| 日来 | |
|------------------------|-------------|
| 80C51单片机 | 4 |
| 1 80C51 逻辑结构 | 5 |
| 微型计算机的基本组成 | 6 |
| 80C51单片机基本结构 | 7 |
| 80C51逻辑结构及信号引脚 | 8 |
| 80C51单片机内部结构 | 9 |
| 1) CPU | 9 |
| 控制器 | 9 |
| 运算器 | 15 |
| 2)程序存储器 | 24 |
| 三种型号 | 24 |
| 3)数据存储器 RAM | 25 |
| 4) 并行I/O口 | 25 |
| 5) 串行I/O口 | 26 |
| 6)定时器/计数器 | 26 |
| 7)中断系统 | 27 |
| 8)振荡器电路及元件 | 27 |
| 2 80C51信号引脚 | 28 |
| 80C51的封装 | 28 |
| AT89C51和AT89C2051主要性能表 | 31 |
| 1) 电源和晶振 | 32 |
| 2) I/O 共4个口, 32根I/O线 | 33 |
| 3)控制线:共4根 | 35 |
| 3 80C51时钟电路、工作时序、工作方式 | 37 |
| 时钟电路作用及组成 | 37 |
| 振荡器及定时控制元件 | 38 |
| 内部时钟发生器 | 39 |
| ALE信号 | 39 |
| 时序定时单位 | 40 |
| 1) 振荡周期/节拍P | 41 |
| 2) 时钟周期/状态周期/状态S | 1 41 |
| 3) 机器周期 | 41 |
| | = |

80C51单片机

| 4) 指令周期 | 42 |
|-------------------|------|
| 指令时序 | 43 |
| 单字节、双字节和三字节指令 | 43 |
| 1) 单机器周期指令 | 43 |
| 2) 双机器周期指令 | 44 |
| 访问外部ROM和RAM的时序 | 47 |
| 80C51工作方式 | 49 |
| 80C51复位 | 50 |
| 80C51 程序执行 | 55 |
| 80C51低功耗工作方式 | 56 |
| 80C51 编程工作方式(烧录) | 57 |
| 1) 闪速存储器编程 | 58 |
| 80C51 布尔(位)处理器 | 65 |
| 4 80C51存储器结构与地址空间 | 66 |
| 4个存储器空间 | 66 |
| 三种基本寻址空间 | 67 |
| 80C51程序存储器 | 68 |
| 80C51片內数据存储器 | 70 |
| 片内数据RAM区 | 72 |
| 工作寄存器区 | 72 |
| 位寻址区 | 73 |
| 字节寻址区 | 74 |
| 堆栈区 | 74 |
| 特殊功能寄存器SFR区 | 75 |
| 80C51片外数据存储器 | 78 |
| 5、80C51 总线、接口与扩展 | 79 |
| 80C51系统总线 | 79 |
| 地址总线 | 79 |
| 数据总线 | 80 |
| 控制总线 | 80 |
| 并行输入/输出端口 | 83 |
| P0□ | 83 |
| P1 🛘 | 2 89 |
| P2□ | 97 |

| P3□ | 100 |
|--------------------|-----|
| 80C51存储器扩展 | 102 |
| 扩展ROM | 102 |
| 片选 | 102 |
| 常用的芯片选择方法 | 103 |
| 程序存储器扩展时的总线功能和操作时序 | 108 |
| 扩展RAM | 111 |
| 随机存储器概述 | 111 |
| 片外数据存储器的读、写操作时序 | 112 |
| 数据存储器扩展用典型芯片 | 115 |
| RAM扩展举例 | 116 |
| 译码法扩展 | 119 |
| 综合扩展 | 120 |
| 同时扩展程序存储器和数据存储器 | 120 |
| 扩展既可读又可写的程序存储器 | 121 |

3



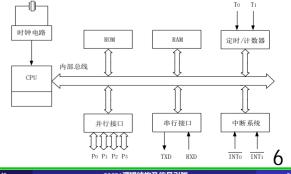
内容提要

| 1 | 80C51 逻辑结构 |
|---|------------------|
| 2 | 80C51 单片机的信号引脚 |
| 3 | 80C51 时钟电路、工作时序、 |
| 4 | 80C51 工作方式 |
| 5 | 80C51 存储器结构与地址空间 |
| 6 | 80C51 总线、接口与扩展 |

80C51逻辑结构

■ 微型计算机的基本组成可概括为三个部分,即中央处理器CPU (通常包括运算器和控制器)+存储器+输入/输出(I/0)接口。

■ 80C51单片机基本结构框图



80C51逻辑结构及信号引脚

- 80C51单片机基本结构
 - (1) 8位微处理器:
 - (2) 数据存储器RAM (128B) 和特殊功能寄存器SFR:
 - (3) 内部程序存储器ROM(4KB):
 - (4) 2个16位**定时/计数器**T0和 T1:
 - (5) **4个**8位**双向**可编程**GPIO**, 并行端口PO[~]P3;
 - (6) 1个UART串行端口:
 - (7) 中断控制系统:
 - (8) 内部时钟电路:

通道0驱动器 通道2驱动器 V_{CC} RAM地址 锁存器 通道0锁 存器 EPROM GND 程序地址寄存器 B寄存器 ACC 堆栈指针SP 缓冲器 TMP, TMP₁ PC递增署 PCON SCON TMOD TCON TH0 TL0 TH1 TL1 SBUF(TX/RX) IE IP 中断、串行口和定时器逻辑 程序计数器PC DPTR指针 通道3锁存器 通道1锁存器

80C51逻辑结构(续)

■ 80C51单片机内部结构

jf x

- ◆1) CPU主要包括控制器、运算器和工作寄存器及时序电路
 - (1) 控制器电路: 识别指令, 并根据指令性质控制计算 机各组成部件进行工作的部件, 与运算器一起构成中央 处理器。在80C51单片机中,控制器包括:
 - ●程序计数器PC(Program Counter)
 - ●程序地址寄存器
 - ●指令寄存器IR
 - ●指令译码器ID
 - 条件转移逻辑电路
 - 定时控制逻辑电路等

9

2020/6/8

80C51逻辑结构及信号引脚

K ji x if

A ji x ij

2020/6/8

80C51逻辑结构

80C51逻辑结构及信号引脚

HIIH XTAL,

XTAL,

週道1驱动器

8

80C51逻辑结构及信号引脚

80C51逻辑结构(续)

●单片机执行指令是在控制电路的控制下进行的,执行一 条指令的大致过程:

读出指令->指令寄存器->指令译码器(译码)->定时与控制 逻辑电路(由控制定时逻辑电路产生各种定时信和控制 信号, 然后送往系统各部件去进行相应的操作)

1 80C51逻辑结构(续)

- 80C51单片机内部结构
 - ◆1) CPU --- (1) 控制器电路 (续)
 - 程序计数器PC。存放的是下条指令的地址。
 - 其基本的工作过程是: 读指令时,程序计数器PC将其中的 数作为所取指令的地址输出给程序存储器,然后程序存储 器按此地址输出指令字节,同时程序计数器PC本身自动加 1,指向下一条指令地址。
 - 程序计数器PC变化的轨迹决定程序的流程。
 - 在执行条件转移或无条件转移指令时,程序计数器将被置 入转移的目的地址,程序的流向发生变化。
 - 在执行调用指令或响应中断时,将子程序的入口地址或者 中断向量地址送入PC,程序流向发生变化

80C51逻辑结构(续)

- 80C51单片机内部结构
 - ◆1) CPU --- (1) 控制器电路 (续)
 - 数据指针 DPTR, 一个16位的特殊功能寄存器, 主要功能是作为 片外数据存储器或I/0寻址用的地址寄存器(间接寻址),也可 以作为两个8位寄存器处理,其高8位用DPH表示,低8位用DPL表
 - 访问片外数据存储器或I/0的指令为:

MOVX A, @DPTR : 读

MOVX @DPTR, A ; 写

DPTR寄存器也可以作为访问程序存储器时的基址寄存器。这 时寻址程序存储器中的表格、常数等单元,不是寻址指令

MOVC A, @A+DPTR

JMP @A+DPTR

10

80C51逻辑结构及信号引脚

11

12

80C51逻辑结构及信号引脚

2020/6/8

80C51逻辑结构及信号引脚



A JE & A

80C51逻辑结构(续)

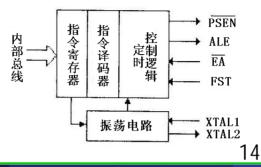
80C51逻辑结构 (续)

- 80C51单片机内部结构
 - ◆1) CPU --- (1) 控制器电路 (续)
 - 指令寄存器IR、指令译码器及定时控制逻辑
 - 指令寄存器 IR是用来存放指令操作码的专用寄存器。执行程 序时,首先进行程序存储器的读操作,也就是根据程序计数 器给出的地址从程序存储器中取指令,送指令寄存器IR,IR 的输出送指令译码器: 然后由指令译码器对该指令进行译码
 - , 译码结果送**定时控制逻辑电路**
 - 定时控制逻辑电路则根据指令的性质发出一系列定时控制信 号,控制计算机的各组成部件进行相应的工作
 - · 条件转移逻辑电路主要用来控制程序的分支转移。在80C51中 , 转移条件也可分为两部分, 一部分是内部条件, 即程序状态 标志位(PSW)和累加器的零状态;另一部分是外部条件,即 F0和所有位寻址空间的状态 (解释位寻址)

13

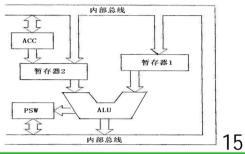
■ 80C51单片机内部结构

- ◆1) CPU --- (1) 控制器电路 (续)
 - 指令寄存器IR、指令译码器及定时控制逻辑(续)



■ 80C51单片机内部结构

◆1) CPU --- (2) 运算器电路 包括: ALU(Arithmetic Logic Unit)、ACC (Accumulator 累加器)、B寄存器、程序状态字 PSW(Program Status Word)、TEMP1和TEMP2两个暂存器等



80C51逻辑结构及信号引脚

2020/6/8

80C51逻辑结构及信号引脚

10

80C51逻辑结构及信号引脚

80C51逻辑结构(续)

- 80C51单片机内部结构
 - ◆ 1) CPU --- (2) 运算器电路 (续)
 - ALU有两个输入
 - 通过暂存器1的输入:输入数据来自寄存器、直接寻址单元(含I/0 口)、内部RAM、寄存器B或是立即数
 - 通过**暂存器 2或累加器 ACC的输入**:通过暂存器 2的运算的指令有 ANL A. #data: AND data to A

ORL A, #data

XRL A, #data

- ALU有两个输出
 - 数据经过运算后, 其结果又通过内部总线送回到累加器中
 - 数据运算后产生的标志位输出至程序状态字 PSW

80C51逻辑结构(续)

- ◆ 80C51单片机内部结构
- ◆ 1) CPU --- (2) 运算器电路 (续)
- ◆ 累加器A
- ◆ 累加器A是CPU中使用最频繁的一个八位专用寄存器,简称ACC或A寄存器。 主要功能:存放操作数,是ALU单元的输入之一,也是ALU运算结果的暂存 单元
- ◆ 由于**累加器的"瓶颈"**作用制约着单片机运算速度的提高,人们又推出寄 存器阵列来代替累加器,赋予更多寄存器以累加器功能,形成了多累加器 结构

80C51逻辑结构(续)

