

**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**

**FACULTAD DE INGENIERÍA**

**DISEÑO DE GENERADORES DE AVENIDAS Y MAREAS EN  
MODELOS FLUVIALES**

**TESIS**

**QUE PARA OBTENER EL TÍTULO DE  
INGENIERO MECÁNICO ELECTRICISTA  
ÁREA ELÉCTRICA ELECTRÓNICA**

**PRESENTAN:**

**ENRIQUE ARTURO DEL RÍO HERNÁNDEZ  
FEDERICO JORGE LENTZ HERRERA**

**DIRECTOR DE TESIS:**

**ING. RODOLFO PETERS LAMMEL**

**MÉXICO, D.F.**

**SEPTIEMBRE DE 2007**



Universidad Nacional  
Autónoma de México

Dirección General de Bibliotecas de la UNAM

**Biblioteca Central**



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

## Dedicatorias

Enrique:

- A Dios.
- A mi padre, a quién siempre recordaré por su ejemplo.
- A mi madre, por esa manera única de ella de brindar apoyo.
- A mi esposa, que ha sido una piedra angular.
- Y a mi máximo logro; mis hijos.

Federico:

- A Dios, que me llena de bendiciones todos los días.
- A mi mamá, por su amor, por su ejemplo de tenacidad y fortaleza ante la vida y por no dejarme desistir.
- A mi papá, por estar a mi lado siempre, por su apoyo, su amor y su alegría.
- A Álvaro mi hermano, por ser mi amigo y por brindarme en todo momento su ayuda incondicional.

## Agradecimientos

Enrique y Federico:

- A la Universidad Nacional Autónoma de México y a nuestros maestros, por ser parte fundamental en nuestra formación como profesionistas y como personas.
- Al Ing. Peters, por su apoyo, su guía y su paciencia durante todo el tiempo que duró la realización de este proyecto.
- A nuestros sinodales:
  - Ing. Francisco José Rodríguez Ramírez
  - Ing. Rodolfo Peters Lammel
  - Ing. Rafael Iriarte Vivar Balderrama
  - M. I. Norma Elva Chávez Rodríguez
  - Ing. Valente Vázquez TamayoPor sus aportaciones y interés en la revisión de este trabajo.

Enrique:

- A mi mejor amigo, quien me ha mostrado en todo momento su apoyo, y paciencia; gracias, Lico.

Federico:

- A Enrique, por su amistad, su apoyo y sus consejos.
- A todos mis familiares y amigos, que me han apoyado y motivado durante este proceso.

# Índice

Índice .....	iv
Introducción.....	1
Capítulo 1. Generadores de Avenidas y Mareas.....	5
1.1. Generadores de Avenidas .....	6
1.1.1. Válvulas tipo Guillotina.....	6
1.1.2. Válvulas de liberación de caudal .....	8
1.1.3. Válvula mixta de guillotina.....	9
1.1.4. Válvula para control de avenidas.....	10
1.2. Generadores de Mareas .....	11
1.2.1. Sistema de bombeo y compuerta .....	13
1.2.2. Sistema de bombeo y válvula de doble flujo. ....	14
1.3. Control en válvulas .....	16
1.3.1. Diagrama de control de motores con circuito tipo H. ....	18
Capítulo 2. Transductores (Sensores).....	22
2.1. Sensores de nivel .....	23
2.1.1. Sensores de nivel por transferencia de calor.....	23
2.1.2. Sensores de nivel de tipo óptico .....	24
2.1.3. Sensores de nivel por amortiguamiento de oscilaciones mecánicas .....	24
2.1.4. Sensores continuos de nivel.....	24
2.1.5. Sensores capacitivos .....	24
2.1.6. Sensores que se utilizarán en el proyecto y cómo se implementarán. ....	28
2.1.7. Circuito Astable .....	31
2.2. Sensores de flujo.....	32
2.2.1. Por presión diferencial.....	32
2.2.2. Por elementos mecánicos.....	33
2.2.3. Por características del fluido.....	33
2.2.4. Sensor de flujo de turbina.....	33
2.2.5. Medidor de flujo por flujo oscilante .....	34
2.2.6. Otros medidores de flujo .....	34
2.2.7. Sensores que se utilizarán en el proyecto y cómo se implementarán .....	34
2.3. Sensores de posición y desplazamiento.....	36
2.3.1. Transductores de desplazamiento capacitivos.....	36
2.3.2. Transductores de desplazamiento inductivos.....	36
2.3.3. Transductores de desplazamiento por reluctancia.....	36
2.3.4. Transductores de desplazamiento potenciométricos.....	37
2.3.5. Transductores de desplazamiento por hilo vibrante.....	37
2.3.6. Transductores de desplazamiento electro ópticos.....	37
2.3.7. Sensores que se utilizarán en el proyecto y cómo se implementarán. ....	37
Capítulo 3. Sistema de Control.....	39
3.1. Transductores .....	41
3.2. Acondicionamiento de señal.....	41
3.3. Transmisión de datos.....	42
3.4. Microcontroladores.....	43
3.4.1. Microcontrolador MC68HC11 .....	43
3.5. Medición de Nivel y de Flujo .....	44
3.5.1. Sistema temporizador del MC68HC11 .....	45
3.5.2. Captura de entradas.....	47
3.5.3. Medición de Período y Frecuencia .....	50
3.6. Posición de las compuertas .....	55
3.6.1. Convertidor Analógico Digital .....	55
3.6.2. Multiplexor del convertidor analógico digital.....	56

3.6.3.	Convertidor Analógico .....	56
3.6.4.	Control Digital .....	57
3.6.5.	Registros de Resultados.....	57
3.6.6.	Relojes del Convertidor A/D .....	57
3.6.7.	Secuencia de conversión.....	58
3.6.8.	Encendido del convertidor A/D y selección de Reloj.....	58
3.6.9.	Proceso de Conversión .....	59
3.6.10.	Asignación de Canales.....	60
3.6.11.	Operación de Canal Simple .....	60
3.6.12.	Operación de canales múltiples .....	61
3.6.13.	Operación en modo STOP y WAIT.....	61
3.6.14.	Registros de estado/control A/D.....	62
3.6.15.	Registros de resultados del convertidor A/D .....	64
3.7.	Computadora Personal.....	65
3.7.1.	Comunicaciones entre el MC68HC11 y la computadora personal .....	65
3.7.2.	Formato de datos.....	66
3.7.3.	Operación de Transmisión.....	67
3.7.4.	Operación de Recepción.....	67
3.7.5.	Detección de errores .....	67
3.7.6.	Registros .....	68
3.7.7.	Registro de Datos de Comunicación en Serie.....	69
3.7.8.	Registro de Control de Comunicación en Serie 1 .....	69
3.7.9.	Registro de Control de Comunicaciones en Serie 2.....	70
3.7.10.	Registro de estado de Comunicaciones en Serie.....	71
3.7.11.	Registro de Tasa de Transferencia.....	72
3.8.	Lógica Difusa .....	74
3.8.1.	Difusión .....	75
3.8.2.	Evaluación de reglas .....	77
3.8.3.	Desdifusión .....	78
3.8.4.	Control difuso .....	80
3.9.	Control de los motores.....	83
3.9.1.	Salidas de Comparación .....	84
3.9.2.	Registros de Salidas de Comparación del Temporizador .....	86
3.9.3.	Registro de comparación forzada del temporizador .....	87
3.9.4.	Registro de Datos de Salida de Comparación.....	88
3.9.5.	Registro contador del temporizador.....	89
3.9.6.	Registro de Máscara de Interrupción del Temporizador 1.....	90
3.9.7.	Registro de Bandera de Interrupción del temporizador 1 .....	90
3.9.8.	Registro de Máscara de Interrupción del Temporizador 2.....	91
3.9.9.	Registro de Banderas de Interrupción del Temporizador 2 .....	92
3.9.10.	Implementación de la modulación por Ancho de pulso.....	93
3.10.	Circuitos de protección.....	94
3.10.1.	Solicitud de Interrupción no mascarable (XIRQ).....	95
3.11.	Multiplexor.....	96
3.11.1.	Salidas paralelas del Microcontrolador.....	98
3.11.2.	Multiplexión en tiempo.....	98
3.12.	Desarrollo del Programa de Control del Microprocesador.....	98
3.12.1.	Funciones principales del programa del microcontrolador.....	99
3.12.2.	El módulo de evaluación y el programa monitor.....	101
Anexo A.	Índice de Tablas y Figuras.....	109
	<i>Índice de Figuras</i> .....	109
	<i>Índice de Tablas</i> .....	109
	Bibliografía.....	110

## **Introducción**

Dentro del estudio de la hidráulica existen actualmente diferentes herramientas y técnicas dedicadas al análisis de los diferentes fenómenos y problemas prácticos de ingeniería; entre las técnicas de estudio se encuentran los modelos físicos a escala reducida, los modelos matemáticos y los modelos análogos.

Debido a la complejidad de los fenómenos de la naturaleza, particularmente dentro del campo de la hidráulica es necesario valerse de los modelos físicos a escala reducida, ya que muchas veces no es suficiente tratarlos únicamente con modelos matemáticos.

Los métodos matemáticos plantean soluciones generalmente idealizadas, lo que permite hacer simplificaciones importantes. Dichas simplificaciones dejan de considerar ciertos aspectos y sus efectos, los que pueden ser valorados mediante ensayos experimentales a través de modelos físicos a escala reducida o de tipo analógico.

El laboratorio de Hidráulica de la Comisión Federal de Electricidad requiere diseñar y construir un generador de mareas y avenidas para un modelo fluvial a gran escala para la Comisión Nacional del Agua.

En dicho modelo se pretenden medir los impactos de avenidas que causan los ríos alimentadores del río Pánuco y los efectos de la marea en el delta del mismo; debido a la orografía que se tiene en el delta del río Pánuco, en las diferentes épocas del año, en las que se presentan ciertos efectos climatológicos como son las lluvias, el crecimiento del río y la variación en las mareas, ya que existen grandes pérdidas y destrucción a causa de los mismos.



Por esto se ha planteado la necesidad de controlar el flujo de agua dentro del río mediante sistemas de presas, para el sobre afluyente debido a la lluvia, y a la marea.

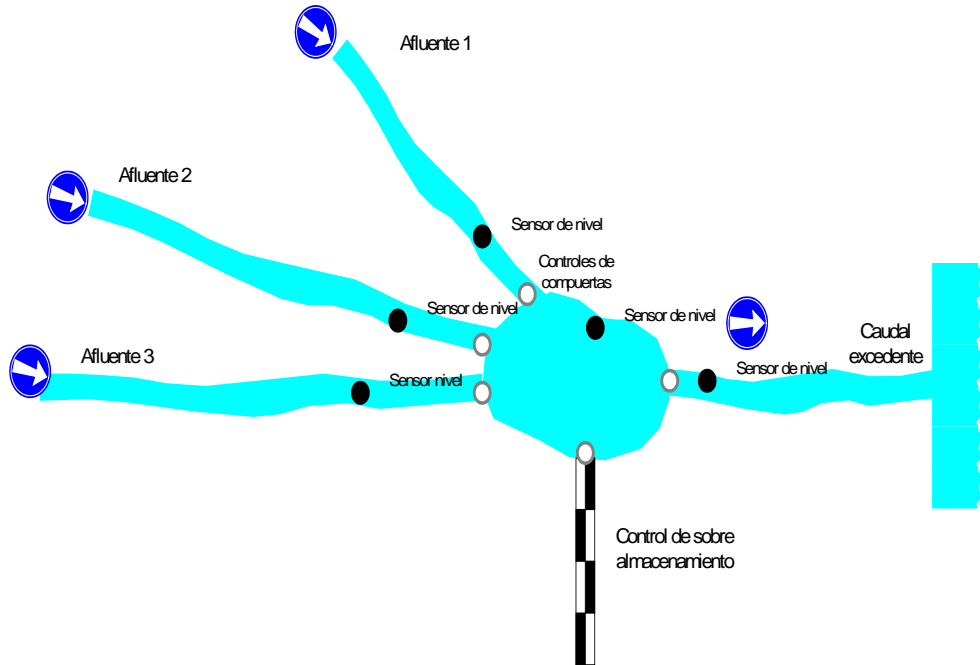
En el presente trabajo de tesis se discutirán los métodos, caminos o procedimientos mediante los cuales se puede resolver el problema e indicarse cual o cuales de ellos adoptar y por qué. Una vez seleccionado el método a seguir, éste se describirá detalladamente. El control de los generadores se hará en función de datos recabados en el Pánuco y contrastados con las mediciones hechas en el modelo.

El procedimiento a emplearse será en función de dispositivos mecánicos activados con motores y sus respectivos controladores. Como elementos sensores se tendrán medidores de nivel de tipo capacitivo. Adicionalmente se generarán una serie de bases de datos con las mediciones realizadas por dichos medidores de nivel. Todo el sistema de control se basa en los datos estadísticos que se han recabado a través del tiempo.

Los generadores que se describen en el capítulo 1 servirán para evaluar el desempeño de los sistemas de medición y de control y para cubrir las necesidades del río.

Para este proyecto se necesitará la utilización de sistemas de medición como son los sensores capacitivos descritos en el capítulo 2 y sistemas de control, descritos en el capítulo 3, basándose en un microcontrolador con un programa capaz de interpretar los datos adquiridos por los diferentes sensores, y transformarlos en señales eléctricas, que a su vez se conectarán a motores, para así lograr el control de los flujos acuíferos.

A continuación se presenta un diagrama esquemático para mostrar de una mejor forma como se hace el desarrollo del sistema; en él se indican las ubicaciones de las compuertas, de los sensores, y de los dispositivos de control.



**Figura 1. Diagrama General del Sistema**

Como se mencionó en el capítulo 3 se describirá el proceso de control del sistema, el cual está dado por un sistema de acondicionamiento de señal, un sistema de transmisión de datos, un microprocesador y una PC, así como un programa que sea capaz de recabar la información proveniente de los diferentes transductores y almacenar dicha información en los dispositivos de memoria, y al mismo tiempo, utilizar esa información y aquella contenida en las diferentes bases de datos, para generar las diferentes señales de control a los diferentes motores, con la finalidad de generar las avenidas y las mareas dentro del modelo.

## **Capítulo 1. Generadores de Avenidas y Mareas**

## 1.1. Generadores de Avenidas

Una avenida es el aumento o crecida impetuosa y violenta de un cauce de agua, generalmente de un río o arroyo.

Los dispositivos para generar avenidas son utilizados en los modelos hidráulicos a escala reducida para proporcionar un caudal, cuyo flujo puede ser controlado para reproducir así las variaciones de gasto propias de un río. Éstos se basan en aquellos componentes destinados al control de gasto o flujo, entre los cuales podemos encontrar diferentes tipos de válvulas y de compuertas.

Las compuertas son los dispositivos que hacen la interacción entre el sistema de control (parte electrónica) y el fluido.

Nos permiten hacer la regulación del caudal, controlar los niveles, de acuerdo a las necesidades específicas para cada canal.

### 1.1.1. *Válvulas tipo Guillotina.*

Las compuertas deslizantes de tipo guillotina son utilizadas como elementos de control y aislamiento, en canales abiertos, fosos, estanques y tuberías. Éstas controlan flujo, mantienen niveles, lo estrangulan y lo distribuyen. Estas compuertas se utilizan para desagües en aguas negras y para hacer el desasolve del fondo de los tanques de almacenamiento o canales abiertos, ver Figura 2.

Entre sus ventajas están:

- Diseño soldado basado en el concepto modular, implica bajo costo y cualquier dimensión disponible.

- Sellos estandarizados que garantizan una permanente hermeticidad, baja abrasión y que son fácilmente intercambiables, lo que se traduce en una larga vida útil, confiabilidad y baja manutención.
- Se encuentran disponibles compuertas de cuatro elementos de aislamiento, también en diseño reforzado, para presiones de trabajo hasta 2 bar. Pero es posible contar con diseños para presiones mayores.
- Muchas maneras de operarlas.
- Diferentes tipos de sujeción.



Figura 2. Compuertas tipo guillotina de desplazamiento vertical

La sujeción de éstas generalmente se hace en las paredes finales de los recipientes y son montadas dentro del armado estructural del hormigón de concreto.

Las válvulas de tipo guillotina funcionan en conjunto con vertederos. Para poder regular la velocidad de descarga generalmente los verteros son de forma “V”, pero también se pueden encontrar de forma cuadrada o sección rectangular.

Para todos estos casos se exponen sistemas de canal abierto.

Entre sus características están:

- Retener fluido, más allá del límite de la compuerta, debido a que está basado en la altura de la misma, siendo también su carrera de apertura, esto para cuando su desplazamiento es vertical.
- Retener el paso de materia suspendida.
- Pueden bloquearse por residuos en el flujo, pero con un buen diseño en el vertedero, se puede evitar este problema.
- Su acción puede ser por movimiento lineal (pistón) o por mecanismo rotacional, permitiendo este segundo, hacer un control muy preciso y de mucha potencia.
- En caso de alta velocidad en el caudal, puede ser controlado el mismo, ya que su acción no es en sentido del movimiento.

Para el caso de sistemas cerrados (o entubados) existen el mismo tipo de válvulas, pero es necesario que tengan una brida en cada extremo de la válvula, ya que la presión ejercida por el fluido puede ser tal, que puede romper las uniones convencionales.

Solamente se mostrarán como nota, ya que el sistema propuesto para todo el desarrollo es de canal abierto.

### *1.1.2. Válvulas de liberación de caudal*

Se utilizan para reducir la cantidad de caudal, pero basado en su velocidad. Cuentan con una esclusa para almacenar aquella cantidad de fluido que se va a tirar por el canal de dicho propósito.

Esta válvula se instala en dirección contraria al caudal, a manera que la velocidad del mismo sea la que active la parte primaria de la válvula; funciona en conjunto con una compuerta de guillotina.

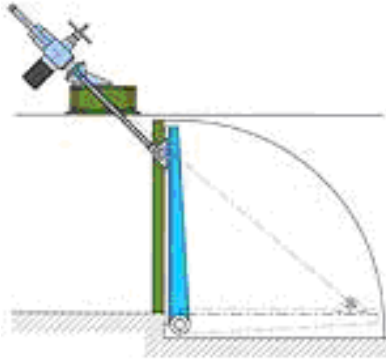


Figura 3. Válvula de liberación de caudal

### 1.1.3. Válvula mixta de guillotina

Esta válvula se utiliza para controlar el flujo de dos canales en un solo vertedero, funciona en conjunto con retículas a cada lado del canal y en base a la apertura de la cortina. Se puede regular el flujo de ambos canales a manera que ellos sean tributarios de tercer canal.

Entre sus características principales están:

- Contar con un control preciso de gran potencia.
- Evitar el paso de materia suspendida y residual, debido a las perforaciones en forma de retícula a lo largo de la carrera de la cortina.
- Conjuntar a dos canales como tributarios a un vertedero.
- Pueden ser reversibles en el sentido de su caudal.

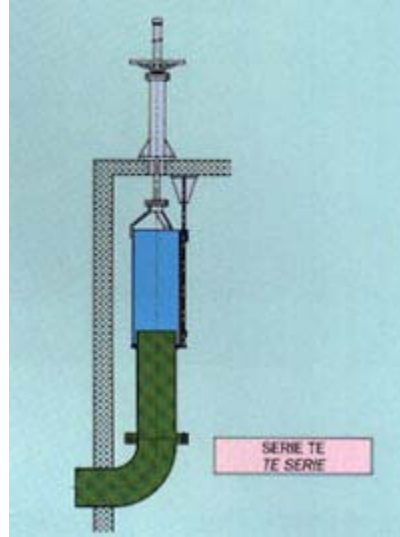


Figura 4. Válvula mixta de guillotina

#### 1.1.4. Válvula para control de avenidas

En este caso particular se pretende utilizar una fuente de gasto constante y adaptarla a los requerimientos del modelo, para ello se utiliza una compuerta repartidora, la cual divide el flujo en dos secciones, donde una parte de éste alimenta al modelo y la otra se desecha o se almacena para utilizarse posteriormente. Dependiendo de la posición de la compuerta repartidora, se puede regular la cantidad de agua o flujo que es alimentado hacia el modelo.

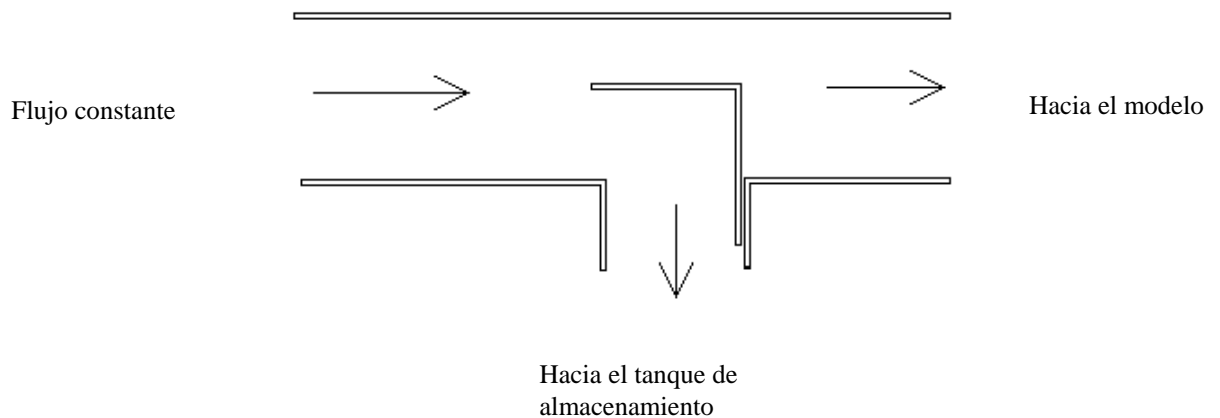


Figura 5. Diagrama de una compuerta en un generador de avenida.

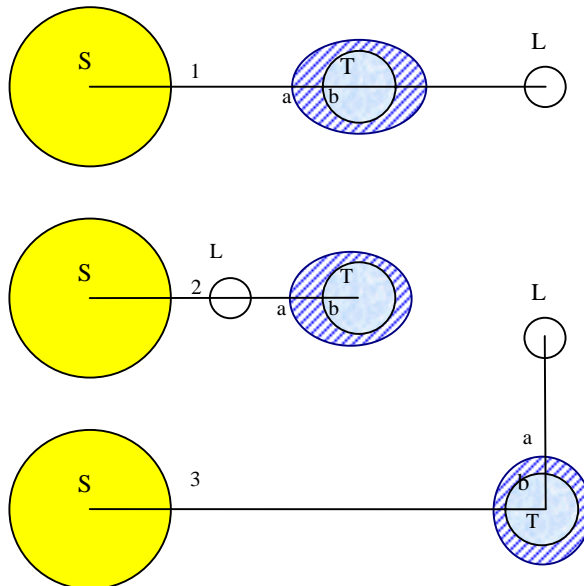


## 1.2. Generadores de Mareas

La marea es un movimiento periódico y alternativo de ascenso y descenso de las aguas del mar, producido por las acciones atractivas del Sol y de la Luna. El fenómeno de las mareas tuvo su explicación al aplicar la teoría de gravitación universal a la acción de la Luna y el Sol sobre las masas de agua de los océanos.

El movimiento en sentido vertical es el denominado propiamente *Marea*, mientras que el desplazamiento horizontal recibe el nombre de *Corriente de Marea*. Los factores esenciales de las mareas son las posiciones relativas de la Luna, la Tierra y el Sol; en el novilunio y el plenilunio, las fuerzas de atracción del Sol y la Luna se suman y originan las mareas *vivas*, y se contrarrestan parcialmente cuando la Luna está en cuadratura; Por la forma elíptica de la órbita lunar, la atracción que ejerce la Luna en el perigeo es 20% mayor que en el apogeo, y si aquella posición coincide con Luna nueva o con plenilunio, ocurren las mareas de máxima intensidad.

Las fechas de estas mareas máximas se retrasan en más de un mes de año en año, a causa del movimiento del perigeo lunar. Debido a la configuración de los fondos marinos y de las costas, las mareas se presentan con retraso con respecto al paso de la Luna por el meridiano del lugar. Este fenómeno se representa en la figura 6.



**Figura 6. Posición del Sol (S), la Luna (L) y la Tierra (T) en las mareas. 1 y 2, posición de los astros en siciquia; 3, posición de éstos en cuadratura; a b, elevación máxima de las aguas.**

Este retraso recibe el nombre de “Establecimiento de puerto” y conocido este dato y la fase de la Luna se puede calcular el momento de la pleamar y el de la bajamar. Como consecuencia del movimiento de la Luna, estos momentos se retrasan unos 50 minutos diarios.

Las mareas de mayor altura llegan a tener hasta 20 metros. Las observaciones de la variación del nivel del mar se efectúan sobre una escala vertical graduada, llamada *escala de marea*, o por medio de los mareógrafos; Los valores medios de estas observaciones se consignan en las *tablas de mareas*, que permiten prever y establecer la periodicidad del fenómeno y señalar su intensidad.

Los generadores de mareas se utilizan en los modelos hidráulicos a escala reducida para generar variaciones en los niveles de los líquidos, reproduciendo

aquellas variaciones de nivel en los cuerpos de agua debidas al efecto de las mareas. En los modelos hidráulicos a escala reducida, existen diferentes formas o sistemas para simular el efecto producido por las mareas, entre los que se encuentran el sistema de bombeo y compuerta y el sistema de bombeo y válvula de doble flujo, los cuales se explican a continuación.

