



芯海科技  
CHIPSEA

# CSU32P20\_ASM\_DEMO\_说明文档

REV 1.1

芯海科技（深圳）股份有限公司

地 址：深圳市南山区蛇口南海大道1079号花园城数码大厦A座9楼

电 话：+(86 755)86169257      传 真：+(86 755)86169057

网 站：www.chipsea.com      邮 编：518067

微信号：芯海科技



版本历史

历史版本	修改内容	版本日期
Rev 1.0	首版发布	2019.02.27
Rev 1.1	增加 adc_offset demo 说明	2019.6.14

目 录

版本历史..... 2

目 录..... 3

1 GPIO..... 6

1.1 相关程序说明 ..... 6

1.2 硬件说明 ..... 6

1.3 测试工具 ..... 6

1.4 测试方法 ..... 6

1.5 测试结果 ..... 6

2 TIMER0 ..... 7

2.1 相关程序说明 ..... 7

2.2 硬件说明 ..... 7

2.3 测试工具 ..... 7

2.4 测试方法 ..... 7

2.5 测试结果 ..... 7

3 TIMER 2 ..... 8

3.1 相关程序说明 ..... 8

3.2 硬件说明 ..... 8

3.3 测试工具 ..... 8

3.4 测试方法 ..... 8

3.5 测试结果 ..... 8

4 TIMER 3 ..... 9

4.1 相关程序说明 ..... 9

4.2 硬件说明 ..... 9

4.3 测试工具 ..... 9

4.4	测试方法 .....	9
4.5	测试结果 .....	9
<b>5</b>	<b>ADC .....</b>	<b>10</b>
5.1	相关程序说明 .....	10
5.2	硬件说明 .....	10
5.3	测试工具 .....	10
5.4	测试方法 .....	10
5.5	测试结果 .....	10
<b>6</b>	<b>SLEEP .....</b>	<b>11</b>
6.1	相关程序说明 .....	11
6.2	硬件说明 .....	11
6.3	测试工具 .....	11
6.4	测试方法 .....	11
6.5	测试结果 .....	11
<b>7</b>	<b>HALT .....</b>	<b>12</b>
7.1	相关程序说明 .....	12
7.2	硬件说明 .....	12
7.3	测试工具 .....	12
7.4	测试方法 .....	12
7.5	测试结果 .....	12
<b>8</b>	<b>WDT .....</b>	<b>13</b>
8.1	相关程序说明 .....	13
8.2	硬件说明 .....	13
8.3	测试工具 .....	13
8.4	测试方法 .....	13
8.5	测试结果 .....	13
<b>9</b>	<b>WWDT .....</b>	<b>14</b>

9.1	相关程序说明 .....	14
9.2	硬件说明 .....	14
9.3	测试工具 .....	14
9.4	测试方法 .....	14
9.5	测试结果 .....	14
<b>10</b>	<b>LCD.....</b>	<b>15</b>
10.1	相关程序说明 .....	15
10.2	硬件说明 .....	15
10.3	测试工具 .....	15
10.4	测试方法 .....	15
10.5	测试结果 .....	15
<b>11</b>	<b>TABLE .....</b>	<b>16</b>
11.1	相关程序说明 .....	16
11.2	硬件说明 .....	16
11.3	测试工具 .....	16
11.4	测试方法 .....	16
11.5	测试结果 .....	16
<b>12</b>	<b>ADC_OFFSET.....</b>	<b>17</b>
12.1	相关程序说明 .....	17
12.2	硬件说明 .....	17
12.3	测试工具 .....	17
12.4	测试方法 .....	17
12.5	测试结果 .....	17

## 1 GPIO

### 1.1 相关程序说明

函数名: Demo\_gpio\_flip, 一个 IO 的输出电平由另外一个 IO 的输入电平控制。选项字节关闭 PT1\_0 默认下拉电阻, 选择字节设置下拉后程序不能关闭下拉。

### 1.2 硬件说明

PT1\_0 根据 PT1\_1 电平变化, PT1\_1 输入高电平, PT1\_0 输出高电平; 反之, PT1\_1 输入低电平, PT1\_0 输出低电平。

### 1.3 测试工具

仿真器、示波器。

### 1.4 测试方法

步骤 1: 仿真器连接到 PC 的 USB 口, IDE 中点击 debug 进入仿真。

步骤 2: 使用短接线将 PT1\_1 短接到 GND 或者断开, 示波器观察 PT1\_0 的电平变化。

### 1.5 测试结果

当 PT1\_1 悬空, 即输入高电平 (内部上拉), PT1\_0 输出高电平; 反之, 当 PT1\_1 接 GND, 即输入低电平, PT1\_0 输出低电平。

