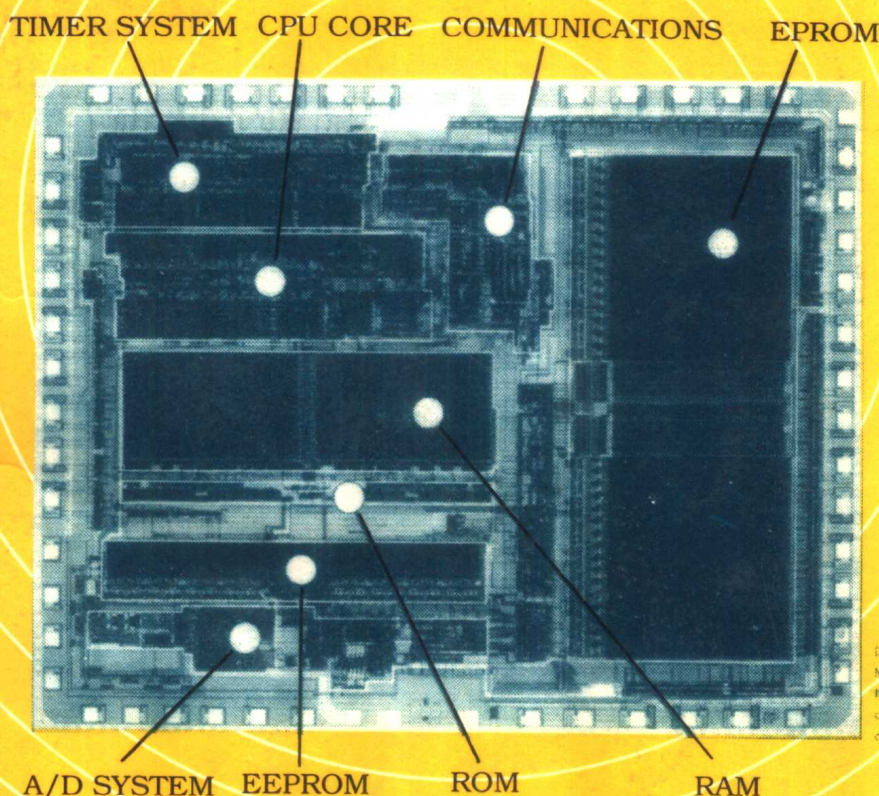


MC 68HC 11

单片机开发技术

邵贝贝 程建平 编著



清华大学出版社

MC68HC11 单片机开发技术

邵贝贝 程建平

清华大学出版社

(京)新登字 158 号

内 容 简 介

这是为大学单片机课程编写的教材。书中介绍了如何从最小硬件系统开始逐步设计一个单片机开发系统,讲述如何编写单片机监控程序,从而建立单片机应用软件的开发环境,介绍了如何用 C 语言编写系统应用程序。单片机应用方面的实验可在与本书配套的 TSB-1 实验与开发板上进行,也可由读者自行安排。

本书为培养学生查阅英文技术资料的能力,特征得 MOTOROLA 公司的授权,将 M68HC11A1 单片机的原文资料作为本书的附录。

本书也可供开发单片机应用系统的工程技术人员参考。

© 版权所有,翻印必究。本书所摘引的图稿和英文资料均获得 MOTOROLA 公司的使用许可。本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

MC68HC11 单片机开发技术/邵贝贝,程建平著. —北京:清华大学出版社,1994
ISBN 7-302-01703-4

I. M… II. ①邵…②程… III. 单片式计算机, MC68HC 11-开发系统-技术
IV. TP368.1

中国版本图书馆 CIP 数据核字(94)第 14490 号

出版者:清华大学出版社(北京清华大学校内,邮编 100084)

印刷者:顺义振华印刷厂

发行者:新华书店总店北京科技发行所

开 本:787×1092 1/16 印张:8.75 字数:202 千字

版 次:1995 年 1 月第 1 版 1995 年 1 月第 1 次印刷

书 号:ISBN 7-302-01703-4/TP·742

印 数:0001—5000

定 价:8.00 元

前 言

1990年,在北京图书馆国外新书阅览室,我见到J. Peatman写的“DESIGN WITH MICROCONTROLLERS”。该书介绍了MOTOROLA单片机68HC11,并与INTEL 8096作了比较。同年年底,我们买到几片68HC11A1,开始编写68HC11的交叉汇编程序,并利用这个交叉汇编工具写成68HC11的监控程序,初步建立了68HC11单片机的基本开发环境,1991年初又进而建立了交叉C语言编译支持的开发环境。设计了68HC11开发系统。在IBM-PC机上生成的交叉汇编程序或C语言程序,可以下装到开发系统中用监控程序进行程序的调试。应用这套开发技术,1991年间,我们在国内开发了多个工程应用项目,有的还出口到南韩等国。

