



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

**FACULTAD DE INGENIERIA EN ELECTRICIDAD Y
COMPUTACION**

"PÉNDULO INVERTIDO USANDO CONTROLADOR POLOLU"

TESINA DE SEMINARIO

Previa a la obtención del Título de:

INGENIERO EN ELECTRICIDAD

**ESPECIALIZACIÓN ELECTRÓNICA Y AUTOMATIZACIÓN
INDUSTRIAL**

PRESENTADO POR:

**Cevallos Suárez Yuliana Ivette
España Mayalita Israel Eduardo
Ortega Sanchez Enrique Pablo**

Guayaquil - Ecuador

2011

AGRADECIMIENTO

A Dios y a nuestros padres por su apoyo incondicional. Al Ing. Carlos Valdivieso por su valiosa ayuda en el desarrollo del presente proyecto.

DEDICATORIA

A Dios y nuestras familias.

TRIBUNAL DE SUSTENTACIÓN

Ing. Carlos Valdivieso

PROFESOR DEL SEMINARIO DE GRADUACION

Ing. Hugo Villavicencio

PROFESOR DELEGADO DECANO

DECLARACIÓN EXPRESA

“La responsabilidad del contenido de esta Tesina de Seminario, nos corresponde exclusivamente; y el patrimonio intelectual de la misma a la Escuela Superior Politécnica del Litoral”.

(Reglamento de Graduación de la ESPOL).

Yuliana Cevallos Suárez

Israel España Mayalita

Enrique Ortega Sánchez

RESUMEN

En el presente proyecto se realiza la construcción de un Péndulo Invertido usando un controlador Pololu y programación mediante el software Visual Studio4. El principal objetivo es el estudio y aplicación de del modelo matemático de este sistema que puede ser utilizado para analizar sistemas más complejos, cuyos modelos son similares al del Péndulo Invertido.

La estructura para la dinámica del sistema se la construyó en acrílico. El movimiento se lo realiza mediante dos motores pequeños manejados con un controlador PID para lograr el equilibrio del sistema. El desarrollo de este proyecto es importante para demostrar la aplicación del control moderno en la identificación de modelos matemáticos ya que el Péndulo Invertido es un modelo clásico de control.

ÍNDICE GENERAL

PORTADA	I
AGRADECIMIENTO	II
DEDICATORIA	III
TRIBUNAL DE SUSTENTACIÓN	IV
DECLARACIÓN EXPRESA	V
RESUMEN	VI
ÍNDICE GENERAL	VII
ÍNDICE DE FIGURAS	X
INTRODUCCIÓN	XII
CAPÍTULO I	1
DESCRIPCIÓN GENERAL DEL PROYECTO	1
1.1. ANTECEDENTES	1
1.2. DESCRIPCIÓN DEL PROYECTO	4
1.3. APLICACIONES DEL PRINCIPIO DE FUNCIONAMIENTO	5
1.3.1. Aplicación en la robótica	6

1.3.2.	Aplicación en el posicionamiento satelital -----	7
1.3.3.	Aplicación en estabilidad de Grúas-----	8
1.4.	PROYECTOS SIMILARES -----	9
1.4.1.	Desplazamiento horizontal sobre un riel -----	9
1.4.2.	Péndulo de Furuta -----	10
CAPÍTULO II	-----	12
FUNDAMENTO TEÓRICO	-----	12
2.1.1.	AVR STUDIO4-----	12
2.2.1.	BABY ORANGUTAN B-328-----	14
2.2.2.	PROGRAMADOR POLOLU USB AVR -----	17
2.2.3.	MICRO METAL GEARMOTOR 30:1 -----	18
2.2.4.	MEMSIC DUAL-AXIS ACCELEROMETER -----	20
CAPÍTULO III	-----	23
DISEÑO DEL PROYECTO	-----	23
3.1.	DIAGRAMA DE BLOQUES DEL DISEÑO PROPUESTO -----	23
3.2.	DISEÑO DE LA ESTRUCTURA FÍSICA-----	24
3.3.	ANÁLISIS DEL CÓDIGO DE PROGRAMACIÓN -----	26

3.4. DIAGRAMA DE FLUJO	27
CAPÍTULO IV	37
MEDICIONES Y PRUEBAS	37
CONCLUSIONES	1
RECOMENDACIONES	2
ANEXOS	3
BIBLIOGRAFIA	¡Error! Marcador no definido.

ÍNDICE DE FIGURAS

CAPÍTULO I

FIGURA 1. 1 PÉNDULO DE FURUTA -----	3
FIGURA 1. 2 PÉNDULO DE MORI (CARRETILLA-PÉNDULO) -----	3
FIGURA 1. 3 MODELO DE MICROCHIP -----	4
FIGURA 1. 4 ESTRUCTURA DEL PROYECTO -----	5
FIGURA 1. 5 PIERNAS ROBÓTICAS -----	6
FIGURA 1. 6 RANGO DE COMUNICACIÓN SATELITAL -----	7
FIGURA 1. 7 GRÚAS DE TORRE -----	8
FIGURA 1. 8 PÉNDULO INVERTIDO SOBRE UN RIEL -----	9
FIGURA 1. 9 PÉNDULO DE FURUTA -----	10
FIGURA 1. 10 PÉNDULO INVERTIDO DE MICROCHIP -----	11

CAPÍTULO II

FIGURA 2. 1 AMBIENTE DE PROGRAMACIÓN DE AVR STUDIO4 -----	13
FIGURA 2. 2 VISTA GENERAL BABY ORANGUTÁN B-328 -----	14
FIGURA 2. 3 CARACTERÍSTICAS GENERALES BABY ORANGUTÁN B- 328 -----	15
FIGURA 2. 4 CONFIGURACIÓN DE PINES ATMEGA 328P -----	16

FIGURA 2. 5 PROGRAMADOR POLOLU USB -----	17
FIGURA 2. 6 FORMA FÍSICA MICRO METAL GEARMOTOR 30:1 -----	18
FIGURA 2. 7 RUEDA POLOLU CON MICRO METAL GEARMOTOR -----	20
FIGURA 2. 8 MODULO DEL ACELERÓMETRO MEMSIC -----	21
FIGURA 2. 9 CHIP DEL ACELERÓMETRO MEMSIC-----	22

CAPÍTULO III

FIGURA 3. 1 DIAGRAMA DE BLOQUES -----	23
FIGURA 3. 2 MOTOR Y LLANTA -----	24
FIGURA 3. 3 CIRCUNFERENCIA BASE DE LA ESTRUCTURA Y PIVOTE	25
FIGURA 3. 4 ESTRUCTURA Y BASE DE ACRÍLICO -----	26

CAPÍTULO IV

FIGURA 4. 1 DIAGRAMA DE CONEXIÓN DEL ACELERÓMETRO -----	37
FIGURA 4. 2 SEÑALES DE RESPUESTA DEL ACELERÓMETRO -----	38
FIGURA 4. 3 PENDULO FINALIZADO -----	39

INTRODUCCIÓN

Nuestro proyecto tiene como objetivo la Construcción de un Péndulo Invertido usando un controlador Pololu poniendo en práctica los conceptos fundamentales del Control Automático Moderno. Uno de ellos, el controlador PID. Se usará el software AVR Studio4 con lenguaje de programación C para el desarrollo de los algoritmos del proyecto.

