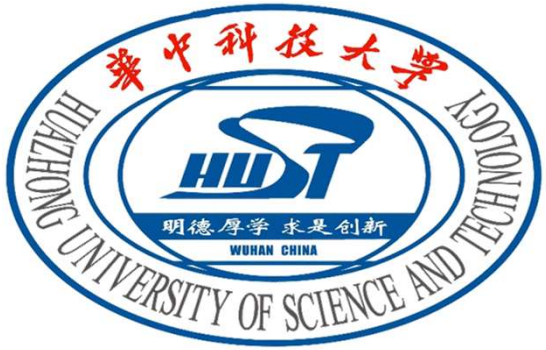
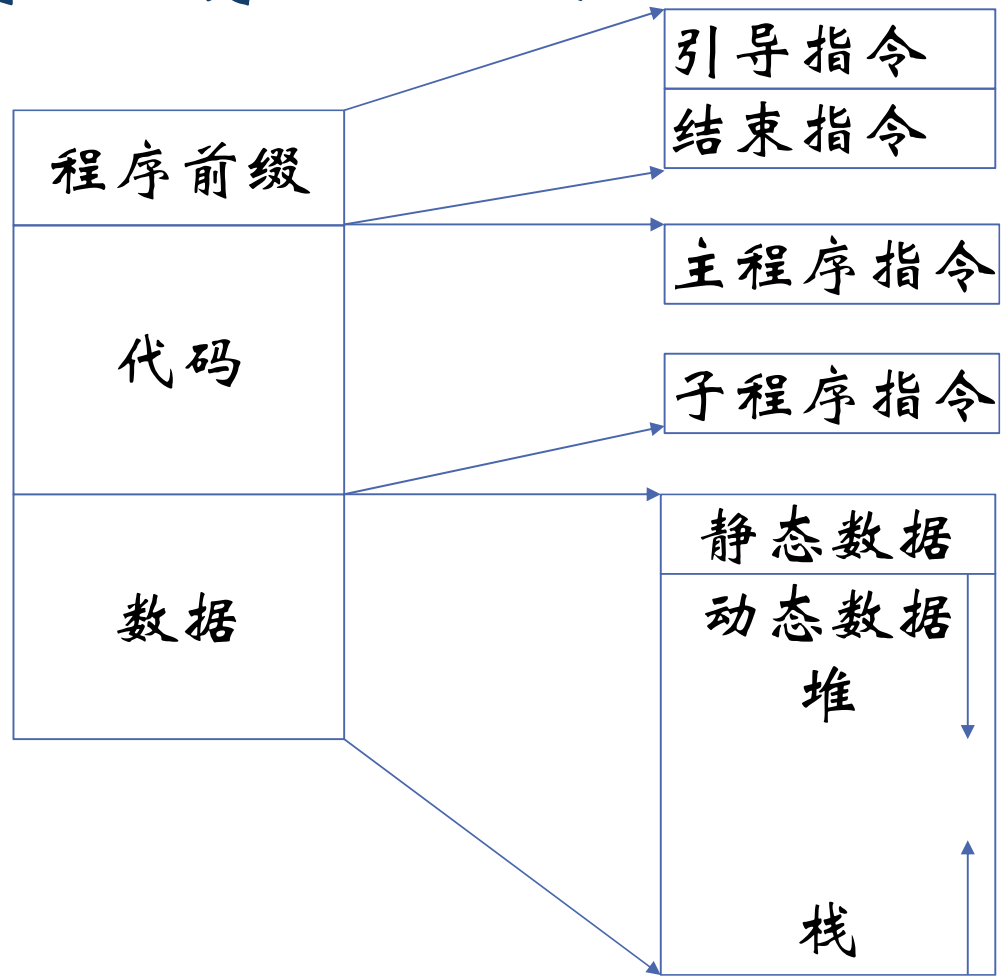
微机原理与接口技术