### *1.2.1. Sistema de bombeo y compuerta*

Este sistema se muestra en la Figura 7. Sistema de bombeo y compuerta, está conformado por el tanque principal del modelo (a), un tanque auxiliar de almacenamiento (b), un sistema de bombeo (c) y una compuerta con control de descarga (d). El sistema de bombeo tiene gasto constante, por lo que aporta un volumen de líquido constante desde el tanque de almacenamiento (el cual debe tener una capacidad suficiente para satisfacer el volumen de agua para llenar el modelo). El tanque del modelo es aquel en el cual es necesario simular el efecto de las mareas (a).

El control de la marea se lleva a cabo por medio de una compuerta, cuya apertura es controlada por un sistema programador de mareas (e). La posición o altura de la compuerta, en este caso, determina el nivel de líquido en el tanque principal, y las variaciones en la posición de la misma simularán las variaciones de nivel que producen las mareas.

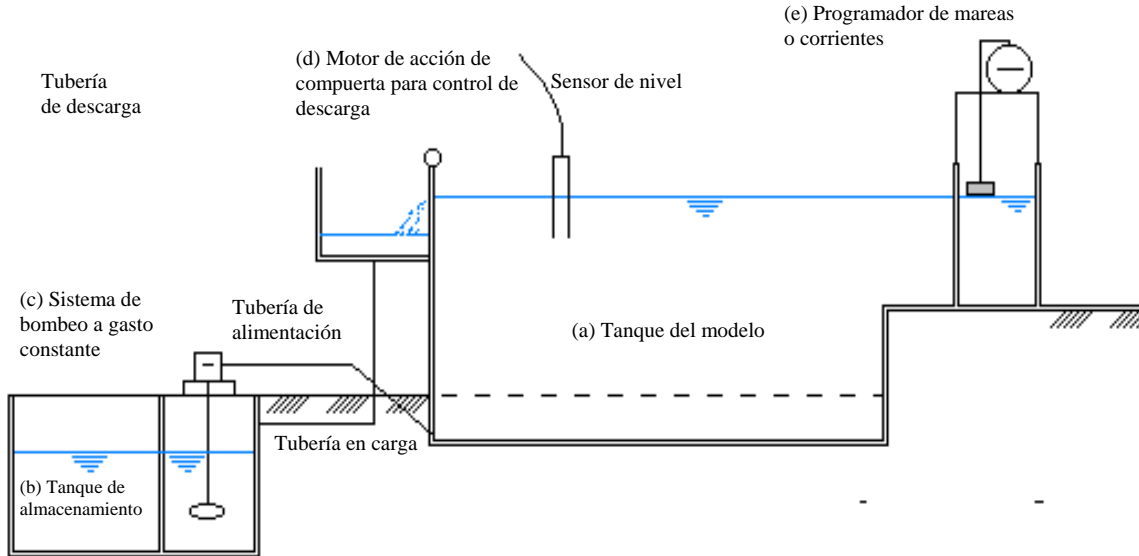


Figura 7. Sistema de bombeo y compuerta

### 1.2.2. Sistema de bombeo y válvula de doble flujo.

En la Figura 8 se muestra que en este sistema se bombea agua desde el tanque de almacenamiento hacia el tanque principal (a) o se retira agua del tanque principal hacia el tanque de almacenamiento (b), por medio de una bomba (c) y una válvula de control de gasto (d).

En este caso la bomba proporciona un flujo a gasto constante, donde el gasto es controlado por una válvula de control de flujo, que es regulada por un sistema de control de mareas (e).

La válvula controla el flujo que llega al tanque principal del modelo, haciendo que parte del gasto que aporta el sistema de bombeo vaya al tanque principal y la otra parte de este gasto se vaya por una tubería de descarga hacia el tanque de almacenamiento, o en un determinado momento, hacer que todo el gasto del sistema de bombeo se vaya por la tubería de descarga y al mismo tiempo se retire agua del tanque principal.

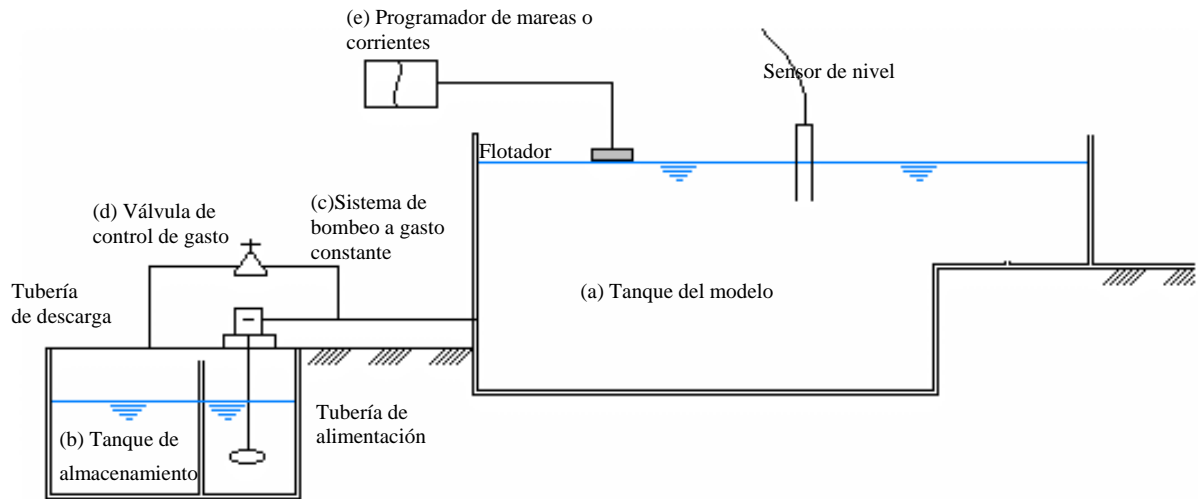


Figura 8. Sistema de bombeo y válvula de doble flujo.

En el caso particular del modelo fluvial se puede utilizar el sistema de bombeo y compuerta, utilizando este método, sólo será necesario controlar la compuerta, de tal forma que la apertura de la misma corresponda con el nivel que debe seguir el comportamiento de la marea.

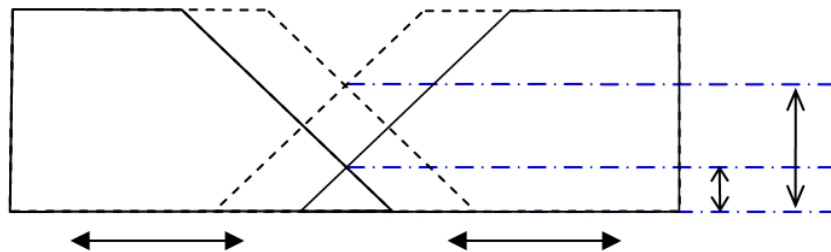


Figura 9. Diagrama de compuerta de marea.

Las compuertas tendrán un desplazamiento horizontal por lo que la altura del vertedero cambiará de acuerdo a la posición de las mismas. Cuando las

compuertas se cierran, la altura del vertedero subirá hasta la altura máxima, que es igual a la altura de las compuertas. Cuando las compuertas se abren, la altura del vertedero bajará hasta la altura mínima en la base de las compuertas.

La forma en que este sistema seguirá la programación de la marea de acuerdo a los datos recabados por la Comisión Nacional del Agua, se discutirá más adelante en el Capítulo 3 “Programa de Control”.

### **1.3. Control en válvulas**

Para realizar el control sobre las válvulas seleccionadas (de tipo guillotina, para la avenida y marea, respectivamente) utilizaremos un motor para cada una de ellas.

El problema de este tipo de controles es que se requiere de mucha fuerza y que los motores giran muy rápido; por ello se utilizarán cajas reductoras, que reducen la alta velocidad de los motores a una velocidad utilizable por las máquinas y aumentan la potencia de par a la salida.

Para potencias bajas se utilizan moto-reductores, que son equipos formados por un motor eléctrico y un conjunto reductor integrado. Las herramientas manuales por ejemplo los taladros poseen un moto-reductor.

Para potencias mayores se utilizan equipos reductores separados del motor. Los reductores consisten en pares de engranes con gran diferencia de diámetros, de esta forma el engrane de menor diámetro debe dar muchas vueltas para que el de diámetro mayor de una vuelta, de esta forma se reduce la velocidad de giro (velocidad angular). Para obtener grandes reducciones se repite este proceso, colocando varios pares de engranes conectados uno a continuación del otro. Las figuras muestran dos cajas de reductores con engranes cilíndricos y cónicos. Una de ellas tiene dos pares de engranajes cilíndricos de diente

helicoidal y la otra posee además un par de engranajes cónicos de diente helicoidal.

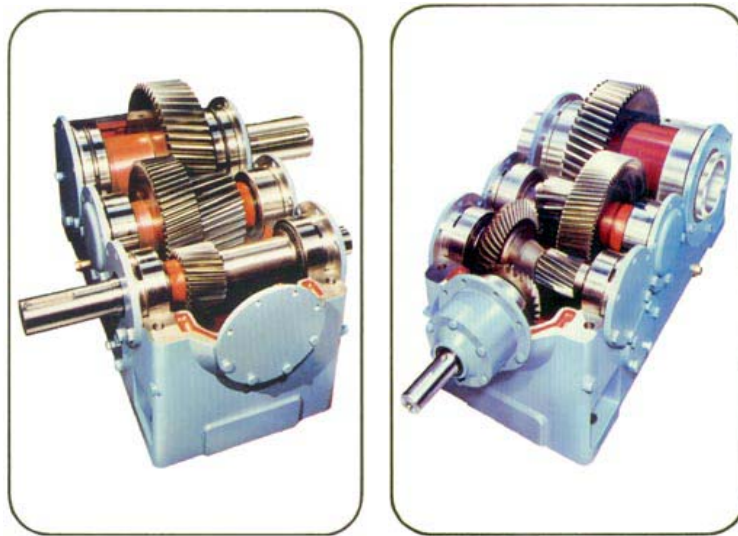


Figura 10. Reductores de velocidad

En estas cajas es importante notar que se abren en dos mitades y la línea de unión está en el plano que forman los ejes. Este diseño se basa en la conveniencia de abrir la caja al nivel de los ejes para extraerlos con facilidad y permitir el cambio de rodamientos, sellos de aceite, revisar el desgaste de los dientes y otras acciones preventivas. La figura 11 muestra una caja con engranes tipo tornillo sinfín y rueda helicoidal.



Figura 11. Reductor de velocidad con engranes de tornillo sinfín y rueda helicoidal

Como ya se dijo, este mecanismo es muy conveniente como reductor de velocidad en un solo paso. El tornillo o gusano se ubica en la parte inferior de la caja para asegurar una lubricación abundante. El husillo se acoplará a la flecha de salida, y con esto se tendrá el movimiento de la cortina de la válvula.

1.3.1. Diagrama de control de motores con circuito tipo H.

Este tipo de circuitos únicamente funciona para motores de corriente directa. También es conocido como “Puente completo”, recibe su nombre porque 4 interruptores se localizan en las esquinas de la H, y el motor hace la barra del centro. Una figura básica de este sistema se muestra en la Figura 12.

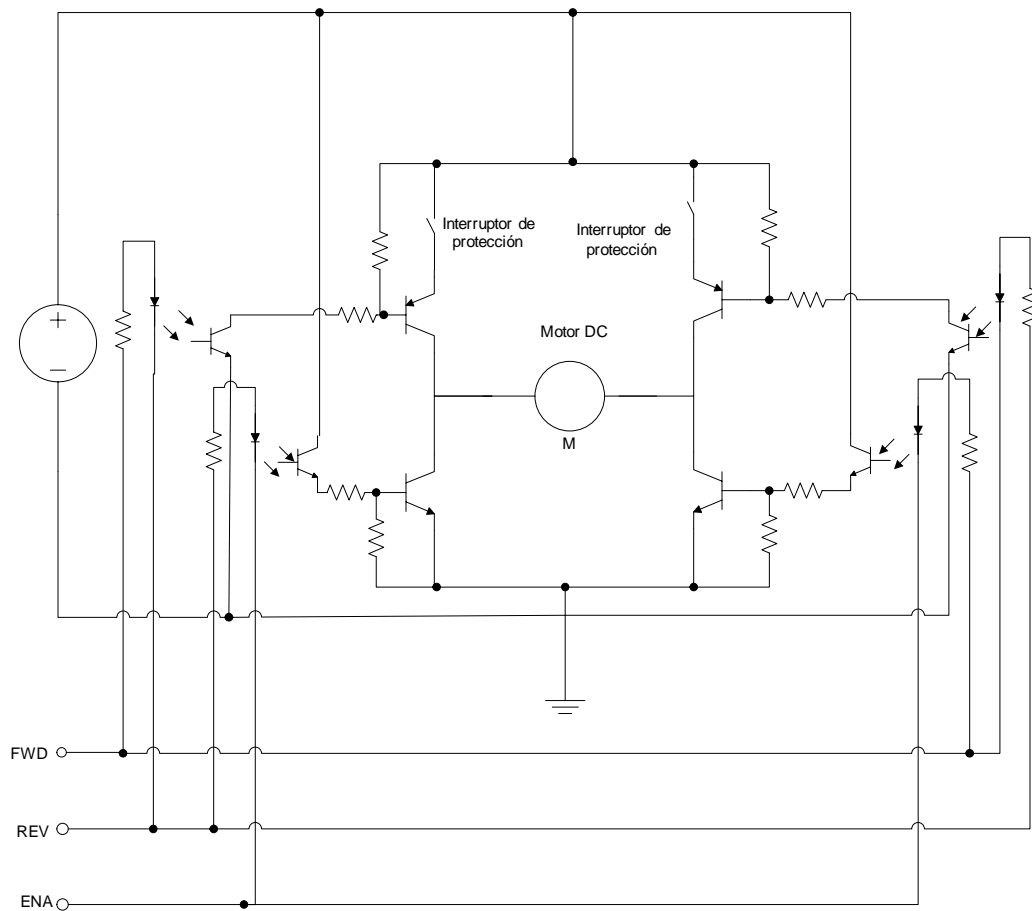


Figura 12. Diagrama de Puente tipo H



El modo de operación de este circuito es por pares opuestos, es decir, cuando se activa un par superior derecho, funciona al mismo tiempo con el inferior izquierdo, permitiendo el flujo de corriente en el motor en un sentido.

Para hacer que el motor haga un giro inverso, se activan los otros pares de conmutadores.

Se escogieron como elemento conmutador, en el extremo positivo de la fuente de voltaje, transistores de potencia, ya que el microcontrolador HC11, entregará una señal del tipo “Modulada por ancho de pulso” para controlar la velocidad del motor, y este elemento nos permite tener un control preciso del motor con dicha alimentación. La lógica de control del circuito tipo “H” obedece a la tabla 1.

FWR	REV	ENA	DESCRIPCION
1	0	0	Activa la esquina superior izquierda e inferior derecha, girando el motor a la derecha.
1	0	1	Activa la esquina superior izquierda, pero no gira el motor.
1	1	0	Activa las esquinas inferiores, derecha e izquierda, corta circuito al motor, haciendo un freno.
1	1	1	Desactiva las esquinas inferiores, entonces la señal PWM modula el frenado del motor.
0	1	0	Activa la esquina superior derecha e inferior izquierda, girando el motor a la izquierda
0	1	1	Desactiva la esquina inferior izquierda, pero el motor no gira.
0	0	0	Apaga toda alimentación, el motor no gira.
0	0	1	Apaga todas las esquinas, el motor no gira.

Tabla 1. Entradas del circuito puente tipo H

Como medida preventiva y evitar que los motores, las compuertas o el sistema de impulsión se pudiesen dañar, se instalan sensores de límite para el inicio y fin de carrera de las compuertas. Dicha protección estará dada en dos formas mediante sensores ópticos e interruptores de corriente.

La señal de los sensores ópticos será enviada al microcontrolador antes de que se activen los interruptores de corriente, para indicar al programa de control que la compuerta está a punto de llegar al final de la carrera y mandar la señal de paro.

Si por alguna razón el programa fallara en detener el movimiento de la compuerta, ésta activará los interruptores de corriente del circuito eléctrico tipo “H” para suspender el suministro de energía. El desarrollo de esta protección por medio del microcontrolador, se explicará detalladamente en el capítulo 3.

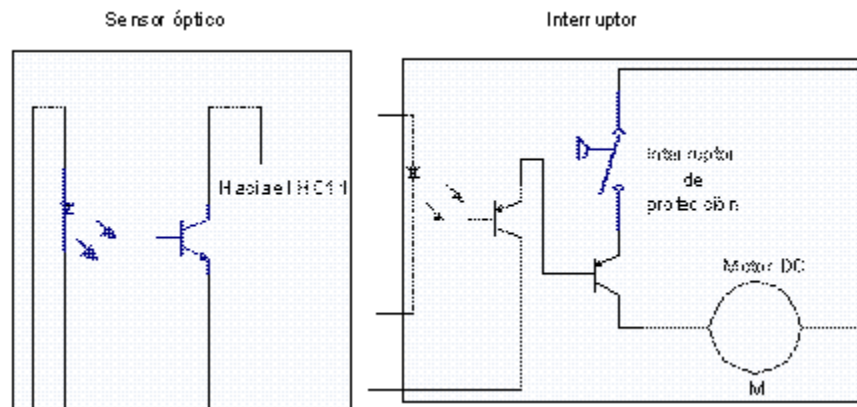


Figura 13. Sistemas de protección

El interruptor de fin de carrera se conecta en serie con la parte del circuito H que estará operando.

El interruptor de fin de carrera de apertura, se conecta en serie con el voltaje de apertura, haciendo que el motor pare en esta posición, pero quedando

en condiciones de operar en su cierre, y viceversa, se instalará un interruptor similar para el cierre de la compuerta.

Este interruptor es activado directamente por la puerta que lo empuja, no habrá control alguno de los elementos eléctricos o electrónicos del sistema sobre este elemento.

Es posible que estos dispositivos de control nunca se utilicen, ya que por software en el microcontrolador se limitará la carrera mínima y máxima de las válvulas, pero como son elementos de seguridad, se están considerando.

Otro elemento que es importante destacar es que los sistemas de actuación (motores y etapas de energía) son idénticos para todas las compuertas, salvo sus adecuaciones mínimas según su tarea. Esto tiene varias ventajas:

- Tener un mismo tipo de refacciones.
- Facilidad de pruebas de etapas entre dispositivos.
- Costos más bajos de implementación y mantenimiento.
- Facilidad de intercambio de piezas para pruebas en fallas.

## **Capítulo 2. Transductores (Sensores)**

## 2.1. Sensores de nivel

Se han desarrollado una gran variedad de sensores y tipos de transductores para determinar el nivel de líquidos y cuasi líquidos, por ejemplo polvos y sólidos granulados, en contenedores abiertos o cerrados (tanques, ductos, etc.). No sólo es el conocimiento del nivel por sí mismo lo importante, pues hay otras mediciones que pueden ser determinadas a partir del nivel. Si son conocidas la geometría y las dimensiones del tanque, el volumen del líquido puede ser determinado. Si adicionalmente es conocida la densidad del líquido puede calcularse su masa.

El nivel es sentido generalmente por uno de dos métodos: obteniendo una indicación discreta cuando ha sido alcanzado determinado nivel (punto de sentido) u obteniendo una representación análoga del nivel conforme éste va cambiando (sentido continuo) El sentido de puntos es también utilizado cuando sólo se desea conocer cuando existe un líquido o gas en cierto punto, por ejemplo en un tubo.

Los sistemas de sentido por puntos son usualmente más simples y baratos que los de sentido continuo y deben ser usados cuando se debe obtener solamente una indicación discreta. Aún cuando dos o más niveles discretos deban ser establecidos en el contenedor, el uso de dos o más puntos de sentido deben preferirse a un sistema de sentido continuo.

Los métodos de sentido de nivel por puntos son usualmente como indicadores de interfase entre un líquido y un gas, o algunas veces entre dos líquidos diferentes.

### *2.1.1. Sensores de nivel por transferencia de calor*

El sentido por transferencia de calor es usado por los de tipo resistivo (o termistor), el cual es calentado en cierta medida por el paso de corriente a través de él, de tal forma que su resistencia cambia debido a un enfriamiento cuando entra en contacto con el líquido.

### *2.1.2. Sensores de nivel de tipo óptico*

Los sensores ópticos se basan en la presencia o ausencia de reflexión de una haz de luz en la interfase entre la superficie de un prisma en contacto con un gas (reflexión), o un líquido (sin reflexión), o debido a una atenuación del haz de luz, cuando éste pasa a través de un líquido en su camino a un sensor de luz.

### *2.1.3. Sensores de nivel por amortiguamiento de oscilaciones mecánicas*

En el sensado por amortiguamiento de oscilaciones, la vibración mecánica de un elemento excitado eléctricamente es detenida (en un elemento piezoeléctrico o magnético) o reducida en amplitud. Debido al amortiguamiento acústico o viscoso, respectivamente, cuando el fluido medido cambia a un líquido.

### *2.1.4. Sensores continuos de nivel.*

El nivel, el volumen o la masa de un líquido en un tanque de geometría conocida pueden determinarse pesando continuamente el tanque, por medio de una celda de carga (transductor de fuerza), y restando la tara del peso del tanque.

El sensado por presión se basa en la presión ejercida en la base de una columna de líquido. La presión se incrementa con la altura de la columna y de acuerdo con el nivel superior al punto en el cual la presión es sensada. La diferencia de presión  $P_d$ , medida por el transductor de presión diferencial, es equivalente a la diferencia de presiones entre el fondo y la parte superior del tanque ( $P_I - P_h$ ). El nivel  $h$  del líquido superior al punto de medición inferior está dado por  $h = (P_I - P_h) / w$  donde  $w$  es el peso específico del líquido.

La presión actuando al fondo del tanque puede ser medida por un diafragma construido en el fondo del tanque, usado como elemento de sensado de presión.

### *2.1.5. Sensores capacitivos*

Un capacitor eléctrico consiste en dos conductores separados por un dieléctrico (sólido, líquido o gaseoso), o el vacío. La relación entre la carga,  $Q$ , y la diferencia de potencial,  $V$ , entre ellos viene descrita por su capacidad,  $C=Q/V$ . Esta capacidad depende de la disposición geométrica de los conductores y del material dieléctrico, dispuesto entre ellos  $C = C(\epsilon, G)$

Por ejemplo, para un condensador formado por  $n$  placas planas paralelas iguales, con área  $A$ , distancia  $d$  entre cada par, y un material entre ellas con constante dieléctrica relativa,  $\epsilon$ , la capacitancia aproximada es

$$C \approx \epsilon_0 \epsilon_r \frac{A}{d} (n - 1)$$

donde  $\epsilon_0 = 8,85 \text{ pF/m}$  es la constante dieléctrica del vacío.

Así pues, cualquier fenómeno o magnitud que produzca una variación de  $\epsilon_r$ ,  $A$  o  $d$ , provocará un cambio en la capacitancia  $C$  y, en principio, puede ser detectado mediante el dispositivo anterior. En general, cualquier cambio en el dieléctrico o en la geometría puede ser considerado para la detección del fenómeno que lo provoca.

En nuestro modelo, la única variación será el nivel del agua, esta variación que ocasionará una variación del área ( $A$ ), por lo que está perfectamente determinada la capacitancia máxima y mínima, todos los otros valores del capacitor se consideran como constantes. El valor de  $d$  de la ecuación anterior es la distancia del espesor del dieléctrico o la distancia que hay entre la barra y el agua, que en nuestro caso se optó por el neopreno que tiene una permitividad relativa  $\epsilon_r = 6.9$ .

Esto sólo es correcto para el agua completamente destilada libre de sales y minerales, ya que éstos son los que generan conductividad. Para el modelo expuesto el agua funcionará como conductor, no como dieléctrico, ya que como se considera el agua del río Pánuco, está llena de sales y minerales.

La sustitución de aire por agua como dieléctrico producirá un cambio apreciable, que se puede aplicar, por ejemplo, a la medida del nivel de agua en un depósito, o a la medida de humedad, si se dispone un dieléctrico que absorba y desabsorba agua sin histéresis.

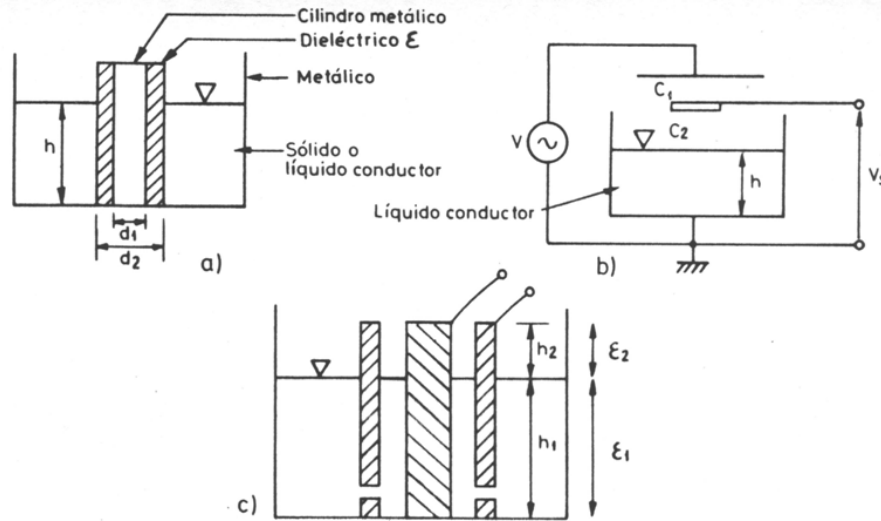
Otra de las aplicaciones muy frecuentes, además de las medidas de desplazamientos, es la medida de nivel de líquidos conductores y no conductores (aceite, gasolina). En la figura 14 (a) se presenta un sensor basado en una variación de área aplicable al caso de un líquido conductor. La expresión de la capacidad considerando que es un sistema de dos electrodos cilíndricos, es

$$C_h = \frac{2\pi\epsilon h}{\ln d_2/d_1}$$

El recipiente metálico debe conectarse a tierra para evitar el peligro de las descargas eléctricas y el efecto de las capacitancias parásitas.