En el Capítulo 1 se da una descripción general del proyecto, mostrando sus antecedentes, aplicaciones y proyectos similares desarrollados. El Capítulo 2 nos mostrará los fundamentos teóricos y todas las herramientas de hardware y software usadas para el desarrollo del proyecto. El diseño será descrito en el Capítulo 3, detallando el diseño preliminar, diagramas de bloques, y los algoritmos de control. La implementación, análisis de resultados, validaciones y pruebas serán temas a tratar en el Capítulo 4

Al finalizar el proyecto presentaremos conclusiones y propondremos recomendaciones que serán útiles para el desarrollo de futuros proyectos relacionados con el péndulo invertido.

CAPÍTULO I

DESCRIPCIÓN GENERAL DEL PROYECTO

1.1. ANTECEDENTES

El Control Automático ha sido un pilar fundamental para el desarrollo de la tecnología que usamos hoy en día. Por ejemplo, sus aplicaciones han sido vitales para la automatización del área industrial. Se le ha otorgado algunas definiciones al Control Automático y todas nos describen el mismo propósito y función, destacando lo más relevante en cuanto esto; podemos decir que dicha ciencia se encarga de estudiar los modelos matemáticos de sistemas dinámicos; sean estos lineales o no. Por medio de otro sistema dinámico al que conocemos como controlador podemos modificar las propiedades de dicho modelo matemático y de ésta manera optimizarlo hasta satisfacer nuestros requerimientos. En todos los procesos que se desarrollan en el ámbito industrial, es

primordial tener un buen control ya que así se gana mayor efectividad en la producción, reducción de costos, etc.

Para aportar con nuevas ideas y técnicas de control se ha usado el concepto del Péndulo Invertido, ya que este dispositivo es un ejemplo clásico del Control Automático Moderno. Un Péndulo Invertido es según los expertos un exitoso banco de pruebas para la ingeniería del control ya que su modelo matemático es muy similar a los modelos matemáticos de procesos más complejos, el cual; cuenta con un sistema de entradas definidas y múltiples salidas, no lineal, con un alto grado de acoplamiento en sus variables. De allí se deriva la importancia de su estudio.

Sus aplicaciones son muy variadas, en su forma más básica tiene como principal utilidad las aplicaciones didácticas, aunque al agregarse a otros sistemas, el péndulo invertido es utilizado para aplicaciones que incluyen hasta la reproducción de la manera como los humanos caminamos tal como lo destacaremos posteriormente.

El primer Péndulo Invertido fue construido por los años 70, y de ahí en adelante han existido muchas variantes de este dispositivo, todas con

el mismo fin. Existe el modelo llamado el péndulo de Furuta, diseñado por el Dr. K. Furuta del Instituto de Tecnología de Tokio.

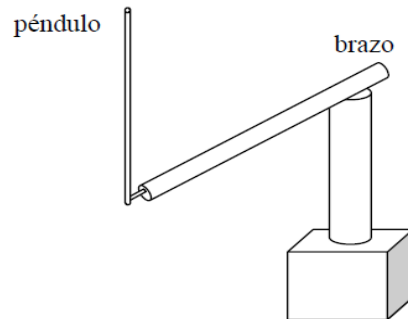


FIGURA 1. 1 PÉNDULO DE FURUTA

El siguiente modelo frecuentemente utilizado es el de Mori, mostrado en la figura 1.2, conocido también como péndulo-carretilla.

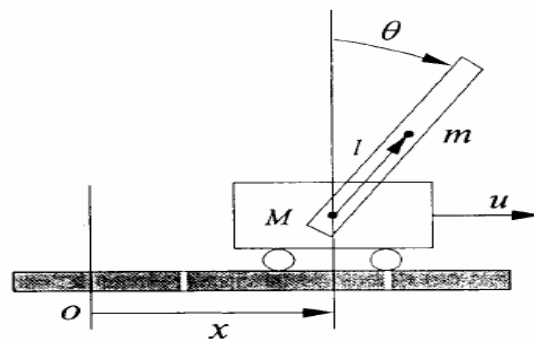


FIGURA 1. 2 PÉNDULO DE MORI (CARRETILLA-PÉNDULO)

El péndulo de Microchip que se denomina Péndulo Invertido con movimiento rotacional, lo observamos en la figura 1.3.

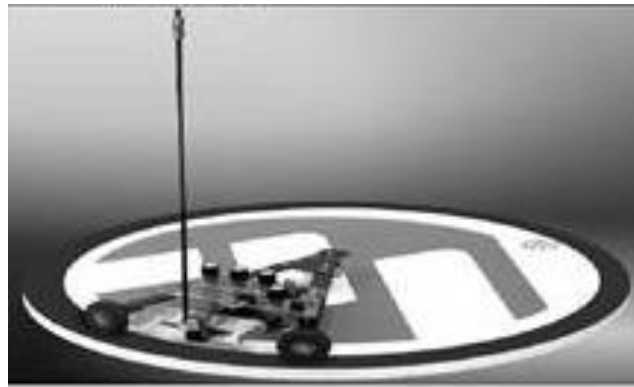


FIGURA 1. 3 MODELO DE MICROCHIP

1.2. DESCRIPCIÓN DEL PROYECTO

Nuestro Péndulo Invertido tendrá movimiento rotacional, similar al modelo de Microchip. El movimiento se da por medio de una estructura que en nuestro caso es de acrílico con dos motores colocados en sus ruedas, las cuales; se desplazan bordeando una circunferencia base, también realizada en acrílico. El carrito gira sostenido de un eje o pivote para dar el equilibrio al péndulo mediante las señales que envía el acelerómetro. Para que los motores a utilizar no se vean forzados más allá de su potencia nominal, se buscó un eje

giratorio que tuviera la menor fricción posible, y además la estructura se la diseñó para que sea muy liviana. Por tal razón se usa acrílico para lograr este cometido tal como se muestra en la figura 1.4.

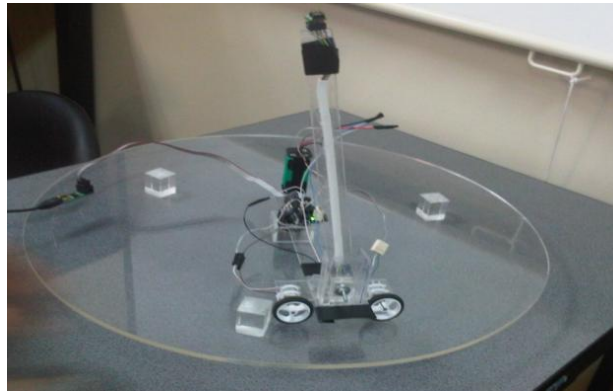


FIGURA 1. 4 ESTRUCTURA DEL PROYECTO

1.3. APLICACIONES DEL PRINCIPIO DE FUNCIONAMIENTO

El Péndulo Invertido tal como se dio a conocer previamente; es un banco de pruebas para la ingeniería del control moderno debido a su modelo matemático. Es por esto, que muchos proyectos de investigación se han llevado adelante usando el péndulo invertido.

1.3.1. Aplicación en la robótica

En el área de la robótica uno de los factores más importantes es la locomoción. Entre los diferentes tipos, tenemos la locomoción con piernas, simulando el caminar de los humanos. Los robots que poseen extremidades inferiores para moverse (Figura 1.5), deben tener un balance preciso, es lo esencial para que el cuerpo en movimiento no se caiga. Mediante el Péndulo Invertido se han desarrollado modelos que capturan una respuesta muy similar al balance de un sistema robótico.



FIGURA 1. 5 PIERNAS ROBÓTICAS

1.3.2. Aplicación en el posicionamiento satelital

En el posicionamiento de los satélites con respecto a la tierra tenemos otra aplicación del sistema del Péndulo Invertido.