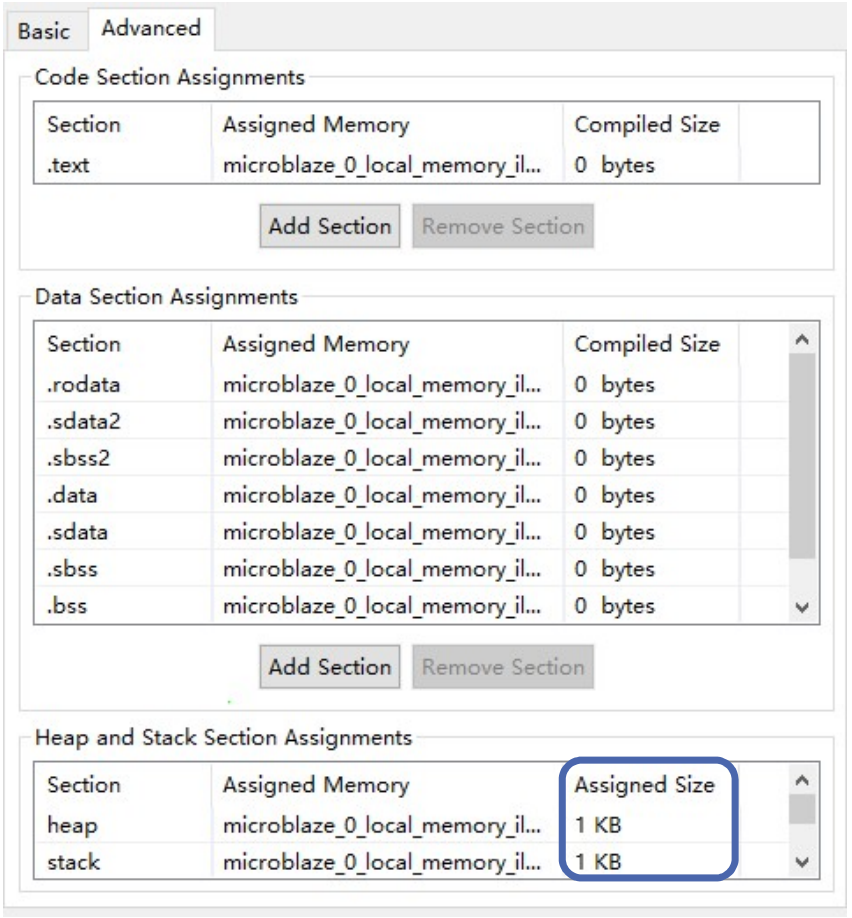
程序的存储映像

华中科技大学　左冬红

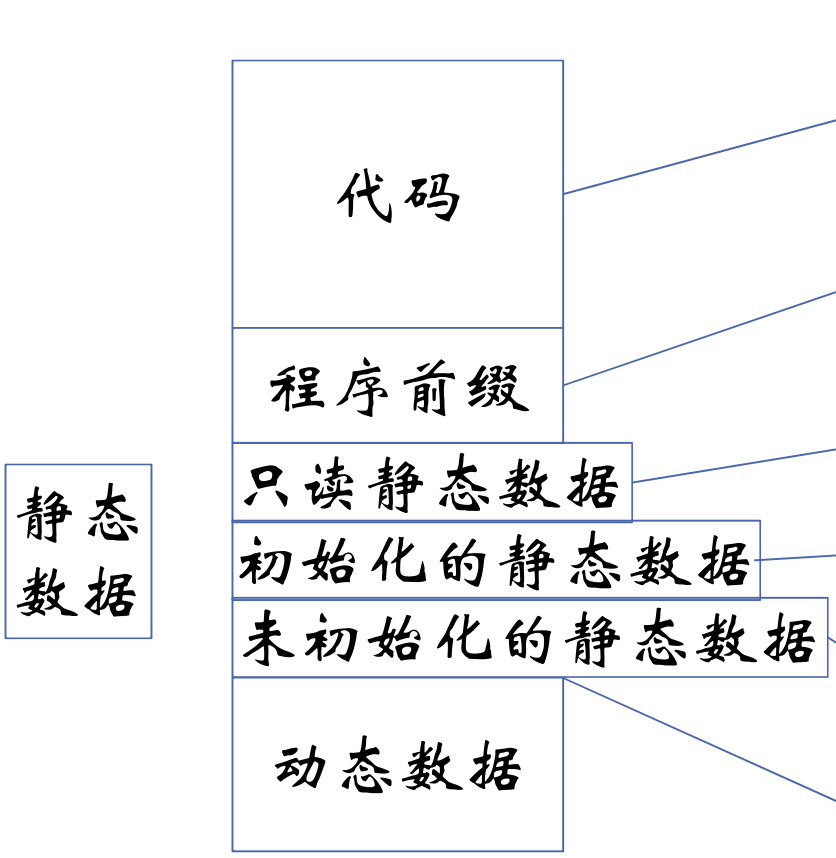


程序的存储映像


程序前缀
代码
数据
引导指令
结束指令
主程序指令
子程序指令
静态数据
动态数据
堆
栈


Basic
Advanced
Code Section Assignments
Section
Assigned Memory
Compiled Size
.text
microblaze_0_local_memory_il...
0 bytes
Add Section
Remove Section
Data Section Assignments
Section
Assigned Memory
Compiled Size
.rodata
microblaze_0_local_memory_il...
0 bytes
.sdata2
microblaze_0_local_memory_il...
0 bytes
.sbss2
microblaze_0_local_memory_il...
0 bytes
.data
microblaze_0_local_memory_il...
0 bytes
.sdata
microblaze_0_local_memory_il...
0 bytes
.sbss
microblaze_0_local_memory_il...
0 bytes
.bss
microblaze_0_local_memory_il...
0 bytes
Add Section
Remove Section
Heap and Stack Section Assignments
Section
Assigned Memory
Assigned Size
heap
microblaze_0_local_memory_il...
1 KB
stack
microblaze_0_local_memory_il...
1 KB

程序存储映像与elf文件关系




```
Idx Name          Size      VMA       LMA       File off  Algn
 4 .text         00002d0c  00000050  00000050  0000009c  2**2
                 CONTENTS, ALLOC, LOAD, CODE
 5 .init         0000003c  00002d5c  00002d5c  00002da8  2**2
                 CONTENTS, ALLOC, LOAD, READONLY, CODE
 6 .fini         00000020  00002d98  00002d98  00002de4  2**2
