利用兼容机主板实现工业控制机的方法

胡建人*
（杭州电子工业学院文理分院 杭州 310037）

【摘要】 讨论了兼容机主板上的BIOS程序结构和改制方法、实际操作及其可行性，证明了DRAM直接运行工业控制监控程序和应用程序可达到采用SRAM的一般工业控制机的抗干扰水平，采用高质量的机箱、电源和合适的兼容机主板制作的工控机可满足工业生产现场的恶劣运行条件。通过软盘直接更改监控程序和各种工控参数、加工对象。

关 键 词 基本输入输出系统；计算机主板；工业计算机/工业控制机；微机；监控程序；上电自检；CMOS参数

中图分类号 TP36；TP273

普通微机用于工业设备的自动化控制，具有扩充能力强、成本低、软件成熟、支持软件众多诸优点。经过改造的微机，可以不带键盘、显示器以及其他外设，成为前端机或专用工控机。普通的IBM兼容机的各项性能很大程度上依赖于固化在ROM上的BIOS。微机开机后BIOS要完成一系列的设备自检工作。兼容机改作工控机后，要对键盘、显示器、驱动器等重新配置，去掉多余的外设。经过修改BIOS程序并对设备重新配置后，可使普通微机成为无键盘、无DOS操作系统的专用电脑控制器，从而为普通微机大规模工业控制应用铺平了道路。

1 工作原理和方法

BIOS（基本输入输出系统）作为微机底层硬件管理软件，是用机器语言写成的，也是改变微机用途的必经途径。

1.1 累加和与设备变动的关系

正常的微机启动时首先通过累加和检查64Kb的BIOS是否正确，还要根据CMOS中的配置信息逐个检查外设和内存是否正常[1, 2]。任何不正常的检查结果都会导致系统停机，这是微机用作工控机首先要解决的问题。

累加和检查有两种解决方法：1）跳过检查；2）直接给出正确的累加和的值。为简化处理，我们选用了第一种方法，在BIOS的上电自检(POST)过程中直接跳过不执行关于累加和检查的一段程序。这样，不管怎样改动BIOS中的内容，都不会出现因加电自检累加和不等于0而停机的情况，为工控机的程序编制铺平了道路。

1.2 CMOS参数的保存与恢复

1.2.1 造成CMOS信息丢失的原因

微机主板在安装多功能卡或其他插卡时很容易造成CMOS放电而引起CMOS储存的设备信息丢失。外界干扰、反复开关微机电源、电池自然放电、计算机程序运行紊乱和软件冲突等原因，都可能造成CMOS掉电或被改写而失去CMOS配置信息。

1.2.2 固化CMOS参数

不带键盘和显示器的专用工业控制装置，一旦发生CMOS信息丢失，操作人员将无法通过键盘恢复正确的CMOS值。工控机的设计人员和操作人员都希望尽量简化操作程序，减少人为差错，提高设备自纠错能力。为保证开机后机器能直接进入正常的工作状态，特在微机加电自检程序执行前增加了一段CMOS信息恢复程序，每次开机或RESET后，都把固化在BIOS中的CMOS设置参

1998年6月29日收稿.1998年8月3日修改定稿

* 男 45岁 学士 副教授

数重新写入 CMOS 芯片，然后再进入自检程序，从而保证了每次运行的 CMOS 参数初始值都是正确的，可避免 CMOS 参数改变后的尴尬场面。即使出现了诸如运行中的 CMOS 参数改变而停机的情况，只要按下 RESET 键重新运行工控机的初始化程序即可恢复。

1.3 监控程序及其引导

普通微机 BIOS 在自检完成后立即进入 INT 19H 启动中断程序，自动引导 DOS 等操作系统进入内存，并把控制权交给被引导入的操作系统。专用工控机在 BIOS 完成各项自检和设置后应立即进入监控程序，并使设备处于工控机的状态。为实现这一目标，可以修改 BIOS 中的 INT 19H 启动中断程序，把指针指向监控程序的入口，让启动中断调用固化在 BIOS 芯片中或在磁盘上的监控程序并把控制权交给监控程序。经过处理过的 BIOS 芯片和改装后的微机主板就成为具有特定控制能力的专用工控机。

