

UNIVERSIDAD MAYOR DE SAN ANDRES
FACULTAD DE TECNOLOGIA
CARRERA DE ELECTRONICA Y TELECOMUNICACIONES



PROYECTO DE GRADO
PRESENTADO PARA OBTENER EL GRADO DE LICENCIATURA EN
ELECTRONICA Y TELECOMUNICACIONES
DISEÑO DE UN SISTEMA DE MEZCLA DE BEBIDAS
PARA EL BAR AQUA
POSTULANTE: CARLOS ANDRES CHUQUIMIA MENDEZ
TUTOR: LIC. HENRY ALCON CHOQUE

LA PAZ - BOLIVIA

Marzo, 2022

AGRADECIMIENTOS

Primeramente, agradezco a DIOS y la VIRGEN por darme, su bendición, la vida y la inteligencia para desarrollar este proyecto. También agradezco a todos mis docentes de la carrera de Electrónica y Telecomunicaciones por la formación recibida que sin ellos no sería posible esta meta. Agradezco a mi TUTOR LIC. Henry Alcon Choque, por todo su apoyo, comprensión, confianza y sobre todo su contribución oportuna en la revisión del presente proyecto.

DEDICATORIA

Dedico este proyecto a mis padres Andrés Carlos Chuquimia Endara y Betty Mendez Mercado que han sido un pilar fundamental en mi formación como profesional, por brindarme la confianza, consejos y apoyarme económicamente para lograrlo. A mis hermanos y demás familia en general por el apoyo que siempre me brindaron día a día en el transcurso de cada año de mi carrera Universitaria.

INDICE

CAPITULO I

1. ANTECEDENTES	1
1.1. PLANTEAMIENTO DEL PROBLEMA.....	1
1.2. OBJETIVOS.....	2
1.2.1. OBJETIVO GENERAL	2
1.2.2. OBJETIVO ESPECIFICO.....	2
1.3. JUSTIFICACION	2
1.4. DELIMITACION.....	3
1.4.1. TEMPORAL.....	3
1.4.2. ESPACIAL	4
1.5. METODOLOGIA	4
1.5.1. PLANIFICAION DE SISTEMAS	4
1.5.2. DESARROLLO DE SISTEMAS.....	4
1.5.3. MANTENIMIENTO DE SISTEMAS.....	4

CAPITULO II

2. INGENIERIA DEL PROYECTO	5
2.1. FUNDAMENTO TEORICO.....	5
2.1.1. COCTELERÍA.....	5
2.1.2. ELEMENTOS DE LA COCTELERÍA PRESENTES EN EL PROTOTIPO.....	6
2.1.2.1. COCTELERA	6
2.1.2.2. VASO MEZCLADOR	7
2.1.2.3. VASOS MEDIDORES O DOSIFICADORES	8
2.1.3. NORMAS PARA LA PREPARACIÓN DE LOS COCTELES	9
2.1.3.1. INGREDIENTES	9
2.1.3.2. CANTIDADES	10
2.1.3.3. BATIDO O MEZCLA	11
2.1.3.4. PRESENTACIÓN.....	12

2.1.3.5. NORMAS GENERALES PARA LA ELABORACIÓN DE COCTELES.....	12
2.1.4. LAS COMBINACIONES.....	14
2.1.4.1. MEDIDAS.....	14
2.1.4.2. DENSIDADES.....	15
2.1.4.3. SERVICIO.....	16
2.1.5. MICROCONTROLADOR PIC16F877	17
2.1.5.1. CONFIGURACIÓN DE PINES	18
2.1.5.1.1. DESCRIPCIÓN DE LOS PINES DEL MICROCONTROLADOR.....	22
2.1.5.2. EL OSCILADOR EXTERNO.....	24
2.1.5.3. RESET.....	26
2.1.5.4. MEMORIA DE PROGRAMA (FLASH).....	28
2.1.5.5. MEMORIA DE DATOS (RAM).....	29
2.1.5.6. RESUMEN DE ALGUNOS DE LOS REGISTROS DE CONFIGURACIÓN.....	30
2.1.5.6.1. FUNCIÓN DE ALGUNOS REGISTROS ESPECIALES.....	31
2.1.5.6.2. REGISTROS DE CONTROL DEL MÓDULO CONVERTIDOR ANÁLOGO/DIGITAL.....	36
2.1.5.6.2.1. REGISTRO ADCON0.....	36
2.1.5.6.2.2. REGISTRO ADCON1.....	37
2.1.5.6.3. REQUERIMIENTO PARA EL MUESTREO.....	39
2.1.5.6.4. SELECCIÓN DE LA FRECUENCIA DE CONVERSIÓN.....	40
2.1.5.6.5. CARACTERÍSTICAS PRINCIPALES DEL PIC16F877A	40
2.1.5.6.6. PRIMERA EJECUCIÓN DEL IDE PIC16F877A Y CONFIGURACION INICIAL PARA EL DESARROLLO DEL PROYECTO.....	41
2.1.5.6.7. CARACTERÍSTICAS PRINCIPALES DEL MIKROC PRO FOR PIC.....	44

2.1.5.6.8. MENEJAR LOS PROYECTOS	45
2.1.5.6.8.1. PROJECT MANAGER (ADMINISTRADOR DEL PROYECTO).....	45
2.1.5.6.8.2. LIBRARY MANAGING (EDITOR DE LIBRERIAS).....	46
2.1.5.6.8.3. CODE EDITOR (EDITOR DE CÓDIGO).....	47
2.1.5.6.8.4. COMPILAR Y SOLUCIONAR LOS ERRORES	48
2.1.5.6.9. GRABAR CON PICKIT 2	49
2.1.6. MODULO BLUETOOTH HC-06	52
2.1.6.1. CARACTERISTICAS	54
2.1.6.2. DIFERENCIAS ENTRE EL MODULO BLUETOOTH HC-06 Y EL HC-05	55
2.1.6.3. MODULO BLUETOOTH HC-06 Y PIC 16F877A	57
2.1.6.4. CONEXIÓN DEL MODULO BLUETOOTH HC-06 DESDE UN CELULAR	59
2.1.7. MODULO BOMBA DE AGUA.....	61
2.1.7.1. COMO FUNCIONA UNA BOMBA DE AGUA	62
2.1.7.2. CARACTERISTICAS	64
2.1.8. MÓDULO PUENTE H L298N	64
2.1.8.1. CONEXIÓN Y FUNCIONAMIENTO.....	66
2.1.8.2. CARACTERISTICAS	67
2.2. DESARROLLO PRACTICO / EXPERIMENTAL.....	68
2.2.1. DIAGRAMA DE BLOQUES	69
2.2.2. COMPONENTES QUE SE UTILIZO PARA EL PROYECTO.....	69
2.2.2.1. PIC 16F877A	69
2.2.2.2. MODULO BLUETOOTH HC-06	70
2.2.2.3. MODULO BOMBA DE AGUA.....	70
2.2.2.4. MODULO PUENTE H L298N	71
2.2.3. DESARROLLO DEL PROTOTIPO	72

2.2.4. CODIGO PIC16F877A.....	73
2.2.5. CIRCUITO FINAL.....	76
CAPITULO III	
3. ANALISIS DE COSTOS.....	77
3.1. COSTOS FIJOS	77
3.2. COSTOS VARIABLES	77
CAPITULO IV	
4. CONCLUSIONES Y RECOMENDACIONES.....	78
4.1. CONCLUSIONES.....	78
4.2. RECOMENDACIONES.....	78
BIBLIOGRAFIA.....	80
ANEXOS	81

INDICE DE FIGURAS

Figura 1. Coctelera.....	7
Figura 2. Vaso mezclador	8
Figura 3. Vasos medidores o dosificadores.....	9
Figura 4. Distribución de pines del PIC16F877A.....	7
Figura 5. Distribución de los puertos del PIC16F877A.....	19
Figura 6. Capacidad de corriente de los pines y puertos.....	21
Figura 7. Conexión de un oscilador XT	25
Figura 8. Conexión de un oscilador RC.....	26
Figura 9. Conexión del botón de reset	27
Figura 10. Memoria de programa del PIC16F877A	29
Figura 11. Registros que conforman el Contador de programa.....	32
Figura 12. Bits del Registro de Estados	34
Figura 13. Modelo circuital de la Entrada Análoga.....	39
Figura 14. Primera ejecución para configurar el IDE.....	42
Figura 15. Sistema operativo a instalar otro programa	42
Figura 16. Carpeta del programa mikroC PRO for PIC	43
Figura 17. Características Principales del Mikroc Pro For Pic.....	44
Figura 18. Ventana Project Manager.....	45
Figura 19. Ventana Library Manager.....	46
Figura 20. Editor de Código.....	48
Figura 21. Compilación y solución de los errores	49
Figura 22. Ventana del quemador de PIC	50
Figura 23. Mensaje del Pic Device.....	51
Figura 24. Lista desplegable Device	51
Figura 25. Mensaje del grabado en el dispositivo	52
Figura 26. Mensaje de programación exitosa.....	52
Figura 27. Módulo Bluetooth HC-06.....	53
Figura 28. Conexión Módulo Bluetooth HC-06 y microcontrolador pic16f877a ...	54

Figura 29. Diferencias entre los módulos bluetooth HC-05 y HC-06	56
Figura 30. Modulo bluetooth HC-05 y Modulo bluetooth HC-06	57
Figura 31. Conexión del Módulo bluetooth HC-06 al PIC16F877A.....	58
Figura 32. Vinculación del dispositivo móvil con el Módulo bluetooth HC-06.....	59
Figura 33. Conexión del dispositivo móvil y el Módulo bluetooth HC-06	60
Figura 34. Selección del dispositivo móvil con el Módulo bluetooth HC-06.....	61
Figura 35. Modulo Bomba de Agua.....	62
Figura 36. Modulo Bomba de Agua visto por dentro	63
Figura 37. Modulo puente H L298N y sus partes.....	65
Figura 38. Conexión del Modulo puente H L298N y los motores DC	67
Figura 39. Diagrama de bloques, funcionamiento del prototipo	69
Figura 40. Micro Controlador Pic 16f877a.....	69
Figura 41. Módulo Bluetooth hc-06.....	70
Figura 42. Modulo Bomba de Agua.....	71
Figura 43. Modulo Puente H L298N	71
Figura 44. Armado del Circuito.....	74
Figura 45. Armado del Circuito en Proteus.....	74

INDICE DE TABLAS

Tabla 1. De pines del puerto A al puerto D.....	21
Tabla 2. Descripción de cada pin del PIC16F877A.....	22
Tabla 3. Frecuencias de conversión para el módulo A/D.....	37
Tabla 4. Selección de los canales análogos a utilizar.....	38
Tabla 5. Costos fijos del proyecto	77
Tabla 6. Costos variables del proyecto.....	77

RESUMEN

El presente proyecto de grado de Licenciatura “DISEÑO DE UN SISTEMA DE MEZCLA DE BEBIDAS PARA EL BAR AQUA” se basa en la integración de diferentes tecnologías estudiadas a lo largo de la carrera de Electrónica y Telecomunicaciones con la finalidad de tratar de ser un aporte para los clientes del BAR AQUA.

Este proyecto tiene como objetivo principal elaborar un diseño de un sistema de mezcla de bebidas para los clientes del Bar Aqua, su operación es que ni bien se prende ya podremos conectarnos inalámbricamente con el dispositivo móvil y el modulo bluetooth, para que pueda mandar datos a la mezcladora y esta pueda elegir que bebida desea que se lo mezcle.

Se utilizó herramientas como un PIC16F877A, Módulo Bluetooth HC-06, Modulo Bomba de Agua y un Modulo Puente H L298N y software como Micro Controlador PIC16F877A se prosiguió con el montaje y los módulos en el prototipo de dispensador para finalmente realizar las pruebas necesarias y verificar el funcionamiento del proyecto.

CAPITULO I

1. ANTECEDENTES

El control manual del mismo proceso, brinda ciertas ventajas y beneficios de orden económico, social y tecnológico uno de los primeros visionarios de este concepto fue Henry Ford (Dearborn, Míchigan; 30 de julio de 1863, 7 de abril de 1947) fue un empresario y emprendedor estadounidense, fundador de la compañía Ford Motor Company y padre de las cadenas de producción modernas utilizadas para la producción en masa pues fue el primero en tratar de construir una máquina cuya constitución interna imitase un organismo viviente.

El brazo móvil mecánico que fue un acontecimiento espectacular. Economizó tiempo, dinero y trabajo humano. A consecuencia de esto y viendo las grandes ventajas, con el paso de los años y de la mano de la tecnología emergente y aplicando el nuevo concepto.

Por el cual se plantea hacer un sistema de mezcla de licores, con la elaboración de un prototipo funcional de un dispensador automático de bebidas, cuya elaboración se realiza mediante una serie de componentes como; pic, modulo bomba de agua, modulo bluetooth, entre otros, que serían accionados por una aplicación móvil a través una red inalámbrica de área personal. Garantizando así mayor dinámica (Agilidad, Movilidad, Calidad) en la barra, uno de los sectores más críticos en un bar.

1.1. PLANTEAMIENTO DEL PROBLEMA

En la actualidad se realizan eventos en el BAR AQUA donde asisten cerca de 100 personas. Los eventos realizados cuentan de una barra libre, período de tiempo en el que una persona puede consumir de manera gratuita las bebidas establecidas para el consumo

en el Bar en la que se sirve un tipo de coctel: wiski mezclado con agua y ron con coca-cola. La barra libre no genera ganancias a partir del costo inicial de la entrada al local. Sin embargo, una gran cantidad de personas (aproximadamente el 50%) prefieren comprar otro tipo de bebidas en la barra que no están incluidas en la barra libre.

Estas bebidas no incluidas generan ganancias durante la noche por lo que es de gran interés por parte del local vender la mayor cantidad de bebidas no incluidas a los clientes sin que las bebidas incluidas ocupen el tiempo del barman.

1.2. OBJETIVOS

1.2.1. OBJETIVO GENERAL

- ❖ Diseñar un sistema de mezcla de bebidas para el Bar Aqua mediante dispositivos electrónicos con comunicación Modulo Bluetooth.

1.2.2. OBJETIVO ESPECIFICO

- Diseñar un sistema para la elección del tipo de coctel
- Diseñar una aplicación móvil para la preparación de los cocteles.
- Diseñar un prototipo con componentes que demuestre el proceso de la mezcla de cocteles
- Diseñar un sistema de dispensador que contenga las diferentes bebidas.

1.3. JUSTIFICACION

Buscando reducción en los tiempos de procesamiento de información, reducción de costos, uso eficiente de la materia prima, flexibilidad para adaptarse a nuevos productos y asegurando una mejora en la calidad del trabajo del operador y en el desarrollo del

proceso, se plantea la elaboración de un prototipo de dispensador de bebidas alcohólicas funcional que vaya acorde a una idea de negocio que en este se trata de un Bar Tecnológico.

Adicional a esto se espera que cualquier persona autorizada independientemente de sus conocimientos pueda llegar a desempeñarse como BARMAN, que es quien se encarga de mezclar y servir bebidas, esto sin llegar a afectar el servicio o dinámica del Bar.

Asimismo, se pretende aplicar un plus novedoso, que en este caso sería ofrecer cierta movilidad a quien deba operar el prototipo, lo anterior se logra con un módulo de control para dispositivos móviles y un módulo bluetooth, con los cuales se busca comunicación inalámbrica.

Ya después de haber conseguido establecer conexión y poder propagar energía sin materia a través del espectro de un punto a otro, utilizando para esto una red inalámbrica de área personal, se podrá accionar el prototipo, sin necesidad de estar al frente de este para operarlo.

Por lo que el impacto no se vería reflejado solo en optimizar la dinámica de la barra, siendo esta uno de los lugares más atareados y concurridos del sitio, sino también en la reducción de tiempos, que todo negocio lo ve reflejado como pérdidas.

1.4. DELIMITACION

1.4.1. TEMPORAL

El sistema se desarrollará entre los meses comprendidos desde marzo de 2022 hasta mayo 2022, en el transcurso de este tiempo se realizará los objetivos planteados anteriormente.

1.4.2. ESPACIAL

El presente proyecto se llevará a cabo en el BAR AQUA ubicado en COTA COTA Av. Muñoz Reyes Calle 26 en la ciudad de La Paz Bolivia.

1.5.METODOLOGIA

Su estructura principal se define por una serie de fases, las cuales a su vez se descomponen en módulos, actividades y tareas.

1.5.1. PLANIFICACION DE SISTEMAS

Tiene como objetivo establecer estratégicamente un modelo de información general.