## 2 Timer0

### 2.1 相关程序说明

初始化设置定时器 0 时钟源为 CPUCLK，选项字节设置 CPUCLK=4MHz，定时时间 8mS，在定时器 0 中断里翻转 IO 电平。选项字节关闭 PT1\_0 默认下拉电阻，选择字节设置下拉后程序不能关闭下拉。

### 2.2 硬件说明

PT1\_0 输出脉宽 8mS 的方波。

### 2.3 测试工具

仿真器、示波器。

### 2.4 测试方法

步骤 1：仿真器连接到 PC 的 USB 口，IDE 中点击 debug 进入仿真。

步骤 2：示波器测试 PT1\_0 的电平变化。

### 2.5 测试结果

PT1\_0 输出脉宽 8mS 的方波。

## 3 Timer 2

### 3.1 相关程序说明

初始化设置定时器 2 时钟源为 MCK，选项字节设置 MCK=16MHz，PT5\_0 或 PT3\_1 输出 PWM 波形，频率固定 50Hz，调节占空比从 20%到 80%变化。

### 3.2 硬件说明

PT5\_0 或 PT3\_1 输出 PWM 波形。

### 3.3 测试工具

仿真器、示波器。

### 3.4 测试方法

步骤 1: 仿真器连接到 PC 的 USB 口，IDE 中点击 debug 进入仿真。

步骤 2: 示波器测试 PT5\_0 或 PT3\_1 的波形。

### 3.5 测试结果

PT5\_0 或 PT3\_1 输出 50Hz 的 PWM 波形，占空比从 20%到 80%变化。



## 4 Timer 3

### 4.1 相关程序说明

初始化设置定时器 3 时钟源为 MCK，选项字节设置 MCK=16MHz，PT5\_0、PT5\_1 或 PT3\_5、PT1\_0 输出互补 PWM 波形，频率固定 200KHz，调节占空比从 20%到 80%变化，死区时间设置为 250nS。

### 4.2 硬件说明

PT5\_0、PT5\_1 或 PT3\_5、PT1\_0 输出互补 PWM 波形。

### 4.3 测试工具

仿真器、示波器。

### 4.4 测试方法

步骤 1: 仿真器连接到 PC 的 USB 口，IDE 中点击 debug 进入仿真。

步骤 2: 示波器测试 PT5\_0、PT5\_1 或 PT3\_5、PT1\_0 的波形。

### 4.5 测试结果

PT5\_0、PT5\_1 或 PT3\_5、PT1\_0 输出 200Hz 的 PWM 波形，占空比从 20%到 80%变化。

## 5 ADC

### 5.1 相关程序说明

初始化设置采样时间、IO 口模式，选项字节设置 MCK=16MHz，CPUCLK=4MHz。参考电源选择 VDD 测量 AIN0 和内部参考电压，参考电源选择内部参考电压测量 AIN1，参考电源选择 PT3.0 外接参考电源测量 AIN2。切换参考电压需要延时 40uS 后再进行 ADC 转换。

### 5.2 硬件说明

无。

### 5.3 测试工具

仿真器、可编程电源。

### 5.4 测试方法

步骤 1：仿真器连接到 PC 的 USB 口，IDE 中点击 debug 进入仿真。

步骤 2：在每个转换结束处设置断点，观察采样值是否正确。

### 5.5 测试结果

输入通道	参考电压	采样理论值	测量值	误差
AIN8:1.4V	VDD:3.3V	0x06c9	0x06c3	-0.34%
AIN0:1.65V	VDD:3.3V	0x0800	0x07f5	-0.54%
AIN1:0.7V	内部参考：1.4V	0x0800	0x07f6	-0.49%
AIN2:1.0V	PT3.0 外部输入：2V	0x0800	0x07fe	-0.01%

## 6 SLEEP

### 6.1 相关程序说明

进入休眠模式前，将需要输出的 IO 设置为与外围器件一样的电平（输出口不要开启上拉电阻），需要输入的 IO 口设置为输入（外围电路是高电平的不能开启下拉电阻，外围电路是低电平的不能开启上拉电阻），使能外部中断 1，关闭 ADC 使能、定时器使能，关闭 WDT 时钟。

