

UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE TECNOLOGÍA
CARRERA DE ELECTRÓNICA Y TELECOMUNICACIONES



“SISTEMA DE REGISTRO DE ASISTENCIA Y APOYO EN
OPERACIONES RUTINARIAS MEDIANTE PIC18F4550
PARA LA EMPRESA ALVAPLAS SRL.”

Trabajo de Aplicación – Examen de Grado presentado para obtener el Grado de
Licenciatura

POR: Univ. CARLOS SAMUEL PUJRO LIMACHI

La Paz – Bolivia

Junio, 2022

Dedicatoria:

A mi hermana Jhovana Soledad Pujro Limachi y a mi madre Asunta Limachi Chañorco, por haberme transmitido su sabiduría, guiándome día tras día, por siempre estar conmigo, brindando su cariño, ternura y comprensión. Sin ustedes nada de esto hubiera sido posible, ya que siempre me apoyaron, depositando toda su confianza en mí, a pesar de mis fallas y estando a mi lado en los buenos, malos y decisivos momentos de mi vida.

Agradecimiento:

Primeramente a Dios por guiarme en tiempos de angustia y brindarme fortaleza, a mi familia: Jhovana Pujro, Asunta Limachi, Bernarda Limachi y Samuel Pujro Vito, a mis amigos y amigas, quienes estuvieron desde mis inicios cerca mío, dando lugar a que hoy yo pueda ser una persona íntegra, tanto en el campo de mi formación intelectual como en la vida misma. Así también a los docentes y personal administrativo de la carrera de Electrónica y telecomunicaciones por brindarme conocimientos, habilidades y la oportunidad de formarme como profesional.

Resumen

El presente trabajo plantea desarrollar un sistema de registro (Control) de asistencia del personal, adecuado para la empresa ALVAPLAS SRL. dedicada a la fabricación de plantas de zapato, tomando en cuenta la situación de la empresa en cuanto su tamaño y características específicas que requiere en base al empleo del microcontrolador PIC18F4550. De esta manera, se describe las opciones más viables de dispositivos de registro de personal que pueda controlar el microcontrolador PIC acorde a las necesidades de la Empresa AVAPLAS SRL. Y así también aplicable a empresas de similar características y tamaño (PYMES). Dentro del análisis, se tratará la implementación de un sistema de bajo costo y apto para el número de dependientes de la empresa; estableciendo un dispositivo RTC como reloj, juntamente con memorias no volátiles para un registro de datos que garantiza un sistema estable y con una interfaz de teclado numérico y pantalla LCD para visualización. Asimismo se tratará el estudio de las características del Microcontrolador PIC18F4550 que permita adicionalmente controlar funciones rutinarias que puedan alivianar las tareas del personal operativo de la Empresa Alvaplas SRL. Tales como la verificación del material en la maquinaria inyectora de caucho y el control de temperatura del ambiente de postproducción.

ÍNDICE

1. PLANTEAMIENTO DEL PROBLEMA	1
1.1 Formulación del Problema	2
2. JUSTIFICACIÓN DEL TRABAJO	2
3. OBJETIVOS.....	3
3.1 Objetivo General.....	3
3.2 Objetivos Específicos	4
4. FUNDAMENTO TEÓRICO	4
4.1 Sistemas de control de personal.....	4
4.1.1 Concepto.....	4
4.1.2 Funciones.....	5
4.1.3 Clasificación.....	6
4.1.4 Procedimiento de Control de personal.....	11
4.2 Microcontroladores	13
4.2.1 Concepto.....	13
4.2.2 Arquitectura Interna.....	15
4.2.3 Microcontroladores PIC.....	20
4.2.4 Familia de microcontroladores PIC	21
4.3 PIC 18f4550.....	22
4.3.1 Características.....	22
4.3.2 Arquitectura Interna.....	23
4.3.3 Organización de memoria.....	24
4.3.4 Registros de Función Especial (SFR).....	29
4.3.5 Conjunto de Instrucciones	31
4.3.6 Puertos de Entrada/Salida.....	34
4.4 Dispositivos externos	35
4.4.1 DTH22	35
4.4.2 DS3231	37
4.4.3 Tarjetas SD	39
4.4.4 Pantalla LCD.....	42
4.4.5 Teclado Matricial.....	45
4.4.6 HC-SR04.....	46

5. DESARROLLO DEL TRABAJO	48
5.1 Desarrollo práctico	48
5.1.1 Diagrama de flujo	48
5.1.2 Diagrama de bloques	50
5.2 Desarrollo Experimental	51
5.2.1 Esquemático.....	51
5.2.2 Código Fuente.....	54
5.3 Análisis de Costos y Duración	66
5.3.1 Costo de Materiales	66
5.3.2 Costo de Mano de Obra	67
5.3.3 Tiempo Empleado.....	67
6. CONCLUSIONES	69
7. BIBLIOGRAFÍA	70
8. ANEXOS.....	72

1. PLANTEAMIENTO DEL PROBLEMA

El deficiente método de registro de asistencia en ingreso y salida para el personal dependiente de la empresa Alvaplas SRL, mediante registro manual (Lista y bolígrafos), implica un retardo al inicio de las actividades laborales, la incomodidad del personal una vez terminada la jornada laboral, la falla en la elaboración de planillas de asistencia debido a errores de escritura manual y posibles evasiones de control de asistencia en el registro de asistencia. Así también se tiene contratiempos en operaciones de rutina al verificar el nivel de material para la maquinaria de la empresa y el control de temperatura de la sala de post producción.

Para la producción de plantas de zapatos, las operaciones complementarias de la empresa Alvaplas SRL. son la verificación del nivel de material en la maquinaria y el control de temperatura del ambiente de la sala de post producción. Estas funciones complementarias son realizadas por los operadores y ocasionan disminución de producción cada cierto intervalo de tiempo. Así también el indebido llenado de material, oportunamente, en las maquinarias pudiera ocasionar daños a las mismas; y el no estabilizar la temperatura correcta puede producir imperfecciones en el producto recién terminado (plantas de zapato).

Puesto que la Empresa Alvaplas SRL. es una Pyme cuenta con un número reducido de trabajadores y recursos limitados por lo que maneja un sistema manual de registro de asistencia que es vulnerable de fallas de control al personal por letra ilegible, correcciones y falta de confiabilidad de las horas de ingreso.

La conservación de un método deficiente en el registro de asistencia del personal de la empresa genera demora en el traspaso de datos a las tablas de control y fallas en la elaboración de planillas de pagos cada mes. De igual modo, las operaciones complementarias de rutina ocasionan un trabajo extra en los operadores lo que significa una producción no óptima.

A nivel empresa, en temporadas de alta demanda, la productividad requiere de personal adicional que realice las funciones complementarias y asistencia en nivel administrativo para reducir esta falta de productividad.

1.1 Formulación del Problema

¿Qué características Técnico-Económico tendrá el diseño de un sistema para mejorar el deficiente método de registro de asistencia en el personal y optimizar las funciones complementarias de operación en la Empresa Alvaplas SRL?

2. JUSTIFICACIÓN DEL TRABAJO

La solución que se ofrece al deficiente método de registro de asistencia del personal en la Empresa Alvaplas SRL., es un sistema dinámico tecnológico que permitirá mejorar la calidad del ambiente laboral en los trabajadores y mejorar el rendimiento de las funciones administrativas en cuanto el control de personal y elaboración de planillas de pago; debido a que llegaría a ser la forma más eficaz para conocer el cumplimiento horario de los trabajadores. Así también conllevará un beneficio al poder automatizar actividades simples rutinarias que significan un retardo en las operaciones de la empresa.

El Sistema de registro de personal, mediante digitación de código de usuario, procurará mejorar la calidad y confianza en el registro y control de asistencia del personal de la empresa Alvaplas SRL.; aplicando métodos tecnológicos con la ayuda un microcontrolador que pueda realizar el registro de ingreso y salida del personal exportando los datos a una memoria externa donde se pueda manipular los datos con mayor facilidad y sencillez. Así también el uso de sensores que permitan medir el nivel de material y temperatura mejorará el rendimiento de las operaciones, ayudarán a los operadores en el desempeño de sus funciones.

En la actualidad, la mayoría de las empresas cuentan con un sistema de registro y control de personal automatizado que mejora su rendimiento de control de asistencia. En el presente caso, se puede aplicar el mismo sistema simplificado de acuerdo a las necesidades de un Pyme, incorporando funciones adicionales específicas a la industria que se dedica.

En este aspecto, para una empresa de elaboración de planta de zapato, el control del nivel de material (caucho) en sus máquinas inyectoras es primordial para un producto bien acabado, del mismo modo la regulación de la temperatura adecuada para el producto recién acabado.

Siendo Alvaplas SRL. una Pyme que está en proceso de crecimiento con un total de diez trabajadores fijos y dos personales eventuales, requiere un sistema automatizado elemental para su registro y control de asistencia de personal que pueda asistir y aliviar las funciones del personal administrativo.

Este sistema tecnológico beneficiara tanto a los trabajadores de la empresa brindándoles seguridad y confianza en el registro de su asistencia, además de aliviar la carga de funciones rutinarias de operación, como también al propietario en el desarrollo de su empresa y maximización de su capacidad de productividad y reducción de uso de material fungible (hojas y bolígrafos).

La realización del trabajo de aplicación es viable debido a la necesidad existente de mejorar el sistema de registro y control de asistencia del personal. El desarrollo del trabajo de aplicación no requiere de un complejo procedimiento de aplicación, por lo que su realización solo requerirá de conocimientos de programación en microcontroladores e instalación de cableado para el funcionamiento del sistema.

Para desarrollar el sistema, los materiales están al alcance de pedido en sitios de electrónicas confiables y certificadas. En consideración a los beneficios que ofrecerán su costo es accesible y razonable.

3. OBJETIVOS

3.1 Objetivo General

Desarrollar un sistema de Registro (digital) de asistencia que permita mejorar el deficiente método de control y gestión de asistencia del personal en la Empresa Alvaplas SRL.

3.2 Objetivos Específicos

- Identificar las características técnicas que tendrá el sistema de registro de asistencia que permita mejorar el control y gestión de asistencia.
- Verificar la posibilidad de incorporar funciones adicionales al sistema de registro de asistencia, que ayuden con las operaciones rutinarias de la empresa Alvaplas SRL.
- Diseñar los circuitos que cumplan las características necesarias para el sistema de registro de asistencia y funciones adicionales con base en un microcontrolador de la manera más viable posible
- Estimar el costo económico que tendrá el sistema de registro de asistencia para la Empresa Alvaplas SRL.
- Realizar la comprobación del correcto funcionamiento del sistema de registro de asistencia para su posterior implementación.

4. FUNDAMENTO TEÓRICO

4.1 Sistemas de control de personal

4.1.1 Concepto

Según Kimaldi (2022), el control de personal dentro de una institución permite gestionar el saldo horario de cada trabajador, permisos laborales, horas extras, asistencia, control de turnos, activación de sirenas y alarmas horarias así como para gestionar las diferentes incidencias de acceso. Por lo que se constituye un sistema de control de personal un elemento indispensable en una organización con un número considerable de trabajadores. A pesar de ello los sistemas de control se pueden implementar en organizaciones de distintos tamaños acorde a las necesidades que se requiera.

