for eg. 5. What will happen if a 6th client comes into picture

8. Will you be able to establish two way communication between processes using pipes?

9. C program to reverse a singly linked list.

10.What are all the logs analyzed till date,how far you are confortable and will rate yourself into it

Overall,the interviewer was more focussing on real time O.S. scenarios.

C,OS and Data strucutre questions

1 .What are static variables?
2. What are volatile variables?
3. What do you mean by const keyword ?
4. What is interrupt latency?
5. How you can optimize it?
What does the keyword const mean?
What do the following declarations mean?
const int a;
int const a;
const int *a;
int * const a;
int const * a const;
What does the keyword volatile mean? Give three different examples of its use.
Can volatile and const be used together?

Write a program for power of two?
Explain bit field,structure padding and masking
how to set and clear a particular bit
difference between structure and union
Explain Enum?
Storage classes in C?

Size of int pointer char pointer

Device drivers and Linux internals Questions

Major and Minor number
Main aspects of linux OS
Types of scheduler used in Linux
what is RTOS
Priority inheritance and Priority inversion in RTOS
Memory allocatn in kernel(Kmalloc, Vmalloc …. syntax and uses)
Driver registration process,what r the main structures and function used iin driver registration
What is nmap and mmap,how they r used in Linux drivers...
Interrupts,Top halves and bottom halves
Interrupt Service routine its basics like return type functn declaration and memory allocation
difference between malloc and calloc
What r system calls and how they are implemented
What is GPIO
Context switching
Tasklets workqueue and their differences
Memory management in linux
Difference between physical and virtual memory....\

Questions on protocol drivers (I2C,SPI)

Working process of I2C
How many I2c clients ca\n be connected to a bus
Addressing modes in I2C
Bus arbitration and bus streching in I2C
Clock rates in I2c

http://vijaynetwork.com/embedded-questions-answers.html

Embedded C Questions And Answers

Write a macro for set and reset, swap.

```
#define SET_BIT( _X_, _NO_ ) ( 1<<(_X_-1)) | _NO_
#define RESET_BIT( _X_, _NO_ ) ~( ( 1<<(_X_-1) ) ) & _NO_
```

```
#define SWAP_BIT( _X_, _NO_ ) ( 1<<(_X_-1)) ^ _NO_
```

how to find give no is 2 power of n?

```
if ( ( no & ( no-1 ) ) == 0 )
printf( "Given number is 2 Power of N\n" );
else
printf( "Given number is not 2 Power of N\n" );
```

Swape two numbers without using third variable.

```
x^= y^= x^= y;
```

volatile int* ptr;
int* volatile ptr;
what is the difference?

volatile int* ptr; - Pointer is volatile.
int* volatile ptr; - Actual address stored in the pointer is volatile.

Assume that 0x7600 is an address. How will you store 50 in that address?

```
*( ( int * ) 0x7600 ) = 50;
```

Difference between memcpy and memmove.

memmove offers guaranteed behavior if the source and destination arguments overlap. memcpy makes no such guarantee, and may therefore be more efficiently implementable. When in doubt, it's safer to use memmove.

How to find the given number is little endian or big endian?

```
#include <stdio.h>
int main()
{
  unsigned int n = 1;
  char *p;
  p = (char*)&n;
  if (*p == 1)
  printf("Little Endian\n");
  else if (*(p + sizeof(int) - 1) == 1)
  printf("Big Endian\n");
  else
  printf("Surprise output!!!!\n");
  return 0;
}
```

Reverse a single linked list

```
Node *Reverse (Node *p)
{
 Node *pr = NULL;
 while (p != NULL)
 {
 Node *tmp = p->next;
 p->next = pr;
 pr = p;
 p = tmp;
 }
 return pr;
}
```

Finding Loop in a single linked list.

(1) If the linked list is read only, take two pointer approach(p1, p2). Both pointing to beginning of linked list. Now increment p1 by 1 and p2 by 2 and compare both. if they are equal there is a cycle. Repeat this untill p2 points to null.
(2) If you have the condition not to modify the node but you can change the links, then reverse the linked list. If you reach the head node then there is a cycle.

Finding middle of the single linked list in a single traversl.

Step 1:
Take two pointers P1 and P2, both pointed to the first element.
Step 2:
Increment P1 by 1 and P2 by two.
Step 3:
Whenever P2 reaches to the end, P1 will be at the middle of the list, just return P1->data.

