Micromega Corporation

uM-PWM1

Pulse-width Modulation Servo Coprocessor

Datasheet

Release V100

Introduction

The uM-PWM1 chip is designed to work with pulse-width modulated signals used for remote control servo applications. It provides up to 8 input channels for monitoring pulse widths and 8 output channels for producing pulse width modulated (PWM) signals at either 50 Hz or a selectable frequency from 10 to 450 Hz. It can be interfaced with a microcontroller using an I²C or serial data interface.

Features

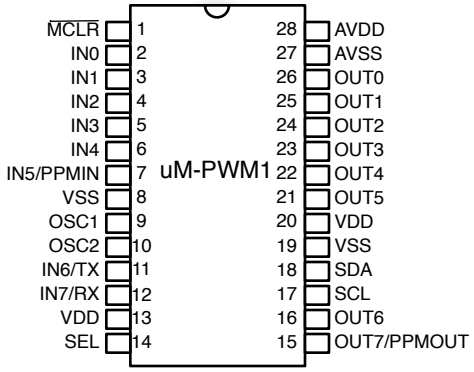
- internal oscillator
- I²C bus interface (up to 400 Kbps)
- serial data interface
- up to 8 input channels
  - pulse widths measured with ± 1 microsecond accuracy
  - pulse count
  - high time
  - low time
  - frequency
- provides decoding of Pulse Position Modulation (PPM) input signal
- 8 output channels
  - pulse widths generated with ± 1 microsecond accuracy
  - selectable output channel frequency
    - standard frequency (50 Hz)
    - alternate frequency (from 10 to 450 Hz)
  - pulse widths restricted to specified range
    - 800 to 2200 microseconds standard pulse widths
    - 500 to 2500 extended pulse widths
  - output channel movements can be coordinated and synchronized
  - speed and time of output channel movements can be controlled
- provides encoding of Pulse Position Modulation (PPM) output signal
- 5V operating voltage
- RoHS compliant

Applications

- remote control servos
- controlling radio-control models
- UAV
- robotics
- automation
- special effects

Pin Descriptions

Pinout Diagram
DIP-28, SOIC-28



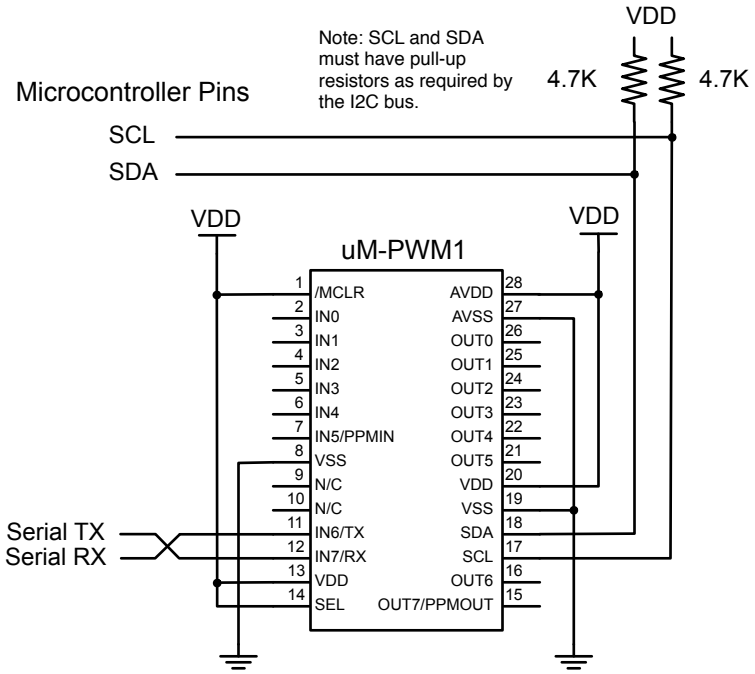
Pin	Name	Type	Description
1	/MCLR	Input	Master Clear
2	IN0	Input	Input Channel 0
3	IN1	Input	Input Channel 1
4	IN2	Input	Input Channel 2
5	IN3	Input	Input Channel 3
6	IN4	Input	Input Channel 4
7	IN5 PPMIN	Input	Input Channel 5 PPM Input
8	VSS	Power	Ground
9	OSC1	Input	N/C
10	OSC2	Output	N/C
11	IN6 TX	Input	Input Channel 6 Serial Output
12	IN7 RX	Input	Input Channel 7 Serial Input
13	VDD	Power	+5V
14	SEL	Input	+5V = RX/TX enabled, GND = IN6/IN7 enabled
15	OUT7 PPMOUT	Output	Output Channel 7 PPM Output
16	OUT6	Output	Output Channel 6
17	SCL	Input	I^2C Clock
18	SDA	In/Out	I^2C Data
19	VSS	Power	Ground
20	VDD	Power	+5V
21	OUT5	Output	Output Channel 5
22	OUT4	Output	Output Channel 4
23	OUT3	Output	Output Channel 3
24	OUT2	Output	Output Channel 2
25	OUT1	Output	Output Channel 1
26	OUT0	Output	Output Channel 0
27	AVSS	Power	Ground
28	AVDD	Power	+5V

Connection Diagrams

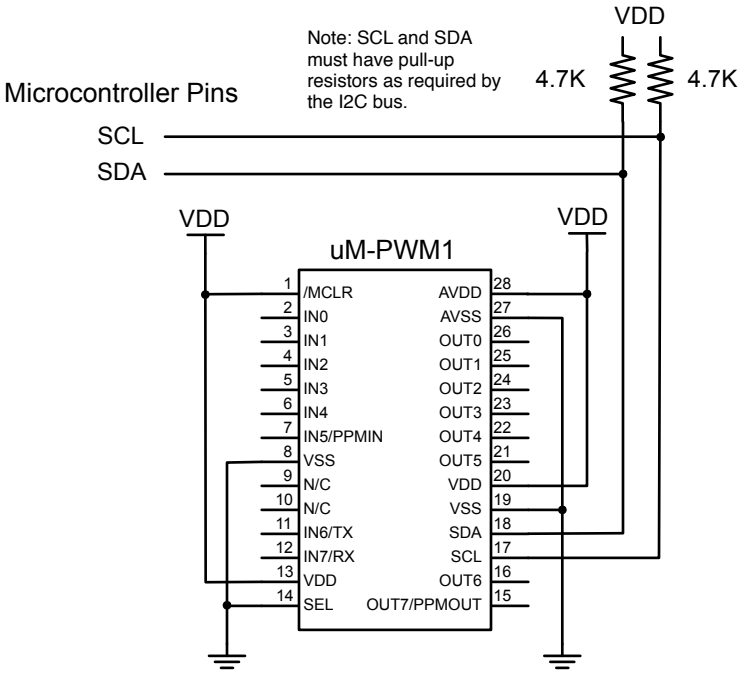
The following diagram shows the power and interface connections for the uM-PWM. Input signals are connected to pins IN0 through IN7, and output signals are connected to pins OUT0 through OUT7. The SEL pin on the uM-PWM chip determines the function of the IN6/TX and IN7/RX pins. It should be tied to either +5V or GND. The resulting configuration is summarized below:

SEL	Description
+5V	TX/RX enabled
GND	IN6/IN7 enabled

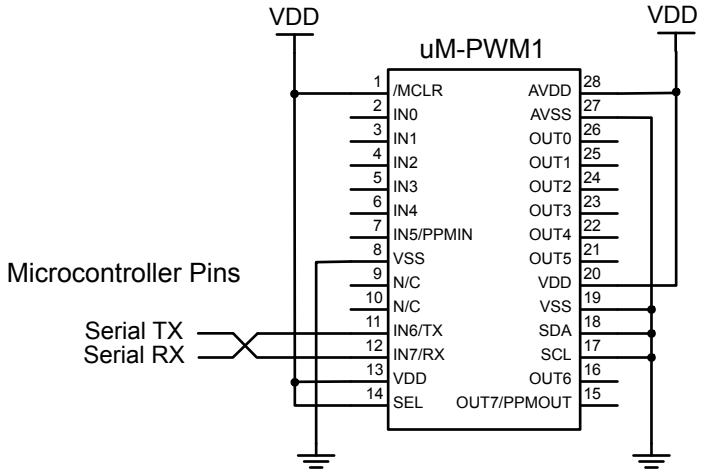
Both I²C and Serial interfaces enabled



I²C interface enabled, Serial interface disabled



Serial interface enabled, I²C interface disabled



I²C Interface

This section describes the process of reading and writing information to the uM-PWM1 using the I²C interface.