Y las barras de sensor capacitivo, deberán estar fuera de contacto de las paredes del recipiente, esto es, sin contacto con las paredes.





**Figura 14. Configuraciones de sensores capacitivos para la medida del nivel de líquidos conductores y no conductores**

El sensor de la figura 14 (b) está basado en una variación de distancia, y es aplicable cuando la conductividad del líquido es muy alta (mercurio, agua, etc.), de modo que su superficie pueda actuar como «electrodo». Si es así, se forma un divisor de tensión capacitivo que da una tensión de salida.

$$V_s = V \frac{C_1}{C_1 + C_2}$$

Donde C1 es constante y C2 variable de forma inversamente proporcional a h. La salida es no lineal, pero se puede linealizar mediante un servosistema que desplace el electrodo de medida y el de referencia, de modo que su distancia al líquido sea constante, y midiendo cuánto hay que desplazar el electrodo de medida para que así suceda.

El sensor de nivel de la figura 14 (c) está basado en una variación de dieléctrico. Si los dos cilindros conductores se consideran concéntricos, la capacidad total será:

$$C \approx \frac{2\pi}{\ln(r_2/r_1)} (\varepsilon_1 h_1 + \varepsilon_2 h_2)$$

Y así, de no haber capacidades parásitas, la relación entre C y h1 sería lineal.

Por otro lado, hay que hablar en esta sección que el efecto de la tensión superficial, puede producir histéresis y por tanto, se debe compensar este efecto en el momento de hacer las mediciones.

#### *2.1.6. Sensores que se utilizarán en el proyecto y cómo se implementarán.*

Los sensores a utilizar serán del tipo capacitivo, ya que éstos cambiarán su valor a la menor perturbación o variación del fluido, lo que nos garantiza una respuesta inmediata. Si bien, es cierto que el sistema no requiere esta velocidad de respuesta, es mejor garantizar que el “cuello de botella” no esté desde la etapa de adquisición de datos.

Con la finalidad de obtener una medición lo más cercana al nivel real del líquido, se utilizará un sensor que permita una medición continua que sea capaz de detectar ligeras variaciones de la variable, a diferencia de sensores discretos de tipo encendido/apagado que sólo permiten la detección del líquido en ciertos puntos, previamente determinados, dentro del modelo.

Los sensores capacitivos formarán parte de un circuito oscilador que tendrá cierta frecuencia, y a medida que el valor de su capacitancia cambie, se producirá cambio en la frecuencia, procurando que estos cambios se encuentren dentro de la región lineal del circuito de generación de frecuencia.

Se calibrarán a un rango mínimo (cuando esté seco el río) y a su rango máximo (cuando el río esté a punto de desbordarse) y con ellos tendremos la frecuencia mínima y máxima que deberán captar el puerto de entrada del microcontrolador.

El rango de frecuencias se deberá adecuar a las características propias del microprocesador y a las rutinas del programa encargadas de medir el período de la señal. (Véase el capítulo 3)

Una vez determinadas estas frecuencias, tenemos un ancho de banda claramente definido y con ello poder hacer un circuito de amplificación de señales para poder transmitir las hasta el microcontrolador.

El sensor capacitivo tendrá la siguiente forma:

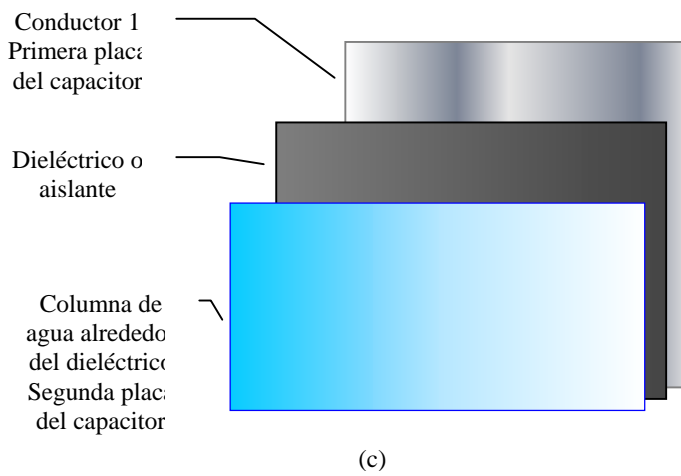
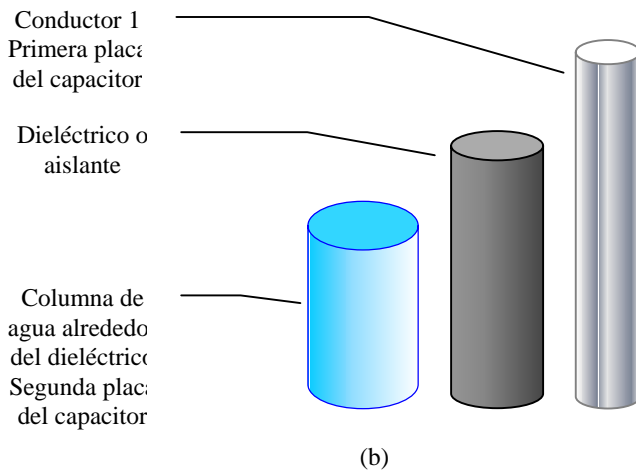
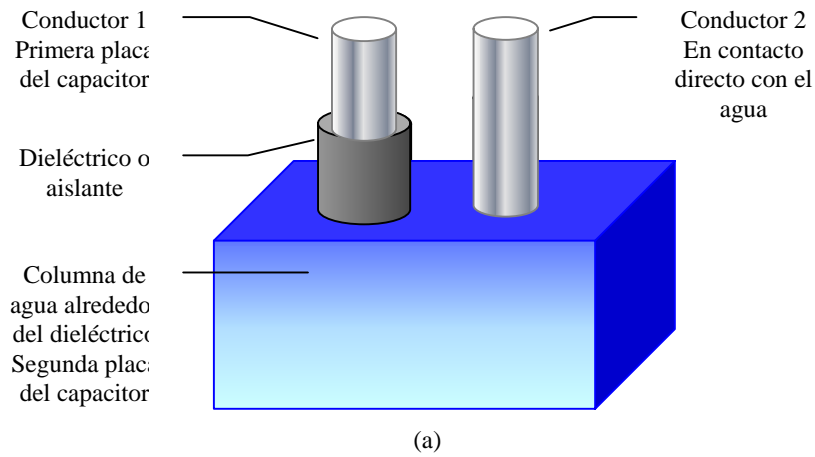


Figura 15. Sensor Capacitivo del Modelo

El circuito de generación de frecuencia, estará basado en el integrado LM555, debido a su gran estabilidad a la temperatura y precisión en la frecuencia de los trenes del pulso de onda cuadrada que genera.

El capacitor (sensor de nivel) será una parte del circuito del que depende la frecuencia del LM555, esto es, que si el capacitor varía su valor, la frecuencia del tren de pulsos varia, en forma lineal.

### 2.1.7. Circuito Astable

El circuito que generará los trenes de pulsos estará conectado como se muestra a continuación:

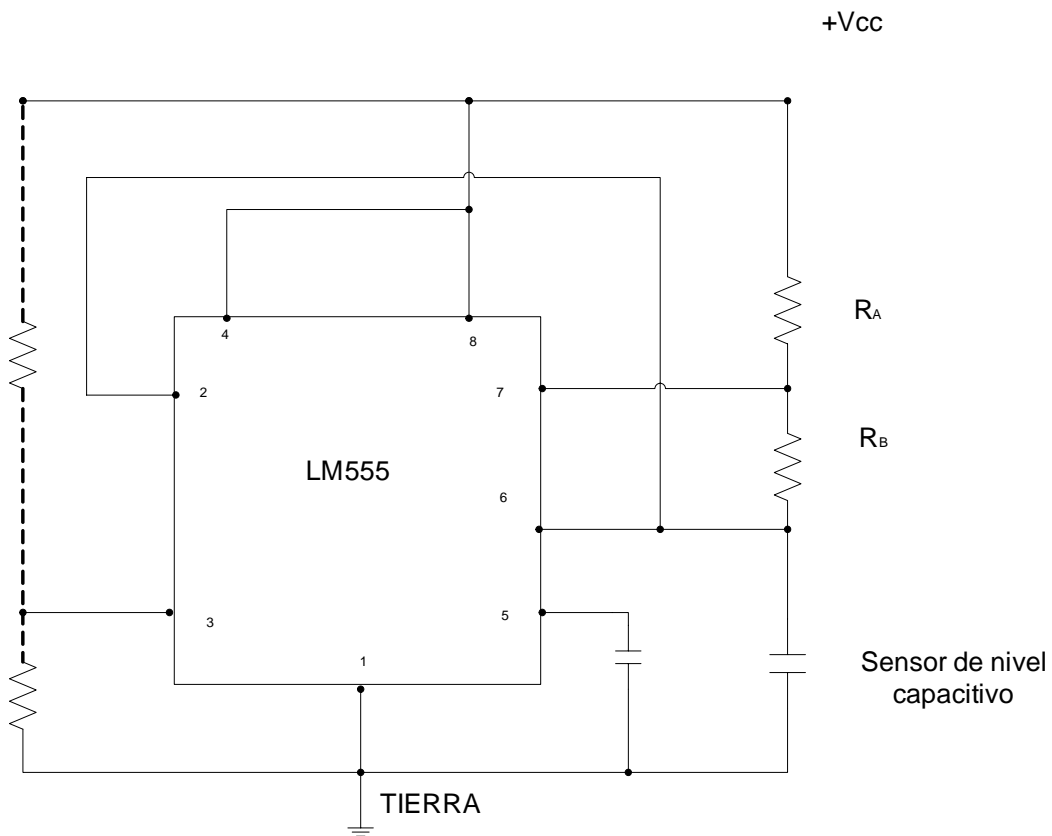


Figura 16. Circuito generador de pulsos

Donde la terminal 3 es la salida del multivibrador.

El capacitor exterior (en nuestro caso, el sensor de nivel), carga a través de RA + RB. Estas resistencias determinan el ciclo de trabajo del tren de pulsos.

De donde el periodo de oscilación del circuito anterior está determinado por la siguiente formula:

$$T = \frac{1}{f} = \frac{1.44}{(RA + 2RB)C}$$

Recordando que la frecuencia es: El número de periodos que ocurren en una unidad de tiempo. En un péndulo, al número de oscilaciones completas (ida y vuelta) que da el objeto en un segundo.

## 2.2. Sensores de flujo

El flujo es el movimiento de un fluido. El flujo es expresado ya sea en volumen de fluido por unidad de tiempo (flujo volumétrico) o como masa de fluido por unidad de tiempo (flujo másico).

La mayoría de los medidores de flujo “miran” el flujo volumétrico, el cual puede ser convertido a flujo másico, midiendo simultáneamente la densidad y calculando el flujo másico a partir de las dos mediciones. La medición de flujo puede clasificarse de la siguiente forma:

### 2.2.1. Por presión diferencial.

Secciones de tubo provistas con una restricción o una curvatura, que producen una diferencia de presión, proporcional al flujo a través de los dos puntos del dispositivo. La salida de un transductor de presión diferencial, cuyas entradas están conectadas a estos dos puntos, es representativa del flujo a través del elemento de sensado.

### *2.2.2. Por elementos mecánicos.*

Elementos de movimiento libre, por ejemplo turbinas o propelas o elementos retenidos mecánicamente, como un flotador en un tubo tapado, un tapón detenido por un resorte, una veleta con bisagra, cuyo desplazamiento, deflexión o velocidad angular es proporcional al flujo.

### *2.2.3. Por características del fluido.*

Algunos elementos transductores pueden diseñarse e instalarse para interactuar con el fluido en movimiento y producir una salida relativa al flujo. El cable caliente de un anemómetro de hilo caliente transfiere más de su calor al fluido conforme el flujo se incrementa, causando que la resistencia del cable se reduzca. Cuando pequeñas cantidades de materiales con radioisótopos se agregan al fluido, un detector de radiación cercano al fluido en movimiento responderá con un incremento en la salida conforme el flujo se incrementa.

En el medidor de flujo de tipo magnético, un incremento en la fuerza electromotriz es inducida en un fluido eléctricamente conductivo, fluyendo a través de un campo magnético transversal, cuando el flujo se incrementa.

En un medidor de flujo térmico, 2 sensores de temperatura se sumergen en el fluido, uno flujo arriba y otro flujo abajo y un calentador es sumergido en el mismo fluido y los dos sensores de temperatura se conectan en forma diferencial, la salida resultante se incrementará con el flujo másico.

### *2.2.4. Sensor de flujo de turbina*

El sensor de flujo de turbina es el transductor de flujo más ampliamente usado. Su principio de operación es similar al tacómetro de rotor dentado. Las hojas del rotor giran a una velocidad angular proporcional al flujo volumétrico. Debido a esta

característica nos entrega una señal en frecuencia directamente proporcional a la velocidad de rotación. La fricción rotacional es reducida tanto como sea posible mediante un diseño especial de rodamientos.

Cada aspa magnética del rotor corta el flujo magnético del polo de la bobina e induce un pulso en la bobina (bobina de sensado). Se utiliza un medidor de frecuencia para mostrar la salida de frecuencia del medidor de flujo, o puede utilizarse un convertidor de frecuencia a voltaje para obtener un voltaje de corriente directa que se incremente con el flujo. Las aspas del rotor pueden fabricarse de tal forma, que el voltaje de frecuencia variable a través de las terminales de la bobina sensora sea virtualmente senoidal. Esto permite el uso de un demodulador de F. M. como un convertidor de frecuencia a voltaje de corriente directa.

#### *2.2.5. Medidor de flujo por flujo oscilante*

En este dispositivo el flujo es forzado a un movimiento de torbellino, luego pasa a una cavidad tipo Venturi en un punto en el cual el flujo oscila alrededor del eje del medidor. Una respuesta rápida de temperatura o fuerza del transductor en ese punto, proporcionan una salida en términos de frecuencia o cambios de resistencia. Esta frecuencia, proporcional al flujo y convertida en variaciones de voltaje, puede ser mostrada en un contador después de ser aplicada.

#### *2.2.6. Otros medidores de flujo*

Otros diseños de medidores de flujo incluyen medidores de flujo ultrasónicos, que usan típicamente pares de transductores piezoeléctricos para establecer trayectorias sónicas. Los cambios en el flujo, producen variaciones en la propagación de la velocidad del sonido, a lo largo de la trayectoria.

#### *2.2.7. Sensores que se utilizarán en el proyecto y cómo se implementarán*



El sensor de flujo que se utilizará en el proyecto es una variación del sensor de tipo turbina. Basado en el mismo principio las hojas del rotor también girarán en relación directa al flujo volumétrico, sólo que en lugar de acoplarse el movimiento rotacional a un generador, se acopla a un dispositivo óptico capaz de producir impulsos eléctricos, cuya frecuencia sea directamente proporcional al flujo volumétrico.

Se utilizará un sensor óptico, el cual trae en el mismo encapsulado, el trasmisor del haz y el receptor (foto sensor); Al momento de encontrar una superficie que refleje la luz, ésta es detectada por su contraparte, enviando un pulso a la salida.

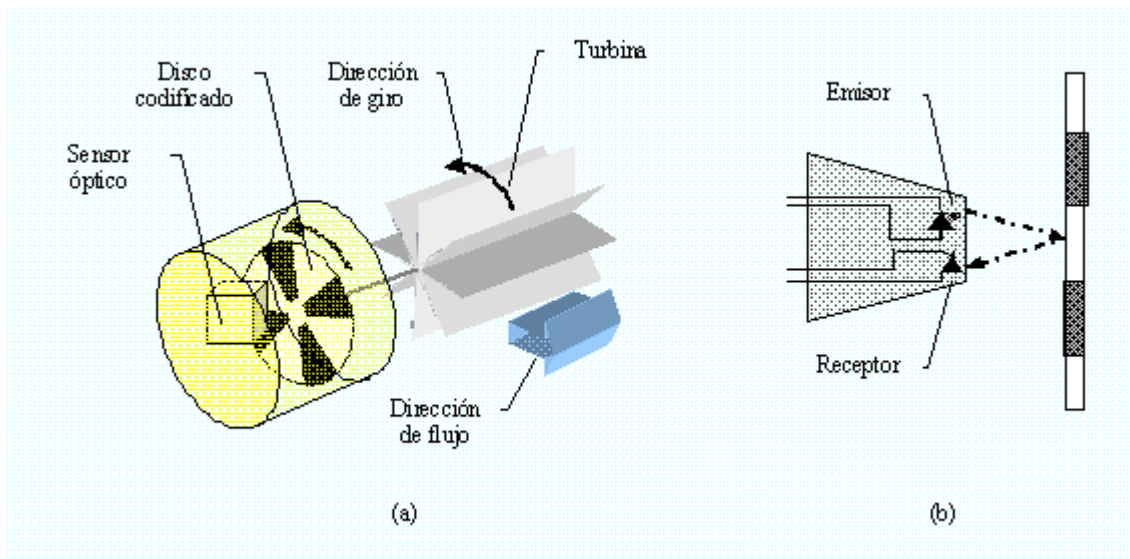


Figura 17. Sensor de flujo óptico de tipo turbina

En la Figura 17 (a) se observa el modelo del sensor de flujo de tipo turbina. En este sensor a la turbina que sensa el flujo, se le acoplará, en relación directa, una rueda que estará pintada en partes alternadas equidistantes, con pintura negra, que no pueda reflejar ningún tipo de luz y con pintura blanca, altamente reflejante. Dicha rueda se encapsulará en una caja opaca, para evitar que las condiciones externas, como luz solar o vientos entorpezcan su funcionamiento. La Figura 17 (b) muestra el dispositivo óptico mediante el cual se realiza el sensado de la velocidad angular de la turbina.

La rueda se conectará con la turbina por una flecha, que pasará por un retén.

## **2.3. Sensores de posición y desplazamiento.**

### *2.3.1. Transductores de desplazamiento capacitivos.*

En este tipo de diseño de dieléctrico móvil ambos electrodos del condensador tienen posiciones fijas. Un casquillo realizado de material aislante con una constante dieléctrica diferente a la del aire, se desliza dentro y fuera del conjunto de electrodos. Cuando el casquillo es extraído del conjunto de electrodos, aumenta la superficie del electrodo que va al aire como dieléctrico, y disminuye la superficie del electrodo que va al material del casquillo como dieléctrico. El resultado global consiste en un cambio de capacidad proporcional al movimiento axial del casquillo.

### *2.3.2. Transductores de desplazamiento inductivos.*

Los transductores que convierten el desplazamiento, usualmente desplazamiento lineal, en un cambio de autoinductancia de un simple devanado.

Por lo general se emplea un núcleo permeable magnético deslizante, que se mueve dentro de un devanado o bobina. El eje sensor determina el movimiento del núcleo. Cuando el núcleo se mueve el devanado cambia su auto inductancia. El devanado puede conectarse a un oscilador LC, de modo que el cambio de inductancia implica un cambio de frecuencia en la salida.

### *2.3.3. Transductores de desplazamiento por reluctancia.*

Son aquellos transductores que convierten un desplazamiento en variaciones de tensión de corriente alterna, mediante el cambio del camino de reluctancia entre dos o más devanados. El más utilizado es el transformador diferencial que se utiliza

tanto para desplazamientos lineales como angulares. Lo mismo ocurre para el tipo de puente de inductancias.

#### *2.3.4. Transductores de desplazamiento potenciométricos.*

En este tipo de dispositivos un contacto deslizante (escobilla) se mueve sobre un elemento resistivo. El contacto está ligado directamente a través de un mecanismo al eje sensor. La palanca está eléctricamente aislada del eje sensor. Existen transductores potenciométricos tanto de desplazamiento lineal como angular. La salida de estos sensores es una fracción de tensión, proporcional al desplazamiento, de la excitación.

#### *2.3.5. Transductores de desplazamiento por hilo vibrante.*

Este principio se utiliza en los transductores de desplazamientos muy pequeños. Un cambio de desplazamiento origina un cambio de tensión en un hilo delgado conectado a un circuito oscilante realimentado, variando también como consecuencia la frecuencia de oscilación.

#### *2.3.6. Transductores de desplazamiento electro ópticos.*

En este tipo de sensores se utilizan pares de fuente/sensor de luz para detectar la posición y el desplazamiento de objetos. Este tipo de sensores tienen la ventaja de ser sensores sin contacto y pueden utilizarse para medir desplazamientos pequeños, ya que la salida del sensor decrece exponencialmente con la distancia del objeto.

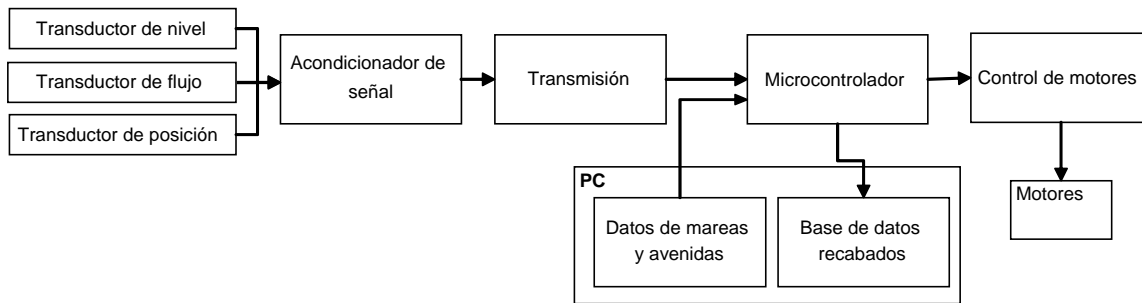
#### *2.3.7. Sensores que se utilizarán en el proyecto y cómo se implementarán.*

Los sensores que se utilizarán para el modelo son los sensores potenciométricos, utilizando uno para cada una de las compuertas. La salida de estos

sensores será una tensión entre 0 y 5 volts. Una salida de 0 volt denotará que la compuerta está totalmente cerrada y un valor de 5 volts indica una compuerta totalmente abierta. Estas señales analógicas deberán ser acondicionadas y transmitidas al microcontrolador, donde serán convertidas a valores digitales para ser procesadas y producir las salidas esperadas.

**Capítulo 3. Sistema de Control.**

En este capítulo se describirá el sistema de control del modelo para el generador de mareas y de avenidas, así como la forma en que se obtendrán los datos provenientes de los diferentes sensores. En la Figura 18. Diagrama de bloques del sistema de control se muestra el diagrama de bloques del sistema de control utilizado en el modelo.



**Figura 18. Diagrama de bloques del sistema de control**

El objeto de cualquier sistema de obtención y procesamiento de datos es coleccionar los datos, procesarlos en la forma deseada y registrar los resultados de manera adecuada para almacenamiento, presentación o proceso adicional como un proceso de control.

En el caso del modelo hidráulico a escala reducida, las diferentes variables de entrada del sistema de control son convertidas por los diferentes transductores (nivel, flujo y posición) en señales eléctricas. Estas señales provenientes de los transductores se acondicionan para ser transmitidas y/o para conectarse al microcontrolador. El microcontrolador recibe la información de los transductores de flujo, nivel y posición, la procesa, almacena y transmite los resultados a una computadora o estación de trabajo, por medio de un sistema de comunicaciones en serie. La información de nivel y de flujo es almacenada en bases de datos diseñadas para este fin. Al mismo tiempo una computadora personal contiene las bases de datos con información estadística de las mareas y avenidas, que serán reproducidas en el

modelo y envía al microcontrolador las señales adecuadas que servirán como directrices de control.

El microcontrolador las recibe y compara con la información de posición de las compuertas, que resultan de los datos obtenidos de los transductores de posición. El microcontrolador por medio del programa de control, basado en lógica difusa, produce las señales de salida que son enviadas a los circuitos de control de motores. Los motores accionan las compuertas del modelo, para producir las variaciones de marea y de las avenidas.

### **3.1. Transductores**

El elemento esencial de un sistema moderno de obtención de datos es el instrumento transductor, el cual proporciona una señal eléctrica que indica la variable física que se está midiendo. La señal puede ser un voltaje, resistencia o corriente analógicos, o una representación digital de cualquiera de estas cantidades, en forma de una serie de pulsos eléctricos, o frecuencias.

En este caso la información proveniente del modelo hidráulico, está dada por los transductores de nivel, de flujo y de posición, como se describió en el capítulo anterior. Los transductores de flujo y de nivel descritos proporcionan, en ambos casos, una señal cuya frecuencia varía de acuerdo a los cambios de la variable física que está siendo medida. Los transductores de posición se basan en elementos resistivos, que al aplicarles una tensión eléctrica producen voltajes determinados, dependiendo de la posición de las compuertas.