1.5.2. DESARROLLO DE SISTEMAS

En esta parte se encuentra todo lo relacionado con las actividades y tareas que se deben de cumplir para culminar el desarrollo del sistema, las cuales van desde el análisis de requisitos hasta la instalación del software

1.5.3. MANTENIMIENTO DE SISTEMAS

La idea principal del mantenimiento es que el sistema de información desarrollado permanezca en el tiempo, adicional que se encuentre preparado a las eventualidades que se pudiesen presentar, con el fin de que cumpla a cabalidad el objetivo para el que fue desarrollado.

CAPITULO II

2. INGENIERIA DEL PROYECTO

2.1. FUNDAMENTO TEORICO

En esta sección se documentó la información relacionada con los conceptos y teorías que permiten contextualizar el proyecto, permitiendo entender las ideas básicas y generalidades de los elementos a tener en cuenta en el desarrollo del mismo. La investigación se realizó seleccionando los datos concernientes a diferentes tipos de clientes que fue analizada y como quisieran que sea la atención en reducción de tiempos que se asemejan a sus objetivos y funciones al prototipo de dispensador de bebidas diseñado.

2.1.1. COCTELERÍA

La coctelería es una técnica de combinación de bebidas, alcohólicas o no, que se realiza siguiendo unos parámetros establecidos, para lograr una bebida armónica con características diferentes a las de sus ingredientes por separado. Para que el resultado sea positivo y agradables, es fundamental conseguir una armonía entre sus ingredientes teniendo en cuenta los siguientes aspectos:

- ❖ Sabor
- ❖ Color
- ❖ Textura
- ❖ Temperatura
- ❖ Presentación

Es importante destacar que los cocteles más conocidos internacionalmente tienen unas pautas estrictas de elaboración y deben seguir fielmente unas recetas que determinan los ingredientes, cantidades, técnicas de preparación y decoración, pues el simple hecho de modificar, aunque sea un poco, cualquiera de estos parámetros, da lugar a un nuevo coctel que debe denominarse de forma diferente.

En vista de lo anterior, se puede decir que el número de cocteles que se pueden crear combinando diferentes bebidas, alcohólicas o no, es prácticamente infinito y está abierto a la creatividad del profesional que lo elabora y al gusto de los clientes.

2.1.2. ELEMENTOS DE LA COCTELERÍA PRESENTES EN EL PROTOTIPO

Para poder elaborar un coctel de una manera correcta y profesional, es preciso disponer de una serie de elementos y utensilios que son totalmente imprescindibles y son las siguientes:

2.1.2.1. COCTELERA

La coctelera es una herramienta fundamental y de gran utilización en la preparación de muchos cocteles, en especial de aquellos que requieran ir fríos y perfectamente mezclados entre sí. La coctelera permite obtener esta mezcla perfecta mediante la agitación energética de los ingredientes, en especial cuando estos poseen diferentes densidades.

La coctelera se compone de tres partes:

- ✚ Tapa que cierra la coctelera para evitar que el líquido se derrame durante la

agitación.

- ✚ Vaso donde se introducen los ingredientes acompañado de hielo.
- ✚ Cubre-vaso con orificios que actúa como colador, evitando que pasen a la copa loshielos o las partículas sólidas que pudiera haber en la mezcla.

Las cocteleras se fabrican de material fuerte, generalmente acero inoxidable, alpaca plateada o plata, y su capacidad puede ser de 250 ml, 750 ml o 1 litro, siendo las más comunes las de 750 ml.



Figura 1. Coctelera

Fuente: <https://stanley-pmi.com.bo/products/coctelera-adventure-shaker-590-ml-acero-stanley>

2.1.2.2. VASO MEZCLADOR

Se trata de un vaso cónico, de unos 750 ml de capacidad, utilizado para mezclar líquidos que no necesiten ser batidos energicamente, normalmente líquidos de poca densidad y mezclas sin azúcar.

El conjunto se compone de tres partes o elementos:

- Vaso: generalmente de vidrio, con un pequeño canal en la parte superior.

- Pasador o gusanillo: es un elemento de acero inoxidable, con mango, y provisto de una espiral que sirve para colar la mezcla, evitando que caigan los hielos, pulpas u otros sólidos indeseados.
- Cucharilla de bar: se trata de una cucharilla de mango muy largo que sirve para mezclar y dosificar los ingredientes. En general una cucharilla de bar tiene una capacidad de 5 cc.



Figura 2. Vaso mezclador

Fuente: <https://www.dispensadores-de-cerveza.es/set-cocteleria-coctelera-boston-con-el-original-vaso-mezclador/a-454061/>

2.1.2.3. VASOS MEDIDORES O DOSIFICADORES

Son vasitos graduados, de distintos tamaños, que sirven para dosificar la cantidad de líquido que se ha de mezclar. Los hay pequeños, especiales para dosificar los licores, y grandes para otros líquidos como zumos, agua, etc.



Figura 3: Vasos medidores o dosificadores

Fuente: <https://www.losvinos.com.ar/tipos-de-vinos/vaso-medidor-o-jigger/>

2.1.3. NORMAS PARA LA PREPARACIÓN DE LOS COCTELES

A la hora de elaborar un coctel hay que centrarse en cuatro aspectos fundamentales:

- Ingredientes
- Cantidades
- Batido o mezcla
- Presentación

2.1.3.1. INGREDIENTES

El principio básico para elaborar correctamente un coctel es atenerse a la receta correspondiente, respetando estrictamente los ingredientes y cantidades establecidos en la misma.

En relación a los ingredientes que van a formar parte del coctel se puede establecer que en un coctel no se deben incluir más de 5 productos diferentes, contando con el elemento

decorativo.

En general, un coctel suele llevar:

- ❖ Una bebida alcohólica destilada (ginebra, whisky, ron) como ingrediente principal.
- ❖ Uno o dos licores para aportar dulzor y color al conjunto.
- ❖ Un poco de limón u otro elemento cítrico o amargo.
- ❖ Un elemento decorativo comestible (fruta, aceituna, hierbabuena).

En el caso de los combinados o long drinks, la bebida principal se acompaña por un refresco, soda o zumo, que disminuyen su fuerza.

Uno de los ingredientes fundamentales, en casi todos los casos es el hielo que se incluye para refrescar y, en ocasiones suavizar, la bebida resultante.

El hielo empleado en coctelería debe estar realizado con agua filtrada o mineral para que esté exento de sabores extraños. Normalmente deben usarse cubitos duros, compactos y de tamaño medio para que enfríen sin aguar el conjunto, aunque en ocasiones, cuando la receta así lo indique, se empleará hielo picado.

Otro elemento que, en ocasiones, interviene en la elaboración de los cocteles es el azúcar. Cuando el azúcar forma parte del propio coctel no se empleará nunca en grano sino en forma de jarabe de azúcar para facilitar su disolución. En algunos casos puede emplearse azúcar en grano para elaborar ciertas decoraciones.

2.1.3.2. CANTIDADES

Se debe ser muy cuidadoso y estricto a la hora de dosificar las cantidades y

proporciones de los distintos ingredientes y sin embargo no es recomendable el uso de medidores pues esto da una imagen de falta de profesionalidad, por tanto, es preciso adquirir el sentido de las medidas para poder hacerlo. Esta es una de las destrezas que más pueden costar a la persona que se inicia en este campo, no obstante, con la práctica se irá adquiriendo la soltura necesaria para realizar las medidas sin ningún tipo de ayuda.

En cuanto a las medidas indicadas en las recetas, estas se refieren, normalmente al vaso o copa en el que se va a servir, así cuando dice $\frac{1}{3}$ de ginebra, se refiere a la cantidad de ginebra necesaria para llenar $\frac{1}{3}$ de la copa.

Otras veces, las medidas vienen expresadas en cc o ml., en este caso conviene utilizar el vaso medidor para conocer a qué cantidad exacta corresponde y tomar la referencia para poder calcularla después sin necesidad de emplear el medidor.

En ocasiones las medidas vienen expresadas en onzas (oz), en estos casos basta con saber que una onza corresponde a 30 ml.

Cuando la receta mencione el término golpe o dash, se refiere a una dosis del gotero y que viene a ser equivalente a 0,9 ml.

2.1.3.3. BATIDO O MEZCLA

El batido es la acción básica necesaria para que se mezclen entre sí los distintos ingredientes que componen el coctel. La cantidad e intensidad del batido, así como el tiempo necesario, dependerá de los ingredientes que se quieran mezclar y sobre todo de su densidad.

La mezcla de los ingredientes puede realizarse de dos formas:

- **Removiendo:** Se remueven todas aquellas bebidas que contienen ingredientes muy fluidos o que se mezclan con facilidad. Para ello se utilizará el vaso mezclador y una cucharilla de mango largo. La forma de realizarlo es introducir los ingredientes en el vaso con el hielo y remover vigorosamente la mezcla durante unos 20 segundos, sirviendo inmediatamente, sin dejarla reposar.
- **Agitando:** En general es preciso agitar todas aquellas mezclas en las que intervienen más de 3 bebidas o con ingredientes densos como licores espesos, zumos o jarabes. Para agitar una mezcla se utilizará la coctelera, en la cual se introducen los ingredientes junto con el hielo y se agita con energía hasta notar en las manos el frío de la mezcla; en ese momento se abre la coctelera con cuidado y se sirve el contenido en la copa o vaso.

2.1.3.4. PRESENTACIÓN

El servicio de cocteles culmina con una presentación y decoración adecuada. Siempre es conveniente adornar un coctel antes de servirlo. Cualquier motivo de decoración bien sea una guinda, una ramita de menta fresca o el borde de una copa escarchada harán del coctel una bebida más sugestiva y agradable a la vista del cliente.

2.1.3.5. NORMAS GENERALES PARA LA ELABORACIÓN DE COCTELES

Se reúnen a continuación algunas de las principales normas que deben respetarse a la hora de preparar un coctel:

- ✓ Preparar todos los elementos e ingredientes necesarios antes de empezar a confeccionar el coctel.

- ✓ Tener preparada también la copa en la que se ha de servir el coctel y los elementos decorativos que correspondan.
- ✓ Respetar fielmente la receta, empleando los ingredientes precisos en las cantidades exactas.
- ✓ El orden correcto de incorporar los ingredientes para la mezcla en un coctel es:
 - ❖ Hielo
 - ❖ Azúcar (si la preparación lo requiere)
 - ❖ Zumos de fruta
 - ❖ Licores
- ✓ Si en un coctel intervienen varias bebidas alcohólicas, el orden de incorporación se establecerá de acuerdo a su graduación alcohólica, comenzando por la de mayor graduación que suele ser la bebida base.
- ✓ Ciertos ingredientes como azúcar, angostura, granadina, menta, zumo de limón, etc., deben usarse siempre con moderación.
- ✓ Batir o mezclar los ingredientes de la forma adecuada: emplear la coctelera para los cocteles con ingredientes densos y el vaso mezclador para los cocteles que se fusionan fácilmente y que no requieren batirse enérgicamente.
- ✓ Emplear el hielo apropiado en tamaño y cantidad. No deben utilizarse trocitos pequeños para no aguar la mezcla.
- ✓ No emplear nunca bebidas gaseosas en la coctelera.
- ✓ Las copas o vasos de coctel no deben llenarse hasta el borde.
- ✓ Utilizar siempre las pinzas para añadir el hielo o cualquier otro elemento

(cerezas, trozos de fruta, rodajas de limón, etc.).

- ✓ La textura de un coctel debe ser uniforme, por lo que muchas veces conviene colarlo.
- ✓ Si se desea suavizar un coctel, utilizar hielo en trozos pequeños y batir durante un rato.
- ✓ La guinda, la aceituna o cualquier otro elemento con el que se vaya a decorar el coctel debe colocarse en la copa antes de servir la preparación.
- ✓ Limpiar bien todos los elementos empleados para la preparación y servicio del coctel.

2.1.4. LAS COMBINACIONES

2.1.4.1. MEDIDAS

A la hora de preparar los distintos combinados de coctelería es muy importante conocer en qué cantidades y proporciones deben servirse y en qué medida debe intervenir cada uno de los ingredientes que componen la receta. Para aclarar un poco estos conceptos y poder realizar los cálculos correspondientes se indican a continuación cuáles son las medidas más comunes:

Capacidades de los vasos y copas:

- Copa de coctel: 75- 100 ml.
- Copa de combinación: 120-160 ml
- Copa de champán tipo flauta: 180 ml
- Copa de licor: 25 ó 50 ml.
- Vaso de Whisky: 200 ml
- Vaso alto: 180 - 300 ml

Medidas y equivalencias: Normalmente las medidas que deben servirse de las distintas bebidas están en función del tipo de bebida de que se trate. Como norma general se puede decir que las cantidades indicadas para una ración o copa de las principales bebidas se establecen de la siguiente forma:

- ✚ Aperitivos: Vermut, Ricard, Pernaud, Dubonnet, etc.: 80 ml (2,5 onzas)
- ✚ Coñacs, brandy y similares: 60 ml (2 onzas)
- ✚ Licores: anís, cointreau, etc.: 55 ml (1 a 2 onzas)
- ✚ Whisky: 55 ml (1,5 a 2 onzas)
- ✚ Ginebra y ron: 55 ml (1,5 a 2 onzas)

Es frecuente que en la receta se indiquen las cantidades mediante fracciones de lo que se denomina unidad de coctel, la equivalencia de estas medidas es la siguiente:

- ✓ 1 Coctel: 50 ml. - 1/5 Coctel: 10 ml. - 3/4 Coctel: 37 ml.
- ✓ 1/2 Coctel: 25 ml. - 1/6 Coctel: 8 ml. - 3/5 Coctel: 30 ml.
- ✓ 1/3 Coctel: 16 ml. - 2/3 Coctel: 33 ml. - 4/5 Coctel: 40 ml.
- ✓ 1/4 Coctel: 12.5 ml. - 2/5 Coctel: 20 ml. - 5/6 Coctel: 41 ml.

2.1.4.2. DENSIDADES

Otro aspecto importante a tener en cuenta al realizar las distintas preparaciones es la densidad de las bebidas que se van a utilizar.

No todas las bebidas que intervienen en una preparación tienen la misma densidad y por ello, para conseguir los resultados deseados, es preciso seguir al pie de la letra las indicaciones de la receta, incorporando las distintas bebidas ordenadamente de acuerdo

con sus respectivas densidades, normalmente las más densas en primer lugar y las otras en orden descendente.

Esto es especialmente importante a la hora de preparar ciertos cocteles en los que se pretende obtener un efecto especialmente decorativo mediante la superposición de productos de diferentes colores. Este resultado sólo es posible si los líquidos empleados poseen diferentes densidades y se sirven de forma que no se mezclen entre sí.

2.1.4.3. SERVICIO

Para realizar adecuadamente las vistosas combinaciones multicolores, cuya denominación genérica recibe el nombre de Pousse-café, es preciso conocer las densidades de los distintos ingredientes que se van a utilizar, dominar la técnica de servicio necesaria para hacerlo y ser muy cuidadoso a la hora de realizar la preparación.

Aunque precisa algo de práctica, la técnica de preparación de estos combinados es más sencilla de lo que en principio pueda parecer y si se tiene un cierto conocimiento de las propiedades y densidades de los ingredientes y con un poco de imaginación es posible crear combinaciones sorprendentes presentando toda una paleta de colores en un mismo vaso.

El principio básico que rige estas preparaciones es la superposición de elementos por su grado de densidad, por lo cual implica colocar siempre en primer lugar aquellos ingredientes más densos como los licores cremosos, para que se mantengan en el fondo y posteriormente ir agregando los demás ingredientes en orden descendente de densidades hasta llegar a los más ligeros.

Para lograr que los distintos productos no se mezclen al servirlos no basta con respetar

el orden sino también hay que cuidar la forma en que se vierten en la copa para que no golpeen al líquido que ya se encuentra allí, lo que haría que ambos se mezclasen. La forma de conseguir que un líquido caiga sobre otro sin mezclarse consiste en incorporarlo muy lentamente, para que se deposite sobre el anterior sin golpear su superficie. Para facilitar este proceso hay que añadirlo a la copa con ayuda de una cuchara de bar, haciendo resbalar el licor suavemente por la superficie de la cuchara y por el borde de la copa o vaso.

Para conseguir mejores resultados conviene respetar las siguientes recomendaciones:

- ❖ Emplear siempre que sea posible los líquidos bien fríos pues de esta forma son más fáciles de manejar.
- ❖ Limpiar o cambiar la cucharilla después de cada ingrediente para evitar que manche el siguiente.
- ❖ Para realizar la preparación situar la copa a la altura del pecho, de esta manera se aprecia mejor como se va colocando cada capa.
- ❖ Una vez elaborada la preparación manejarla con mucho cuidado, sin movimientos bruscos para que no se mezclen los productos con la agitación.

2.1.5. MICROCONTROLADOR PIC16F877A

El PIC16F877A es un microcontrolador con memoria de programa tipo FLASH, lo que representa gran facilidad en el desarrollo de prototipos y en su aprendizaje ya que no se requiere borrarlo con luz ultravioleta como las versiones EPROM, sino que permite reprogramarlo nuevamente sin ser borrado con anterioridad.