La parte fija la componen las antenas ubicadas en tierra y la parte móvil el satélite (Figura 1.6). En este sistema, las antenas no deben permitir que el satélite se mueva demasiado, porque si lo hace llegará a un punto en el cual se saldrá del rango de comunicación, cumpliendo así la función de un péndulo invertido

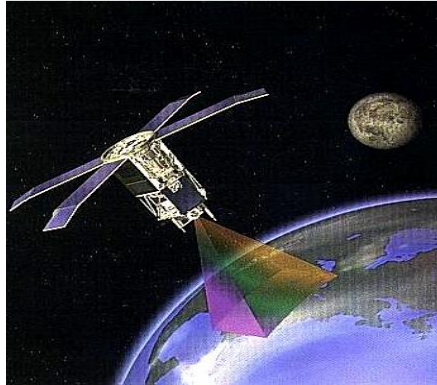


FIGURA 1. 6 RANGO DE COMUNICACIÓN SATELITAL

1.3.3. Aplicación en estabilidad de Grúas

En el control de estabilidad de grúas de torre (Figura 1.7), es otras de las aplicaciones que se le da al modelo matemático del péndulo invertido y que sin duda es una de las más utilizadas. Para lograr el equilibrio de estas estructuras, se estudia mucho el comportamiento del péndulo invertido en estado de reposo, con perturbaciones muy pequeñas.



FIGURA 1. 7 GRÚAS DE TORRE

1.4. PROYECTOS SIMILARES

Nuestro proyecto se basa en el movimiento rotacional de un dispositivo móvil que dirigirá su movimiento de acuerdo a los valores de un potenciómetro para equilibrar el péndulo. Pero existen otros modelos desarrollados, todos con el mismo objetivo, lograr el equilibrio del péndulo.

1.4.1. Desplazamiento horizontal sobre un riel

La dinámica de éste sistema se da en un riel como se puede observar en la figura 1.8. Un motor mueve una banda o cadena pequeña que le da movimiento al soporte del péndulo para equilibrarlo.

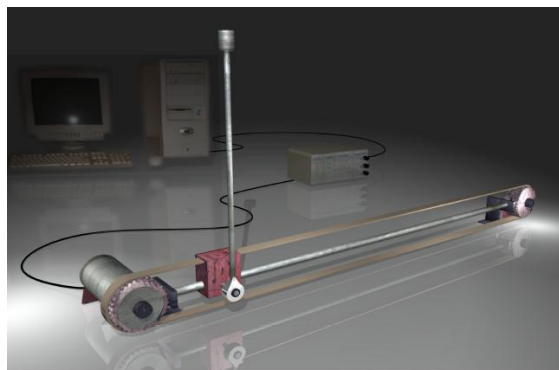


FIGURA 1. 8 PÉNDULO INVERTIDO SOBRE UN RIEL

1.4.2. Péndulo de Furuta

El péndulo de Furuta mencionado anteriormente, tiene como estructura base un eje giratorio que hace de pivote, el cual; presenta en su parte superior otro eje colocado en forma perpendicular que es el brazo sobre el cual se equilibrará el péndulo, fue uno de los primeros modelos en ser desarrollado para investigaciones. En la Figura 1.9 podemos ver la estructura física de este modelo.



FIGURA 1. 9 PÉNDULO DE FURUTA

1.4.3 Péndulo Invertido de Microchip

El péndulo invertido de Microchip se compone de tres partes principales: la plataforma de base, el péndulo y el controlador, como se muestra en la fig. 1.10. El nuestro además de diferenciarse por el controlador presenta un acelerómetro para obtener la distancia e inclinación del péndulo.



FIGURA 1. 10 PÉNDULO INVERTIDO DE MICROCHIP

CAPÍTULO II

FUNDAMENTO TEÓRICO

En el presente capítulo describiremos cada uno de los elementos que son utilizados para el desarrollo de nuestro proyecto. La herramienta de Software es AVR Studio4 y entre nuestras principales herramientas de Hardware tenemos; el controlador Pololu Baby Orangutan B-328, programador Pololu USB AVR, 2 Micro Metal Gearmotor 30:1 y el Memsic Dual-Axis Accelerometer

2.1. HERRAMIENTAS DE SOFTWARE

2.1.1. AVR STUDIO4

Es una herramienta de desarrollo para aplicaciones de Atmel AVR, la cual; nos permite realizar una programación C/C++ y utilizar código ensamblador. Entre sus características principales podemos destacar que posee un depurador que nos permite controlar la ejecución del programa y nos brinda apoyo durante la programación.

Se tiene el área de programación donde se escribe el código en el lenguaje seleccionado, así también; como diferentes áreas que nos darán indicaciones importantes o que nos permitirán realizar alguna corrección importante en nuestra programación.

En la figura 2.1 a continuación, observamos las subdivisiones de la ventana al momento de programar, la parte remarcada en rojo nos muestra la sección donde se detalla si la programación contiene errores o no; y así, proceder a realizar las debidas correcciones, mientras que en la parte superior remarcada con negro observamos el despliegue del código que vamos escribiendo.

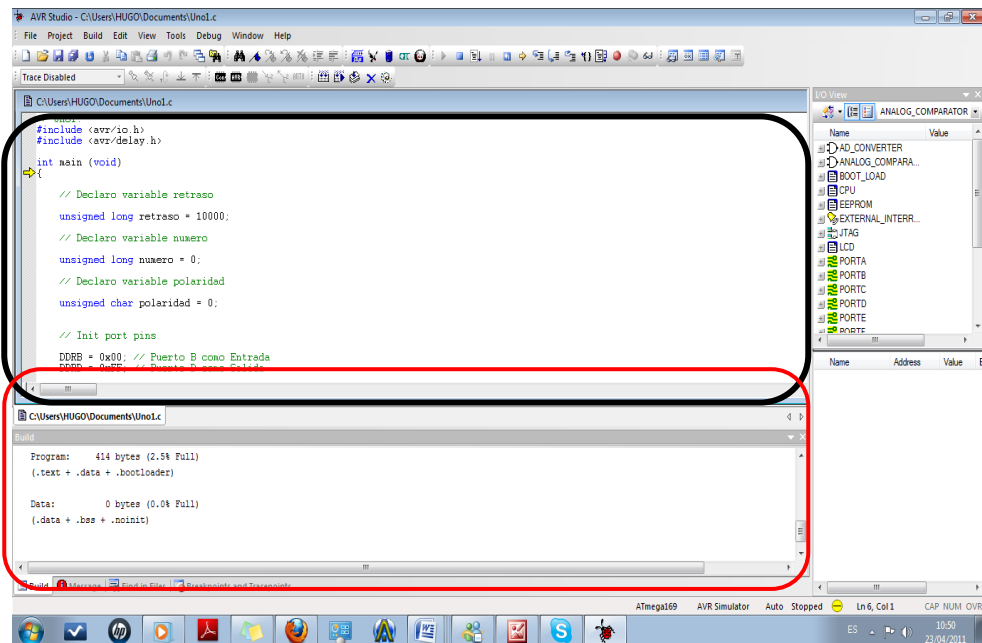


FIGURA 2. 1 AMBIENTE DE PROGRAMACIÓN DE AVR STUDIO4

2.2. HERRAMIENTAS DE HARDWARE

2.2.1. BABY ORANGUTAN B-328

Éste es el controlador Pololu que usaremos para el desarrollo de nuestro Péndulo Invertido, el cual; recibirá y enviará señales a todos los demás elementos utilizados. Por lo tanto; dicha herramienta de hardware es la más importante para el desarrollo de nuestro proyecto.