### **3.2. Acondicionamiento de señal.**

Por medio del acondicionamiento de la señal, se pretende eliminar ruido y amplificar la señal para ser transmitida, o prepararla para la siguiente etapa.

Para la eliminación del ruido eléctrico producido por las diferentes variables físicas, se utiliza el filtrado eléctrico de las señales. Existen diferentes circuitos que sólo permiten el paso de cierto rango o parte de las frecuencias que tienen a su entrada. Algunos de ellos están compuestos solamente por elementos pasivos (resistores, capacitores e inductores), en tanto que otros contienen amplificadores. Los circuitos del primer tipo se llaman filtros pasivos, en tanto que los segundos se denominan filtros activos.

Una vez que se filtra la señal, es necesario considerar los medios disponibles, a fin de transmitirla si el transductor se encuentra a una distancia considerable con respecto del microprocesador. Debe mencionarse, que en algunas situaciones conviene filtrar la señal antes y después de la transmisión.

### **3.3. Transmisión de datos.**

Cuando se tiene un transductor en una ubicación y es necesario trasladar la salida del transductor a otro lugar, probablemente donde se va a procesar la información de dicho transductor, se debe elegir el canal por medio del cual ocurrirá la transmisión necesaria. Hay muchas elecciones posibles, que varían desde corriente y voltaje en alambres de cobre, hasta la propagación de ondas electromagnéticas a través del espacio, o la transmisión de señales de luz mediante fibras ópticas. Al seleccionar un canal de comunicación, se deben considerar dos puntos vitales, que son: costo e inmunidad al ruido.

En este caso, aunque se trata de un sistema a escala reducida, el modelo hidráulico puede ser lo suficientemente extenso, como para pensar en la elección de un canal de transmisión adecuado desde los sensores del modelo al dispositivo de adquisición de datos, en este caso el microprocesador.

El cable coaxial proporciona un medio muy seguro y económico, para transmitir una señal de un lugar a otro. Consiste en un conductor concéntrico mediante el cual



se envían las señales de interés, rodeado por otro conductor que se mantiene a un voltaje cero (tierra). Este blindaje exterior atenúa cualquier ruido disperso electromagnético, antes de que pueda alterar la señal presente en el conductor interior.

### 3.4. Microcontroladores

Un microcontrolador es definido como un sistema computacional completo, incluyendo una unidad central de proceso (CPU), memoria, oscilador de reloj, entradas y salidas en una sola pastilla de circuito integrado. El CPU es el centro de todos los sistemas computacionales, su función principal es ejecutar las instrucciones del programa que fue proporcionado por un programador. El programa computacional instruye al CPU a leer información desde sus entradas, tomar decisiones simples, y escribir información en la memoria, así como en las terminales de salida.

Existen en el mercado un gran número de microcontroladores cuyas especificaciones varían de acuerdo al modelo y al fabricante. En nuestro caso particular ocuparemos un microcontrolador de Motorola de la familia de controladores HC11.

#### 3.4.1. Microcontrolador MC68HC11

Los microcontroladores HCMOS (high-density complementary metal-oxide semiconductor) MC68HC11 son una familia de microcontroladores (MCU's) de 8 bits con capacidades periféricas altamente sofisticadas. Su velocidad nominal de bus es de 2 MHz. Además tiene un bajo consumo de energía.

Contiene un convertidor análogo a digital de 8 bits de resolución con 8 canales. Una interfaz de comunicaciones asíncrono y una interfaz serial síncrona periférica independiente. El sistema temporizador de 16 bit tiene tres líneas de captura de entradas, cinco líneas de salida de comparación y una función de interrupción en

tiempo real. Un subsistema de acumulador de pulsos de 8 bits puede contar eventos externos o medir periodos externos.

El microcontrolador incluye un circuito de auto-monitoreo para proteger contra errores del sistema. El sistema COP (computer operating properly) protege contra errores de software. Un sistema de monitoreo de reloj genera un reset de sistema en caso que el reloj se pierda o trabaje demasiado lento. Un sistema de detección de código ilegal provee una interrupción no mascarable, si se detecta un código ilegal. Dos modos de ahorro de energía, "wait" y "stop", están disponibles para conservar energía adicional, lo cual hace a la familia del HC11 especialmente atractiva para aplicaciones alimentadas con baterías y en automovilismo.

Además cuenta con 38 pines de entrada/salida de propósito general, 16 de ellos son bidireccionales de entrada/salida, 11 son solamente de entrada y 11 solamente de salida.

El microcontrolador de nuestro sistema, como ya se mencionó, cumple con las siguientes funciones principales.

- 1.- Muestreo de las señales provenientes de los transductores de nivel, flujo y posición de las compuertas.
- 2.- Comunicación serial con la PC para envío y recepción de datos.
- 3.- Control difuso para producir las señales de salida (control de motores) en base a las señales de entrada y de las directrices enviadas por la PC.

### **3.5. Medición de Nivel y de Flujo**

Como se mencionó en el capítulo 2, los transductores de nivel producen una señal, cuya frecuencia es inversamente proporcional al nivel del líquido en el cual se sumergen.

Por tanto, conociendo la frecuencia proveniente de un transductor, se puede conocer el nivel de líquido en un punto en particular. Hay que tomar en cuenta que ya que los transductores de nivel se sitúan en diferentes puntos del modelo, un transductor determinado puede estar situado en una posición más alta o más baja con respecto a otro transductor.

De cualquier modo, los valores medidos estarán referenciados al punto en el que se hace la medición. Para obtener los valores absolutos de nivel, deberá entonces agregarse la posición relativa en la cual se ubique cada uno de los transductores.

Para hacer más simple la operación del microcontrolador y optimizar sus recursos, la obtención de los valores absolutos deberá realizarse por medio de la computadora personal.

En el caso de las señales provenientes de los sensores de flujo, la frecuencia será directamente proporcional al gasto másico en el punto de medición.

### *3.5.1. Sistema temporizador del MC68HC11*

Para conocer el período y la frecuencia de las señales se utilizará el sistema temporizador del microcontrolador. Este sistema está basado en un contador de 16-bits que corre libremente. Todas las actividades del sistema temporizador principal se encuentran referenciadas a este contador. Este contador es incrementado a partir de \$0000 hasta la cuenta máxima \$FFFF. En la cuenta máxima el contador regresa al valor \$0000 y activa una bandera de desbordamiento (“overflow”) y continúa incrementándose. Este contador no puede ser modificado interrumpido o inicializado.

**TCNT**- Timer Counter

\$100E	Bit 15	14	13	12	11	10	9	Bit 8	TCNT(High)
\$100F	Bit 7	6	5	4	3	2	1	Bit 0	TCNT(Low)

TCNT es \$0000 en el reset. En los modos normales, TCNT es de sólo lectura.

Un pre-escalador programable permite al usuario seleccionar una entre cuatro posibles tasas, para conducir el contador del temporizador principal. La tasa más rápida, que es seleccionada por omisión, causa que el contador se incremente a la tasa del pulso de reloj producido por el microcontrolador conocido como E (cuya frecuencia es la frecuencia del cristal dividida entre 4). Seleccionar la tasa más lenta causará que el contador se incremente a razón de E/16.

El valor del pre-escalador se selecciona a través de los bits seleccionadores del pre-escalador, PR1 y PR0, que se encuentran en el registro de máscara del temporizador 2 (TMSK2)

**TMSK2** - Timer Interrupt Mask Register 2

**\$1024**

	Bit7	6	5	4	3	2	1	Bit 0
	TOI	RTII	PAOVI	PAII	-	-	PR1	PR0
RESET :	0	0	0	0	0	0	0	0

PR[1:0] - Timer Prescaler Select

Estos bits son usados para seleccionar tasa de divisor del pre-escalador. En modos normales de operación, puede ser escrito sólo una vez, y la escritura debe ser dentro de los primeros 64 ciclos después del reset.

En la Tabla 2 se pueden observar los valores del pre-escalador según el estado de los bits PR[1:0] y los periodos para un conteo y el desbordamiento.

PR[1:0]	Factor Preescalador	Frecuencia del Cristal		
		2 <sup>23</sup> Hz	8 MHz	4 MHz
		Un Conteo (Resolución)/Desbordamiento (Rango)		
0 0	1	477 ns / 31.25 ms	500 ns / 32.77 ms	1 µs / 65.54 ms
0 1	4	1.91 µs / 125 ms	2 µs / 131.1 ms	4 µs / 262.1 ms
1 0	8	3.81 µs / 250 ms	4 µs / 262.1 ms	8 µs / 524.3 ms
1 1	16	7.63 µs / 0.5 s	8 µs / 524.3 ms	16 µs / 1.049 s
		2.1 MHz	2 MHz	1 MHz
Frecuencia de Reloj ( Reloj E)				

**Tabla 2. Pre-escalador**

### 3.5.2. Captura de entradas

El microcontrolador cuenta con la función de captura de entradas. Esta función es de gran importancia, ya que es la que nos va a permitir medir el periodo y por tanto, la frecuencia de las señales provenientes de los transductores de nivel y de flujo.

Los microcontroladores de la serie E de la familia del MC68HC11 cuentan con tres funciones de captura de entradas y una adicional, que puede ser configurada ya sea como la cuarta captura de entrada, o como un quinto comparador. (este concepto se verá con mayor detalle dentro de este capítulo)

Como se vio anteriormente, el tiempo del microcontrolador se representa a través de la cuenta que se lleva en el contador de 16 bits TCNT. La función de captura de entradas se utiliza para registrar el tiempo en el cual ocurre algún evento externo. Esto es posible almacenando el contenido del contador, cuando se detecta un flanco seleccionado en la terminal de entrada correspondiente. Este tiempo es guardado en el registro de captura de entradas.

Grabando los tiempos de flancos sucesivos de una señal de entrada se puede determinar, a través de software, el periodo de una señal, y por consiguiente también la frecuencia.

Cada función de captura de entrada incluye un registro de 16 bits, lógica de detección de flanco, y lógica de generación de interrupciones. El registro de 16 bits captura el valor actual del contador, cuando detecta un flanco seleccionado en la terminal de entrada correspondiente. La lógica de detección de flanco incluye pares de bits de control, de tal forma que puede seleccionarse qué tipo de flanco debe detectarse, el flanco de elevación, el flanco de caída, ambos flancos o ningún flanco. La lógica de generación de interrupciones incluye una bandera de estado, que indica cuando un flanco ha sido detectado, y un bit habilitador de interrupciones local, el cual determina cuando la correspondiente función de captura de entrada generará una solicitud de interrupción por hardware.

Los flancos detectados, generalmente, son asíncronos a la señal interna del contador del temporizador, que es incrementado por el reloj interno PH2. Por tanto, estas solicitudes de captura deben ser sincronizadas con PH2, de tal forma que la captura del contador, se lleve a cabo en la mitad del ciclo opuesta al incremento del contador. Gracias a esta característica, se evitan errores en la captura del contador. Sin embargo, este proceso de sincronización introduce un pequeño retraso, desde que ocurre el flanco de entrada, hasta que el valor del contador es almacenado. Este pequeño retraso puede ser ignorado, ya que cuando se mide el tiempo entre dos flancos sucesivos, ambos flancos son sometidos al mismo retraso, por lo que ambos retrasos se contrarrestan.

El elemento central de cada función de captura de entrada, es el registro de captura, el cual puede ser leído por software como un par de registros de 8 bits. Estos registros (TICx) no son afectados al momento del reset y no pueden ser escritos por medio de software.

**TIC1 - TIC3 - Timer Input Capture**

\$1010	Bit 15	14	13	12	11	10	9	Bit 8	TIC1 (High)
\$1011	Bit 7	6	5	4	3	2	1	Bit 0	TIC1 (Low)
\$1012	Bit 15	14	13	12	11	10	9	Bit 8	TIC2 (High)
\$1013	Bit 7	6	5	4	3	2	1	Bit 0	TIC2 (Low)
\$1014	Bit 15	14	13	12	11	10	9	Bit 8	TIC3 (High)
\$1015	Bit 7	6	5	4	3	2	1	Bit 0	TIC3 (Low)

TICx no es afectado por reset

**Timer Input Capture 4/Output Compare 5 Register**

Registro de Captura de entradas 4/Comparación de Salida 5. Utilice TI4/OC5 ya sea como un registro de captura de entradas o como registro de comparación de salida, dependiendo de la función seleccionada para el pin PA3. Para habilitarla como una terminal de captura de entradas, fije el bit I4/O5 el registro de control de acumulador de pulsos (PACTL) a un nivel cero lógico.

**TI/O5 - Timer Input Capture 4/Output Compare 5**

\$101E	Bit 15	14	13	12	11	10	9	Bit 8	TI4/O5 (High)
\$101F	Bit 7	6	5	4	3	2	1	Bit 0	TI4/O5 (Low)

El par de registros TI4/O5 es fijado en unos (\$FFFF) después del reset

Se puede programar la función de captura de entradas, para detectar la polaridad de un flanco de entrada en particular en la correspondiente terminal de entrada del temporizador, mediante pares de bits del Registro de Control del Temporizador 2 (TCTL2)

**TCTL2- Timer Control 2**

**\$100A**

	Bit7	6	5	4	3	2	1	Bit 0
	EDG4B	EDG4A	EDG1B	EDG1A	EDG2B	EDG2A	EDG3 B	EDG3A
RESET :	0	0	0	0	0	0	0	0

**EDGxB y EDGxA- Input Capture Edge Control**

Control del flanco de Captura de Entradas. Existen cuatro pares de estos bits. Cada par es puesto en cero por reset y debe ser codificado para configurar la entrada correspondiente al circuito detector de flanco de captura de entradas. IC4 funciona sólo si el bit I4/O5 bit en el registro PACTL es 1.

Las combinaciones posibles y sus configuraciones se encuentran descritas en la tabla 3.

EDGxB	EDGxA	Configuración
0	0	Captura Deshabilitada
0	1	Captura únicamente en flanco de elevación
1	0	Captura únicamente en flanco de caída
1	1	Captura en cualquier flanco

**Tabla 3. Control de flanco de captura de entrada**

### 3.5.3. Medición de Período y Frecuencia

Para medir el período de una señal de entrada al microcontrolador, como es el caso de la señal proveniente de los transductores de nivel y flujo, basta con detectar dos flancos de elevación consecutivos y registrar los tiempos en los que ocurrieron. La diferencia entre ambos tiempos detectados dará como resultado el período.

En el caso de los transductores de nivel de tipo capacitivo utilizados, el período es directamente proporcional al nivel de líquido. De tal forma que se puede caracterizar cada transductor y determinar una función lineal período-nivel, así como los límites inferior y superior de dicha función.

El período mínimo que puede ser medido depende de las instrucciones utilizadas para registrar la primera captura y preparar al sistema para la siguiente. Para lograr esto se deben contar los ciclos de reloj necesarios para llevar a cabo estas funciones. Este valor puede variar dependiendo de la instrucción que se está ejecutando y el ciclo de reloj de dicha instrucción, en la que ocurre la detección del flanco. Dependiendo de esto, se tendrá el mejor y el peor caso para la medición del período mínimo. También se debe considerar el fenómeno de latencia, ya que la captura de la entrada puede ocurrir cuando el microcontrolador ya está sirviendo una rutina de interrupción.

El período más largo que puede medirse de esta, forma será el tiempo que le lleva al contador llegar de un desbordamiento (overflow) a otro. Si el período a medir



es mayor, entonces deberá llevarse un conteo de los desbordamientos, para conocer el valor exacto del período.

Para garantizar un valor más confiable del período, se pueden registrar varios ciclos de la señal de entrada y se calculará un período promedio. Se programarán las capturas de entradas, de tal forma, que soliciten interrupciones una vez que detecten los flancos de elevación de la señal de entrada.

Por lo tanto, es necesario incluir en el programa un bloque de inicialización de las variables de las captura de entradas, de tal forma que se borren las banderas que pudieran existir, así como establecer las opciones para la detección de los flancos.

En este caso, la rutina de interrupción captura nueve flancos de elevación y determina el período promedio de los ocho períodos medidos. La Figura 19. Diagrama de flujo de la rutina de interrupción IC1 muestra el diagrama de flujo de la rutina, mediante la cual se obtiene el período promedio de la señal muestreada.

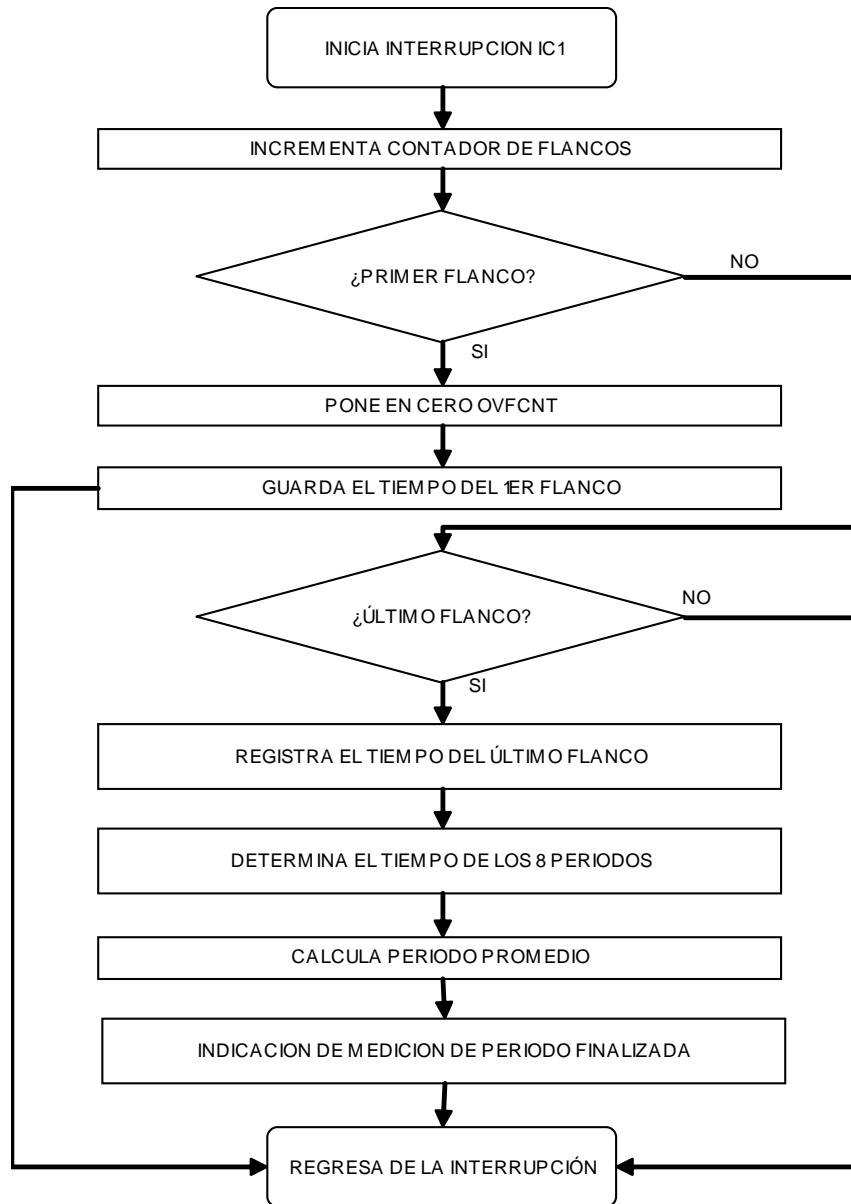


Figura 19. Diagrama de flujo de la rutina de interrupción IC1

En la Figura 20. Listado de la rutina de interrupción OC1 se muestra el listado de la rutina de servicio IC1, para la medición del periodo de la señal, el número de desbordamientos (overflow) se guardan en dos registros diferentes, que corresponden al primer y al último flanco respectivamente.

```

***
* RSIC1 - Rutina de servicio de OC1
*
* Cuando ocurre algún flanco de elevación en OC1
*
***
RSIC1      LDX      #REGINI      Apunta al inicio del bloque de
registros
      INC      IC1CNT      FF-IC1 apagado; 0-1er flanco; 8-último f
      BNE      FLNSIG      Si es 0, primer flanco

* Procesa el primer pulso
LDD      OVFCNT      Lee el contador de overflow
STD      MPOVC1      Guarda el contador de overflow
LDD      TIC1,X      Lee el tiempo del primer flanco
STD      TMP1        Guarda el tiempo del primer flanco
BMI      IC1SAL      Termina si IC ocurre antes de cualquier OVF
LDAA     TFLG2,X     Revisa TOF en el bit más significativo
BPL      IC1SAL      Si no hay overflow entonces termina
DEC      MPOVC1      Si hay overflow, éste no debe contar
BRA      IC1SAL      Termina de procesar el primer flanco

* Procesa los flancos siguientes
FLNSIG    BRCLR      IC1CNT #$08 ICISAL
  * Si no es el último entonces termina
LDD      OVFCNT      Lee el contador de overflow
STD      MPOVC2      Guarda el contador de overflow
LDD      TIC1,X      Obtiene el tiempo del último flanco
BMI      PERCAL      Si IC ocurre antes de TOF, no verifica TOF
TST      TFLG2,X     Revisa si existe desbordamiento
BPL      PERCAL      Si no existe TOF no incrementar
INC      MPOVC2      Si TOF ocurre antes de flanco, contar
PERCAL    SUBD      TEMP1Tiempo del último menos el del primero
STD      TEMP1        Actualiza el resultado
BCC      RES1OK      Revisa si existe préstamo
DEC      MPOVC2      Si existe préstamo descontar
LDD      MPOVC1      Contador de desbordamiento 1
SUBD     MPOVC2      Último menos primero
STD      TEMP2        Actualiza el resultado
DESIC1    BCLR      TCTL2,X $30Deshabilita IC1
INC      IC1FIN      Medición de período terminada
IC1SAL    BCLR      TFLG1,X $FBBorra IC1F
RTI      ** Regresa de la rutina de servicio IC1 **

```

Figura 20. Listado de la rutina de interrupción OC1

La diferencia entre la primera y última captura de OC1 se almacena en TEMP1 y la diferencia de los contadores de desbordamiento se guardan en TEMP2 Si se unen (TEMP2:TEMP1) Se obtiene un registro de 32 bits que contiene el número total de ciclos de reloj durante los 8 períodos. Para obtener el período promedio debe

dividirse el número de ciclos entre 8. Como en este caso E es de 2MHz cada ciclo es de 0.5µs.

Podemos entonces saber el valor del período promedio en µs dividiendo el número total de ciclos entre 16, esto se logra realizando un corrimiento lógico del registro TEMP de cuatro posiciones hacia la derecha.

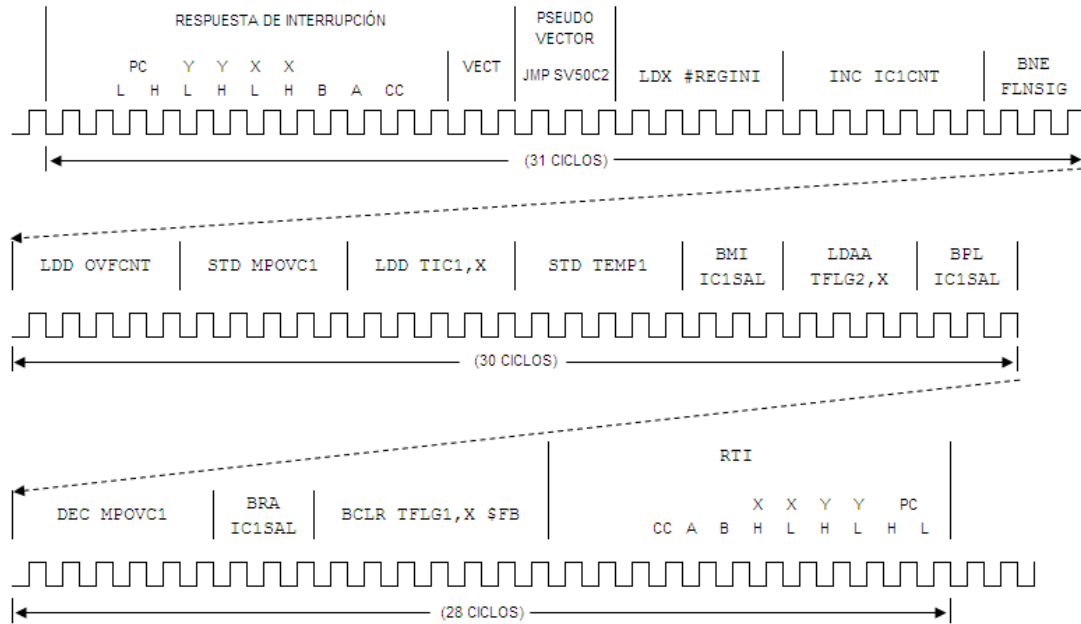


Figura 21. Análisis de tiempo de la rutina de interrupción IC1

En la Figura 21 se muestra el análisis de tiempo de la rutina de interrupción de IC1. La rutina de interrupción de IC1, que procesa el primer periodo requiere de 89 ciclos de reloj, y es más largo con respecto a los que procesan los periodos siguientes, a excepción del último, que es el más largo de todos, sin embargo, este último se ejecuta, cuando la última captura ya se realizó. Si E es igual a 2 MHz cada ciclo dura 0.5 µs y el tiempo total de los 89 ciclos es de 44.5 µs lo que permite medir una frecuencia máxima de hasta 22.4 kHz. La frecuencia mínima, estará dada por el período más largo, debido a que una vez que se obtiene el periodo promedio éste se guarda en TEMP1, que es una variable de 16 bits, entonces la cuenta máxima será de 65535 ciclos o un periodo de 32.8 ms que equivale a una frecuencia de 30.5 Hz.

