Embedded System Software Technologies

Project Guide :

P.UdayaSree
Assistant Professor,
Dept Of CSE
IIIT-RGUKT, Rk Valley.

TeamMembers:
K.Prathyusha
R170902

Content

- Introduction
- Scope of the Project
- Technologies Used
- Why Advance C?
- Some Advance - 'C' Topics
- About BitFields.
- Future Scope
- Conclusion

INTRODUCTION

- Engineers use C programming for embedded systems to produce efficient and compact code. C is important in embedded systems because there is often limited memory and processing power.

- C programming produces very reliable code enabling embedded systems to execute mission-critical tasks.

Scope of the Learning 'C'

- Although many computer languages are used for writing computer applications, the computer programming language.

- If you are interested in a career in computer programming, it would be wise to start by learning the C programming language.

Continutation…….

- C, is that it's the basic language of all advanced computer languages.
- For example, if you want to learn C++, which is an object oriented language, you need to know the C language well beforehand.

Technologies Used

Why we use Advance C?

- Designed to take your basic C skills to the next level and help you obtain mastery of the language by helping you understand advanced concepts of the C programming language, enabling you to master the art of problem-solving in programming using efficient, proven methods.

Some Advance - 'C' Topics

- These are the Advance C topics:-

  Pointers

  BitFields

  Memory / Dynamic Memory.

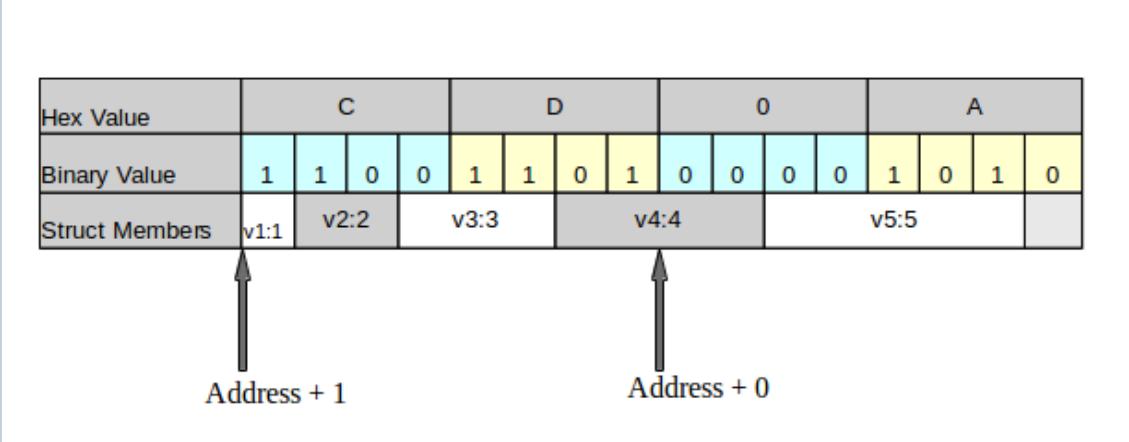
  I/O & Function Topics.

  Debugging / Exception Handling.

Some Examples Regrading Advance ‘c’.

- BITFIELDS:

Use of bit fields is one of the key optimisation methods in embedded C programming, because these allow one to pack together several related entities, where each set of bits and single bits can be addressed.



One Simple Example for Bitfields

```
#include<stdio.h>
struct date{
unsigned int d:5;
unsigned int m:4;
unsigned int y:23;
};
int main()
{printf("Size of date is %d is %d bytes\n",sizeof(struct date);
struct date dt={09,06,2020};
printf("Date is %d%d%d",dt.d,dt.m,dt.y);
}
output:Size of date is 4 bytes
Date is 09/06/2020
```

- **Purpose of this code is ...Here we are assiging particular required bits to the date,month,minutes.**

By this we can save memory.

Interesting facts about bit fields in c

- A Special unnamed bit field of size 0 is used to force alignment on next boundary.
- We Cannot have pointers to bit field members as they may not start at a byte boundary.
- Assigning an out-of-range value to a bit field member is implementation defined(complier dependent).
- In c++,we can have static members in a structure/class,but bit fields cannot be static.

Pointers

- Pointer is a variable usedd to hold the address of another variable of the same type.
- Two impotant operators;
- & - address operator
- * - value at address operator.

Examples for pointer

A Pointer Example

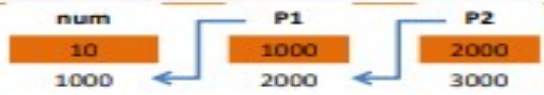
```
main()
{
  int num = 10 ;
  int *ptr;
  ptr = &num;//ptr = 1000
   printf(" Address of Num = %d\n", ptr);
  printf("Value of num = %d   %d\n", *ptr, num);

 num = 20;
 printf("Value of num = %d   %d\n", *ptr,  num);

*ptr = 50;
 printf("Value of num = %d   %d\n", *ptr, num);
 }
```

Pointer to a pointer Example

```
main()
{
  int num = 10 ;
  int *p1;
  int **p2;
  p1 = &num;
  p2 = &p1;
  printf(" %d  %d  %d", num,  p1,  p2);
  printf(" %d  %d", *p1,  *p2);
  printf("  %d  ",  **p2 );
}
```



Future Scope

- Future scope in Embedded C is really good. As most of Electronics Engineer go for job in IT domain..so its good if you don't go with the crowd.
- Also as a Fresher we think there is very less opportunities in this field but that's not at all true because if you have good knowledge of microcontrollers and Embedded C coding you can go for companies which work in automation domain, automotive domain, IOT is also very good. You can work on/create your own Device drivers, can work as an Linux developer.
- There are so many well established companies as well as startup companies in thease domain .

Conclusion

- C provides optimized machine instructions for the given input, which increases the performance of the embedded system. Most of the high-level languages rely on libraries, hence they require more memory which is a major challenge in embedded systems.

Thank you..!!!

Presented by:

K.PRATHYUSHA