Entonces, un Sistema de control de personal es un conjunto de dispositivos que permiten integrar una serie de procedimientos con la finalidad de registrar y controlar al personal que trabaja en una empresa o institución. Para EDEBE (2022), la finalidad de un sistema de control de personal se puede concretar en dos objetivos:

- Evaluación del desempeño, que se refiere a detectar los problemas de funcionamiento del personal y determinar las causas que los producen para tomar decisiones que mejoren la situación.
- Cumplimiento de la disciplina. Controlar el cumplimiento de las normas por parte de los trabajadores respecto a las entradas y salidas, el cumplimiento del horario de trabajo, la realización de horas extras, los permisos, las vacaciones, las licencias y los retrasos.

Así también cuando se refiere a control de acceso de personal no necesariamente se hace sobre el horario. Si bien una empresa gusta de saber cuándo entran sus trabajadores y el tiempo que están en el puesto de trabajo, con el primero solo sabremos que la persona es la correcta. RRHHDigital (2021), lo que significa que también se utiliza como un control de acceso a distintas áreas o ambientes restringidos.

4.1.2 Funciones

En el aspecto de Controlar la asistencia, ingresos y salidas del puesto de trabajo, un sistema de control (registro) de personal automatizado dentro de una empresa Según CENAC (2009) permiten;

- Monitoreo de horas extras. Un sistema de control de asistencia laboral permite el registro de horas extras, mejorando la relación de los trabajadores con la empresa y facilitando las gestiones de los administrativos de rrhh.
- Gestión de nóminas. Las entradas de asistencia son cruciales para calcular con precisión los salarios de los empleados y, así, evitar inconvenientes de todo tipo. Los salarios se pagan por los días trabajados y esta información se obtiene a partir de los datos de asistencia. Además de esto, los datos de asistencia también se utilizan para calcular las asignaciones por turno y la bonificación por asistencia, entre otras cosas Delpueche S. (2022). Por lo que todos esos datos los puede proveer un sistema de control de asistencia.
- Ahorra tiempo y dinero. El control de asistencia automatizado cuenta con tecnología que registra datos en segundos y se obtiene un concentrado completo y

actualizado de los datos personales y laborales de cada empleado. Así los trabajadores (usuarios) pueden aprovechar mejor su tiempo. Además, solucionan los errores humanos en anteriores tipos de gestión de información manual que pueden tener un impacto negativo en gestión de gastos.

- Controla el absentismo laboral. Permite detectar inasistencias de trabajadores de manera rápida mediante un software que analice los registros diarios, semanales, mensuales dependiendo de la empresa.
- Centraliza la información. Los datos registrados en un software de asistencia permanecen por tiempo indefinido para el libre acceso por los encargados de Recursos Humanos. Por lo que se tendrá toda la documentación requerida en un mismo lugar y accesible y también permite el respaldo y recuperación de la información
- Permite la generación de reportes de pago que se entregan al departamento de Recursos Humanos e incluso para la información que requieren los órganos de contraloría para llevar a cabo sus funciones

Optimizar el control de asistencia laboral y su correcta implementación significa mejorar ostensiblemente varios aspectos que en definitiva repercuten en la mejora constante, sostenida y el crecimiento exponencial de una empresa mediante el refuerzo de su capital humano. (Delpueche S.; 2022).

4.1.3 Clasificación

Para la autenticación de cada usuario puede realizarse mediante tecnología biométrica, proximidad, tarjeta inteligente, banda magnética, código de barras o password. Para Kimaldi (2022) estas modalidades son las más utilizadas y eficientes. De modo que existe una variedad de dispositivos para el control de personal y para determinar un sistema adecuado se puede considerar los puntos expuestos en la Figura N°1:

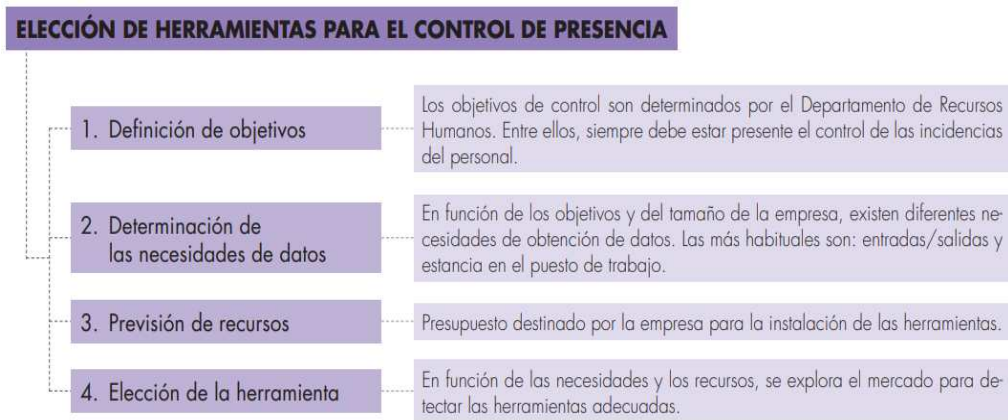


Figura N° 1. Elección de Herramientas para el control de presencia

Fuente: EDEBE(2022)

Siendo el presupuesto destinado para el sistema de control de personal un aspecto importante debido a que no solo es el equipo que se adquiere sino también la instalación del equipo y accesorios adicionales además de pago por software.

Cualquier proceso de identificación personal puede ser comprendido mediante un modelo simplificado. Ondasat (2021) postula la existencia de tres indicadores de identidad que definen el proceso de identificación:

- Conocimiento: la persona tiene conocimiento (por ejemplo: un código)
- Posesión: la persona posee un objeto (por ejemplo: una tarjeta)
- Característica: la persona tiene una característica que puede ser verificada (por ejemplo: una de sus huellas dactilares o rostro).

Entonces, los dispositivos para el control de personal se clasifican en cuanto sus características técnicas siendo:

Sistemas de control Físico

Según Wokrmeter (n.d.), son sistemas de control de empleados incompatibles con los nuevos entornos laborales (teletrabajo, horario flexible etc.). Son aquellos que no requieren una mayor intervención por el usuario debiendo realizar algún procedimiento

sencillo que depende de objetos físicos portables o códigos con identificación única del usuario como ser:

- Tarjetas con banda magnética: La tarjeta cuenta con una banda magnética que puede ser leída por un dispositivo electrónico y es en el Lector donde está registrada la información del usuario en cada caso. Se basa en el registro de datos en la pista magnética de una tarjeta que utiliza la propiedad de que ciertos materiales se magnetizan de manera duradera bajo la acción de un campo magnético.
- Tarjetas de proximidad: Son aquellas que utiliza la tecnología RFID (Radio Frequency IDentification, Identificación por radiofrecuencia). Es una tarjeta plástica que lleva incrustada en su núcleo un circuito integrado y una antena de comunicación. Cada usuario tiene una tarjeta que contiene un número de serie que es detectado por un lector de frecuencia de radio, típicamente de 125kHz. El propósito fundamental de esta tecnología es transmitir como identidad un número de serie único mediante ondas de radio. Su durabilidad es superior a las de banda ya que la lectura se produce sin contacto físico, por lo que el deterioro es mínimo.
- Código de barras: Se asigna un código único a cada usuario, Los códigos de barras codifican datos alfanuméricos como una serie de líneas blancas y negras de varios anchos Ruoti S.(2022) de acuerdo al tipo de codificación que se utilice, dicho código está grabado en algún objeto físico portado por el usuario. La información se recoge a través de un lector habilitado para ello en el lugar de acceso a la empresa o al puesto.
- Código QR: De manera similar al código de barras codifica información en pixeles blancos y negros de distintos tamaños, están diseñados para que los escanee una cámara, como las que se encuentran en un teléfono inteligente. Los códigos QR se utilizan con mayor frecuencia para almacenar enlaces web; sin embargo, pueden almacenar datos arbitrarios, como texto o imágenes. Ruoti S.(2022)
- Código por teclado: Aquellos sistemas en que se requiere de un código único para cada usuario que le identifica, como ser Cedula de identidad o alguna combinación

clave de fecha de nacimiento e iniciales de nombre. Dicho código es introducido en un dispositivo que tiene un registro almacenado de los códigos válidos para establecer asistencia de personal.

Sistemas biométricos

Para Wokrmeter (n.d.) son el resultado de la aplicación de técnicas matemáticas y estadísticas sobre los rasgos físicos o de conducta de una persona para verificar identidades o identificar a personas. Como ser las huellas dactilares, los patrones faciales, las retinas, el iris o la geometría de la palma de la mano. Su principal desventaja es su alto precio de implantación, debido al coste de la tecnología que utilizan y así también los requisitos que se necesitan para que cumpla su total funcionamiento.

- Identificación por huellas dactilares: Se registra a los usuarios por reconocimiento de la huella digital de alguno de los dedos de la mano. El sistema de identificación automatizada de huellas dactilares, tiene un índice de seguridad del 99.9%. Sin embargo suele tener fallas debido al desgaste de las huellas y el pronto desgaste del detector si no se le hace un uso cuidadoso.
- Identificación por biometría vascular: Consiste en leer el patrón de las venas de las manos del usuario, es la tecnología más fiable y segura dentro del campo de la biometría de la mano. Funciona mediante un escáner que captura una imagen del tramado de las venas de la palma de la mano a través del reflejo de ondas de frecuencia corta (muy similares a los infrarrojos).
- Identificación por biometría facial: El reconocimiento por biometría facial es una tecnología que detecta rasgos y características del rostro del usuario y las compara con una base de datos para establecer coincidencias y verificación, requiere de unas condiciones muy concretas, sobre todo de luz, por lo que su rendimiento puede verse afectado por circunstancias ajenas a las personas.
- Identificación por escáner de iris u ojos: Consiste en detectar rasgos minúsculos de los ojos con tecnología muy sofisticada. Este sistema es tecnológicamente muy

seguro pero actualmente es también una técnica bastante cara. Suele utilizarse para proteger recintos que requieren de un elevado índice de seguridad.

- Biometría de perfil de mano: Funciona por reconocimiento de la morfología de la mano
- Hojas de firmas: Es el método más tradicional de registrar manualmente las entradas y salidas, mediante firmas únicas del usuario. Los datos son reportados en hojas de cálculo.

Los sistemas online declarativos

Los sistemas online declarativos son otra opción para llevar un control de empleados tanto de su entrada y salida como de la supervisión de su horario.

Los empleados suelen acceder a un entorno web para marcar cuándo inician y acaban sus jornadas laborales. Son herramientas declarativas que pueden estar sujetas al error ya que sólo proporcionan datos de control de presencia laboral. Wokrmeter (n.d.)

En este aspecto los sistemas de control pueden ser remotos con el avance de la tecnología, siempre y cuando se cuente con los recursos mínimos. Cortés N.(2020) indica tres tipos de control de personal a distancia como son

- Marcación por Web: Plataformas que cuentan con la posibilidad de registrar la asistencia de cada trabajador desde su cuenta personal. De esta forma, permite registrar el inicio y finalización de jornada laboral y/o descansos. El registro, puede hacerlo desde cualquier computador, tablet o dispositivo móvil con conexión a Internet. Los administradores podrán realizar un exacto cálculo de horas trabajadas desde cualquier lugar, donde quiera que se hayan realizado.
- Aplicación Georeferenciada: La aplicación móvil dispone de sistema de geolocalización. Se activa al abrir la aplicación y realizar el fichaje. En caso de no disponer de conexión, la app guardará esos datos de registro.
- Llamado telefónico: Registra asistencia de trabajadores con una simple llamada telefónica. Este registro y gestión de asistencia podrá realizarse sólo desde

teléfonos autorizados de su empresa (fijos o móviles). Los softwares contratados podrán analizar el fraude con el reconocimiento de fraude.

