

Chapter 10

Programming in C

Lesson 05

Functions in C

C program

- Consists of three parts
- preprocessor directives
- macros
- main function
- functions

Function

- Each has a name (for identity ID)
- May have the arguments
- A function may or may not return certain data or data-set or object
- A standard C declaration—

<[>return_type<]> functionName
(<[>argumnets<]>).

Routine

- A set of instructions
- The instructions of the routine execute as follows—
- Has a distinct start address
- The program counter (PC) acquires the start address value after the earlier value of PC is saved, execute

Routine

- The start address has the first instruction, which executes first in a routine
- The routine has a return instruction at the end
- The previously saved value of program counter is restored back into the PC on the return from the routine
- A routine equivalent–*function* in C

Main Function in C

- *main* in C, if it is the first function to execute at the beginning of a program
- Each function also has a start address
- The program source file execution starts from the main function invoked by *void main () {; }*.
- Void means that this function (program codes) does not return any variable to any other function (routine)

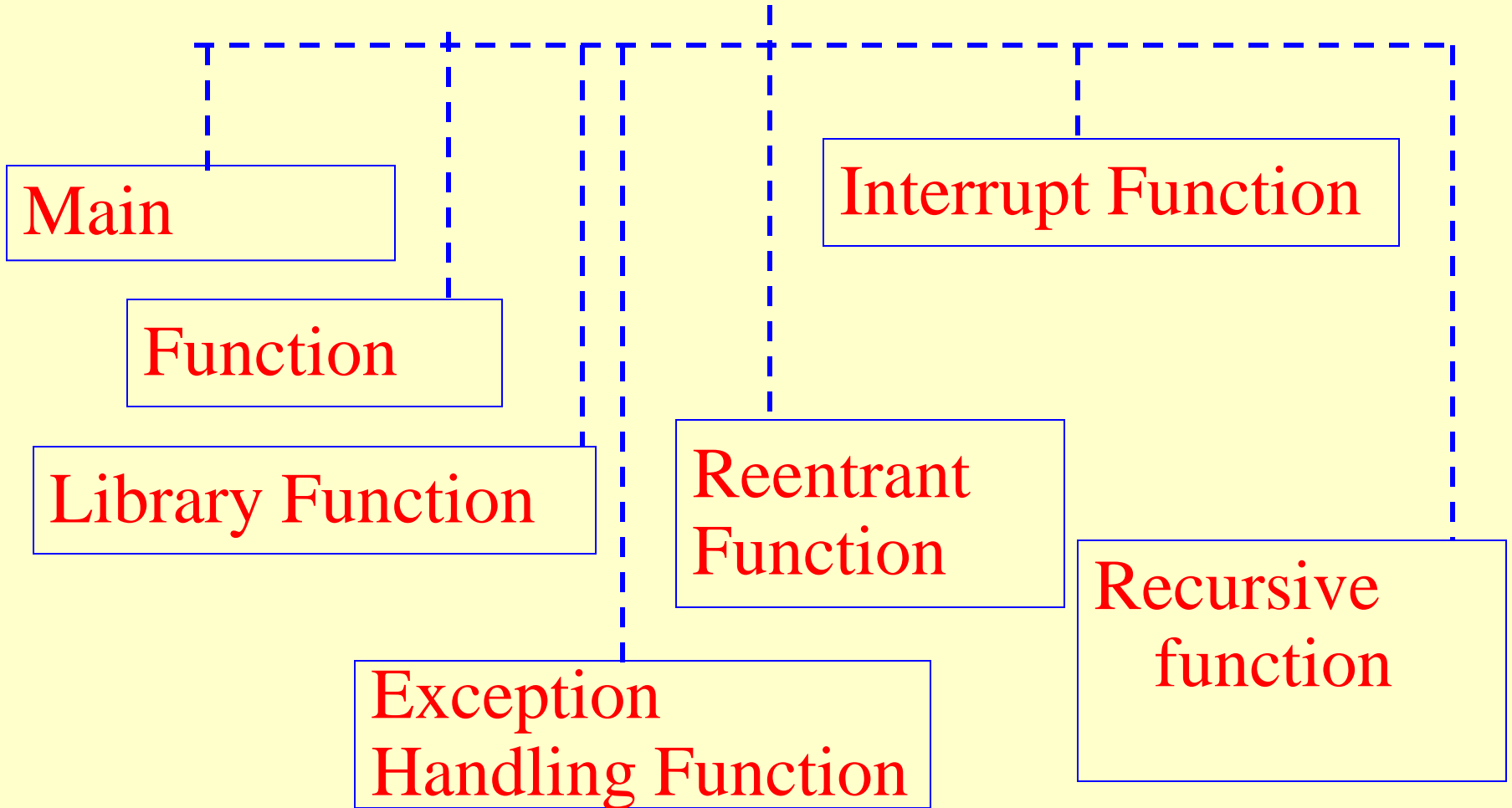
Before the main

- Before this statement, there are preprocessor directives like *include*.

Interrupt Function in C

- Interrupt service routine—*interrupt function* in C

Function



Reentrant Function in C

- *F*unction, which on interrupt or which on calling another function, reenters on the return to the same state of the variables and CPU registers as before the interrupt or call

void before function ID

- void timeSet (int numTicks)
- The void means function return type is void and it returns no variable or object
- The name of the function (Function ID) is timeSet

timeSet

- timeSet— one argument
- Uses the integer value of numTicks as input
- The numTicks— a variable