El PIC16F877A es un microcontrolador de Microchip Technology fabricado en

tecnología CMOS, su consumo de potencia es muy bajo y además es completamente estático, esto quiere decir que el reloj puede detenerse y los datos de la memoria no se pierden.

El encapsulado más común para este microcontrolador es el DIP (Dual In-line Pin) de 40 pines, propio para usarlo en experimentación. La referencia completa es PIC16F877-04 para el dispositivo que utiliza cristal oscilador de hasta 4 MHz, PIC16F877-20 para el dispositivo que utiliza cristal oscilador de hasta 20 MHz o PIC16F877A-I para el dispositivo tipo industrial que puede trabajar hasta a 20 MHz. Sin embargo, hay otros tipos de encapsulado que se pueden utilizar según el diseño y la aplicación que se quiere realizar. Por ejemplo, el encapsulado tipo surface mount (montaje superficial) tiene un reducido tamaño y bajo costo, que lo hace propio para producciones en serie o para utilizarlo en lugares de espacio muy reducido.

2.1.5.1. CONFIGURACIÓN DE PINES

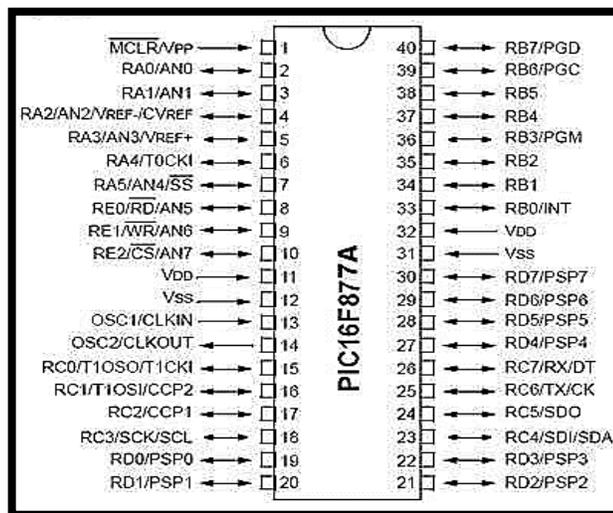


Figura 4. Distribución de pines del PIC16F877A.

Fuente: <https://electocrea.com/products/pic16f877a-i-p>

Los pines de entrada/salida de este microcontrolador están organizados en cinco puertos, el puerto A con 6 líneas, el puerto B con 8 líneas, el puerto C con 8 líneas, el puerto D con 8 líneas y el puerto E con 3 líneas. Cada pin de esos puertos se puede configurar como entrada o como salida independiente programando un par de registros diseñados para tal fin. En ese registro un bit en "0" configura el pin del puerto correspondiente como salida y un bit en "1" lo configura como entrada. Dichos pines del microcontrolador también pueden cumplir otras funciones especiales, siempre y cuando se configuren para ello, según se verá más adelante.

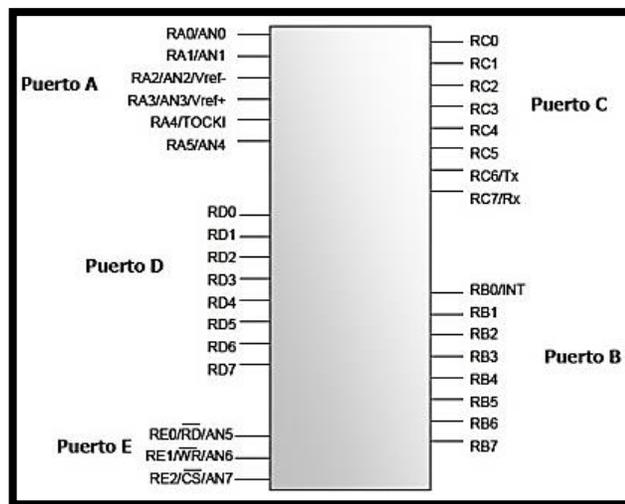


Figura 5. Distribución de los puertos del PIC16F877

Fuente: <https://cifpn1hectorm.wordpress.com/2013/04/10/estudio-de-la-estructura-interna-del-pic-16f877/>

Los pines del puerto A y del puerto E pueden trabajar como entradas para el convertidor Analógico a Digital interno, es decir, allí se podría conectar una señal proveniente de un sensor o de un circuito analógico para que el microcontrolador la convierta en su

equivalente digital y pueda realizar algún proceso de control o de instrumentación digital. El pin RB0/INT se puede configurar por software para que funcione como interrupción externa, para configurarlo se utilizan unos bits de los registros que controlan las interrupciones.

El pin RA4/TOCKI del puerto A puede ser configurado como un pin de entrada/salida o como entrada del temporizador/contador. Cuando este pin se programa como entrada digital, funciona como un disparador de Schmitt (Schmitt trigger), puede reconocer señales un poco distorsionadas y llevarlas a niveles lógicos (cero y cinco voltios). Cuando se usa como salida digital se comporta como colector abierto (open collector), por lo tanto, se debe poner una resistencia de pull-up (resistencia externa conectada a un nivel de cinco voltios). Como salida, la lógica es inversa: un "0" escrito al pin del puerto entrega en el pin un "1" lógico. Además, como salida no puede manejar cargas como fuente, sólo en el modo sumidero.

El puerto E puede controlar la conexión en modo microprocesador con otros dispositivos utilizando las líneas RD (read), WR (write) y CS (chip select). En este modo el puerto D funciona como un bus de datos de 8 bits (pines PSP).

La máxima capacidad de corriente de cada uno de los pines de los puertos en modo sumidero (sink) o en modo fuente (source) es de 25 mA. La máxima capacidad de corriente total de los puertos es:

Tabla 1. De pines del puerto A al puerto D

	PUERTO A	PUERTO B	PUERTO C	PUERTO D
MODO SUMIDERO	150 mA	200 mA	200 mA	200 mA
FUENTE	150 mA	200 mA	200 mA	200 mA

El consumo de corriente del microcontrolador para su funcionamiento depende del voltaje de operación, la frecuencia y de las cargas que tengan sus pines. Para un oscilador de 4 MHz el consumo es de aproximadamente 2 mA; aunque este se puede reducir a 40 microamperios cuando se está en el modo sleep (en este modo el micro se detiene y disminuye el consumo de potencia). Se sale de ese estado cuando se produce alguna condición especial que veremos más adelante.

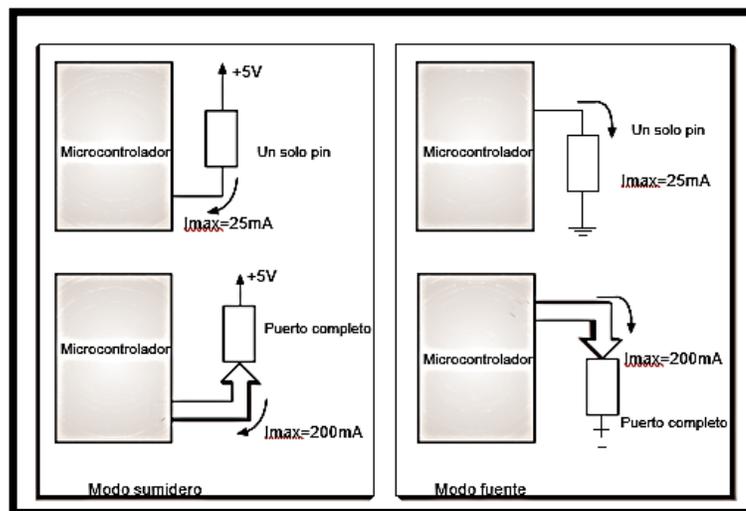


Figura 6. Capacidad de corriente de los pines y puertos.

Fuente: <https://1library.co/document/zgwj3jvy-construccion-prototipo-control-ternos-electronico-puestos-diferentes-llamada.html>

2.1.5.1.1. DESCRIPCIÓN DE LOS PINES DEL MICROCONTROLADOR

Tabla 2. Descripción de cada pin del PIC16F877A.

Nombre pin	Pin	Descripción
RA0/AN0	2	E/S Digital o Entrada análoga 0.
RA1/AN1	3	E/S Digital o Entrada análoga 1.
RA2/AN2 V_{ref} -	4	E/S Digital o Entrada análoga 2.
RA3/AN3/V_{ref} +	5	E/S Digital o Entrada análoga 3.
RA4/T0CKI	6	Bit 4 del puerto A (E/S bidireccional). También se usa como entrada de reloj altemporizador/contador TMR0. Salida de colector abierto.
RA5/SS/AN4	7	E/S Digital o Entrada análoga 4. También lo usa el puerto serial síncrono.
RB0/INT	33	Bit 0 del puerto B (E/S bidireccional). Buffer E/S: TTL/ST. También se usa como entrada de interrupción externa (INT).
RB1	34	Bit 1 del puerto B (E/S bidireccional). Buffer E/S: TTL
RB2	35	Bit 2 del puerto B (E/S bidireccional). Buffer E/S: TTL
RB3/PGM	36	Bit 3 del puerto B (E/S bidireccional). Buffer E/S: TTL (Programación en bajovoltaje)
RB4	37	Bit 4 del puerto B (E/S bidireccional). Buffer E/S: TTL. Interrupción por cambio del pin.
RB5	38	Bit 5 del puerto B (E/S bidireccional). Buffer E/S: TTL. Interrupción por cambio del pin.
RB6/PGC	39	Bit 6 del puerto B (E/S bidireccional). Buffer E/S: TTL/ST. Interrupción por cambio del pin. Entrada de reloj para

		programación serial.
RB7/PGD	40	Bit 7 del puerto B (E/S bidireccional). Buffer E/S: TTL/ST. Interrupción por cambio del pin. Entrada de datos para programación serial.
RC0/T1OSO/T1CKI	15	E/S Digital. Salida del oscilador Timer 1 o entrada de reloj Timer 1.
RC1/T1OSI/CCP2	16	E/S Digital. Entrada del oscilador Timer 1. Entrada Captura 2; Salida Compara2; Salida PWM 2
RC2/CCP1	17	E/S Digital. Entrada Captura 1; Salida Compara 1; Salida PWM 1
RC3/SCK/SCL	18	E/S Digital. Línea de reloj serial asíncrono en el modo SPI y el modo I²C
RC4/SDI/SDA	23	E/S Digital. Línea de datos en el modo SPI o en el modo I²C
RC5/SDO	24	E/S Digital.
RC6/TX/CK	25	E/S Digital. Transmisión asíncrona (USART) o reloj síncrono (SSP).
RC7/RX/DT	26	E/S Digital. Recepción asíncrona (USART) o línea de datos (SSP).
VDD	11, 32	Voltaje de alimentación DC (+)
VSS	12, 31	Referencia de voltaje (GND).
MCLR	1	Entrada de RESET al microcontrolador. Voltaje de entrada durante la programación. En nivel bajo resetea el microcontrolador.
OSC1/CLKIN	13	Entrada oscilador cristal oscilador / Entrada fuente de reloj

		externa.
OSC2/CLKO UT	14	Salida oscilador cristal. Oscilador RC: Salida con un $\frac{1}{4}$ frecuencia OSC1
RD0/PSP0	19	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD1/PSP1	20	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD2/PSP2	21	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD3/PSP3	22	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD4/PSP4	27	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD5/PSP5	28	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD6/PSP6	29	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD7/PSP7	30	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RE0/RD/AN5	8	E/S Digital. Puede se pin de lectura (read) en modo microprocesador.
RE1/WR/AN6	9	E/S Digital. Puede ser pin de escritura (write) en modo microprocesador.
RE2/CS/AN7	10	E/S Digital. Puede ser pin de selección de chip (chip select) en modomicroprocesador.

2.1.5.2. EL OSCILADOR EXTERNO

Todo microcontrolador requiere un circuito externo que le indique la velocidad a la que debe trabajar. Este circuito, que se conoce como oscilador o reloj, es muy simple, pero de vital importancia para el buen funcionamiento del sistema. El PIC16F877A puede utilizar cuatro tipos de oscilador diferentes. Estos tipos son:

- **RC.** Oscilador con resistencia y condensador.
- **XT.** Cristal (por ejemplo, de 1 a 4 MHz).

- **HS.** Cristal de alta frecuencia (por ejemplo 10 a 20 MHz).
- **LP.** Cristal para baja frecuencia y bajo consumo de potencia.

En el momento de programar o “quemar” el microcontrolador se debe especificar qué tipo de oscilador se usa. Esto se hace a través de unos fusibles llamados “fusibles de configuración”.

El tipo de oscilador que se sugiere para las prácticas es el XT con un cristal de 4MHz, porque garantiza precisión y es muy comercial. Internamente esta frecuencia es dividida por cuatro, lo que hace que la frecuencia efectiva de trabajo sea de 1 MHz en este caso, por lo que cada instrucción se ejecuta en un microsegundo. El cristal debe ir acompañado de dos condensadores.

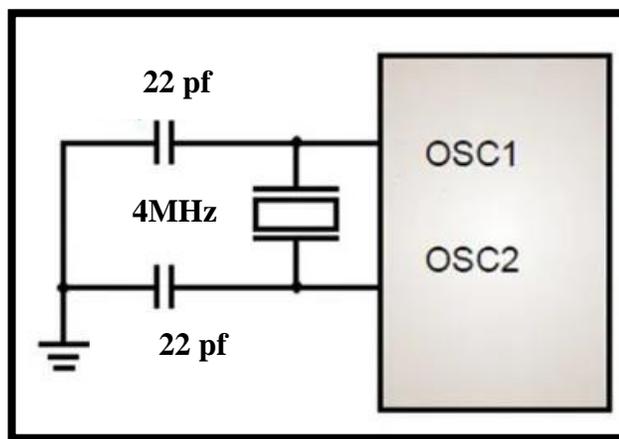


Figura 7. Conexión de un oscilador XT

Fuente: <https://1library.co/document/zlg5342y-diseno-construccion-instalacion-sistema-control-automatizado-electrogeno-grifos.html>

Si no se requiere mucha precisión en el oscilador y se quiere economizar dinero, se puede utilizar una resistencia y un condensador.

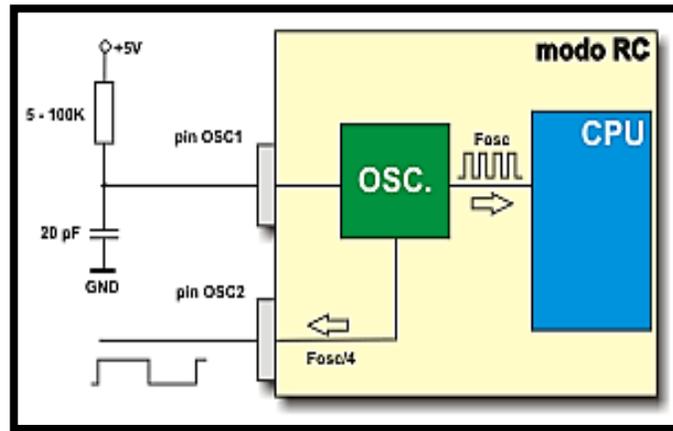


Figura 8. Conexión de un oscilador RC.

Fuente: <https://www.mikroe.com/ebooks/microcontroladores-pic-programacion-en-c-con-ejemplos/oscilador-de-reloj>

2.1.5.3. RESET

En los microcontroladores se requiere un pin de reset para reiniciar el funcionamiento del sistema cuando sea necesario, ya sea por una falla que se presente o porque así se haya diseñado el sistema. El pin de reset en los PIC es llamado MCLR (master clear). Existen varias formas de resetear o reiniciar el sistema:

- ❖ Al encendido (Power On Reset)
- ❖ Pulso en el pin MCLR durante operación normal
- ❖ Pulso en el pin MCLR durante el modo de bajo consumo (modo sleep)
- ❖ El rebase del conteo del circuito de vigilancia (watchdog) durante operación normal
- ❖ El rebase del conteo del circuito de vigilancia (watchdog) durante el modo de bajoconsumo (sleep)

El reset al encendido se consigue gracias a dos temporizadores. El primero de ellos es

el OST (Oscillator Start-Up Timer: Temporizador de encendido del oscilador), orientado a mantener el microcontrolador en reset hasta que el oscilador del cristal es estable. El segundo es el PWRT (Power-Up Timer: Temporizador de encendido), que provee un retardo fijo de 72 ms (nominal) en el encendido únicamente, diseñado para mantener el dispositivo en reset mientras la fuente se estabiliza. Para utilizar estos temporizadores, sólo basta con conectar el pin MCLR a la fuente de alimentación, evitándose utilizar las tradicionales redes de resistencias externas en el pin de reset.