4.1.4 Procedimiento de Control de personal

Para el control de personal dentro de una empresa se sigue una serie de etapas que iterativas que se pueden dividir en nivel directivo y personal. En la Tabla N°1 se puede observar el procedimiento periódico para un sistema de control por tarjetas.

Comienza desde el ingreso de un nuevo personal Con la entrega de las credenciales (tarjeta) con código de barras por parte del Directivo (Administrativo). Una vez registrado el personal acude al lugar asignado a registrar la entrada a sus labores con la Tarjeta para el Control de Asistencia de Personal (o algún otro tipo de medio y formato de registro) quedando asentada la hora en que ésta ocurre. Después se comienza con el Desarrollo de las actividades que le fueron asignadas al personal.

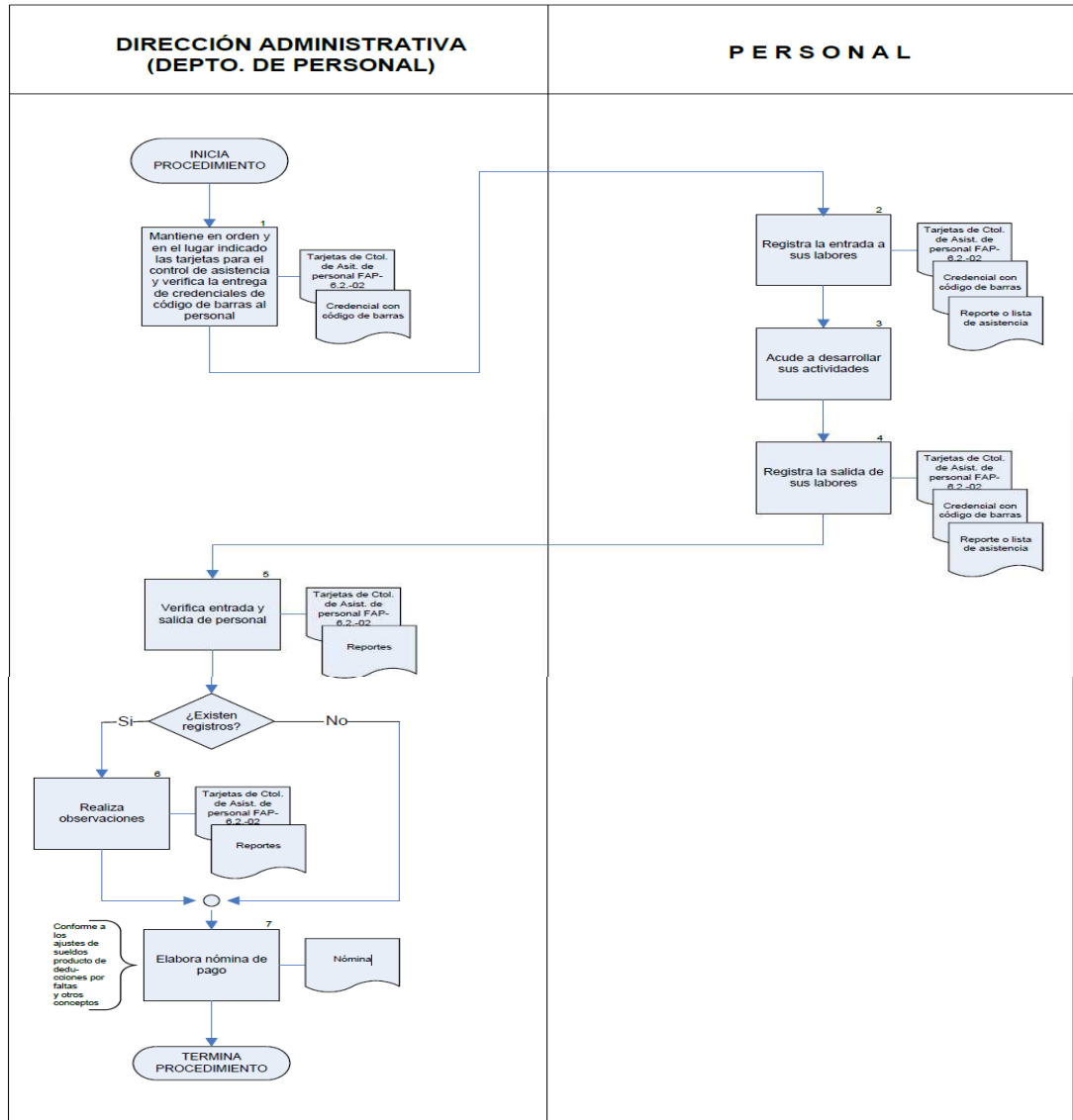
Una vez concluido cada horario, se procede a registrar la salida de sus labores con la Tarjeta para el Control de Asistencia de Personal (o algún otro tipo de medio y formato de registro) quedando grabada la hora de salida en que ésta ocurre.

El nivel directivo revisa cada una de las tarjetas y reportes de asistencia y verifica los horarios de entradas y salidas en el espacio correspondiente,

Si no hay registros de asistencia de los horarios de entradas y salidas, se realiza observaciones en los espacios vacíos. Si no es el caso se elabora la nómina de pago conforme a los ajustes de sueldos producto de deducciones por falta y otros conceptos.

Tabla N° 1

Procedimiento para Control de asistencia de personal con tarjetas credenciales



Fuente: Colegio de Bachilleres del Estado de Baja California (2014)

El procedimiento es similar a cualquier tipo de sistema de control de personal dependiendo de los requerimientos de la empresa, solamente varia el dispositivo de control de personal.

Para el registro de datos en el nivel administrativo se puede generar dependiendo del modelo de documento que se requiera como se puede señalar en la Tabla N°2.

Tabla N°2
Modelos de recogida de datos

DOCUMENTO	DESCRIPCIÓN
Ficha resumen del expediente personal	Forma parte del expediente personal del trabajador e incluye un resumen de los datos más importantes, incluido el índice de absentismo.
Diario de asistencia	Listado en el que consta el nombre de todos y cada uno de los trabajadores junto a sus horas de entrada y salida. Este documento lo crea el reloj del control horario.
Diario de incidencias	Listado en el que consta el nombre de los trabajadores que han presentado incidencias ese día (ausencias, retrasos, salidas imprevistas...). Se elabora a partir de los datos obtenidos del diario de asistencia.
Resumen mensual de incidencias	Resumen del listado de incidencias en el que figura el nombre de los trabajadores que han tenido incidencias con los datos de cada una de ellas.
Comunicación de incidencias	Si la incidencia es prevista (visita médica, obligación legal, licencias y permisos...), el trabajador debe presentar al departamento esta comunicación con antelación. En ella constará el tiempo de ausencia y el motivo de la incidencia. Estas comunicaciones se archivan en el expediente del trabajador.

Fuente: EDEBE(2022)

4.2 Microcontroladores

4.2.1 Concepto

Un microcontrolador es un circuito integrado fabricado de Silicio que puede programarse para realizar tareas (ejecutar instrucciones) de forma automática al conectarlo a una fuente de energía. El microcontrolador contiene todos los componentes de un computador. Se emplea para controlar el funcionamiento de una tarea determinada, así también llamado controlador incrustado. Una vez programado y configurado el microcontrolador solamente sirve para gobernar la tarea asignada Angulo J.M. & Angulo I, (1999).

Existe una diferencia entre Micropocesor y Microcontrolador, siendo que el microprocesador es un Sistema abierto con el que puede construirse un computador con las características que se desee, acoplándole los módulos que necesite comenzando con una memoria como se puede ver en la Figura N°2.

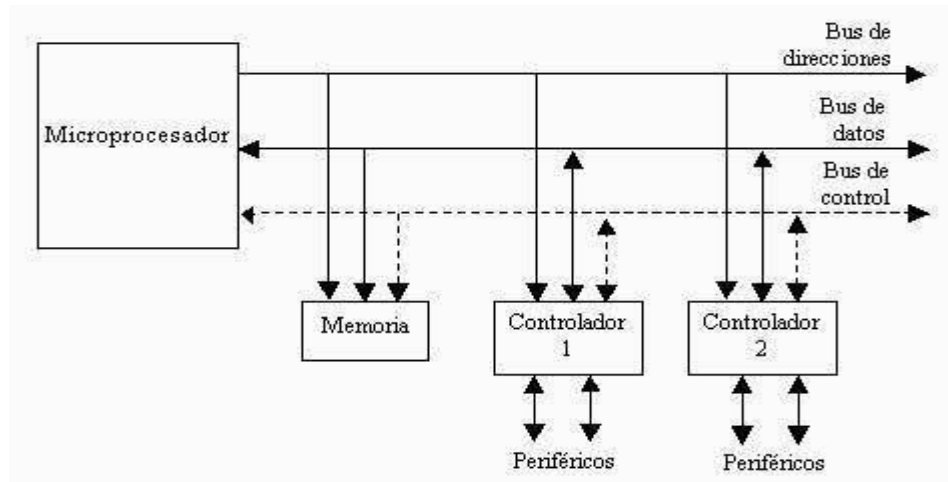


Figura N°2 Estructura de un sistema abierto basado en un microprocesador
Fuente Angulo J.M. & Angulo I, (1999).

Por otro lado el Microcontrolador es un Sistema cerrado que contiene un computador completo y de prestaciones limitadas que no se puede modificar, al microcontrolador se le diseña de tal manera que tenga todas las componentes integradas en el mismo chip. Además, que tiene ya sus terminales predeterminadas para una función específica (periféricos) como se puede observar en la Figura N°3.

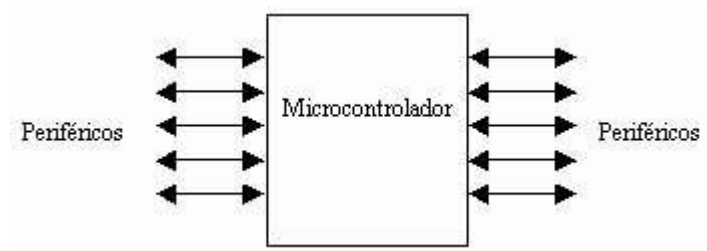


Figura N°3 Microcontrolador es un sistema cerrado
Fuente Angulo J.M. & Angulo I, (1999).

Existen así también una variedad de microcontroladores que varían sus especificaciones y rendimientos de acuerdo a familias y marcas como ser ATMEL, MOTOROLA, INTEL y MICROCHIP.

4.2.2 Arquitectura Interna

Procesador

Según Verle M.(n.d.) el procesador es una unidad que controla todos los procesos dentro del microcontrolador. Así también está compuesto por varias unidades más pequeñas como se puede ver en la Figura N°4, de las que las más importantes son:

- Decodificador de instrucciones, es la parte que descodifica las instrucciones del programa y acciona otros circuitos basándose en esto.
- Unidad lógica aritmética (Arithmetical Logical Unit - ALU) realiza todas las operaciones matemáticas y lógicas sobre datos
- Acumulador o registro de trabajo. Es un registro SFR estrechamente relacionado con el funcionamiento de la ALU. Es utilizado para almacenar todos los datos sobre los que se debe realizar alguna operación (sumar, mover). También almacena los resultados preparados para el procesamiento futuro. Uno de los registros SFR, denominado Registro Status (PSW), está estrechamente relacionado con el acumulador. El acumulador es denominado registro de trabajo (working register), o sea, registro W o solamente W.