Dado que se requiere medir el nivel en varias estaciones, a lo largo del modelo y sólo se puede disponer de hasta 4 capturas de entrada, se pueden multiplexar las señales provenientes de todos los transductores de nivel, de tal forma, que se tenga solamente una señal, siendo muestreada a la vez o en turnos de hasta cuatro señales.

Para lograr esto es necesario escoger el multiplexor adecuado, dependiendo del número de señales que se van a medir, y el número de puertos de captura de entrada que se van a utilizar. Una vez definido esto, se deben programar las señales de salida necesarias para el control del multiplexor.

### **3.6. Posición de las compuertas**

Como se discutió en el capítulo anterior, la posición de las compuertas será determinada utilizando sensores potenciométricos que proporcionarán una salida lineal entre 0 y 5 volts. Como la señal proveniente de dichos sensores es una señal analógica, ésta será convertida a valores digitales, a través del convertidor A/D del microcontrolador.

#### *3.6.1. Convertidor Analógico Digital*

Este sistema, es un convertidor por aproximaciones sucesivas, que utiliza una técnica de redistribución de carga capacitiva, para convertir señales analógicas a valores digitales.

El sistema A/D es un convertidor de entrada multiplexada de 8 bits 8 canales. El convertidor no requiere de circuitos externos de muestreo y mantenimiento de las señales, debido a la técnica utilizada de redistribución de carga. Los tiempos del convertidor A/D pueden ser sincronizados al sistema de reloj E, o al de un oscilador interno de tipo resistor capacitor (RC). El sistema convertidor A/D consiste de cuatro

bloques funcionales: multiplexor, convertidor análogo, control digital y almacenamiento del resultado.

### *3.6.2. Multiplexor del convertidor analógico digital.*

El multiplexor selecciona una de 16 entradas para conversión. La selección de la entrada es controlada por los valores de los bits CD:CA en el registro ADCTL. Los ocho pines del puerto E son entradas análogas de dirección fija hacia el multiplexor, y otras líneas de señales análogas internas adicionales están direccionadas hacia él.

Las terminales del puerto E pueden ser también usadas como entradas digitales. Las lecturas digitales de las terminales del puerto E no se recomiendan durante la porción de muestreo del ciclo de conversión A/D, cuando la señal de la compuerta hacia la entrada del canal N está encendida. Debido a que no existen dispositivos canal P conectados directamente a las terminales de entrada o a las terminales de referencia de voltaje, los voltajes mayores a VDD no causan daños al microcontrolador, aunque la corriente debe ser limitada de acuerdo a los rangos máximos.

### *3.6.3. Convertidor Analógico*

La conversión de una entrada analógica seleccionada por el multiplexor ocurre en esta etapa. Éste contiene un arreglo de capacitores digital a analógico, un comparador, un registro de aproximaciones sucesivas (SAR). Cada conversión es una secuencia de ocho operaciones de comparación, comenzando con el bit más significativo (MSB). Cada comparación determina el valor de un bit en el registro de aproximaciones sucesivas.

El arreglo DAC desempeña dos funciones. Actúa como un circuito de muestreo y mantenimiento de la señal durante toda la secuencia de conversión, y proporciona voltajes de comparación al comparador durante cada comparación sucesiva.

El resultado de cada comparación sucesiva se almacena en el SAR. Cuando se completa una conversión secuencial, el contenido del SAR es transferido al registro de resultados apropiado.

Una bomba de carga proporciona voltajes de transición a las compuertas de los interruptores análogos en el multiplexor. La bomba de carga debe estabilizarse entre 7 y 8 volts, dentro de un período de hasta 100µs antes de que el convertidor pueda ser usado. La bomba de carga está habilitada por el bit ADPU en el registro OPTION.

#### *3.6.4. Control Digital.*

Todas las operaciones están controladas por los bits en el registro ADCTL. Adicionalmente para seleccionar la entrada analógica a ser convertida, los bits ADCTL indican el estado de la conversión y controlan cuando se realizan conversiones simples o continuas. Finalmente, los bits ADCTL determinan cuando las conversiones se realizan en canales simples o múltiples.

#### *3.6.5. Registros de Resultados.*

Cuatro registros de 8 bits [4:1] almacenan los resultados de la conversión. El procesador de la CPU puede acceder a cada uno de estos registros. La bandera de conversión completa (CCF) indica cuando están presentes datos válidos en los registros de resultados. Los registros de resultados son escritos durante una porción del ciclo del reloj del sistema donde no ocurren lecturas, de tal forma que no existe conflicto.

#### *3.6.6. Relojes del Convertidor A/D*

El bit CSEL en el registro OPTION selecciona cuando el convertidor A/D utiliza el reloj del sistema E o un oscilador interno RC para sincronización. Cuando la frecuencia del reloj E está por debajo de 750kHz, la fuga de carga en el arreglo de

capacitores puede causar errores, y el oscilador interno debe ser usado. Cuando el reloj RC es usado, los errores adicionales pueden ocurrir debido a que el comparador es sensible al ruido adicional del reloj del sistema.

### 3.6.7. Secuencia de conversión

Las operaciones del convertidor A/D se realizan en secuencias de cuatro conversiones cada una. Una secuencia de conversión puede repetirse continuamente o detenerse después de cada interacción. La bandera de conversión completada (CCF) es fijada en 1 después de la cuarta conversión en cada secuencia, para mostrar la disponibilidad de datos en los registros de resultados.

### 3.6.8. Encendido del convertidor A/D y selección de Reloj

El bit 7 en el registro OPTION controla el encendido del convertidor A/D. Cuando ADPU es 0 se retira la energía y se deshabilita el sistema de conversión A/D. Cuando ADPU es 1 se habilita el sistema del convertidor A/D. Para que se establezcan los voltajes analógicos de bias se requiere de una demora de cuando mucho 100  $\mu$ s después de encender el convertidor A/D. Cuando el sistema de conversión está operando con el reloj E, todas las operaciones de transición y comparación están sincronizadas inherentemente con los relojes del MCU principal. Esto permite que la salida del comparador sea muestreada en períodos relativamente estables, durante los ciclos del reloj de la MCU. Dado que el oscilador interno RC es asíncrono con el reloj MCU, existen más errores atribuibles al ruido interno del reloj del sistema. La exactitud del convertidor A/D es reducida ligeramente, cuando el oscilador interno RC es utilizado (CSEL=1).

**OPTION**- System Configuration Options

**\$1039**

	Bit7	6	5	4	3	2	1	Bit 0
	ADPU	CSEL	IRQE1	DLY1	CME	-	CR11	CR01
RESET:	0	0	0	1	0	0	0	0



NOTAS:

1. Puede ser escrita una vez dentro de los primeros 64 ciclos desde el reset en modos normales, o en cualquier momento en los modos especiales

ADPU- Analog to Digital Converter Power Up

0 = A/D apagado

1 = A/D encendido

CSEL- Clock Select

0 = A/D y EEPROM utilizan el reloj del sistema E

1 = A/D y EEPROM utilizan el reloj interno RC

IRQE- Configura IRQ para operación sólo sensible a flanco.

0 = IRQ está configurado para operación sensible al nivel.

1 = IRQ está configurado para operación sólo sensible a flanco.

DLY- Habilita Oscillator Startup Delay

0 = El retardo del inicio del oscilador saliendo del estado STOP es desviado y el microcontrolador continúa procesando dentro de aproximadamente cuatro ciclos de bus.

1 = Un retardo de aproximadamente 4000 ciclos de reloj E es impuesto cuando el microcontrolador es iniciado del estado STOP de ahorro de energía. Este retardo permite que el oscilador de cristal se estabilice.

CME Habilitar monitor de Reloj.

Este bit de control puede ser leído o escrito en cualquier momento y determina si el circuito monitor del reloj interno debe disparar o no una secuencia de reset cuando el reloj del sistema es lento o ausente. Cuando está limpio (es cero), el monitor del circuito del reloj está deshabilitado, y cuando está fijado en 1, el circuito monitor de reloj está habilitado. Durante el RESET se limpia el bit CME.

0 = Monitor de Reloj deshabilitado

1 = Reloj lento o detenido causa reset.

Bit 2 - No implementado

Siempre se lee como cero

CR[1:0] - COP Timer Rate Select

El reloj interno E es primero dividido en 215 antes de entrar al sistema de vigilancia COP. Estos bits de control determinan el factor de escala para el vigía del temporizador.

### 3.6.9. Proceso de Conversión

La secuencia de conversión empieza un ciclo de reloj E después de una escritura al registro de control/estado de A/D, ADCTL. Los bits en ADCTL seleccionan el canal y el modo de conversión.

Un voltaje de entrada equivalente a VRL se convierte en \$00 y un voltaje de entrada igual a VRH se convierte en \$FF (escala completa), sin indicación de desbordamiento. Para conversiones de este tipo, la fuente de cada entrada analógica debe usar VRH como fuente de voltaje y debe estar referenciada a VRL

**3.6.10. Asignación de Canales.**

El mutiplexor permite al convertidor A/D seleccionar una de 16 señales analógicas. Ocho de estos canales corresponde a las entradas del puerto E al MCU, cuatro de los canales son puntos de referencia internos o funciones de prueba, y otros cuatro canales están reservados.

Número de canal	Señal del Canal	Resultado en ADRx
1	AN0	ADR1
2	AN1	ADR2
3	AN2	ADR3
4	AN3	ADR4
5	AN4	ADR1
6	AN5	ADR2
7	AN6	ADR3
8	AN7	ADR4
9 – 12	Reserved	---
13	VRH	ADR1
14	VRL	ADR2
15	VRH/2	ADR3
16	Reserved	ADR4

Tabla 4. Canales del Convertidor AD

**3.6.11. Operación de Canal Simple**

Existen dos tipos de operación de canal simple. Cuando SCAN = 0, el primer tipo, el canal simple seleccionado es convertido cuatro veces consecutivas. El primer resultado es almacenado en el registro de resultados A/D 1 (ADR1), y el cuarto resultado se almacena en ADR4. Después de que la cuarta conversión está completa,

toda la actividad de conversión es suspendida, hasta que un nuevo comando de conversión es escrito en el registro ADCTL. En el segundo tipo de operación de canal simple, SCAN = 1, las conversiones continúan siendo efectuadas en el canal seleccionado y la quinta conversión es almacenada en el registro ADR1 (sobrescribiendo el resultado de la primera conversión), la sexta conversión sobrescribe ADR2 y así, sucesivamente.

### 3.6.12. *Operación de canales múltiples*

Existen dos tipos de operación de canales múltiples. Cuando SCAN = 0, el primer tipo, un grupo de cuatro canales seleccionados es convertido una vez cada uno. El primer resultado es almacenado en el registro de resultados A/D 1 (ADR1), y el cuarto resultado es almacenado en ADR4. Después que la cuarta conversión está completa, toda la actividad de conversión es suspendida, hasta que un nuevo comando de conversión es escrito en el registro ADCTL. En el segundo tipo de operación de canales múltiples, SCAN = 1, las conversiones continúan siendo efectuadas en el grupo de canales seleccionados, donde la quinta conversión es almacenada en el registro ADR1 (reemplazando la conversión anterior resultado del primer canal en el grupo).

### 3.6.13. *Operación en modo STOP y WAIT*

Si la secuencia de conversión está en progreso cuando es activado el modo STOP o WAIT, la conversión del canal actual es suspendida. Cuando la MCU regresa a la operación normal, el canal es muestreado nuevamente y la secuencia es reanudada. Cuando la MCU sale del modo WAIT, los circuitos A/D son estables y se pueden obtener resultados válidos desde la primera conversión. De cualquier forma, en el modo STOP, todas las corrientes de bias están deshabilitadas y es necesario permitir un periodo de estabilización cuando se sale del modo STOP. Si se sale del modo STOP con demora (DLY = 1), existe suficiente tiempo para esos circuitos para estabilizarse antes de la primera conversión. Si se sale del modo STOP sin demora

(DLY = 0), es necesario permitir 10 ms para que el circuito A/D se estabilice y así evitar resultados no válidos.

### 3.6.14. Registros de estado/control A/D

Todos los bits en el registro pueden ser leídos o escritos, excepto el bit 7, el cual es un indicador de estado de sólo lectura, y el bit 6, que siempre se lee como cero. Escriba en ADCTL para iniciar una conversión. Para salir de una conversión en proceso, escriba en este registro y una nueva secuencia de conversión comienza inmediatamente.

ADCTL- A/D Control Status							\$1030
Bit7	6	5	4	3	2	1	Bit 0
CCF	-	SCAN	MULT	CD	CC	CB	CA
RESET:	1	0	1	1	1	1	1

#### CCF- Conversions Complete Flag

Un indicador de estado de sólo lectura, este bit cambia a cuando los cuatro registros de resultados A/D contienen resultados de conversiones válidos. Cada vez que el registro ADCTL es sobrescrito, este bit automáticamente cambia a cero y se inicia una secuencia de conversión. En el modo continuo CCF es 1 al final de la primera secuencia de conversión.

#### Bit 6 - No implementado

Siempre se lee cero

#### SCAN - Continuous Scan Control

Cuando este bit de control es cero, las cuatro conversiones son realizadas una vez para llenar los cuatro registros. Cuando este bit de control es 1, las conversiones son realizadas continuamente y los registros de conversión son actualizados conforme los datos están disponibles.

#### MULT - Multiple Chanel/Single Channel Control

Cuando este bit es cero, el sistema convertidor A/D está configurado para realizar cuatro conversiones consecutivas en el canal simple especificado por los cuatro bits de selección de canal CD:CA (bits [3:0] del registro ADCTL). Cuando este bit es 1, el sistema A/D está configurado para realizar una conversión en cada uno de los cuatro canales donde cada resultado corresponde a un canal.

Nota. Cuando se selecciona el modo continuo de canal múltiple se requiere cuidado especial en el diseño de los circuitos que conducen las entradas A/D. La carga en el arreglo de capacitores DAC antes del tiempo de muestreo está relacionada con el voltaje del canal previamente convertido. Existe una situación de repartición de carga entre la capacitancia interna DAC y la capacitancia externa del circuito. Aunque la cantidad de carga involucrada es pequeña, la tasa a la cual se repite es cada 64  $\mu$ s para un reloj E de 2 MHz. La tasa de carga RC del circuito externo debe ser balanceada contra este efecto para evitar errores de exactitud.

#### CD:CA - Channel Selects D:A

Cuando se selecciona el modo de canal múltiple (MULT = 1), los dos bits de selección de canal menos significativos (CD y CA) no tienen ningún significado y los bits CD y CC especifican cual grupo de cuatro canales será convertido.

Bits de Control de Seleccion de Canal CD:CC:CB:CA	Canal de Señal	Resultado en ADRx si MULT = 1
0000	AN0	ADR1
0001	AN1	ADR2
0010	AN2	ADR3
0011	AN3	ADR4
0100	AN4	ADR1
0101	AN5	ADR2
0110	AN6	ADR3
0111	AN7	ADR4
10XX	Reservado	-
1100	VRH1	ADR1
1101	VRL1	ADR2
1110	(VRH)/21	ADR3
1111	Reservado1	ADR4

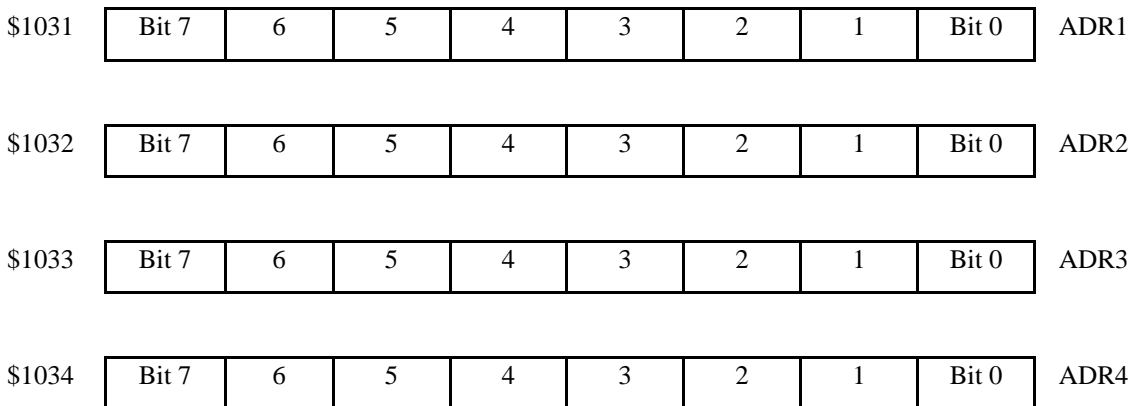
Notas:

1. Usado para pruebas de fábrica.

**Tabla 5. Selección de canales del convertidor A/D**

3.6.15. Registros de resultados del convertidor A/D

Estos registros de sólo lectura mantienen un resultado de conversión de 8 bits. Las escrituras a este registro no tienen ningún efecto. Los datos en los registros del convertidor A/D son válidos cuando la bandera CCF en el registro ADCTL es 1, indicando que la secuencia de conversión está completa.



```

ADCTL EQU    $30
PORTA EQU    $00
ADR1  EQU    $31
INICIO LDX   #$1000
LDAA  #$80
STAA  OPTION,X      ; encender el conversor
LDAA  #$20           ; configuración conversor:
STAA  ADCTL,X       ; SCAN -> activo
; MULT -> inactivo
; ADR1 -> seleccionar primer canal
SIGUE BRCLR ADCTL,X $80 sigue ; espera a que termine conversión
LDAA  ADR1,X        ; leer el resultado de la conversión
CMPA  #$7F          ; comparar con la mitad (127 en decimal)
; que corresponde a 2.5v de entrada.
BLO   MENOR         ; si es menor apagar el led
LDAA  #$40           ; No--> encender el led
STAA  PORTA,X
BRA   SUGUE         ; Realizar la siguiente conversión
MENOR CLRA
STAA  PORTA,X      ; Apagar el led.
BRA   SIGUE        ; Realizar la siguiente conversión
END
    
```

Figura 22. Listado del convertidor A/D

### 3.7. Computadora Personal

La computadora personal desempeña un papel importante como parte del sistema, ya que se encargará de almacenar las diversas bases de datos. Estas bases de datos son primeramente las que contienen la información estadística (de las mareas y avenidas), que servirán como directrices de control que deberán ser enviadas al microcontrolador. La otra base de datos es aquella donde se almacenarán los datos recabados desde el modelo, que serán recibidas desde el microcontrolador. A partir de los datos estadísticos el programa de la PC generará las directrices de control, estableciendo los nuevos valores a alcanzar dentro del modelo. Así mismo, deberá ser capaz de comunicarse con el microcontrolador, para el envío de dichas directrices y la recepción de los datos medidos en el modelo.

La computadora personal no hará ningún tipo de control, esa tarea está dedicada al HC11, la función de la computadora sólo es para almacenamiento de información, y sobre comandar puntos como niveles de flujo o caudal dentro del sistema. Será la interfaz gráfica con el usuario y el sistema de control.

La comunicación con el microcontrolador será a través del puerto serie de la computadora.

No es alcance de esta Tesis el desarrollar el manejo de base de datos dentro de la computadora.

#### *3.7.1. Comunicaciones entre el MC68HC11 y la computadora personal*

La comunicación entre la computadora personal será a través de comandos definidos que servirán por un lado para solicitar que el microcontrolador envíe información del estado actual de los transductores de nivel y flujo para ser almacenados en las bases de datos de la computadora. Por otro lado existirán comandos destinados para establecer los nuevos valores meta que deberán enviarse

al microcontrolador para establecer las entradas del sistema de control. Estos comandos o instrucciones se explican más adelante en este capítulo.

Para comunicar el microcontrolador con una computadora personal se utilizará la interfaz de comunicaciones serial o SCI (Serial Communications Interface) del microcontrolador, la cual es un sistema full-duplex UART (universal asynchronous receiver transmitter), que utiliza el formato no retorno a cero o NRZ (un bit de inicio, ocho o nueve bits de datos y un bit de paro). Un generador de tasas de transferencia produce frecuencias de tasas de transferencia Standard, a partir del oscilador del MCU. Ambos, el receptor y el transmisor, son funcionalmente independientes pero utilizan el mismo tipo de formato de datos y la misma tasa de transferencia.

En la recepción se toman tres mediciones del nivel lógico cercano a la mitad del tiempo de cada bit, para decidir el sensado correcto de cada bit, aún cuando el ruido cause que alguna de esas mediciones sea incorrecta.

El receptor también tiene la capacidad de entrar en modo de espera, para ignorar mensajes de diferentes receptores.

### *3.7.2. Formato de datos.*

El formato de datos en serie requiere de las siguientes condiciones:

1. Una línea libre en estado alto antes de la transmisión o recepción de un mensaje.
2. Un bit de inicio, cero lógico, transmitido o recibido, que indica el inicio de cada caracter.
3. Los datos que son transmitidos y recibidos empezando por el bit menos significativo (LSB).
4. Un bit de paro, uno lógico, usado para indicar el final de la trama. (Una trama consiste de un bit de inicio, un caracter de 8 o 9 bits de datos, y un bit de paro)



5. Una pausa (definida como una transmisión o recepción de un cero lógico por un número múltiple de tramas)

La selección de la longitud de la palabra es controlada por el bit M del registro de control de SCI SCCR1

### *3.7.3. Operación de Transmisión.*

El transmisor SCI incluye un registro de datos de transmisión en paralelo (SCDR) y un registro de corrimiento serial. El contenido del registro de corrimiento serial puede ser escrito solamente a través de SCDR. Esta operación doble permite a un caracter ser recorrido hacia afuera serialmente mientras que otro caracter está esperando en el registro SCDR para ser transferido en el registro de corrimiento serial. La salida del registro de corrimiento serial es aplicada a TxD mientras la transmisión está en proceso o el bit de habilitar transmisión (TE) de registro de control de comunicación serial 2 (SCCR2) es uno.

### *3.7.4. Operación de Recepción*

Durante las operaciones de recepción, la secuencia de transmisión es a la inversa. El registro de corrimiento en serie recibe los datos y los transfiere en un registro paralelo de recepción de datos (SCDR) como una palabra completa. Esta operación doble permite que un caracter sea recorrido en serie mientras otro caracter ya se encuentre en SCDR. Un esquema de recuperación de datos avanzado distingue los datos válidos, del ruido en el canal de datos en serie. La entrada de datos es muestreada selectivamente para detectar los datos recibidos, un circuito de votación de mayoría determina el valor y la integridad de cada bit.

### *3.7.5. Detección de errores*

Tres condiciones de error pueden ocurrir durante la generación de interrupciones por el sistema SCI, SCDR overrun, ruido en el bit recibido, y framing. Tres bits (OR, NF, y FE) en el registro de estado de comunicación serial (SCSR) indican si existe alguna de estas condiciones de error.

El bit de error overrun (OR) es fijado cuando el siguiente byte está listo para ser transferido desde el registro de corrimiento de recepción al registro SCDR y SCDR ya está lleno (el bit RDRFes uno). Cuando ocurre un error overrun, los datos causantes del overrun son perdidos y los datos que ya se encontraban en SCDR no son alterados. El OR es borrado cuando el SCSR es leído (con OR fijado), seguido de una lectura de SCDR.

La bandera de ruido (NF) es fijada en uno si existe ruido en alguno de los bits recibidos, incluyendo los bits de inicio y paro. El bit NF no es fijado hasta que la bandera RDRF es fijada. El bit NF es borrado cuando el SCSR es leído (con FE igual a uno) seguido de una lectura del SCDR.

Cuando no se detecta un bit de paro en el caracter de recepción de datos, el bit de error framing (FE) es fijado. FE es fijado al mismo tiempo que el RDRF. Si el byte recibido causa ambos errores framing y overrun, el procesador sólo reconoce el error overrun. La bandera de error framing inhibe más transferencias de datos en SCDR hasta que es borrada. El bit FE es borrado cuando el SCSR es leído (con FE igual a uno), seguido de una lectura de SCDR.

### *3.7.6. Registros*

Existen cinco registros direccionables asociados con el sistema SCI. SCCR1, SCCR2 y BAUD son registros de control. SCDR es el registro de datos de SCI y SCSR el registro de estado de SCI. Los registros SCI son los mismos para toda la serie E de la familia MC68HC11 excepto por una excepción. El sistema SCI del

MC68HC11E20 contiene un bit extra en el registro BAUD que proporciona una mayor selección del preescalador de la tasa de transmisión.

### 3.7.7. Registro de Datos de Comunicación en Serie.

SCDR es un registro paralelo que realiza dos funciones. Es el registro de recepción de datos cuando es leído y registro de transmisión de datos cuando es escrito. Las lecturas accesan a la información recibida y las escrituras accesan al canal de transmisión de datos.