I²C Slave Address

The default slave address for the uM-PWM1 is as follows:

7-bit address:	0x65
8-bit write address	0xCA
8-bit read address	0xCB

The I²C device address can be changed using register 04. Once the device address is changed, the uM-PWM1 will only respond to the new address. The I²C device address is stored in Flash memory, and is retained after power is removed.

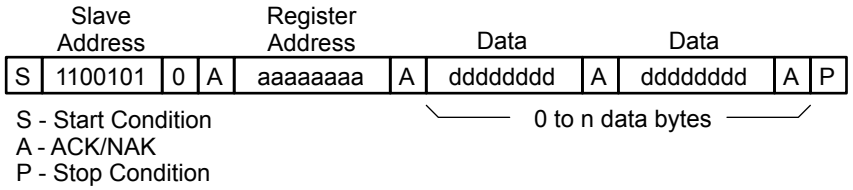
I²C Bus Speed

The uM-PWM1 can handle I²C data speeds up to 400 Kbps.

I²C Reading and Writing Data

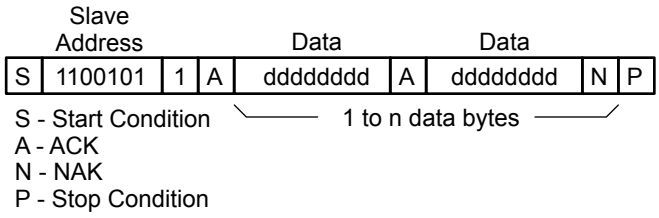
The following diagrams show the write and read data transfers.

I²C Write Data Transfer



A write transfer sends data from the microcontroller to the uM-PWM1 chip, and consists of a slave address, a register address, and 0 to n data bytes. The register address selects the uM-PWM1 register to read or write. All registers are 16-bit registers, and data is sent with the most-significant data byte first. If the register address is greater than 00, it will automatically increment after every two data bytes are written. This allows consecutive registers to be written with a single write transfer. If the register address is 00, the address will not increment. This allows multiple move commands to be sent to register 00 with a single write transfer.

I²C Read Data Transfer



A read transfer sends data from the uM-PWM1 to the microcontroller, and consists of a slave address, followed by 1 to n data bytes. Read transfers are normally preceded by a write transfer to select the register to read from. Data is read with the most-significant byte first. If the register address is greater than 00, it will automatically increment after every two data bytes are read. This allows consecutive registers to be read with a single read transfer. If the register address is 00, the address will not increment. This allows multiple move status reads to occur when waiting for the move status to indicate that a move has completed.

Registers

The uM-PWM1 is controlled and configured using a number of on-chip registers which are described in this section. The table below lists all register addresses, and provides a description, the access type, and the units for each register. All register values are 16 bits. A detailed description of each register is provided following the table.

Address (hexadecimal)	Description	Access	Units
00	Move Command / Move Status	Read/Write	
01	Device ID1 = 0x5057 ('PW')	Read	
02	Device ID2 = 0x4D31 ('M1')	Read	
03	Device ID3 = version code	Read	
04	I2C device address	Read/Write	
05	Mode	Read/Write	bit mask
06	Input Sample Edge	Read/Write	bit mask
07	Input Change	Read/Write	bit mask
08	Input Port	Read	bit mask
09	Output Port	Read/Write	bit mask
0A	Alternate Frequency	Read/Write	Hz
0B	Countdown Timer	Read/Write	msec
0C	Elapsed Time (low 16 bits)	Read/Write[1,2]	msec
0D	Elapsed Time (high 16 bits)	Read/Write[1,2]	msec
0E	Stored Configurations	Read/Write	
0F	Clock Tuning	Read/Write	
10 - 2F	<reserved>[3]		
30	Input PPM - Channel 0	Read	usec
31	Input PPM - Channel 1	Read	usec
32	Input PPM - Channel 2	Read	usec
33	Input PPM - Channel 3	Read	usec
34	Input PPM - Channel 4	Read	usec
35	Input PPM - Channel 5	Read	usec
36	Input PPM - Channel 6	Read	usec
37	Input PPM - Channel 7	Read	usec
38	Input PPM - Channel 8	Read	usec
39	Input PPM - Channel 9	Read	usec
3A - 4F	<reserved>[3]		
40	Input Frequency - Channel 0	Read	Hz
41	Input Frequency - Channel 1	Read	Hz
42	Input Frequency - Channel 2	Read	Hz
43	Input Frequency - Channel 3	Read	Hz
44	Input Frequency - Channel 4	Read	Hz
45	Input Frequency - Channel 5	Read	Hz
46	Input Frequency - Channel 6	Read	Hz
47	Input Frequency - Channel 7	Read	Hz
48 - 4F	<reserved>[3]		
50	Input High - Channel 0	Read	usec
51	Input High - Channel 1	Read	usec
52	Input High - Channel 2	Read	usec
53	Input High - Channel 3	Read	usec

54	Input High - Channel 4	Read	usec
55	Input High - Channel 5	Read	usec
56	Input High - Channel 6	Read	usec
57	Input High - Channel 7	Read	usec
58-5F	<reserved>[3]		
60	Input Low - Channel 0	Read	usec
61	Input Low - Channel 1	Read	usec
62	Input Low - Channel 2	Read	usec
63	Input Low - Channel 3	Read	usec
64	Input Low - Channel 4	Read	usec
65	Input Low - Channel 5	Read	usec
66	Input Low - Channel 6	Read	usec
67	Input Low - Channel 7	Read	usec
68-6F	<reserved>[3]		
70	Input Count - Channel 0	Read/Write[2]	
71	Input Count - Channel 1	Read/Write[2]	
72	Input Count - Channel 2	Read/Write[2]	
73	Input Count - Channel 3	Read/Write[2]	
74	Input Count - Channel 4	Read/Write[2]	
75	Input Count - Channel 5	Read/Write[2]	
76	Input Count - Channel 6	Read/Write[2]	
77	Input Count - Channel 7	Read/Write[2]	
78-7F	<reserved>[3]		
80	Output Frequency - Channel 0	Read/Write	Hz
81	Output Frequency - Channel 1	Read/Write	Hz
82	Output Frequency - Channel 2	Read/Write	Hz
83	Output Frequency - Channel 3	Read/Write	Hz
84	Output Frequency - Channel 4	Read/Write	Hz
85	Output Frequency - Channel 5	Read/Write	Hz
86	Output Frequency - Channel 6	Read/Write	Hz
87	Output Frequency - Channel 7	Read/Write	Hz
88-8F	<reserved>[3]		
90	Output Current Position - Channel 0	Read	usec
91	Output Current Position - Channel 1	Read	usec
92	Output Current Position - Channel 2	Read	usec
93	Output Current Position - Channel 3	Read	usec
94	Output Current Position - Channel 4	Read	usec
95	Output Current Position - Channel 5	Read	usec
96	Output Current Position - Channel 6	Read	usec
97	Output Current Position - Channel 7	Read	usec
98-9F	<reserved>[3]		
A0	Output Next Position - Channel 0	Read/Write	usec
A1	Output Next Position - Channel 1	Read/Write	usec
A2	Output Next Position - Channel 2	Read/Write	usec
A3	Output Next Position - Channel 3	Read/Write	usec
A4	Output Next Position - Channel 4	Read/Write	usec
A5	Output Next Position - Channel 5	Read/Write	usec
A6	Output Next Position - Channel 6	Read/Write	usec
A7	Output Next Position - Channel 7	Read/Write	usec