2 BIOS 的实际修改过程

兼容机用作工控机的关键是破译 BIOS 程序，只有打通 BIOS 才能进入操纵控制。

2.1 修改 BIOS 的整体步骤

要获得实用的工控机 BIOS 可以分为如下几步：

1) 读出 BIOS 的内容　首先用仿真器将微机主板 BIOS 芯片的内容读出来，得到真实的 BIOS 内容，并反汇编出汇编语言；

2) 分析 BIOS 程序　根据反汇编分析程序流向，确定各相关出入口；

3) 修改 BIOS 程序　根据预定的目标，用 DEBUG 修改相应地址的程序内容；

4) 程序仿真　把修改后的 BIOS 内容写入 EPROM 中，并插入微机主板 BIOS 插槽中试运行。重复步骤 3)、4)，直到系统能稳定的工作为止；

2.2 BIOS 基本情况和 CMOS 读写方法

2.2.1 BIOS 的基本情况

微机加电启动(复位)，CPU 首先将 CS 段寄存器设置成 FFFFH，清零所有其他寄存器，然后执行 CS：IP（FFFF:0000 地址也就是 BIOS 地址 F000：FFF0）处指令进入自检。加电入口地址 F000：FFF0 存放一条 JMP F000：E05B 指令，从 286 开始的微机主板 BIOS 中 F000：E05B 处存放着一条 JMP 指令转跳到真正的自检程序入口，从该入口处 BIOS 开始进行一系列的自检，完成自检后，执行 INT 19H 启动中断程序，调入操作系统并把控制权交给该操作系统。

BIOS 自检一般从累加和开始，该自检程序应在离入口不远处，正确的累加和值应等于 0。

2.2.2 CMOS 读写方法

从 F000：FEF3 地址开始的 64 个字节是从 INT 8H 开始的中断向量地址表，其中 F000：FF15 开始的两个字节的内容是 INT 19H 中断入口地址。

CMOS 共有 64 个字节常规内容，有些主板则共有 128 个字节内容，其读写在口地址为 70H、71H 的两个输入输出口上。70H 口是 CMOS 的地址寄存器，71H 口是数据寄存器。要读一个 CMOS 字节内容，例如 10H 字节内容，首先向 70H 口输出值 10H，然后从 71H 口读出数据就成了；要写 CMOS 字节，则首先向 70H 口输出值如 10H，然后再向 71H 口输出要写的值，就完成一个 CMOS 字节的读写。

2.3 BIOS 操作概要

由于微机的实际情况各不相同，对 BIOS 的修改也各不相同。实践中发现，有些版本的 BIOS 容易破译，而有些版本的 BIOS 相当难破译。随着 BIOS 版本的不断改进和升级，BIOS 芯片中的程序越来越大，可供使用的空白程序区越来越小，给破译和改写 BIOS 原有程序带来了更大的难度。尤其是一些版本的 BIOS 程序是在原有的程序的基础上改写的，其内部结构复杂，程序块比较零乱，块与块之间缺乏逻辑联系，并多次使用转跳语句，程序逻辑性差，使得汇编 / 反汇编语句的可读性

下降，程序流向和程序出口不甚明了。为此，我们选择可读性好的BIOS芯片的主板，在开发BIOS程序之前，初步读出BIOS程序和确定程序流向、程序出入口以确定该BIOS的开发难度。对于难以破译的芯片应及时舍弃，另寻合适的芯片继续开发。在确定开发目标时，应兼顾被开发的主板的来源。在确定被开发的主板时，一定要有超前意识。应预测被开发的主板的流行周期，以保证开发成功后，仍然有足够的主板货源。特别要防止出现开发成功后，该型主板退出市场的情况发生。