2.2.1.1. Vista general de Controlador

Es un controlador muy completo que además es una solución eficaz y eficiente para nosotros en cuanto a su tamaño como podemos apreciar en la figura 2.2. El ser compacto, es una característica de suma importancia para la estructura de nuestro proyecto ya que necesitamos que sea lo más liviana posible para que tenga una dinámica mucho más libre.



FIGURA 2. 2 VISTA GENERAL BABY ORANGUTÁN B-328

Algunas de las características básicas del hardware del Baby Orangután son dos indicadores LED en la placa, un potenciómetro, un resonador de 20 MHz, y la protección reversa de la batería, características generales que se observan en la figura a continuación.

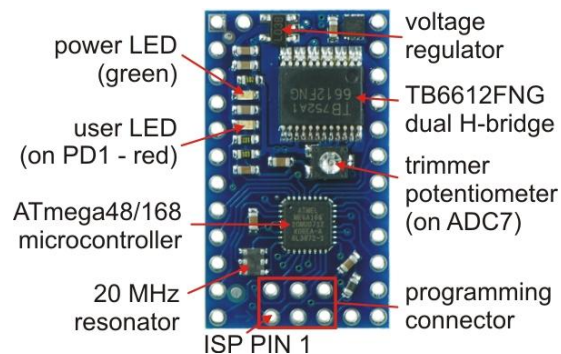


FIGURA 2. 3 CARACTERÍSTICAS GENERALES BABY ORANGUTÁN B-328

2.2.1.2. Características Físicas

- Tiene 24 pines
- Dimensiones de 1.2" x 0.7"
- Microcontrolador AVR ATmega 328P
- Doble puente H para controlar 2 motores C.C
- Dos indicadores Led
- Un Potenciómetro
- Resonador de 20 MHz
- Peso de 1.0 gr

2.2.1.3. ATmega 328P

Características

- Memoria Flash 32 KBytes
- Memoria RAM 2948 KBytes
- Memoria EEPROM 1024 Bytes
- Tamaño Vector Interrupción 2 palabras
- Alimentación Entre 2.7 V a 5.5 V DC

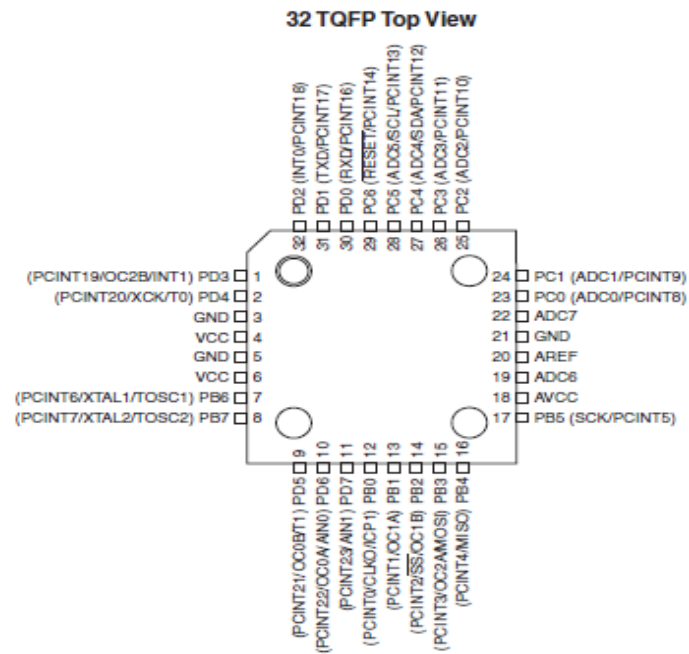


FIGURA 2. 4 CONFIGURACIÓN DE PINES ATMEGA 328P

2.2.2. PROGRAMADOR POLOLU USB AVR

2.2.2.1. DESCRIPCIÓN GENERAL

Éste dispositivo nos permite realizar la programación del controlador a través de comunicación ISP. La energía que utiliza para funcionar proviene de un ordenador personal que es conectado a través de un mini cable USB, tal como se observa en la figura 2.5.

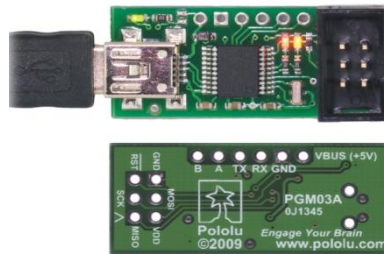


FIGURA 2. 5 PROGRAMADOR POLOLU USB

De ATMEL el programador para Microcontroladores más pequeño, por esto; también se lo considera como un dispositivo muy compacto. Además integra dos nuevas funciones, una de ellas es que funciona como Puerto Serial Virtual, para realizar prácticas de Comunicaciones Serial RS232, sin necesidad de conectar un cable adaptador Serial-USB.

2.2.3. MICRO METAL GEARMOTOR 30:1

Son pequeños motores que funcionan entre un rango de 3V a 9V, pero su voltaje ideal es de 6V, ya que; los voltajes más bajos podrían no ser prácticos y expuestos a tensiones más elevadas podrían verse afectados negativamente en cuanto a su vida útil. Casi todos tienen la misma forma física, la diferencia entre ellos radica en las relaciones de transmisión; existen desde los 5:1 a 128: 1.



FIGURA 2. 6 FORMA FÍSICA MICRO METAL GEARMOTOR 30:1

Encontramos en el mercado dos opciones diferentes de estos motores, lo que nos permite elegir que sean de alta potencia (HP) o de nivel estándar de acuerdo a nuestras necesidades en cuanto a

voltaje, corriente y torque. En nuestro proyecto utilizaremos la versión estándar del Micro Metal GearMotor.

2.2.3.1. Características

- Tamaño	24x10x12 mm
- Peso	0.34 gr
- Diámetro eje	3 mm.
- Transmisión	30:1
- Velocidad	440 rpm

2.2.3.2. Especificaciones Generales

Cuando el motor trabaja con su voltaje ideal, es decir; a 6V tenemos las siguientes especificaciones, en cuanto su corriente bajo la condición de voltaje ésta sería de 40mA y su velocidad en revoluciones por minuto 440.

El eje del motor con engranajes coincide favorablemente a las ruedas necesarias en nuestro proyecto, lo cual nos facilita enormemente en cuanto al ensamblaje de todas las partes que intervienen. La rueda a

motor es de caucho y el mecanismo para el eje de salida del motor con engranajes de metal Micro. Observamos esto en la figura 2.8.



FIGURA 2. 7 RUEDA POLOLU CON MICRO METAL GEARMOTOR

2.2.4. MEMSIC DUAL-AXIS ACCELEROMETER

El acelerómetro es un dispositivo que en general nos permite medir la aceleración, la cual; sabemos que es una medida de la rapidez de los cambios de la velocidad. Al mismo tiempo conocemos que para llegar a tener dicha aceleración es necesario inicialmente obtener otras medidas como la distancia e inclinación, es por esto; que tal dispositivo forma parte de nuestro proyecto. A continuación presentamos el modulo del acelerómetro en la figura 2.9.



FIGURA 2. 8 MODULO DEL ACELERÓMETRO MEMSIC

Esencialmente el péndulo describirá un movimiento rotacional y para la realización del mismo debemos adquirir los valores de todas las medidas antes mencionadas, como la inclinación y rotación. Gracias a éste acelerómetro obtendremos las mediciones en el eje requerido.