Nibble Swap And Bit Swapping

```
int main( void )
{
 unsigned char a = 40, b=20;
 a = ( a>>4 ) | ( a<<4 );
 b = ( ( b & 0xAA ) >> 1 ) | ( ( b & 0x55 ) << 1 );
 clrscr();
 printf( "After Nibble Swap %d\n", a );
 printf( "Bit swapping %d\n", b );
 getch();
 return 0;
}
```

C program to count the number of set bits in an unsigned integer

```
/*
Program to count no. of bits in an unsigned integer
```

```
*/
void main( void )
{
unsigned int a = 15;
int count = 0;

while( a )
{
++count;
a = a & ( a - 1 );
}

clrscr();
printf( "Count is %d\n", count );
getch();
}
```

Why is sizeof('a') not 1?

Perhaps surprisingly, character constants in C are of type int, so sizeof('a') is sizeof(int) (though it's different in C++).

Result:
In Turbo C output is: 2
In Turbo C++ output is: 1

Determine if a Variable is Signed or Not?

Using Typecasting

```
#define IS_SIGNED( _typedef_ ) ( (_typedef_) 0-1 > 0 )
```

Using 1's Compliment

```
#define IS_SIGNED2(__VAR__) ( ( __VAR__ > 0 ) && ( ~__VAR__ > 0 ) )
```

why n++ executes faster than n+1?

The expression n++ requires a single machine instruction such as INR to carry out the increment operation whereas, n+1 requires more instructions to carry out this operation.

Little endian to big endian

```
//2-byte number
int SHORT_little_endian_to_big_endian( int i)
{
return (( i>>8)&0xff)+((i<<8)&0xff00);
}
//4-byte number
int INT_little_endian_To_big_endian(int i)
{
return((i&0xff)<<24)+((i&0xff00)<<8)+
((i&0xff0000)>>8)+((i>>24)&0xff);
}
```

1. what is the role in your project?

2. what are the issues you find in the project?

How did you rectify those issues?

3. How interrupts are handled in linux?
Ans: When interrupt occurs the CPU stop executing the instructions that it is doing and go to interrupt handler by changing the mode to "INTERRUPT MODE". once the interrupt has been handled CPU goes to normal mode and start executing stoped instructions.

when ever the interrupt occures on interrupt line "do_irq" will execute and it quiree the interrupt controller and find out the IRQ line look up in the interrupt descriptor table for address of the handler and execute.

4. Difference between semaphore and mutex?
Ans: Semaphore: Another process can able to unlock the current process that is in critical section
Mutex: The process which is in critical section can only able unlock. Another process cannot...

5. Why won't you use global varable insted of semaphore and mutex?
Ans: Global variables may be interruptable so its value may be corruptable. But in case of semaphore and mutex these will use atomic variable which are not interruptable.

6. what are the IPC mechanisms?
Ans: pipes, message queues, shared memory, sockets.

7. Diff between shared memory and PIPES?
Ans: shared memory have boundaries, but pipes doesnt have. read followed by write must be there but in case of pipes not require. Synchronization technics require to handle shared memory to over come concurency.

8. what is function pointer? how to use, where to use?
Ans: pointer which holds address of a function. Used for call back functions. Interrupt handling

C-LANGUAGE:

1. can you please write a program to find the size of an integer without using sizeof( )

pointers:

```
#include<stdio.h>
int main(){
 int size, num;
 size = (char *)(&num + 1) - (char *)(&num);
 printf("size : %d", size);
}
```

Bitwise operators:

```
#include<stdio.h>
int main(){
 int size = 0, num = 1;
 while(num){
  size++;
  num = num << 1;
 }
 printf("size : %d", size);
 return 0;
}
```

2. Write a program to whether arch is little or big

```
#include<stdio.h>
int main(){
 int num = 1;
 if((*(char *)&num) == 1)
  printf("Little Endian\n"");
 else
  printf("Big  Endian\n");
 return 0;
}
```

3. Program to print how many number of time the given character is repeated

```
#include<stdio.h>

int main()
{
 char str[100], ch;
 int i, count = 0;

 printf("Enter the string :");
 scanf("%s", str);

 printf("\nEnter the character that you want to know how many times it is repeated :");
 scanf("%c", &ch);

 for(i=0; str[i] != '\0'; i++)
  if(str[i] == ch)
   count ++;

 printf("%c is repeated %d times", ch, count);
}
```