- 80C51单片机内部结构
 - ◆1) CPU --- (2) 运算器电路 (续)
 - 程序状态字PSW



- · 按位定义的8位寄存器, 其内容主要部分是ALU的输出。 其中有些位是根据指令执行结果,由硬件自动生成,而 有些位状态可用软件方法设定。
- 除PSW.1(保留位)、RS1和RSO(工作寄存器组选择控制 位) 及用户标志F0之外,其他四位: 奇偶校验位P、溢出 标志位OV、辅助进位标志位AC及进位标志位CY都是ALU 运算结果的直接输出
- 条件转移指令就是根据PSW中的相关标志位的状态实现 程序的转移。它是一个程序可访问的寄存器,而且可以 按付访问。

18

16

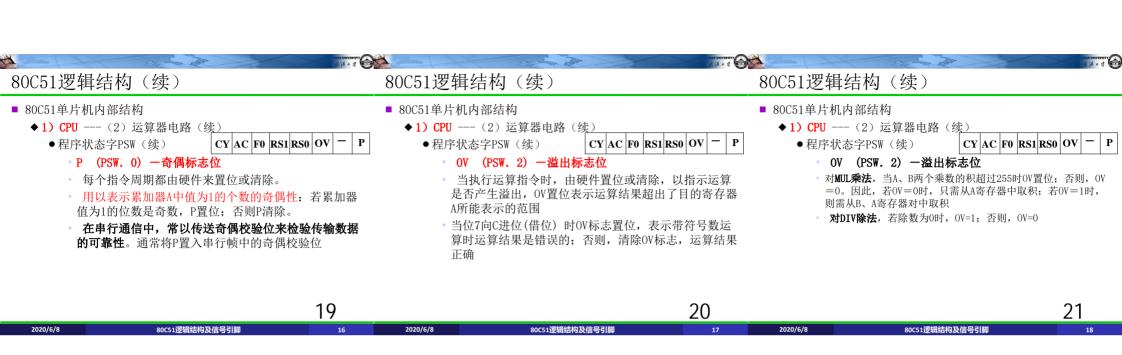
80C51逻辑结构及信号引脚

2020/6/8

80C51逻辑结构及信号引脚

2020/6/8

80C51逻辑结构及信号引脚



- 80C51单片机内部结构
 - ◆1) CPU --- (2) 运算器电路 (续)
 - ●程序状态字PSW(续)

CY|AC|F0|RS1|RS0|OV| - |P|

• RS1、RS0 (PSW. 4、PSW. 3) -工作寄存器组选择位

用于设定当前工作寄存器的组号

| RS1 | RS0 | 组号 | R0~R7地址 |
|-----|-----|----|---------|
| 0 | 0 | 0组 | 00H∼07H |
| 0 | 1 | 1组 | 08H∼0FH |
| 1 | 0 | 2组 | 10H∼17H |
| 1 | 1 | 3组 | 18H∼1FH |

80C51逻辑结构(续)

- 80C51单片机内部结构
 - ◆1) CPU --- (2) 运算器电路 (续)
 - ●程序状态字PSW(续)
- CY AC FO RS1 RS0 OV -
 - AC (PSW. 6) 辅助进位标志位

当进行加法或减法运算时, 若低4位向高4位数发生进位或借位 时AC将被硬件置位; 否则,被清除。

在十进制调整指令DA中要用到AC标志位状态。

• CY (PSW. 7) 一进位标志位

在进行算术运算时,可以被硬件置位或清除,以表示运算结果 中高位是否有进位或借位。在布尔处理机中CY被认为是位累加器

• FO (PSW. 5) -用户标志位/软件标志位

开机时该位为"0"。用户可根据需要,通过位操作指令置 "1"或者清"0"

80C51逻辑结构 (续)

- 80C51单片机内部结构
 - ◆2)程序存储器
 - 根据内部是否带有程序存储器而形成三种型号:

内部没有程序存储器的称80C31

内部带ROM的称80C51

内部以EPROM代替ROM的称87C51

内部以FLASH代替ROM的称89C51

- (1) 片内只读存储器: 片内掩膜ROM, 程序必须在制作单片机时写入
- (2) 片内可编程的ROM: 可直接由用户进行编程
 - *紫外线可擦除型ROM-必须脱机固化,不能在线改写。
 - 电可擦除型ROM——称为Flash单片机(如89C51)
- (3) 片外只读存储器

23

22

80C51逻辑结构及信号引脚

80C51逻辑结构及信号引脚

80C51逻辑结构及信号引脚



80C51逻辑结构(续)

■ 80C51单片机内部结构

◆ 3) 数据存储器 RAM

80C51逻辑结构(续)

- 在单片机中,寄存器的运行速度高于数据存储器 RAM
- 当内部RAM容量不够时,可通过串行或并行总线外扩数据存储器

◆4) 并行I/0口

- 单片机往往提供了许多功能强、使用灵活的并行输入/输出引脚,用 于检测与控制。有些I/0引脚还具有多种功能,比如可以作为数据总 线的数据线、地址总线的地址线、控制总线的控制线等。
- 80C51有四个8位的I/0口(P0、P1、P2、P3),以实现数据的并行 输入输出

■ 80C51单片机内部结构

◆5) 串行I/0口

●目前大部分8位单片机均设置了全双工串行I/0口,用以实 现与某些终端设备进行串行通信。

◆6) 定时器/计数器

- 在单片机的实际应用中,往往需要精确的定时,或者需对 外部事件进行计数。为了减少软件开销和提高单片机的实 时控制能力, 故均在单片机内部设置定时器/计数器电路。 80C51共有二个16位的定时器/计数器,80C52则有三个16位 的定时器/计数器。
- 定时器/计数器是嵌入式芯片的核心部件。(定时、计数、 捕获、PWM等都是通过定时器/计数器部件实现的)

■ 80C51单片机内部结构

◆7) 中断系统

●80C51单片机具有内、外五个中断源,即外中断两个,定时 /计数中断2个,串行中断1个。全部中断分为高级和低级二 个中断优先级

◆8) 振荡器电路及元件

• 计算机的整个工作是在时钟信号的驱动下,按照严格的时 序有规律地一个节拍一个节拍地执行各种操作。单片机内 部设计有定时电路,只需外接振荡元件即可工作。外接振 荡元件一般选用晶体振荡器(替代品:陶瓷振荡器),或用价廉 的RC振荡器,也可用外部时钟源,作为振荡元件。也有的 单片机将振荡元件也集成在芯片内部, 叫内部振荡器 (精度 不高, 1-5%的误差)

27

2020/6/8 80C51逻辑结构及信号引脚

A 注 A 读

80C51逻辑结构及信号引脚

80C51逻辑结构及信号引脚

25

26

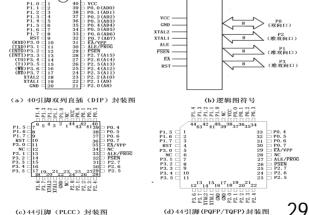
A JE & A

80C51信号引脚

■ 80C51的封装

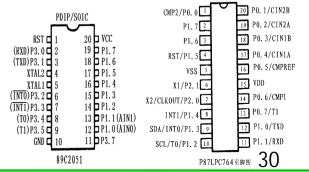
- ◆40引脚双列直插 (**DIP**, Dual In-line Package)
- ◆44引脚 (PLCC, plastic leaded chip carrier)
- ◆44引脚 (PQFP/TQFP, plastic quad flat pack / Thin Quad Flat Package) 封装形式

80C51信号引脚 (续)



80C51信号引脚

■ 在某些场合,不需通过并行总线扩展芯片,常采用20引脚双列 直插(DIP)/14引脚的单片机,如ATMEL公司的1051/2051/4051 单片机等,或PHILIPS公司的P87LPC764单片机



28

80C51逻辑结构及信号引脚

80C51逻辑结构及信号引脚

80C51逻辑结构及信号引脚



jf x

80C51信号引脚(续)

AT89C51和AT89C2051主要性能表

| AT89C51 | AT89C2051 |
|--------------------------|-----------------------------------|
| 4KB可编程Flash存储器(可擦写1000次) | 2KB可编程Flash存储器(可擦写1000次) |
| 三级程序存储器保密 | 两级程序存储器保密 |
| 工作频率: OHz-24MHz | 工作频率: OHz-24MHz |
| 128字节内部RAM | 128字节内部RAM |
| 2个16位定时/计数器 | 2个16位定时/计数器 |
| 一个串行通讯口 | 一个串行通讯口 |
| 5个中断源 | 5个中断源 |
| 32条I/0引线 | 15条I/0引线 |
| 片内时钟振荡器 | 片内时钟振荡器 , 1个片内 <mark>模拟比较器</mark> |
| | (AIN0-同相输入, AIN1-反相输入) 31 |

80C51信号引脚(续)

■ 1) 电源和晶振:

• Vcc——电源 • Vss——地线

电源有三类: TYPE A:5V TYPE B:3.3V TYPE C:1.8V 电源波动范围: 国标DC电源误差范围± 10%, 少量芯片只达到± 5% 宽电源芯片:同时支持TYPEA/B等

- XTAL1——片内振荡器的反相放大器输入端
- XTAL2——片内振荡器的反相放大器输出端。
- ◆ 使用外部振荡器时,外部信号应直接加到XTAL1,而XTAL2悬空。根据频 率微调电容取15-30P左右。



■ 2) I/0 共4个口, 32根I/0线

80C51信号引脚(续)

- ◆ P0——8位、漏极开路的双向I/0口
 - 当使用片外存储器 (ROM及RAM) 时, 作低八位地址和数据总 线分时复用
 - PO口 (作为总线时) 能驱动 8个 LSTTL (Low-power Schottky Transistor - Transistor Logic -- 低功耗肖特基TTL) 负载
- ◆ P1——8位、准双向I/0 □
 - 在编程/校验期间,用做输入低位字节地址
 - ●P1口可以驱动 4个 LSTTL负载
- ◆ P2---8位、准双向I/0口
 - 当使用片外存储器 (ROM及RAM) 时,输出高8位地址
 - 在编程/校验期间,接收高位字节地址
 - P2口可以驱动4个LSTTL负载

33

2020/6/8

80C51逻辑结构及信号引脚

2020/6/8

80C51逻辑结构及信号引脚

80C51逻辑结构及信号引脚

80C51信号引脚(续)

■ 2) I/0 共4个口, 32根I/0线(续)

- ◆ P3——8位、准双向I/0口,具有内部上拉电路
 - P3提供了各种替代的第二功能。在提供这些功能时, 其输出锁存器应由程 序置 1。P3口可以输入/输出4个LSTTL负载。
 - P3口第二功能(实际使用时,总是按需要优先选择第二功能,剩下不用的 才作口线使用)

| I/O 🗆 | 第二功能 | 注释 | |
|------------------|-------------------|-------------|---|
| P _{3.0} | RXD | 串行口敷据接收嘴 | |
| P _{3.1} | TXD | 串行口敷据发送嘴 | |
| P _{3.2} | /INT ₀ | 外部中断请求0 | |
| P _{3.3} | /INT ₁ | 外部中断请求1 | |
| P _{3.4} | T ₀ | 定时/计数器0外部输入 | |
| P _{3.5} | T ₁ | 定时/计数器1外部输入 | |
| P _{3.6} | /WR | 外部RAM写信号 | |
| P _{3.7} | /RD | 外部RAM读信号 | |
| | | | 7 |
| | | | |

80C51信号引脚 (续)

■ 3) 控制线: 共4根

- ◆ · RST (VPD: 备用电源引入端, 当电源发生故障, 电源降到下限 值时,备用电源经此端向内部RAM提供电压,以保护内部RAM中的 数据不丢失)——复位输入信号,高电平有效。在振荡器工作时 ,在RST上作用两个机器周期以上的高电平,将器件复位。
- ◆ /EA (Vpp: 编程电压, 具体电压值视芯片而定) ——片外程序 存储器访问允许信号,低电平有效。/EA=1,选择片内程序存储 器: /EA=0,则程序存储器全部在片外而不管片内是否有程序存 储器。