El reset por MCLR se consigue llevando momentáneamente este pin a un estado lógico bajo, mientras que el watchdog WDT produce el reset cuando su temporizador rebasa la cuenta, o sea que pasa de 0FFh a 00h. Cuando se quiere tener control sobre el reset del sistema se puede conectar un botón.

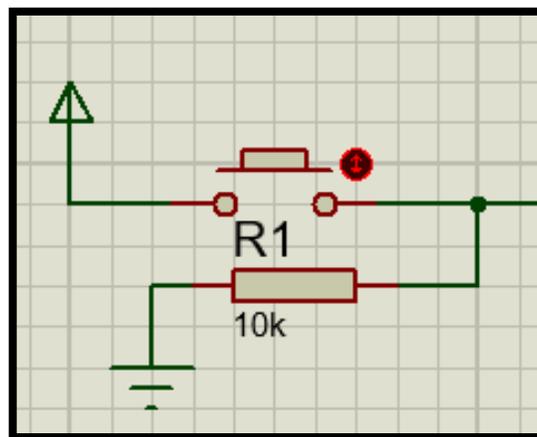


Figura 9. Conexión del botón de reset.

Fuente: www.todopic.com.ar/foros/index.php?topic=47539.0

2.1.5.4. MEMORIA DE PROGRAMA (FLASH)

Es una memoria de 8K de longitud con datos de 14 bits en cada posición. Como es del tipo FLASH se puede programar y borrar eléctricamente, lo que facilita el desarrollo de los programas y la experimentación. En ella se graba o almacena el programa o códigos que el microcontrolador debe ejecutar.

La memoria de programa está dividida en cuatro bancos o páginas de 2K cada uno. El primero va de la posición de memoria 0000h a la 07FFh, el segundo va de la 0800h a la 0FFFh, el tercero de la 1000h a la 17FFh y el cuarto de la 1800h a la 1FFFh.

Vector de reset. Cuando ocurre un reset al microcontrolador, el contador de programa se pone en ceros (0000H). Por esta razón, en la primera dirección del programa se debe escribir todo lo relacionado con la iniciación del mismo.

Vector de interrupción. Cuando el microcontrolador recibe una señal de interrupción, el contador de programa apunta a la dirección 04H de la memoria de programa, por eso, allí se debe escribir toda la programación necesaria para atender dicha interrupción.

Pila (Stack). Estos registros no forman parte de ningún banco de memoria y no permiten el acceso por parte del usuario. Se usan para guardar el valor del contador de programa cuando se hace un llamado a una subrutina o cuando se atiende una interrupción; luego, cuando el micro regresa a seguir ejecutando su tarea normal, el contador de programa recupera su valor leyéndolo nuevamente desde la pila. El PIC16F877A tiene una pila de 8 niveles, esto significa que se pueden anidar 8 llamados a subrutina sin tener problemas.

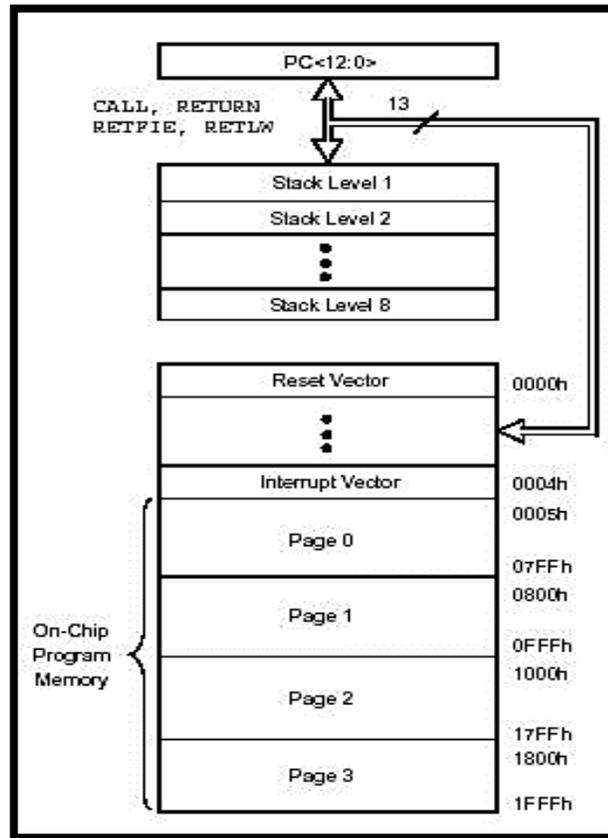


Figura 10. Memoria de programa del PIC16F877A.

Fuente: <https://estuelectronic.wordpress.com/2013/02/05/estudio-de-la-estructura-interna-del-pic16f877/>

2.1.5.5. MEMORIA DE DATOS (RAM)

El PIC16F877A posee cuatro bancos de memoria RAM, cada banco posee 128 bytes. De estos 128 los primeros 32 (hasta el 1Fh) son registros que cumplen un propósito especial en el control del microcontrolador y en su configuración. Los 96 siguientes son registros de uso general que se pueden usar para guardar los datos temporales de la tarea que se está ejecutando.

Todas las posiciones o registros de memoria se pueden acceder directa o indirectamente (esta última forma a través del registro selector FSR). Para seleccionar que página o banco de memoria se trabaja en un momento determinado se utilizan los bits RP0 y RP1 del registro STATUS.

2.1.5.6. RESUMEN DE ALGUNOS DE LOS REGISTROS DE CONFIGURACIÓN

BANCO 0:

- ❖ **TMR0:** Registro del temporizador/contador de 8 bits.
- ❖ **PCL:** Byte menos significativo del contador de programa (PC).
- ❖ **STATUS:** Contiene banderas (bits) que indican el estado del procesador después de una operación aritmética/lógica.
- ❖ **FSR:** Registro de direccionamiento indirecto.
- ❖ **PORTA, PORTB, PORTC, PORTD, PORTE:** Registro de puertos de E/S de datos. Conectan con los pines físicos del micro.
- ❖ **PCLATH:** Byte alto (más significativo) del contador de programa (PC).
- ❖ **INTCON:** Registro de control de las interrupciones.
- ❖ **ADRESH:** Parte alta del resultado de la conversión A/D.
- ❖ **ADCON0:** Controla la operación del módulo de conversión A/D

BANCO 1:

- **OPTION:** Registro de control de frecuencia del TMR0.
- **TRISA, TRISB, TRISC, TRISD. TRISE:** Registros de configuración de la

operación de los pines de los puertos.

- **ADRESL**: Parte baja del resultado de la conversión A/D.
- **ADCON1**: Controla la configuración de los pines de entrada análoga.

BANCO 2:

- **TMR0**: Registro del temporizador/contador de 8 bits.
- **PCL**: Byte menos significativo del contador de programa (PC).
- **FSR**: Registro de direccionamiento indirecto.
- **EEDATA**: Registro de datos de la memoria EEPROM.
- **EEADR**: Registro de dirección de la memoria EEPROM.
- **PCLATH**: Byte alto (más significativo) del contador de programa (PC).
- **INTCON**: Registro de control de las interrupciones.

BANCO 3:

- ❖ **OPTION**: Registro de control de frecuencia del TMR0.
- ❖ **EECON1**: Control de lectura/escritura de la memoria EEPROM de datos.
- ❖ **EECON2**: No es un registro físico.

2.1.5.6.1. FUNCIÓN DE ALGUNOS REGISTROS ESPECIALES

00h o INDF: Registro para direccionamiento indirecto de datos. Este no es un registro disponible físicamente; utiliza el contenido del registro FSR y los bits RP0 y RP1 del registro STATUS para seleccionar indirectamente la memoria de datos, la instrucción que lo acompañe determinará que se debe realizar con el registro señalado.

01h o TMR0. Temporizador/contador de 8 bits. Este es un contador que se puede incrementar con una señal externa aplicada al pin RA4/TOCKI o de acuerdo a una señal interna proveniente del reloj de instrucciones del microcontrolador. El incremento del registro se puede determinar por medio de un pre escalador o divisor de frecuencia, localizado en el registro OPTION. Como una mejora con respecto a referencias anteriores, se le ha agregado la generación de interrupción cuando se rebasa la cuenta (el paso de 0FFh a 00h).

02h o PCL: Contador de programa. Se utiliza para direccionar las palabras de 14 bits del programa que se encuentra almacenado en la memoria ROM; este contador de programas es de 13 bits de ancho. Sobre el byte bajo, se puede escribir o leer directamente, mientras que, sobre el byte alto, no. El byte alto se maneja mediante el registro PCLATH (0Ah). Ante una condición de reset el microcontrolador inicia el contador de programa con todos sus bits en “cero”. Durante la ejecución normal del programa, y dado que todas las instrucciones ocupan sólo una posición de memoria, el contador se incrementa en uno con cada instrucción, a menos que se trate de alguna instrucción de salto.

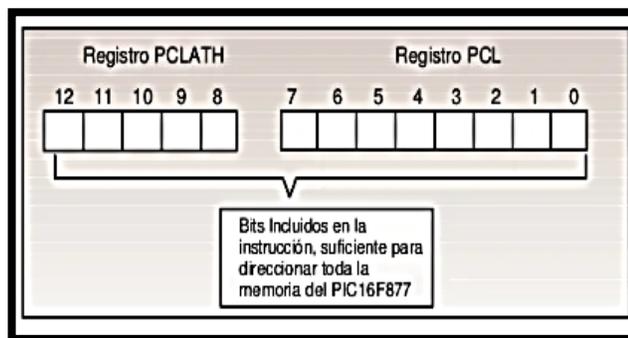


Figura 11. Registros que conforman el Contador de programa

Fuente: <https://es.slideshare.net/GULG90/pic16-f877>

En una instrucción CALL o GOTO la dirección de memoria a donde se quiere ir, es decir, los bits 12 a 0 del contador de programa se cargan desde el código de operación de la instrucción.

En algunas instrucciones donde la parte baja del contador de programa, es decir los ocho bitsbajos del registro PCL, es el destino, los otros cinco bits se cargan directamente desde el PCLATH (bits 4 a 0), por ejemplo, en el caso de la instrucción ADDWF. Esto se debetener en cuenta cuando se desea hacer lectura de tablas usando el comando: ADDWF PC,1, en este caso se debe tener en cuenta que la tabla debe estar comprendida dentro de un solo bloque de 256 bytes (0-255, 256-511, etc.).

03h o STATUS: Registro de estados. Contiene el estado aritmético de la ALU, la causa del reset y los bits de preselección de página para la memoria de datos. Los bits correspondientes a este registro. Los bits 5 y 6 (RP0 y RP1) son los bits de selección de página para el direccionamiento directo de la memoria de datos; solamente RP0 se usa en los PIC16F84. RP1 se puede utilizar como un bit de propósito general de lectura/escritura. Los bits TO y PD no se pueden modificar por un proceso de escritura; ellos muestran la condición por la cual se ocasionó el último reset.

STATUS	R/W (0)	R/W (0)	R/W (0)	R (1)	R (1)	R/W (x)	R/W (x)	R/W (x)	Características
	IRP	RP1	RP0	TO	PD	Z	DC	C	Nombre de bit
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	

Leyenda	
R/W	Bit de lectura/escritura
R	Bit de solo lectura
(0)	Después del reinicio, el bit se pone a cero
(1)	Después del reinicio, el bit se pone a uno
(x)	Después del reinicio, el estado de bit es desconocido

Figura 12. Bits del Registro de Estados.

Fuente: <https://www.mikroe.com/ebooks/microcontroladores-pic-programacion-en-c-con-ejemplos/principales-registros-sfr>

04h o FSR: Registro selector de registros. En asocio con el registro IND0, se utiliza para seleccionar indirectamente los otros registros disponibles. Si en el programa no se utilizan llamadas indirectas, este registro se puede utilizar como un registro de propósito general.

05h o PORTA: Puerto de Entrada/Salida de 6 bits. Este puerto, al igual que todos sus similares en los PIC, puede leerse o escribirse como si se tratara de un registro cualquiera, en la posición 85h y se llama TRISA. El puerto A también puede ser configurado para que trabaje como entradas análogas para el convertidor Análogo a Digital interno del microcontrolador.

06h o PORTB: Puerto de entrada/salida de 8 bits. Al igual que en todos los PIC, este puede leerse o escribirse como si se tratara de un registro cualquiera; algunos de sus pines tienen funciones alternas en la generación de interrupciones, en la dirección 86h y se llama TRISB. Puede ser configurado también para cumplir otras funciones.

07h o PORTC: Puerto de entrada/salida de 8 bits. Al igual que en todos los PIC, este puede leerse o escribirse como si se tratara de un registro cualquiera; algunos de sus pines tienen funciones alternas, en la dirección 87h y se llama TRISC. Puede ser configurado también para cumplir otras funciones.

08h o PORTD: Puerto de entrada/salida de 8 bits. Al igual que en todos los PIC, este puede leerse o escribirse como si se tratara de un registro cualquiera; algunos de sus pines tienen funciones alternas cuando se utiliza el micro en modo microprocesador, en la dirección 88h y se llama TRISD. Puede ser configurado también para cumplir otras funciones.

09h o PORTE: Puerto de Entrada/Salida de 3 bits. Este puerto, al igual que todos sus similares en los PIC, puede leerse o escribirse como si se tratara de un registro cualquiera, en la posición 89h y se llama TRISE. El puerto E también puede ser configurado para que trabaje como entradas análogas para el convertidor Análogo a Digital interno del microcontrolador o para que maneje las señales de control en el modo microprocesador.

85h o TRISA: Registro de configuración del puerto A. Como ya se mencionó, es el registro de control para el puerto A. Un “cero” en el bit correspondiente al pin lo configura como salida, mientras que un “uno” lo hace como entrada.

86h o TRISB: Registro de configuración del puerto B. Orientado hacia el control del puerto B. Son válidas las mismas consideraciones del registro TRISA.

87h o TRISC: Registro de configuración del puerto C. Orientado hacia el control del puerto C. Son válidas las mismas consideraciones del registro TRISA.

88h o TRISD: Registro de configuración del puerto D. Orientado hacia el control del puerto D. Son válidas las mismas consideraciones del registro TRISA.

89h o TRISE: Registro de configuración del puerto E. Orientado hacia el control del puerto E. Son válidas las mismas consideraciones del registro TRISA.

020h a 7Fh: Registros de propósito general. Estas 96 posiciones están implementadas en la memoria RAM estática, la cual conforma el área de trabajo del usuario. Pueden ser utilizadas para almacenar cualquier dato de 8 bits.

Registro de trabajo W. Este es el registro de trabajo principal, se comporta de manera similar al acumulador en los microprocesadores. Este registro participa en la mayoría de las instrucciones. Está directamente relacionado con la Unidad Aritmética y Lógica ALU.

2.1.5.6.2. REGISTROS DE CONTROL DEL MÓDULO CONVERTIDOR ANÁLOGO/DIGITAL

2.1.5.6.2.1. REGISTRO ADCON0

Este es un registro que permite seleccionar cuál de las entradas análogas va a ser leída y permite dar la orden de iniciar el proceso de conversión, sus ocho bits son los siguientes:

ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/DONE	-	ADON
Bit 7							Bit 0

- ❖ Bit 0 (ADON): Bit de activación del módulo. ADON = 1, Módulo A/D operando. ADON = 0, Módulo A/D desactivado.
- ❖ Bit 2 (GO/DONE): Estado de conversión: GO = 1, Empieza conversión.

GO = 0, conversión finalizada.

- ❖ Si ADON = 0, Este bit es cero.
- ❖ Bits 3, 4 y 5 (CHS0, CHS1, CHS2): Selección del canal a convertir (canal 0 - 7).
- ❖ Bits 6 y 7 (ADCS0, ADCS1): Selección del reloj de conversión.

Tabla 3. Frecuencias de conversión para el módulo A/D

ADCS1	ADCS0	FRECUENCIA DE CONVERSION
0	0	FOSC / 2
0	1	FOSC / 8
1	0	FOSC / 32
1	1	FRC

2.1.5.6.2.2. REGISTRO ADCON1

Este es un registro que permite seleccionar como se ubican los diez bits resultado de la conversión A/D y permite seleccionar cuales de los pines del puerto A trabajarán como entradas análogas y cuales como entradas digitales. Adicionalmente, permite seleccionar los voltajes de referencia del convertidor.

ADFM	-	-	-	-	PCFG2	PCFG1	PCFG0
BIT 7							BIT 0

El bit 7 (ADFM) selecciona el formato del resultado de la conversión:

Si ADFM = 1, el resultado se justifica a la derecha: Los 6 bits más significativos de ADRESH son cero.

Si ADFM = 0, el resultado se justifica a la izquierda: Los 6 bits menos significativos de ADRESL son cero.

Con los tres bits (PCFG0, PCFG1, PCFG2) se configuran los pines del puerto A como de entradas análogas o entrada/salida digital, así como la referencia de voltaje que utilizará el convertidor.