2.4 特定版本BIOS操作实例

用于每种BIOS的实际程序多少存在一些差异，下面以日期为08/08/93的(C)1993 AMI版的BIOS为例，具体说明修改过程。

1）去除累加和检查　从F000：E05B处找到自检入口地址F000:D644H，向下找出如下所列的程序段（已标上找到的地址）：

```
D7C5    XOR  BX,BX
D7C7    MOV  CX,8000
D7CA    XOR  SI,SI
D7CC    CS:
D7CD    LODSW
D7CE    ADD  BX,AX    ; BX 中为累加和值
D7D0    LOOP D7CC
D7D2    JNZ  D7DA
```

将程序最后一行D7D2处指令改为无条件转移指令JMP D7DA，不检查累加和值就直接进入下一步的工作。

2）CMOS内容刷新方法　首先将待开发的主板组装成一台完整的微机，根据实际使用的工控机的外设配置设置好CMOS值，然后编写如下程序段读取CMOS各字节值：

```
        MOV    CX,30H          ;  共读 48 个字节
        MOV    AX,1000H        ;  从 10H 字节开始读
L1:     CALL   READONEBYTE
        CALL   DISPLAYAX       ;  显示 AX 寄存器值，程序略
        INC    AH
        LOOP   L1
        RET
READONEBYTE   PROC
        XCHG   AH,AL
        OUT    70H,AL
        XCHG   AH,AL
        JMP    N1
N1:     IN   AL,71H
        RET
READONEBYTE   ENDP
```

编写一段写CMOS值的程序，流程图见图1（实际程序较长，不予例出），放入原BIOS芯片中首地址为DB40H的空白区。

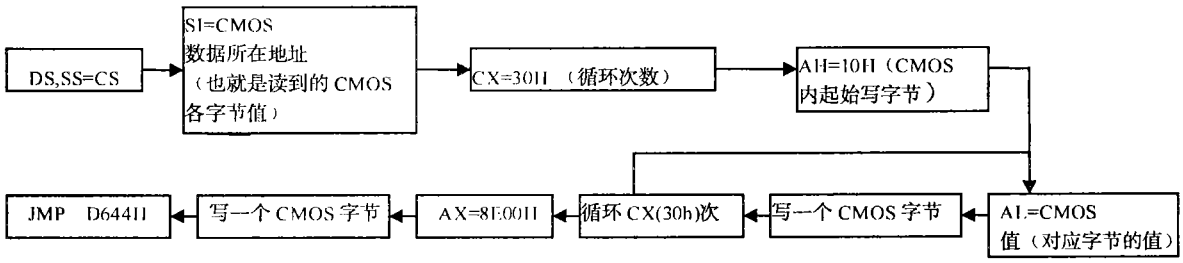

图 1 写 CMOS 的程序流程图

将 F000：E05B 处指令改为 JMP DB40，这样每次加电自检时先执行 DB40 处的写 CMOS 程序，然后再跳回到 D644 进行自检。这样改进后的主板不需要 CMOS 专用电池即能通过自动恢复 CMOS 参数程序配置 CMOS 参数。

3）BIOS 加密版本的脱密和修改过程 每种 BIOS 都有其不同的特点，对不同的 BIOS 版本，需要具体研究具体对待。有些 BIOS 的程序采用了压缩甚至加密。根据加密后的 BASIC 应用程序在 BASIC 状态运行时呈解密状态，只要设法中断程序运行，并将内存中的 BASIC 程序存盘就得到已经解密的程序，并可以直接运行该已解密的程序的思路，解剖了带加密的 BIOS。以 10/10/94 版 AMI BIOS 为例，通过读出该 BIOS 芯片的静态内容与实际运行时存放在机器内 F000 段的内容比较，发现两者的代码不同。研究发现该 BIOS 在上电自检(POST)时，首先把芯片中的加密代码解密后移入映像成的 F000 段，然后再执行 F000：FFF0 处的指令自检。解决的方案是顺其自然，让机器自动完成 BIOS 解密工作，解密结束后立即执行一条新插入的指令(FEDC:0972 CALL 2000:ED80)，并把 CMOS 自动恢复程序、新的 INT 19H 和监控程序移入对应地址；修改 3000H 开始放置的解密程序段中的 E05B、E6F2 处的跳转地址，而 2000H 开始的程序段仍然放置加密内容。