1991、1992年间,我在美国参与一个工程项目设计,也使用了MC68系列的单片机。并对MOTOROLA单片机系列是世界单片机的主流系列有了更深的印象。

1994年回到国内后,我从长期的科研工作转向计算机方面的教学工作,感到国内单片机的教学应当符合国际上的潮流,从宏观上应把重点从INTEL8031, 8051转到MC68系列上来。

从应用与开发的角度看,国外进口的单片机开发系统价格昂贵,一般单位买不起。根据多年的工作经验,我认为68HC11的开发并不需要昂贵的仿真系统。为此,我们设计了廉价的、以教学为主要目的的68HC11 TSB-1实验与开发系统,它既可以用于教学,也可以用于应用开发,配合本教材一起使用。

我们高兴地看到,最近一、两年,国内相继出版了一些关于68HC11单片机的教材与参考书。这些书的构思多倾向于把国外的手册翻译成中文,并加上作者的一些理解和解释,这对读者加速理解单片机无疑是很有帮助的。这类书讲究完整性,往往面面俱到,书很厚,有三、四百页之多,读者不易抓住要点,而翻译出版过程中又有不少错误,迫使读者真到用的时候非查原文不可。MOTOROLA 68MC11A1在器件手册上只有三、四十页。里面包含了开发68HC11所需要的全部信息。从我长期科研经历中接触到的大学毕业生的弱点看,一些名牌大学电子学与计算机类刚毕业的大学生看原文的器件手册还有困难,或不习惯。而所谓三看懂,即看得懂芯片手册,看得懂硬件原理图,看得懂程序源码是对从事电子学与计算机工作毕业生的起码要求。所以我们宁愿把原文的芯片手册作为本书的附录。一来有助于学生学会看懂手册,二来保证信息的准确性和使我们的教材写得尽量精炼。

本教材的思路在于利用笔者自己多年开发单片机产品的经验,教给学生如何利用最基本的资料,例如三、四十页纸的芯片手册,将从未用过的一种单片机搞活,直到运用自如。俗话说,赠与一条鱼,不如授与如何钓鱼。我们希望读者能以开发68HC11单片机为例,学会一种解决问题的思路和方法。

本教材的重点在于如何尽快使单片机与开发者“沟通”,即可以互相“对话”,如何建

立最基本的开发环境,如何处理好建立开发手段与开发应用程序的关系,以及如何用这套思想开发更高档的,更低档,或更先进的单片机及系统。总之引导学生尽快入门。书中所有的例题都是我们过去在工程项目的实际应用程序中用到的程序段落。对单片机各种 I/O 端口的使用,只做简单介绍与引导,而在实验与开发板上为各种 I/O 接口实验提供条件。将广泛的应用前景留给读者去发挥他们丰富的想象力和创造力。

本书的后半部分,在 TSB-1 实验与开发系统上的实验设计,是由程建平同志完成的。他在微机教学与实验方面有多年的经验,设计过几种微型机、单片机的教学实验系统。我们希望多年科研与教学经验的结合能把实实在在的经验与体会奉献给读者,也对大学教材的构思做一点儿探索。

国外集成电路从诞生到广泛用于产品要好几年。从国内用上了,到老师理解了,翻译成中文,出版教材,讲给学生又是好几年。而电子产品的更新换代周期一般只有五年。我们希望把如何去发现和开发刚刚问世的一代单片机的思路与方法教给学生。

本书近于脱稿时,我们荣幸地得到了 MOTOROLA 公司的支持与帮助。该公司向我校赠送了 68HC11 的芯片、资料及教学用评估板,并书面许可我们在本书中使用有关图稿和英文技术资料。在此,我们向 MOTOROLA 公司表示诚挚的感谢。

参与本工作的还有刘慧银、张燕冰等同志,在此表示衷心感谢!

