

1. Warning 280:'i':unreferenced local variable

说明 局部变量 i 在函数中未作任何的存取操作

解决方法 消除函数中 i 变量的宣告

2 Warning 206:'Music3':missing function-prototype

说明 Music3()函数未作宣告 或未作外部宣告 所以无法给其他函数调用

解决方法 将叙述 void Music3(void)写在程序的最前端作宣告 如果是其他文件的函数则要写成 extern void Music3(void),即作外部宣告

3 Compling :C:\8051\MANN.C

Error:318:can't open file 'beep.h'

说明 在编译 C:\8051\MANN.C 程序过程中 由于 main.c 用了指令#include "beep.h",但却找不到所致

解决方法 编写一个 beep.h 的包含档并存入到 c:\8051 的工作目录中

4 Compling:C:\8051\LED.C

Error 237:'LedOn':function already has a body

说明 LedOn()函数名称重复定义 即有两个以上一样的函数名称

解决方法 修正其中的一个函数名称 使得函数名称都是独立的

5 *WARNING 16:UNCALLED SEGMENT,IGNORED FOR OVERLAY PROCESS SEGMENT: ?PR?_DELAYX1MS?DELAY**

说明 DelayX1ms()函数未被其它函数调用 也会占用程序记忆体空间

解决方法 去掉 DelayX1ms()函数 或利用条件编译#if#endif,可保留该函数并不编译

6 *WARNING 6 :XDATA SPACE MEMORY OVERLAP FROM : 0025H**

TO: 0025H

说明 外部资料 ROM 的 0025H 重复定义地址

解决方法 外部资料 ROM 的定义如下

Pdata unsigned char XFR_ADC _at_0x25 其中 XFR_ADC 变量的名称为 0x25,请检查是否有其它的变量名称也是定义在 0x25 处并修正它

7 WARNING 206:'DelayX1ms': missing function-prototype C:\8051\INPUT.C

Error 267 :'DelayX1ms `:requires ANSI-style prototype C:\8051\INPUT.C

说明 程序中有调用 DelayX1ms 函数 但该函数没定义 即未编写程序内容或函数已定义但

未作宣告

解决方法 编写 DelayX1ms 的内容 编写完后也要作宣告或作外部宣告 可在 delay.h 的包含档宣告成外部 以便其它函数调用

8 *WARNING 1:UNRESOLVED EXTERNAL SYMBOL**

SYMBOL:MUSIC3

MODULE:C:\8051\MUSIC.OBJ(MUSIC)

*****WARNING 2:REFERENCE MADE TO UNRESOLVED EXTERNAL**

SYMBOL:MUSIC3

MODULE:C:\8051\MUSIC.OBJ(MUSIC)

ADDRESS:0018H

说明 程序中有调用 MUSIC 函数 但未将该函数的含扩档 C 加入到工程档 Prj 作编译和连接

解决方法 设 MUSIC3 函数在 MUSIC C 里 将 MUSIC C 添加到工程文件中去

9 *ERROR 107:ADDESS SPACE OVERFLOW**

SPACE: DATA

SEGMENT: _DATA_GOUP_

LENGTH: 0018H

*****ERROR 118: REFERENCE MADE TO ERRONEOUS EXTERNAL**

SYMBOL: VOLUME

MODULE: C:\8051\OSDM.OBJ(OSDM)

ADDRESS: 4036H

说明 data 存储空间的地址范围为 0~0x7f,当公用变量数目和函数里的局部变量 如果存储模式设为 SMALL 则局部变量先使用工作寄存器 R2~R7 作暂存 当存储器不够用时则会以 data 型别的空间作暂存的个数超过 0x7f 时就会出现地址不够的现象

解决方法 将以 data 型别定义的公共变量修改为 idata 型别的定义

Ambiguous operators need parentheses

不明确的运算需要用括号括起

Ambiguous symbol ``xxx``

不明确的符号

Argument list syntax error

参数表语法错误

Array bounds missing

丢失数组界限符

Array size toolarge

数组尺寸太大

Bad character in paramenters

参数中有不适当的字符

Bad file name format in include directive

包含命令中文件名格式不正确

Bad ifdef directive synatax

编译预处理 `ifdef` 有语法错

Bad undef directive syntax

编译预处理 `undef` 有语法错

Bit field too large

位字段太长

Call of non-function

调用未定义的函数

Call to function with no prototype

调用函数时没有函数的说明

Cannot modify a const object

不允许修改常量对象

Case outside of switch

漏掉了 case 语句

Case syntax error

Case 语法错误

Code has no effect

代码不可达不可能执行到

Compound statement missing{

分程序漏掉"{"

Conflicting type modifiers

不明确类型说明符

Constant expression required

要求常量表达式

Constant out of range in comparison

在比较中常量超出范围

Conversion may lose significant digits

转换时会丢失意义的数字

Conversion of near pointer not allowed

不允许转换近指针

Could not find file ``xxx``

找不到 XXX 文件

Declaration missing ;