A8-AF	<reserved>[3]		
B0	Output Maximum Speed - Channel 0	Read/Write	usec/sec
B1	Output Maximum Speed - Channel 1	Read/Write	usec/sec
B2	Output Maximum Speed - Channel 2	Read/Write	usec/sec
B3	Output Maximum Speed - Channel 3	Read/Write	usec/sec
B4	Output Maximum Speed - Channel 4	Read/Write	usec/sec
B5	Output Maximum Speed - Channel 5	Read/Write	usec/sec
B6	Output Maximum Speed - Channel 6	Read/Write	usec/sec
B7	Output Maximum Speed - Channel 7	Read/Write	usec/sec
B8-FF	<reserved>[3]		

Notes:

1 The value of register 0D (the high 16 bits of the 32-bit elapsed time) is latched when register 0C is read. Register 0C must always be read before reading register 0D.

2 Writing any 16-bit value to this register clears the register.

3 Reading from a reserved register returns zero. Writing to a reserved register is ignored.

Move Command/Move Status

The move command/move status register is a special register that behaves differently than the other registers. This register is used to send commands to the uM-PWM1 to control the movement of servos (by changing the output channel settings), and to read the move status to determine when a move has been completed.

Register 00 (write)

Byte 1	Byte 2	Byte 3
Command	MSB of argument	LSB of argument

Register 00 (read)

Byte
Status

Move Command

Writing commands to the move command register provides a means of controlling the movement of the output channels. Each command sent to this register is three bytes in length. The first byte is the command byte, the second byte is the most significant byte of the 16-bit argument, and the third byte is the least significant byte of the 16-bit argument. Output channel 0 is selected by default at the start of each Move Command. To select another channel for the `P`, `Q`, `S`, and `F` commands, the `C` command must be sent first, in the same I²C write transaction.

e.g

The following I²C write transaction sets the pulse width of output channel 2 to 1000.

```
<Start condition>
CA          Start Address
00          Move Command/Status Register
43          'C' Select output channel command
00          channel = 2
02
50          'P' Set 50Hz pulse width command
03          pulse width  =  1000
E8
<End Condition>
```

The following table lists the commands and arguments.

Command	Description	Argument	Units
0x43 ('C')	Select output channel	0 to 7	channel number
0x50 ('P')	Set 50 Hz pulse width	0 to 65535 [1]	usec
0x51 ('Q')	Set alternate Hz pulse width	0 to 65535 [1]	usec
0x53 ('S')	Set speed of movement	0 to 65535	usec/sec
0x54 ('T')	Set time of movement	0 to 65535	msec
0x46 ('F')	Set alternate frequency	10 to 450	Hz
0x4D ('M')	Move	0	
0x48 ('H')	Set all output channels to home setting	0	
0x5A ('Z')	Disable all channels	0	
0x4C ('L')	Log input channel highs to TX	500 to 65535	msec

Notes:

1 The actual pulse width is constrained by the minimum and maximum allowable pulse widths (800 to 2200 usec in standard mode, or 500 to 2500 usec in extended mode).

The commands are used to specify frequency, pulse width, speed and time of movement for the output channels. The register address is not automatically incremented by reading or writing to this register, so multiple commands can be sent to this register. Movement only occurs after the M or H commands are sent.

C Select output channel

The C command is used to select the output channel for a move. If no channel is selected, output channel 0 is assumed.

P Set 50 Hz pulse width

The P command sets the next pulse width for the currently selected channel, and sets the frequency to 50 (Hz). The value is restricted to 800 to 2200 microseconds in standard mode, or 500 to 2500 microseconds in extended mode. A value of zero will disabled the channel. If no S or T command is specified, the output channel will move immediately to the next position.

Q Set alternate pulse width

The Q command sets the next pulse width for the currently selected channel, and sets the frequency to the alternate frequency. The value is restricted to 800 to 2200 microseconds in standard mode, or 500 to 2500 microseconds in extended mode. A value of zero will disabled the channel. If no S or T command is specified, the output channel will move immediately to the next position.

S Set speed of movement

The S command sets the speed of movement for the currently selected channel. The units of speed are usec/sec, which provides for very precise control of the speed. For example, if a servo is moving from its current position of 1000 usec to a next position of 2000 usec, it will move a 'distance' of 1000 usec.

- if the speed is set to 10000 usec/second, the move would take 0.1 seconds.
- if the speed is set to 1000 usec/second, the move would take one second.
- if the speed is set to 100 usec/second, the move would take 10 seconds

If a maximum speed has been set for the output channel (Registers B0 to B7), then the maximum speed will be limited to this value. The T command provides an alternate way of controlling the speed of movement. If the S and T commands are both specified for an output channel, the one which specifies the slowest movement for that channel will take precedence.

T Set time of movement

The T command sets the time of movement (in milliseconds) for all output channels specified in the current move. The movement will be coordinated so that all output channels arrive at the next position simultaneously. The S command provides an alternate way of controlling the speed of movement. If the S and T commands are both specified for an output channel, the one which specifies the slowest movement for that channel will take precedence. If a maximum speed has been set for any of the output channels (Registers B0 to B7), the time of movement may be longer if the maximum speed for any channel requires a longer time of movement.

F Set frequency

The F command selects the frequency for the selected channel. The frequency can be 10 to 450 (Hz), or 0 to disable the channel. If the frequency is equal to the standard frequency of 50 (Hz), only the selected channel is affected. If the frequency is not 50 (Hz), the alternate frequency is set to the new value, and all output channels currently using an alternate frequency will be changed to the new alternate frequency. There is only one alternate frequency for all output channels.

M Move

The M command initiates the movement for all channels selected in previous commands. No movement occurs until the M or H command is sent.

H Home

The H command sets all output channels to the home (startup) position for each output channels. The move occurs immediately. Any preceding S commands are ignored, but if a maximum output speed (Registers B0 to B7) has been set for any output channel, the speed of movement will be limited to this value. The T command can precede this command to control the time of movement.

Z Disable all output channels

The Z command disables all output channels. None of the output channels will have a PWM or PPM signal after sending this command. This command is immediate.

L Log input channel Highs to TX

If the serial interface RX/TX pins have been enabled, the L command will log the duration of the last high pulse for each input channel to the serial interface TX pin. The values are output as unsigned 16-bit values (0 to 65535), separated by commas, and terminated with a carriage return (CR) and linefeed (LF) character. The argument specifies the log interval in milliseconds. If the interval is zero, no logging occurs. The log is terminated if the interval is zero, or any character is received on the serial interface RX pin.