80C51信号引脚(续)

- 3) 控制线: 共4根(续)
 - ◆ · ALE (PROG: 编程脉冲) ——地址锁存允许信号,输出
 - 在访问片外存储器或I/0时,用于锁存低八位地址,以实现低 八位地址与数据的隔离。即使不访问外部存储器, ALE端仍以 固定的频率输出脉冲信号(此频率是振荡器频率的1/6)。在访 问外部数据存储器时,出现一个ALE脉冲
 - ALE可以驱动8个LS TTL负载
 - 对片内程序存储器编程时,该引脚用于输入编程脉冲PROG
 - ◆ /PSEN——片外程序存储器读选通信号, 低电平有效
 - 在从片外程序存储器取指期间,在每个机器周期中,当其有 效时,程序存储器的内容被送上 P0口(数据总线)
 - ●它可以驱动 8个LSTTL负载

80C51逻辑结构及信号引脚

36

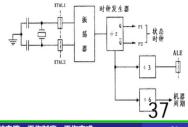
80C51逻辑结构及信号引脚

80C51逻辑结构及信号引脚



80C51时钟电路

- 时钟电路用于产生单片机工作所需要的时钟信号
- 在80C51内带有时钟电路,在片外通过XTAL1和XTAL2引脚接入定时控 制元件(晶体振荡器和电容),即构成一个稳定的自激振荡器
- 在80C51芯片内部有一个高增益反相放大器,而在芯片的外部, XTAL1和XTAL2之间跨接晶体振荡器和微调电容
- 时钟电路组成:振荡器及定时控制元件、时钟发生器、地址锁存允 许信号 ALE



■ 振荡器及定时控制元件

80C51时钟电路(续)

- ◆ 在80C51芯片内部有一个高增益反相放大器, 其输入端为芯片引脚XTAL1 , 其输出端为引脚XTAL2
- ◆ 在片外通过XTAL1和XTAL2引脚跨接晶体振荡器和微调电容,形成反馈电 路,振荡器即可工作
- ◆ 振荡器的工作由PD位(特殊功能寄存器PCON中的一位)控制。当 PD置1 时,振荡器停止工作,系统进入低功耗工作状态
- ◆ 振荡器的工作频率在1.2~12 MHz之间,由于制造工艺的改讲,有些单 片机的频率范围, 高端可达40 MHZ
- ◆ 在由多片单片机组成的系统中, 为了使各单片机之间时钟信号的同步, 应当引入唯一的公用外部脉冲信号作为各单片机的振荡脉冲
- ◆ 当由外部输入时钟信号时,外部信号接入XTAL1,XTAL2悬空不用。对外 部信号的占空比没有要求, 高/低电平持续时间应不小于 20 ns

■ 内部时钟发生器

80C51时钟电路(续)

◆ 内部时钟发生器实质上是一个**2分频**的触发器。其输入由振荡器 引入的,输出为两个节拍的时钟信号。输出的前半周期,节拍1 (P1) 信号有效: 后半周期, 节拍2(P2) 信号有效。每个输出 周期为一个计算机CPU的状态周期,即时钟发生器的输出为状态 时钟。每个状态周期内包括一个P1节拍和一个P2节拍,形成CPU 内的基本定时时钟

■ ALE信号

◆ 状态时钟经过3分频之后,产生ALE引脚上的信号输出

39

80C51 时钟电路、工作时序、工作方式

A jé x d

80C51 时钟电路、工作时序、工作方式

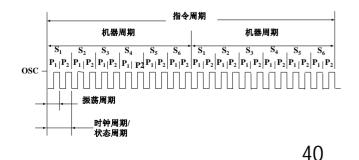
2020/6/8

80C51 时钟电路、工作时序、工作方式

38

时序定时单位

■ 单片机执行指令是在时序电路的控制下一步一步进行的。 80C51的时序定时单位共有4个:振荡周期/节拍、时钟周 期/状态、机器周期和指令周期



时序定时单位(续)

■ 1) 振荡周期/节拍P

◆ 为单片机提供定时信号的振荡脉冲的周期称为振荡周期。

■ 2) 时钟周期/状态周期/状态S

◆ 时钟周期是振荡周期的两倍,又称状态周期或状态S。一个状态S 有两个节拍, 其前半周期对应节拍叫P1, 后半周期对应节拍叫P2

3) 机器周期

- ◆ 通常将完成一个基本操作所需的时间称为机器周期。 定时控制方式, 因此它有固定的机器周期
- ◆ 规定一个机器周期的宽度为6个状态,表示为S1~S6。一个机器 周期共有12个振荡脉冲周期,机器周期就是振荡脉冲的12分频
- ◆机器周期是单片机的最小时间单位

时序定时单位(续)

■ 4) 指令周期

◆ 执行一条指令所需要的时间称为指令周期。80C51的指令周期根 据指令的不同,可包含有一、二、三、四个机器周期

例如,设单片机外接晶振为12MHz时,四个周期的具体值为:

振荡周期=1/12MHz=1/12 µs=0.0833 µs

时钟周期=1/6 µs=0.167 µs

机器周期=1µs

指今周期=1~4 us

41 42

80C51 时钟电路、工作时序、工作方式

2020/6/8

80C51 时钟电路、工作时序、工作方式



指令时序(续)

A JE & A

指令时序(续)

- 80C51指令按其长度分: 单字节、双字节和三字节指令
 - ◆ 单字节或双字节指令可能是单机器周期或双机器周期的
 - ◆ 三字节指令是双机器周期的
 - ◆ 乘除指令是四个机器周期的
- 1) 单机器周期指令
 - ◆ 双字节时, 执行在S1P2开始, 操作码被读入指令寄存器: 在S4P2 时, 再读入第二个字节
 - ◆ 单字节时, 执行在S1P2开始, 操作码被读入指令寄存器: 在S4P2 时仍有读操作,被读入的字节(即下一操作码)被忽略,且此时 PC并不增量
 - ◆以上两种情况均在S6P2时结束操作

- 2) 双机器周期指令
 - ◆ 单字节第一种情况: 执行在S1P2开始, 在两个机器周期中, 共发 生四次读操作,但是后三次操作都无效。
 - ◆ 单字节第二种情况: 执行在S1P2开始, 操作码被读入指令寄存器 ;在S4P2时,再读入的字节被忽略。由S5开始送出外部数据存储 器的地址,随后是读或写的操作。在读、写期间,ALE不输出有 效信号。在第二个机器周期, 片外数据存储器寻址和选通, 但不 产生取指操作
- ALE信号是为地址锁存而定义的,每有效一次对应单片机进行一次读 指令操作。ALE信号以振荡脉冲六分之一的频率出现,因此在一个机 器周期中, ALE信号两次有效,第一次在S1P2和S2P1期间, 第二次在S4P2和S5P1期间,有效宽度为一个状态周期S
 - ◆ 单字节单周期指令 (如 INC A)
 - 只需进行一次读指令操作,第二个ALE有效时,PC没有加1,读出的还是原指令
 - ◆ 双字节单周期指令 (如 ADD A, #data)
 - 两次读操作都有效,第一次读指令操作码,第二次读指令第二字节#data
 - ◆ 单字节双周期指令 (如 INC DPTR)
 - 两个机器周期共进行四次读指令的操作,后三次的读操作全是无效的
 - ◆ 单字节双周期指令 (如MOVX)
 - MOVX类指令情况有所不同。因为执行这类指令时,先在ROM读取指令,然后 对外部RAM进行读/写操作。第一机器周期时,与其它指令一样,第一次读 指令(操作码)有效,第二次读指令操作无效。第二周期时,进行外部RAM 访问,此时与ALE信号无关,因此不产生读指令操作

45

2020/6/8

80C51 时钟电路、工作时序、工作方式

80C51 时钟电路、工作时序、工作方式

80C51 时钟电路、工作时序、工作方式

43

40

A ji a if

44

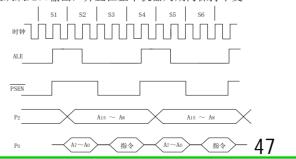
jf +

指令时序(续)

(a) 单字节单周期指令、如INC - 读操作码 S4 S5 S6 (b) 双字节单周期指令, 如ADD A, #data 一读操作码 S1 S2 S3 S4 S5 S6 (c) 单字节双周期指令,如INC DPTR - 读操作码 _ 个操作码(丢弃) 51 52 53 54 55 (d) 单字节双周期指令, 如MOVX类型 读下一个 操作码 不取指 S1 52 S1 S2 S3 S6 -地址-数据-46 ---访问片外存储器

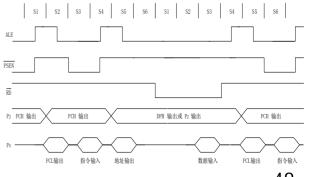
访问外部ROM和RAM的时序

- 对外部程序存贮器的访问使用/PSEN作读选通信号
- 1) 访问外部ROM的时序
 - ◆ 当从外部程序存贮器读取指令时,需要使用16位地址,且高8位 地址从从P2口输出,并且在整个机器周期内保持不变



访问外部ROM和RAM的时序

■ 2) 访问外部RAM的时序

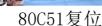


48

80C51 时钟电路、工作时序、工作方式

80C51 时钟电路、工作时序、工作方式





80C51复位(续)

■ 80C51单片机工作方式

80C51工作方式

- ◆复位
- ◆程序执行
- ◆低功耗
- ◆编程和校验

■ 1) 复位操作

- ◆ 复位是单片机的初始化操作, 主要功能是把PC初始化为0000H, 使单片机从0000H单元开始执行程序。当由于程序运行出错或操 作错误使系统处于死锁状态时,为摆脱困境,可以按复位键以重 新启动,也可以通过WDT看门狗定时器来强迫复位(WDT可在单片 机系统受干扰使程序不能正常运行时,自动产生复位信号。
- ◆除PC之外, 复位操作还对其它一些特殊功能寄存器有影响。
- ◆ 复位操作还对单片机的个别引脚信号有影响。例如在复位期间, ALE和/PSEN信号变为无效状态,即 ALE=0, /PSEN =1

| 1 |) 复位操作 | (续) |
|---------------------|--------|-----|
|---------------------|--------|-----|

| _1) 复址探作 | (娑) | | |
|----------|------------|---------|-----------|
| 特殊功能寄存器 | 初始状态 | 特殊功能寄存器 | 初始状态 |
| ACC/A | 00Н | TMOD | 00Н |
| В | 00H | TCON | 00Н |
| PSW | 00H | ТН0 | 00H |
| SP | 07H | TL0 | 00H |
| DPL | 00Н | TH1 | 00H |
| DPH | 00Н | TL1 | 00H |
| P0~P3 | FFH | SBUF | xxxxxxxB |
| IP | xx0000000B | SCON | 00Н |
| IE | 0x000000B | PCON | 0xxx0000B |

50

80C51 时钟电路、工作时序、工作方式

80C51 时钟电路、工作时序、工作方式

MAN UNIVERSITY 表注 A 提

80C51 时钟电路、工作时序、工作方式

2020/6/8

49

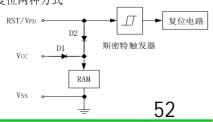
2020/6/8

A 16 A A

80C51复位(续)

■ 2) 复位信号

- ◆ RST引脚是复位信号的输入端。复位信号是高电平有效,其有效 时间应持续24个振荡周期(即2个机器周期)以上。若使用频率为 6MHz的晶振,则复位信号应持续4μs以上
- ◆ 整个复位电路包括芯片内、外两部分。外部电路产生的复位信号 送施密特触发器,再由片内复位操作。
- ◆有上电自动复位、按键复位两种方式