说明缺少"; "

Declaration syntax error

说明中出现语法错误

Default outside of switch

Default 出现在 switch 语句之外

Define directive needs an identifier

定义编译预处理需要标识符

Division by zero

用零作除数

Do statement must have while

Do-while 语句中缺少 while 部分

Enum syntax error

枚举类型语法错误

Enumeration constant syntax error

枚举常数语法错误

Error directive :xxx

错误的编译预处理命令

Error writing output file

写输出文件错误

Expression syntax error

表达式语法错误

Extra parameter in call

调用时出现多余错误

File name too long

文件名太长

Function call missing)

函数调用缺少右括号

Fuction definition out of place

函数定义位置错误

Fuction should return a value

函数必需返回一个值

Goto statement missing label

Goto 语句没有标号

Hexadecimal or octal constant too large

16 进制或 8 进制常数太大

Illegal character ``x``

非法字符 x

Illegal initialization

非法的初始化

Illegal octal digit

非法的 8 进制数字

Illegal pointer subtraction

非法的指针相减

Illegal structure operation

非法的结构体操作

Illegal use of floating point

非法的浮点运算

Illegal use of pointer

指针使用非法

Improper use of a typedefsymbol

类型定义符号使用不恰当

In-line assembly not allowed

不允许使用行间汇编

Incompatible storage class

存储类别不相容

Incompatible type conversion

不相容的类型转换

Incorrect number format

错误的数字格式

Incorrect use of default

Default 使用不当

Invalid indirection

无效的间接运算

Invalid pointer addition

指针相加无效

Irreducible expression tree

无法执行的表达式运算

Lvalue required

需要逻辑值 0 或非 0 值

Macro argument syntax error

宏参数语法错误

Macro expansion too long

宏的扩展以后太长

Mismatched number of parameters in definition

定义中参数个数不匹配

Misplaced break

此处不应出现 break 语句

Misplaced continue

此处不应出现 continue 语句

Misplaced decimal point

此处不应出现小数点

Misplaced elif directive

不应编译预处理 elif

Misplaced else

此处不应出现 `else`

Misplaced else directive

此处不应出现编译预处理 `else`

Misplaced endif directive

此处不应出现编译预处理 `endif`

Must be addressable

必须是可以编址的

Must take address of memory location

必须存储定位的地址

No declaration for function ``xxx``

没有函数 `xxx` 的说明

No stack

缺少堆栈

No type information

没有类型信息

Non-portable pointer assignment

不可移动的指针（地址常数）赋值

Non-portable pointer comparison

不可移动的指针（地址常数）比较

Non-portable pointer conversion

不可移动的指针（地址常数）转换

Not a valid expression format type

不合法的表达式格式

Not an allowed type

不允许使用的类型

Numeric constant too large

数值常太大

Out of memory

内存不够用

Parameter ``xxx`` is never used

参数 xxx 没有用到

Pointer required on left side of ->

符号->的左边必须是指针

Possible use of ``xxx`` before definition

在定义之前就使用了 xxx (警告)

Possibly incorrect assignment

赋值可能不正确

Redeclaration of ``xxx``

重复定义了 xxx

Redefinition of ``xxx`` is not identical

xxx 的两次定义不一致

Register allocation failure

寄存器定址失败

Repeat count needs an lvalue

重复计数需要逻辑值

Size of structure or array not known

结构体或数组大小不确定

Statement missing ;

语句后缺少"; "

Structure or union syntax error

结构体或联合体语法错误

Structure size too large

结构体尺寸太大

Sub scripting missing]

下标缺少右方括号

Superfluous & with function or array

函数或数组中有多余的"&"

Suspicious pointer conversion

可疑的指针转换

Symbol limit exceeded

符号超限

Too few parameters in call

函数调用时的实参少于函数的参数不

Too many default cases

Default 太多(switch 语句中一个)

Too many error or warning messages

错误或警告信息太多

Too many type in declaration

说明中类型太多

Too much auto memory in function

函数用到的局部存储太多

Too much global data defined in file

文件中全局数据太多

Two consecutive dots

两个连续的句点

Type mismatch in parameter xxx

参数 xxx 类型不匹配

Type mismatch in redeclaration of ``xxx``

xxx 重定义的类型不匹配

Unable to create output file ``xxx``

无法建立输出文件 xxx

Unable to open include file ``xxx``

无法打开被包含的文件 xxx

Unable to open input file ``xxx``

无法打开输入文件 xxx

Undefined label ``xxx``

没有定义的标号 xxx

Undefined structure ``xxx``

没有定义的结构 xxx

Undefined symbol ``xxx``

没有定义的符号 xxx

Unexpected end of file in comment started on line xxx

从 xxx 行开始的注解尚未结束文件不能结束

Unexpected end of file in conditional started on line xxx

从 xxx 开始的条件语句尚未结束文件不能结束

Unknown assemble instruction

未知的汇编结构

Unknown option

未知的操作

Unknown preprocessor directive: ``xxx``

不认识的预处理命令 xxx

Unreachable code

无路可达的代码

Unterminated string or character constant

字符串缺少引号

User break

用户强行中断了程序

Void functions may not return a value

Void 类型的函数不应有返回值

Wrong number of arguments

调用函数的参数数目错

``xxx`` not an argument

xxx 不是参数

``xxx`` not part of structure

xxx 不是结构体的一部分

xxx statement missing (

xxx 语句缺少左括号

xxx statement missing)