邵贝贝 1994 年 8 月于清华

目 录

第一章 单片机概念与 MOTOROLA 单片机系列	1
1.1 单片机概念	1
1.2 MOTOROLA 单片机系列	1
1.3 关于单片机 68HC11	2
第二章 MC68HC11A1 的结构	5
2.1 单片机外观	5
2.2 工作模式	5
2.3 引脚	6
2.4 68HC11 的 CPU 结构	7
第三章 最基本的硬件系统	9
3.1 串行接口	9
3.2 最小系统设计	10
第四章 MC68HC11 指令集与寻址方式	13
4.1 指令按功能分类	13
4.2 指令按寻址方式分类	13
4.3 指令的机器码组织	15
第五章 内存空间分配	24
5.1 68HC11A1 内部的内存空间	24
5.2 可扩展的存储空间	24
5.3 译码电路设计	25
5.4 存储空间的进一步扩展	26
第六章 汇编语言程序设计举例	27
6.1 最基本的 I/O 子程序	27
6.1.1 检查 RS-232 输入端口状态	27
6.1.2 输入一个字符	27
6.1.3 输出一个字符	28
6.1.4 输出空格	28
6.1.5 显示字符串	28
6.1.6 输入并显示字符	29
6.2 码的转换类子程序	29
6.2.1 输入一个 16 进制数	29
6.2.2 输入一个字节	30
6.2.3 输入两个字节的 16 进制数	30

6.2.4	输出两个字节的 16 进制数	30
第七章	复位与中断	32
7.1	复位	32
7.1.1	加电复位和外部 RESET 引脚置低复位	32
7.1.2	计算机工作正常(COP)看门狗定时复位	33
7.1.3	时钟监控电路产生的复位	34
7.2	中断类型与中断处理能力	34
7.2.1	软件中断	34
7.2.2	非屏蔽中断(XIRQ)	34
7.2.3	可屏蔽中断	35
7.3	怎样写中断服务子程序	37
第八章	单片机监控程序设计	39
8.1	监控程序	39
8.2	建立 CPU 在内存中的映象	40
8.3	执行程序	41
8.4	显示与修改内存	42
8.5	程序的向下装入(Download)	43
8.6	设置断点	44
第九章	用汇编语言编程序	45
9.1	怎样使用交叉汇编程序	45
9.2	汇编语言编程技巧	45
9.3	TSB-1 开发系统监控程序的框架	47
第十章	怎样用 C 语言开发 68HC11 单片机系统应用程序	55
10.1	C 语言是开发单片机应用系统软件的有效工具	55
10.2	交叉编译和 C 语言程序运行环境的建立	56
10.3	应用程序的构成与模块化程序结构	58
10.4	全程变量与局部变量	58
10.5	函数的结构与函数间参数的传递	59
10.6	C 语言中的 I/O 语句	61
10.7	程序模块的框架与组织	62
第十一章	I/O 接口与应用	65
11.1	从应用角度看 68HC11A1	65
11.2	与定时有关的 I/O	66
11.3	并行 I/O 接口	67
11.4	D 口中的同步串行口	68
11.5	A/D 变换端口	69
第十二章	TSB-1 实验与开发系统	70
12.1	TSB-1 型实验与开发系统简介	70

12.1.1	M68HC11 TSB-1 的主要技术特性	71
12.1.2	M68HC11 TSB-1 的主要功能	71
12.2	TSB-1 型实验与开发系统的电路介绍	72
12.2.1	TSB-1 型实验与开发系统实验板的组成	72
12.2.2	TSB-1 实验及扩展线路	74
12.3	TSB-1 型实验与开发系统的使用	80
12.3.1	安装	80
12.3.2	PC 与 TSB-1 实验板通信	80
12.3.3	监控程序中的命令	81
第十三章	MC68HC11 单片机开发与应用实验	84
13.1	实验一、监控命令练习	84
13.2	实验二、交叉汇编程序的使用	85
13.3	实验三、并行 I/O 实验	86
13.4	实验四、中断实验	86
13.5	实验五、模数转换(A/D)实验	87
13.6	实验六、数模转换(D/A)实验	87
13.7	实验七、定时器系统实验	88
13.8	实验八、串行外设接口(SPI)实验	88
13.9	实验九、步进电机驱动与可编程定时器使用实验	89
13.10	实验十、电子音响控制实验	90
附录	MC68HC11A1 Technical Data	91