Move Status

Reading the move status register returns a single byte value as follows:
0 - all channel moves have been completed
1 - channels moves are still in progress

The move status is particularly useful for determining when a speed or time limited move is complete. A status of zero means that the uM-PWM1 is currently sending pulses for the end position of all movements, but it doesn't necessarily mean that the servo has actually stopped moving. When a servo receives a new signal, there's a delay before it actually reaches the new position. This delay varies depending on the type of servo and the distance travelled.

To use the status return to determine when actual movement is complete, a few approaches can be used. If the speed parameter is used, and matched with (or slower than) the actual speed of the servo, then servo movement will be complete when the status return is zero. If the time parameter is used, and is longer than the maximum

time required to move a servo, then servo movement will be complete when the status return is zero. Alternatively, after the status return is zero, a fixed delay can be added to allow movement to complete. The countdown register can be used for this purpose.

Device ID

These registers are used to read device identification information.

Register 01 (read-only**)**

MSB	LSB
0x50 ('P')	0x4D ('M')

Register 02 (read-only**)**

MSB	LSB
0x57 ('W')	0x31 ('1')

Register 03 (read-only**)**

MSB	LSB
Version Code	

I²C device address

This register is used to read or write the I²C device address.

Register 04 (Read/Write)

MSB	LSB
Device Address	Device Address

The device address is an 8-bit value that must be written to both the MSB and LSB of the register. The I²C device address will only be changed if both values are identical. Once the device address has been changed, the uM-PWM1 will only respond to the new I²C address. The I²C device address is stored in Flash memory, and is retained after power is removed.

Mode

This register is used to read or write the mode settings for the uM-PWM1.

Register 05 (read/write)

MSB

Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
-	-	-	-	-	-	OPPM	EP

LSB

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
RX	TX	IPPM	-	-	-	-	-

OPPM	Enable output PPM on OUT7/PPMOUT pin
EP	Enable extended pulse widths
RX	Enable serial data receive on IN7/RX pin (read-only)
TX	Enable serial data transmit on IN6/TX pin (read-only)
IPPM	Enable input PPM on IN5/PPMIN pin

The mode register is used to enable the input and output PPM signals and to enable extended pulse widths. The RX and TX bits are read-only, their value is determined by the setting of the SEL pin.

When the OPPM bit is set, a PPM signal is output on pin OUT7. The PPM signal is encoded sequentially from output channels 0 through 6, using only those channel with a frequency of 50 (Hz).

When the EP bit is set, the allowable range for pulse widths is changed from the normal range of 800 to 2200 microseconds, to the extended range of 500 to 2500 microseconds.

When the IPPM bit is set, a PPM signal can be input on pin IN5. The signal is decoded and the input channel pulse widths are stored in register 30 to 39 (hex).

Input Sample Edge

This register is used to read or write the input edge trigger for each input channel.

Register 06 (read/write)

MSB

Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
-	-	-	-	-	-	-	-

LSB

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IN7	IN6	IN5	IN4	IN3	IN2	IN1	IN0

IN7	Sample edge for IN7/RX pin
IN6	Sample edge for IN6/TX pin
IN5	Sample edge for IN5/PPMIN pin
IN4	Sample edge for IN4 pin
IN3	Sample edge for IN3 pin
IN2	Sample edge for IN2 pin
IN1	Sample edge for IN1 pin
IN0	Sample edge for IN0 pin

The bits corresponding to each input channel are set as follows:

0 - input channel sampled on falling edge

1 - input channel sampled on rising edge

The default setting for all channels is to sample on the rising edge. When the input channel is sampled, the pulse count is incremented and the frequency is calculated. The frequency is calculated using the sum of the previous Low and High pulse durations.

Input Change

This register is used to read or clear the input change bits for the uM-PWM1.

Register 07 (read/write)

MSB

Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
OVR	-	-	-	-	-	CLRP	CLRA

LSB

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IN7	IN6	IN5	IN4	IN3	IN2	IN1	IN0

OVR	Input overrun
CLRP	Clear all input PPM input registers
CLRA	Clear input frequency/high/low/count registers
IN7	Input change on IN7/RX pin
IN6	Input change on IN6/TX pin
IN5	Input change on IN5/PPMIN pin
IN4	Input change on IN4 pin
IN3	Input change on IN3 pin
IN2	Input change on IN2 pin
IN1	Input change on IN1 pin
IN0	Input change on IN0 pin

The IN0 through IN7 input change bits are set whenever the corresponding input pins change value. The IN0 through IN7 input change bits can be cleared by writing a one to the corresponding bit. Writing a one to the CLRA bit will clear the input frequency, last input high, last input low, and input count registers for all input channels. Writing a one to the CLRP bit will clear all PPM input registers. The OVR bit will be set if an input overrun condition occurs. This can occur if the input signals exceed the maximum frequency specifications. Writing a one to the OVR bit will clear the OVR bit.

Input Port

This register is used to read the digital input value of all input channels.

Register 08 (read-only**)**

MSB

Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
-	-	-	-	-	-	-	-

LSB

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IN7	IN6	IN5	IN4	IN3	IN2	IN1	IN0

IN7	Digital value of IN7/RX pin
IN6	Digital value of IN6/TX pin
IN5	Digital value of IN5/PPMIN pin
IN4	Digital value of IN4 pin
IN3	Digital value of IN3 pin
IN2	Digital value of IN2 pin
IN1	Digital value of IN1 pin

IN0	Digital value of IN0 pin

Output Port

This register is used to read or write the digital output value of all output channels.

Register 09 (read/write)

MSB

Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
-	-	-	-	-	-	-	-

LSB

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OUT7	OUT6	OUT5	OUT4	OUT3	OUT2	OUT1	OUT0

OUT7	Digital value of OUT7/PPMOUT pin
OUT6	Digital value of OUT6 pin
OUT5	Digital value of OUT5 pin
OUT4	Digital value of OUT4 pin
OUT3	Digital value of OUT3 pin
OUT2	Digital value of OUT2 pin
OUT1	Digital value of OUT1 pin
OUT0	Digital value of OUT0 pin

Writing to this register can be used to set the digital output value for any output channel not being used to generate a PWM or PPM signal. If an output channel is currently generating a PWM or PPM signal, the output bit setting will not be changed by writing to this register.

Alternate Frequency

This register is used to read or write the alternate frequency.

Register 0A (read/write)

MSB	LSB
Frequency (Hz)	

Each output channel can be configured to use either the standard 50 Hz frequency or an alternate frequency. The alternate frequency is specified by writing to this register and must be in the range or 10 to 450 (Hz). There is only one alternate frequency for all output channels. All output channels currently using an alternate frequency, will be changed to the new frequency. Any output channels using the standard 50 Hz frequency will be unaffected by writing to this register.

Countdown Timer

This register is used to read or write the countdown timer.

Register 0B (read/write)

MSB	LSB
Countdown timer (msec)	

The countdown timer is a 16-bit integer value that counts down each millisecond until the counter is zero. If the

counter is zero, no further countdown occurs. To implement a time delay, the register is set with the delay period, then polled until the value reaches zero.

Elapsed Time

These registers are used to read or reset the elapsed time counter.

Register 0C (read/write)

MSB	LSB
Elapsed time - low 16 bits (msec)	

Register 0D (read/write)

MSB	LSB
Elapsed time - high 16 bits (msec)	

The elapsed time is a 32-bit integer value representing the number of milliseconds that have elapsed since the elapsed time counter was last cleared. The counter is cleared at power-up and when any value is written to either of the elapsed time registers. Since a clock tick could occur between the time the first register is read and the second register is read, the uM-PWM1 latches the high 16-bits of the elapsed time whenever the low 16-bits are read. The latched value is returned when the high 16-bits are read. The correct procedure to read the 32-bit elapsed time is to always read register 0C before reading register 0D. If an application only requires a 16-bit elapsed time, then only the low 16 bits (register 0C) need to be read.