Tabla 4. Selección de los canales análogos a utilizar.

PCFG3: PCFG0	AN7 RE2	AN6 RE1	AN5 RE0	AN4 RA5	AN3 RA3	AN2 RA2	AN1 RA1	AN0 RA0	VREF+	VREF-	CHAN/REFS
0000	A	A	A	A	A	A	A	A	VDD	VSS	8/0
0001	A	A	A	A	VREF+	A	A	A	RA3	VSS	7/1
0010	D	D	D	A	A	A	A	A	VDD	VSS	5/0
0011	D	D	D	A	VREF+	A	A	A	RA3	VSS	4/1
0100	D	D	D	D	A	D	A	A	VDD	VSS	3/0
0101	D	D	D	D	VREF+	D	A	A	RA3	VSS	2/1
011x	D	D	D	D	D	D	D	D	VDD	VSS	0/0
1000	A	A	A	A	VREF+		A	A	RA3	RA2	6/2
1001	D	D	A	A	A	A	A	A	VDD	VSS	6/0
1010	D	D	A	A	VREF+	A	A	A	RA3	VSS	5/1
1011	D	D	A	A	VREF+	VREF-	A	A	RA3	RA2	4/2
1100	D	D	D	A	VREF+	VREF-	A	A	RA3	RA2	3/2
1101	D	D	D	D	VREF+	VREF-	A	A	RA3	RA2	2/2
1110	D	D	D	D	D	D	D	A	VDD	VSS	1/0
1111	D	D	D	D	VREF+	VREF-	D	A	RA3	RA2	1/2

A = Entrada Análoga

D = Entrada/Salida Digital

Cuando se completa la conversión A/D, el resultado se carga en los registros ADRESH y ADRESL (en el formato configurado por el bit ADFM).

El bit GO/DONE (ADCON0<2>) se pone en cero y el bit bandera de la interrupción A/D (ADIF) se pone en uno.

Después de que el módulo ha sido configurado, al canal seleccionado se debe hacer un muestreo antes de empezar la conversión. El tiempo requerido para el muestreo es definido como T_{ad} .

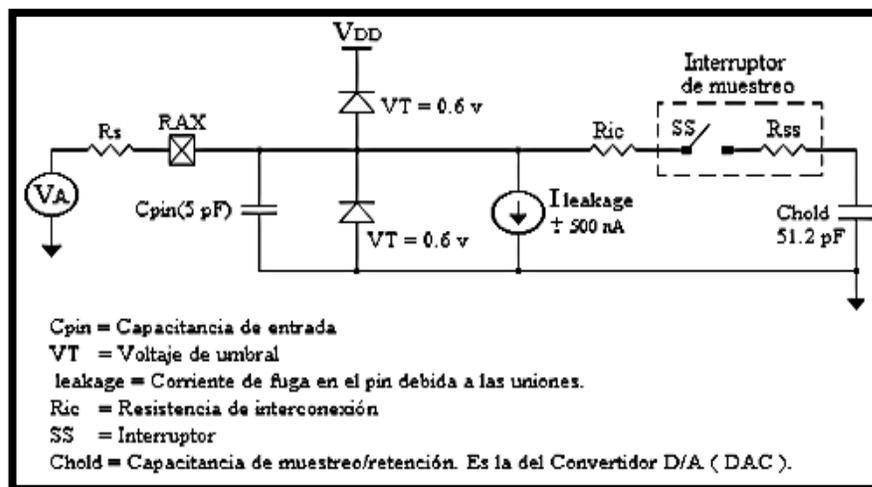


Figura 13. Modelo circuital de la Entrada Análoga.

Fuente: <https://es.slideshare.net/GULG90/pic16-f877>

2.1.5.6.3. REQUERIMIENTO PARA EL MUESTREO

Para que el convertidor A/D tenga precisión, se debe permitir que el condensador de retención se cargue con todo el nivel de voltaje del canal de entrada, se muestra el modelo de entrada análoga. La máxima impedancia recomendada para las fuentes

análogas es de 10 Kohm. Después que se selecciona el canal de entrada análoga (o es cambiado) se debe esperar un tiempo de muestreo antes de que la conversión se inicie.

2.1.5.6.4. SELECCIÓN DE LA FRECUENCIA DE CONVERSACIÓN

La conversión A/D requiere 10 Tad. La fuente del reloj de conversión es seleccionada por software. Las cuatro opciones posibles para Tad son:

- 2 T_{osc}
- 8 T_{osc}
- 32 T_{osc}
- Oscilador interno RC

Para conversiones correctas, el reloj de conversión (Tad) debe ser seleccionado para tener un tiempo mínimo Tad de 1.6 useg.

2.1.5.6.5. CARACTERÍSTICAS PRINCIPALES DEL PIC16F877A

- ❖ Memoria de programa: FLASH, 8 K de instrucciones de 14 bits c/u.
- ❖ Memoria de datos : 368 bytes RAM, 256 bytes EEPROM.
- ❖ Pila (Stack): 8 niveles (14 bits).
- ❖ Fuentes de interrupción 13
- ❖ Instrucciones 35
- ❖ Encapsulado : DIP de 40 pines.
- ❖ Frecuencia oscilador : 20 MHz (máxima)

- ❖ Temporizadores/Contadores: 1 de 8 bits (Timer 0); 1 de 16 bits (Timer 1); 1 de 8bits (Timer 2) con pre y post escalador. Un perro guardián (WDT)
- ❖ Líneas de E/S: 6 del puerto A, 8 del puerto B, 8 del puerto C, 8 del puerto D y 3del puerto E, además de 8 entradas análogas.
- ❖ Dos módulos de Captura, Comparación y PWM:
- ❖ Captura: 16 bits. Resolución máx. = 12.5 nseg.
- ❖ Comparación: 16 bits. Resolución máx. = 200 nseg.
- ❖ PWM: Resolución máx. = 10 bits.
- ❖ Convertidor Análogo/Digital de 10 bits multicanal (8 canales de entrada).
- ❖ Puerto serial síncrono (SSP) con bus SPI (modo maestro) y bus I²C (maestro/esclavo).
- ❖ USART (Universal Synchronous Asynchronous Receiver Transmitter) con dirección de detección de 9 bits.
- ❖ Corriente máxima absorbida/suministrada (sink/source) por línea (pin): 25 mA
- ❖ Oscilador: Soporta 4 configuraciones diferentes: XT, RC, HS, LP.
- ❖ Tecnología de Fabricación: CMOS
- ❖ Voltaje de alimentación: 3.0 a 5.5 V DC
- ❖ Puede operar en modo microprocesador

2.1.5.6.6. PRIMERA EJECUCIÓN DEL IDE PIC16F877A Y CONFIGURACION INICIAL PARA EL DESARROLLO DEL PROYECTO

Tras instalar el controlador de mikroC PRO for PIC, ahora realizaremos una primera

ejecución para configurar el IDE de desarrollo, la instalación del mikroC PRO for PIC es similar a la instalación de cualquier programa en Windows. Todo el procedimiento se lleva a cabo por medio de los wizards (asistentes de instalación):

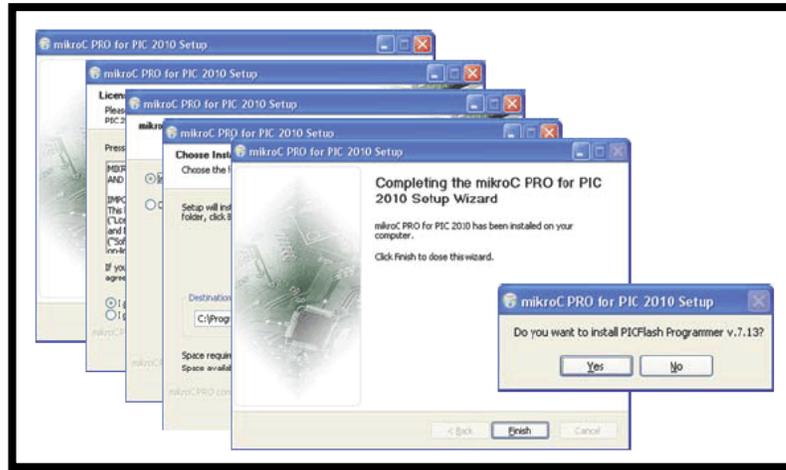


Figura 14. Primera ejecución para configurar el IDE.

Fuente: Propia.

Pulsar sobre Next, OK, Next, Next, En general, es el mismo procedimiento menos la última opción: 'Do you want to install PICFLASH v7.11 programmer.

Una vez completada la instalación del PICflash, el sistema operativo le preguntará a instalar otro programa similar, un software para programar un grupo especial de los microcontroladores PIC que funcionan en modo de bajo consumo (3.3 V). Salte este paso, el último paso - ¡la instalación del controlador (driver)!



Figura 15. Sistema operativo a instalar otro programa.

Fuente: Propia.

Pulse sobre Yes. Un controlador es un programa que permite al sistema operativo comunicar con un periférico. En este caso, este dispositivo es el programador (hardware) del sistema de desarrollo.

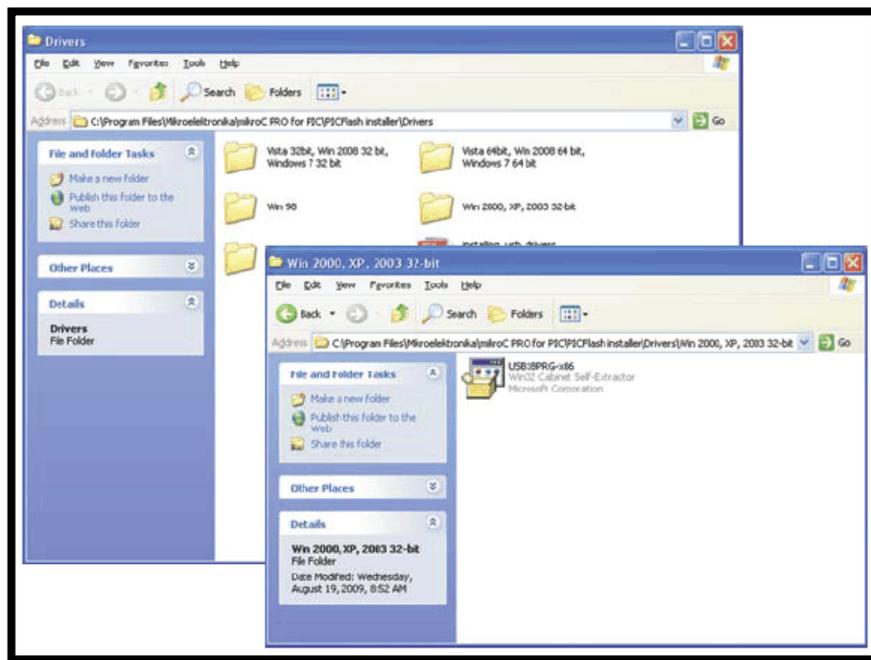


Figura 16. Carpeta del programa mikroC PRO for PIC

Fuente: Propia.

El controlador a instalar depende del sistema operativo utilizado. Seleccione el controlador correspondiente al SO (sistema operativo) utilizado (por medio de abrir la carpeta correspondiente) e inicie la instalación. Otra vez, Next, OK, Next, Next, con todo ese paso ya estaría instalado para iniciar a programar.

2.1.5.6.7. CARACTERÍSTICAS PRINCIPALES DEL MIKROC PRO FOR PIC

Al iniciar el IDE del compilador mikroC PRO for PIC por primera vez, aparecerá una ventana como se muestra a continuación:

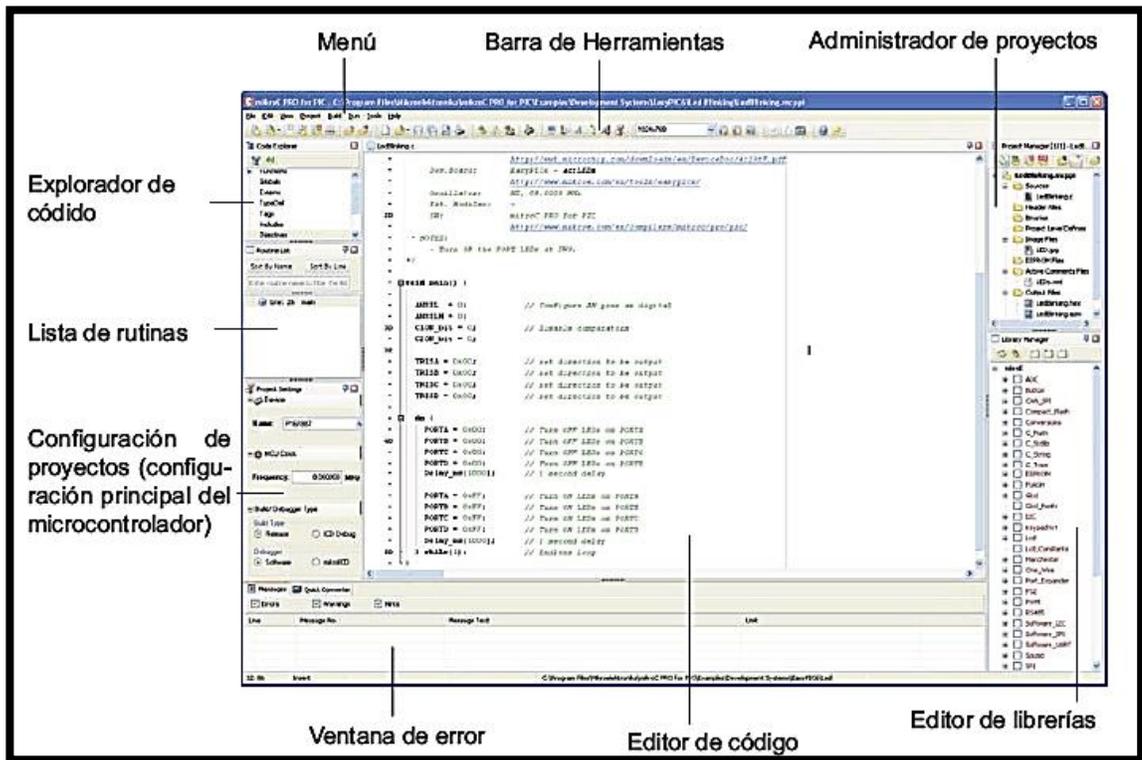


Figura 17. Características Principales del MikroC Pro For Pic

Fuente: Propia.

Una descripción detallada de todas las opciones disponibles de este IDE nos tomaría mucho tiempo. Por eso vamos a describir sólo lo más importante del compilador mikroC PRO for PIC. De todos modos, para obtener más información presione el botón de Ayuda (Help) [F1].

2.1.5.6.8. MANEJAR LOS PROYECTOS

Antes de empezar a escribir el código, usted debe crear un proyecto. Un programa escrito en el compilador mikroC PRO for PIC no es un archivo fuente autónomo, sino que forma parte de un proyecto que incluye un código hex, un código ensamblador, cabecera y otros archivos. Algunos de ellos se requieren para compilar el programa, mientras que otros se crean durante el proceso de compilación. Un archivo con extensión. mcppi le permite abrir cualquiera de estos proyectos. Para crear un proyecto, basta con seleccionar la opción Project/New Project, y un wizard aparecerá automáticamente.

2.1.5.6.8.1. PROJECT MANAGER (ADMINISTRADOR DEL PROYECTO)

Una vez creado el proyecto, es posible manejar todos los archivos que contiene al utilizar la ventana Project Manager. Basta con pulsar con el botón derecho del ratón sobre una carpeta y seleccionar la opción que necesita para su proyecto.

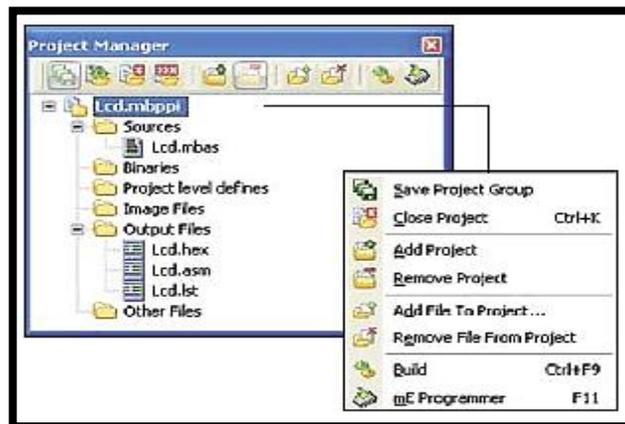


Figura 18. Ventana Project Manager.

Fuente: Propia.

