

模块 13

实验：定时器



实验：定时器

13.0 目标

本实验的目的是开发转动电机所需的软件。该软件可以使用脉冲宽度调制（PWM）改变传送到每个电机的电功率。在这个模块中，

1. 您将学习 MSP432 Timer_A 模块。
2. 您将配置 Timer A0 以创建两个 PWM 输出。
3. 您将配置 Timer A1 以创建一个额外的周期性中断。
4. 您将开发用于移动的低层机器人命令。

小知识： PWM 是微控制器影响其世界的有用且高效的手段。它是有用的，因为将定时器重载值设置为 10000 将创建一个精度约为 14 位的输出。它是高效的，因为定时器和数字开关电路（DRV8838）的成本远低于等效的模拟放大器。

13.1 入门

13.1.1 从下面的软件工程起步

浏览以下 3 个项目：

PWMSine (使用 PWM 和定时器来创建正弦波输出)

PeriodicTimerA0Ints (使用 Timer_A0 创建一个周期性中断)

Lab13_Motors (本实验开始项目)

注： 您将无法在机器人上运行 PeriodicTimerA0Ints 项目，因为该项目使用 Timer_A0，您需要将 Timer_A0 用于机器人的两个 PWM 输出。

13.1.2 参考资料

- MSP432P4xx Technical Reference Manual, Timer_A (SLAU356)
- MSP432P401R Datasheet, msp432p401m.pdf (SLAS826)
- JN_PDB_SHEET02 - POWER_DISTRI.pdf Data sheet for power board
- RSLK China basic construction guide 2.1.pdf How to build the robot
- drv8838.pdf Data sheet for the H-bridge driver

13.1.3 阅读材料

- Volume 1 Sections 8.7, and 9.7
- Embedded Systems: Introduction to the MSP432 Microcontroller", Volume 2 Sections 6.2, 6.3, and 6.5
- Embedded Systems: Real-Time Interfacing to the MSP432 Microcontroller"

13.1.4 本实验所需的组件

数量	组件描述	制造商	型号
1	MSP-EXP432P401R LaunchPad	TI	MSP-EXP432P401R
1	Al Alloy Chassis & Structure	JiangNiu	
1	Power Module	JiangNiu	JN_PDB_2RSLKB01_V0.2
1	Motor Driver Module	JiangNiu	JN_MD_2RSLKB01_V0.2
1	Interface Board For Exp432	JiangNiu	JN_IB_2RSLKB01_V0.2
2	φ 65mm Rubber wheel	JiangNiu	
2	Encoder+DC Motor+Gear reducer	JiangNiu	
1	Battery Holder	JiangNiu	
2	Li Battery 3.7V 320mAh	JiangNiu	
1	Charger(110-220)	JiangNiu	



实验：定时器

13.1.5 所需 实验设备

示波器 (至少 10 kHz 采样的 1 个或 2 个通道)
逻辑分析仪 (至少 10 kHz 采样的 4 个通道)

13.2 系统设计要求

本实验的第一个目标是编写可以调整两个电机的运行功率的软件。您将在 P2.6 和 P2.7 引脚上创建 PWM 输出，这些引脚连接到电机驱动板的 PWML 和 PWMR (EN 输入到 DRV8838)。两个输出的周期应固定为 7 ms (70 Hz)。但是，软件应该能够独立地将 EN 引脚的占空比设置为 0 到 9999 (0 到 99.99%)。在 70 Hz 时，电机不会响应个别的高点和低点；相反，电机将响应平均水平。更具体地说，传递的功率将是