- timeSet gets the value of numTicks from outside the function
- The value is passed to the functions from outside the function block [to a local variable memory address from another function]

int timeGet ()

- int means function return type is not void but an integer
- The function returns value of an integer variable
- The name (ID) of the function—timeGet
- It gets the time using a global variable for specifying the time
- There are no arguments in the function

Library Function

- Library means the readily available functions
- Library functions are directly used by the programmer
- The compiler uses the functions statements from the library

C Function

1. Use of modifiers-

unsigned int; unsigned char;

2. Use of data types and struc

3. Declarations and Statements

4. Comments at appropriate places and lines

5. Control structures; if then else, case and for, while and repeat until loops

C Function

6. Use of pointers-

```
unsigned long *time_date;
```

7. Use of arrays

```
char [ ] name;
```

8. Inline assembly codes

C Function Parameter passing

9. Use of arguments

void profile (name, address)

10. Use of passing values and addresses of variables in the arguments

11. Returning nothing (void) or data or data struc

char [] welcomeMsg (msg1, sig);

Keil Cx51 compiler library functions.

- A set of mathematical, memory, string and other ANSI standard library functions
- String manipulation functions—used to manipulate the strings.

String manipulation functions

- For example, `strlen ()` is a String manipulation library-function. It evaluates the length of the string (number of characters in a string).

Memory allocation functions

- Allot the memory
- `calloc ()` is a library-function
- `calloc ()` allots the memory for an array from available memory pool

memcpy ()

- Additional library function
- copy from one memory to other memory space
-

Exclusion in Keil C

- Exclusion of certain ANSI standard function
- For qsort () for quick sorting of array is excluded.

