Structures

Nick Scheele

By default the compiler stores variables using entire words, but when the packed attribute is used only the required amount of space is used.

main.c

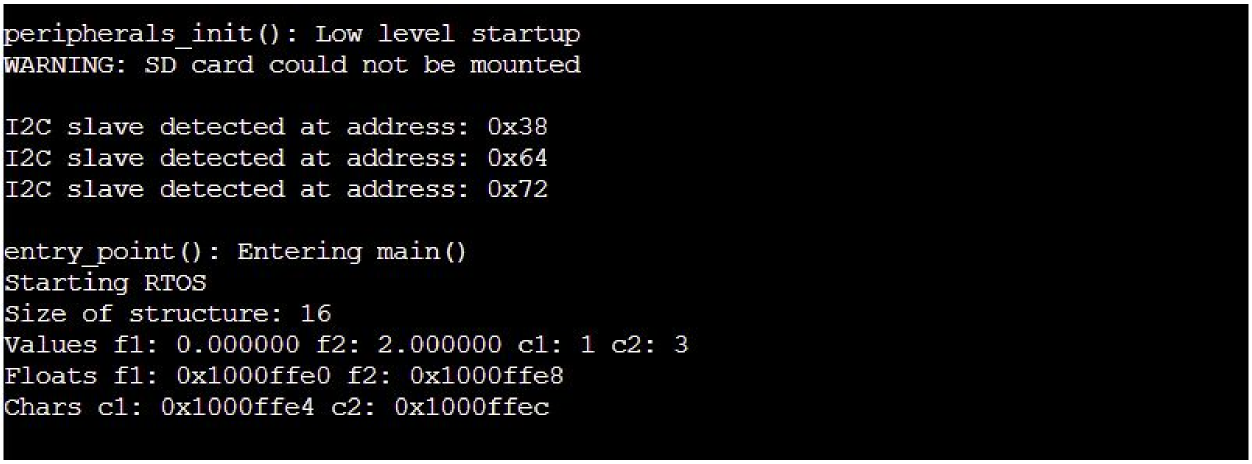
```c
typedef struct __attribute__((packed)) {
 float f1;
 char c1;
 float f2;
 char c2;
} my_s;

#define outOfTheBox 1

int main(void) { // main function for project
 puts("Starting RTOS");

#if outOfTheBox

 //  create_blinky_tasks();
 //  create_uart_task();
 my_s s;
 s.f1 = 0;
 s.c1 = '1';
 s.f2 = 2;
 s.c2 = '3';
 printf("Size of structure: %d\n"
        "Values f1: %f f2: %f c1: %c c2: %c\n"
        "Floats f1: %p f2: %p\n"
        "Chars c1: %p c2: %p\n",
        sizeof(s), s.f1, s.f2, s.c1, s.c2, &s.f1, &s.f2, &s.c1, &s.c2);
```

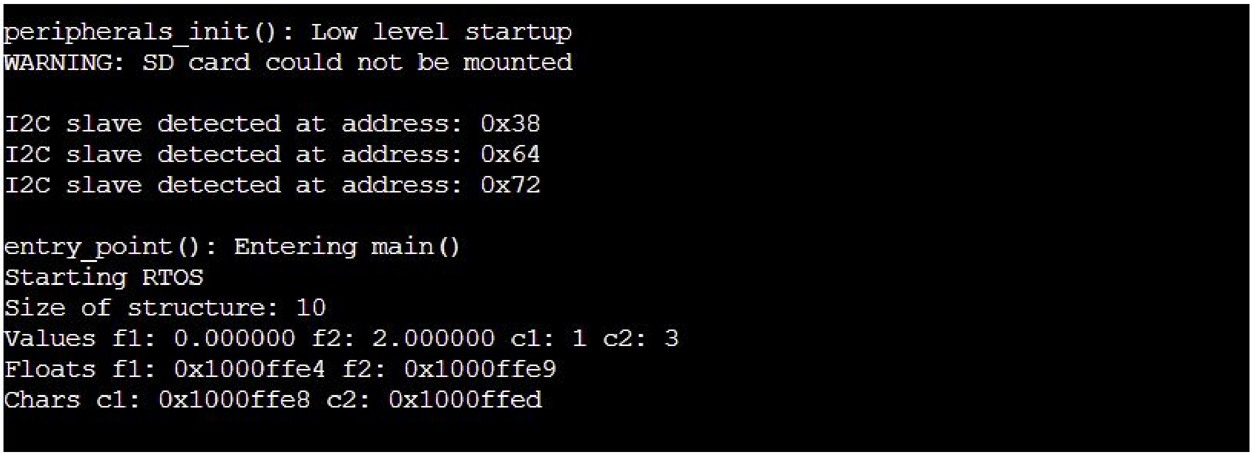


```
peripherals_init(): Low level startup
WARNING: SD card could not be mounted

I2C slave detected at address: 0x38
I2C slave detected at address: 0x64
I2C slave detected at address: 0x72

entry_point(): Entering main()
Starting RTOS
Size of structure: 16
Values f1: 0.000000 f2: 2.000000 c1: 1 c2: 3
Floats f1: 0x1000ffe0 f2: 0x1000ffe8
Chars c1: 0x1000ffe4 c2: 0x1000ffec
```

Figure 1: default



```
peripherals_init(): Low level startup
WARNING: SD card could not be mounted

I2C slave detected at address: 0x38
I2C slave detected at address: 0x64
I2C slave detected at address: 0x72

entry_point(): Entering main()
Starting RTOS
Size of structure: 10
Values f1: 0.000000 f2: 2.000000 c1: 1 c2: 3
Floats f1: 0x1000ffe4 f2: 0x1000ffe9
Chars c1: 0x1000ffe8 c2: 0x1000ffed
```

Figure 2: Packed