$$P = V * I * \text{duty}/10000; \quad 0 \leq \text{duty} \leq 9999$$

其中 **V** 是电压，**I** 是电流，如实验 12 中先前所测量的。

本实验的第二个目标是使用 Timer_A1 创建一个额外的周期性中断。高级别主程序将在运行时使用函数指针初始化此周期性中断，从而提供抽象和代码重用。

与实验 12 相似，本实验的结果是一个直线行驶的系统，直到其中一个碰撞传感器检测到碰撞。但是，与实验 12 相反，此解决方案将需要非常少的软件开销。

13.3 实验准备

与实验 12 相同。请参阅 DRV8838 的数据手册，了解这六个信号的软件输出值如何影响电机性能，如图 1 所示。

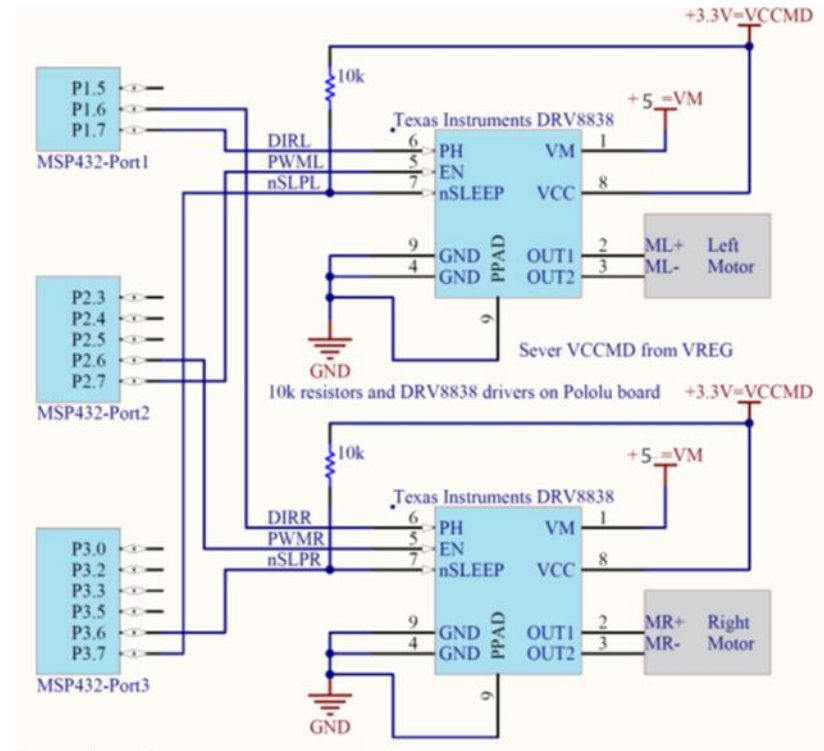


图 1. 接口电路



实验：定时器

LaunchPad	MDPDB	DRV8838	描述
P1.6	DIRR	PH	Right Motor Direction
P3.6	nSLPR	nSLEEP	Right Motor Sleep
P2.6	PWMR	EN	Right Motor PWM
P1.7	DIRL	PH	Left Motor Direction
P3.7	nSLPL	nSLEEP	Left Motor Sleep
P2.7	PWML	EN	Left Motor PWM

13.4 实验步骤

13.4.1 底层软件驱动

将实验 12 中内置的软件功能套件替换为使用 Timer_A0 创建两个 PWM 输出的功能。这套功能将控制机器人上的两个轮子。当 PWM 激活时，两个电机的 PWM 均为 70Hz (7ms)，但具有独立的占空比。驱动程序的原型是：

void Motor_Init(void);

Initializes the 6 lines and Timer A0 and puts driver to sleep
Returns right away

void Motor_Stop(void);

Stops both motors, puts driver to sleep
Returns right away

void Motor_Forward(uint16_t leftDuty, uint16_t rightDuty)

Drives left motor forward at **leftDuty** (0 to 9999)
Drives right motor forward at **rightDuty** (0 to 9999)
The motors run until software issues another command
Returns right away

void Motor_Backward(uint16_t leftDuty, uint16_t rightDuty)

Drives left motor backward at **leftDuty** (0 to 9999)
Drives right motor backward at **rightDuty** (0 to 9999)
The motors run until software issues another command
Returns right away

void Motor_Left(uint16_t leftDuty, uint16_t rightDuty)

Drives left motor backward at **leftDuty** (0 to 9999)
Drives right motor forward at **rightDuty** (0 to 9999)
The motors run until software issues another command
Returns right away

void Motor_Right(uint16_t leftDuty, uint16_t rightDuty)

Drives left motor forward at **leftDuty** (0 to 9999)
Drives right motor backward at **rightDuty** (0 to 9999)
The motors run until software issues another command
Returns right away

13.4.2 电机测试

将电压表放在电机驱动板的 VM 线 (+5) 和 VREG 线 (+3.3) 上，同时调试电机软件。将机器人放在木块上，使车轮不接触地面，并使用 **Program13_1** 等程序测试低级别电机功能。

```
// Driver test
void TimedPause(uint32_t time){
    Clock_Delay1ms(time); // run for a while and stop
    Motor_Stop();
    while(LaunchPad_Input()==0); // wait for touch
    while(LaunchPad_Input()); // wait for release
}
int Program13_1(void){
    Clock_Init48MHz();
    LaunchPad_Init(); // built-in switches and LEDs
    Bump_Init(); // bump switches
    Motor_Init(); // your function
    while(1){
        TimedPause(4000);
        Motor_Forward(7500,7500); // your function
        TimedPause(2000);
        Motor_Backward(7500,7500); // your function
        TimedPause(3000);
        Motor_Left(5000,5000); // your function
        TimedPause(3000);
        Motor_Right(5000,5000); // your function
    }
}
```