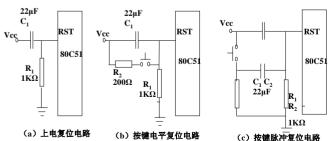
80C51复位(续)

■ 2) 复位信号(续)

- ◆ 上电自动复位是通过外部复位电路的电容充电来实现的, 只要电 源的上升时间不超过1ms,就可以实现上电自动复位,即接通电 源就完成了系统的复位初始化。
- ◆ 按键复位有电平方式和脉冲方式两种。
 - 按键电平复位是通过使复位端经电阻与Vcc电源接通而实现的
 - ●按键脉冲复位则是利用RC微分电路产生的正脉冲来实现的
- ◆ 图中电阻电容参数适用于6MHz晶振,能保证复位信号高电平持续 时间大于2个机器周期

80C51复位(续)

■ 2) 复位信号(续)



54

53

80C51 时钟电路、工作时序、工作方式

80C51 时钟电路、工作时序、工作方式



80C51 程序执行

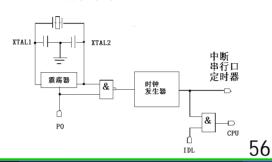
■ 程序执行方式是单片机的基本工作方式

- ◆ 由于复位后PC=0000H, 因此程序执行总是从0000H开始的。一般 在0000H开始的单元中存放一条无条件转移指令,以便跳转到实 际主程序的入口去执行
- ◆比如:

ORG 0000H

S.TMP MAIN: 转主程序 80C51低功耗工作方式

- 80C51有两种低功耗方式,由电源控制寄存器(PCON)的有关 位来控制
 - ◆ 待机/空闲(IDLE)方式
 - ◆ 掉电 (POWER DOWN) 保护方式



80C51 编程工作方式(烧录)

- 对于片内具有EPROM型程序存储器的87C51(87C52) 和片内具有 闪速存储器的89C51 (89C52) 、78E51 (78E52) 等单片机可以 通过编程来修改程序存储器中的程序
- 89C51内部有一个4KB的Flash EEROM。编程接口可接收高电压 (12V) 或低电压(Vcc) 的允许编程信号。低电压编程方式可以 很方便地与89C51内的用户系统进行编程;而高压编程方式则 可与通用的EPROM编程器兼容。

57

55

80C51 时钟电路、工作时序、工作方式

80C51 时钟电路、工作时序、工作方式

2020/6/8

80C51 时钟电路、工作时序、工作方式

A JE & A

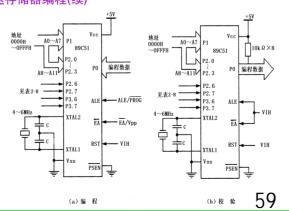
80C51 编程工作方式(续)

- 1) 闪速存储器编程
 - ◆根据特征字节(地址为032H)的内容选择合适的编程电压 (VPP=12V或5V)
 - ◆片擦除操作时,要求PROG的脉冲宽度为10ms

| 方式↩ | RST₽ | PSEN ← | ALE/PROG | <u>EA</u> /Vpp₽ | P2.6₽ | P2.7₽ | P3.6₽ | P3.7₽ |
|-----------|------|--------|----------|-----------------|-------|-------|-------|-------|
| 写代码数据₽ | H₽ | L₽ | 负脉冲₽ | H/12V(1)₽ | L⊎ | H⇔ | Н€ | H₽ |
| 读代码数据₽ | H₄J | L₽ | Н€ | H/12V₽ | L₽ | L⊎ | H€J | H₽ |
| 写锁定位 LB1√ | H↔ | L↔ | 负脉冲↩ | H/12V₽ | H↔ | H↔ | H↔ | H↔ |
| LB2↔ | H↓ | L⊎ | 负脉冲↩ | H/12V⊬ | H↔ | H↔ | H↔ | L↔ |
| LB3¢³ | H₽ | L₽ | 负脉冲₽ | H/12V₽ | H₽ | L⊎ | L₽ | L₽ |
| 片擦除₽ | H₽ | L₽ | 负脉冲(2)₽ | H/12V₽ | HeJ | L₽ | L₽ | L₽ |
| 读特征标记。 | H⇔ | L₽ | Н€ | H₽ | L₽ | L₽ | L₽ | L₽ |

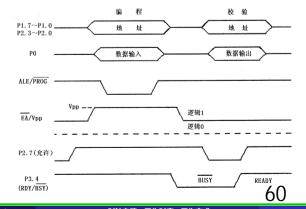
80C51 编程工作方式(续)

■ 1) 闪速存储器编程(续)



80C51 编程工作方式(续)

■ 1) 闪速存储器编程(续)



58

80C51 时钟电路、工作时序、工作方式

80C51 时钟电路、工作时序、工作方式



80C51 编程工作方式(续)

80C51 编程工作方式(续)

■ 1) 闪速存储器编程(续)

- ◆ 步骤
 - 在地址线上输入要编程存储器单元地址。
 - 在数据线上输入要写入的数据。
 - 按要求输出编程和校验的时序。
 - ●对于高压编程模式,将VPP升至12V
 - 每对Flash存储阵列写入一个字节或每写入一个程序加密位, 要向ALE/PROG输出一个编程脉冲。
 - 改变地址和数据, 重复以上几步操作, 直至目标程序(OBI文 件)结束
 - 字节写周期是自动定时的,一般不超过1.5ms

■ 1) 闪速存储器编程(续)

◆ 杳询数据

● 89C51可以通过数据查询来检测一个写周期是否结束。若数据未写完,则从 P0.7引脚上读到的是该数据的最高位的反码。当写周期结束后,读出值即 为写入值。

◆准备好/忙(RDY/BSY) 信号

● 编程期间ALE(/PROG)为高电平后, P3.4引脚被拉成低电平, 表示BUSY; 编 程结束后, ALE (/PROG) 为低电平, 表示READY

◆ 编程校验

● 如果锁定位LB1和LB2没有被编程,代码数据则可读回,用来校验。锁定位 不能直接被校验,只能通过观察它们的功能是否被允许而间接得到证实。

◆ 芯片擦除

● 通过正确的控制信号的组合, 并保持ALE/PROG引脚脉冲宽度(低电平) 约 10ms,则可对EEPROM阵列和三个锁定位进行电擦除,擦除后代码阵列全为 "1"。注意,在对程序存储器进行重新编程前必须执行片擦除操作。

■ 1) 闪速存储器编程(续)

- ◆ 读特征字节
 - 89C51单片机内有三个特征字节, 地址为030H、031H和032H, 分别用来指示该器件的生产厂商、型号和编程电压
 - 比如:

(030H) = 1EH表示ATMER公司生产 (031H) = 51H表示型号为89C51 表示型号为89LV51 =61H表示编程电压为12V (032H) = FFH表示编程电压为5V =0.5H

63

2020/6/8

80C51 时钟电路、工作时序、工作方式

2020/6/8

80C51 时钟电路、工作时序、工作方式

2020/6/8

80C51 时钟电路、工作时序、工作方式

K ji A ij

61

A JE A ST

A jé x d

80C51 编程工作方式(续)

1) 闪速存储器编程(续)

- ◆程序锁定位的功能和编程
 - 80C51片内有三个锁定位, 但不含密码阵列。
 - 当第一级加密时,逻辑电压被取样并锁存。
 - 在复位期间, 若器件为上电而不是复位, 则锁存器内容初始化为一个随机 值,直到复位操作结束

| 1 | 序 | 锁定位编程状况₽ | | Я.₽ | 保护作用₽ | - |
|---|----|----------|------|------|---------------------------------|---|
| | 号₽ | LB1₽ | LB2₽ | LB3₽ | 7.2 | |
| | 1₽ | 未编↩ | 未编↩ | 未编↩ | 锁定作用,若密码阵列已编程,则校验时将得密码型 | |
| | 2₽ | 已編↩ | 未编↩ | 未编↩ | 禁止片外程序存储器中的MOVC指令从片内程序中读取代 | |
| | | | | | 码字; EA 值在复位时被采样并锁入内部,禁止 EPROM 进 | |
| | | | | | 一步编程₽ | |
| | 3₽ | | 已編↩ | | 同 2,但校验也被禁止↩ | |
| | 4₽ | 已編↩ | 已編↩ | 已編↩ | 同 3,并禁止执行片外程序₽ 6.4 | |
| | | | • | • | U 1 | _ |

80C51 布尔(位)处理器

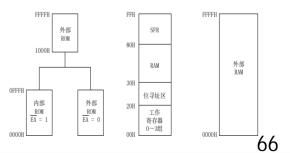
- 在80C51中,与字节处理器相对应,还特别设置了一个结构完 整、功能极强的布尔(位)处理器
 - ◆ 位累加器: 借用进位标志位CY/C。在布尔运算中CY是数据源之一 ,又是运算结果的存放处,位数据传送的中心。根据CY的状态实 现程序条件转移: JC rel、JNC rel、JBC rel
 - ◆ 位寻址的RAM: 内部RAM位寻址区中的0~127位(20H~2FH):
 - ◆ 位寻址的寄存器: 特殊功能寄存器 (SFR) 中的可以位寻址的位
 - ◆ 位寻址的I/0口: 并行I/0口中的可以位寻址的位(如P1.0)。
 - ◆ 位操作指令: 位操作指令可实现对位的置位、清0、取反、位状 态判跳、传送、位逻辑运算、位输入/输出等操作

80C51存储器结构与地址空间

- Intel的MCS-51和80C51系列单片机采用哈佛结构
 - ◆ 物理上4个存储器空间

● 程序存储器: 片内程序存储器, 片外程序存储器

● 数据存储器: 片内数据存储器, 片外数据存储器



80C51 存储器结构与地址空间

65

80C51 时钟电路、工作时序、工作方式

80C51 时钟电路、工作时序、工作方式 2020/6/8

2020/6/8



80C51程序存储器

80C51程序存储器(续)

- 逻辑上有3个存储器地址空间(三种基本寻址空间)
 - ◆ 片内、片外统一的 64 KB程序存储器地址空间:

80C51存储器结构与地址空间(续)

- ◆ 片内256B(80C52为384B)数据存储器地址空间:
- ◆ 片外64 KB的数据存储器地址空间。
- 在访问这3个不同的逻辑空间时,应选用不同形式的指令(分别 用MOVC、MOV、MOVX指令来区分三个不同的逻辑空间)
- 用于存放程序和常数
- 采用16位的地址总线,可直接扩展的地址空间为64KB
- CPU访问片内和片外存储器,可由/EA引脚所接的电平来确定
 - ◆ /EA接高电平时,程序从片内程序存储器0000H开始执行; 当PC值 超出片内ROM容量时,会自动转向片外程序存储器空间执行
 - ◆ /EA接低时,系统执行片外程序存储器0000H开始存放的程序
 - ◆ 对于片内无ROM的80C31/80C32单片机, 应将/EA引脚固定接低, 以使系统执行片外程序存储器程序

- 程序存储器的某些单元被保留用于特定的程序入口地址
 - ◆ 0003H~002BH: 6个中断源的中断服务程序入口地址
 - ◆ 以下7个特定地址被保留用于中断入口地址, 这种中断模式叫 独立向量模式:

复位 0000H~0002H(3个单元) 外部中断0 0003H~000AH (8个单元) 000BH~0012H (8个单元) 计时器T0溢出 外部中断1 0013H~001AH (8个单元) 计时器T1溢出 001BH~0022H (8个单元) 串行口中断 0023H~002AH (8个单元)

*计时器 T2/T2EX 002BH~0032H (8个单元, 80C52)

◆ 在程序设计时,通常在这些中断入口处设置无条件转移指令,使之转向 对应的中断服务程序段处执行

68 80C51 存储器结构与地址空间

69

2020/6/8

80C51 存储器结构与地址空间

67

15 x 2 CONTROL OF

80C51 存储器结构与地址空间

80C51片内数据存储器

80C51片内数据存储器(续)