                 CONTENTS, ALLOC, LOAD, READONLY, CODE
 7 .ctors        00000008  00002db8  00002db8  00002e04  2**2
                 CONTENTS, ALLOC, LOAD, DATA
 8 .dtors        00000008  00002dc0  00002dc0  00002e0c  2**2
                 CONTENTS, ALLOC, LOAD, DATA
 9 .rodata       00000008  00002dc8  00002dc8  00002e14  2**2
                 CONTENTS, ALLOC, LOAD, READONLY, DATA
10 .sdata2       00000000  00002dd0  00002dd0  00003350  2**0
                 CONTENTS
11 .data         00000534  00002dd0  00002dd0  00002e1c  2**2
                 CONTENTS, ALLOC, LOAD, DATA
12 .sdata        00000004  00003304  00003304  00003350  2**0
                 ALLOC
13 .sbss         00000000  00003308  00003308  00003350  2**0
                 CONTENTS
14 .bss          00000068  00003308  00003308  00003350  2**2
                 ALLOC
15 .heap         00000400  00003370  00003370  00003350  2**0
                 ALLOC
16 .stack        00000400  00003770  00003770  00003350  2**0
                 ALLOC
```

C语言程序示例

```
#include "stdlib.h"
typedef struct monkey{
    int age;
    int master;
    int code;
}monkey;
monkey jack;
int sum(void *param)
{
    monkey *thismonkey=(monkey*)param;
    int sum =thismonkey->age+thismonkey->master+thismonkey->code;
    return sum;
}
```