xxx 语句缺少右括号

xxx statement missing ;

xxx 缺少分号

xxx`` declared but never used

说明了 xxx 但没有使用

xxx`` is assigned a value which is never used

给 xxx 赋了值但未用过

Zero length structure

结构体的长度为零

1.第一种错误信息

```
***WARNING L15: MULTIPLE CALL TO SEGMENT
SEGMENT: ?PR?_WRITE_GMVLX1_REG?D_GMVLX1
CALLER1: ?PR?VSYNC_INTERRUPT?MAIN
CALLER2: ?C_C51STARTUP

***WARNING L15: MULTIPLE CALL TO SEGMENT
SEGMENT: ?PR?_SPI_SEND_WORD?D_SPI
CALLER1: ?PR?VSYNC_INTERRUPT?MAIN
CALLER2: ?C_C51STARTUP

***WARNING L15: MULTIPLE CALL TO SEGMENT
SEGMENT: ?PR?SPI_RECEIVE_WORD?D_SPI
CALLER1: ?PR?VSYNC_INTERRUPT?MAIN
CALLER2: ?C_C51STARTUP
```

该警告表示连接器发现有一个函数可能会被主函数和一个中断服务程序(或者调用中断服务程序的函数)同时调用,或者同时被多个中断服务程序调用。

出现这种问题的原因之一是这个函数是不可重入性函数,当该函数运行时它可能会被一个中断打断,从而使得结果发生变化并可能会引起一些变量形式的冲突(即引起函数内一些数据的丢失,可重入性函数在任何时候都可以被 **ISR** 打断,一段时间后又可以运行,但是相应数据不会丢失)。

原因之二是用于局部变量和变量(暂且这样翻译,arguments,[自变量,变元一数值,用于确定程序或子程序的值])的内存区被其他函数的内存区所覆盖,如果该函数被中断,则它的内存区就会被使用,这将导致其他函数的内存冲突。

例如,第一个警告中函数 `WRITE_GMVLX1_REG` 在 `D_GMVLX1.C` 或者 `D_GMVLX1.A51` 被定义,它被一个中断服务程序或者一个调用了中断服务程序的函数调用了,调用它的函数是 `VSYNC_INTERRUPT`,在 `MAIN.C` 中。

解决方法:

如果你确定两个函数决不会在同一时间执行(该函数被主程序调用并且中断被禁止),并且该函数不占用内存(假设只使用寄存器),则你可以完全忽略这种警告。

如果该函数占用了内存,则应该使用连接器(linker)`OVERLAY` 指令将函数从覆盖分析

(overlay analysis)中除去,例如:

```
OVERLAY (?PR?_WRITE_GMVLX1_REG?D_GMVLX1!*)
```

上面的指令防止了该函数使用的内存区被其他函数覆盖。如果该函数中调用了其他函数,而这些被调用在程序中其他地方也被调用,

你可能会需要也将这些函数排除在覆盖分析(overlay analysis)之外。这种 OVERLAY 指令能使编译器除去上述警告信息。

如果函数可以在其执行时被调用,则情况会变得更复杂一些。这时可以采用以下几种方法:

- 1.主程序调用该函数时禁止中断,可以在该函数被调用时用 #pragma disable 语句来实现禁止中断的目的。必须使用 OVERLAY 指令将该函数

从覆盖分析中除去。

- 2.复制两份该函数的代码,一份到主程序中,另一份复制到中断服务程序中。

- 3.将该函数设为重入型。例如:

```
void myfunc(void) reentrant {  
    ...  
}
```

这种设置将会产生一个可重入堆栈,该堆栈被用于存储函数值和局部变量,用这种方法时重入堆栈必须在 STARTUP.A51 文件中配置。

这种方法消耗更多的 RAM 并会降低重入函数的执行速度。

2.第二种错误信息

```
*** WARNING L16: UNCALLED SEGMENT, IGNORED FOR OVERLAY PROCESS
```

```
SEGMENT: ?PR?_COMPARE?TESTLCD
```

说明: 程序中有些函数(或片段)以前(调试过程中)从未被调用过,或者根本没有调用它的语句。

这条警告信息前应该还有一条信息指示出是哪个函数导致了这一问题。只要做点简单的调整就可以。不理它也没什么大不了的。

解决方法:去掉 COMPARE()函数或利用条件编译 #if#endif,可保留该函数并不编译。