2.2.4.1. Características

El acelerómetro es el MEMSIC Parallax 2125 de doble eje. Este módulo mide menos de $1 / 2$ "x $1 / 2$ " x $1 / 2$ ", y el chip del acelerómetro en sí, es menos de $1/4$ " x 1.4 "x 1.8 " tal como se muestra en la figura a continuación.



FIGURA 2. 9 CHIP DEL ACELERÓMETRO MEMSIC

Sus requisitos de alimentación son en cuanto a voltaje de 3,3 a 5V DC, La corriente de alimentación debe ser <5 mA. Su temperatura de operación se encuentra entre los 32°F a 158°F (0 a 70 ° C). Consta de una comunicación TTL/CMOS, compatible con PWM de 100Hz de la señal de salida con ciclo de trabajo proporcional a aceleración.

CAPITULO III

DISEÑO DEL PROYECTO

En este capítulo presentamos la programación e implementación de nuestro proyecto, desarrollado con el software definido anteriormente, además; realizamos una descripción de la estructura física diseñada para lograr una óptima dinámica del péndulo.

3.1. DIAGRAMA DE BLOQUES DEL DISEÑO PROPUESTO

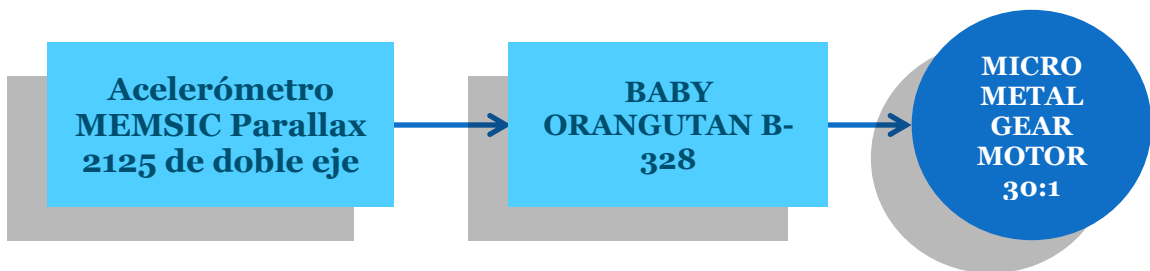


FIGURA 3. 1 DIAGRAMA DE BLOQUES

3.2. DISEÑO DE LA ESTRUCTURA FÍSICA

Como se mencionó en el capítulo 1, existen diferentes modelos para desarrollar una estructura física del péndulo invertido. En el mencionado capítulo pusimos de manifiesto que utilizaremos la estructura con movimiento rotacional, es decir; nuestra estructura está ligeramente basada en el modelo de Microchip ya que presenta muchas diferencias que surgieron de acuerdo a las necesidades de nuestro diseño.

Una de las modificaciones más visibles e importantes se realizó en la base, la cual; fue diseñada en acrílico de 2mm de espesor. Para generar el movimiento de la base del péndulo se usó dos motores Micro Metal Gearmotor 30:1, descrito anteriormente, los cuales; fueron sujetos con un par de correíllas en los extremos opuestos al eje de giro como podemos visualizar en las figuras 3.1.



FIGURA 3. 2 MOTOR Y LLANTA

Para el soporte de giro se usa un eje de motor sobre el cual se asienta un extremo de la base de la estructura (Figura 3.2). Éste motor no tiene fuente de alimentación alguna ya que solamente cumple la función de pivote para nuestro movimiento rotacional. Su movimiento dependerá de las revoluciones que den los motores instalados en la parte delantera de la estructura.

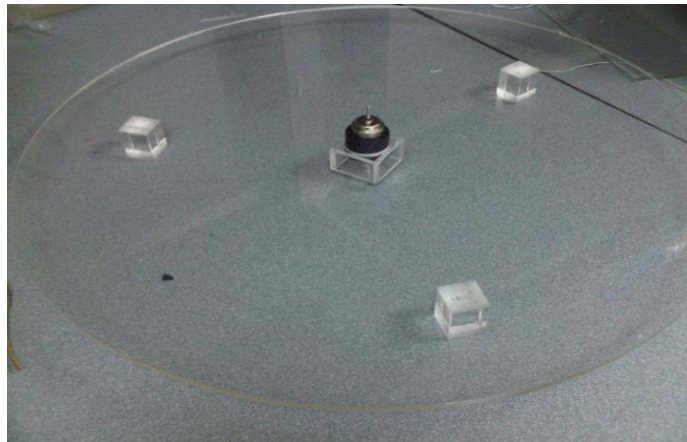


FIGURA 3. 3 CIRCUNFERENCIA BASE DE LA ESTRUCTURA Y PIVOTE

El péndulo está fijo a un pivote en la parte frontal de la estructura para que se genere el giro del mismo. El movimiento de toda la estructura antes descrita se realiza sobre una base o mesa circular que también

fue hecha en material de acrílico lo que podemos claramente observar en la figura 3.3.

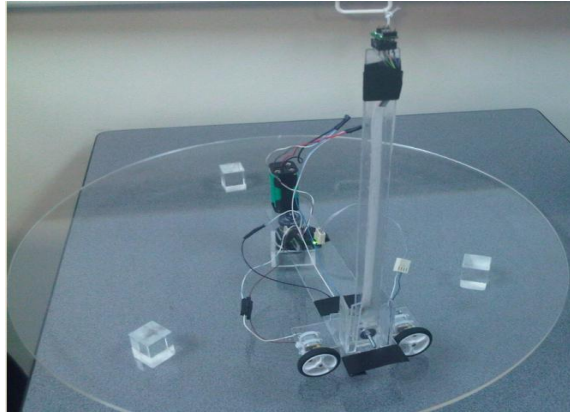
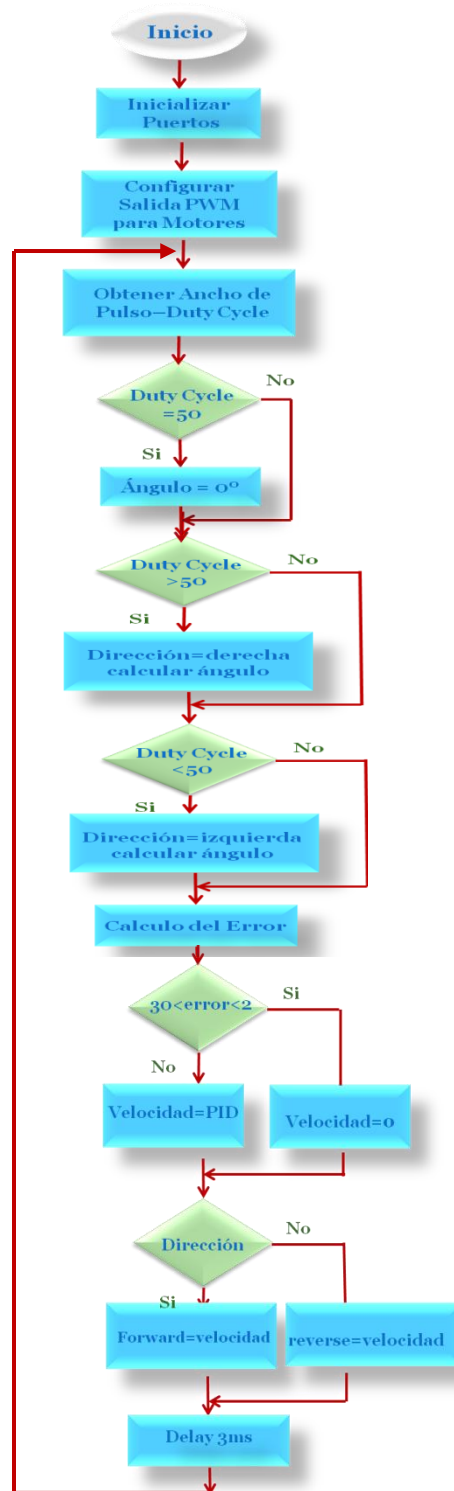


FIGURA 3. 4 ESTRUCTURA Y BASE DE ACRÍLICO

3.3. ANÁLISIS DEL CÓDIGO DE PROGRAMACIÓN

La programación realizada en lenguaje C describe el funcionamiento de los motores según la necesidad de movimiento de la estructura para lograr el equilibrio en el péndulo. Por otra parte, describe también; por medio de la programación del acelerómetro el ángulo y distancia del movimiento del péndulo durante la búsqueda de dicho equilibrio.