**SCDR- SCI Data Register** **\$102F**

	Bit7	6	5	4	3	2	1	Bit 0
	<b>R7/T7</b>	<b>R6/T6</b>	<b>R5/T5</b> t	<b>R4/T4</b>	<b>R3/T3</b>	<b>R2/T2</b>	<b>R1/T1</b>	<b>R0/T0</b>
RESET:	1	1	1	1	1	1	1	1

### 3.7.8. Registro de Control de Comunicación en Serie 1

El registro SCCR1 proporciona los bits de control que determinan el largo de la palabra y seleccionan el método usado por la característica del despertador

**SCCR1 - SCI Control Register** **\$102C**

	Bit7	6	5	4	3	2	1	Bit 0
	<b>R8</b>	<b>T8</b>	-	<b>M</b>	<b>WAKE</b>	-	-	-
RESET	1	1	0	0	0	0	0	0

R8 - Receive Data bit 8

Si el bit M es 1, R8 almacena el noveno bit en el caracter de recepción de datos.

T8 - Transmt Data bit 8

Si el bit M es 1, T8 almacena el noveno bit en el caracter de transmisión de datos.

Bit 5 - No implementado

Siempre se lee en cero

M - Mode (Selecciona el Formato de Caracter)  
 0 = Bit de inicio, 8 bits de datos, 1 bit de paro  
 1 = Bit de inicio, 9 bits de datos, 1 bit de paro

WAKE - Wakeup by Address Mark/Idle  
 0 = Despertador por reconocimiento de línea libre  
 1 = Despertador por marca de dirección (el bit mas significativo de los datos es uno)

Bits [2:0] - No implementados  
 Siempre se leen en cero

### 3.7.9. Registro de Control de Comunicaciones en Serie 2

El registro SCCR2 proporciona los bits de control que habilitan o deshabilitan funciones individuales SCI

	SCCR2 - SCI Control Register 2							\$102D
	Bit7	6	5	4	3	2	1	Bit 0
	TIE	TCIE	RIE	ILIE	TE	RE	RWU	SBK
RESET	0	0	0	0	0	0	0	0

TIE - Transmit Interrupt Enable  
 0 = Interrupciones TDRE deshabilitadas  
 1 = Interrupción SCI solicitada cuando la bandera TC es fijada

TCIE - Transmit Complete Interrupt Enable  
 0 = Interrupciones TC deshabilitadas  
 1 = Interrupción SCI solicitada cuando la bandera TC es fijada

RIE - Receiver Interrupt Enable  
 0 = Interrupciones RDRF y OR deshabilitadas  
 1 = Interrupción SCI solicitada cuando la bandera RDRF o la bandera de estado OR es fijada

ILIE - Idle Line Interrupt Enable  
 0 = Interrupciones IDLE deshabilitadas  
 1 = Interrupción SCI solicitada cuando la bandera de estado IDLE es fijada

TE - Transmitter Enable  
 Cuando TE cambia de cero a uno, una unidad de carácter de tiempo libre (uno lógico) es enviada a la cola como preámbulo  
 0 = Transmisor deshabilitado  
 1 = Transmisor habilitado

RE - Receiver Enable  
 0 = Receptor deshabilitado  
 1 = Receptor habilitado

RWU - Receiver Wakeup Control  
 0 = Receptor SCI normal  
 1 = Despertador habilitado e interrupciones de receptor inhibidas

SBK - Send Break

Al menos un caracter de tiempo libre es enviado cada vez que SBK es escrito a uno. Mientras el bit SBK está fijado, los caracteres break son puestos en la cola y enviados. Más de un break puede ser enviado si el transmisor está libre el tiempo que el bit SBK es encendido y apagado, como el flanco de reloj de la tasa de comunicación ocurre entre la escritura de un uno y el cero a SBK.

- 0 = Generador de break apagado
- 1 = Codigos Break generados

### 3.7.10. Registro de estado de Comunicaciones en Serie

El SCSR proporciona entradas a los circuitos lógicos de interrupción para la generación de la interrupción de sistema SCI

SCSR - SCI Status Register

\$102E

	Bit7	6	5	4	3	2	1	Bit 0
	<b>TDRE</b>	<b>TC</b>	<b>RDRF</b>	<b>IDLE</b>	<b>OR</b>	<b>NF</b>	<b>FE</b>	-
RESET	1	1	0	0	0	0	0	0

TDRE - Transmit Data Register Empty Flag

Esta bandera es fijada cuando SCDR está vacía. Limpie la bandera TDRE leyendo SCSR con TDRE fijado y luego escribiendo en SCDR.

- 0 = SCDR ocupado
- 1 = SCDR vacío

TC - Transmit Complete Flag

Esta bandera es fijada cuando el transmisor está libre (sin datos, preámbulo, o transmisión de break en progreso). Limpie la bandera TC leyendo SCSR con TC fijado en 1 y luego escribiendo a SCDR.

- 0 = Transmisor ocupado
- 1 = Transmisor libre

RDRF - Receive Data Register Full Flag

Esta bandera es fijada si un caracter recibido está listo para ser leído desde SCDR. Limpie la bandera RDRF leyendo SCSR con RDRF fijado y luego leyendo SCDR.

- 0 = SCDR vacío
- 1 = SCDR lleno

IDLE - Idle Line Detected Flag

Esta bandera es fijada si la línea RxD está libre. Una vez limpiada, IDLE no es fijada nuevamente hasta que la línea RxD ha estado activa y vuelve a quedar libre. La bandera IDLE es inhibida cuando RWU = 1. Limpie IDLE leyendo SCSR con IDLE fijado en 1 y luego leyendo SCDR.

- 0 = Línea RxD activa
- 1 = Línea RxD libre

OR - Overrun Error Flag

OR es fijado si un nuevo caracter es recibido antes de que un caracter recibido previamente sea leído de SCDR. Limpie la bandera OR leyendo SCSR con OR fijado en 1 y luego leyendo SCDR.

- 0 = Sin overrun
- 1 = Overrun detectada

NF - Noise Error Flag

NF es fijada si la lógica de muestreo detecta cualquier otra cosa que no sea una decisión unánime. Limpie

NF leyendo SCSR con NF fijada en uno y luego leyendo SCDR.  
 0 = Decisión unánime  
 1 = Ruido detectado

FE - Framing Error

FE es fijado cuando un cero es detectado donde un bit de paro es esperado. Limpie la bandera leyendo SCSR con FE fijado luego leyendo SCDR.  
 0 = Bit de Paro Detectado  
 1 = Cero detectado

Bit 0 - No implementado

Siempre se lee en cero

### 3.7.11. Registro de Tasa de Transferencia

Utilice este registro para seleccionar diferentes tasas de transferencia para el sistema SCI. Los bits SCP[1:0] funcionan como un pre-escalador para los bits SCR[2:0]. Juntos estos cinco bits proporcionan múltiples combinaciones de tasas de transferencia para una frecuencia dada del cristal oscilador del reloj. Normalmente este registro es escrito una vez durante la inicialización. El pre-escalador es fijado a su tasa más rápida por omisión al salir del reset, y puede ser cambiada en cualquier tiempo.

**BAUD** - Baud Rate

**\$102B**

	Bit7	6	5	4	3	2	1	Bit 0
	<b>TCLR</b>	<b>SCP2</b>	<b>SCP1</b>	<b>SCP0</b>	<b>RCKB</b>	<b>SCR2</b>	<b>SCR1</b>	<b>SCR0</b>
RESET	1	1	0	0	0	0	0	0

TCLR - Clear Baud Rate Counters (Test)

SCP[2:0] - SCI Baud Rate Prescaler Selects

Note que SCP2 aplica sólo para MC68HC(7)11E20. Cuando SCP2 = 1, SCP[1:0] deben ser ceros. Cualesquiera otros valores para SCP[1:0] no son decodificados en el pre-escalador y los resultados son impredecibles. Refiérase a los diagramas de bloques del generador de tasas de transferencia SCI

				Cristal				
SCP2	SCP1	SCP0	Divide el reloj interno entre	4.0	4.9152	8.0	8.3886	12.0
0	0	0	1	62500	76800	125000	131072	187500
0	0	1	2	20833	25600	41667	43691	62500
0	1	0	4	15625	19200	31250	32768	46875
0	1	1	13	4800	5907	9600	10082	14423
1	0	0	39	1602	1969	3205	3361	4808
1	0	1						
1	1	0						
1	1	1						

**Tabla 6. Selector del prescalador de Tasa de Transferencia**

RCKB - SCI Baud Rate Clock Check (Test)

SCR[2:0] - SCI Baud Rate Selects

Selecciona la tasa de bit para transmisión y recepción basado en el preescalador de tasa de transferencia. Refiérase a los diagramas de bloques del generador de tasas de transferencia SCI

Selector de Tasa de Transferencia.

SCR[2:0]	Preescalador Dividido entre	131072	76800	32768	19200	4800
0 0 0	1	131072	76800	32768	19200	4800
0 0 1	2	65536	38400	16384	9600	2400
0 1 0	4	32768	19200	8192	4800	1200
0 1 1	8	16384	9600	4096	2400	600
1 0 0	16	8192	4800	2048	1200	300
1 0 1	32	4096	2400	1024	600	150
1 1 0	64	2048	1200	512	300	75
1 1 1	128	1024	600	256	150	-

**Tabla 7. Tasa de transferencia más alta (Salida del pre-escalador de la tabla anterior)**

Los bits del pre-escalador, SCP[2:0], determinan la tasa más alta, y los bits SCR[2:0] seleccionan un submúltiplo binario adicional ( $\div 1$ ,  $\div 2$ ,  $\div 4$ , hasta  $\div 128$ ) de su tasa de transferencia más alta. El resultado de estos dos divisores en serie es la tasa de transferencia del reloj receptor 16x. Los bits SCR[2:0] no son afectados por reset y

pueden cambiarse en cualquier momento, sin embargo, no deben ser cambiados cuando una transferencia SCI está en progreso.

### 3.8. Lógica Difusa

La lógica difusa, una teoría desarrollada por Lotfi A. Zadeh, Ph. D. provee una serie de definiciones que tienen bordes difusos, en lugar de bordes definidos de la lógica aristotélica, estos bordes se pueden traslapar, de tal forma que una función puede evaluarse en un momento dado y ser verdadera en cierto grado para 2 o más variables etiquetadas lingüísticamente. Esto quiere decir, que a pesar de la connotación negativa que se le da al término “difuso”, los sistemas basados en lógica difusa pueden producir resultados tan precisos y repetibles, como aquellos producidos por los métodos tradicionales. El término difuso se refiere entonces a que en el mundo real los conjuntos pueden tener bordes graduales. Por ejemplo para la expresión “la temperatura es tibia”, no existe una temperatura específica en la que esta expresión cambia de ser completamente verdadera, a ser completamente falsa, es decir, existe un borde gradual o difuso que requiere una descripción no binaria de verdad.

La lógica difusa utiliza funciones de membresía bidimensionales para expresar el significado de un parámetro tal como el de “la temperatura es tibia”. En la Figura 23 se muestra un juego de conjuntos difusos para definir la temperatura, los cuales son frío, tibio y caliente. Al nombre que recibe cada conjunto se le denomina etiquetas. Cuando se dice que un parámetro tiene un cierto número de etiquetas, cada una de las etiquetas corresponde a un conjunto difuso. Debido a que estos conjuntos tienen bordes graduales, éstos se traslapan y un determinado valor de temperatura puede pertenecer al mismo tiempo a dos o más conjuntos. El grado de verdad, es decir que tanto pertenecen a cada uno de los conjuntos, está dado por un valor que va de 0 a 1.



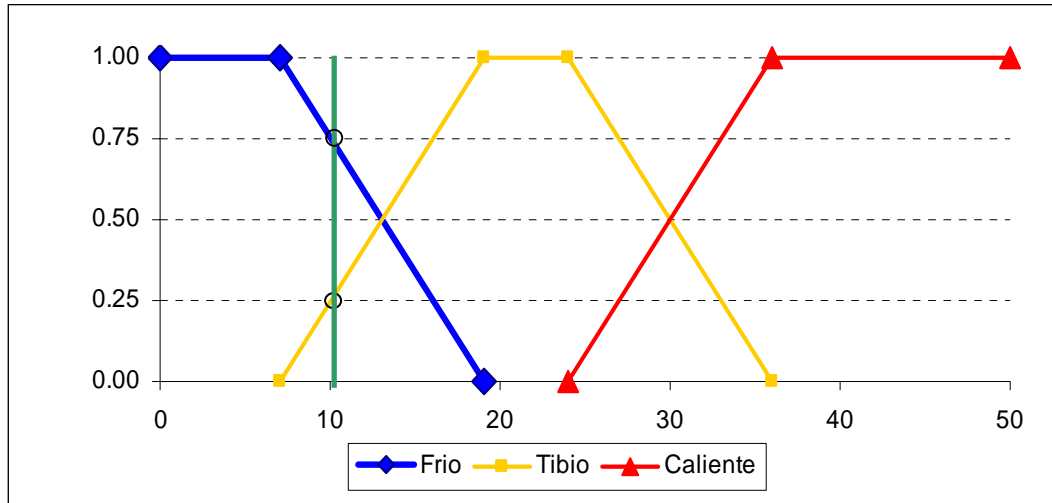


Figura 23. Conjuntos difusos (temperatura)

Una característica de la lógica difusa es que se basa en el conocimiento experto, por ejemplo una expresión como “si la temperatura es caliente entonces la velocidad del ventilador es máxima” aunque este tipo de enunciados pudieran parecer imprecisos, a través de la lógica difusa se pueden procesar eficientemente. Estas expresiones están basadas en el conocimiento de algún experto en el sistema de control y son conocidas como “reglas expertas”. Estas reglas son fundamentales para la implementación de los sistemas difusos.

Las etapas básicas de un control difuso son las siguientes:

- Difusión (Fuzzification)
- Evaluación de Reglas (Rule Evaluation)
- Desdifusión (Defuzzification)

### 3.8.1. Difusión

En la etapa de difusión, el valor actual de cada variable de entrada es comparado con las funciones de membresía para cada etiqueta de la entrada correspondiente. A partir de esto, es posible determinar un valor verdadero numérico para cada etiqueta de cada entrada.

En el ejemplo de la Figura 23 una temperatura de 10°C es representada por la línea de referencia vertical. Esta línea intersecta a la función de membresía “Frío” a una altura de 0.75, lo que indica que pertenece en gran medida a esta etiqueta. A su vez intersecta a la función de membresía “Tibio” en 0.25 lo que indica que la expresión “la temperatura es tibia” es verdadera en un pequeño grado. Por otro lado se observa que la línea de referencia no intersecta a la función de membresía “Caliente” lo cual indica que el grado en que la expresión “la temperatura es caliente” puede considerarse verdadera es cero (no es verdadera), y por lo tanto, la entrada difusa para esta etiqueta será cero.

Para lograr implementar un sistema difuso por medio de un microcontrolador como el HC11, es necesario definir las variables y sus funciones de membresía dentro de un rango que éste pueda interpretar. Los parámetros de entrada como el gasto y el nivel que son recibidos por el microcontrolador, deberán ser preprocesadas y escaladas para tener un valor numérico que encaje dentro del rango de \$00 a \$FF. En la etapa de difusión, los parámetros de entrada se compararán con las funciones de membresía definidas para cada una de sus etiquetas, y se definirá el grado de pertenencia a cada una de ellas. El resultado de esta comparación en lugar de dar como resultado un valor entre 0 y 1, estará comprendido entre \$00 y \$FF, donde \$00 indica que no pertenece en absoluto o la evaluación es falsa (no verdadera) y \$FF indica que es verdadera o que pertenece completamente. Los resultados de estas comparaciones se almacenarán en la memoria RAM del microcontrolador, por lo tanto, para cada una de las etiquetas de cada una de las entradas deberá existir un byte en la memoria RAM.

Una forma de definir las funciones de membresía, es utilizar funciones de membresía trapezoidales (como las de la Figura 23) definidas por dos puntos y dos pendientes que se almacenan en memoria permanente. Una aplicación con dos entradas y cinco etiquetas por entrada necesitará 10 funciones de membresía de 4

bytes cada una, por lo tanto requiere de 40 bytes en memoria ROM o EEPROM y 10 entradas difusas, es decir 10 bytes de memoria RAM.

El programa del microprocesador, en esta etapa, se encargará de encontrar la intersección entre la función de membresía para una etiqueta determinada, con el valor correspondiente de la entrada al sistema y almacenar el resultado en la localidad de memoria de dicha etiqueta.

### *3.8.2. Evaluación de reglas*

Esta etapa también se conoce como inferencia difusa, ya que utiliza la técnica de inferencia de máximos y mínimos para obtener resultados numéricos de las reglas lingüísticas, basadas en los valores de las variables de entrada del sistema. Los resultados numéricos de este proceso son las llamadas salidas difusas.

Aún cuando al definir las reglas éstas parecen estar expresadas en lenguaje natural, éstas deben seguir ciertas reglas y cumplir con una sintaxis estricta. El núcleo típico de la lógica difusa en un sistema de control limita las reglas de la siguiente forma:

```
IF entrada_del_sistema x es etiqueta_a
    AND entrada_del_sistema_y es etiqueta_b
    THEN salida_w es etiqueta_c
```

O dicho de otra forma:

```
SI entrada_del_sistema x es etiqueta_a
    Y entrada_del_sistema_y es etiqueta_b
    ENTONCES salida_w es etiqueta_c
```

Un ejemplo de una regla se enuncia a continuación:

Si la temperatura es baja y la humedad es alta, entonces la potencia del calentador es alta.

Cada una de las expresiones lingüísticas tales como “entrada\_de\_sistema\_x es etiqueta\_a”, son conocidas como antecedentes, corresponde con un valor almacenado en la memoria RAM como resultado de la etapa anterior (difusión). La expresión “salida\_w es etiqueta\_c” es conocida como consecuente y corresponde a una salida difusa específica. El operador AND es un operador de la lógica difusa que corresponde con el mínimo matemático. Todos los antecedentes de una regla son conectados por operadores AND. Existe un OR implícito entre reglas sucesivas, que corresponden a la operación del máximo matemático.

Entonces, la función principal de la etapa de evaluación de reglas, consiste en determinar el antecedente de regla más pequeño (mínimo) que es tomado como valor verdadero de la regla y aplicar el valor correspondiente a todos los consecuentes de la regla. Cuando una salida resulta ser el consecuente de más de una regla, la salida será el valor verdadero más alto (máximo) de todas las reglas que lo incluyen como consecuente.

Antes de empezar a procesar las reglas, todas esas salidas difusas se inicializan a \$00 (lo que significa nada verdadero). Conforme las reglas van siendo procesadas, el valor verdadero para la regla actual es almacenado en cada salida difusa para cada consecuente, a menos que el valor que se encuentra almacenado sea mayor (operación de máximo).

### *3.8.3. Desdifusión*

Después de la etapa de evaluación de reglas, cada una de las salidas difusas, tiene un valor correspondiente al grado en que esa salida debe ser aplicada. Esto puede ser considerado como recomendaciones para el nivel de la salida del sistema.

La etapa de la desdifusión, combina estas recomendaciones separadas, en un solo valor compuesto de salida del sistema.

A diferencia de las funciones de membresía de las entradas, para las salidas difusas normalmente se utilizan funciones tipo singleton, aunque pueden utilizarse de otro tipo. Las funciones tipo singleton están dadas simplemente por la posición en el eje x de una etiqueta del sistema, donde el valor de la salida difusa representa la altura (valor y) de esa función de membresía o el grado en que esa salida debe aplicarse. Para combinar todas las salidas y obtener una salida única, una de las opciones es utilizar la salida más fuerte, lo cual se conoce como el método de desdifusión máxima. Este método se considera muy pobre, ya que ignora la contribución de las demás salidas difusas. Un método más aceptado, es el método del centroide o centro de gravedad, el cual toma en cuenta todas las salidas difusas y el grado en que cada una es verdadera. El centroide de todas las salidas difusas se calcula de la siguiente forma.

$$\frac{\sum_{i=1}^n F_i * S_i}{\sum_{i=1}^n F_i}$$

Donde:

n es el número de salidas difusas asociadas con la salida del sistema.

$F_i$  es un peso (valor de salida difusa de la RAM).

$S_i$  es una posición de la función de membresía singleton (de la base de conocimientos).

El resultado de este cálculo es la acción de salida del sistema.  $F_i$  y  $S_i$  son valores de 8 bits

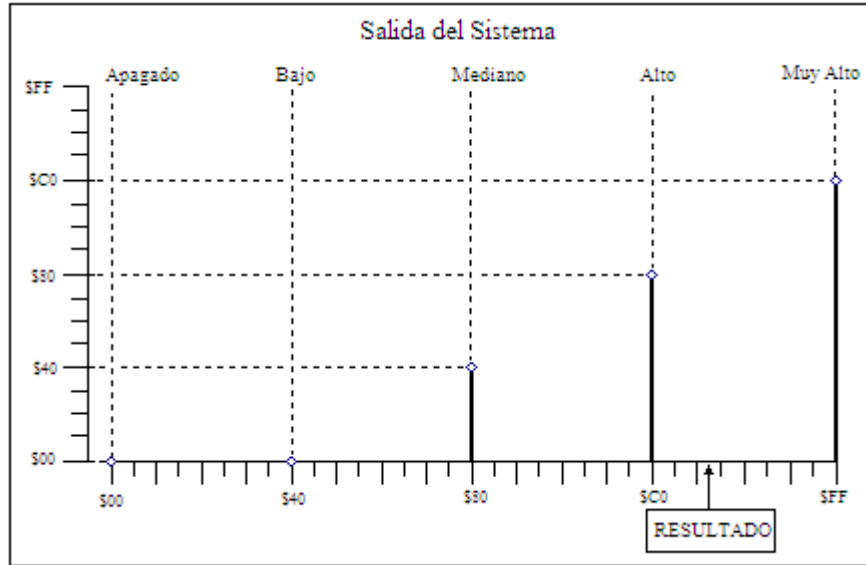


Figura 24. Desdifusión por método del centroide

Por lo tanto en el ejemplo de la Figura 24, el resultado o salida del sistema estará dado de la siguiente forma:

$$\begin{aligned}
 \text{Salida\_del\_sistema} &= \frac{\sum_{i=1}^n F_i \times S_i}{\sum_{i=1}^n F_i} \\
 &= \frac{(\$00 \times \$00) + (\$40 \times \$00) + (\$80 \times \$40) + (\$C0 \times \$80) + (\$FF \times \$C0)}{\$00 + \$00 + \$40 + \$80 + \$C0} \\
 &= \frac{\$0000 + \$0000 + \$2000 + \$6000 + \$BF40}{\$180} = \frac{\$13F40}{\$180} = D4
 \end{aligned}$$

### 3.8.4. Control difuso

Como se mencionó anteriormente en el caso del modelo hidráulico a escala reducida, la función del control difuso será el movimiento de las compuertas que definen el caudal de los afluentes que generan las avenidas del río, así como la compuerta que define el nivel de agua que representa la marea. Para lograr esto el programa que está instalado en la computadora personal solicitará mediante un

comando previamente definido el valor de cada uno de los parámetros mencionados de una forma periódica y los comparará con las directrices almacenadas en su base de datos que se deben reproducir en el modelo.

Si el valor actual de cada uno de los parámetros es el mismo que se debe tener de acuerdo a la directriz, entonces no se ejecuta ninguna acción y las compuertas permanecen en su lugar. Si por el contrario el valor es diferente se envía un comando a través del canal de comunicación al microprocesador, que indica el parámetro que debe modificarse y el valor meta que debe alcanzar.

El valor meta es comparado con el valor actual, mediante una resta para determinar que tan cerca o lejos se encuentra el valor actual del valor meta. Esta diferencia es el valor de entrada al sistema de control difuso. La Figura 25 muestra las funciones de membresía de las variables de entrada.

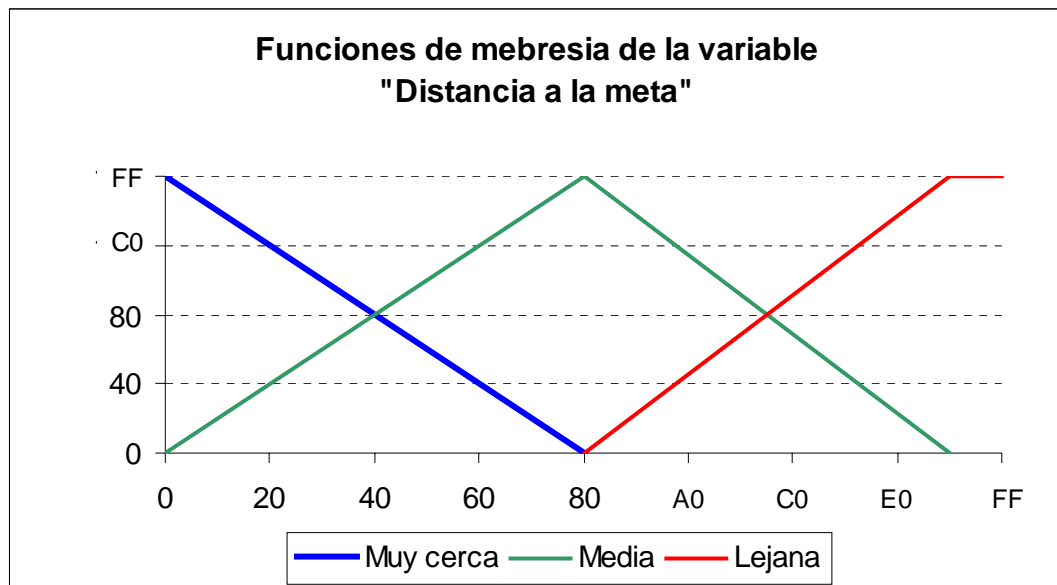


Figura 25. Funciones de membresía de las variables de entrada al sistema

Como se mencionó anteriormente, en la etapa de difusión la variable de entrada “distancia a la meta” es evaluada para cada una de sus etiquetas y los resultados son almacenados en memoria RAM.