第 一 章

单片机概念与 MOTOROLA 单片机系列

1.1 单片机概念

单片机(Single Chip Computer),字面上的意思已很明显,一块集成电路芯片本身就是一部计算机。通常,计算机的概念包括中央处理单元(CPU)、存储器和输入输出接口三部分内容。用现代超大规模集成电路技术将以上三部分集成在一起就构成了单片机。单片机成本低,体积小,运行可靠,I/O 功能丰富,选择性强,特别适用于自动控制与智能装置,所以也称作微控制器(Microcontroller)。正是由于单片机具有这样一些特点,世界上单片机的产量已达数十亿片。从家用电器到工业控制,从民用到军工,都离不开单片机。

1.2 MOTOROLA 单片机系列

选择 MOTOROLA 单片机系列作为教材,主要是因为该公司生产的单片机在世界上占主导地位,无论从品种上还是数量上都领先于其它公司。MOTOROLA 单片机能成为世界上单片机的主流有一个很主要的原因是,该公司的单片机产品成系列,包括 4 位机、8 位机、16 位机及 32 位机。而每个档次的单片机又都有多种不同型号可供选择。目前看来,世界上最流行的 MOTOROLA 单片机有三大类,分属于低、中、高三个不同档次。

68HC05 系列属低档 8 位机,有 80 多个品种。68HC05 单片机是真正的单片机,一般只能工作在纯单片机方式下。开发者的任务是把控制对象接好,用汇编语言做软件,然后把程序写到单片机内部的只读存储器(ROM)中去。软件设计一般不大可能一次成功,除非有很好的仿真环境和开发工具。一般在研制样机阶段使用含有可擦除只读存储器的 EPROM 类型的单片机。小批量生产时可用一次性写入的 ROM 型单片机。大批量生产时则交工厂生产掩模型单片机产品,由工厂在制做芯片时写入程序。

68HC11 属中档单片机,是增强型 8 位机,即外部数据总线是 8 位的,内部为 16 位总线。CPU 功能比 68HC05 的 CPU 强很多。68HC11 可以工作在单片方式,也可以工作在扩展方式。68HC11 的扩展工作方式允许用户把存储器扩展到芯片外面来。在系统调试阶段,程序可以放在芯片外面的存储器中。这就为程序的调试带来许多方便之处。系统调试完成之后,最终产品还可以回到单片机工作方式中去。68HC11 有 40 多个品种可供选用。

683xx 系列是 32 位单片机系列。它采用与 MC68000/MC68020 用户级指令兼容的 CPU68000 或 CPU32,配上各种类型的 I/O 单元及不同容量的存储器。MC68000 是设计

上很成功的一种CPU,在世界上用得颇为广泛,现成的软件非常丰富。MOTOROLA 已将CPU68000 作成专用集成电路(ASIC)设计的一个标准库元件,加上其它 I/O 功能的标准库元件,用户可以自行设计所需要的单片机,由该公司为您生产。683xx 系列中,6830x 系列用的是CPU68000,6833x 用的是CPU32。这是一个功能极强前景很好的单片机系列。

在三大主流系列之间还有 68HC08,68HC16 等,我们不在这里介绍。

1.3 关于单片机 68HC11

选择 68HC11 作为教材的原因是清楚的,即它的档次适中,易于开发。而 68HC11 的开发过程与思路原则上也适用于低档的 68HC05 系列、68HC08 系列或高档的 68HC16 系列、68300 系列。

读者也许会问,既然 MOTOROLA 68 系列在世界上是主流,为什么在中国却显得陌生,不象 INTEL 公司 8031,8051,8096 等那样流行?而直到最近一两年,国内才逐渐对 MC68 系列单片机有所认识。简要分析一下其中的原因或许会拓宽我们的思路。

原因之一可能是由于我国过去对外交流不够,我们的技术人员往往习惯于市场上有什么就用什么,不大注意分析世界上的流行产品与技术动向。其次就是存在一个设计方法的问题。

在分析后者之前,我们给出正向设计与逆向设计的概念。正向设计指经过充分技术调研,以我为主地实现自己的产品。而逆向设计指解剖和分析别人的产品消化理解后仿制。逆向设计产品一般要落后正向设计产品数年。我们主张以正向设计为主,注意研究国际上的新技术发展动向,不要等到数年后打开国外某产品,才发现主流在哪里。对于自己用,主流与否倒也无妨,但以教材授之于人,则应该交待清楚。