The 32-bit elapsed time provides a maximum interval of:
4,294,967,295 milliseconds, or
49 days, 17 hours, 2 minutes, 47.295 seconds

The 16-bit elapsed time provides a maximum interval of:
65535 milliseconds, or
1 minute, 5.535 seconds

Stored Configurations

This register is used to store configurations in Flash memory.

Register 0E (write)

MSB

Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
-	-	-	-	-	-	C1	C0

LSB

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit2 Bit1 Bit 0
-	-	-	-	-	Flash Number

C1, C0 — 00 - load configuration
01 - save configuration
10 - set initial configuration
11 - save and set initial configuration

Flash Number — Flash storage area (0 to 7)

Register 0E (read)

MSB	LSB
Initial Configuration Flash Number	

There are eight Flash memory areas that can be used for saved configurations. Flash memory area 0 is the default configuration, and is read-only. Flash storage areas 1 to 7 can be used to save user-defined configurations. The following register values are stored in each configuration:

Mode (the read-only bits are not stored)
Input Sample Edge
Alternate Frequency
Output Frequency (for all channels)
Output Current Position (for all channels)
Output Maximum Speed (for all channels)

The default configuration is as follows:

Mode	0x00		
Input Sample Edge	0xFF		Sample all channels on rising edge.
Alternate Frequency	270	(Hz)	
Output Frequency	50	(Hz)	All channels.
Output Current Position	1500	(usec)	All channels.
Output Maximum Speed	0	(usec/sec)	No maximum speed.

The initial configuration is loaded at startup and by the H (home) command. Any of the eight configurations can be set as the initial configuration. If the configuration selected to be loaded has not been previously saved, the default configuration is loaded.

Bits C1 and C0 specify the action taken when a value is written to this register. The Flash memory area selected as the initial configuration is returned when reading from this register.

Clock Tuning

This register can be used to fine tune the clock frequency.

Register 0F (read/write)

MSB	LSB
Clock Tuning Value	

The internal oscillator can vary up to 1% from its specified frequency. As a result, the pulse widths and pulse frequencies can vary accordingly. In many applications, a 1% difference is not significant, but if more accuracy is required, the clock tuning value can be used to fine tune the frequency. The value can be from -1028 to +1027.

To select a clock tuning value:

- specify a known pulse width for one of the output channels
- measure the pulse width using an oscilloscope or logic analyzer
- calculate the clock tuning value as follows:

$$clockTune = \left(\frac{Expected}{Actual} - 1 \right) * 25600$$

Input PPM

These registers are used to read the pulse widths for the PPM input channels. PPM input is available as an alternate function for pin IN5/PPMIN pin. The option is enabled by setting the appropriate bit in the Mode register.

Registers 30 to 39 (read-only**)**

MSB	LSB
Input PPM pulse width (msec)	

The value returned is the time in microseconds for each channel in the PPM signal. A zero value indicates that the input channel has not received a signal. A value of 65535 indicates that the pulse length is greater than or equal to 65535 microseconds.

Input Frequency

These registers are used to read the input frequency of the eight input channels.

Registers 40 to 47 (read only)

MSB	LSB
Input channel frequency (Hz)	

The value returned is the input frequency in Hz. A zero value indicates that the input channel has no signal, or the frequency is less than 1 Hz. A value of 65535 indicates that the frequency is too high for detection by the uM-PWM1.

Input High

These registers are used to read the last high pulse width for the eight input channels.

Registers 50 to 57 (read-only**)**

MSB	LSB
Input channel High pulse width (msec)	

The value returned is the time in microseconds for the last high pulse. A zero value indicates that the input channel has not received a signal. A value of 65535 indicates that the pulse length is greater than or equal to 65535 microseconds.

Input Low

These registers are used to read the last low pulse duration of the eight input channels.

Registers 60 to 67 (read-only**)**

MSB	LSB
Input channel Low pulse width (msec)	

The value returned is the time in microseconds for the last low pulse. A zero value indicates that the input channel has not received a signal. A value of 65535 indicates that the pulse length is greater than or equal to 65535 microseconds.

Input Count

These registers are used to read the number of pulses received by each of the eight input channels.

Registers 70 to 77 (read/write)

MSB	LSB
Input channel pulse count	

The register is a 16-bit register. Once 65535 pulses have been received, the count will roll over to zero.

Output Frequency

These registers are used to read and write the output frequency of the eight output channels.

Registers 80 to 87 (read/write)

MSB	LSB
Output channel frequency (Hz)	

The frequency can be 10 to 450 (Hz), or 0 to disable the channel. If the frequency is equal to the standard frequency of 50 (Hz), only the selected channel is affected. If the frequency is not 50 (Hz), the alternate frequency is set to the new value, and all output channels currently using an alternate frequency will be changed to the new alternate frequency. There is only one alternate frequency for all output channels.

Output Current Position

These registers are used to read the current position of the eight output channels.

Registers 90 to 97 (read-only)

MSB	LSB
Current output channel pulse width (usec)	

A movement that is specified with a speed or time parameter, may take a period of time to complete. During a move, the current position will change incrementally until the output channel reaches the next position. To determine if all moves have completed, the Move Status (register 00) can be used. To determine if a particular channel has completed is movement, the current and next positions can be compared. If they are the same, the output channel has completed its movement.

Output Next Position

These registers are used to read and write the next position of the eight output channels.

Registers A0 to A7 (read/write)

MSB	LSB
Next output channel pulse width (usec)	

Writing to this register will cause the output channel to move immediately to the next position specified. The value is restricted to 800 to 2200 microseconds in standard mode, or 500 to 2500 microseconds in extended mode. A value of zero will disabled the channel.

Output Maximum Speed

These registers are used to read and write the maximum speed the eight output channels.

Registers B0 to B7 (read/write)

MSB	LSB
maximum speed (usec/sec)	

Writing to this register sets the maximum speed of movement for the currently selected channel. The units of maximum speed are usec/sec, which provides for very precise control of the speed. For example, if a servo moves from a position of 1000 usec to a position of 2000 usec, it will move a 'distance' of 1000 usec.

- if the maximum speed is set to 10000 usec/second, the move will take at least 0.1 seconds.
- if the maximum is set to 1000 usec/second, the move would take at least one second.
- if the maximum is set to 100 usec/second, the move would take at least 10 seconds
- if the maximum is set to 0 usec/second, there is no maximum speed enforced

Setting the maximum output speed ensures that the servo action is not too fast for the mechanism being controlled. For example, moving a robotic arm too quickly can cause excessive strain on the joints. The maximum output speed is stored in user-defined configurations.

Serial Interface

This section describes the process of reading and writing information to the uM-PWM1 using the serial interface. The serial interface operates at 19,200 baud with 8 data bits, no parity, and 1 stop bit.

The SEL pin on the uM-PWM1 chip must be tied to +5V to enable the RX and TX pins for the serial interface.