2.1.5.6.8.2. LIBRARY MANAGING (EDITOR DE LIBRERIAS)

El compilador tiene que conocer todas las dependencias de su archivo fuente en mikroC para compilarlo apropiadamente. Por ejemplo, si las librerías forman parte de su proyecto, debe especificar cuáles de ellas se utilizan. Las librerías contienen un gran número de funciones listas para ser utilizadas. Las librerías en mikroC proporcionan muchas facilidades para escribir programas para los microcontroladores PIC. Abra la ventana Library Manager, y marque las que quiere utilizar en el programa. Al marcar una librería, se añade automáticamente al proyecto y se enlaza durante el proceso de la compilación. Así, no necesita incluir las librerías manualmente en sus archivos del código fuente por medio de la directiva del preprocesador #include.



Figura 19. Ventana Library Manager.

Fuente: Propia.

Por ejemplo, si su programa utiliza un LCD no hace falta escribir nuevas funciones ya que al seleccionar la librería Lcd, usted podrá utilizar funciones listas para ser utilizadas de la librería LCD (Lcd_Cmd, LCD_Init) en su programa. Si esta librería no está seleccionada en la ventana Library Manager, cada vez que intente utilizar una función de la librería LCD, el compilador le informará de un error. Una descripción de cada librería está disponible al pulsar con el botón derecho del ratón sobre su nombre y seleccionar la opción Help.

2.1.5.6.8.3. CODE EDITOR (EDITOR DE CÓDIGO)

El proceso de editar programas se debe realizar dentro de la ventana principal del IDE denominada Code Editor. Al escribir el programa no se olvide de los comentarios. Los comentarios son muy importantes para depurar y mejorar el programa. Además, aunque el compilador no tenga las restricciones de formateo, siempre debe seguir a las mismas reglas de editar (como en los ejemplos proporcionados en este libro). Como no hay limitaciones de tamaño, no vacile en utilizar los espacios en blanco para hacer su código más legible.

```
30 //kit oldstate; // Old state flag
31
32 main() {
33     ANSEL = 0; // Configure AN pins as digital I/O
34     ANSELH = 0; // Disable comparators
35     C1ON_bit = 0; // Disable comparators
36     C2ON_bit = 0;
37
38     TRISB0_bit = 1; // set RB0 pin as input
39
40     TRISC = 0x00; // Configure PORTC as output
41     PORTC = 0xAA; // Initial PORTC value
42     oldstate = 0;
43
44     do {
45         if (Button(&PORTB, 0, 1, 1)) { // Detect logical one
46             oldstate = 1; // Update flag
47         }
48         if (oldstate << Button(&PORTB, 0, 1, 0)) { // Detect one-to-zero transition
49             PORTC = ~PORTC; // Invert PORTC
50             oldstate = 0; // Update flag
51         }
52     } while(1); // Endless loop
53 }
```

Figura 20. Editor de Código

Fuente: Propia.

Al escribir un programa, no espere que termine la redacción del programa para compilarlo. Compile su código de forma regular con el propósito de corregir cuánto más errores de sintaxis. Asimismo, usted puede compilar su programa cada vez que se complete la redacción de una nueva función, así como probar su comportamiento al utilizar modo de depuración. De este modo, resulta más fácil solucionar los errores de programa para no tomar un camino erróneo en redactar su programa. De lo contrario, usted tendrá que editar el programa entero.

2.1.5.6.8.4. COMPILAR Y SOLUCIONAR LOS ERRORES

Para compilar su código, pulse sobre la opción Build en el menú Project. En realidad, el proyecto entero se ha compilado, y si la compilación se ha realizado con éxito, se generarán los archivos de salida (asm, hex etc.). Una compilación se ha realizado con éxito

si no se ha encontrado ningún error. Durante el proceso de compilación se generan muchos mensajes que se visualizan en la ventana Messages. Estos mensajes consisten en información, advertencia y errores. Cada error encontrado se asocia con su línea de programa y su descripción. Como un error en su código puede generar mucho más error, simplemente debe intentar solucionar el primer error en la lista y después recompila su programa. En otras palabras, es recomendable solucionar los errores uno a uno.

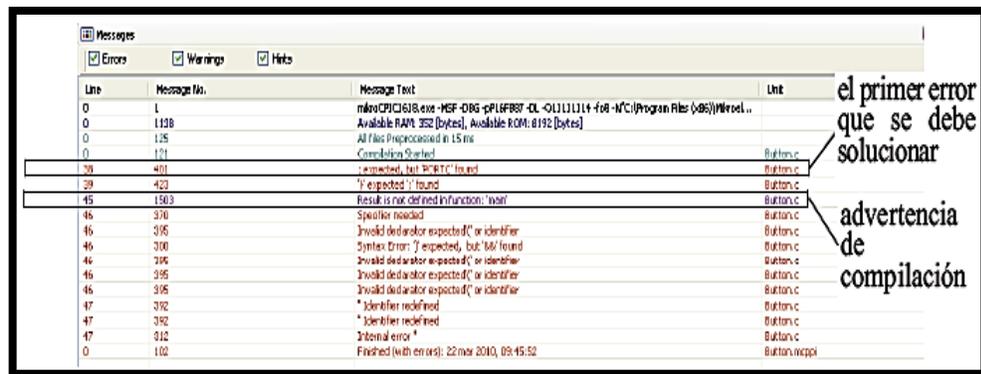


Figura 21. Compilación y solución de los errores.

Fuente: Propia.

En el ejemplo anterior hay dos errores y una advertencia: faltan un punto y coma y una declaración de variable la advertencia le informa que falta el tipo del valor devuelto de la función main. La compilación le permite corregir su programa por medio de solucionar todos los errores en mikroC. Cuando todos los errores se solucionen, su programa está listo para ser cargado en el microcontrolador. De todas formas, su tarea todavía no está terminada, porque aún no sabe si su programa se comporta como se esperaba o no.

2.1.5.6.9. GRABAR CON PICKIT 2

El uso del quemador de PIC es muy sencillo, se recomienda deshabilitar la opción

Tools > Fast programming para reducir la probabilidad de errores durante la grabación de los dispositivos, debido a la incapacidad de algunos componentes electrónicos del programador para conmutar a altas frecuencias. El procedimiento indicado también se puede aplicar a otros dispositivos soportados.

a) Conecte el microcontrolador al grabador de PIC según el modelo específico. La programación se realiza a través de los pines VPP (voltaje de programación), VDD (voltaje de alimentación), VSS (referencia), PGD (datos) y PGC (reloj) del programador y los pines correspondientes del microcontrolador. Si el programador no dispone de zócalo para la conexión de los microcontroladores, debe emplear un tablero de proyectos (Protoboard) para realizar las conexiones.

b) Conecte el cable USB al grabador y el otro extremo del cable a un puerto USB del computador.

c) Inicie el programa de aplicación PICKkit2 v2.61.

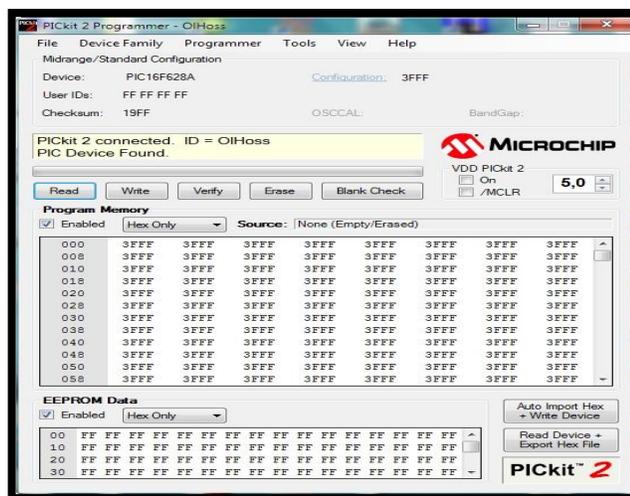
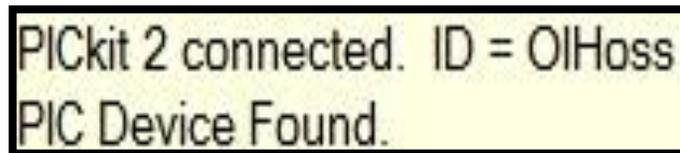


Figura 22. Ventana del quemador de PIC.

Fuente: Propia.

d) El programador de PIC será detectado automáticamente, al igual que el dispositivo a ser programado. Para que la auto detección del dispositivo tenga efecto, el comando Programmer->Manual Device Select debe estar desactivado. Por ejemplo, para el PIC16F877A se verá Device: PIC16F877A y adicionalmente el siguiente mensaje:



PICkit 2 connected. ID = 01Hoss
PIC Device Found.

Figura 23. Mensaje del Pic Device.

Fuente: Propia.

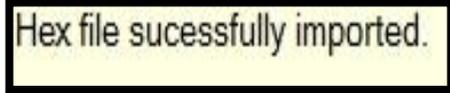
Si el dispositivo no soporta autodetección, debe ser seleccionado manualmente, para lo cual el comando Programmer->Manual Device Select debe estar activado. Por ejemplo, para seleccionar manualmente el PIC16F877A se debe seleccionar el comando Device Family->Midrange->Standard, y a continuación seleccionar el PIC de la lista desplegable Device:



Figura 24. Lista desplegable Device

Fuente: Propia.

e) Con el comando File->Import Hex abra el archivo ejecutable (*.hex) que va a ser grabado en el dispositivo. Deberá observar el siguiente mensaje:



Hex file sucessfully imported.

Figura 25. Mensaje del grabado en el dispositivo.

Fuente: Propia.

f) Programe el dispositivo haciendo clic en el botón Write. Espere hasta que aparezca el mensaje de programación exitosa:



Programming Successful.

Figura 26. Mensaje de programación exitosa.

Fuente: Propia.

g) Desconecte el cable USB del computador y extraiga el dispositivo programado.

h) Si va a programar otro dispositivo, conéctelo al quemador de PIC, conecte nuevamente el cable USB al computador y luego seleccione el comando Tools->Check Communication..

2.1.6. MODULO BLUETOOTH HC-06

El módulo Bluetooth HC-06 nos permite conectar nuestros proyectos con PIC16F877A a un smartphone, celular o PC de forma inalámbrica (Bluetooth), con la facilidad de operación de un puerto serial. La transmisión se realiza totalmente en forma transparente al programador, por lo que se conecta en forma directa a los pines seriales de nuestro microcontrolador preferido (respetando los niveles de voltaje, ya que el módulo se

alimenta con 3.3V). Todos los parámetros del módulo se pueden configurar mediante comandos AT. La placa también incluye un regulador de 3.3V, que permite alimentar el módulo con un voltaje entre 3.6V - 6V. Este módulo es el complemento ideal para nuestros proyectos de robótica, domótica y control remoto con Arduino, PIC, Raspberry PI, ESP8266, ESP32, STM32, etc.

Incluye la tecnología inalámbrica Bluetooth en tus proyectos electrónicos con el módulo HC-06 esclavo, todo de forma sumamente sencilla y sin requerir ser un experto en electrónica.

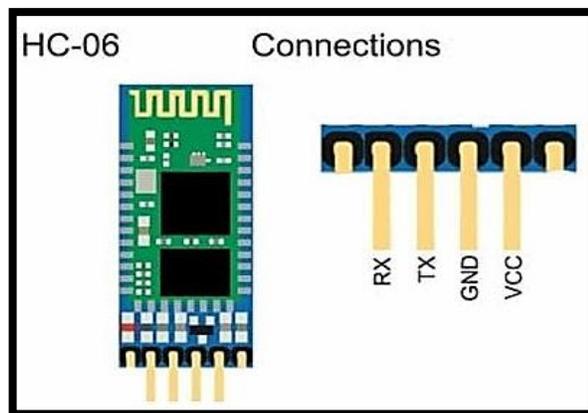


Figura 27. Módulo Bluetooth HC-06

Fuente: <https://uelectronics.com/producto/modulo-bluetooth-hc-06/>

En este módulo todas las señales necesarias para establecer un enlace serial se encuentran disponibles en cómodos headers estándar que pueden usarse en protoboard o tarjetas perforadas para armar tus proyectos.

El módulo HC-06 ofrece el servicio de puerto serie (RFCOMM), creando un enlace de datos transparente entre una PC, celular, tablet o cualquier dispositivo con Bluetooth y el microcontrolador. La salida del módulo es una señal serial asíncrona que puede ser

recibida e interpretada fácilmente por cualquier microcontrolador.

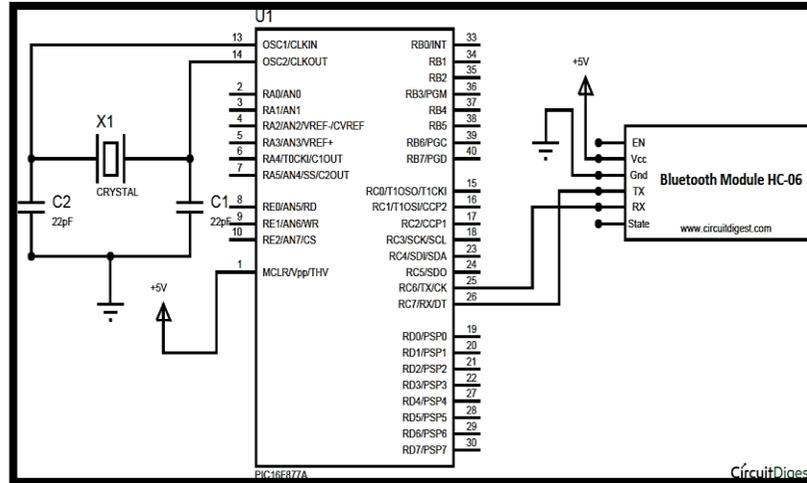


Figura 28. Conexión Módulo Bluetooth HC-06 y microcontrolador pic16f877a

Fuente: <https://circuitdigest.com/microcontroller-projects/bluetooth-interfacing-with-pic-microcontroller>

El bluetooth HC-06 funciona correctamente con la mayoría de los teléfonos móviles con sistema operativo android, pero no así con sistemas iOS.

2.1.6.1. CARACTERISTICAS

- Funciona como dispositivo esclavo bluetooth
- Configurable mediante comandos AT
- Bluetooth V2.0+EDR
- Frecuencia de operación: 2.4 GHz Banda ISM
- Modulación: GFSK (Gaussian Frequency Shift Keying)
- Potencia de transmisión: ≤ 4 dBm, Class 2

- Sensibilidad: $\leq -84\text{dBm}$ @ 0.1% BER
- Seguridad: Autenticación y encriptación
- Perfiles Bluetooth: Puerto serie bluetooth.
- Distancia de hasta 10 metros en condiciones óptimas
- Voltaje de Operación: 3.6 VDC a 6 VDC
- Consumo Corriente: 30 mA a 50mA
- Chip: BC417143
- Versión de firmware: 3.0-20170609
- Baudios por defecto: 9600
- Baudios soportados: 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200
- Interfaz: Serial TTL
- Antena: Integrada en el PCB
- Seguridad: Autenticación y encriptación (Contraseña por defecto: 1234)
- Temperatura de trabajo (Max): 75°C
- Temperatura de trabajo (Min): -20°C
- Dimensiones: 4.4 x 1.6 x 0.7 cm

2.1.6.2. DIFERENCIAS ENTRE EL MODULO BLUETOOTH HC-06 Y EL HC-05

Los módulos bluetooth HC-05 y HC-06 son elementos que te permitirán comunicar tu microcontrolador por protocolos bluetooth con cualquier otro elemento que cuente con el mismo protocolo, desde un celular hasta un portátil o incluso otro módulo bluetooth conectado a otro microcontrolador.

La principal diferencia entre estos dos módulos reside en que el HC-05 puede ser configurado para ser usado como dispositivo maestro o esclavo, mientras que el HC-06 sólo puede ser usado como esclavo. Ambos dispositivos pueden ser configurados para cambiar el nombre del dispositivo, cambiar su clave de apareamiento, el nombre por defecto es HC 05 (ó HC 06) y la clave por defecto es: 1234. Se debe mencionar que toda configuración a los dispositivos debe ser hecha por comandos AT y que ambos módulos pueden alimentarse con 5 o 3.3 V.

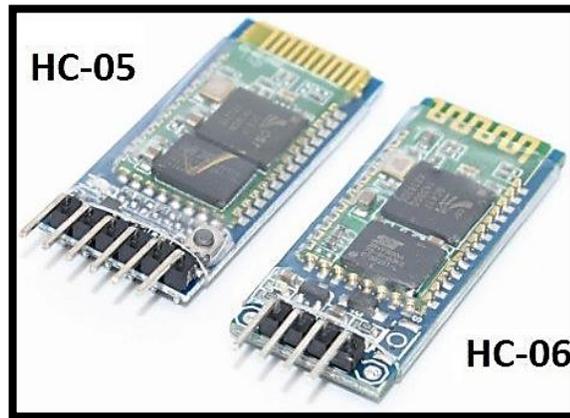


Figura 29. Diferencias entre los módulos bluetooth HC-05 y HC-06

Fuente: <https://arduinoofacil.com/65-modulo-bluetooth-hc-06/>

Una vez apareado el dispositivo con, por ejemplo, tu celular Android (a la fecha no hay una aplicación IOS compatible con estos módulos) podrás enviar comandos como letras o caracteres desde tu celular para controlar funciones en tu programa. Existen muchas aplicaciones en la Playstore de Google que pueden servirte para este fin, sólo basta con ingresar "Bluetooth Arduino" en el buscador y buscar la que necesites, algunas incluso son modificables de forma que puedes adaptar los botones en la aplicación de acuerdo a

tu programa para tener un control totalmente personalizado.