Figura N°4 UNIDAD CENTRAL DE PROCESAMIENTO

Fuente: Verle M.(n.d.)

Así también se establece la arquitectura Harvard para los microcontroladores que es independiente la memoria de instrucciones y la memoria de datos y cada una dispone de su propio sistema de buses para el acceso como se puede observar en la Figura N°5. Para

los microcontroladores modernos se utiliza la arquitectura RISC (computadores de juego de instrucciones Reducido) que se caracteriza por tener un juego de instrucciones maquina simple que se ejecutan por lo general en un ciclo de instrucción y se puede generar instrucciones más complejas en base a un conjunto de instrucciones simples.

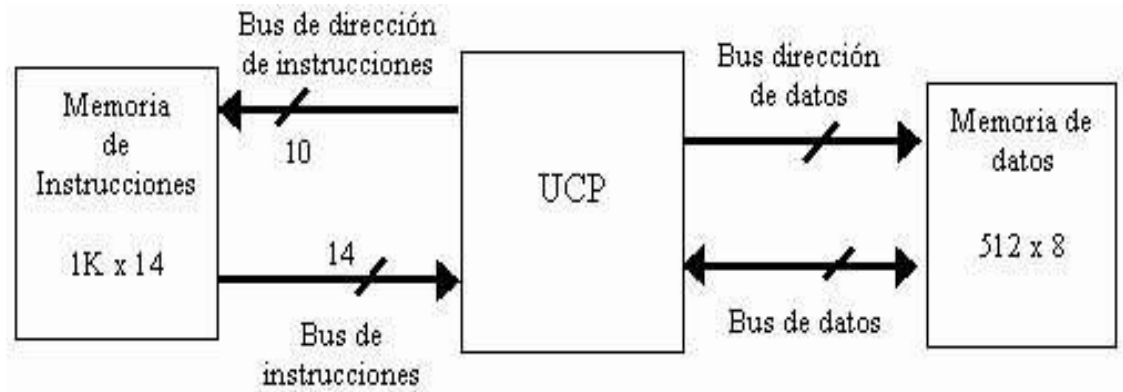


Figura N°5 Arquitectura Harvard para Microcontroladores
Fuente: Angulo J.M. & Angulo I, (1999).

El alto rendimiento y elevada velocidad que alcanza los procesadores modernos se debe al uso de Arquitectura Harvard, Arquitectura RISC y Segmentación

Memoria no Volátil

Esta memoria debe retener los datos almacenados después de apagar la alimentación (fuente de energía). Para el microcontrolador es denominada memoria de programa, en la que almacena todas las instrucciones del programa de control. Siendo así se considera las memorias ROM (acrónimo en inglés de read-only memory) ideales para esta tarea. Para Angulo J.M. & Angulo I, (1999) las memorias no volátiles se clasifican en:

- ROM con mascara, se caracteriza por su grabado de programa en el chip durante el proceso de su fabricación mediante el uso de máscaras. Siendo estas memorias más adecuadas para series grandes debido a los altos costos de diseño e instrumental.

- EPROM La grabación de esta memoria se realiza mediante un dispositivo físico gobernado desde un computador personal, que recibe el nombre de grabador. En la superficie de la cápsula del microcontroladores existe una ventana de cristal para la que se puede someter al chip de la memoria a rayos ultravioletas para producir su borrado y emplearla nuevamente.
- OTP (Programable una sola vez) Este modelo de memoria solo se puede grabar una vez por parte del usuario, utilizando el mismo procedimiento que con la memoria EPROM. Posteriormente no se puede borrar.
- EEPROM La grabación es similar a las memorias OTP y EPROM, pero el borrado es mucho más sencillo al poderse efectuar la misma forma que el grabado, o sea, eléctricamente. Sobre el mismo zócalo del grabador puede ser programada y borrada tantas veces como se quiera. Aunque se garantizan 1,000,000 de ciclos de escritura / borrado en una EEPROM, todavía esta tecnología de fabricación tiene obstáculos para alcanzar capacidades importantes y el tiempo de escritura de las mismas es relativamente grande y con elevado consumo de energía.
- Memoria FLASH, Es una memoria no volátil, de bajo consumo, que se puede escribir y borrar en el circuito al igual que las EEPROM, pero disponen de mayor capacidad que las EEPROM. Se caracteriza por el borrado con bloques completos y no se puede realizar sobre posiciones concretas.

Memoria de Lectura y Escritura

De acuerdo con Tocci R. J. & Widmer N. S. (2003), el nombre de estas memorias proviene del hecho de que puede grabarse o recuperarse información de ella sin necesidad de un orden secuencial, sino que puede accederse al RAM de la manera más rápida posible, con un tiempo de espera igual para cualquier posición de memoria.

Para el microcontrolador es denominada Memoria de datos, por lo que los datos varían constantemente debido al manejo de los programas que los utilizan, de allí que estas memorias de datos deben ser escritas y leídas constantemente de manera dinámica, por lo que las memorias ROM no cumplen con este requisito. En tal caso se hace uso de

memorias RAM (del acrónimo en inglés Random Access Memory). Verle M.(n.d.) establece que tiene la principal característica de almacenar temporalmente los datos y los resultados inmediatos creados y utilizados durante el funcionamiento del microcontrolador, pero cuenta con la desventaja de ser volátil (Al apagar la fuente de alimentación, se pierde el contenido de la memoria RAM).

Tocci R. J. & Widmer N. S. (2003) define dos tipos de memoria RAM:

- SRAM. Acrónimo de Static Random Access Memory (Memoria Estática de Acceso Aleatorio), es aquella que designa un tipo de memoria que se sustenta en semiconductores y capaz de mantener los datos sin necesidad de circuitos de refrescamiento, siempre y cuando se mantenga alimentada. De este tipo son las memorias NVRAM (Non-volatile Random Access Memory, o RAM no volátil) y MRAM (Magnetoresistive Random Access Memory, o RAM magnética).
- DRAM. Acrónimo de Dynamic Random Access Memory (Memoria Dinámica de Acceso Aleatorio), basa su tecnología en condensadores, que al perder carga progresivamente, requieren de un circuito de refresco que revisa su carga y la repone. Es el tipo más común de aplicación, pues permite crear módulos de enorme densidad de posiciones y alta velocidad de recuperación. De este tipo son las memorias DRAM Asíncrona y SDRAM (Synchronous Dynamic Random Access Memory, o DRAM síncrona).

Lineas de Entradas y Salidas

Son aquellas terminales de un microcontrolador que se conectan a un dispositivo externo (un periférico). Verle M.(n.d.) menciona que cada microcontrolador tiene uno o más registros (denominados puertos) conectados a los pines en el microcontrolador. Estos puertos se puede configurar como entradas o salidas mediante configuración por software (lo que significa que la función de algún pin puede ser cambiada durante el funcionamiento).

Una de las características más importantes de los pines de entrada/salida (E/S) es la corriente máxima que pueden entregar y recibir. En la mayoría de los microcontroladores la corriente obtenida de un pin es suficiente para activar un LED u otro dispositivo de baja corriente (10-20mA). Mientras más pines de E/S haya, más baja es la corriente máxima de un pin. Lo que significa que, todos los puertos de E/S comparten la corriente máxima declarada en la hoja de especificación técnica del microprocesador. Otra característica importante de los pines es que pueden disponer de los resistores pull-up. Estos resistores conectan los pines al polo positivo del voltaje de la fuente de alimentación y su efecto se puede ver al configurar el pin como una entrada conectada a un interruptor mecánico o a un botón de presión como se puede observar en la Figura N°6. Por lo general los microcontroladores tienen las resistencias pull-up configurables por software.

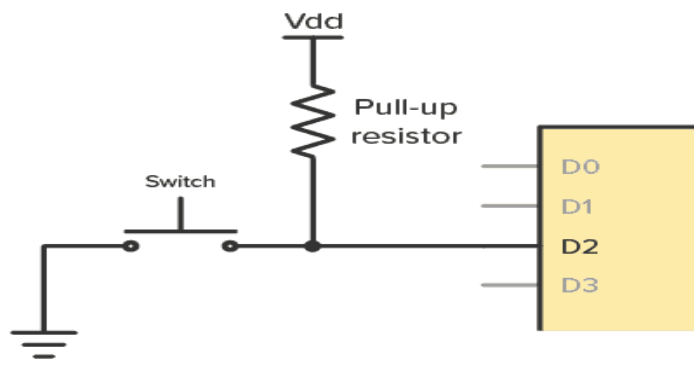


Figura N°6 Resistencia Pull- up
Fuente: Dahl N.(2021)

Todos los pines se pueden configurar a excepción de los pines destinados a recibir la alimentación, el cristal de cuarzo, que regula la frecuencia de trabajo, y una pin para provocar el Reset, las restantes pines de un microcontroladores sirven para soportar su comunicación con los periféricos externos que controla de acuerdo con Angulo J.M. & Angulo I, (1999). Las líneas de E / S que se adaptan con los periféricos manejan información en paralelo y se agrupan en conjuntos de ocho, que reciben el nombre de Puertas. Hay modelos con líneas que soportan la comunicación en serie; otros disponen

de conjuntos de líneas que implementan puertas de comunicación para diversos protocolos, como el I2C, el USB, etc.

Recursos auxiliares

En los microcontroladores se requiere distintas funciones especiales para un propósito específico por lo que se cuenta con complementos que refuerzan la potencia y flexibilidad del microcontrolador, según Angulo J.M. & Angulo I, (1999) son:

- Circuito de reloj, encargado de generar los impulsos que sincronizan el funcionamiento de todo el sistema.
- Temporizadores, orientados a controlar tiempos.
- Perro Guardián (watchdog), destinado a provocar una reinicialización cuando el programa queda bloqueado.
- Conversores AD y DA, para poder recibir y enviar señales analógicas.
- Comparadores analógicos, para verificar el valor de una señal analógica.
- Sistema de protección ante fallos de la alimentación.
- Estado de Reposo, en el que el sistema queda congelado y el consumo de energía se reduce al mínimo.

4.2.3 Microcontroladores PIC

El nombre exacto es PICmicro pero se le da el significado de Peripheral Interface Controller o Control de Interfaz Periférico. Microchip es la empresa fabricante de los pic, microcontroladores o circuitos integrados programables con una arquitectura computacional de tipo RISC.