A continuación se efectúa la evaluación de reglas. Las reglas de este sistema fueron definidas de la siguiente manera:

- Si la “distancia a la meta” es “muy cerca” entonces, la “velocidad del motor” es “muy lenta”.
- Si la “distancia a la meta” es “media”, entonces la “velocidad del motor” es “regular”.
- Si la “distancia a la meta” es “lejana”, entonces la “velocidad del motor” es “máxima”.

Una vez que se realiza la evaluación de las reglas se tendrá un valor para cada una de las salidas difusas, qué representa y en qué grado cada una de ellas aplica de acuerdo a las variables de entrada.

Las salidas difusas del sistema, en este caso, representan la velocidad del motor y estarán representadas por funciones tipo singleton, de acuerdo a lo siguiente:

Velocidad máxima en  $x = \$FF$

Velocidad regular en  $x = \$80$

Velocidad muy lenta en  $x = \$00$

Como se explicó la salida del sistema será la combinación de las salidas difusas, mediante la obtención del centroide.

La velocidad resultante aunque está dada por un valor hexadecimal entre \$00 y \$FF, representa la salida del modulador por ancho de pulso, es decir de 0% a un 100% y debe tomar en cuenta las limitaciones propias de esta rutina.



Es muy importante aclarar que existe también una salida implícita, que es la dirección de giro del motor que estará dada por el resultado de la comparación del valor actual con el valor meta.

### **3.9. Control de los motores.**

Una vez que el control difuso determina si un motor debe moverse, la velocidad con la que debe moverse y la dirección en la que debe moverse, entonces se deben generar las señales de control hacia los motores que produzcan los resultados esperados.

Como se mencionó anteriormente, la dirección de giro del motor será controlada por un circuito Puente H, por lo que se requiere de una señal de control, que indique al circuito el sentido de giro de los motores.

La velocidad de los motores es controlada a través de la modulación por ancho de pulso, de tal forma que cuando se tenga un ciclo de trabajo del 100%, se tendrá una alimentación continua del motor obteniendo la máxima velocidad. A su vez, un ciclo de trabajo de 0%, representa que no se aplicará alimentación en las terminales del motor y por lo tanto estará detenido, lo que representa la velocidad mínima. Por consiguiente, los ciclos de trabajo intermedios, producirán un valor de velocidad que se encontrará dentro de este rango.

Para producir una señal modulada por ancho de pulso, se utilizará otra función del sistema temporizador del microcontrolador: las salidas de comparación.

### 3.9.1. Salidas de Comparación

Como se mencionó en la función de Captura de entradas, el elemento central del sistema temporizador principal es el contador de 16 bits que corre libremente. Las funciones de salidas de comparación se utilizan para programar que ocurra una acción en un tiempo específico (cuando el contador de 16 bits alcanza un valor específico). Para cada función de salida de comparación, existe un registro de comparación de 16 bits y un comparador de 16 bits dedicado. El valor en el registro de comparación es comparado con el contador libre en cada ciclo. Cuando el registro de comparación coincide con el valor del contador, se genera una salida, que fija la bandera de estado de salida de comparación (OCxF) e inicia las acciones automáticas para esa función de salida de comparación. Las acciones automáticas iniciadas por una salida de comparación, incluyen solicitud de interrupción por hardware y cambio de estado en las terminales de salida relacionadas.

Los microcontroladores de la serie E de la familia del MC68HC11 cuentan con cuatro funciones de salidas de comparación y una adicional, que puede ser configurada, ya sea como la cuarta entrada de captura, o como una quinta salida de comparación.

Algunos de los usos comunes para esta función, son la generación de pulsos de una duración específica, así como la generación de retardos específicos.

Para producir un pulso de una duración específica, se debe escribir un valor en el registro de comparación de salida, que represente el tiempo en el que el flanco principal del pulso debe ocurrir. El circuito de comparación de salida es configurado para fijar la salida apropiada ya sea alta o baja, dependiendo de la polaridad del pulso que está siendo producido. Después de ocurrir la coincidencia, el registro de comparación de salida es reprogramado, para cambiar el estado en la terminal de salida de regreso a su nivel inactivo en la siguiente coincidencia. Un valor representando el ancho del pulso es agregado al valor original, y después escrito en el

registro de comparación de salida. Debido a que los cambios en el estado de la terminal ocurren en valores específicos del contador libre, el ancho de pulso puede ser controlado en forma precisa con la resolución del contador libre, independientemente de las latencias del software. Para generar una señal de salida a una frecuencia específica y con un ciclo de trabajo, se debe repetir este procedimiento de generación de pulsos.

Existen cuatro registros de comparación de salidas de lectura/escritura de 16 bits, TOC1, TOC2, TOC3 y TOC4, y el registro TI4/O5, el cual funciona bajo control de software ya sea como IC4 o OC5. Cada uno de los registros OC son fijados en \$FFFF durante el reset. Un valor escrito en un registro OC es comparado con el valor del contador libre durante cada ciclo de reloj E. Si se encuentra una coincidencia, la bandera particular de la salida de comparación es puesta en 1 en el registro de banderas de interrupción del temporizador 1 (TFLG1). Si esta interrupción en particular, está habilitada en el registro de máscaras de interrupción 1 (TMSK1), entonces se genera una interrupción. Adicionalmente a una interrupción, una acción específica puede iniciarse en uno o más pines de salida. Para OC[5:2], la acción del pin es controlada por pares de bits (OMx y OLx) en el registro TCTL1. La acción de salida es tomada en cada comparación exitosa, independientemente que la bandera OCxF del registro TFLG1 haya sido previamente borrada o no.

OC1 es diferente de otras salidas de comparación en cuanto a que una comparación exitosa de OC1 puede afectar cualquier pin o todos los cinco pines OC. La acción de salida OC1 tomada cuando se encuentra una coincidencia, es controlada por dos registros de 8-bit con tres bits sin implementar: el registro de máscara de salida de comparación, OC1M, y el registro de datos de salida de comparación 1, OC1D. OC1M especifica cuales salidas del puerto A serán utilizadas, y OC1D especifica qué datos son situados en esos pines del puerto.

### 3.9.2. Registros de Salidas de Comparación del Temporizador

Todos los registros de salidas de comparación son de lectura-escritura de 16 bits. Cada uno es inicializado a \$FFFF durante el reset. Si un registro de salidas de comparación no es utilizado por una función de salida de comparación, puede ser utilizado como una localidad de almacenamiento. Una escritura al byte de orden superior de un par de registros de salidas de comparación inhibe la función de salida de comparación por un ciclo de reloj. Esta inhibición previene comparaciones inapropiadas subsecuentes. Para que exista coherencia requiere de una lectura o escritura completa de 16 bit. De cualquier forma, si la coherencia no es necesaria, se puede utilizar el acceso al byte.

Para las funciones de salida de comparación, se debe escribir un valor de comparación en el registro de salidas de comparación TOC1 - TOC4 y TI4/O5. Cuando el valor de TCNT coincida con el valor de comparación, ocurrirán las acciones de las terminales específicas.

TOC1 - TOC3 - Timer Output Compare

\$1016	Bit 15	14	13	12	11	10	9	Bit 8	TOC1 (High)
\$1017	Bit 7	6	5	4	3	2	1	Bit 0	TOC1 (Low)

\$1018	Bit 15	14	13	12	11	10	9	Bit 8	TOC2 (High)
\$1019	Bit 7	6	5	4	3	2	1	Bit 0	TOC2 (Low)

\$101A	Bit 15	14	13	12	11	10	9	Bit 8	TIC3 (High)
\$101B	Bit 7	6	5	4	3	2	1	Bit 0	TIC3 (Low)

\$101C	Bit 15	14	13	12	11	10	9	Bit 8	TOC4 (High)
\$101D	Bit 7	6	5	4	3	2	1	Bit 0	TOC4 (Low)

Todos los registros TOCx son fijados en unos (\$FFFF) en el reset.

### 3.9.3. Registro de comparación forzada del temporizador

El registro CFORC permite comparaciones forzadas tempranas. FOC[1:5] corresponden a las cinco salidas de comparación. Estos bits se fijan en 1 para cada salida de comparación que va a ser forzada. La acción tomada como resultado de una comparación forzada, es la misma que si existiera una coincidencia entre el registro OCx y el contador libre, a excepción de que la bandera de interrupción correspondiente no esté fijada en uno. Los canales forzados disparan las acciones de las terminales programadas, para que ocurran en la siguiente transición de la cuenta del temporizador posterior a la escritura de CFORC.

Los bits de CFORC no deben utilizarse en una función de salida de comparación que esté programada para invertir su salida en una comparación exitosa, debido a que una comparación normal que ocurra inmediatamente antes o después de una comparación forzada, puede dar por resultado una operación no deseada.

CFORC- Timer Compare Force							\$100B
Bit7	6	5	4	3	2	1	Bit 0
FOC1	FOC2	FOC3	FOC4	FOC5	-	-	-
RESET: 0	0	0	0	0	0	0	0

#### FOC[1:5]- Force Output Comparison

Salida de Comparación Forzada. Cuando el bit FOC asociado es fijado en 1, el circuito de salida de comparación inmediatamente realiza la acción que está programada a efectuarse cuando ocurre una coincidencia.

0 = No es afectado

1 = Ocurre una acción de salida x

#### Bits[2:0]- Sin implementar

Siempre se leen en cero.

#### Registro de Máscara de Salida de Comparación

Utilice OC1M con OC1 para especificar los bits del puerto A que son afectados por una comparación exitosa OC1. Los bits del registro OC1M corresponden con PA[7:3].

**OC1M**- Output Compare 1 Mask

\$100C

	Bit7	6	5	4	3	2	1	Bit 0
	OC1M7	OC1M6	OC1M5	OC1M4	OC1M3	-	-	-
RESET:	0	0	0	0	0	0	0	0

OC1M[7:3]- Output Compare Masks

0 = OC1 es deshabilitado

1 = OC1 es habilitado para controlar el pin correspondiente del puerto A

Bits[2:0]- Sin implementar

Siempre se leen en cero.

### 3.9.4. Registro de Datos de Salida de Comparación

Este registro se utiliza con OC1 para especificar los datos que serán almacenados en las terminales afectadas del puerto A después de una comparación OC1 exitosa. Cuando una comparación exitosa OC1 ocurre, un bit de datos en OC1D es almacenado en la terminal correspondiente del puerto A para cada bit cuyo valor es 1 en OC1M.

**OC1D**- Output Compare 1 Data

\$100D

	Bit7	6	5	4	3	2	1	Bit 0
	OC1D7	OC1D6	OC1D5	OC1D4	OC1D3	-	-	-
RESET:	0	0	0	0	0	0	0	0

Si OC1M<sub>x</sub> es 1, los datos en OCD<sub>x</sub> es enviado a la salida del puerto A bit x en una comparación exitosa  
OC1

Bits[2:0]- Sin implementar

Siempre se leen en cero.

3.9.5. Registro contador del temporizador.

El registro de 16 bit de sólo lectura TCNT contiene el valor pre-escalado del temporizador de 16 bit. Una lectura completa direcciona primero el byte más significativo (MSB). Una lectura de esta dirección ocasiona que el byte menos significativo (LSB) sea guardado en un buffer para el siguiente ciclo del CPU, de tal forma que una lectura de byte doble regrese el estado completo de 16 bits del contador en el tiempo del ciclo de lectura del MSB.

TCNT- Timer Counter

\$100E	Bit 15	14	13	12	11	10	9	Bit 8	TCNT(High)
\$100F	Bit 7	6	5	4	3	2	1	Bit 0	TCNT(Low)

TCNT es \$0000 en el reset. En los modos normales, TCNT es de sólo lectura.

Registro de control del Temporizador 1

TCLT1-Timer Control 1

\$1020

	Bit7	6	5	4	3	2	1	Bit 0
	OM2	OL2	OM3	OL3	OM4	OL4	OM5	OL5
RESET:	0	0	0	0	0	0	0	0

OM[2:5] - Output Mode

OL[2:5] - Output Level

Estos pares de bits de control están codificados para tomar una acción específica después de una comparación exitosa OCx. OC5 funciona únicamente si el bit I4/O5 en el registro PACTL es cero.

OMx	OLx	Acción Tomada en una Comparación exitosa
0	0	Timer es desconectado de la lógica del pin de salida
0	1	Inversión de la línea de salida OCx
1	0	Borra la línea de salida OCx a cero
1	1	Fija la línea de salida OCx a uno

Tabla 8. Tabla Acciones de Salida de Comparación del Temporizador

### 3.9.6. Registro de Máscara de Interrupción del Temporizador 1

Este registro de 8 bits se utiliza para permitir o inhibir las interrupciones de las capturas de entradas del temporizador y las salidas de comparación.

**TMSK1**-Timer Interrupt Mask Register 1 \$1022

	Bit7	6	5	4	3	2	1	Bit 0
	OC1I	OC2I	OC3I	OC4I	I4/O5I	IC1I	IC2I	IC3I
RESET:	0	0	0	0	0	0	0	0

**OC1I - OC4 I - Output Compare x Interrupt Enable**

Si el bit OCxI está fijado en 1 cuando la bandera OCxF es 1, una secuencia de interrupción es solicitada.

**I4/O5I - Input Capture 4/Output Compare 5 Interrupt Enable**

Cuando I4/O5 en PACTL es uno, I4/O5I es el bit de interrupción habilitado de la entrada de captura 4. Cuando I4/O5 en PACTL es cero, I4/O5I es el bit de interrupción habilitado de la salida de comparación 5.

**IC1I - IC3CI - Input Capture x Interrupt Enable**

Si el bit ICxI es fijado en uno cuando la bandera ICxF es uno, una secuencia de interrupción por hardware es solicitada.

### 3.9.7. Registro de Bandera de Interrupción del temporizador 1

Los bits en este registro indican cuando han ocurrido eventos del sistema temporizador. Acoplado con los bits del registro TMSK1, los bits de TFLG1 permiten al subsistema del temporizador operar, ya sea en un sistema conducido por conteo o por interrupción. Cada bit en TFLG1 corresponde con un bit en el TMSK1 en la misma posición.

**TFLG1**-Timer Interrupt Flag 1 \$1023

	Bit7	6	5	4	3	2	1	Bit 0
	OC1F	OC2F	OC3F	OC4F	I4/O5F	IC1F	IC2F	IC3F
RESET:	0	0	0	0	0	0	0	0



OC1F - OC4F - Output Compare x Flag

Es fijado en uno cada vez que el contador coincide con el valor de salida de comparación x

I4/O5F - Input Capture 4/Output Compare 5 Flag

Se fija en uno por IC4 o OC5, dependiendo de la función habilitada por el bit I4/O5 de PACTL

IC1F - IC3CF - Input Capture x Flag

Es fijado en uno cada vez que un flanco seleccionado es detectado en la línea de entrada ICx

### 3.9.8. Registro de Máscara de Interrupción del Temporizador 2

Este registro de 8 bits se utiliza para habilitar o inhibir las interrupciones de desbordamiento del temporizador y la interrupción en tiempo real. Los bits de control del pre-escalador están incluidos en este registro.

**TMSK2** - Timer Interrupt Mask Register 2

\$1024

	Bit7	6	5	4	3	2	1	Bit 0
	TOI	RTII	PAOVI	PAII	-	-	PR1	PR0
RESET:	0	0	0	0	0	0	0	0

TOI - Timer Overflow Interrupt Enable

0 = interrupciones TOF deshabilitadas

1 = interrupción es solicitada cuando TOF es uno.

RTII = Real Time Interrupt Enable

Habilitar Interrupción en tiempo real.

PAOVI - Pulse Accumulator Overflow Interrupt Enable

Habilitar Interrupción del desbordamiento del acumulador de pulsos.

PAII - Pulse Accumulator Input Edge Interrupt Enable

Habilitar interrupción del flanco de entrada del acumulador.

Bits [3:2] - Sin implementar

Siempre se leen en cero.

PR[1:0] - Timer Prescaler Select

Estos bits son usados para seleccionar tasa de divisor del pre-escalador. En modos normales de operación, PR[1:0] puede ser escrito sólo una vez, y la escritura debe ser dentro de los primeros 64 ciclos después del reset.

PR[1:0]	Preescalador
0 0	1
0 1	4
1 0	8
1 1	16

Los bits del registro TMSK2 corresponden bit por bit con los bits de banderas en TFLG2. Los unos en TMSK2 habilitan las fuentes de interrupción correspondientes.

**Tabla 9. Pre-escalador del Temporizador**

### 3.9.9. Registro de Banderas de Interrupción del Temporizador 2

Los bits en este registro indican cuando han ocurrido ciertos eventos del sistema del temporizador. Acoplado con los cuatro bits de orden superior de TMSK2, los bits de TFLG2 permiten al subsistema del temporizador operar, ya sea en un sistema conducido por muestreo o por interrupción. Cada bit en TFLG2 corresponde con un bit en el TMSK1 en la misma posición.

**TFLG2** - Timer Interrupt Flag 2

\$1025

	Bit7	6	5	4	3	2	1	Bit 0
	TOF	RTIF	PAOVF	PAIF	-	-	-	-
RESET:	0	0	0	0	0	0	0	0

Limpie las banderas escribiendo un uno en las posiciones de bit correspondientes.

**TOF** - Timer Overflow Interrupt Flag

Es fijada en uno cuando TCNT cambia de \$FFFF a \$0000

**RTIF** - Real Time (Periodic) Interrupt Flag

Bandera de interrupción de tiempo real.

**PAOVF**- Pulse Accumulator Overflow Interrupt Flag

Bandera de interrupción del desbordamiento del acumulador de pulsos.

**PAIF** - Pulse Accumulator Input Edge Interrupt Flag

Bandera de interrupción del flanco de entrada del acumulador de pulsos.

Bits [3:0] - Sin implementar

Siempre se leen en cero

### 3.9.10. Implementación de la modulación por Ancho de pulso

En el siguiente listado se muestra la rutina de servicio para la creación de las salidas moduladas por ancho de pulso (PWM).

```

LDX      #REGINI      Apunta al inicio del bloque de registros
LDAA     #%11111100   OMx:OLx = 1:1 cambia a nivel alto
STAA     TCTL1,X      OC2, OC3 y OC4 = 1 en comparación exitosa
LDAA     #%01110000   OC1M6,5, & 4 = 1
STAA     OC1M,X       Para controlar OC2/PA6, OC3/PA5, & OC4/PA4
LDAA     #%00000000   OC1 cambia nivel bajo, OC2,OC3 y OC4 a alto
STAB     OC1D,X       Fija los valores para OC1D
LDAA     #10000       Establece duración del periodo en no de ciclos
STD      PERIODO      Almacena el periodo
STD      TOC1,X       Establece el primer periodo PWM en TCNT=PWMPER
STD      TOC2,X       Programa el primer OC2
STD      TOC3,X       Programa el primer OC3
STD      TOC3,X       Programa el primer OC4
* Termina inicialización
LDAA     #$80
STAA     TFLG1,X      OC1F=1 para borrar cualquier bandera vieja OC1
STAA     TMSK1,X      entonces OC1I=1 habilita la interrupción OC1
CLI

```

\*\*\*

\* RSOC1 - Rutina de Servicio de la Salida de Comparación 1

\*\*\*

```

RSOC1     LDX      #REGINI
           Apunta al inicio del bloque de registros
LDD      TOC1,X      Obtiene el último valor de comparación OC1
ADDD     PERIODO     Agrega el periodo al valor
STD      TOC1,X      Actualiza OC1 (programa el siguiente OC1)
LDD      TOC1,X      Obtiene el nuevo valor de comparación OC1
SUBD     PULS01      Obtiene el nuevo valor de OC2
STD      TOC2,X      Actualiza OC1 (programa el siguiente OC2)
LDD      TOC1,X      Obtiene el nuevo valor de comparación OC1
SUBD     PULS01      Obtiene el nuevo valor de OC3
STD      TOC2,X      Actualiza OC1 (programa el siguiente OC3)
LDD      TOC1,X      Obtiene el nuevo valor de comparación OC1
SUBD     PULS01      Obtiene el nuevo valor de OC4
STD      TOC2,X      Actualiza OC1 (programa el siguiente OC4)
BCLR     TFLG1,X $7F Borra la bandera OC1F
RTI      ** Regresa de la rutina de servicio OC1 **

```

En este programa se utiliza la opción de salida de comparación 1 conjuntamente con las salidas OC2, OC3 y OC4, para generar la señal modulada por ancho de pulso. OC1, cambia las salidas OC2, OC3 y OC4 a un nivel bajo, y genera interrupción en la cual programa la siguiente interrupción de OC1 y programa los valores de comparación de OC2, OC3 y OC4. Cuando ocurre una comparación

exitosa de OC2, OC3 y OC4 no se genera interrupción, pero la salida correspondiente cambia a un nivel alto. Se programó un periodo de 10,000 ciclos que cuando  $E = 2\text{MHz}$  corresponde a una frecuencia de 200Hz.

Debido a que la rutina de interrupción programa los valores de comparación de OC2, OC3 y OC4 después de una comparación exitosa de OC1, éstos no pueden programarse con un valor muy cercano al valor de OC1, debido a que este evento puede ocurrir antes de que pueda ser programado, por lo que no se podrán tener ciclos de trabajo del 100% o se podrían obtener resultados no deseados. La rutina de interrupción requiere de 72 ciclos para ser completada, por lo que habría tiempo suficiente para programar un ciclo de trabajo del 99%, sin embargo, deberá considerarse el fenómeno de latencia para las otras rutinas de interrupción del sistema.

### **3.10. Circuitos de protección**

Los circuitos de protección están pensados para evitar daños al sistema debidos a alguna falla en el mismo, para lo cual se implantaron dos tipos de protecciones, la primera es una protección por hardware que está dada por interruptores que se encuentran en serie con el circuito de alimentación en el puente H. Este sistema se explica con mayor detalle en el capítulo 1. La otra protección es por software y consiste en un circuito formado por un sensor óptico, que al activarse envía una señal de interrupción al microcontrolador, para detener todos los motores.

En este caso, todos los sensores de protección por software que se envían desde todas las compuertas, deben configurarse en un arreglo tipo OR y conectarse a la terminal XIRQ del microprocesador que es una terminal de interrupción no mascarable.

Cuando el microprocesador reconoce la fuente de interrupción, completa la instrucción que está siendo ejecutada en ese momento, posteriormente almacena

todos los registros del CPU, después almacena el valor del CCR y debido a que XIRQ queda pendiente, se fija en 1 el bit X para inhibir más interrupciones. Entonces el vector de interrupción para la fuente pendiente con mayor prioridad es traída, y la ejecución continúa en la dirección especificada por el vector. Al final del servicio de la rutina de interrupción, la instrucción de retorno de interrupción es ejecutada y los registros salvados son jalados desde la pila en orden inverso, de tal forma que la ejecución normal del programa puede continuar.

Localidad de Memoria	Registros del CPU
SP	PCL
SP – 1	PCH
SP – 2	IYL
SP – 3	IYH
SP – 4	IXL
SP – 5	IXH
SP – 6	CCR

**Tabla 10. Orden de Apilamiento o Entrada a Interrupciones**

### 3.10.1. *Solicitud de Interrupción no mascarable (XIRQ)*

Las interrupciones no mascarables son muy útiles, debido a que siempre pueden interrumpir las operaciones del CPU. El uso más común para este tipo de interrupción es para problemas serios del sistema, tales como un programa fuera de control o una falla por pérdida de energía. La entrada XIRQ es una versión actualizada de NMI (interrupción no mascarada) de los MCUs previos.

Durante el reset, ambos, el bit X y el bit I del CCR, son fijados para inhibir todas las interrupciones mascarables y XIRQ. Después de una mínima inicialización del sistema, el software puede borrar el bit X por medio de la instrucción TAP, habilitando interrupciones X. A partir de ese momento el software no puede fijar nuevamente el bit X. Por tal motivo una interrupción XIRQ es una interrupción no mascarable. Debido a la estructura de operación de las interrupciones relacionadas con el bit I, no tiene efecto en el bit X, el pin interno XIRQ que permanece no mascarado. En la lógica de

prioridad de interrupción, la interrupción de XIRQ tiene prioridad mayor que cualquier otra fuente de interrupción, que es mascarable por el bit I. Todas las interrupciones relacionadas con el bit I operan normalmente, con su propia relación de prioridad.