- ◆片内数据RAM区
 - 对80C51, 为地址空间的低128B。
 - 对80C52, 为地址空间的0~256B。
- ◆ 特殊功能寄存器SFR区
 - 对80C51, 为地址空间的高128B
 - 对80C52, 高128B的RAM区和SFR区的地址空间是重叠的。究竟访问哪 一个区是通过不同的寻址方式来加以区别的,访问高128B RAM区时
 - , 选用间接寻址方式; 访问SFR区, 选用直接寻址方式

■ 片内数据存储器(IRAM)地址只有8位,最大寻址范围为256个字节

255 D8H DOH SFR中 (80052申春) 位寻址区 128 127 120 2FH 位寻址区 24 R7~R0 3组 R7~R0 2组 16 R7~R0 1组

■片内数据RAM区

- ◆ 工作寄存器区

80C51片内数据存储器(续)

• 用寄存器直接寻址区域, 指令数量最多, 均为单周期指令, 速度最快 ● 片内数据 RAM区的 0~31 (00H~1FH), 共 32个单元, 是 4个通用工作寄

存器组,每个组包含8个8位寄存器,编号为 RO~R7

● 在某一时刻,只能选用一组寄存器。通过程序状态字(PSW)中的RSO、RS1 二位的设置来实现的,

| RS1 | RS0 | 组号 | 寄存器R0~R7地址 |
|-----|-----|----|------------|
| 0 | 0 | 0组 | 00H~07H |
| 0 | 1 | 1组 | 08H~0FH |
| 1 | 0 | 2组 | 10H∼17H |
| 1 | 1 | 3组 | 18H∼1FH |

70

片内数据RAM

71

80C51 存储器结构与地址空间

72

80C51 存储器结构与地址空间

80C51 存储器结构与地址空间







80C51片内数据存储器(续)

◆ 位寻址区

2020/6/8

■ 片内数据RAM区(续)

- 片内数据RAM区的32-47 (20H-2FH) 的16个字节单元, 共128位, 是可位 寻址的RAM区。既可进行字节寻址,又可位寻址
- 这16个位寻址单元,再加上可位寻址的特殊功能寄存器一起构成了布尔(位) 处理器的数据存储器空间

| 字节地 址 | 位 地 | | | | | | | |
|----------|------------------|-------|------------------|-------|----------------|----------------|------------------|----------------|
| | \mathbf{D}_{7} | D_6 | \mathbf{D}_{s} | D_4 | \mathbf{D}_3 | \mathbf{D}_2 | \mathbf{D}_{1} | $\mathbf{D_0}$ |
| 2FH | 7FH | 7EH | 7DH | 7CH | 7BH | 7AH | 79H | 78H |
| 2EH | 77H | 76H | 75H | 74H | 73H | 72H | 71H | 70H |
| 2DH | 6FH | 6EH | 6DH | 6CH | 6BH | 6AH | 69H | 68H |
| 2CH | 67H | 66H | 65H | 64H | 63H | 62H | 61H | 60H |
| 2BH | 5FH | 5EH | 5DH | 5CH | 5BH | 5AH | 59H | 58H |
| 2AH | 57H | 56H | 55H | 54H | 53H | 52H | 51H | 50H |
| 29H | 4FH | 4EH | 4DH | 4CH | 4BH | 4AH | 49H | 48H |
| 28H | 47H | 46H | 45H | 44H | 43H | 42H | 41H | 40H |
| 27H | 3FH | 3EH | 3DH | 3CH | 3BH | 3AH | 39H | 38H |
| 26H | 37H | 36H | 35H | 34H | 33H | 32H | 31H | 30H |
| 25H | 2FH | 2EH | 2DH | 2CH | 2BH | 2AH | 29H | 28H |
| 24H | 27H | 26H | 25H | 24H | 23H | 22H | 21H | 20H |
| 23H | 1FH | 1EH | 1DH | 1CH | 1BH | 1AH | 19H | 18H |
| 22H | 17H | 16H | 15H | 14H | 13H | 12H | 11H | 10H |
| 21H | 0FH | 0EH | 0DH | 0CH | 0BH | 0AH | 09H | 08H |
| 20H | 07H | 06H | 05H | 04H | 03H | 02H | 01H | 00H |
| | | | | | | | | |

80C51 存储器结构与地址空间

■ 片内数据RAM区(续)

- ◆ 字节寻址区
 - ▶ 片内数据RAM区的48~127(30H~7FH), 共80个字节单元,可以采用直接 字节寻址的方法访问
 - 对于 80C52, 还有高 128 B的数据 RAM区, 只能采用间接字节寻址
- ◆ 堆栈区及堆栈指示器SP
 - 堆栈是在片内数据RAM区中,数据先进后出/后进先出的区域。堆栈指针/堆栈指示器 SP(Stack Pointer): 存放当前的堆栈栈顶所指存储单元地址的一个8位寄存器
 - 堆栈有二种形式,一是向上生成,二是向下生成。80C51单片机的堆栈向上生成。向 上生成型堆栈: 进栈操作: 先SP+1, 后写入数据; 出栈操作: 先读出数据, 后SP+1
 - 系统复位后SP内容为07H。如不重新定义,则07H为栈底,压栈的内容从08H单元开始 存放。通过软件对SP的内容重新设定,使堆栈区设定在片内数据RAM区(最好是片内 RAM的30H~7FH单元)中的某一区域内,堆栈深度不能超过片内RAM空间
 - 堆栈是为子程序调用和中断操作而设立。其具体功能有两个: 保护断点和保护现场。 在80C51单片机中, 堆栈在子程序调用和中断时会把断点地址自动进栈和出栈, 还有 对堆栈的进栈和出栈的指令(PUSH、POP)操作,用于保护现场和恢复现场
 - 子程序调用和中断都允许嵌套,并可多级嵌套,现场的保护也往往使用堆栈,要注意 给堆栈以一定的深度,以免造成堆栈内容的破坏而引起程序执行的"跑人人

80C51片内数据存储器(续)

- 片内数据RAM区(续)
 - ◆特殊功能寄存器SFR区(Special Function Registers)
 - 用以存放相应功能部件的控制命令、状态或数据的区域
 - ●80C51设有128B片内数据RAM的特殊功能寄存器空间区。除程 序计数器PC和4个通用工作寄存器组外,其余所有的寄存器都 在这个地址空间之内
 - 对于80C51共定义了21个特殊功能寄存器。在80C52中,还增 加了5个特殊功能寄存器, 共计26个
 - ◆ 在80C51的21个(80C52的26个)特殊功能寄存器中,字节地 址中低位地址为OH或8H的特殊功能寄存器(80C51有11个, 80C52还增加了T2CON),除可字节寻址外,还可位寻址

75

6

80C51 存储器结构与地址空间

80C51片内数据存储器(续)

■ 片内数据RAM区(续) --- SFR区(续)

| 序号≠ | 标识符 | 名 称: | 字节地址。 | 位地址∞ |
|-----|--------|--------------------------|----------|----------|
| 1.0 | ACC₽ | 累加器₽ | EOH+2 | ЕОН∼Е7Н₽ |
| 2₽ | Be≀ | B 寄存器。 | FOH+2 | FOH~F7H₽ |
| 3₽ | PS₩₽ | 程序状态字₽ | DOH₽ | DOH~D7H₽ |
| 4₽ | SP↔ | 堆栈指针↩ | 81H₽ | ₽ |
| 5₽ | DPTR+2 | 数据指针(DPH、DPL)↔ | 83H√82H₽ | 47 |
| 6₽ | P0+2 | P0 □ ₽ | 80He2 | 80H~87H₽ |
| 7₽ | P1€ | P1 □ <i>ϕ</i> | 90H₽ | 90H~97H₽ |
| 8₽ | P2+2 | P2 □+ | AOH⇔ | AOH~A7H₽ |
| 9₽ | P3+2 | P3 □+ | BOH- | вон~в7н₽ |
| 10₽ | IP+2 | 中断优先级控制寄存器。 | B8H≠2 | B8H~BFH₽ |
| 11₽ | IE↔ | 中断允许控制寄存器。 | ASH⇔ | ASH~AFH€ |
| 12₽ | TOMD₽ | 定时器 / 计数器方式控制寄存器→ | 89H₽ | e |
| 13₽ | TCON-2 | 定时器/计数器控制寄存器- | 88H+2 | 88H~8FH₽ |
| 14₽ | T2CON₽ | 定时器/计数器 2 控制寄存器→ | C8H₽ | C8∼CFH₽ |
| 15₽ | THO ₽ | 定时器/计数器 0 (高位字节)₽ | 8CH₽ | P |
| 16₽ | TL0= | 定时器/计数器 0 (低位字节)₽ | 8AH⇔ | e |
| 17₽ | TH1₽ | 定时器/计数器 1(高位字节)₽ | 8DH+2 | 4) |
| 18₽ | TL1₽ | 定时器/计数器 1(低位字节)₽ | 8BH₽ | ē. |
| 19₽ | TH2€ | 定时器/计数器 2(高位字节)₽ | CDH₽ | P |
| 20₽ | TL2₽ | 定时器/计数器 2(低位字节)₽ | CCH↔ | 4 |
| 21₽ | RLDH₽ | 定时器/计数器 2 自动重装载 (高位字节) ₽ | CBH₽ | <i>Q</i> |
| 22₽ | RLDL+3 | 定时器/计数器 2 自动重装载(低位字节)→ | CAH+2 | ē. |
| 23₽ | SCON₽ | 串行口控制寄存器。 | 98H₽ | 98H~9FH₽ |
| 24₽ | SBUF₽ | 串行数据缓冲器₽ | 991⊬ | P |
| | PCON- | 电源控制及波特率选择寄存器。 | 97H₽ | ٥ |

80C51片内数据存储器(续)

■ 片内数据RAM区(续) --- SFR区(续)

| • S | F R₽ | | | 位 力 | 也址 | P | | | |
|--------|-------|------|------------|-------|-------------------|--------|------|------|---------------|
| 名称₽ | 字节地址↩ | 7₽ | 6₽ | 5₽ | 4₽ | 3₽ | 2₽ | 1₽ | 0₽ |
| В₽ | FOH₽ | F7H€ | F6H₽ | F5H€ | F4H₽ | F3H₽ | F2H€ | F1H€ | FOH₽ |
| ACC₽ | E0H€ | Е7Н€ | Е6Н₽ | E5H€ | E4H€ | ЕЗН₽ | E2H₽ | E1H€ | EOH€ |
| PS₩↔ | DOH42 | CY₽ | AC₽ | F0₽ | RS1↔ | RSO₽ | 0₩ | -0 | P↔ |
| | | D7H₽ | D6H€ | D5H₽ | D4H₽ | D3H₽ | D2H€ | D1H€ | DOH₽ |
| T2CON₽ | COH₽ | TF2₽ | EXF2€ | RCLK₽ | TCLK₽ | EXEN2₽ | TR2₽ | C/T | CP/RL2 |
| | | CFH₽ | CEH₽ | CDH₽ | CCH€ | CBH₽ | CAH₽ | С9Н₽ | C8H€ |
| IP₽ | B8H₽ | -0 | - <i>₽</i> | PT2₽ | PS₽ | PT1₽ | PX1₽ | PT0€ | PXO⇔ |
| | | BFH₽ | BEH₽ | BDH₽ | BCH₽ | BBH₽ | BAH₽ | В9Н€ | B8H₽ |
| P3€ | BOH₽ | В7Н₽ | B6H₽ | B5H₽ | В4Н₽ | ВЗН₽ | В2Н₽ | B1H€ | BOH₽ |
| IE↔ | A8H₽ | EA₽ | - <i>⇔</i> | ET2€ | ES₽ | ET1₽ | EX1₽ | ETO€ | EXO⇔ |
| | | AFH₽ | AEH₽ | ADH₽ | ACH+ ² | ABH₽ | AAH₽ | А9Н₽ | A8H₽ |
| P2₽ | AOH⇔ | А7Н₽ | А6Н₽ | A5H₽ | A4H↔ | АЗН₽ | A2H₽ | A1H₽ | AOH₽ |
| SCON₽ | 98H₽ | SMO₽ | SM1₽ | SM2€ | REN⊕ | TB8₽ | RB8₽ | TI↩ | RI↔ |
| | | 9FH₽ | 9EH₽ | 9DH₽ | 9CH₽ | 9BH₽ | 9AH₽ | 99₩ | 98H₽ |
| P1₽ | 90H₽ | 97H₽ | 96H₽ | 95H₽ | 94H₽ | 93H₽ | 92H₽ | 91₩ | 90₩ |
| TCON₽ | 88H₽ | TF1₽ | TR1₽ | TF0₽ | TRO₽ | IE1₽ | IT1₽ | IEO€ | ITO₽ |
| | | 8FH₽ | 8EH₽ | 8DH€ | 8CH↔ | 8BH₽ | 8AH₽ | 89H₽ | 88H= |
| P0₽ | 80H₽ | 87H₽ | 86H₽ | 85H₽ | 84H₽ | 83H₽ | 82H₽ | 81H₽ | 8 / H₽ |