Command [1]	Description	Argument	Units
R*n*	read register	0 to 255	register address
W*n*=*m*	write register	*n*: 0 to 255 *m*: 0 to 65535	register address value
C*n*	select output channel	0 to 7	channel number
P*n*	set 50 Hz pulse width	0 to 65535 [2]	usec
Q*n*	set alternate Hz pulse width	0 to 65535 [2]	usec
S*n*	set speed of movement	0 to 65535	usec/sec
T*n*	set time of movement	0 to 65535	msec
F*n*	set alternate frequency	10 to 450	Hz
M	move	<none>	
H	set all output channels to home setting	<none>	
Z	disable pulse output on all channels	<none>	
L	log input channel highs on TX	500 to 65535	msec
V	print version string	<none>	
?R	display registers	<none>	
?I	display input channels	<none>	
?P	display input PPM channels	<none>	
?O	display output channels	<none>	
?F*d*	display Flash Storage Area	<none>	

Notes:

1. *n* and *m* are numeric arguments specified in decimal or hexadecimal format. *d* is one of the digits 0 through 7
2. The actual pulse width is constrained by the minimum and maximum allowable pulse widths (800 to 2200 usec in standard mode, or 500 to 2500 usec in extended mode).

Multiple commands can be entered on a single command line. Serial command lines are terminated by a carriage return (0x0D), backspace (0x08) or escape character (0x1B). The commands are executed when a carriage return is sent. If a backspace or escape character is sent, the command line is ignored. The serial input is not echoed, but a carriage return (0x0D) and linefeed (0x0A) is sent at the end of each command line. This can be used to pace the serial output if necessary. Command lines must be less than 128 characters in length, including the terminating character. Commands can be upper or lower case. Spaces and commas can optionally be used to improve the readability of the command line. The following examples both set output channel 2 to a 50Hz pulse width of 2000 microseconds:

```
C2P2000
C2, P2000
```

Numeric arguments can be specified in decimal or hexadecimal form. Hexadecimal numbers must have either a '$' 'x' or '0x' prefix. The following examples read the output frequency for channel 0 (register address is 0x80, or 128 decimal):

```
R$80
R0x80
R128
```

R*n* Read register

This R command is used to read the uM-PWM1 registers. The value *n* specifies the register address. The value returned is a 16-bit unsigned integer for all registers except registers 01 to 09 which return a 16-bit hexadecimal value. See the section, entitled *I²C interface,* for a description of the registers. Multiple read commands entered on the same line will display their results on a single line with a comma and space between each value.

e.g. Read register 1 and register 2.

serial input: `R1,R2<CR>`

serial output: `5057, 4D31<CR><LF>`

W*n*=*m* Write register

The W command is used to write to the uM-PWM1 registers. The value *n* specifies the register address, and the value *m* specifies the value to write to the register. See the section, entitled *I²C interface,* for a description of the registers.

e.g. Set the EP bit (bit 8) in mode register.

serial input: `W5=$100<CR>`

serial output: `<CR><LF>`

C*n* Select output channel

The C command is used to select the output channel for a move. The value *n* specifies the output channel. If no channel is specified, output channel 0 is assumed. This command is used before a P or Q command.

P*n* Set 50 Hz pulse width

The P command sets the selected output channel frequency to 50 Hz, and sets the next position to the value (in microseconds) specified by *n*. The value is restricted to 800 to 2200 microseconds in standard mode, or 500 to 2500 microseconds in extended mode. A value of zero will disabled the channel. If no S or T command is specified, the output channel will move immediately to the next position.

e.g. Set channel 0 to pulse width of 1000 microseconds at a frequency of 50 Hz.

serial input: `C0P1000<CR>`

serial output: `<CR><LF>`

Q*n* Set alternate pulse width

The Q command sets the selected output channel frequency to the alternate frequency, and sets the next position to the value (in microseconds) specified by *n*. The value is restricted to 800 to 2200 microseconds in standard mode, or 500 to 2500 microseconds in extended mode. A value of zero will disabled the channel. If no S or T command is specified, the output channel will move immediately to the next position.

e.g. Set channel 5 to pulse width of 900 microseconds at the alternate frequency.

serial input: `C5P900<CR>`

serial output: `<CR><LF>`

S*n* Set speed of movement

The S command sets the speed of movement for the currently selected channel. The units of speed specified by *n* are usec/sec, which provides for very precise control of the speed. For example, if a servo is moving from its current position of 1000 usec to a next position of 2000 usec, it will move a 'distance' of 1000 usec.

- if the speed is set to 10000 usec/second, the move would take 0.1 seconds.
- if the speed is set to 1000 usec/second, the move would take one second.
- if the speed is set to 100 usec/second, the move would take 10 seconds

If a maximum speed has been set for the output channel (Registers B0 to B7), then the maximum speed will be limited to this value. The T command provides an alternate way of controlling the speed of movement. If the S and T commands are both specified for an output channel, the one which specifies the slowest movement for that channel will take precedence.

e.g. Set channel 1 to pulse width of 1200 microseconds. The move from the current position will occur at the rate of 100 usec/second.

serial input: C1P1200S100<CR>
serial output: <CR><LF>

T*n* Set time of movement

The T command sets the time of movement (in milliseconds) for all output channels specified in the current move. The movement time is specified by *n*. The movement will be coordinated so that all output channels arrive at the next position simultaneously. The S command provides an alternate way of controlling the speed of movement. If the S and T commands are both specified for an output channel, the one which specifies the slowest movement for that channel will take precedence. If a maximum speed has been set for any of the output channels (Registers B0 to B7), the time of movement may be longer if the maximum speed for any channel requires a longer time of movement.

e.g. Set channel 1 to pulse width of 1200 microseconds and channel 2 to a pulse width of 800 microseconds, with both moves synchronized to complete in 1 second.
serial input: C1P1200C2P800T1000<CR>
serial output: <CR><LF>

F*n* Set alternate frequency

The F command selects the frequency for the selected channel. The frequency specified by *n*, can be 10 to 450 (Hz), or 0 to disable the channel. If the frequency is equal to the standard frequency of 50 (Hz), only the selected channel is affected. If the frequency is not 50 (Hz), the alternate frequency is set to the new value, and all output channels currently using an alternate frequency will be changed to the new alternate frequency. There is only one alternate frequency for all output channels.

e.g. Set the alternate frequency to 300 Hz.
serial input: F300<CR>
serial output: <CR><LF>

M Move

The M command initiates the movement for all channels selected in previous commands. No movement occurs until an M command, H command, or carriage return is sent.

H Home

The H command sets all output channels to the home position (initial configuration). The move occurs immediately. Any preceding S commands are ignored, but if a maximum output speed (Registers B0 to B7) has been set for any output channel, the speed of movement will be limited to this value. The T command can precede this command to control the time of movement.

Z Disable all output channels

The Z command disables all output channels. None of the output channels will have a PWM or PPM signal after sending this command. This command is immediate.