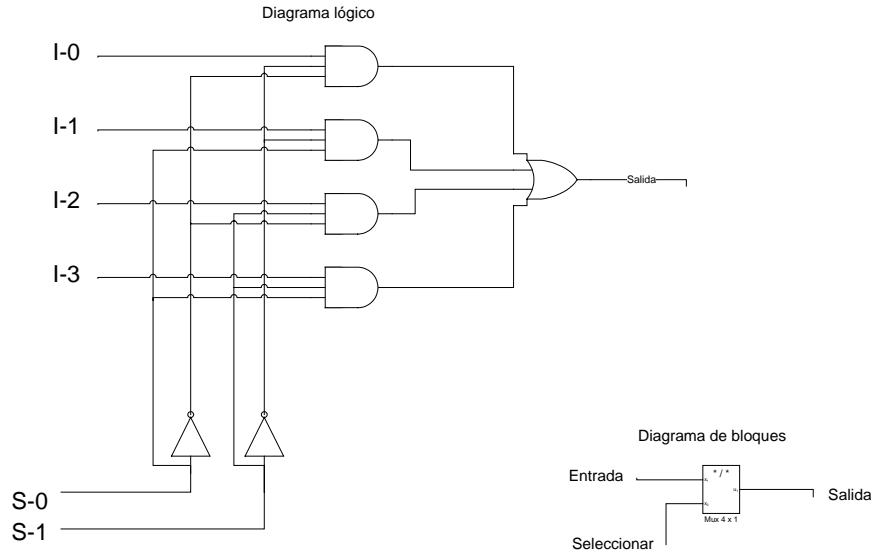
Cuando una interrupción relacionada con el bit I ocurre, el bit I es automáticamente fijado por hardware después de apilar el byte CCR. El bit X no es afectado. Cuando ocurre la interrupción relacionada con el bit X, ambos, el bit X y el I, son automáticamente fijados por hardware, después de apilar el byte CCR. Una instrucción de retorno, desde interrupción restablece los bits X e I a su estado previo a la solicitud de interrupción.

Si alguno de los sensores de protección por software es activado, debido a que alguna de las compuertas ha llegado a su fin de carrera, entonces causará que se ejecute la rutina de interrupción correspondiente, la cual establecerá todas las salidas de comparación del temporizador en un nivel bajo, deteniendo así la alimentación a todos los motores. En este punto también es recomendable que envíe una señal de error, que puede incluso ser auditiva, indicando que se requiere de intervención humana para solucionar el problema.

### **3.11. Multiplexor.**

Debido a que existen múltiples señales de entrada provenientes de los diferentes sensores, y al número limitado de puertos de entrada, se hace necesario el uso de los multiplexores.

Un multiplexor es una función digital que recibe información binaria de varias líneas y transmite información en una sola línea de salida. La única línea de entrada que es seleccionada, se determina a partir de combinación de bits de las  $2^n$  líneas seleccionadas. Un ejemplo de un multiplexor 4 x 1 se muestra en la Figura 26.



**Figura 26. Diagrama de multiplexor**

Las cuatro líneas de entrada son aplicadas a cuatro compuertas AND cuyas salidas van a una sola compuerta OR. Solamente una línea de entrada tiene una ruta a la salida en cualquier instante particular. Las líneas de selección S1 y S0 determinan cual de las entradas se selecciona, para formar una ruta directa hacia la salida. Así, con S1 y S0 = 10, la compuerta AND asociada con la entrada I2 tiene dos de sus entradas iguales a 1. Si I1 = 1 la salida de la compuerta es también igual a 1; Si I2 = 0, la salida es igual a 0.

Todas las otras compuertas AND tienen la salida en 0. La salida de la compuerta OR será igual al valor de la salida de la compuerta AND asociada con la entrada I2. Un multiplexor también se le denomina selector de datos, puesto que selecciona una de las múltiples líneas de entrada de datos, y dirige la información binaria a la línea de salida.

Las compuertas AND en un multiplexor, se parecen a un circuito decodificador, y en realidad, ellas decodifican las líneas de selección de entrada. En general, un circuito multiplexor se construye a partir de un decodificador  $n$  por  $2n$ , añadiendo una línea de entrada a cada compuerta AND y aplicando todas las salidas de las compuertas AND a una sola compuerta OR. El tamaño de un multiplexor es

especificado por el número de sus entradas,  $2^n$ . Esto, entonces implica que él también contiene una línea de salida, y  $n$  líneas de selección. Un multiplexor puede tener una o más líneas de habilitación, como un decodificador. Las líneas de habilitación pueden utilizarse para expandir dos o más multiplexores  $C_i$  a un multiplexor digital con un mayor número de entradas.

### 3.11.1. *Salidas paralelas del Microcontrolador*

Las señales de control para el multiplexor, que servirán para seleccionar la señal que va a ser muestreada, se realizarán a través del puerto B del microcontrolador, el cual es un puerto de salida de propósito general de 8 bits, de dirección fija. Las escrituras al registro del puerto B (PORTB) causan que los datos sean enclavados (latched) y llevados a las terminales de salida del puerto B. Las lecturas a este puerto regresan los últimos datos que fueron escritos al puerto B.

### 3.11.2. *Multiplexión en tiempo*

Existe también el multiplexado en tiempo el cual transmite todas las señales de entrada alternativamente, por una misma línea de salida a intervalos definidos de tiempo lo cual elimina la necesidad de tener líneas de control para seleccionar una señal en particular, pero agrega la necesidad de contar con un sistema de medición de tiempo, muy exacto para reconocer las señal, que está siendo transmitida en un momento en particular.

## 3.12. Desarrollo del Programa de Control del Microprocesador

Antes de comenzar a escribir el programa del microcontrolador, es necesario comprender todas y cada una de las funciones que se deben realizar, y presentar esta información en pequeños segmentos de operación, agrupadas entre sí en un diagrama de bloques que presente la operación general que realizará el controlador. Las funciones que en un momento dado sean repetitivas, o que se ejecuten como



resultado de una decisión, pueden agruparse en bloques llamados subrutinas. En otras ocasiones el microcontrolador puede ser interrumpido de la secuencia continua del programa, para realizar ciertas tareas en tiempo real llamadas rutinas de interrupción. Tanto para las subrutinas como para las rutinas de interrupción, es importante realizar también sus respectivos diagramas de flujo.

### 3.12.1. *Funciones principales del programa del microcontrolador*

Las funciones principales del programa están divididas en rutinas y subrutinas. Las principales rutinas del programa son las siguientes:

1. Rutina para la medición de frecuencia utilizando las entradas de captura del sistema temporizador. Se utiliza para determinar las variables de nivel y flujo.
2. Rutina de conversión analógico-digital para determinar la posición de las compuertas.
3. Rutina para generar tres señales de modulación por ancho de pulso para el control de la velocidad de los motores que mueven las compuertas.
4. Un conjunto de salidas debe utilizarse para activar un multiplexor que seleccionará entre muchas señales provenientes de los sensores, la que debe entrar al puerto donde se mide la frecuencia.
5. Se debe realizar un código de comandos, al estilo de los comandos del programa monitor, para el intercambio de información entre el microcontrolador y la computadora personal. De tal forma que, una vez establecida la comunicación entre ambos el control sea tomado desde la PC, para solicitar datos de los sensores de nivel, así como indicar el movimiento de las compuertas. Entre los comandos que debe enviar la PC pueden estar los siguientes.

- a. Consultar nivel del sensor x.
- b. Obtener todos los niveles.
- c. Mover la compuerta n a la posición y (o fijar objetivo de avenida1 a x y Marea a y).

6. Otra rutina es utilizada para servir a la interrupción generada por el circuito de protección de las compuertas para detener el movimiento de los motores

El diagrama de flujo utilizado para el programa principal se muestra en la Figura 27. Éste consiste en la inicialización de las variables, los registros y los vectores de interrupción. Una vez que ha terminado la etapa de inicialización se habilita el puerto serial para comenzar a recibir los comandos de control desde la computadora personal, por lo cual, el programa entra en un ciclo infinito hasta recibir alguno de los comandos, o hasta atender alguna fuente de interrupción.

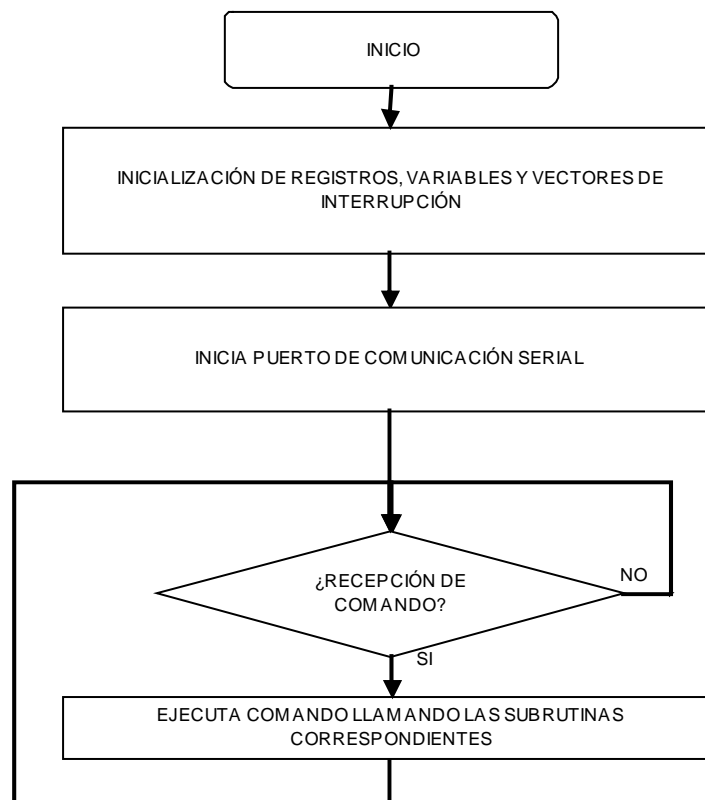


Figura 27. Diagrama de flujo del programa principal

Por tanto, es necesario realizar un diagrama de flujo para cada una de las subrutinas. Una vez que se han determinado los diferentes diagramas de flujo, tanto para el programa principal como para las subrutinas, se puede empezar a escribir el código asociado con cada uno de los bloques del diagrama. Sin embargo, antes de escribir propiamente el código de los programas, se pueden escribir otro tipo de instrucciones que no forman parte del conjunto de instrucciones del microcontrolador, pero que instruyen al programa ensamblador a realizar ciertas tareas, por nosotros, al momento de ensamblar.

Entre estas instrucciones se encuentra la instrucción EQU (equate), la cual puede utilizarse, por ejemplo, para asignar una etiqueta o una dirección de memoria. Tal es el caso de la siguiente instrucción:

PORTA EQU 100

Esto es muy útil en el desarrollo del programa, ya que es más fácil de recordar la etiqueta, que la dirección de los puertos y registros que tiene el controlador. Y por otro lado, es más fácil seguir o leer la secuencia del programa cuando existen estas etiquetas. que si se tuvieran simplemente números hexadecimales.

Por otro lado, se pueden utilizar instrucciones para definir las variables a utilizar durante la ejecución del programa, y al mismo tiempo reservar espacio para ellas en la memoria RAM, de tal forma que no puedan sobrescribirse por error durante la ejecución del programa.

### 3.12.2. *El módulo de evaluación y el programa monitor*

Con un módulo de evaluación se puede descargar el programa a utilizar, y evaluar paso a paso la ejecución del programa para encontrar posibles errores. Esto se realiza a través de un programa monitor llamado BUFFALO (bit user fast friendly

aid to logical operations) que se encuentra en la memoria del microcontrolador. La transmisión y el control de los datos hacia el microcontrolador se realizan a través de un puerto RS232

Al momento del reset la terminal PE0 es evaluada, si se detecta un estado bajo, se ejecuta el programa monitor, si se detecta un estado alto, el programa de usuario es ejecutado automáticamente en la dirección \$B600

La programación se hace en el lenguaje ensamblador del microcontrolador usando los comandos propios del HC11 (Ver Anexo xx) y se guardan en un archivo con extensión “.asm”, después es necesario utilizar un programa para compilar el programa y generar un archivo en lenguaje de máquina que tiene extensión S19.

El programa es descargado transmitiendo caracteres tipo ASCII desde la PC al microcontrolador. La descarga de los programas requiere de una comunicación a baja velocidad (300 bauds), ya que tanto en el borrado, como la descarga de esta información necesita de, al menos, 10 milisegundos entre caracteres. Es recomendable realizar un borrado de la memoria EEPROM donde se va a grabar el programa. Para esto se pueden utilizar los comandos BULK o BULKALL.

Para cambiar la velocidad es necesario modificar el registro 102B de 30 (9600 bauds) a 35 (300 bauds). Esto se puede hacer utilizando el comando Memory Modify (MM) de la siguiente manera

```
>MM 102B <CR>
```

Al introducir este comando y oprimir la tecla Enter aparecerá

```
102B 30
```

A continuación se debe escribir el nuevo valor, en este caso 35, y oprimir la tecla Enter. Al hacer esto la velocidad del microcontrolador se cambiará a 300 bauds pero la del programa de la PC continúa en 9600, por lo que algunos símbolos raros pueden aparecer en la pantalla. Por esta razón, ahora es necesario cambiar también este parámetro en la PC.

Una vez sincronizada la velocidad, se debe oprimir la tecla Enter para acceder nuevamente a la línea de comando. A continuación se prepara al microcontrolador para recibir el programa utilizando el siguiente comando:

```
>LOAD T <CR>
```

Y se envía el programa desde la computadora personal utilizando el protocolo de transferencia de archivos ASCII. En el caso del programa IASM11 se ejecuta la función F6 "Download" y se indica el archivo con extensión S19 que se desea descargar.

Cuando se ha descargado el archivo S19, en la pantalla de la terminal se desplegará el mensaje: done. Una vez realizado esto, se debe reiniciar el microcontrolador mediante el switch SW1. Al reiniciarse la velocidad de transmisión regresa a 9600 bauds, por lo que es necesario cambiar nuevamente este parámetro en la aplicación de la PC. Una vez que se muestra el "prompt", en la pantalla de la terminal, se puede probar el programa descargado en el microcontrolador.

Esto se puede hacer con el comando EXECUTE PROGRAM o GO y tomando en cuenta que el programa se descarga en la dirección B600 se puede realizar de la siguiente forma:

```
>G B600
```

o realizar la ejecución paso a paso o un número determinado de Pasos utilizando la función TRACE

>T FF

El programa Monitor (BUFFALO) contiene subrutinas de utilerías, que pueden ser llamadas para utilizarse en el programa del usuario, algunas de éstas son las que tienen que ver con las comunicaciones del puerto de entradas y salidas (I/O). Existe una tabla que se encuentra en la memoria ROM antes de los vectores de interrupción. Se pueden utilizar estas subrutinas ejecutando la instrucción JSR (jump to subroutine) a la dirección adecuada, como se indica en la tabla 11.

Dirección	Rutina	Descripción
\$FF7C	.WARMST	Va al punto del prompt ">" (salta el mensaje BUFFALO...).
\$FF7F	.BPCLR	Borra la tabla de los puntos de paro.
\$FF82	.RPRINT	Despliega los registros de usuario.
\$FF85	.HEXBIN	Convierte un carácter ASCII en el registro A, en un número de 4 bits. Recorre el número en SHFTREG desde la derecha. SHFTREG es registro de 2 byte (4 dígitos hexadecimales). Si el registro A no es hexadecimal, TMP1 es incrementado y SHFTREG permanece sin cambios.
\$FF88	.BUFFAR	Lee un argumento 4 dígitos de la memoria de entrada a SHFTREG.
\$FF8B	.TERMAR	Lee un argumento 4 dígitos de la terminal del dispositivo a SHFTREG.
\$FF8E	.CHGBYT	Escribe el valor (si hubiere) de SHFTREG+1 a la localidad de memoria apuntada por X. (La operación sólo aplica para localidades de EEPROM).
\$FF91	.READBU	Lee el siguiente caracter de INBUFF.
\$FF94	.INCBUF	Incrementa el apuntador de la memoria de entrada intermedia.
\$FF97	.DECBUF	Decrementa el apuntador de la memoria de entrada intermedia.
\$FF9A	.WSKIP	Lee la memoria de entrada intermedia hasta que encuentra un carácter diferente a espacio en blanco.
\$FF9D	.CHKABR	Monitorea la entrada para solicitudes (CTRL)X, (DELETE), o (CTRL)W. Si el caracter en el acumulador A es minúscula, lo convierte en mayúscula.
\$FFA0	.UPCASE	Revisa el caracter en el acumulador A y fija en 1 el bit Zes un carácter de espacio en blanco (espacio, coma, tabulador).
\$FFA3	.WCHEK	Revisa el carácter en el acumulador A y fija en 1 el bit Zes un carácter delimitador (retorno de carro o espacio en blanco).
\$FFA6	.DCHEK	

Dirección	Rutina	Descripción
\$FFA9	.INIT	Inicializa el dispositivo de entradas/salidas.
\$FFAC	.INPUT	Lee el dispositivo de entrada/salida.
\$FFAF	.OUTPUT	Escribe el dispositivo de entrada/salida.
\$FFB2	.OUTLHL	Convierte el contenido del nibble izquierdo del acumulador A a ASCII y le da salida por el puerto terminal.
\$FFB5	.OUTRHL	Convierte el contenido del nibble derecho del acumulador A a ASCII y le da salida por el puerto terminal.
\$FFB8	.OUTA	Da salida al carácter ASCII del acumulador A.
\$FFBB	.OUT1BY	Convierte un byte binario en la dirección de registro de índice X en dos caracteres ASCII y les da salida. Regresa la dirección en el registro de índice X apuntando al siguiente byte.
\$FFBE	.OUT1BS	Convierte un byte binario en la dirección de registro de índice X en dos caracteres ASCII y les da salida seguidos de un espacio. Regresa la dirección en el registro de índice X apuntando al siguiente byte.
\$FFC1	.OUT2BS	Convierte dos bytes binarios consecutivos comenzando en la dirección de registro de índice X en cuatro caracteres ASCII y les da salida seguidos de un espacio. Regresa la dirección en el registro de índice X apuntando al siguiente byte.
\$FFC4	.OUTCRL	Da salida a un carácter de retorno de carro seguido por uno de alimentación de línea.
\$FFC7	.OUTSTR	Da salida a una cadena de caracteres apuntadas por la dirección en el registro de índice X hasta que el carácter es el de fin de transmisión (\$04)
\$FFCA	.OUTST0	Igual que OUTSTR a excepción que se salta el carácter de retorno de carro y alimentación de línea.
\$FFCD	.INCHAR	Ingresa un carácter ASCII al acumulador A y lo repite hasta que el carácter es recibido.
\$FFD0	.VECINIT	Utilizado durante la inicialización para fijar vector de interrupción indirecto en el área RAM. Esta rutina o una rutina similar debe incluirse en el programa de usuario que es invocada por la función de salto a la dirección \$B600 del BUFFALO.

**Tabla 11. Tabla de saltos a las Subrutinas**

**Conclusiones**

**Conclusiones**



Una parte muy importante del generador de mareas, son las entradas del sistema que provienen de los diferentes transductores, por lo que, para obtener los mejores resultados las mediciones deberán apearse, lo más posible, a los valores reales de las variables.

Para la medición de nivel se ha definido el uso de sensores capacitivos, sin embargo, deberá comprobarse en la práctica, la conveniencia de su utilización y evaluar en qué forma afectan algunos factores, entre los que se encuentran la histéresis debida a la tensión superficial del agua sobre el sensor, o la variación en la salinidad del agua conforme ésta se mezcle con agua de mar.

En un principio se está considerando que todos los sensores del mismo tipo son exactamente iguales, aunque esto en la práctica puede ser diferente, en tal caso, se puede caracterizar cada sensor de manera independiente.

Por otro lado, deberá definirse la resolución que deberán tener cada una de las mediciones y el margen de error que estará permitido dentro del sistema, y en consecuencia, tomar las medidas necesarias. Esto es importante debido a que los sensores basados en frecuencia, tienen la ventaja de poder medir periodos en base al número de ciclos de reloj permitiendo una resolución muy alta, y por el contrario los sensores de posición basados en elementos resistivos, la resolución está dada por el convertidor A/D de 8 bits. En contraparte, debe considerarse que el control difuso en este caso, está considerando entradas difusas de 8 bits, por lo que todas las entradas por muy alta resolución que tengan, deberán escalarse para ajustarse al control difuso, o en su defecto, establecer la lógica para manejar entradas difusas con un mayor número de bits, lo cual incrementará la complejidad de los cálculos y el tiempo de ejecución.

La lógica difusa se ha convertido en una herramienta muy útil en los sistemas de control, debido a la característica de poder aplicar el conocimiento adquirido por

medio de la experiencia, sin necesidad de funciones complejas que requieran de mucho tiempo y recursos de procesamiento.

Debido a que tanto las funciones de membresía, como las reglas y las salidas difusas, fueron definidas sin consultar a un experto en modelos hidráulicos a escala, éstos deberán ajustarse al modelo real, cuando éste se encuentre en la etapa de implementación, para apegarse a los resultados esperados.

Las reglas definidas parecen muy sencillas, debido a que a cada consecuente corresponde un solo antecedente, es decir, para definir el ciclo de trabajo del motor (velocidad), únicamente se toma en cuenta la distancia a la meta, pero se puede agregar otro factor, como la velocidad lineal a la cual se aproxima a la meta para seguir lo más fiel posible, la directriz de las mareas y avenidas.

La ventaja de implementar el control en un sistema de evaluación del microprocesador, es que las variables definidas pueden modificarse con relativa facilidad, hasta obtener los resultados deseados al momento de las pruebas.

Además del control de las compuertas por parte del microcontrolador, éste realizará un muestreo de las señales provenientes de los diferentes sensores repartidos a lo largo del modelo. Al ejecutar las pruebas, es importante tomar en cuenta los periodos de tiempo latentes debido a las diferentes rutinas y fuentes de interrupción, de tal forma que, aún en el peor de los casos, estos permitan cumplir con el periodo de muestreo que se han de fijar para estas señales.

## Bibliografía

- Diccionario Enciclopédico UTEHA. 1951.
- Mecánica de fluidos y Máquinas Hidráulicas, Claudio Mataix, Segunda edición. Editorial Harla.
- Sensores y acondicionadores de señal, Ramón Pallas Areny., 2ª De. Marcombo.
- Métodos experimentales para ingenieros, Jack P. Holman, 5ª edición, Mc Graw Hill.
- HC11 M68HC11 E Series Technical Data, Gene Bates, Motorola, Inc, 1995
- Fuzzy Logic for Embedded Microcontrolers, Jim Sibigroth, Circuit Cellar Ink, Issue 56 March 1995,
- Implementing Fuzzy Expert Rules in Hardware, James M. Sibigroth, AI Expert, April 1992.
- Técnicas de modelación en hidráulica, Miguel A. Vergara H., Alfaomega Grupo Editor S. A de C. V., 1995.
- Sensores y analizadores, Harry N. Norton, Colección Electrónica/Informática, Ed. Gustavo Gili S. A., 1982, Prentice Hall Inc.
- M68HC11 Reference Manual, Motorola, 2001.

## Anexo A. Índice de Tablas y Figuras

### Índice de Figuras

Figura 1. Diagrama General del Sistema .....	4
Figura 2. Compuertas tipo guillotina de desplazamiento vertical.....	7
Figura 3. Válvula de liberación de caudal .....	9
Figura 4. Válvula mixta de guillotina .....	10
Figura 5. Diagrama de una compuerta en un generador de avenida.....	10
Figura 6. Posición del Sol (S), la Luna (L) y la Tierra (T) en las mareas. 1 y 2, posición de los astros en sicigia; 3, posición de éstos en cuadratura; a b, elevación máxima de las aguas. ....	12
Figura 7. Sistema de bombeo y compuerta.....	14
Figura 8. Sistema de bombeo y válvula de doble flujo.....	15
Figura 9. Diagrama de compuerta de marea. ....	15
Figura 10. Reductores de velocidad.....	17
Figura 11. Reductor de velocidad con engranes de tornillo sinfín y rueda helicoidal .....	17
Figura 12. Diagrama de Puente tipo H .....	18
Figura 13. Sistemas de protección .....	20
Figura 14. Configuraciones de sensores capacitivos para la medida del nivel de líquidos conductores y no conductores .....	27
Figura 15. Sensor Capacitivo del Modelo .....	30
Figura 16. Circuito generador de pulsos.....	31
Figura 17. Sensor de flujo óptico de tipo turbina .....	35
Figura 18. Diagrama de bloques del sistema de control .....	40
Figura 19. Diagrama de flujo de la rutina de interrupción IC1.....	52
Figura 20. Listado de la rutina de interrupción OC1 .....	53
Figura 21. Análisis de tiempo de la rutina de interrupción IC1 .....	54
Figura 22. Listado del convertidor A/D.....	64
Figura 23. Conjuntos difusos (temperatura) .....	75
Figura 24. Desdifusión por método del centroide.....	80
Figura 25. Funciones de membresía de las variables de entrada al sistema .....	81
Figura 26. Diagrama de multiplexor .....	97
Figura 27. Diagrama de flujo del programa principal .....	100

### Índice de Tablas

Tabla 1. Entradas del circuito puente tipo H .....	19
Tabla 2. Pre-escalador .....	47
Tabla 3. Control de flanco de captura de entrada .....	50
Tabla 4. Canales del Convertidor AD.....	60
Tabla 5. Selección de canales del convertidor A/D .....	63
Tabla 6. Selector del prescalador de Tasa de Transferencia.....	73
Tabla 7. Tasa de transferencia más alta (Salida del pre-escalador de la tabla anterior).....	73
Tabla 8. Tabla Acciones de Salida de Comparación del Temporizador .....	89
Tabla 9. Pre-escalador del Temporizador .....	92
Tabla 10. Orden de Apilamiento o Entrada a Interrupciones .....	95
Tabla 11. Tabla de saltos a las Subrutinas .....	105