80C51片外数据存储器

- 片外数据存储器是在外部存放数据的区域,这一区域只能用寄存器 间接寻址的方法访问, 所用的寄存器为DPTR、R1或R0。指令助记符 为MOVX
 - ◆用RO、R1寻址时,RO、R1为8位寄存器,最大寻址范围为256B在 80C51中,有一个专门的数据存储器的地址指示器——数据指针 DPTR, 用于访问片外数据存储器(ERAM)。DPTR也是16位的寄存 器,80C51具有64 KB的数据存储器扩展能力

78

80C51 存储器结构与地址空间 80C51 存储器结构与地址空间



80C51系统总线(续)

A JE & A

80C51系统总线(续)

80C51系统总线

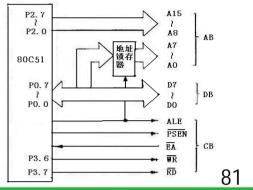
- 所谓总线,就是连接计算机各部件的一组公用信号线。使用并行总 线结构的80C51系列单片机,按其功能通常把系统总线分为三组,即 地址总线、数据总线和控制总线。具有总线的外部芯片都通过这三 组总线讲行扩展
- 地址总线 (Address Bus, 简写为AB)
 - ◆ 地址总线上传送的是地址信号,用于存储单元和I/0端口的选择。地址 总线是单向的, 地址信号只能由单片机向外发出
 - ◆ 地址总线的数目决定着可直接访问的存储单元的数目。如n位地址可访 问2n个存储单元,即通常所说的寻址范围为2ⁿ 地址单元
 - ◆ 80C51单片机存储器最多可扩展64KB, 即2^{16KB}, 地址总线有16条

■ 数据总线(Data Bus, 简写为DB)

- ◆ 数据总线用于在单片机与存储器或I/0端口之间传送数据
- ◆ 单片机系统数据总线的位数与单片机处理数据的字长一致,如80C51单 片机是8位字长, 所以数据总线的位数也是8位。数据总线是双向的, 可 以进行两个方向的数据传送
- 控制总线 (Control Bus, 简写为CB)
 - ◆ 控制总线实际上就是一组控制信号线,包括单片机发出的,以及从其它 部件传送给单片机的。对于一条具体的控制信号来说,其传送方向是单 向的,但是由不同方向的控制信号组合的控制总线则表示为双向
 - ◆ 由于采用总线结构形式, 因此大大减少了单片机系统中传输线的数目, 提高了系统的可靠性,增加了系统的灵活性。此外,总线结构也使扩展 易干实现,各功能部件只要符合总线规范,就可以很方便地接入系统, 实现单片机扩展

■ 80C51单片机并没有专用的地址线和数据线,采用I/0口线的复用技术

■ 80C51单片机并行总线扩展,往总线上"挂"存储或I/0接口芯片



80C51 总线、接口与扩展

79

2020/6/8

80C51 总线、接口与扩展

80

80C51 总线、接口与扩展

2020/6/8

80C51系统总线(续)

■ 以P0口的8位口线作地址/数据线

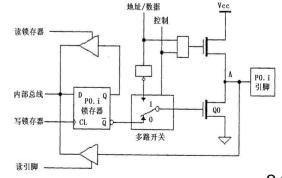
- ◆ 此处的地址是指系统的低8位地址。因为P0口线既作地址线使用又作为 数据线使用,具有双重功能,因此需采用复用技术,对地址和数据进行 分离,为此在构造地址总线时要增加一个8位锁存器。首先由锁存器暂 存并为系统提供8位地址,其后P0口线就作为数据线使用。
- ◆ 一般选择高电平或下降沿选通的锁存器作为地址锁存器, 常用的器件有 74LS273、74LS373。 八D锁存器74LS373的锁存允许信号G是电平锁存。 当G从高电平转为低电平时,将其输入端的数据锁存在输出端。当ALE为 高电平时,八D锁存器74LS373的输入和输出是透明的。当ALE出现下降 沿后, 八D锁存器74LS373的输出即为A0~A7, 这时P0口上出现的是数据 ,实现了地址低8位和数据线的分离。
- ◆ 实际上单片机P0口的电路逻辑已考虑了地址和数据复用的需要,口线电 路中的多路转接电路MUX及地址/数据控制就是为此目的而设计的

并行输入/输出端口

- 80C51单片机有4个双向并行的8位I/0口P0~P3
 - ◆ P0口为三杰双向口,可驱动8个LSTTL电路
 - ◆ P1、P2、P3口为准双向口(作为输入时,要把口线拉成高电平,故称为准 双向口),其负载能力为4个LSTTL电路
- P0□
 - ◆ 一个多功能的8位口,可以字节访问也可位访问,其字节访问地址为80H , 位访问地址为80H~87H
 - ◆ 位结构
 - 在访问外部存储器时, P0口是一个真正的双向数据总线口, 并分时 地送出地址的低8位和数据
 - P0口位结构,包括一个输出锁存器,两个三态缓冲器,一个输出驱动 电路和一个输出控制电路。其中输出驱动电路由一对FET(场效应管) 组成, 其输出状态受输出控制电路的控制

并行输入/输出端口(续)

■ P0口(续)



84

82

2020/6/8

80C51 总线、接口与扩展

2020/6/8

80C51 总线、接口与扩展

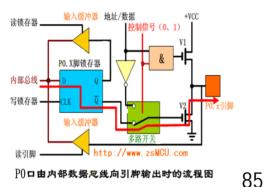
83

80C51 总线、接口与扩展



■ P0口(续)

◆ 做I/O端口使用 --- OUTPUT



80C51 总线、接口与扩展

■ P0口(续)

A jé x dj

◆ 做I/0端口使用 --- INPUT(续)

并行输入/输出端口(续)

- 在输入状态下,从锁存器和从引脚上读来的信号一般一致, 但也有例外。
- 例如, 当从内部总线输出低电平后, 锁存器Q=0, Q非=1, 场效应管T2开通,端口线呈低电平状态。此时无论端口线上 外接的信号是低电乎还是高电平, 从引脚读入单片机的信号 都是低电平, 因而不能正确地读入端口引脚上的信号。
- 又如, 当从内部总线输出高电平后, 锁存器Q=1, Q非=0, 场效应管T2截止。如外接引脚信号为低电平,从引脚上读入 的信号就与从锁存器读入的信号不同。
- 为此,8051单片机在对端口P0一P3的输入操作上,有如下约 定:凡属于读-修改-写方式的指令,从锁存器读入信号,其 它指令则从端口引脚线上读入信号 86

并行输入/输出端口(续)

■ P0□(续)

A JE & A

◆ 做 I / 0 端 口 使 用 --- INPUT (续)

- 先读口锁存器, 随之对读入的数据进行修改, 再写到端口上
- ●如执行指令ORL PO, #xH(PO=PO|x)时,则先把PO锁存器的内容 读入CPU, 然后与变量x按位进行逻辑'或'运算, 最后把 或'的结果送回P0口
- ●能使单片机产生这种读一修改一写操作的指令,其目的操作 数一般为某1/0口或口的某一位,这些指令是:位与(&)、位 或(|)、取反(~)、增1、减1等
- 例如:

ANL PO, #立即数

INC P1

DEC P3

CPL P2

87

80C51 总线、接口与扩展

2020/6/8

88

80C51 总线、接口与扩展

2020/6/8

并行输入/输出端口(续)

■ P0口(续)

2020/6/8

◆作为地址/数据复用口使用

- ●在访问外部存储器时P0口作为地址/数据复用口使用
- 在输出"地址/数据"信息时, V1、V2管交替导通, 负载能力 很强,可以直接与外设存储器相连,无须增加总线驱动器
- 在访问外部程序存储器时, P0口输出低8位地址信息后, 将变 为数据总线,以便读指令码(输入)
- 在读指令码或输入数据前, CPU自动向P0口锁存器写入0FFH, 破 坏了P0口原来的状态。因此,不能再作为通用的I/0端口。程 序中不能再含有以P0口作为操作数(包含源操作数和目的操 作数)的指令

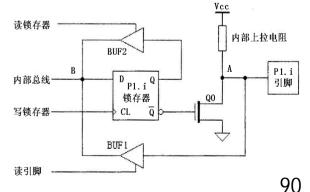
并行输入/输出端口(续)

■ P1 □

- ◆ P1口是一个标准的准双向口
- ◆P1口 包含输出锁存器、输入缓冲器BUF1、BUF2(读锁存器)以 及由 FET场效应管 Q0与上拉电阻组成的输入 / 输出驱动器
- ◆ P1口是一个8位口,可以字节访问也可按位访问,其字节访问地 址为90H, 位访问地址为90H~97H

并行输入/输出端口(续)

■ P1口(续)



80C51 总线、接口与扩展

80C51 总线、接口与扩展

2020/6/8

80C51 总线、接口与扩展



A IS A IS

并行输入/输出端口(续)

■ P1 □ (续)

- ◆特点
 - ●输出锁存,输出时没有条件

并行输入/输出端口(续)

- ●输入缓冲,输入时有条件:即当P1口作为输入口使用时,即 先将其锁存器写入'1',使FET截止,然后才能输入,具有 这种操作特点的输入/输出端口, 称为准双向I/0口。8051单 片机的P1、P2、P3都是准双向口
- 工作过程中无高阻状态,也就是该口不是输入态就是输出态
- ●单片机复位后,各个端口已自动地被写入了1,此时,可直接 作输入操作。如果在应用端口的过程中,已向P1一P3端口线 输出过0,则再要输入时,必须先写1后再读引脚,才能得到 正确的信息。此外, 随输入指令的不同, 端口也有读锁存器 与读引脚之分

并行输入/输出端口(续)

- P1 □ (续)
 - ◆P1口 字节操作和位操作
 - CPU对于P1口不仅可以作为一个 8位口(字节)来操作,也可 以按位来操作,有关字节操作的指令有

输出: MOV P1. A • P1←A Pl, #data : P1←#data MOV Pl, direct : P1←direct 输入: MOV A. P1 • A←P1 MOV direct, Pl : direct←P1

92

■ P1 □ (续)

- ◆P1口 字节操作和位操作(续)
 - 有关位操作的指令有:

置位、清除: SETB Pl.i : P1. i ←1 ; P1.i ←0 CLR Pl.i 输入、输出: MOV Pl.i. C ; P1. i ←CY MOV C, Pl.i : CY←Pl.i 判跳: IB Pl.i, rel : Pl. i=1, 跳转 ; Pl. i=1, 跳 转且Pl. i=0 JBC Pl.i, rel 逻辑运算: ; CY←(P1.i • CY) ANL C, Pl.i ORL C, Pl. i : CY ← (P1. i+CY)

P1. i中的i=0, ··· , 7

93

2020/6/8

80C51 总线、接口与扩展

91

88

2020/6/8

80C51 总线、接口与扩展

2020/6/8

80C51 总线、接口与扩展

并行输入/输出端口(续)

■ P1口(续)

◆P1口不仅可以以8位一组进行输入、输出操作,还可以逐位分别 定义各口线为输入线或输出线。例如:

ORL P1, #0 0 0 0 0 0 1 0 B

使P1. 1位口线输出1,而使其余各位不变

ANL P1, #1111101B

使P1. 1位线输出0,而使其余各位不变

并行输入/输出端口(续)

■ P1口(续)

- ◆ 读引脚操作和读锁存器操作
 - 在响应CPU输出的读引脚信号时,端口本身引脚的电平值通过缓冲器BUF1讲 入内部总线。这种类型的指令,执行之前必须先将端口锁存器置1,使A点 处于高电平,否则会损坏引脚,而且也使信号无法读出。这种类型的指令 有:

A, P1 : A←P1 direct, P1 ; direct←P1

- 在执行读锁存器的指令时, CPU首先完成将锁存器的值通过缓冲器BUF2读入 内部, 进行修改, 然后重新写到锁存器中去, 这就是"读一修改一写"指 令
 - 这种类型的指令包含所有口的逻辑操作(ANL、ORL、XRL)和位操作 (IBC、CPL、MOV、SETB、CLR等) 指令

并行输入/输出端口(续)

■ P1口(续)

- ◆P1口的多功能线
 - 在80C52中, P1.0和P1.1口线是多功能的, 即除作一般双向 I/0口线之外,这两根口线还具有下列功能:

P1.0一定时器/计数器2的外部输入端T2:

P1.1-定时器/计数器2的外部控制端T2EX

94

80C51 总线、接口与扩展

96

2020/6/8

95

2020/6/8

80C51 总线、接口与扩展

80C51 总线、接口与扩展

2020/6/8



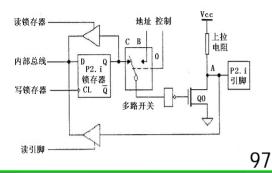
并行输入/输出端口(续)

并行输入/输出端口(续)

■ P2□

并行输入/输出端口(续)

◆ P2口是准双向口。 P2口是一个多功能的8位口,可以字节访问也 可位访问, 其字节访问地址为AOH, 位访问地址为AOH~A7H



■ P2口(续)

- ◆ P2口的功能
 - ●作I/0口使用时,P2口为一准双向口
 - ●作地址输出时,P2口可以输出程序存储器或片外数据存储器 的高8位地址,与P0输出的低地址一起构成16位地址线,从而 可分别寻址64KB的程序存储器或片外数据存储器。地址线是8 位一起自动输出的

■ P2口(续)

◆ P2口使用中注意的问题

- 由于P2口的输出锁存功能,在取指周期内或外部数据存储器读、写选通期 间,输出的高8位地址是锁存的,故无需外加地址锁存器
- 在系统中如果外接有程序存储器,由于访问片外程序存储器的连续不断的 取指操作,P2口需要不断送出高位地址,这时P2口的全部口线均不宜再作 1/0口使用。
- 无外接程序存储器而有片外数据存储器, P2口使用可分为两种情况:

若片外数据存储器的容量<256B: 可使用 "MOVX A, @Ri"及 "MOVX @ Ri, A"类指令访问片外数据存储器,这时P2口不输出地址,P2口仍可作 为1/0口使用:

若片外数据存储器的容量>256 B:使用 "MOVX A, @DPTR" 及 "MOVX@ DPTR, A"类指令访问片外数据存储器,P2口需输出高8位地址。在片外 数据存储器读、写选通期间,P2口引脚上锁存高8位地址信息,但是在选 通结束后,P2口内原来锁存的内容又重新出现在引脚上

98

99

2020/6/8

80C51 总线、接口与扩展

94

2020/6/8

80C51 总线、接口与扩展

A JE & A

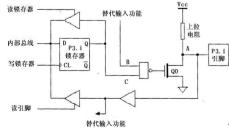
2020/6/8

80C51 总线、接口与扩展

并行输入/输出端口(续)

■ P3 □

- ◆ P3口是一个多功能的8位口,可以字节访问也可位访问,其字节访问地 址为BOH, 位访问地址为BOH~B7H
- ◆ 有两个输入缓冲器, 第二输入功能取自第一个缓冲器的输出端; I/0口 的通用输入信号取自第二个缓冲器的输出端



并行输入/输出端口(续)

■ P3口(续)

- ◆ P3口是一个多功能口
 - 可作I/0口使用,为准双向口。 既可以字节操作,也可以位操作: 既可以8位口操作,也可以逐位定义口线为输入线或输出线,既可以 读引脚,也可以读锁存器,实现"读一修改一输出"操作
 - 可以作为第二(替代)功能的输入、输出

第二输入功能:

P3.0 —— RXD, 串行输入口

P3.2--/INTO, 外部中断0的请求

---/INT1, 外部中断1的请求

P3.4 — T0, 定时器/计数器0外部计数脉冲输入

P3.5 — T0, 定时器/计数器0外部计数脉冲输入

第二输出功能:

P3.1 ——TXD, 串行输出口。

P3.6 ---/WR, 外部数据存储器写选通, 输出

P3.7 ——/RD, 外部数据存储器读选通, 输出

80C51存储器扩展

- 扩展ROM的地址与芯片内是否有程序存储器有关,如果没有片内程序存储器 ,扩展ROM的地址从0000H开始,如果有片内程序器,则扩展ROM的地址从 1000H开始
- 扩展RAM的地址,不管容量大小,都是从0000H开始
- 如何使用系统提供的地址线,通过适当连接,使系统中的一个存储单元只 唯一地对应一个地址,存储器编址分两个层次
 - ◆ 存储芯片的选择,实质就是如何产生芯片的"<mark>片选</mark>"信号
 - ◆ 芯片内部存储单元的编址, 是由芯片自身的译码电路完成的, 只需把存 储器芯片的地址引脚与相应的系统地址线直接连接即可

102

100

80C51 总线、接口与扩展

101

2020/6/8

80C51 总线、接口与扩展

2020/6/8

80C51 总线、接口与扩展



A ji k of

80C51存储器扩展(续)

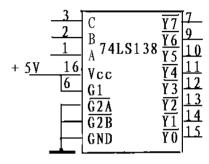
- "片选" 保证每次读或写时,只选中某一片存储器芯片或I/0接口芯片
- 通常把单片机系统地址笼统地分为低位地址和高位地址,芯片内部存储单 元地址译码使用低位地址,剩下的高位地址才作为芯片选择使用实际上
- 除了研究地址线的连接外,还讨论各存储器芯片在整个存储空间中所占据 的地址范围
- 常用的芯片选择方法(寻址方法)
 - ◆ 线选法: 利用单片机系统的位地址信号(如P2.7)作为某一片存储器芯 片或I/0接口芯片的"片选" 控制线,用于扩展芯片较少的场合
 - ◆ 译码法: 用译码器对高位地址线进行译码, 译码器的输出作为"片选" 控制线。常用译码器有3/8译码器74LS138、双2/4译码器74LS139、4/16 译码器74LS154等。它们的CMOS型芯片分别是74HC139、74HC138、 74HC154

80C51存储器扩展(续)

- 常用的芯片选择方法 --- 译码法(续)
 - ◆ 74LS139片中有两个2-4译码器
 - /G-使能端, 低电平有效。
 - A、B-选择端,即译码输入,控制译码输出的有效性。
 - Y0、/Y1、/Y2、/Y3-译码输出信号,低电平有效。
 - 74LS139对两个输入信号译码后得到4个输出状态
 - ◆ 74LS138译码器
 - G1、/G2A、/G2B: 使能端。当G1=1、/G2A=/G2B=0时,芯片使能
 - C、B、A: 译码器输入, 高电平有效
 - /Y0[~]/Y7: 译码器输出,低电平有效。正常情况下,只有一根输出是 低电平,其余输出都是高电平。当译码器输出作为单片机应用系统 中外扩芯片的片选控制线时,保证每次读或写时只选中一个芯片

80C51存储器扩展(续)

- 常用的芯片选择方法 --- 译码法(续)
 - ◆ 74LS138译码器(续)



105

103

2020/6/8

80C51 总线、接口与扩展

104

80C51 总线、接口与扩展

2020/6/8

80C51 总线、接口与扩展

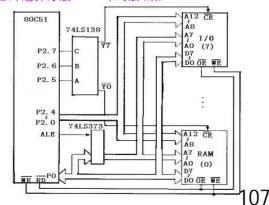
80C51存储器扩展(续)

■ 常用的芯片选择方法 --- 译码法(续)

- ◆ 部分地址线参加译码时, 称为部分地址译码, 这时芯片的地址会有重叠 。16根地址线全部参加译码的, 称为全地址译码。
- ◆ 下图示意的是通过3/8译码器74LS138获得64K地址。图中3/8译码器 74LS138已经使能,其输出由C、B、A的状态决定,作为各个扩展芯片的 片选控制信号,加上芯片本身所具有的地址线共同决定每一个存储单元 或I/0端口的地址,全地址译码的地址是唯一的

80C51存储器扩展(续)

■ 常用的芯片选择方法 --- 译码法(续)



80C51存储器扩展(续)

- ROM扩展(续) --- 程序存储器扩展时的总线功能和操作时序
 - ◆ /EA为片外程序存储器读选择信号。正常运行时,该引脚不能浮空,根据连接电 平的不同,单片机有两种取指过程。
 - ◆ 当/EA=1时,80C51单片机所有片内程序存储器有效
 - 当程序计数器PC运行于片内程序存储器的寻址范围内(对80C51/87C51/ 89C51为0000H~0FFFH, 共4KB; 对80C52 / 87C52 / 89C52为0000H~1FFFH, 共8KB) 时,PO口、P2口及/PSEN线没有信号输出;当PC的值超出上述范围 后,PO口、P2 口及/PSEN线才有信号输出
 - ◆ 当/EA=0时,80C51单片机所有片内程序存储器无效,只能访问片外程序存储器 , 使用如下的信号
 - P0口:分时输出程序存储器的低8位地址和8位数据
 - ALE线:输出,在ALE的下降沿时,PO口上出现稳定的程序存储器的低8位地 址,用ALE信号锁存这低8位地址
 - P2口: 在整个取指周期中,输出稳定的程序存储器的高8位地址
 - /PSEN线: 输出,低电平有效。在ALE的下降沿之后,/PSEN由高变为低,片 外程序存储器的内容(指令字)送到 PO口,而后在/PSEN的上升沿将指令 字送入指令寄存器。/PSEN信号作为片外程序存储器的"读" 108号

106

2020/6/8

80C51 总线、接口与扩展

80C51 总线、接口与扩展

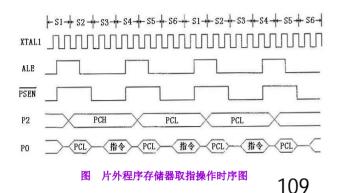
80C51 总线、接口与扩展

80C51存储器扩展(续)