3.4. DIAGRAMA DE FLUJO



PROGRAMACION

```
// LIBRERIAS
```

```
#include <avr/io.h>
```

```
#include <util/delay.h>
```

```
#include <pololu/ORANGUTAN.h>
```

```
#include <math.h>
```

```
// CONSTANTES
```

```
#define pi 3.141592
```

```
#define LEDP PORTD1
```

```
#define max_motor_duty 225 // MAX PORCENTAJE DUTY
```

```
/* PID CONSTANTES */
```

```
#define kp 100 // Proporcional
```

```
#define ki 0 // Integral
```

```
#define kd 0.001 // Derivativo
```

```
// VARIABLES GLOBALES
```

```
volatile int pid_sum_error; // SUMA DE ERRORES
```



```
volatile char pid_prev_error;          // RESTA DE
ERRORES

const unsigned char pulseInPins[] = { IO_C5 }; // MEDIR
ANGULO

// FUNCIONES

void inicializar_puertos(void);

void M1_forward(unsigned char pwm);

void M1_reverse(unsigned char pwm);

void M2_forward(unsigned char pwm);

void M2_reverse(unsigned char pwm);

void motors_init();

unsigned char PID(unsigned char error1);

int main()
{
    unsigned char pwm1, error, direccion, value;

    inicializar_puertos();

    motors_init();
```

```

pulse_in_start(pulseInPins, 1); //MIDE PULSO EN
PC5
while(1) // LAZO PRINCIPAL
{
    float A, angle;
    unsigned long high_pulse =
get_last_high_pulse(0);
    unsigned long period_in_ticks = high_pulse +
get_last_low_pulse(0);
    unsigned long T = 0.4 * high_pulse;
    unsigned long duty_cycle_percent = (100 *
high_pulse + period_in_ticks/2) / period_in_ticks;
    if (duty_cycle_percent == 50)
    {
        angle=0;
        pid_sum_error=0;
        error=angle;
    }else if (duty_cycle_percent >50)
    {

```

```

A = (((float)T / 10.0) - 500) * 8) /
1000;

angle = ((asin((float)A))*360.0)/ (2*
pi);

error=angle;
direccion=1;
}else if (duty_cycle_percent <50)
{
A=(((float)T/10.0)-500)*8)/1000;
angle=((asin((float)A)*-
1)*360.0)/(2*pi);

error= angle;
direccion=0;
}
if ((error<8) || (error > 20))
{
value = 0;
}else value = PID(error);
if (direccion==0)
{

```

```
        M1_forward(value);
        M2_forward(value);
    }else{
        M1_reverse(value);
        M2_reverse(value);
    }
    value = PID(error);
    if (value == 0)
    {
        PORTD &= (0<<LEDP);
    }else PORTD |= (1<<LEDP);
    _delay_ms( 10 );
    }
}
```

```
void inicializar_puertos(void)
```

```
{
    DDRD=0x6A; //0110 1011 0,1,3,5,6 Salidas
    PORTD=0x00;
    DDRB=0x0A; //0000 1010 1,3 Salidas
```

```
PORTB=0x00;
DDRC=0x01; //0000 0001 0 Salida
PORTC=0x00;
}
```

```
void M1_reverse(unsigned char pwm)
{
    OCR0A = 0;
    OCR0B = pwm;
}
```

```
void M1_forward(unsigned char pwm)
{
    OCR0B = 0;
    OCR0A = pwm;
}
```

```
void M2_reverse(unsigned char pwm)
{
    OCR2A = 0;
    OCR2B = pwm;
```

```
}  
  
void M2_forward(unsigned char pwm)  
{  
    OCR2B = 0;  
    OCR2A = pwm;  
}  
  
  
void motors_init()  
{  
    // CONFIGURAR PWM INVERTIDO:  
    // Timer0 , Timer2 EN MODO COMPARADOR  
    TCCR0A = TCCR2A = 0xF3; //system_clock/8(=2.5 MHz)  
    TCCR0B = TCCR2B = 0x02;  
    OCR0A=OCR0B=OCR2A= OCR2B = 0; //inicializa PWMs en  
0%  
    DDRD |= (1 << PORTD3) | (1 << PORTD5) | (1 <<  
PORTD6);  
    DDRB |= (1 << PORTB3);
```

```
}
```

```
unsigned char PID(unsigned char error1)
```

```
{
```

```
    float prop_term;
```

```
    float integ_term;
```

```
    float deriv_term;
```

```
    float sum_terms;
```

```
    int velocity;
```

```
    prop_term = 0;
```

```
    integ_term = 0;
```

```
    deriv_term = 0;
```

```
    sum_terms = 0;
```

```
    if (ki != 0) {
```

```
        pid_sum_error += error1;
```

```
        integ_term = ki*(float)pid_sum_error;
```

```
        sum_terms += integ_term;
```

```
    }
```

```
    if (kd != 0) {
```

```
        deriv_term = kd*(float)(error1-pid_prev_error);
```

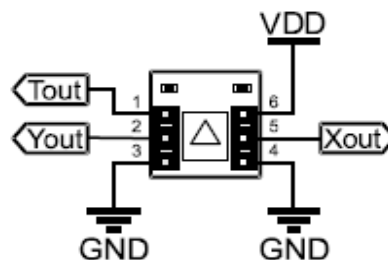
```
        pid_prev_error = error1;
        sum_terms += deriv_term;
    }
    prop_term = kp*(float)error1; // termino proporcional PID
    sum_terms += prop_term;
    if (sum_terms > max_motor_duty) { velocity =
max_motor_duty; }
    else { velocity = (int)sum_terms; }
    return velocity;
}
```


CAPÍTULO IV

MEDICIONES Y PRUEBAS

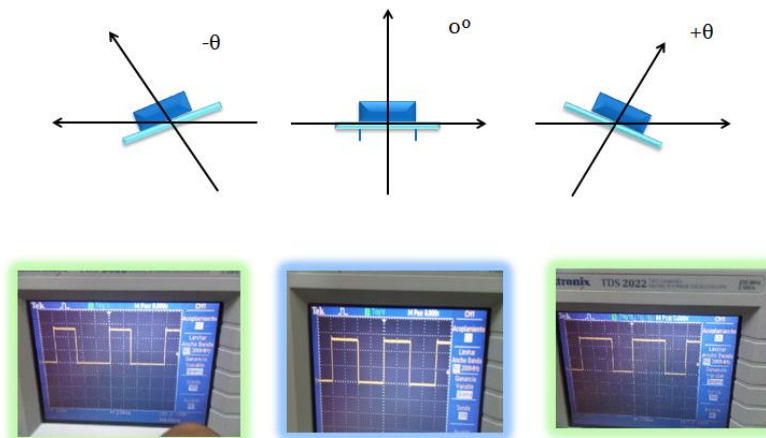
En este capítulo presentaremos las pruebas realizadas durante la implementación del Péndulo Invertido y la obtención de los valores que entregaba el sensor al controlador durante las mediciones y variaciones de parámetros.