实验：定时器

将机器人放在地面并尝试调整 *Motor_Forward* 和 *Motor_Backward* 调用中的每个 7500 参数，以便机器人沿直线移动。将调用中的 5000 参数调整为 *Motor_Left* 和 *Motor_Right*，并暂停 3000 参数，使机器人旋转 90 度。

注： 调整这些参数以运行机器人开环几乎是不可能的。要求您尝试解决一个不可能的问题将激发对输入的需求并创建一个闭环控制器。

13.4.3 周期性中断

编写软件以使用 *Timer_A1* 创建额外的周期性中断。如果使用 12 MHz SMCLK 并除以 24，则 16 位定时器的时钟频率为 500 kHz。在此时钟速率下，可以创建的最慢中断大约为 130 ms (65535 * 2μs)。您可以使用 **Program13_2** 之类的程序来测试此驱动程序。注意使用位带来移除通常在共享全局上使用读 - 修改 - 写序列发生的临界区。

```
// Test of Periodic interrupt
#define REDLED (*(volatile uint8_t *) (0x42098060))
#define BLUELED (*(volatile uint8_t *) (0x42098068))
uint32_t Time;
void Task(void){
    REDLED ^= 0x01;        // toggle P2.0
    REDLED ^= 0x01;        // toggle P2.0
    Time = Time + 1;
    REDLED ^= 0x01;        // toggle P2.0
}
int Program13_2(void){
    Clock_Init48MHz();
    LaunchPad_Init(); // built-in switches and LEDs
    TimerA1_Init(&Task,50000); // 10 Hz
    EnableInterrupts();
    while(1){
        BLUELED ^= 0x01; // toggle P2.1
    }
}
```

使用双跟踪范围观察 P2.0（中断线程）和 P2.1（主线程）。触发中断信号并使用 P2.1 振荡中的间隙来估计服务定时器 A1 中断所需的时间。

在单独测试 PWM 和 *Timer A1* 之后，将它们组合成一个运行机器人的软件系统，如程序 13.1，但使用周期性中断来检查碰撞开关，在碰撞时停止机器人。

13.5 疑难解答

PWM 或中断是不正确的周期：

- 检查定时器时钟源。
- 确保处理器以 48MHz 运行。

PWM 输出没有发生：

- 运行 **PWMSine** 项目以查看硬件是否正常。
- 使用调试器确保 *Timer_A0* 寄存器已设置。

中断没有发生：

- 运行 **PeriodicTimerA0Ints** 项目并使用调试器观察 *Timer A0* 寄存器。运行程序并观察 *Timer A1* 寄存器。
- 使用调试器观察 NVIC 中的寄存器。
- 通过调用 **EnableInterrupts ()** 确保 I 位清零；



实验：定时器

13.6 请思考

在本节中，我们列出了完成本实验后要考虑的问题。这些问题旨在测试您对本实验中概念的理解。本模块的目标是让您了解 **Timer_A** 及其用于 **PWM** 和周期性中断的用途。

- 软件如何选择 **Timer_A** 的输入时钟？
- 预分频器为 **Timer_A** 做了什么？为什么预分频器很重要（即更改预分频器时会发生什么？）
- 本实验中生成的 **PWM** 的精度是多少？
- 如果在 **Program13_2** 主程序中循环执行 **P2->OUT ^= 0x04** 会发生什么情况；相反呢？
- 您如何使用 **Timer A1** 每秒执行一次定期任务？
- 什么是函数指针？为什么在本实验中使用函数指针？

13.7 其它挑战

在本节中，我们列出了您可以执行的其它活动，以进一步探索此模块的概念。例如，

- 如果您没有电机板，则必须更改软件的运行方式。幸运的是，可以在 **P2.4**，**P2.5**，**P2.6** 或 **P2.7** 中的任何一个上创建 **PWM** 输出。
- 现在可以将实验 7（**FSM**），实验 12（电机）和本实验结合起来，创建一个循线机器人。

13.8 接下来是哪些模块？

- 1) 本实验中构思的机器人有两个主要限制。
- 2) 1) 软件消耗所有处理器时间。2) 电机的速度取决于许多因素，其中大部分因素无法提前预测。在余下的实验中，我们将解决这些限制。