2.5 监控程序

1）监控程序放置于 BIOS 芯片中 一般工控机的监控程序不大，可以直接放入 BIOS 芯片中的空白区。从地址 F000:FF15 找到 INT 19H 中断入口地址 E6F2，将 E6F2 处指令改为 JMP DC50（BIOS 芯片空白区），并在 DC50 处放一段把 BIOS 中的监控程序搬到 RAM 地址 0900：0100 处，并将指针转移到该地址即完成了工控机的初始化。

2）监控程序放在磁盘中 监控程序也可以放在磁盘中，但程序必须放在磁盘的第一个数据扇区开始的连续扇区中（与 DOS 的 IO.SYS 相同，方法是格式化磁盘，然后立即拷贝入监控程序），监控程序必须是 COM 文件格式，而 INT 19H 处的程序仍然采用原 BIOS 中的 LOADBOOTSECT，装载有效的引导扇区，实现系统引导[4]，这样机器启动后就会自动进入监控程序。

3）监控程序存放形式比较 监控程序调入内存运行的好处是编程容易，通过 CMOS 的 SHADOW 设置可以使微机运行速度快。RAM 运行监控程序，尤其是磁盘存放监控程序，可以用软件编制的方式直接调试、运行新的软件，改变不同的控制对象，使工业控制变得容易，而监控程序存放在 ROM BIOS 中的缺点是不能随意改动监控程序。两种方法的数据都可以通过磁盘收集和交换。例如，纺织行业的提花机，运用磁盘文件改变数据可以实现不同的织物共享同一监控软件，运用临时编制的磁盘数据文件方便的改变生产，以适应当今世界流行的生产模式：批量小花式品种多的特点。

4）RAM 中运行监控程序的可靠性 由于微机采用开关电源，电源中包含了共模干扰抑止器，抗干扰性较好；DRAM 本身的稳定性尚好。只要工控机的 I/O 控制口作好充分的隔离，电网没有瞬态断电，工业控制机使用 RAM 作监控程序工作区的实际效果好[5]，已经能满足实际需要。

3 结 论

微机主板改作工业控制器，为适应工作环境的变化，修改 BIOS 是必要的。经过特殊改造的微机仍然可以利用软盘、硬盘、显示器等其他外设，也可以用汇编语言或 C 语言（不能有 DOS 功能调用）编写监控程序，其优点十分明显。这种方法使大规模工业自动化改造降低了成本，提高了性能价格比。

参 考 文 献

1 为 林、维 钢. 8088·286·386 维修调试工具与 286BIOS 分析注解. 北京: 北京希望电脑公司, 1993

2 李新育. IBM PC AT 286 微型机 BIOS 接口技术参考手册. 北京: 北京希望电脑公司, 1991

3 王兆全. 微型计算机—INTEL8086 基础. 北京: 人民邮电出版社, 1986

4 张怀莲. MS-DOS 3.30 STD 系统 BIOS 分析应用与虚盘管理. 北京: 电子工业出版社, 1991

5 胡建人. HMOS 微机超低电源电压运行技术. 微电子学与计算机, 1993, 10(5): 31~32, 48

6 于春凡 朱耀庭. IBM-PC 及长城 0520(INTEL8088/8086)宏汇编语言程序设计. 天津: 南开大学出版社, 1990

Method of Making Industrial Computer with Microcomputers Mother Board

Hu Jianren

(School of Science and Art, Hangzhou Institute of Electronics Engineering, Hangzhou 310037)

Abstract This paper discusses the structure of BIOS programs. The programming and operation with BIOS are also introduced. It is proved that SRAM as reliable as DRAM in the industrial controller. It is good for anti-industrial interference with good quality of the electric power and the suitable mainboard with its special BIOS. It is easy to change the program and control parameters through disk with monitor. So, special industrial computers can be made up of common personal computers' mother boards.

Key words BIOS; industrial computer/industial controller; motherboard; microcomputer; monitor program; power-on self-checking; CMOS parameter