为了给熟悉 INTEL 单片机的读者一些更形象的概念,我们就 MC68 系列单片机与 INTEL 系列单片机做归类比较。

MC68HC05 的 CPU 功能大约在 8048 与 8051 之间。MC68HC11 与 8096 属同一档次。再高档次的单片机,例如与 386 兼容的,目前 INTEL 公司还没有。

68HC11 大致相当于 INTEL8096。区别于:8096CPU 内部地址总线为 8 位,数据线为 16 位,68HC11 芯片内部数据线 8 位,地址线 16 位。8096 内部没有 EEPROM(可用电信号擦除的只读存储器)而 68HC11 内部一般有 0.5 KB 的 EEPROM。

和 INTEL 公司单片机相比,MOTOROLA 单片机有以下一些特点:

MC68 系列微机采用 Memory Maped I/O,即不区分是存储器空间还是 I/O 空间。故对 I/O 的操作采用与存储器相同的寻址方式与时序。因此 CPU 指令与寻址方式的对应关系整齐,用起来很方便。

MC68HC11 类单片机中许多都有 0.5K 字节 EEPROM。这是 INTEL 系列单片机中不多见的。EEPROM 除用于保存系统参数外还用于对应用程序加密,这是其它类型 ROM 代替不了的。

从时序上看,68HC11 单片机的总线速度指 E 信号的频率,E 为高电平表示总线的读/写操作;E 为低电平,是 CPU 内部执行指令的操作。E 的频率就是读写频率。而 INTEL

系列单片机,总线速度是 clock 信号的速度,一次读/写一般要 4 个 clock 周期。因此,当两种单片机工作在同一速度下,MC68 系列单片机在总线上的频率只有 INTEL 系列单片机的四分之一。因而,对外围芯片的速度要求相对低些,外部总线上信号显得“干净”,抗干扰能力相对强些。

表 1-1 列出了 68HC11 单片机的主要型号与功能。

表 1-1 部分 68HC11 单片机型号及性能

Device	RAM	ROM	EEPROM	Comments
68HC11A0	256	—	—	
68HC11A1	256	—	512	
68HC11A8	256	8K	512	
68HC11C0	256	—	512	256K Extern Memory, 6 Chip Select
68HC11D0	192	—	—	No A/D
68HC11D3	192	4K	—	No A/D
68HC11E0	512	—	—	
68HC11E1	512	—	512	
68HC11E8	512	12K	—	
68HC11E9	512	12K	512	
68HC11E20	768	20K	512	
68HC11F1	1024	—	512	10bits A/D
68HC11G0	512	—	—	Nomux bus, 10bits A/D, 66 I/O pins
68HC11G5	512	16K	—	
68HC11G0	512	24K	—	
68HC11J6	512	16K	—	Nomux bus, 10bits A/D, 1 Chip Select
68HC11K0	768	—	—	Nomux bus, Mem Expansion 1M, 4 CS
68HC11K3	768	24K	—	
68HC11K4	768	24K	640	
68HC11K0	768	—	—	
68HC11KA1	768	—	640	Nomux bus, 4ch PWM
68HC11KA2	1024	32K	640	
68HC11KA3	768	24K	—	
68HC11KA4	768	24K	640	
68HC11KL0	512	—	—	Multiplex Bus
68HC11KL1	512	—	512	

续表

Device	RAM	ROM	EEPROM	Comments
68HC11KL5	512	16K	—	
68HC11KL6	512	16K	512	
68HC11KM2	1280	32K	640	Math Coprocessor, DMA
68HC11KN4	768	24K	640	12ch A/D, 2ch D/A, Math Coprocessor
68HC11KP4	1024	32K	640	3 SCI, 62 I/O

归纳起来,ROM 或 EPROM 可为 0 到 32K,以 4K 为增量,RAM 一般为 0.25 到 1.25K,EEPROM 多为 0.5K。它们都有定时器和串口,有的还有两个或三个串口,输入输出位少的有 14 个,多的有 66 个,一般有 8 路 8 位 A/D,个别有 10 位 A/D 或没有 A/D。

ROM 有 EPROM 型的,也有一次写入型的。其中 MC68HC711 是 EPROM 型号的,表中没有列出。

I/O 功能的选择及内存大小的选择完全看用户需要。我们选择 68HC11A1 作为教材与开发对象是基于如下考虑:

- (1) 存储器一定要能外接,开发时好处理。
- (2) 片内有 EEPROM 和 8 路 8 位 A/D,具有典型性。

实际上,68HC11A0, 68HC11A1, 68HC11A8 用的是同一种硅片,68HC11A1 中的 ROM、68HC11A0 中的 EEPROM 及 ROM,出厂前已被禁止掉了,不准用户使用,即使将它们“打开”,性能也不保证。

用户可以在开发样机时使用 68HC11A1,在最终产品上也很容易改用其它 I/O 功能更多的或价格更便宜的其它类 68HC11 芯片。

第二章

MC68HC11A1 的结构

2.1 单片机外观

从外观看 MC68HC11A1 单片机就是一片集成电路芯片。它有三种封装形式。一种双列直插的(DIP), 48 条引脚。引脚间间距 100mil(2.54mm), 列间间距 600mil。另一种方型 52 条引脚, PLCC 封装形式。引脚间距 50mil(1.27mm)。这种封装形式的片子引脚向内侧弯曲, 使侧面有弹性, 使用时要加 52 芯集成电路方形插座。比较这两种封装, 前者少 4 条引线。两者的区别在于, 前者的 A/D 变换通道只有 4 路, 而后者有 8 路。此外, 还有一种表面封装式芯片, 有 64 个引脚。多出来的引脚有 10 条是空的, 另外多两个地端。这种封装的片子用于大批量生产的表面焊接工艺的印刷线路板。

可以想象, 这些引脚无非是电源端、接地端、时钟即工作频率输入端、信号输入、输出端等等。问题是显而易见的: 若有一片计算机在手, 如何让它为您工作?

2.2 工作模式

68HC11 有 4 种工作模式。初始工作模式的确定是靠在 MODA 和 MODB 两条引脚上加高低不同电平而实现的。

1. 单片工作模式。

这不乏是一种好的工作方式。问题是如果片内 ROM 中没有合适的应用程序, 单片机就无法工作。故只能是先有了应用程序并写入片子中, 才能使之工作在单片方式下。在我们学习和开发单片机时, 显然不喜欢这种工作方式。

2. 扩展模式。

单片机提供给外界数据总线和地址总线。允许用户在 64K 空间上扩展 RAM 和 EPROM, 扩展 I/O。这给用户开发程序带来了极大的方便。这种方式的缺点是扩展地址和数据总线占用了两个 8 位的 I/O 端口, 而另外扩展出两个 8 位并行口并不困难。当开发工作基本完成后, 也还不妨回到单片方式去, 将两个并口还给系统, 这种工作方式是开发阶段最感兴趣的。

3. 自引导模式。

在自引导模式下, 单片机加电复位后, 可以从串行口向芯片内部的 256 个字节 RAM 中装入一段程序, 并开始执行这段程序。可以想象, 装入的程序不同, 单片机能实现很多不

同的功能。例如对系统作不同的初始化,然后跳转到不同的 ROM 中程序起始地址。这种工作方式也是要先有应用程序在 ROM 中才行。只能等到对 68HC11 应用娴熟时才能用得巧妙。这种模式的使用将留给读者在读完本书后去发挥他们的创造精神。