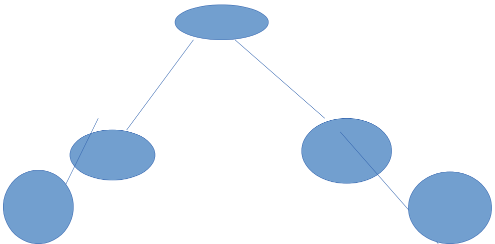
```
#include<stdio.h>

c(){
 printf("Present stack pointer value is %p",c);
}

a(){
 c();
}

b(){
 a();
}

main()
{
 b();
}
```

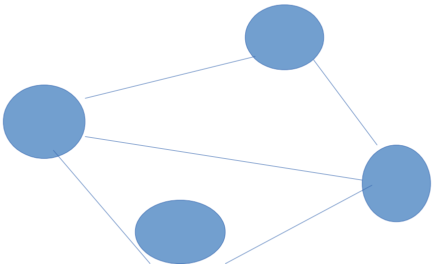
Samsung R&D,Bangalore Interview Process

1) Written test (C language (40 bits))
2) Technical Round 1
3) Technical Round 2
4) HR Round

Technical Round 1 Questions

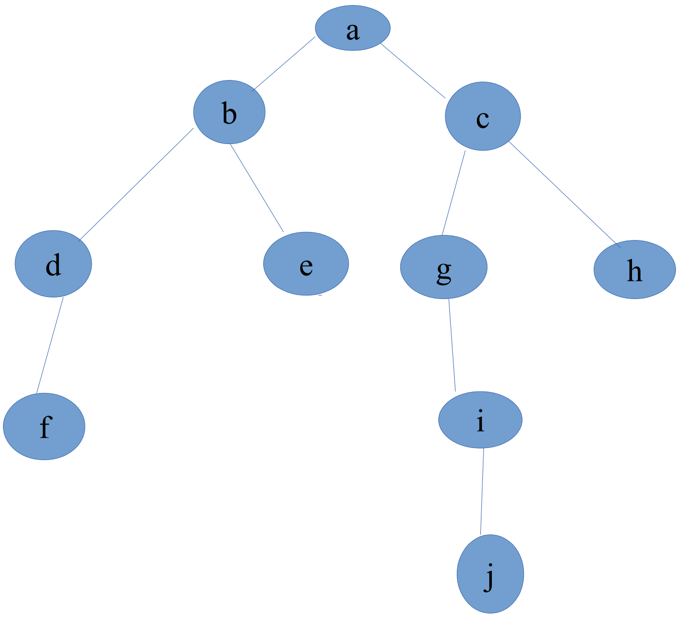
1) Explain your project display subsystem ?
2) How you did the porting ?
3) After downloading the kernel from kernel.org, what you did to create the binary image?
4) Write a character driver for 3 leds on/off ?
   I wrote the driver with nearly 125 lines, he wants explanation for each and every line
5) Draw the kernel diagram?
6) What are the topics you know in the above diagram?
7) What is malloc() and kmalloc() ?
8) What is difference between kmalloc() and kvmalloc()?
9) What is GFP_KERNEL and GFP_ATOMIC ? What is the main difference ?
10)Explain booting sequence for your board ?
11)Why DRAM was attached external not in system-on-chip?
12) How system call works ?
13) What is interrupt? How it works ?
14) How can you register ISR function? Explain arguments?
15)What is tophalf and bottomhalf? What is the main difference?
16) Where bottomhalf scheduled?
17)What is softirq?Give one example?
18)What are the synchronization techniques?
19)What is the difference between mutex and semaphore and spinlock?
20)What is difference between pipes ans message queues ans shared memory?
21) In the below figures identify tree and graph ?





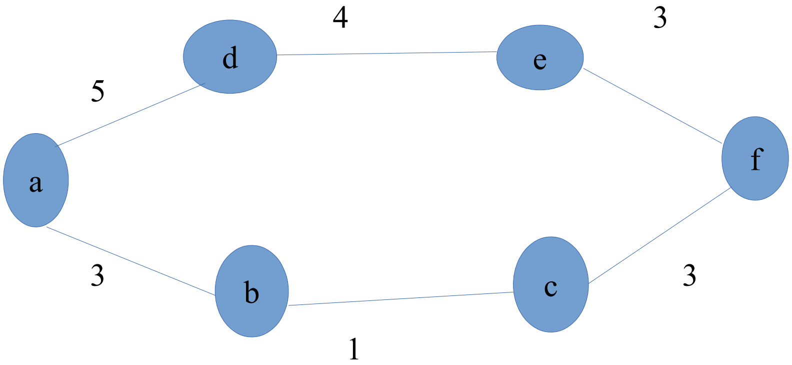
22)What is binary search tree ?Draw diagram?