Microchip ofrece una gran variedad de microcontroladores de 8, 16 y 32 bits, de bajo costo, con una arquitectura potente, varias tecnologías de memoria, herramientas de desarrollo fáciles de usar y documentación técnica completa. De acuerdo con Tecmikro (2022). Las principales ventajas del uso de microcontroladores PIC son:

- Fácil migración y actualización de diseños

- Amplio rango de velocidad de operación para satisfacer las múltiples necesidades de las aplicaciones
- Dispositivos con memoria de programa desde 384B a 512 KB para cumplir los requerimientos de código
- Arquitecturas compatibles para preservar la inversión económica en el desarrollo de códigos
- Compatibilidad Pin a Pin en múltiples encapsulados, facilitando así el remplazo
- Menor coste total del sistema con Periféricos Integrados
- Comunicaciones: SPI, I2C™, UART, CAN, USB, Ethernet, IrDA®, LIN
- Control: captura / comparación, contadores, reloj en tiempo real y PWM
- Módulos Analógicos: Convertidores A / D, comparadores, amplificadores operacionales, Convertidores D / A

4.2.4 Familia de microcontroladores PIC

Para poder clasificar los microcontroladores PIC se hace mención de tres familias según su capacidad, estas familias son las de: 8 bits, 16 bits y 32 bits. Estas a su vez tienen subdivisiones las cuales ya son por las gamas de los microcontroladores que se puede usar. Microcontroladores (n.d.)

- PIC DE 8 BITS, mejoran el rendimiento en su sistema reduciendo la carga de entrada y salida en su CPU. Los de 8 bits se pueden dividir según su gama como se puede ver en la Figura N°7:

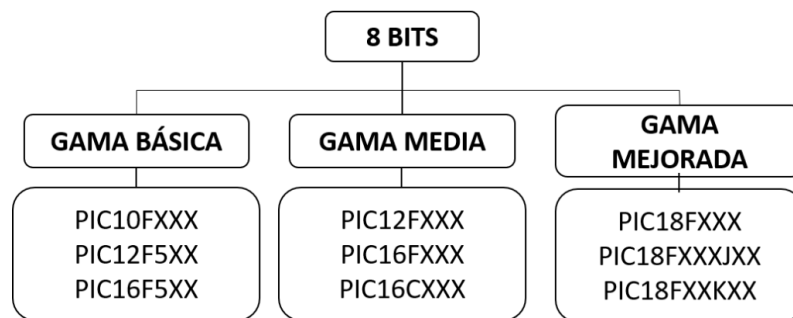


Figura N°7 PIC's de 8 bits
Fuente: Microcontroladores (n.d.)

- PIC DE 16 BITS, Cuentan con capacidad para almacenar 3072 palabras lógicas en su memoria. La familia de 16 bits también se la puede dividir, en dos grupos específicamente, los MCU range y los DSC range como se puede ver en la Figura N° 8.

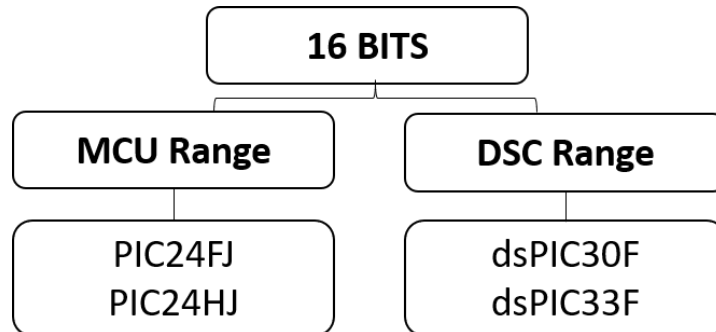


Figura N°8 PIC's de 16 bits
Fuente: Microcontroladores (n.d.)

4.3 PIC 18f4550

4.3.1 Características

De acuerdo a la hoja de datos especificada por Microchip (2009). Toda la familia PIC18F cuenta con las siguientes Características fundamentales:

- Arquitectura RISC avanzada Harvard: 16- bit con 8- bit de datos.
- 77 instrucciones
- Desde 18 a 80 pines
- Hasta 64K bytes de programa (hasta 2 Mbytes en ROMless)
- Multiplicador Hardware 8x8
- Hasta 3968 bytes de RAM y 1KBytes de EEPROM
- Frecuencia máxima de reloj 40Mhz. hasta 10 MIPS.
- Pila de 32 niveles.
- Múltiples fuentes de interrupción
- Periféricos de comunicación avanzados (CAN y USB)

Así también se considera el tipo de encapsulado en el que se presenta, donde Microchip (2009) cuenta con las presentaciones en 40-Pin PDIP, 44-Pin TQFP y 44-Pin QFN. Como se puede ver en la Figura N°9 el Encapsulado más común para prácticas.

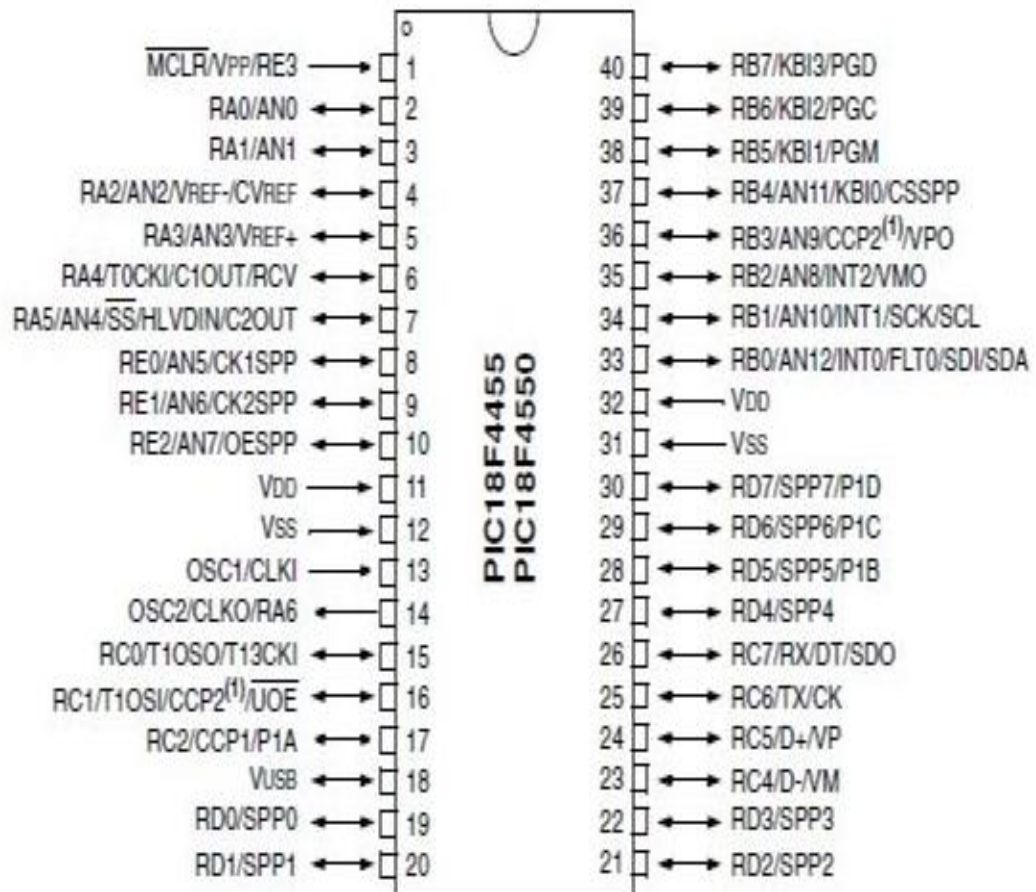


Figura N°9 Encapsulado para PIC18F4550
Fuente: Microchip (2009)

4.3.2 Arquitectura Interna

La arquitectura interna de un microcontrolador de la Familia PIC 18F se puede representar mediante un diagrama en bloques como se ve en la figura N°10.

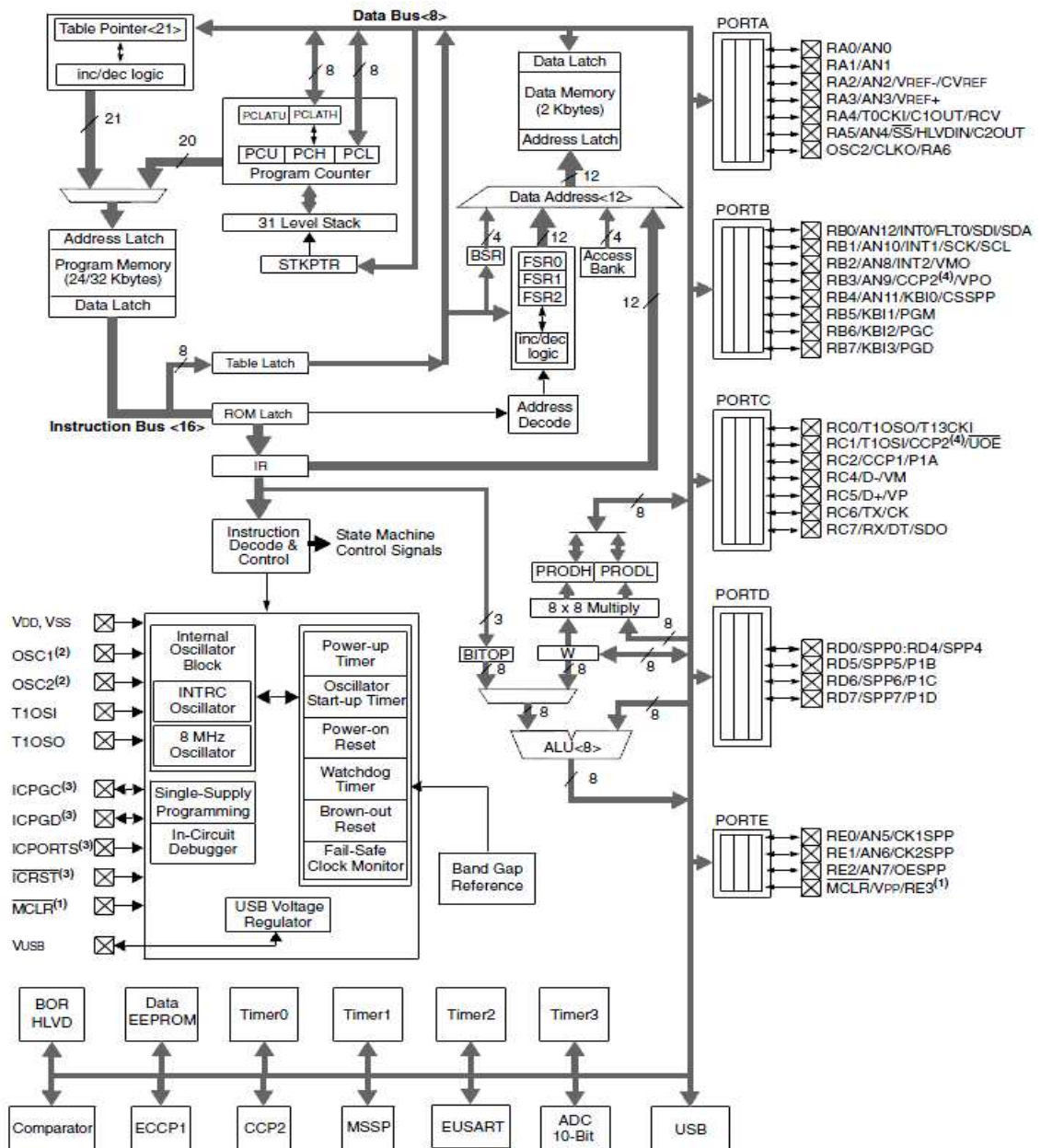


Figura N° 10 Diagrama en bloques de PIC18FXXXX
Fuente: Microchip (2009)

4.3.3 Organización de memoria

Guachalla R. (2020) resume la organización de memoria mediante el tipo de memoria y su función:

Memoria de Programa o Código (ROM Flash)

Para el Microcontrolador PIC18F4550 cuenta con 32 KBytes que ocupan las direcciones de 0000H hasta 7FFFH, como se puede ver en la Figura N°11

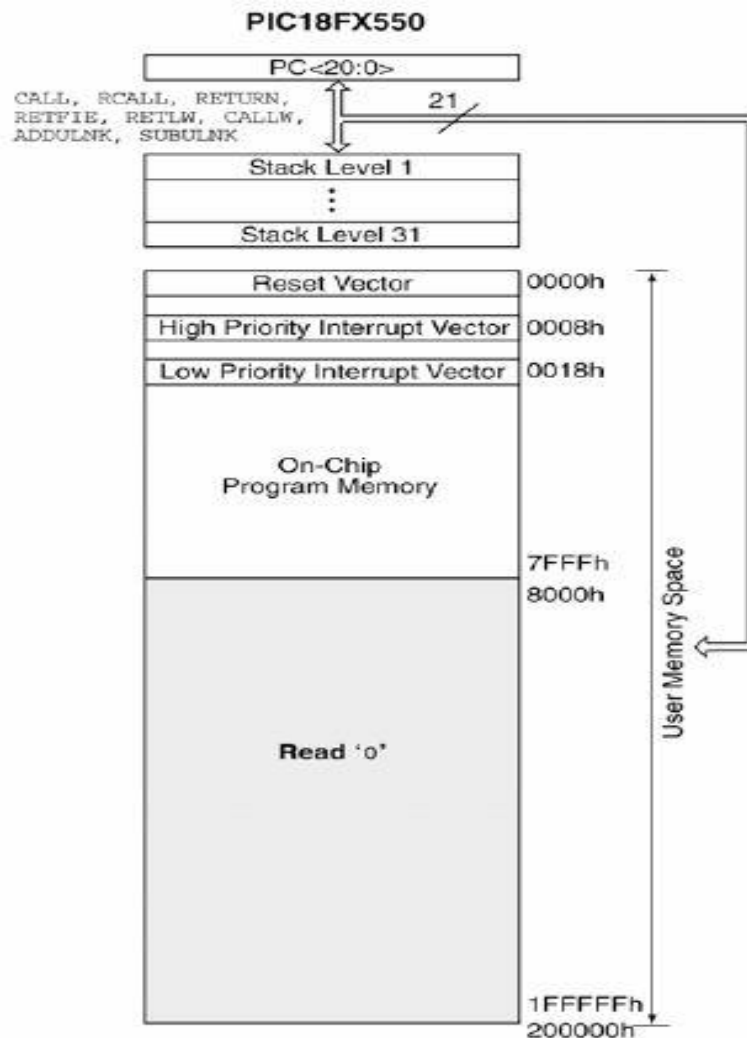


Figura N°11 Estructura interna de memoria de programa
Fuente: Microchip (2009)

Esta memoria realiza las siguientes funciones:

- Almacena instrucciones y constantes/datos

- Las instrucciones ocupan 2 bytes (excepto CALL, MOVFF, GOTO y LSFR que ocupan 4). Por lo tanto la memoria de programa puede almacenar hasta 16.384 instrucciones.
- La operación de lectura en posición de memoria por encima de 7FFFH da '0' como resultado (equivalente a la instrucción NOP)
- Direcciones especiales de la memoria de programa son:
 - Vectorización del Reset es 0000H
 - Vectorización de las interrupciones de alta prioridad es la 0008H
 - Vectorización de las interrupciones de baja prioridad es la 0018H

Memoria de Configuración (12 Bytes de ROM Flash para Bits de configuración 2 Bytes de ROM Flash para Registros de Identificación)

Se trata de un bloque de memoria situado a partir de la posición 30000H de memoria de programa (más allá de la zona de memoria de programa de usuario). Guachalla R. (2020)

Los Bits de configuración permiten configurar de algunas opciones del uC como:

- Opciones del oscilador
- Opciones de reset
- Opciones del watchdog
- Opciones de la circuiteria de depuración y programación
- Opciones de protección contra escritura de memoria de programa y memoria EEPROM de datos

Los Registros de identificación están situados en las direcciones 3FFFFEH y 3FFFFFH que contienen información del modelo y revisión del dispositivo. Son registros de solo lectura y no pueden ser modificados por el usuario.

Memoria EEPROM de datos (256 Bytes de EEPROM)

Memoria no volátil incorporada en el PIC18F4550 para almacenar datos que se deben conservar aun en ausencia de tensión de alimentación

- El acceso a esta memoria se realiza mediante los registros SFR: EECON1, EECON2, EEDATA, EEADR.
- Esta memoria permite hasta 1.000.000 de ciclos de borrado/escritura
- Se puede leer/escribir de forma individual en cada una de las 256 posiciones de memoria
- Cuando se realiza una operación de escritura la circuitería interna del microcontrolador se encarga de borrar previamente la posición en la que se desea escribir. La duración de un ciclo completo de borrado/escritura de un byte en la memoria EEPROM suele ser de unos 4ms.

Memoria de Datos (2048 Bytes de SRAM)

En este espacio de 2kByte están incluidos los SRF (Registros de Función Especial)

Almacena datos de forma temporal durante la ejecución del programa

La SRAM de 2048 bytes está dividida en 8 bancos de 256 bytes como se puede ver en la Figura N°12.

Además dispone de 160 bytes dedicados a los registros de función especial (SFR) situados en la parte alta del banco 15.

- Para acceder a un byte de la memoria RAM de datos primero debe seleccionarse el banco al que pertenece el byte mediante el registro de selección de banco (BSR) y a continuación direccionar el byte dentro del banco. Además existe una modalidad de acceso rápido a las 96 posiciones de la parte baja del banco 0 y a los 160 bytes de SFR
- Los bancos 4, 5, 6 y 7 se utilizan también para el USB

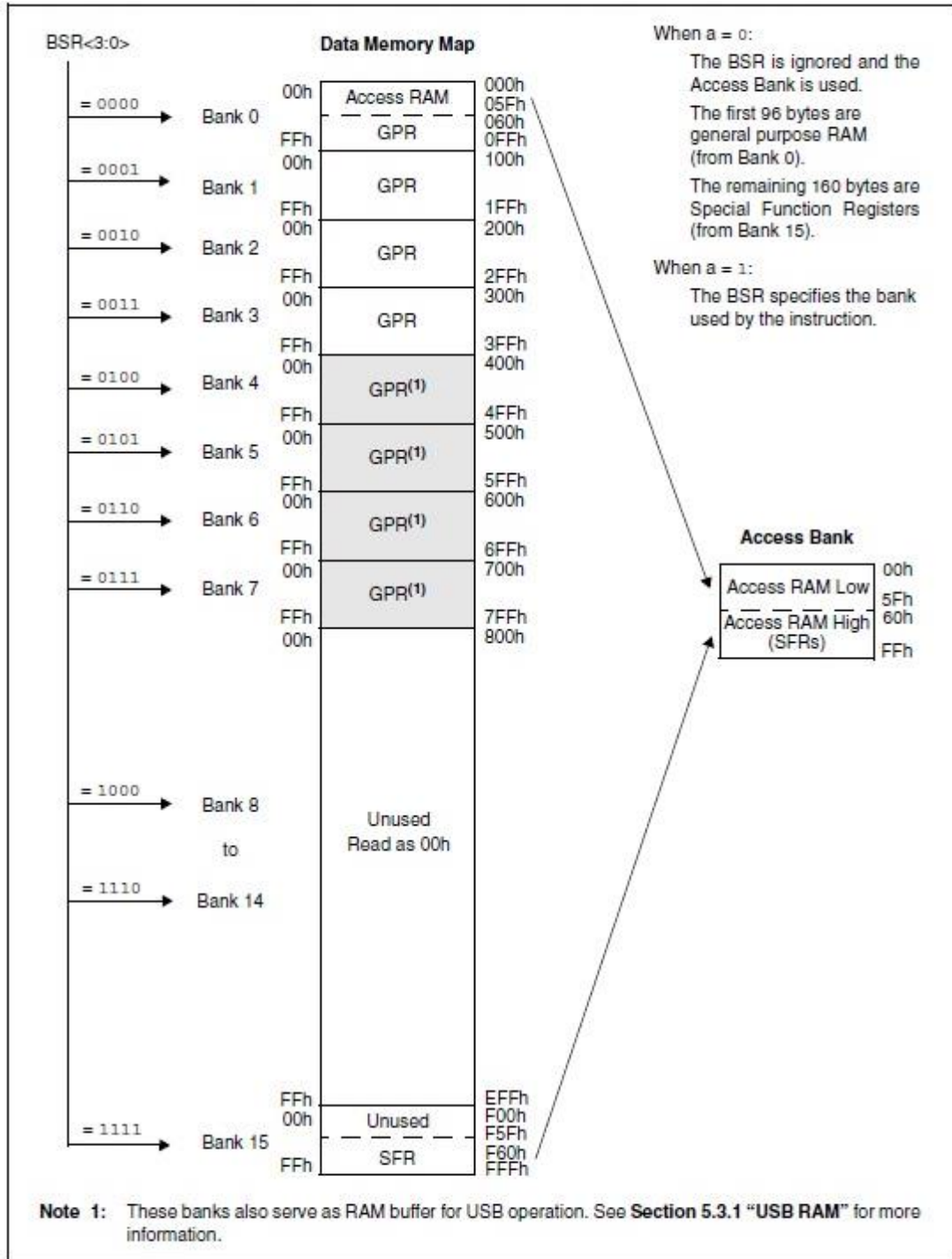


Figura N° 12 Estructura interna memoria de datos
Fuente: Microchip (2009)

Memoria de Pila (Stack) (31 Palabras de 21 Bytes de SRAM)

Es aquella que almacena la dirección de la instrucción que debe ser ejecutada después de una interrupción o subrutina

La Pila es un bloque de memoria SRAM independiente que sirve para almacenar temporalmente el valor del Contador de Programa (PC) cuando se produce una llamada a subrutina o una interrupción.

“El stack tiene tres funciones principales, que tienen que ver con la llamada a funciones y con la evaluación de expresiones”. Quiroga D. (2020).:

- El stack proporciona el lugar de almacenamiento para las variables locales declaradas dentro de las funciones (conocidas en C como 'variables automáticas')
- El stack almacena la información de estado y retorno cuando se llama a una función, denominado stack frame (marco de la pila de una función)
- El stack también se utiliza por la aplicación como zona temporal de almacenamiento, para evaluar algunas expresiones, almacenar resultados parciales, etc. También cuando almacenamos espacio con la función `alloca()` la memoria proviene del stack

El puntero de pila (registro SFR STKPTR) es un contador de 5 bits que indica la posición actual del final de pila.

4.3.4 Registros de Función Especial (SFR)

La memoria RAM se cuenta con una zona donde no se puede utilizar como memoria de datos la cual comprende de las direcciones que especifica la Tabla N° 3, la cual se compone de registros de propósito general (GPR) y de registros de función especial (SFR). Guachalla R. (2020) menciona que los SFR son los registros mediante los cuales se pueden monitorizar/controlar el funcionamiento de la CPU y de las unidades funcionales del microcontrolador.