Estos módulos se comunican por medio del puerto serial, por lo tanto, para programar tu placa, el módulo debe estar desconectado en primera instancia. Luego, la comunicación por puerto serial nos facilita en gran manera obtener datos desde el módulo, ni necesitamos siquiera instalar librería alguna, sólo basta con verificar si hay datos entrando al Buffer del puerto serial, si los hay, es porque el módulo está recibiendo información, por lo cual leemos la información leyendo los datos en el Buffer.

2.1.6.3. MODULO BLUETOOTH HC-06 Y PIC 16F877A

El bluetooth es un estándar de comunicación que permite la transmisión de datos de radiofrecuencia en la banda de 2.4 GHz. Existen muchos módulos Bluetooth para proyectos de Electrónica, dos modelos más usados para PIC16F877A son el HC05 para transmisión y recepción y HC06 solamente para recepción.

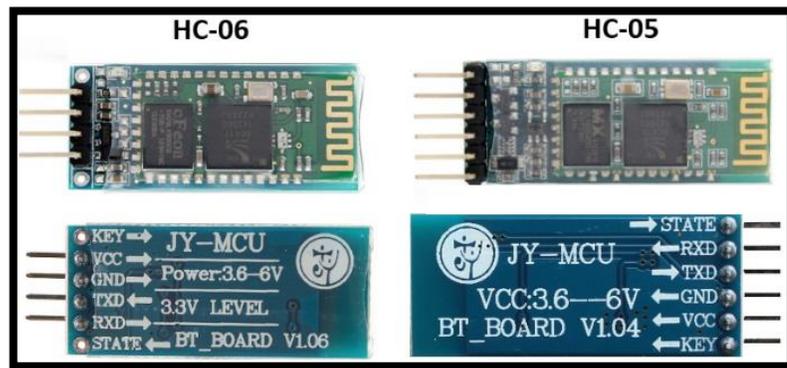


Figura 30. Modulo bluetooth HC-05 y Modulo bluetooth HC-06.

Fuente: <https://sites.google.com/site/bluetoothypic16f877amicrocode/>

Para que el enlace entre el PIC-Bluetooth y el celular Android es necesario tener una aplicación. Se puede bajar de algún sitio gratis o puede elaborarse una usando appInventor, así se genera la aplicación conforme al diseño con el que se pretende hacer el control, personalizando cada botón a convenientemente.

2.1.6.4. CONEXIÓN DE MODULO BLUETOOTH HC-06 DESDE UN CELULAR

Primero tenemos que instalar una aplicación que haga la función de un Terminal Serial, existen otras aplicaciones, pero con ésta aplicación se realizará ejemplo. Antes de empezar a usar la aplicación, debemos vincular nuestro celular con el HC-06, esto al igual que en la PC solo se hace una sola vez, para esto vamos a Ajustes->Bluetooth y Buscar dispositivos, en la búsqueda nos debería aparecer nuestro HC06, al seleccionarlo nos pedirá el código de vinculación, que como recordamos lo explicado al inicio generalmente es 1234.

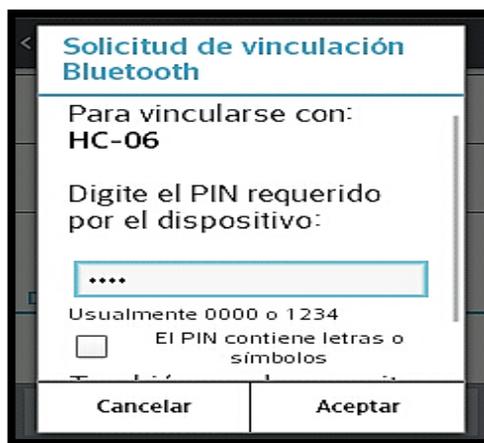


Figura 32. Vinculación del dispositivo móvil con el Módulo bluetooth HC-06.

Fuente: Propia.

Una vez vinculado abrimos la aplicación, antes de hacer la conexión, con el botón menú entramos a preferencias y activamos la opción de Echo Local para que se muestre en pantalla los caracteres que se ingresan por teclado, que son los que se envían.

Luego entramos a menú ->conectarse a un dispositivo.



Figura 33. Conexión del dispositivo móvil y el Módulo bluetooth HC-06.

Fuente: Propia.

En la lista muestra todos los dispositivos vinculados en nuestro celular, escogemos el HC-06, La aplicación intentará conectar con el HC-06, si logra la conexión en la parte superior debe salir el estado de conectado.



Figura 34. Selección del dispositivo móvil con el Módulo bluetooth HC-06.

Fuente: Propia.

Una vez conectado, podemos empezar a enviar los datos, es bueno mencionar que la aplicación envía el carácter tan pronto se presione la tecla.

2.1.7. MODULO BOMBA DE AGUA

Una bomba de agua es una máquina hidráulica que permite incrementar la energía cinética de un caudal de agua.

Las bombas hidráulicas son elementos ampliamente conocidos y empleados en la industria desde antaño, y constituyen toda una rama de la técnica. Existe una gran variedad de bombas, que abarcan un amplio rango de potencias y características hidráulicas.



Figura 35. Modulo Bomba de Agua.

Fuente: <https://www.luisllamas.es/bomba-de-agua-con-arduino/>

Independientemente de sus características o potencia, siempre podemos controlar un equipo de bombeo mediante un procesador, siendo de hecho frecuente que estén controlados por un autómata. PIC16F877A, por supuesto, no es una excepción, y podemos encender cualquier tipo de bomba de agua mediante las salidas digitales y el uso de un MOSFET o una salida por relé.

Las bombas hidráulicas son componentes fundamentales en instalaciones e infraestructuras, en sistemas de abastecimiento y depuración de agua y sistemas de climatización. Industrialmente, forman parte de un sin fin de equipamientos e impulsan todo tipo de fluidos.

2.1.7.1. COMO FUNCIONA UNA BOMBA DE AGUA

Una bomba de agua consta de un accionamiento, en la mayoría de los casos un motor eléctrico, acoplado a un elemento rotativo denominado rodete.

El rodete está formado por álabes que, al girar, transmiten parte de la energía al fluido que lo atraviesa. Normalmente los álabes están curvados formando una guía para las partículas, y su forma determina la cantidad de energía que se transmite al fluido y el grado en que esta se reparte entre velocidad o presión.

Sin embargo, en algunas bombas de muy pequeña potencia los álabes son rectos, formando una simple aspa.

En las bombas axiales, el agua entra en la bomba por el centro del rodete, incrementa su energía a medida que lo atraviesa girando con el mismo, y finalmente abandona la bomba en sentido tangencial.

Existen bombas que emplean otro tipo de fluidos, en lugar de agua. Por ejemplo, existen bombas para mover hidrocarburos, aceites, o disoluciones. Sin embargo, aunque la teoría dice que cualquier máquina hidráulica puede funcionar al cambiar el fluido que impulsa, en el mundo real deberemos comprobar en las especificaciones que la bomba está preparada para el tipo de fluido que vamos a emplear.

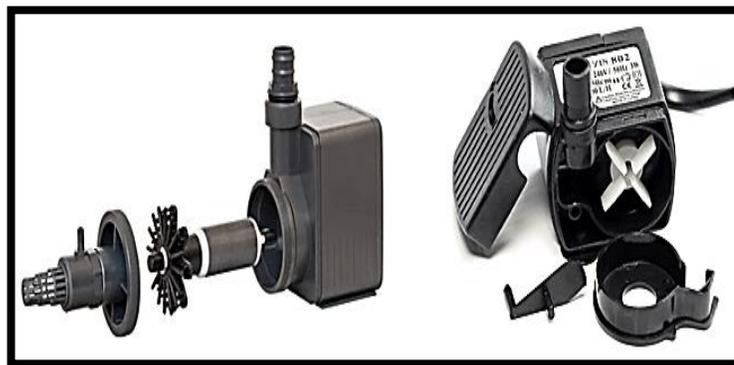


Figura 36. Modulo Bomba de Agua visto por dentro.

Fuente: <https://www.luisllamas.es/bomba-de-agua-con-arduino/>

También podemos clasificar las bombas entre sumergibles y bombas no sumergibles.

Las bombas sumergibles el motor se encuentra sellado en un encapsulado, por lo que toda la bomba se introduce en el fluido, lo que evita tener una tubería de aspiración.

En las bombas no sumergibles el motor no está impermeabilizado, por lo que no puede ser introducido dentro del fluido. Por tanto, necesitan una tubería de admisión, que debe cumplir unas determinadas de condiciones para que la bomba funcione correctamente.

2.1.7.2. CARACTERISTICAS

- ✚ Voltaje de operación: 3V – 6V
- ✚ Elevación máxima: 40-110cm / 15.75"-43,4"
- ✚ Tasa de flujo : 80-120L/H
- ✚ Diámetro de salida del agua: 7.5mm / 0.3"
- ✚ Diámetro de entrada del agua: 5mm / 0.2"
- ✚ Diámetro: aprox. 24mm / 0.95"
- ✚ Tiempo de trabajo: lapsos cortos (no mayores a 20 min/hora)
- ✚ Vida útil: 500 horas

2.1.8. MÓDULO PUENTE H L298N

El módulo controlador de motores L298N H-bridge nos permite controlar la velocidad y la dirección de dos motores de corriente continua o un motor paso a paso de una forma muy sencilla, gracias a los 2 los dos H-bridge que monta.

Ya hemos hablado de ellos antes, pero básicamente un puente-H o H-bridge es un

componente formado por 4 transistores que nos permite invertir el sentido de la corriente, y de esta forma podemos invertir el sentido de giro del motor.

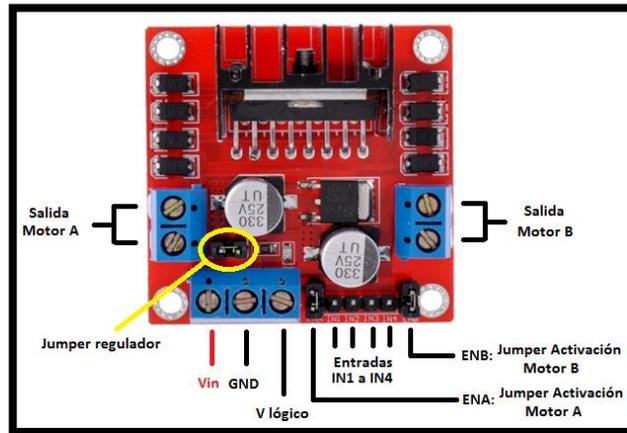


Figura 37. Módulo puente H L298N y sus partes.

Fuente: <https://www.prometec.net/l298n/>

El rango de tensiones en el que trabaja este módulo va desde 3V hasta 35V, y una intensidad de hasta 2A. A la hora de alimentarlo hay que tener en cuenta que la electrónica del módulo consume unos 3V, así que los motores reciben 3V menos que la tensión con la que alimentemos el módulo.

Además, el L298N incluye un regulador de tensión que nos permite obtener del módulo una tensión de 5V, perfecta para alimentar nuestro PIC16F877A. Eso sí, este regulador sólo funciona si alimentamos el módulo con una tensión máxima de 12V.

Es un módulo que se utiliza mucho en proyectos de robótica, por su facilidad de uso y su reducido precio.

2.1.8.1. CONEXIÓN Y FUNCIONAMIENTO

La entrada de tensión V_{in} admite tensiones entre 3V y 35V, y justo a su derecha en la imagen tenemos el pin que debemos conectar a GND.

La tercera conexión de ese grupo V lógico puede funcionar de dos maneras:

- Si el jumper del regulador está cerrado activaremos el regulador de tensión del L298N, y en V lógico tendremos una salida de 5V, que podremos usar para lo que queramos, por ejemplo, para alimentar una placa Arduino.
- Si el quitamos el jumper desactivaremos el regulador, necesitaremos alimentar la parte lógica del módulo, así que tendremos que meter una tensión de 5V por la conexión V lógico para que el módulo funcione.
- Cuidado si introducimos corriente por V lógico con el jumper de regulación puesto podríamos dañar el módulo.
- Además, el regulador sólo funciona con tensiones hasta 12V en V_{in} , por encima de este valor tendremos que quitar el jumper y alimentar la parte lógica del módulo desde otra fuente.
- El resto de conexiones se usan de una u otra forma dependiendo si vamos a manejar dos motores de continua o un motor paso a paso. En esta sesión nos vamos a centrar en el control de motores DC.

Las salidas para los motores A y B nos darán la energía para mover los motores. Tener en cuenta la polaridad al conectarlos, para que cuando más tarde hagamos que se muevan adelante, funcionen como deberían. Si no fuera así, no tendríamos más que invertir las

conexiones.

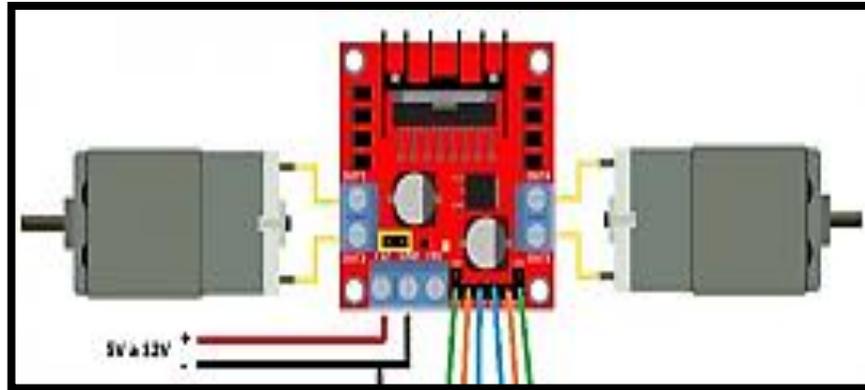


Figura 38. Conexión del Modulo puente H L298N y los motores DC.

Fuente: <https://www.prometec.net/l298n/>

Los pines IN1 e IN2 nos sirven para controlar el sentido de giro del motor A, y los pines IN3 e IN4 el del motor B. Funcionan de forma que, si IN1 está a HIGH e IN2 a LOW, el motor A gira en un sentido, y si está IN1 a LOW e IN2 a HIGH lo hace en el otro. Y lo mismo con los pines IN3 e IN4 y el motor B.

Para controlar la velocidad de giro de los motores tenemos que quitar los jumpers y usar los pines ENA y ENB. Los conectaremos a dos salidas PWM de PIC16F877A de forma que le enviemos un valor entre 0 y 255 que controle la velocidad de giro. Si tenemos los jumpers colocados, los motores girarán a la siempre a la misma velocidad.

2.1.8.2. CARACTERISTICAS

- ❖ Chip: L298N
- ❖ Canales: 2 (soporta 2 motores DC o 1 motor PAP)

- ❖ Voltaje lógico: 5V
- ❖ Voltaje de Operación: 5V-35V
- ❖ Consumo de corriente (Digital): 0 a 36mA
- ❖ Capacidad de corriente: 2A (picos de hasta 3A)
- ❖ Potencia máxima: 25W
- ❖ Peso: 30g
- ❖ Dimensiones: 43 * 43 * 27 mm
- ❖ Voltaje de alimentación, mínimo de 5 V. Posee dos entradas, una de 5V para controlar la parte lógica y otra para alimentar las salidas al motor, que pueden ser de 5V o más.
- ❖ Posee un regulador de voltaje de 5V 78M05, para alimentar la etapa lógica del L298N, sin embargo, cuando la alimentación supera los 12V, se recomienda, utilizar una fuente de 5V externa como fuente de alimentación.
- ❖ Admite entradas de señal PWM para el control de velocidad.
- ❖ Posee 8 diodos de protección contra corriente inversas.

2.2. DESARROLLO PRACTICO / EXPERIMENTAL

El proyecto que se realizo es un prototipo de dispensador de bebidas que ayudara al BAR AQUA para que tenga mejores ganancias en reducción de tiempos. Tiene la operación de que ni bien se prende el Microcontrolador PIC16F877A ya podremos conectarnos inalámbricamente con el dispositivo móvil y el modulo bluetooth, para que pueda mandar datos a la mezcladora y esta pueda elegir que bebida desea que se lo mezcle.

2.2.1. DIAGRAMA DE BLOQUES

El siguiente diagrama de bloques muestra los pasos para el funcionamiento de la mezcladora de bebidas a desarrollar.