当 PT1\_3 为高电平时进入休眠，PT1\_3 为低电平时唤醒。

注：1、WDT 功耗 5uA，如休眠不关闭 WDT，则不能关闭 WDT 时钟；

2、悬空的 IO 设置为输入会增加休眠功耗，需要开启上拉或者下拉电阻。悬空的 IO 也可以设置为输出口；

### 6.2 硬件说明

无。

### 6.3 测试工具

电源、万用表。

### 6.4 测试方法

步骤 1：将 hex 文件烧录 CSU32P20。

步骤 2：将万用表调到电流档串接在 VDD 或者 VSS，在 VDD 与 VSS 之间加 5V 电源，当 PT1\_3 电平为高时观察万用表测量值。

### 6.5 测试结果

当 PT1\_3 电平为高时，电流为 0.4uA，低于 1uA，说明进入了 sleep；当 PT1\_3 电平为低时，电流为 126uA，说明退出了 sleep。

## 7 Halt

### 7.1 相关程序说明

设置定时器 0 溢出时间 1S，开启定时器 0，使能中断。进入停止模式前，将 PT1.1 电平由高电平变为低电平，进入停止模式，定时器 0 溢出中断唤醒停止模式，将 PT1.1 电平由低电平变为高电平。

### 7.2 硬件说明

无。

### 7.3 测试工具

仿真器、示波器。

### 7.4 测试方法

步骤 1：仿真器连接到 PC 的 USB 口，IDE 中点击 debug 进入仿真。

步骤 2：用示波器测量 PT1.1 低电平脉宽时间。

### 7.5 测试结果

PT1.1 低电平脉宽时间为 1.008S。

## 8 WDT

### 8.1 相关程序说明

设置看门狗溢出时间 16mS。进入主循环前，将 PT1.0 置高电平，在主循环喂狗，则 PT1.0 一直是高电平；如果不喂狗，则 16mS 后看门狗溢出，32P20 复位，PT1.0 每 16mS 被拉低一次。选项字节关闭 PT1\_0 默认下拉电阻，选择字节设置下拉后程序不能关闭下拉。

### 8.2 硬件说明

无。

### 8.3 测试工具

仿真器、示波器。

### 8.4 测试方法

步骤 1: 仿真器连接到 PC 的 USB 口，IDE 中点击 debug 进入仿真。

步骤 2: 用示波器测量 PT1.0 是否一致是低电平。

步骤 3: 将主循环的喂狗屏蔽，PT1.0 每 16mS 被拉低一次。

### 8.5 测试结果

在主循环喂狗，则 PT1.0 一直是高电平；如果不喂狗，PT1.0 每 16mS 被拉低一次。

## 9 WWDT

### 9.1 相关程序说明

设置窗看门狗计数值为 0x7F，窗口值为 0x5E。进入主循环前，将 PT1.0 置高电平，在主循环查询计数值在窗口内后喂狗，则 PT1.0 一直是高电平；如果不喂狗，则计数值由 0x40 变为 0x3F 触发复位，PT1.0 被拉低。选项字节关闭 PT1\_0 默认下拉电阻，选择字节设置下拉后程序不能关闭下拉。

### 9.2 硬件说明

无。

### 9.3 测试工具

仿真器、示波器。

### 9.4 测试方法

步骤 1：仿真器连接到 PC 的 USB 口，IDE 中点击 debug 进入仿真。

步骤 2：用示波器测量 PT1.0 是否一致是低电平。

步骤 3：将主循环的喂狗屏蔽，每次复位 PT1.0 都被拉低。

### 9.5 测试结果

在主循环喂狗，则 PT1.0 一直是高电平；如果不喂狗，每次复位 PT1.0 都被拉低。

## 10 LCD

### 10.1 相关程序说明

从 00%到 100%每秒增加 1%循环显示，com 口扫描一个循环的频率 62.5Hz。

### 10.2 硬件说明

1/2bias 显示屏。



## 11 TABLE

### 11.1 相关程序说明

设置一个 16 位的数据表，数据表地址可以由编译器自动分配也可以指定地址。将数据表保存的 16 位数据一一读出存储在变量里。

### 11.2 硬件说明

无。

### 11.3 测试工具

仿真器。

### 11.4 测试方法

步骤 1：仿真器连接到 PC 的 USB 口，IDE 中点击 debug 进入仿真。

步骤 2：读取每个地址的数据读取后观察变量的值。

### 11.5 测试结果

读取的值和数据表的值一致。



## 12 ADC\_OFFSET

### 12.1 相关程序说明

从 F001H 读出 ADC 失调电压校正值，将数据由高位至地位从 PT1.1 口输出，将 PT1.0 口设置成时钟输出

### 12.2 硬件说明

无。

### 12.3 测试工具

仿真器，示波器。

### 12.4 测试方法

步骤 1：仿真器连接到 PC 的 USB 口，IDE 中点击 debug 进入仿真。

步骤 2：用示波器测试 PT1.0 口与 PT1.1 口波形，读出校正数据

### 12.5 测试结果

可根据时钟信号读出芯片校正电压值