Tabla N° 3

Mapa de registro de Funciones especiales

TABLE 5-1: SPECIAL FUNCTION REGISTER MAP

Address	Name	Address	Name	Address	Name	Address	Name	Address	Name
FFh	TOSU	FDh	INDF2 ⁽¹⁾	FBh	CCPR1H	F9h	IPR1	F7h	UEP15
FFEh	TOSH	FDEh	POSTINC2 ⁽¹⁾	FBEh	CCPR1L	F9Eh	PIR1	F7Eh	UEP14
FFDh	TOSL	FDDh	POSTDEC2 ⁽¹⁾	FBDh	CCP1CON	F9Dh	PIE1	F7Dh	UEP13
FFCh	STKPTR	FDCh	PREINC2 ⁽¹⁾	FBCh	CCPR2H	F9Ch	__ ⁽²⁾	F7Ch	UEP12
FFBh	PCLATU	FD8h	PLUSW2 ⁽¹⁾	FBBh	CCPR2L	F9Bh	OSCTUNE	F7Bh	UEP11
FFAh	PCLATH	FDAh	FSR2H	FBAh	CCP2CON	F9Ah	__ ⁽²⁾	F7Ah	UEP10
FF9h	PCL	FD9h	FSR2L	FB9h	__ ⁽²⁾	F99h	__ ⁽²⁾	F79h	UEP9
FF8h	TBLPTRU	FD8h	STATUS	FB8h	BAUDCON	F98h	__ ⁽²⁾	F78h	UEP8
FF7h	TBLPTRH	FD7h	TMR0H	FB7h	ECCP1DEL	F97h	__ ⁽²⁾	F77h	UEP7
FF6h	TBLPTRL	FD6h	TMR0L	FB6h	ECCP1AS	F96h	TRISE ⁽³⁾	F76h	UEP6
FF5h	TABLAT	FD5h	T0CON	FB5h	CVRCON	F95h	TRISD ⁽³⁾	F75h	UEP5
FF4h	PRODH	FD4h	__ ⁽²⁾	FB4h	CMCON	F94h	TRISC	F74h	UEP4
FF3h	PRODL	FD3h	OSCCON	FB3h	TMR3H	F93h	TRISB	F73h	UEP3
FF2h	INTCON	FD2h	HLVDCON	FB2h	TMR3L	F92h	TRISA	F72h	UEP2
FF1h	INTCON2	FD1h	WDTCON	FB1h	T3CON	F91h	__ ⁽²⁾	F71h	UEP1
FF0h	INTCON3	FD0h	RCON	FB0h	SPBRGH	F90h	__ ⁽²⁾	F70h	UEP0
FEFh	INDF0 ⁽¹⁾	FCFh	TMR1H	FAFh	SPBRG	F8Fh	__ ⁽²⁾	F6Fh	UCFG
FEeh	POSTINC0 ⁽¹⁾	FCEh	TMR1L	FAEh	RCREG	F8Eh	__ ⁽²⁾	F6Eh	UADDR
FEDh	POSTDEC0 ⁽¹⁾	FCDh	T1CON	FADh	TXREG	F8Dh	LATE ⁽³⁾	F6Dh	UCON
FECh	PREINC0 ⁽¹⁾	FCCh	TMR2	FACH	TXSTA	F8Ch	LATD ⁽³⁾	F6Ch	USTAT
FEbh	PLUSW0 ⁽¹⁾	FC8h	PR2	FABh	RCSTA	F8Bh	LATC	F6Bh	UEIE
FEAh	FSR0H	FCAh	T2CON	FAAh	__ ⁽²⁾	F8Ah	LATB	F6Ah	UEIR
FE9h	FSR0L	FC9h	SSPBUF	FA9h	EEADR	F89h	LATA	F69h	UIE
FE8h	WREG	FC8h	SSPADD	FA8h	EEDATA	F88h	__ ⁽²⁾	F68h	UIR
FE7h	INDF1 ⁽¹⁾	FC7h	SSPSTAT	FA7h	EECON2 ⁽¹⁾	F87h	__ ⁽²⁾	F67h	UFRMH
FE6h	POSTINC1 ⁽¹⁾	FC6h	SSPCON1	FA6h	EECON1	F86h	__ ⁽²⁾	F66h	UFRML
FE5h	POSTDEC1 ⁽¹⁾	FC5h	SSPCON2	FA5h	__ ⁽²⁾	F85h	__ ⁽²⁾	F65h	SPPCON ⁽³⁾
FE4h	PREINC1 ⁽¹⁾	FC4h	ADRESH	FA4h	__ ⁽²⁾	F84h	PORTE	F64h	SPPEPS ⁽³⁾
FE3h	PLUSW1 ⁽¹⁾	FC3h	ADRESL	FA3h	__ ⁽²⁾	F83h	PORTD ⁽³⁾	F63h	SPPCFG ⁽³⁾
FE2h	FSR1H	FC2h	ADCON0	FA2h	IPR2	F82h	PORTC	F62h	SPPDATA ⁽³⁾
FE1h	FSR1L	FC1h	ADCON1	FA1h	PIR2	F81h	PORTB	F61h	__ ⁽²⁾
FE0h	BSR	FC0h	ADCON2	FA0h	PIE2	F80h	PORTA	F60h	__ ⁽²⁾

- Note** 1: Not a physical register.
 2: Unimplemented registers are read as '0'.
 3: These registers are implemented only on 40/44-pin devices.

Fuente: Microchip (2009).

Se distinguen dos conjuntos de SFR:

- SFR asociados con el núcleo del microcontrolador
 - CPU: WREG, STATUS, BSR, etc...
 - Interrupciones: INTCON, PIE1, PIR1, IPR1, etc...
 - Reset: RCON
- SFR asociados con las unidades funcionales:
 - Timers: T0CON, TMR1H, TMR1L, T1CON, etc...
 - Convertidor A/D: ADRESH, ADRESL, ADCON0, ADCON1, etc...
 - EUSART: TXREG, TXSTA, RCSTA, etc...
 - CCP: CCPR1H, CCPR1L, CCP1CON, etc...
 - MSSP: SSPSTAT, SSPDATA, SSPCFG, etc...
 - Puertos de E/S: TRISA, PORTA, TRISB, PORTB, etc...

4.3.5 Conjunto de Instrucciones

El set de Instrucciones es proporcionado por Microchip (2009). Como se observa en la Tabla N°4 En la cual se detalla los ciclos máquina, tamaño de la instrucción y banderas afectadas por las instrucciones. El set de instrucciones para el PIC 18F4550 está dividido en cinco grupos: Operaciones orientadas a bytes, operaciones orientadas a bits, Operaciones de Control, Operaciones Literales y Operaciones de memoria de programa y datos.

Tabla N°4
Set de instrucciones PIC18F4550

Mnemonic, Operands	Description	Cycles	16-Bit Instruction Word				Status Affected	Notes	
			MSb	LSb					
BYTE-ORIENTED OPERATIONS									
ADDWF	f, d, a	Add WREG and f	1	0010	01da	####	####	C, DC, Z, OV, N	1, 2
ADDWFC	f, d, a	Add WREG and Carry bit to f	1	0010	00da	####	####	C, DC, Z, OV, N	1, 2
ANDWF	f, d, a	AND WREG with f	1	0001	01da	####	####	Z, N	1, 2
CLRF	f, a	Clear f	1	0110	101a	####	####	Z	2
COMF	f, d, a	Complement f	1	0001	11da	####	####	Z, N	1, 2
CPFSEQ	f, a	Compare f with WREG, Skip =	1 (2 or 3)	0110	001a	####	####	None	4
CPFSGT	f, a	Compare f with WREG, Skip >	1 (2 or 3)	0110	010a	####	####	None	4
CPFSLT	f, a	Compare f with WREG, Skip <	1 (2 or 3)	0110	000a	####	####	None	1, 2
DECf	f, d, a	Decrement f	1	0000	01da	####	####	C, DC, Z, OV, N	1, 2, 3, 4
DECFSZ	f, d, a	Decrement f, Skip if 0	1 (2 or 3)	0010	11da	####	####	None	1, 2, 3, 4
DCFSNZ	f, d, a	Decrement f, Skip if Not 0	1 (2 or 3)	0100	11da	####	####	None	1, 2
INCF	f, d, a	Increment f	1	0010	10da	####	####	C, DC, Z, OV, N	1, 2, 3, 4
INCFSSZ	f, d, a	Increment f, Skip if 0	1 (2 or 3)	0011	11da	####	####	None	4
INFSNZ	f, d, a	Increment f, Skip if Not 0	1 (2 or 3)	0100	10da	####	####	None	1, 2
IORWF	f, d, a	Inclusive OR WREG with f	1	0001	00da	####	####	Z, N	1, 2
MOVF	f, d, a	Move f	1	0101	00da	####	####	Z, N	1
MOVFF	f _s , f _d	Move f _s (source) to f _d (destination) 1st word	2	1100	####	####	####	None	
		2nd word		1111	####	####	####		
MOVWF	f, a	Move WREG to f	1	0110	111a	####	####	None	
MULWF	f, a	Multiply WREG with f	1	0000	001a	####	####	None	1, 2
NEGF	f, a	Negate f	1	0110	110a	####	####	C, DC, Z, OV, N	
RLCF	f, d, a	Rotate Left f through Carry	1	0011	01da	####	####	C, Z, N	1, 2
RLNCF	f, d, a	Rotate Left f (No Carry)	1	0100	01da	####	####	Z, N	
RRCF	f, d, a	Rotate Right f through Carry	1	0011	00da	####	####	C, Z, N	
RRNCF	f, d, a	Rotate Right f (No Carry)	1	0100	00da	####	####	Z, N	
SETF	f, a	Set f	1	0110	100a	####	####	None	1, 2
SUBFWB	f, d, a	Subtract f from WREG with Borrow	1	0101	01da	####	####	C, DC, Z, OV, N	
SUBWF	f, d, a	Subtract WREG from f	1	0101	11da	####	####	C, DC, Z, OV, N	1, 2
SUBWFB	f, d, a	Subtract WREG from f with Borrow	1	0101	10da	####	####	C, DC, Z, OV, N	
SWAPF	f, d, a	Swap Nibbles in f	1	0011	10da	####	####	None	4
TSTFSZ	f, a	Test f, Skip if 0	1 (2 or 3)	0110	011a	####	####	None	1, 2
XORWF	f, d, a	Exclusive OR WREG with f	1	0001	10da	####	####	Z, N	
BIT-ORIENTED OPERATIONS									
BCF	f, b, a	Bit Clear f	1	1001	bbbba	####	####	None	1, 2
BSF	f, b, a	Bit Set f	1	1000	bbbba	####	####	None	1, 2
BTFSC	f, b, a	Bit Test f, Skip if Clear	1 (2 or 3)	1011	bbbba	####	####	None	3, 4
BTFSS	f, b, a	Bit Test f, Skip if Set	1 (2 or 3)	1010	bbbba	####	####	None	3, 4
BTG	f, d, a	Bit Toggle f	1	0111	bbbba	####	####	None	1, 2
CONTROL OPERATIONS									
BC	n	Branch if Carry	1 (2)	1110	0010	nnnn	nnnn	None	
BN	n	Branch if Negative	1 (2)	1110	0110	nnnn	nnnn	None	
BNC	n	Branch if Not Carry	1 (2)	1110	0011	nnnn	nnnn	None	
BNN	n	Branch if Not Negative	1 (2)	1110	0111	nnnn	nnnn	None	
BNOV	n	Branch if Not Overflow	1 (2)	1110	0101	nnnn	nnnn	None	
BNZ	n	Branch if Not Zero	1 (2)	1110	0001	nnnn	nnnn	None	
BOV	n	Branch if Overflow	1 (2)	1110	0100	nnnn	nnnn	None	
BRA	n	Branch Unconditionally	2	1101	0nnn	nnnn	nnnn	None	
BZ	n	Branch if Zero	1 (2)	1110	0000	nnnn	nnnn	None	
CALL	n, s	Call Subroutine 1st word	2	1110	110s	kkkk	kkkk	None	
		2nd word		1111	kkkk	kkkk	kkkk		
CLRWDT	—	Clear Watchdog Timer	1	0000	0000	0000	0100	TO, PD	
DAW	—	Decimal Adjust WREG	1	0000	0000	0000	0111	C	
GOTO	n	Go to Address 1st word	2	1110	1111	kkkk	kkkk	None	
		2nd word		1111	kkkk	kkkk	kkkk		
NOP	—	No Operation	1	0000	0000	0000	0000	None	
NOP	—	No Operation	1	1111	xxxx	xxxx	xxxx	None	4
POP	—	Pop Top of Return Stack (TOS)	1	0000	0000	0000	0110	None	
PUSH	—	Push Top of Return Stack (TOS)	1	0000	0000	0000	0101	None	
RCALL	n	Relative Call	2	1101	1nnn	nnnn	nnnn	None	
RESET	—	Software Device Reset	1	0000	0000	1111	1111	All	
RETFIE	s	Return from Interrupt Enable	2	0000	0000	0001	000s	GIE/GIEH, PEIE/GIEL	
RETLW	k	Return with Literal in WREG	2	0000	1100	kkkk	kkkk	None	
RETURN	s	Return from Subroutine	2	0000	0000	0001	001s	None	
SLEEP	—	Go into Standby mode	1	0000	0000	0000	0011	TO, PD	