Adquirimos los valores de los ángulos de inclinación que tomaba el Péndulo por medio de una lectura en un eje del elemento sensor que utilizamos que es el Acelerómetro que se lo conecta como se muestra en la figura 4.1.



**FIGURA 4. 1 DIAGRAMA DE CONEXIÓN DEL
ACELERÓMETRO**

En la figura 4.2 podemos observar la señal que nos daba el acelerómetro al realizar las pruebas, la cual; es una señal PWM que al estar en posición horizontal nos entrega una onda cuadrada con frecuencia de 100 Hertz y ancho de pulso de 5 ms. Dichas señales fueron observadas mediante el uso de un osciloscopio. Se pudo notar que al girar el acelerómetro en un sentido disminuía el ancho de pulso y al girarlo en el sentido opuesto este ancho de pulso aumentaba.



**FIGURA 4. 2 SEÑALES DE RESPUESTA DEL
ACELERÓMETRO**

En las pruebas realizadas a los motores se requería que los mismos alcancen la velocidad adecuada para que el péndulo recupere su posición inicial. A la vez necesitábamos que giren en el sentido en el cual el péndulo cae para así lograr un movimiento que compense la caída del péndulo y este logre su equilibrio.

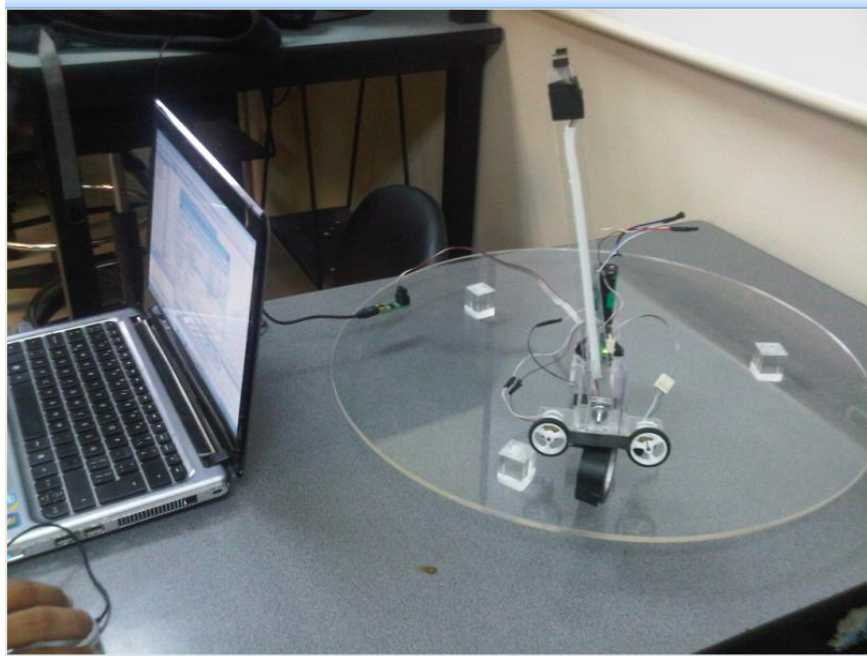


FIGURA 4. 3 PENDULO FINALIZADO

CONCLUSIONES

- 1) Durante la realización de este trabajo aprendimos que para seleccionar los dispositivos adecuados para la realización de un proyecto debemos tener muy en cuenta las características físicas de cada uno de los elementos para que puedan cumplir con el trabajo que deben realizar.
- 2) Al momento de realizar la programación con el software que estemos utilizando, debemos tener muy en claro la forma de escribir las sentencias para de esa manera poder plantear el algoritmo que deseamos implementar en nuestro proyecto, y así poder aprovechar todas la herramientas que nos brinda.
- 3) Se concluye que es importante desarrollar la programación en base a al análisis previo de los principios de funcionamiento de cada elemento que participa en el proyecto porque de esta manera no tendremos inconvenientes ni contratiempos que perjudiquen la realización del mismo.

RECOMENDACIONES

- 1) Es recomendable revisar y comprobar que el hardware a utilizar se encuentre en buen estado y a la vez nos sea útil hará las tareas que le son encomendadas y no olvidar considerar: peso, funcionalidad, características eléctricas, etc.
- 2) Para desarrollar aplicaciones con el sensor utilizado, es necesario conocer qué tipo de señal nos entrega para de esa manera saber de qué forma vamos a leer dicha señal en nuestro Microcontrolador y así poder interpretarla de manera correcta.
- 3) Al momento de realizar las conexiones, tener muy en cuenta la temperatura a la que está funcionando el elemento que estemos probando y de esta manera evitar daños en el mismo, por lo que debemos estar permanentemente tocando el elemento.

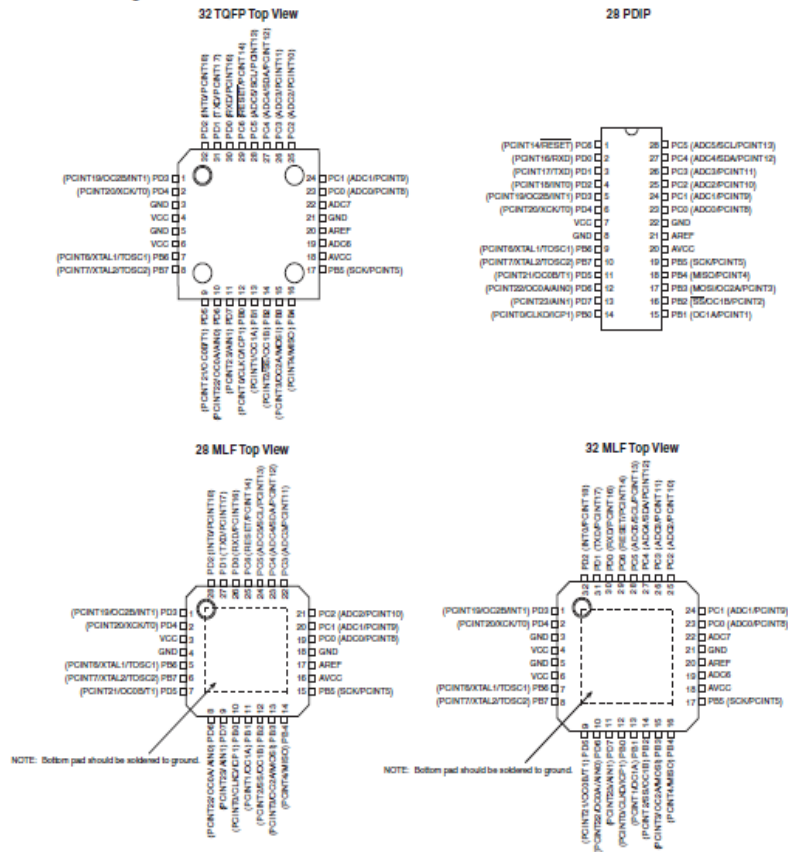
ANEXOS

Atm 328

ATmega48A/48PA/88A/88PA/168A/168PA/328/328P

1. Pin Configurations

Figure 1-1. Pinout ATmega48A/48PA/88A/88PA/168A/168PA/328/328P



ATmega48A/48PA/88A/88PA/168A/168PA/328/328P

1.1 Pin Descriptions

1.1.1 VCC

Digital supply voltage.