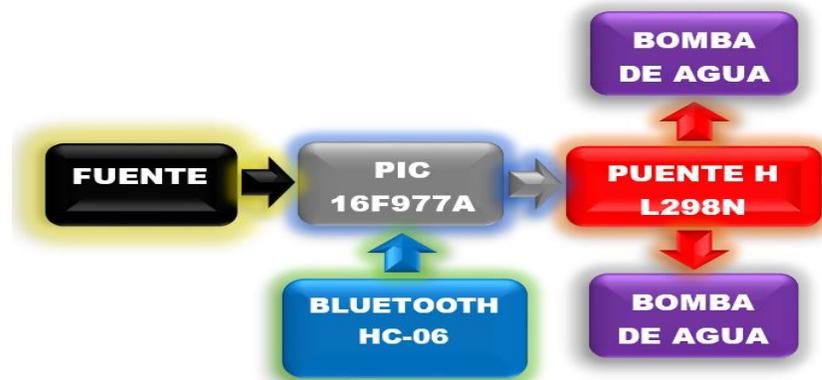


Figura 39. Diagrama de bloques, funcionamiento del prototipo.

Fuente: Propia

2.2.2. COMPONENTES QUE SE UTILIZO PARA EL PROYECTO

2.2.2.1. PIC 16F877A

En este Microcontrolador subiremos el código necesario para el control de todo el proyecto. Además se va a encargar de alimentar todo el proyecto con 5 voltios.

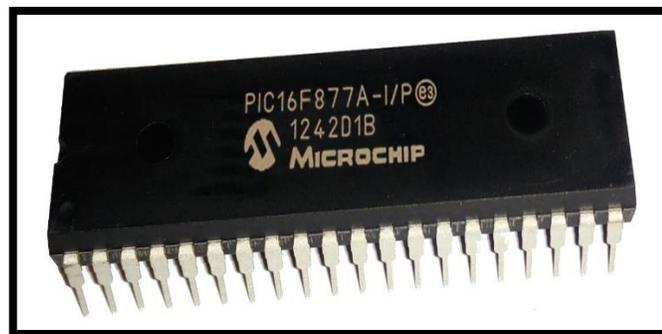


Figura 40. Micro Controlador Pic 16f877a.

Fuente: <https://apmelectronica.com/producto/pic-16f877a/>

2.2.2.2. MODULO BLUETOOTH HC-06

Este módulo bluetooth hc-06 nos permitirá conectarnos al proyecto con el Microcontrolador PIC16F877A a un dispositivo móvil de forma inalámbrica.

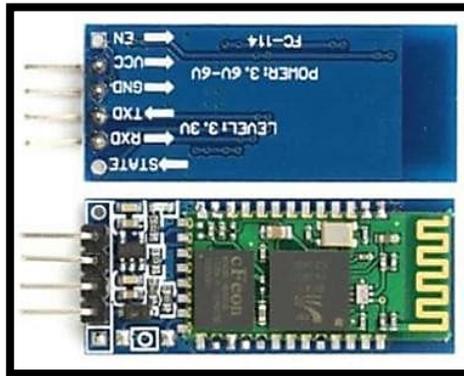


Figura 41. Módulo Bluetooth hc-06.

Fuente: <https://uelectronics.com/producto/modulo-bluetooth-hc-06/>

2.2.2.3. MODULO BOMBA DE AGUA

Este modulo bomba de agua es un dispositivo capaz de generar un caudal de líquido usando energía cinética, dispone de unos elementos básicos: entrada por donde se absorbe el líquido, Motor+Hélice el encargado de generar la energía cinética que extrae el agua desde la entrada y la envía a través de la salida y una salida por donde saldrá el líquido propulsado por la potencia de la bomba de agua, en la cual este módulo bomba de agua nos servirá para la mezcla de las bebidas.



Figura 42. Modulo Bomba de Agua.

Fuente: <https://www.importacioneswest.com/producto/bomba-de-agua-sumergible-dc-12v/>

2.2.2.4. MODULO PUENTE H L298N

Este driver puente H L298N es un módulo para manejar motores DC tiene un chip L298N internamente posee dos puentes H completos que permiten controlar 2 motores DC o un motor paso a paso, en el proyecto lo utilizaremos para el giro de motor del módulo bomba de agua.

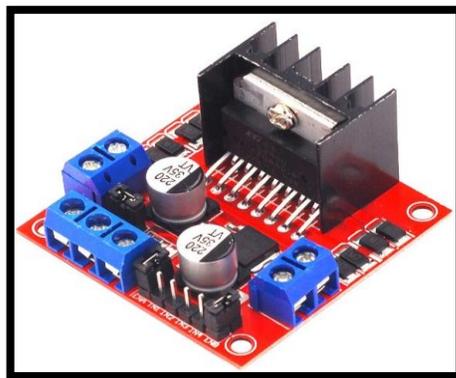


Figura 43. Modulo Puente H L298N.

Fuente: <https://www.okybolivia.com/product/puente-h-l298n/>

2.2.3. DESARROLLO DEL PROTOTIPO

Todo el circuito funciona con 5 voltios, para obtener esta tensión conectamos el Microcontrolador PIC16F877A a un cable USB. El módulo HC-06, es un bluetooth, que soporta conexiones inalámbricas.

Este módulo HC-06 están montados sobre una interfaz en la que se incluye cuatro pines para su conexión, Vcc, Gnd, Txd y Rxd, además cuentan con un LED para indicar el estado de conexión del Bluetooth; si el LED parpadea es que no está emparejado, si este está activado de forma continua el Bluetooth esta emparejado. El pin Tx va conectado al pin número 26 del PIC16F877A, mientras que el pin Rx va al pin 25.

El modo de operación es mediante comunicación inalámbrica, donde ni bien se le mande un dato de la Aplicacion al módulo bluetooth este se comunicará con el Microcontrolador para que este le mande datos al módulo bomba de agua y así se haga la mezcla de la bebida que el cliente desee.

Para que pueda funcionar el modulo bomba de agua se necesitara del modulo puente H L298N para que este del giro del motor de la bomba de agua.

2.2.4. CODIGO PIC16F877A

```
char dato;  
  
void cocacola();  
void whisky();  
void ron();  
void agua();  
void roncola();  
void whisagua();  
  
void main() {  
    CMCON = 0x07;  
    TRISA = 0x00;  
    PORTA = 0x00;  
    UART1_Init(9600);  
    Delay_ms(100);  
    while(1){  
        if(UART1_Data_Ready() > 0){  
            dato = UART1_Read();  
        }  
        switch (dato) {  
            case 'G':  
                cocacola();  
                break;  
            case 'H':  
                whisky();  
                break;
```

```
    case 'J':
        ron();
        break;
    case 'I':
        agua();
        break;
    case 'K':
        roncola();
        break;
    case 'L':
        whisagua();
        break;
}
}
}
```

```
void cocacola(){
    RA2_bit = 0;
    RA3_bit = 1;
    RA0_bit = 1;
    RA1_bit = 0;
}
```

```
void whisky(){
    RA2_bit = 1;
    RA3_bit = 0;
    RA0_bit = 1;
    RA1_bit = 0;
}
```

```
void ron(){  
    RA2_bit = 0;  
    RA3_bit = 1;  
    RA0_bit = 0;  
    RA1_bit = 1;  
}
```

```
void agua(){  
    RA2_bit = 0;  
    RA3_bit = 0;  
    RA0_bit = 0;  
    RA1_bit = 0;  
}
```

```
void roncola(){  
    RA2_bit = 1;  
    RA3_bit = 0;  
    RA0_bit = 0;  
    RA1_bit = 1;  
}
```

```
void whisagua(){  
    RA2_bit = 1;  
    RA3_bit = 0;  
    RA0_bit = 0;  
    RA1_bit = 1;  
}
```

2.2.5. CIRCUITO FINAL

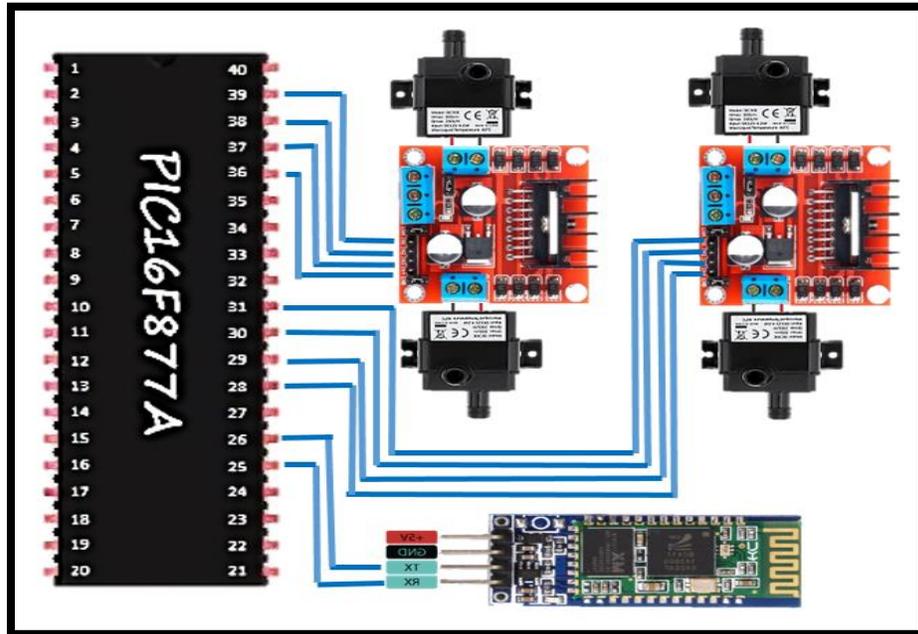


Figura 44. Armado del Circuito.

Fuente: Propia.

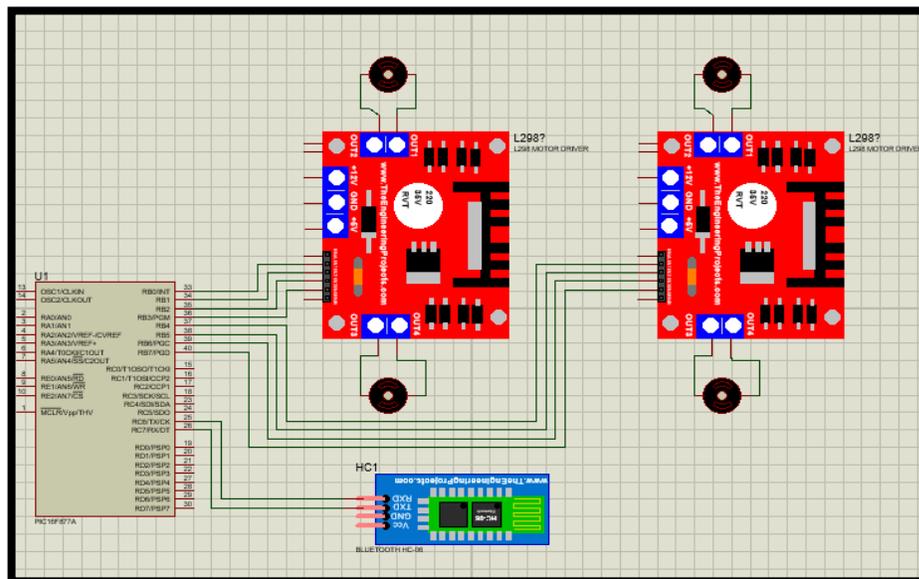


Figura 45. Armado del Circuito en Proteus.

Fuente: Propia

CAPITULO III

3. ANALISIS DE COSTOS

3.1. COSTOS FIJOS

El proyecto de grado de licenciatura cuenta con los siguientes costos fijos vale la pena hacer notar que los costos de estos componentes no tienen un efecto directo en cuanto al costo final del proyecto, la cual se utilizó un prototipo de dispensador de bebidas hecho de madera para el proyecto.

Tabla 5. Costos fijos del proyecto

DESCRIPCION	COSTOS Bs.
PROTOTIPO DE DISPENSADOR	150
TOTAL	150

3.2. COSTOS VARIABLES

Debido que los componentes electrónicos sufren constantes variaciones en su precio final; el costo total del proyecto presentado puede llegar a tener ligeros cambios con el pasar de los días. A continuación, se detalla el costo final de cada uno de los componentes al momento de ser adquiridos tanto de proveedores locales como de proveedores en el interior del país.

Tabla 6. Costos variables del proyecto

DESCRIPCION	COSTO Bs.
Microcontrolador Pic16f877a	50
Modulo Bluetooth Hc-06	60
Modulo Bomba de Agua	60
Modulo Puente H L298n	30
Cristal 16 Mhz	6
Capacitor Cerámico	0.50
Placa Virgen	5
TOTAL	211.50

CAPITULO IV

4. CONCLUSIONES Y RECOMENDACIONES

4.1. CONCLUSIONES

- Utilizar los materiales adecuados que garantice la no presencia de agentes contaminantes e higiene a la hora de la mezcla de las diferentes bebidas.
- El uso de circuitos permite la fácil conexión entre todos los dispositivos electrónicos que hacen parte de la mezcladora de bebidas.
- La simulación de los circuitos en software especializado permite tener el valor de corriente y voltaje que pasa por los diferentes componentes para evitar prevenir daños en el diseño electrónico.
- El desarrollo de la aplicación móvil permite un uso interactivo entre el usuario y la máquina del mismo modo hace más eficiente la solicitud de las diferentes bebidas disponibles en esta.
- La máquina presenta un incremento en el tiempo de preparación de las bebidas a medida que la botella del trago que se requiere se encuentra más lejos del home del prototipo.

4.2. RECOMENDACIONES

- ❖ Se recomienda hacer limpieza a diario o en periodos constantes de tiempo, con el fin de mantener la asepsia adecuada en toda la estructura y los componentes que hacen parte de la misma de tal manera que no se almacenen bacterias o líquidos en lugares no deseados.

- ❖ Se debe prevenir la humedad sobre las instalaciones eléctricas para que estas no presenten cortos circuitos y la máquina deje de funcionar.
- ❖ La máquina cuenta con un lugar especialmente diseñado para situar las copas y que se realice el proceso de dosificación adecuado, en caso de que esta no se encuentre en el lugar indicado existirán derrames de líquidos.
- ❖ En caso de que la máquina no se encuentre en funcionamiento se recomienda que el usuario no tenga en marcha la aplicación Android, para prevenir solicitudes indeseadas.

BIBLIOGRAFIA

- ✚ Baena Herrera, J. D. (2017). Prototipo funcional de un dispensador automático de bebidas alcohólicas accionado a través de bluetooth.
<https://repositorio.ucp.edu.co/bitstream/10785/4819/1/DDMIST25>
- ✚ Wayland Valverde, R. D., Aguirre Alvarado, E. A. (2020). Mezclador de bebidas semi-automatico para la preparación de cocteles (piña colada y margarita) con el controlador LOGO! en la discoteca 360°.
<https://repositorio.unan.edu.ni/15241/1/15241.pdf>
- ✚ Flor Rodriguez, M. E. (2019). Diseño de una maquina semiautomática para la mezcla de cocteles. <https://repositorio.uide.edu.ec/bitstream/37000/3871/1/T-UIDE-2228.pdf>
- ✚ El Microcontrolador PIC16F877A.
<https://latecnicalf.com.ar/descargas/material/electronicadigital/PIC16F877%20apunte.pdf>

ANEXOS

Componentes que se utilizara en el proyecto.



Figura: Microcontrolador PIC16F877A, Modulo Bomba de Agua, Puente H L298N,
Modulo Bluetooth HC-06

Batería, placa virgen, jumpers, y conector de batería que se utilizara para conectar al PIC16F877A y alimentar con la batería de 9v.



Figura: Placa Virgen, jumpers, batería 9v, conector de batería a Jack DC.

Conectamos el Modulo Bluetooth H-06 al PIC16F877A en los pines TX-PIN26 Y RX-PIN25



Figura: PIC16F877A conectado al Módulo Bluetooth.

Seguidamente conectamos los Módulos Bomba de Agua en los pines Modulo de Bomba de Agua 1: IN1-PIN, IN2-PIN, IN3-PIN, IN4-PIN, Modulo de Bomba de Agua 2: IN1-PIN, IN2-PIN, IN3-PIN, IN4-PIN



Figura: PIC16F877A conectado al Módulo Bomba de Agua y al Módulo Bluetooth.

Finalmente introducimos el PIC16F877A y todos los componentes en el prototipo de dispensador.



Figura: Componentes dentro del prototipo de dispensador.

Podemos observar que prototipo de dispensador está en perfecto funcionamiento.



Figura: Prototipo de dispensador finalizado visto de frente.

Prototipo de dispensador listo para que utilice un cliente.



Figura: Prototipo de dispensador de costado.