L Log input channel Highs to TX

If the serial interface RX/TX pins have been enabled, the L command will log the duration of the last high pulse for each input channel to the serial interface TX pin. The values are output as unsigned 16-bit values (0 to 65535), separated by commas, and terminated with a carriage return (CR) and linefeed (LF) character. The argument specifies the log interval in milliseconds. If the interval is zero, no logging occurs. The log is terminated if the interval is zero, or any character is received on the serial interface RX pin.

e.g. Log the input channels every five seconds.
serial input: L5000<CR>
serial output: <CR><LF>
1500, 2000, 1500, 1000, 1500<CR><LF>
1500, 2000, 1600, 1000, 1500<CR><LF>
1500, 2000, 1900, 1000, 1500<CR><LF>

```
1500, 2000, 2000, 1000, 1500<CR><LF>
...
```

V Print version string

The V command will return the version string.

e.g. Get the version string.

serial input: `L300<CR>`

serial output:
```
<CR><LF>
uM-PWM1 V1.0<CR><LF>
```

?R Display Registers

The ?R command displays the contents of all registers.

e.g.

serial input: `?R`

serial output:
```
Registers
$00: $0000 $5057 $4D31 $1003 $00CA $01C0 $00FF $0000
$08: $003F $0000   270     0 15933     0     1     0
$10:     0     0     0     0     0     0     0     0
$18:     0     0     0     0     0     0     0     0
$20:     0     0     0     0     0     0     0     0
$28:     0     0     0     0     0     0     0     0
$30:  1510  1475   900  1020  1200  1500     0     0
$38:     0     0     0     0     0     0     0     0
$40:    50    50    50    50   270   270     0     0
$48:     0     0     0     0     0     0     0     0
$50:  1500  1400  1380   850  1500   800     0     0
$58:     0     0     0     0     0     0     0     0
$60: 18500 18600 18620 19150  2204  2904     0     0
$68:     0     0     0     0     0     0     0     0
$70:   196   196   196   196  1058  1058     0     0
$78:     0     0     0     0     0     0     0     0
$80:    50    50    50    50    50    50    50    50
$88:     0     0     0     0     0     0     0     0
$90:  1510  1475  1400  1480  1495  1800  1500  1500
$98:     0     0     0     0     0     0     0     0
$A0:  1510  1475  1400  1480  1495  1800  1500  1500
$A8:     0     0     0     0     0     0     0     0
$B0:  2000  1000  1000  2000 20000 20000     0     0
$B8:     0     0     0     0     0     0     0     0
$C0:     0     0     0     0     0     0     0     0
$C8:     0     0     0     0     0     0     0     0
$D0:     0     0     0     0     0     0     0     0
$D8:     0     0     0     0     0     0     0     0
$E0:     0     0     0     0     0     0     0     0
$E8:     0     0     0     0     0     0     0     0
$F0:     0     0     0     0     0     0     0     0
$F8:     0     0     0     0     0     0     0     0
```

?I Display Input Channels

The ?I command displays the input channels.

e.g.

serial input: `?I`

serial output:
```
Input Channels
   Freq   High    Low  Count
0:   50,  1500, 18500,   196
1:   50,  1400, 18600,   196
2:   50,  1380, 18620,   196
```

```
3:   50,   850, 19150,   196
4:  270,  1500,  2204,  1058
5:  270,   800,  2904,  1058
6:    0,     0,     0,     0
7:    0,     0,     0,     0
```

?P Display Input PPM Channels

The ?P command displays the input PPM channels.

e.g.

serial input:
```
?P
```
serial output:
```
Input PPM Channels
0: 1510
1: 1475
2:  900
3: 1020
4: 1200
5: 1500
6:    0
7:    0
8:    0
9:    0
```

?O Display Output Channels

The ?O command displays the output channels.

e.g.

serial input:
```
?O
```
serial output:
```
Output Channels
   Freq   Curr   Next  MaxSpeed
0:   50,  1510,  1510,  2000
1:   50,  1475,  1475,  1000
2:   50,  1400,  1400,  1000
3:   50,  1480,  1480,  2000
4:   50,  1495,  1495, 20000
5:   50,  1800,  1800, 20000
6:   50,  1500,  1500,     0
7:   50,  1500,  1500,     0
```

?F*d* Display Flash storage area

The ?F0 command displays the I²C address, initial configuration and clock tuning value. The ?F1 *to* ?F7 commands displays the configurations storage in the Flash areas specified by the digit. If no configuration have been stored in the Flash area, a 'Not Set' message is displayed.

e.g.

serial input:
```
?F0
```
serial output:
```
Flash Block: 0
I2C Config   Tune
$CA,     1,     0
```

serial input:
```
?F1
```
serial output:
```
Flash Block: 1
 Mode   Edge  AltHz
$0100, $00FF,   270

   Freq   Home  MaxSpeed
0:   50,  1510,  2000
```

```
1:   50,  1475,  1000
2:   50,  1400,  1000
3:   50,  1480,  2000
4:   50,  1495, 20000
5:   50,  1800, 20000
6:   50,  1500,     0
7:   50,  1500,     0
```

serial input: `?F2`

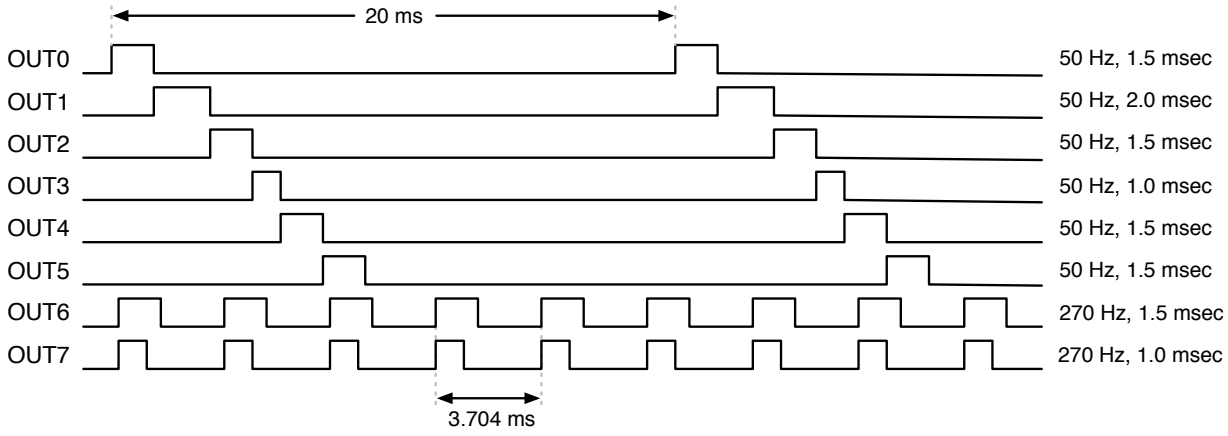
serial output:
```
Flash Block: 2
Not set.
```

Output Waveform Examples

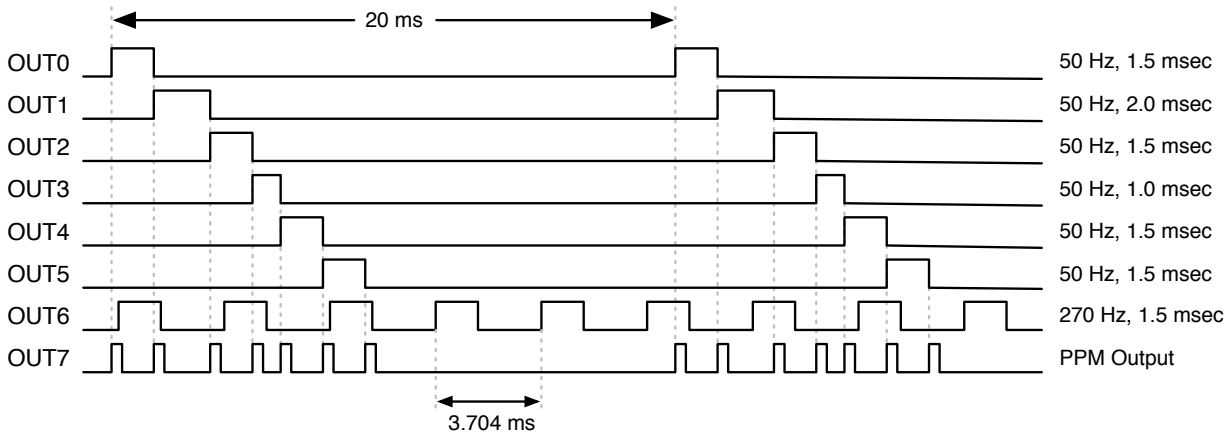
The following examples show the output waveforms for the following settings:

	Frequency (Hz)	Pulse Width (msec)
OUT0	50	1.5
OUT1	50	2.0
OUT2	50	1.5
OUT3	50	1.0
OUT4	50	1.5
OUT5	50	1.5
OUT6	270	1.5
OUT7	270	1.0

Example 1, PPM disabled

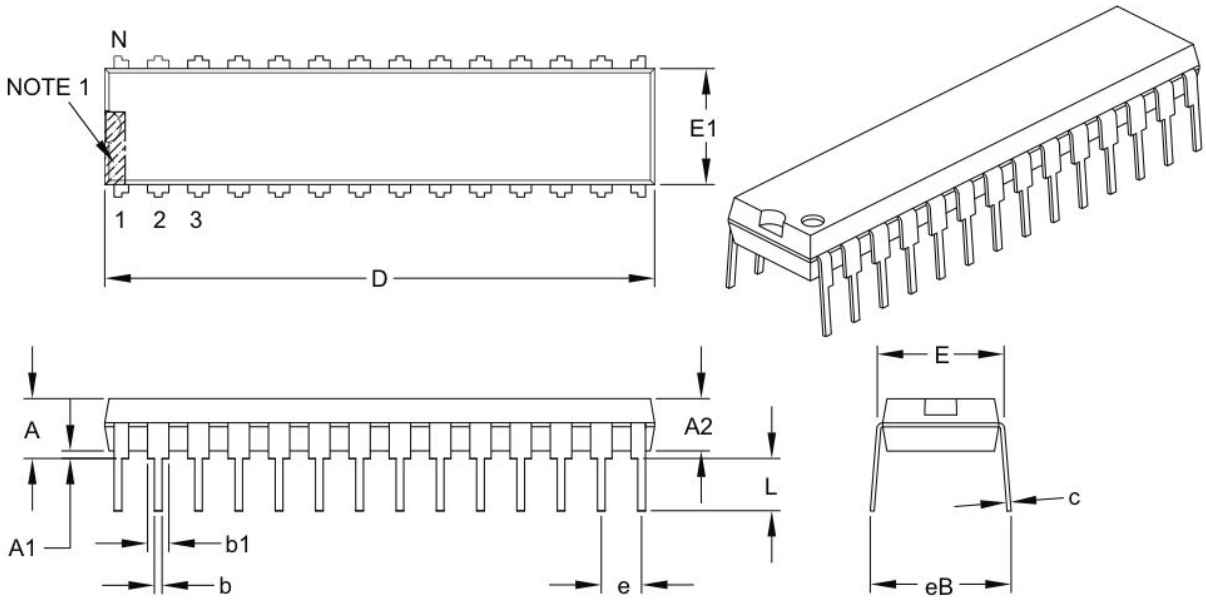


Example 2, PPM enabled



Note: Outputs that use the alternate frequency are not included in the PPM output (only 50 Hz outputs).

PDIP-28 Through-Hole Package



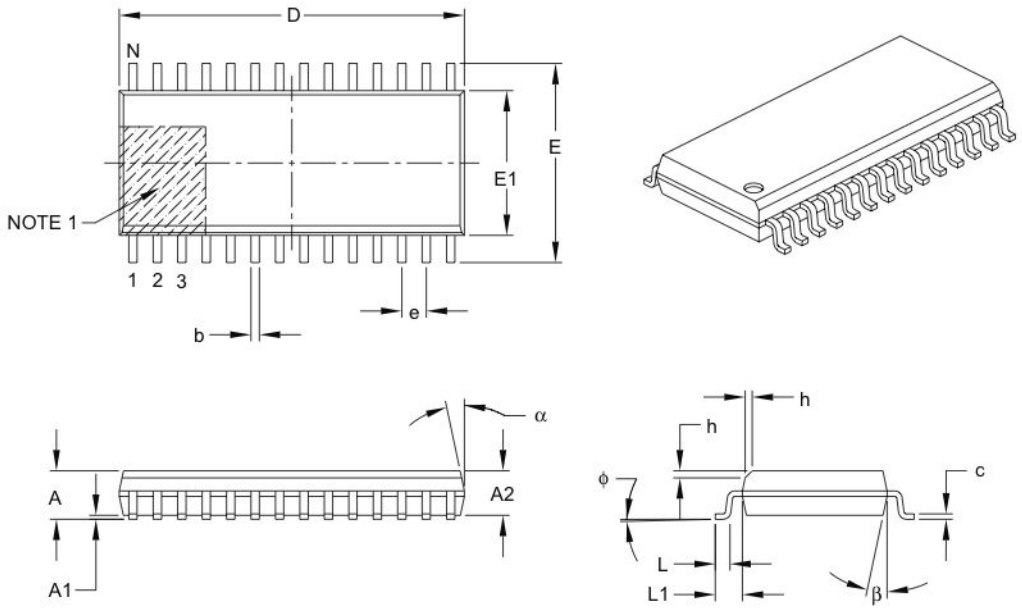
Units		INCHES		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		28	
Pitch	e		.100 BSC	
Top to Seating Plane	A	–	–	.200
Molded Package Thickness	A2	.120	.135	.150
Base to Seating Plane	A1	.015	–	–
Shoulder to Shoulder Width	E	.290	.310	.335
Molded Package Width	E1	.240	.285	.295
Overall Length	D	1.345	1.365	1.400
Tip to Seating Plane	L	.110	.130	.150
Lead Thickness	c	.008	.010	.015
Upper Lead Width	b1	.040	.050	.070
Lower Lead Width	b	.014	.018	.022
Overall Row Spacing §	eB	–	–	.430

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. § Significant Characteristic.
3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

SOIC-28 Surface Mount Package



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		28	
Pitch	e		1.27 BSC	
Overall Height	A	–	–	2.65
Molded Package Thickness	A2	2.05	–	–
Standoff §	A1	0.10	–	0.30
Overall Width	E		10.30 BSC	
Molded Package Width	E1		7.50 BSC	
Overall Length	D		17.90 BSC	
Chamfer (optional)	h	0.25	–	0.75
Foot Length	L	0.40	–	1.27
Footprint	L1		1.40 REF	
Foot Angle Top	ϕ	0°	–	8°
Lead Thickness	c	0.18	–	0.33
Lead Width	b	0.31	–	0.51
Mold Draft Angle Top	α	5°	–	15°
Mold Draft Angle Bottom	β	5°	–	15°

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. § Significant Characteristic.
3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.15 mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

   REF: Reference Dimension, usually without tolerance, for information purposes only.

Absolute Maximum Ratings

Parameter	Minimum	Maximum	Units
Storage Temperature	-65	+150	° Celsius
Ambient Temperature with Power Applied	-40	+125	° Celsius
Voltage on any pin with respect to VSS	-0.3	VDD+0.3	V
Voltage on VDD with respect to VSS	-0.3	+5.5	V
Maximum Current out of VSS pin		300	mA
Maximum Current into VDD pin		250	mA
Maximum Current sourced by any I/O pin		25	mA
Maximum Current sinked by any I/O pin		25	mA
Maximum Current sourced by all I/O pins		200	mA
Maximum Current sinked by all I/O pins		200	mA

DC Characteristics

Parameter	Minimum	Typical	Maximum	Units
Supply Voltage	4.5		5	V
I/O Pin Input Low Voltage	VSS	-	0.2 VDD	V
I/O Pin Input High Voltage	0.8 VDD	-	VDD	V
Supply Current	-	TBD	-	mA

Further Information

Check the Micromega website at www.micromegacorp.com