In-line Assembly in Fuction

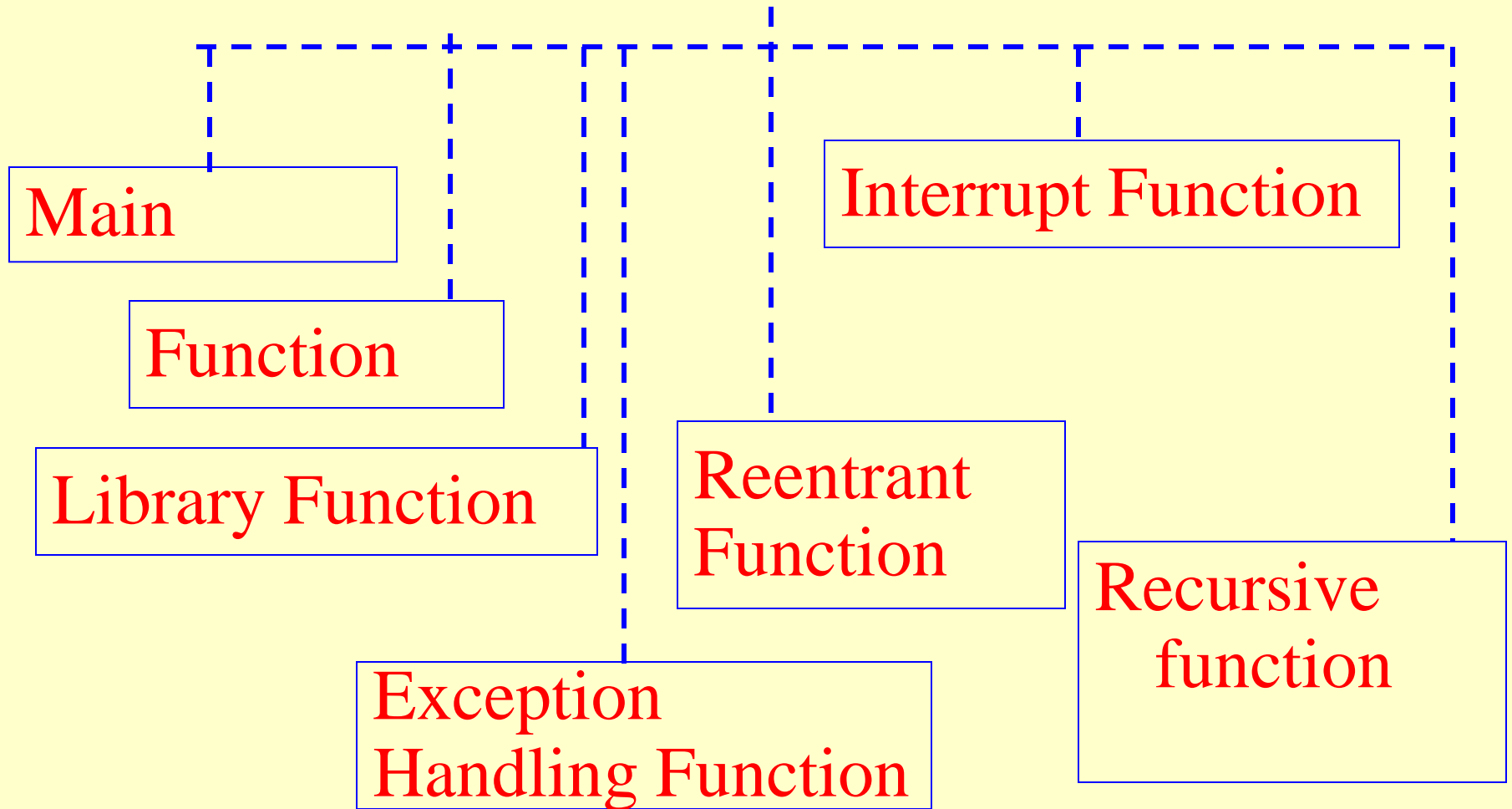
- Function `outportbpinv (b, p); /*b is the address of bit 1 of a port and p is the memory address of bit n to be transferred after complementing.*/`
- `void outportbpinv (b, p) {`
- `asm { CPL p; MOV C, p; MOV b, C;}`
- `};`

Summary

We learnt

- C program consists of functions only
- A function can also be declared as *void* function

We learnt meanings of Function



We learnt

- A function is a routine, which can be passed the values or pointers through the arguments and which additionally can return a value or data-object or pointer by the function name

We learnt

- Parameters to a function can be passed by value or by reference

End of Lesson 05 on

Functions in C