1.1.2 GND

Ground.

1.1.3 Port B (PB7:0) XTAL1/XTAL2/TOSC1/TOSC2

Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Depending on the clock selection fuse settings, PB6 can be used as input to the inverting Oscillator amplifier and input to the internal clock operating circuit.

Depending on the clock selection fuse settings, PB7 can be used as output from the inverting Oscillator amplifier.

If the Internal Calibrated RC Oscillator is used as chip clock source, PB7...6 is used as TOSC2...1 input for the Asynchronous Timer/Counter2 if the AS2 bit in ASSR is set.

The various special features of Port B are elaborated in ["System Clock and Clock Options"](#) on page 26.

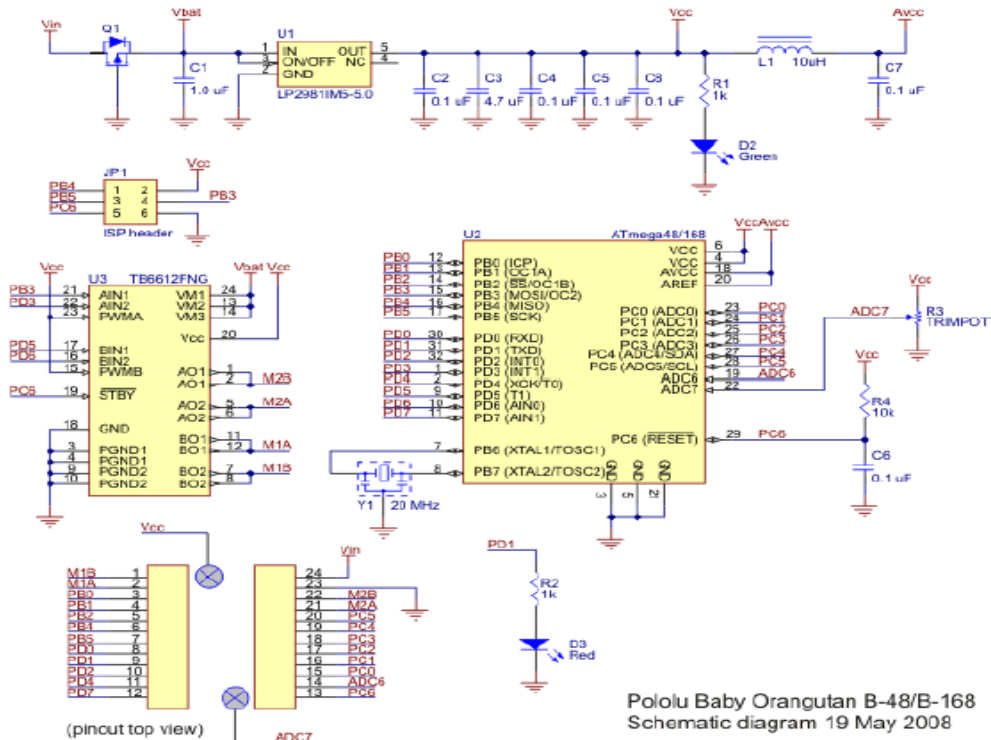
1.1.4 Port C (PC5:0)

Port C is a 7-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The PC5...0 output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running.

1.1.5 PC6/RESET

If the RSTDISBL Fuse is programmed, PC6 is used as an I/O pin. Note that the electrical characteristics of PC6 differ from those of the other pins of Port C.

If the RSTDISBL Fuse is unprogrammed, PC6 is used as a Reset input. A low level on this pin for longer than the minimum pulse length will generate a Reset, even if the clock is not running.



Pololu Baby Orangutan B-48/B-168/B-328 schematic diagram.

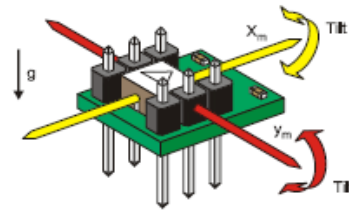
Acelerómetro

Memsic 2125 Dual-Axis Accelerometer (#28017)

The Memsic 2125 is a low-cost thermal accelerometer capable of measuring tilt, collision, static and dynamic acceleration, rotation, and vibration with a range of ± 3 g on two axes. Memsic provides the 2125 IC in a surface-mount format. Parallax mounts the circuit on a tiny PCB providing all I/O connections so it can easily be inserted on a breadboard or through-hole prototype area.

Features

- Measures ± 3 g on each axis
- Simple pulse output of g-force for each axis
- Convenient 6-pin 0.1" spacing DIP module
- Analog output of temperature (TOut pin)
- Fully temperature compensated over 0 to 70 °C operating temperature range



Key Specifications

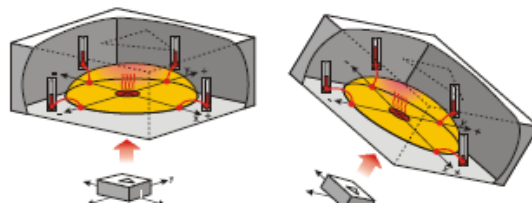
- Power Requirements: 3.3 to 5 VDC; < 5 mA supply current
- Communication: TTL/CMOS compatible 100 Hz PWM output signal with duty cycle proportional to acceleration
- Dimensions: 0.42 x 0.42 x 0.45 in (10.7 x 10.7 x 11.8 mm)
- Operating temperature: 32 to 158 °F (0 to 70 °C)

Application Ideas

- Dual-axis tilt and acceleration sensing for autonomous robot navigation
- R/C tilt controller or autopilot
- Tilt-sensing Human Interface Device
- Motion/lack-of-motion sensor for alarm system
- Single-axis rotational angle and position sensing

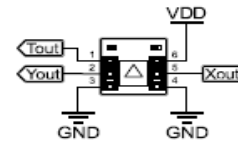
Theory of Operation

The MX2125 has a chamber of gas with a heating element in the center and four temperature sensors around its edge. When the accelerometer is level, the hot gas pocket rises to the top-center of the chamber, and all the sensors will measure



at www.parallax.com.

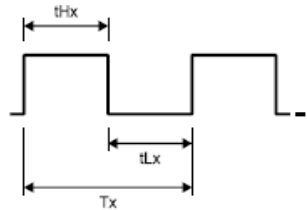
Pin	Name	Function
1	Tout	Temperature Out
2	Yout	Y-axis PWM output
3	GND	Ground -> 0 V
4	GND	Ground -> 0 V
5	Xout	X-axis PWM output
6	VDD	Input voltage: +3.3 to +5 VDC



Communication Protocol

Each axis has a 100 Hz PWM duty cycle output in which acceleration is proportional to the ratio t_{Hx}/T_x . In practice, we have found that T_x is consistent so reliable results can be achieved by measuring only the duration of t_{Hx} . This is easy to accomplish with the BASIC Stamp PULSIN command or with the Propeller chip's counter modules.

With $V_{dd} = 5V$, 50% duty cycle corresponds to 0 g, but this will vary with each individual unit within a range of 48.7% to 51.3%. This zero offset may be different when using $V_{dd} = 3.3 V$. See the manufacturer's datasheet for details.



Example Circuit

The example schematic and wiring diagram below are for the BASIC Stamp and Board of Education.