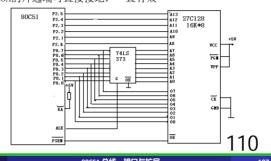
80C51存储器扩展(续)

80C51存储器扩展(续) --- RAM扩展

■ ROM扩展(续) --- 程序存储器扩展时的总线功能和操作时序(续)



- ROM扩展(续) --- 程序存储器扩展举例
 - ◆ 扩展16K *8位片外程序存储器
 - 在电路中/EA是接高电平的。27128A是16KB容量的EPROM, 所以用到 了14根地址线A0~A13。系统中只扩展了一片程序存储器,所以 27128A的片选端可直接接地,一直有效



■ 随机存储器概述

- ◆ 数据存储器就是随机存储器, 简称RAM (RAM: Random Access Memory) 。与ROM不同,对RAM可以进行读写两种操作
- ◆ 按工艺, RAM分为MOS型和双极型。MOS型集成度高, 功耗低, 价格便宜 ,但速度较慢。而双极型则正好相反。在单片机系统中使用的大多数是 MOS型的RAM,它们的输入输出信号能与TTL相兼容,因此在扩展中信号 连接是很方便的
- ◆ 按工作方式, RAM分为静态 (SRAM) 和动态 (DRAM) 。静态RAM只要加上 电源,所存信息就能可靠保存。而动态RAM使用的是动态存储单元,需 要不断进行刷新以便周期性地再生,才能保存信息。动态RAM集成密度 大、功耗低、价格便宜,但需增加刷新电路,因此只适用于较大的系统 , 而在单片机系统中很少使用
- ◆ 按扩展方式, RAM分为并行RAM和串行RAM

111

80C51 总线、接口与扩展

2020/6/8

80C51 总线、接口与扩展

106

2020/6/8

80C51 总线、接口与扩展

A JE & A

114

80C51存储器扩展(续) --- RAM扩展(续)

80C51存储器扩展(续) --- RAM扩展(续)

80C51存储器扩展(续) --- RAM扩展(续)

■ 片外数据存储器的读、写操作时序

◆ 80C51单片机,对片外数据存储器读、写操作的指令有以下四条:

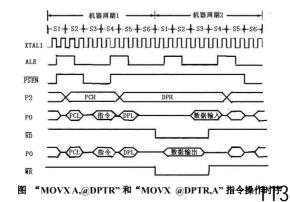
MOVX A, @Ri; 片外RAM→(A), 读(/RD)操作 MOVX @Ri, A; (A)→片外RAM, 写(/WR)操作 由于@Ri只能提供8位地址,仅能直接扩展256个字节的片外RAM

MOVX A, @DPTR ; 片外RAM→(A), 读操作 MOVX @DPTR, A; (A)→片外RAM, 写操作

由于@DPTR能提供16位地址,因此,可以扩展 64 KB的片外 RAM

这四条指令都是双机器周期指令

■ 片外数据存储器的读、写操作时序(续)



- 片外数据存储器的读、写操作时序(续)
 - ◆ 从图中可以看出, 执行该组指令时, 机器周期1为取指周期, 在取指周 期的S5状态时,ALE的下降沿,在P0总线上出现的是数据存储器的低8位 地址,即DPL;在P2口上出现的是数据存储器的高8位地,即DPH。
 - ◆ 取指操作后,直至机器周期2的S3状态之前,/PSEN一直维持高电平。而 在机器周期2的S1与S2状态之间的ALE不再出现。
 - ◆ 执行"MOVX A, @DPTR"时, 从机器周期2开始到S3状态, /RD出现低电 平。此时允许将片外数据存储器的数据送上PO口,在/RD的上升沿将数 据读入累加器A。数据为输入。执行"MOVX @DPTR, A"时, 从机器周 期2开始到S3状态,/WR出现低电平。此时P0口上将送出累加器A的数据 ,在/WR的上升沿将数据写入片外数据存储器中。数据为输出
 - ◆ 总之,此时P0口为地址、数据复用总线:P2口在机器周期1的S4状态之 后出现锁存的高8位地址(DPH);用控制线来控制数据总线上的数据传 输方向:/RD有效时数据为输入,/WR有效时数据为输出

112

80C51 总线、接口与扩展 2020/6/8 80C51 总线、接口与扩展



80C51存储器扩展(续) --- RAM扩展(续)

80C51存储器扩展(续) --- RAM扩展(续)

数据存储器则使用/RD和/WR分别作为读、写选通信号

80C51存储器扩展(续) --- RAM扩展(续)

- 数据存储器扩展用典型芯片
 - ◆ 数据存储器扩展常用随机存储器芯片,用的较多的是Intel的 $6116(2K \times 8)$, $6264(8K \times 8)$, $62128(16K \times 8)$, $62256(32K \times 8)$, 62512(64K×8) 等型号,它们都是SRAM, CMOS工艺,因此具有低功耗的 特点。在维持状态下只需几个微安电流,很适宜作需断电保护或需长期 低功耗状态下工作的存储器。另外EEPROM除可用作程序存储器扩展外, 还可作为数据存储器扩展。
 - ◆ 6116 SRAM引脚, 说明如下:

A10~A0: 地址线 D7∼D0: 数据线

/WE: 写选通信号

/CE: 片选信号

/OE: 数据输

出允许信号

6116共有四种工作方式:未选中、禁止、读出、写入

◆ 单片数据存储器扩展

■ RAM扩展举例

● 使用一片6116实现了2KB RAM扩展。在扩展连接中,以/RD信号接芯 片的/OE端,以/WR信号接/WE端,讲行RAM芯片的读写控制。由于假 定系统只有一片6116,因此没有使用片选信号,而把/CE端直接接地 。6116的地址范围是0000~07FFH

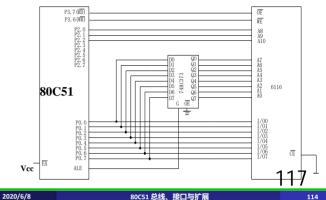
◆ 片外数据存储器一般由随机存取存储器组成,最大可扩展64KB。数据存

储器扩展与程序存储器扩展在数据线、地址线的连接上是完全相同的。

所不同的只在于控制信号,程序存储器使用/PSEN作为读选通信号,而

■ RAM扩展举例(续)

◆ 单片数据存储器扩展(续)



116

115

80C51 总线、接口与扩展

2020/6/8

80C51 总线、接口与扩展

112

jf +

80C51存储器扩展(续) --- RAM扩展(续)

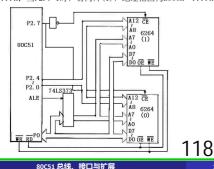
80C51存储器扩展(续) --- RAM扩展(续)

80C51存储器扩展(续) --- 综合扩展

■ RAM扩展举例(续)

2020/6/8

- ◆ 线选法多片存储器扩展
 - 如图所示的是用两片6264扩展16K*8位片外数据存储器的电路。
 - 在图中,采用线选法寻址。用一根口线P2.7来寻址: 当 P2.7=0时,访问片(0),地 址范围为6000H~7FFFH; 当P2.7=1时,访问片(1),地址范围为E000H~FFFFH

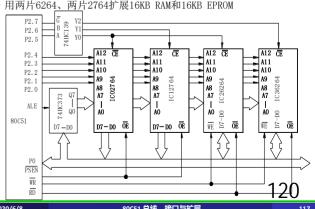


- RAM扩展举例(续)
 - ◆ 译码法扩展

2020/6/8

■ 同时扩展程序存储器和数据存储器

◆ 用两片6264、两片2764扩展16KB RAM和16KB EPROM



119

80C51 总线、接口与扩展 80C51 总线、接口与扩展



80C51存储器扩展(续) --- 综合扩展

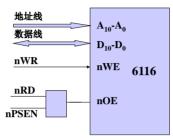
80C51存储器扩展(续) --- 综合扩展

■ 扩展既可读又可写的程序存储器

◆ 在仿真器或实验器等单片机系统中, 为了程序仿真和实验的需要, 希望 有既能运行程序又能修改程序的存储器,这就是既可读又可写的程序存 储器。这种存储器是通过把RAM存储芯片经过特殊的连接而实现的

80C51存储器扩展(续) --- 综合扩展

- ◆ 程序存储器与数据存储器的扩展连接在数据和地址线上没有什么区别, 不同的在于控制信号。程序存储器使用/PSEN作读选通信号,而数据存 储器使用/RD作读选通信号,若把这两个读选通信号通过与门输出后, 再作为RAM存储芯片的读选通信号,即可达可改写程序存储器的目的
- 扩展既可读又可写的程序存储器(续)
 - ◆ 例: 把6116 RAM改造为既可读又可写的程序存储器
 - 由RAM改造而成的程序存储器虽然可以运行和修改程序, 但是不能掉 电保存程序,与真正的程序存储器有所毕竟不同



- 扩展既可读又可写的程序存储器(续)
 - ◆ E2PROM也可按这里可读写存储器的方法进行改造,不仅解决了程序调试 的问题,而且也解决了程序的保存问题
 - ◆ 例: 用2864 E2PROM扩展8KB程序存储器
 - 图中2864的片选端/CE与高位地址线P2.7(A15)连接, P2.7=0时才能 选中2864, 因此2864的地址范围为0000H~1FFFH, 这8KB存储器既可 用作程序存储器,又可用作数据存储器(掉电时,数据不易失)

122

123

2020/6/8

80C51 总线、接口与扩展

118

2020/6/8

80C51 总线、接口与扩展

119

80C51 总线、接口与扩展

121

80C51存储器扩展(续) --- 综合扩展

jf x

80C51存储器扩展(续) 综合扩展

■ 扩展既可读又可写的程序存储器(续) --- 例: 用2864 E2PROM扩展8KB程序存储器(续)

80C51存储器扩展(续) --- 综合扩展



◆ 例: 用2864 E2PROM扩展8KB程序存储器(续)

写16字节E2PROM 2864A的子程序WR1如下: 被写入的数据取白源数据区 子程序入口参数为 R1=写入的字节数(10H)

RO= E2PROM的低位地址 P2= E2PROM的高位地址 DPTR=源数据区首址 A. @DPTR · 取数据 R4. A

: 写入2864A TNC DPTR INC RO RO, #00, NEXT ; 低位地址指针未满, 转移

; 否则, 高位指针加1 TNC P2 NEXT: DINZ R1, WR1 ;数据未写完,转移

: 数据查询与最后一个字节的原始数据比较 CHECK · MOVX A @RO XRL A R4 ACC. 7, CHECK; 最高位不同, 再查

; 最高位相同, 一页写完

■ 扩展既可读又可写的程序存储器(续)

- ◆ 例: 用2864 E2PROM扩展8KB程序存储器(续)
 - 2864片内设置有16字节的页缓冲器,并提供了页面写入方式。把数 据写入2864的存储单元可分两步完成:第一步,在软件控制下把数 据写入页缓冲器,此过程称为"页加载周期";第二步,8264在内 部定时电路控制下,把页缓冲器的内容送到地址指定的E2PROM单元 内,称为"页存储周期"
 - 在软件中,可用数据查询方式检测写操作中"页存储周期"是否完 成。"页存储"期间,如果对2864执行读操作,那么读出的是最后 写入的字节, 若芯片的转储工作未完成, 则读出数据的最高位是原 来写入字节最高位的反码。据此,CPU判断芯片的编程是否结束。如 果CPU读出的数据与写入的数据相同,表示芯片已完成编程,CPU可 继续向芯片加载下一页数据

WR WE P2. 7 P2. 28C64 80C31 G OE

> 124 RET 121 2020/6/8

80C51 总线、接口与扩展

80C51 总线、接口与扩展

126

2020/6/8

RD PSEN

ALE

80C51 总线、接口与扩展

: 数据暂存, 供查询

2020/6/8