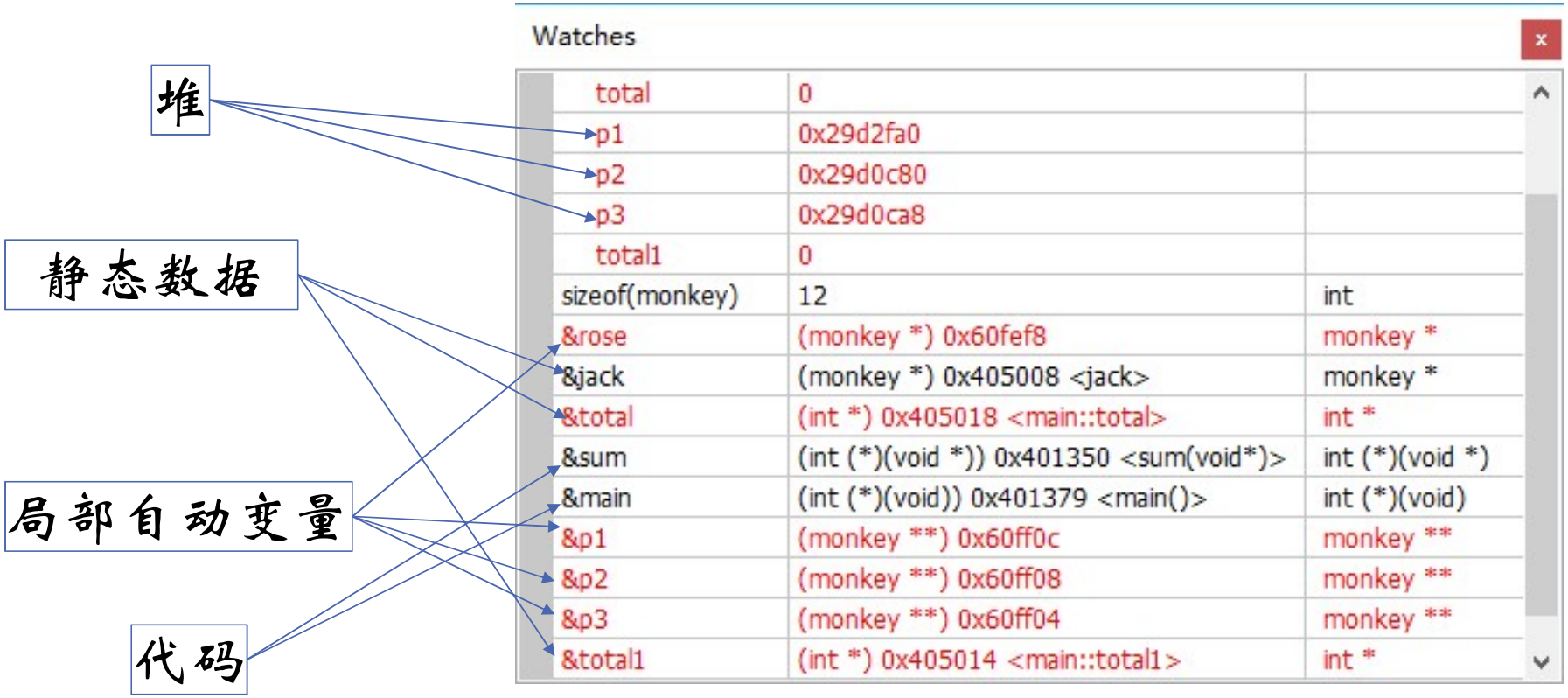
全局变量

局部自动变量

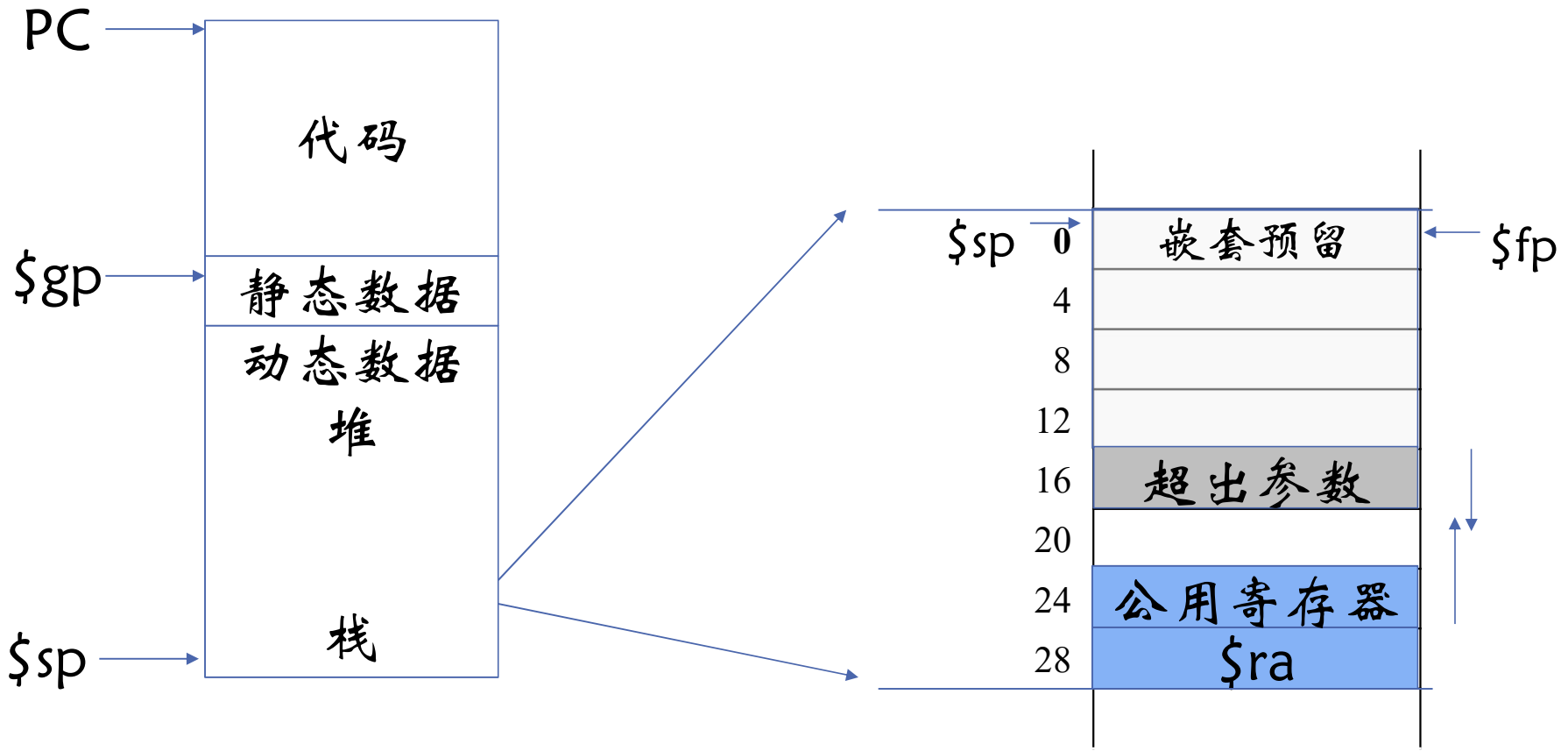
静态变量

```
int main()
{
    monkey rose;
    monkey *p1,*p2,*p3;
    p1=(monkey *)malloc(sizeof(monkey));
    p2=(monkey *)malloc(sizeof(monkey));
    p3=(monkey *)malloc(sizeof(monkey));
    static int total;
    static int total1;
    return 0;
}
```

示例C语言程序的数据存储映像



MIPS寄存器特殊功能



小结

- 程序在内存中的一般结构
- elf文件与存储结构相关部分的解析
- C语言程序数据存储解析
- 编译器栈管理

下一讲：字符处理程序的实现原理