23) What is depth of the tree for the following figure?



24)Draw the minimam heap and maximum heap tree?

25)



I want to travel from a-f, calculate the shortest distance?

26)Write a program for calculating sum of array int arr[]={1,2,3,4,5}

27)Write a program for calculating no of 1's in 10011010?

Technical Round 2

1)Explain I2C flow ?

2)What are the modifications you did in the project?

3)What is your slave address?

4)What is start and stop conditions?

5)How will you transfer the data from user space to kernel space?

6)By using read(),write() also ae can transfer the data,why you mention copy_from_user() and copy_to_user()?tell me the reason?

7)What are the different memory zones?Explain?

8)Explain the memory subsystem?

9)Write a program for architecture is a little indian/big indian?

10)Write a program for reverse of a string?

11)Write a program for size of a structure without using bitwise operators?

12)What is volatile?Can you explain what is it?

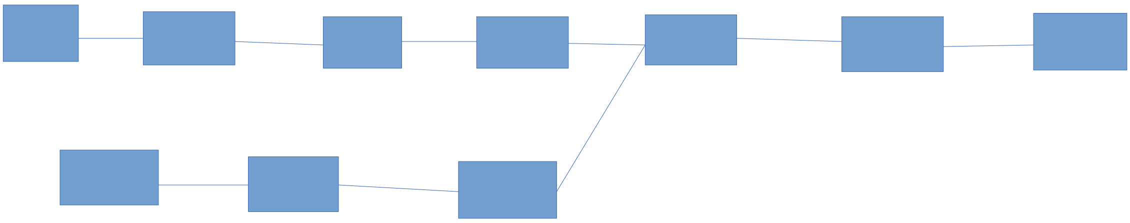
13)What type of instructions are stored in cache memory?

14)What is function pointer?Give function pointer example with three functions?

The interviewers and me discussed nearly 1/2 hour on this topic.

15)What is difference between global static and local static variable?

16)Can you tell the logic for to meet the two linked lists



18)Write a program for any of the sorting technique?

19)Write a program for swap the two numbers without using third variable?

20)How can you identify the middle element in the single linked list without using count variable? write the logic?

TECHNICAL 1:

puzzles:

1. you have a cake and you need to cut into 8 equal pieces by using three cuttings

2. you have 8 coins and a weighting machin. All are looks similar and 7 coins weights equal and one coin weight is diferent

(i.e,less weight comapared to remaining). how will you find out the less weigth coin.

3. 3 bulbs in one room and corresponding 3 switches in another room. you need to identify which switch is for which bulb

in a minimu no. of chances. if you go from one room to another room one chance will be completed. you have a choice that

you can switch ON the switch and you can go to see the bulb in the another room.

1. What is process and thread.
2. Difference between process and thread.
3. Is it possible to create a process with in a thread.
4. what we call the variables which are not a constant variables
5. How will you store the data into memory.
6. Difference between little endian and big endian and write a program. explan
7. Swap the two numbers with out using third variable.

```
 Ans1: num1 = num1 + num2;
       num2 = num1 - num2;
       num1 = num1 - num2;
```

If a variable is 16 bit and num1 = (2 ^ 16) -1 and num2 = (2^16) - 2. your code will fail because of of over flow.

So can you modify your code.

TECHNICAL 2:
1)explain what is bord bringup?
2)how integrated gpio in boot and kernel?
3)how the connection establish from your external touchscreen device to processor?
4)explain omap processor internally what architecture it used?
5)how arm architecture divided in application ,rtos,microcontroller domains?
6)what is difference between spinlocks and mutex?
7)explain difference between tasklets,workquese?

c questions:
1) what is static variable?
2)why we use volatile keyword?
3)write reverse linked list program?
4)how we cal whether given linked list is circular?
4)difference between “char *const ptr” and “const char *ptr”?
5)we have str1=”wednesday”; and str2=”nes”; then explain logic how determined whether str2 is in str1 or not?