4. 测试模式。

该模式用于工厂中对产品进行测试。对这种模式,我们也暂不考虑。

确切地说,工作模式的确定是靠 68HC11 复位后锁存 MODA 和 MODB 端的电平状态完成的,工作模式的切换也可用写寄存器指令在程序中进行。

以下,让我们以 68HC11A1 工作在扩展模式下的眼光,看单片机的引脚和内部结构。

2.3 引脚

图 2-1 是 68HC11A1 的内部结构框图。让我们找到那些最重要的引脚,暂时忽略那些

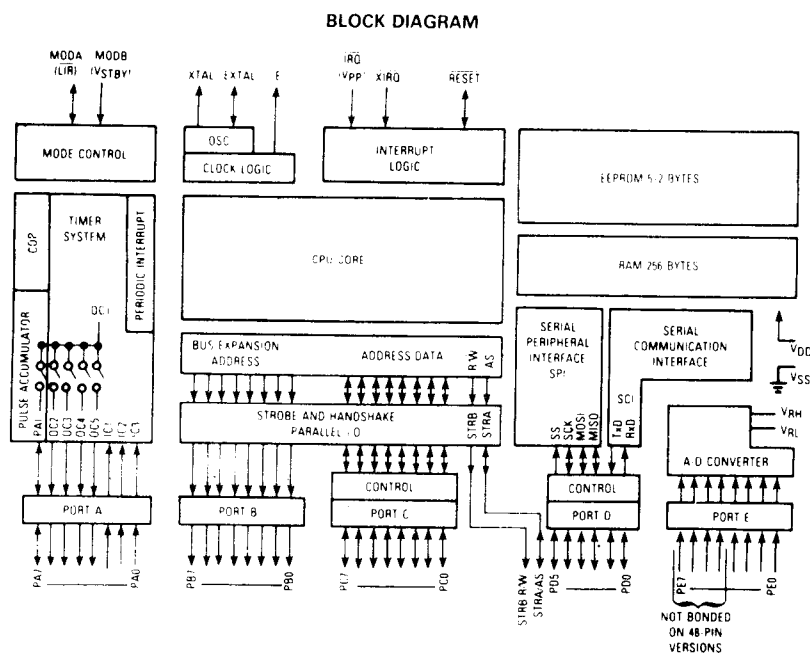


图 2-1 68HC11A1 内部结构框图

无关紧要的引脚。这是看单片机内部结构图,也是看逻辑图的一般方法。以引脚的重要性为序,我们来读 68HC11 结构框图:

1. 电源和地线

VDD 电源,一般接+5V。也有低电源芯片,用+3.3V 供电。

VSS 地线,单片机由这两端得到能量才能工作。

2. 时钟线

EXTAL 和 XTAL 提供单片机的工作节拍时钟。

E 信号 是时钟信号频率的 1/4。高表示外部总线操作,低表示 CPU 内部操作。

3. 数据和地址总线

数据总线口 PC0—PC7 在扩展方式下,数据总线和地址总线低位是复用的。C 口提供数据总线 D0—D7,并提供地址总线的低 8 位 A0—A7。在扩展时,要用一片 8D 锁存器如 74HC373 将地址信号锁存,故地址触发信号 AS 要作为 C 口的控制线用,提供地址锁存信号。此外,从框图看,旁边还有一条 R/W 信号,用于存储器的读/写。地址总线高 8 位由 PB0—PB7 提供。

4. 串行接口 SCI 的两条引线

TXD 数据发送。

RXD 数据接送。

这是单片机中最重要的 I/O 线。只要能把这两条线“搞活”,就能掌握人和单片机交谈的基本手段。

5. 复位和中断线

RESET 给出单片机工作的起点。

IRQ 及 XIRQ 外部中断请求线。用到时再作分析,不用时须加上拉电阻吊高电平。

6. 工作模式选择线

MODA 与 MODB 复位信号锁存这两条引线的电平状态决定单片机的初始工作模式。工作过程中这两条引线不起作用。

以上 6 类引线是单片机 68HC11 的基本引线,是关系到扩展工作方式且和开发手段相关的。而以下引脚只涉及到 I/O 功能,由于这些引脚功能相对独立,作用是局部的,暂时不予考虑。

7. 口 A 与定时/计数有关的 8 条引线 PA0—PA7。

8. 口 E 与 A/D 变换有关的 8 条引线 PE0—PE7。

VRH 与 VRL A/D 变换用的参考电平输入线。

9. SPI 口 与串行外设接口相关的 4 条引线。

这 3 类引线总计有 22 根,暂时忽略这 22 根引脚,68HC11A1 就显得简单多了。

从框图上看,还有几块是没有任何引脚的,那就是 256 字节 RAM、512 字节 EEPROM 及核心 CPU。

2.4 68HC11 的 CPU 结构

68HC11 的 CPU 结构如图 2-2 所示。

68HC11 CPU 有两个 8 位数据寄存器:寄存器 A 和寄存器 B。由于数据寄存器有运算功能,所以也称作累加器 A 和累加器 B。A、B 两个寄存器可以联合起来使用,看成一个 16 位寄存器,此时称作双累加器 D。注意,此时累加器 A 中是 16 位数据的高 8 位, B 中是 16 位数据的低 8 位。