模块 15) 使用 **ADC** 连接距离传感器。两个距离传感器可用于以固定距离和与墙壁固定的角度驱动机器人。

模块 16) 连接转速计并使用定时器捕获直接测量每个车轮的速度。

模块 17) 组合模块 12，13 和 16 创建一个控制系统，该控制系统以期望的速度旋转电机。

13.9 您应该已经学会

在本节中，我们将回顾您应该在本单元中学到的重要概念：

- 了解电机的电压，电流和功率。
- 能够使用 **PWM** 输出来调节电机的功率。
- 了解 **H** 桥的基本操作和目的。
- 知道如何编写和测试底层软件驱动程序。

IMPORTANT NOTICE FOR TI DESIGN INFORMATION AND RESOURCES

Texas Instruments Incorporated ("TI") technical, application or other design advice, services or information, including, but not limited to, reference designs and materials relating to evaluation modules, (collectively, "TI Resources") are intended to assist designers who are developing applications that incorporate TI products; by downloading, accessing or using any particular TI Resource in any way, you (individually or, if you are acting on behalf of a company, your company) agree to use it solely for this purpose and subject to the terms of this Notice.

TI's provision of TI Resources does not expand or otherwise alter TI's applicable published warranties or warranty disclaimers for TI products, and no additional obligations or liabilities arise from TI providing such TI Resources. TI reserves the right to make corrections, enhancements, improvements and other changes to its TI Resources.

You understand and agree that you remain responsible for using your independent analysis, evaluation and judgment in designing your applications and that you have full and exclusive responsibility to assure the safety of your applications and compliance of your applications (and of all TI products used in or for your applications) with all applicable regulations, laws and other applicable requirements. You represent that, with respect to your applications, you have all the necessary expertise to create and implement safeguards that (1) anticipate dangerous consequences of failures, (2) monitor failures and their consequences, and (3) lessen the likelihood of failures that might cause harm and take appropriate actions. You agree that prior to using or distributing any applications that include TI products, you will thoroughly test such applications and the functionality of such TI products as used in such applications. TI has not conducted any testing other than that specifically described in the published documentation for a particular TI Resource.

You are authorized to use, copy and modify any individual TI Resource only in connection with the development of applications that include the TI product(s) identified in such TI Resource. NO OTHER LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE TO ANY OTHER TI INTELLECTUAL PROPERTY RIGHT, AND NO LICENSE TO ANY TECHNOLOGY OR INTELLECTUAL PROPERTY RIGHT OF TI OR ANY THIRD PARTY IS GRANTED HEREIN, including but not limited to any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information regarding or referencing third-party products or services does not constitute a license to use such products or services, or a warranty or endorsement thereof. Use of TI Resources may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

TI RESOURCES ARE PROVIDED "AS IS" AND WITH ALL FAULTS. TI DISCLAIMS ALL OTHER WARRANTIES OR REPRESENTATIONS, EXPRESS OR IMPLIED, REGARDING TI RESOURCES OR USE THEREOF, INCLUDING BUT NOT LIMITED TO ACCURACY OR COMPLETENESS, TITLE, ANY EPIDEMIC FAILURE WARRANTY AND ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT OF ANY THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

TI SHALL NOT BE LIABLE FOR AND SHALL NOT DEFEND OR INDEMNIFY YOU AGAINST ANY CLAIM, INCLUDING BUT NOT LIMITED TO ANY INFRINGEMENT CLAIM THAT RELATES TO OR IS BASED ON ANY COMBINATION OF PRODUCTS EVEN IF DESCRIBED IN TI RESOURCES OR OTHERWISE. IN NO EVENT SHALL TI BE LIABLE FOR ANY ACTUAL, DIRECT, SPECIAL, COLLATERAL, INDIRECT, PUNITIVE, INCIDENTAL, CONSEQUENTIAL OR EXEMPLARY DAMAGES IN CONNECTION WITH OR ARISING OUT OF TI RESOURCES OR USE THEREOF, AND REGARDLESS OF WHETHER TI HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

You agree to fully indemnify TI and its representatives against any damages, costs, losses, and/or liabilities arising out of your non-compliance with the terms and provisions of this Notice.

This Notice applies to TI Resources. Additional terms apply to the use and purchase of certain types of materials, TI products and services. These include; without limitation, TI's standard terms for semiconductor products (<http://www.ti.com/sc/docs/stdterms.htm>), [evaluation modules](#), and [samples](http://www.ti.com/sc/docs/sampterm.htm) (<http://www.ti.com/sc/docs/sampterm.htm>).

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2018, Texas Instruments Incorporated