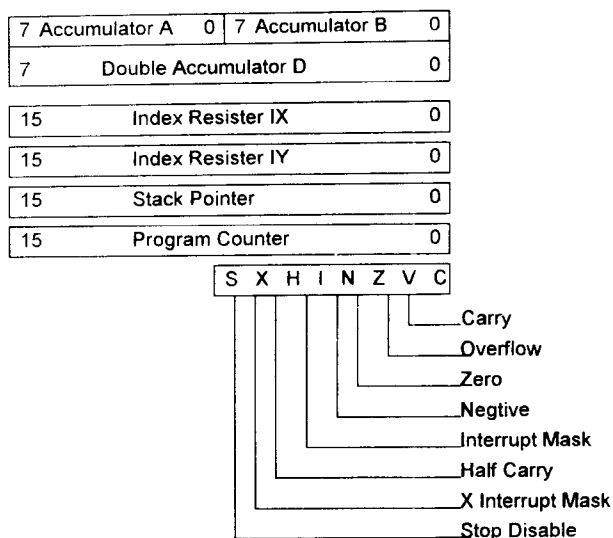


图 2-2 68HC11 CPU 结构

MC68 系列微机高位在前，低位在后，也就是说偶地址是高位，奇地址是低位，和 INTEL 系列正好相反，使用中要注意。

68HC11 CPU 有两个 16 位地址寄存器，称作变址寄存器 IX 和 IY。用以存放地址和实现地址的运算。

一个 16 位栈指针 SP。

一个 16 位程序计数器 PC。

一个 8 位条件码与标志寄存器，该寄存器中 5 个标志位是：

进位与借位标志 C；

溢出标志 V；

结果为零标志 Z；

结果为负数标志 N；

半进位标志 H。

两个中断屏蔽字 I 和 X 用于屏蔽外部中断。S 标志的意义是，当程序遇到 STOP 命令时，是将系统时钟停下来，从而进入省电方式，还是执行 NOP 空操作以等待中断信号来临。以上 3 个条件码是由程序写入的。

注意，我们宁愿称 SP 为栈指针而不叫作堆栈指针。堆和栈是两种不同的数据结构，数据入栈(STACK)，栈指针减一。相应操作在堆(HEAP)结构中，指针是加一的。

与 CPU 结构相关的是 68HC11 CPU 指令集与寻址方式。

第三章

最基本的硬件系统

3.1 串行接口

鉴于串行通信接口在单片机开发中所处的重要地位,我们首先将这部分内容介绍给读者。开发单片机的第一步是要能实现与单片机的对话。与计算机对话最标准、最传统也是最简单的办法是使用 RS-232C 标准的串行口,用计算机终端来实现。一台计算机终端或用 IBM-PC 机仿真一台终端,是开发单片机的基本工具。

有关串行接口 SCI 在附录中有详尽的描述。串行口的工作方式、用法都相当灵活。单片机设计者一定是希望功能越多越好,用法越灵活越好,这反而使初学者有时感到摸不着头脑,不知如何下手。

让我们顺着自己的思路走,从大量信息中拣出我们需要的来。

假如我们手上有一台终端机,通信约定是我们惯用的,例如,8 位数据传送,不校验,一个停止位,4800 波特率。

用过异步串行通信口的读者知道,用查询方式接收、发送来得最简单。发送时,查看发送缓冲寄存器是不是空?即前一个字节是否已送出,是否可以发送下一个字节了?不空就一直查下去,直到为空时发送字符。接收时,查看接收寄存器状态,看是不是满?即有没有接收到什么字符?不满就一直查下去,直到满时把字符读进来。标准的串行接口还有许多应答线,如 RTS,CTS,DTR 等,这些也暂不考虑,硬件上只要有 TXD,RXD 就够了。

单片机串行口和微机串行接口芯片类似,如 INTEL8250, MC68661。它们外面要加与 RS-232C 标准匹配的驱动器件,如 1488, 1489 等。

对 68HC11 单片机串行口,让我们按以下步骤去收集串口信息:

- (1) 找到串口的控制寄存器所在地址,即 SCCR1 与 SCCR2 的地址。
- (2) 想办法初始化成我们所需设置,例如,8 位数据传送,无校验,1 个停止位,4800 波特率。
- (3) 找到状态寄存器(SCSR)所在地址和发送状态位,接收状态位。
- (4) 找到发送、接收数据寄存器(SCDR)的地址。

至此,用 SCI 作人机对话用的信息就够了。关于数据格式,相信用 RS-232 通信,格式都是一样的。用中断方式接收、发送也可暂不考虑,这个实验留给读者在 TSB-1 实验开发系统上自行完成。用中断方式作接收是应用系统中最常用的方法。

串行口的唤醒功能是为 68HC11 单片机工作在省电方式设计的,暂不考虑。在整个应