LITERAL OPERATIONS									
ADDLW	k	Add Literal and WREG	1	0000	1111	kkkk	kkkk	C, DC, Z, OV, N	
ANDLW	k	AND Literal with WREG	1	0000	1011	kkkk	kkkk	Z, N	
IORLW	k	Inclusive OR Literal with WREG	1	0000	1001	kkkk	kkkk	Z, N	
LFSR	f, k	Move Literal (12-bit) 2nd word to FSR(f) 1st word	2	1110	1110	00ff	kkkk	None	
MOVLB	k	Move Literal to BSR<3:0>	1	0000	0001	0000	kkkk	None	
MOVLW	k	Move Literal to WREG	1	0000	1110	kkkk	kkkk	None	
MULLW	k	Multiply Literal with WREG	1	0000	1101	kkkk	kkkk	None	
RETLW	k	Return with Literal in WREG	2	0000	1100	kkkk	kkkk	None	
SUBLW	k	Subtract WREG from Literal	1	0000	1000	kkkk	kkkk	C, DC, Z, OV, N	
XORLW	k	Exclusive OR Literal with WREG	1	0000	1010	kkkk	kkkk	Z, N	
DATA MEMORY ↔ PROGRAM MEMORY OPERATIONS									
TBLRD*		Table Read	2	0000	0000	0000	1000	None	
TBLRD*+		Table Read with Post-Increment		0000	0000	0000	1001	None	
TBLRD*-		Table Read with Post-Decrement		0000	0000	0000	1010	None	
TBLRD*+		Table Read with Pre-Increment		0000	0000	0000	1011	None	
TBLWT*		Table Write	2	0000	0000	0000	1100	None	
TBLWT*+		Table Write with Post-Increment		0000	0000	0000	1101	None	
TBLWT*-		Table Write with Post-Decrement		0000	0000	0000	1110	None	
TBLWT*+		Table Write with Pre-Increment		0000	0000	0000	1111	None	

Fuente: Microchip (2009).

Así también de la Tabla N° 4 para hacer uso de las instrucciones se toma en cuenta:

Notación para Números

- Decimal: D'100'ó.100
- Hexadecimal: H'64'ó0x64ó64
- Octal: O'144'
- Binario: B'01101100'
- ASCII: A'C'ó'C'

Notación para Registros y Literales

- w: Registro W, similar al acumulador, es el registro de trabajo.
- f: Campo de 5 bits (ffff), contiene la dirección del banco de registros, que ocupa el banco 0 del área de datos. Direcciona uno de esos registros.
- k: Representa una constante de 8bits.
- d: Bit del código OP de la instrucción. Selecciona el destino donde se guarda el resultado de una operación. Si d=0, el destino es W, y si d=1 el destino es f.
- b: Determina la posición de un bit dentro de un registro de 8bits, (ósea, tomará valores entre 0 y 7)

Símbolos

- []: Identifica opciones
- (): Identifica Contenido
- = o >: Se asigna a(especificar)
- < >: Campo de bits de un registro
- PC: Contador de programa que direcciona la memoria de instrucciones

Banderas (Flags)

Se encuentran dentro del registro STATUS

- Z: Flag de cero, cambia a 1 cuando una operación lógica o aritmética da 0 como resultado
- C: Flag de carry, cambia a 1 cuando una operación sobrepasa el nivel de representación del procesador (8 bits para el PIC18F4550)
- DC: Flag de carry del nibble inferior, de manera similar a la bandera C pero limitada a la representación de los 4 bits inferiores

4.3.6 Puertos de Entrada/Salida

En el PIC18F4550 hay hasta cinco puertos disponibles. “algunos pines de los puertos de E/S están multiplexados con un puerto alternativo desde las características periféricas del dispositivo” Microchip (2009). En general, cuando un periférico está habilitado, es posible que ese pin no se pueda utilizar como un pin de E/S de propósito general. Todas las líneas digitales de E/S disponen de al menos una función alternativa asociada a alguna circuitería específica del microcontrolador. Así también mencionar que la Selección de resistencia PULL-UP está habilitada solo para el Puerto B

Cada puerto de E/S normalmente está bajo el control de un registro SFR determinado, Guachalla (2020) sintetiza que cada bit de ese registro determina el estado del pin correspondiente en el microcontrolador de la siguiente manera:

- Registro TRIS: mediante este registro se configuran cada una de las líneas de E/S del puerto como:

- ENTRADA (bit correspondiente a '1')
- SALIDA (bit correspondiente a '0').
- Registro PORT: mediante este registro se puede LEER el nivel en el pin de E/S (Entrada de Datos)
- Registro LAT: mediante este registro se ESCRIBIR el valor de del pin de E/S (Salida de Datos)

4.4 Dispositivos externos

4.4.1 DTH22

El dispositivo DTH22 es un sensor digital de temperatura y humedad relativa es calibrado en fábrica para obtener unos coeficientes de calibración grabados en su memoria OTP. Aosong (n.d.) expone las Especificaciones Técnicas del sensor DTH22:

- Voltaje de Operación: 3V - 6V DC
- Rango de medición de temperatura: -40°C a 80 °C
- Precisión de medición de temperatura: $<\pm 0.5$ °C
- Resolución Temperatura: 0.1°C
- Rango de medición de humedad: De 0 a 100% RH
- Precisión de medición de humedad: 2% RH
- Resolución Humedad: 0.1%RH
- Tiempo de censado: 2s
- Interface digital: Single-bus (bidireccional)
- Modelo: AM2302
- Dimensiones: 20*15*8 mm
- Peso: 3 gr.
- Carcasa de plástico blanco

El dispositivo DTH22 consta de cuatro pines como se puede observar en la Figura N° 13, de los cuales solo un pin es utilizado para enviar datos de 8 bits mediante una señal digital hacia un microcontrolador.

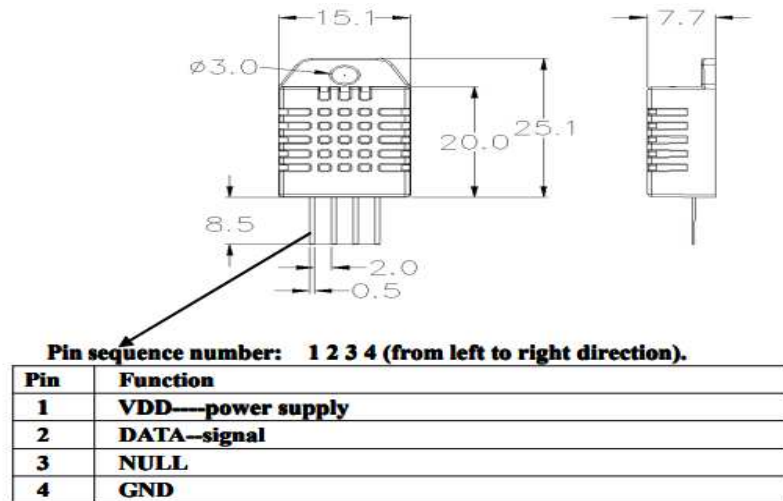


Figura N°13 Dimensión y especificación de pines DTH
Fuente: Aosong (n.d.)

Los datos se componen de parte integral y decimal, con un arreglo de:

DATO = datos RH integrales de 8 bits + datos RH decimales de 8 bits + datos T integrales de 8 bits + datos T decimales de 8 bits + suma de verificación de 8 bits

Donde “RH” se refiere a la humedad y “T” a temperatura, este proceso de obtener datos se realiza cada 2 segundos de acuerdo a la hoja técnica. Su procedimiento de acuerdo a Aosong (n.d.) consiste en

- 1) Microcontrolador envía señal de inicio a DTH22, este proceso debe superar al menos 18 ms para garantizar que DTH22 pueda detectar la señal del Microcontrolador, luego se espera de 20 a 40 us para la respuesta.
- 2) DTH22 envía señal de respuesta a Microcontrolador, es cuando DTH22 detecta la señal de inicio y envía una señal de bajo voltaje durante 80us como señal de respuesta, luego el programa de DTH22 transforma el nivel de voltaje del bus de datos de bajo a alto nivel, preparando los datos para enviar por 80us.

- 3) DRH22 envía datos al Microcontrolador, la transmisión de cada bit comienza con un nivel de bajo voltaje que dura 50us, la longitud de la siguiente señal de nivel de alto voltaje decide que el bit es "1" o "0".

4.4.2 DS3231

DS3231 es un dispositivo RTC de alta precisión que cuenta con un oscilador a cristal con compensación de temperatura (TCXO), sus Especificaciones Técnicas Según el fabricante Dallas (2005) son:

- Voltaje de alimentación de 3.0 a 5 volts.
- RTC de alta exactitud, maneja todas las funciones para el mantenimiento de fecha/hora.
- Exactitud de ± 2 ppm operando a una temperatura de 0°C a +40°C.
- Módulo cuenta con reloj DS3231 y memoria EEPROM I2C.
- El módulo cuenta con batería de respaldo
- Registro de segundos, minutos, horas, día de la semana, fecha, mes y año con compensación de años bisiestos hasta 2100.
- El DS3231 Incluye sensor de temperatura con exactitud de +/- 3 grados centígrados.
- 2 alarmas programables por hora/fecha.
- Salida de señal cuadrada programable.

RTC dispone de segundos, minutos, horas, días, fecha, mes, año y la información deseada. El reloj funciona en cualquiera de los formatos de 24 horas o banda / AM / PM (indicación del formato de 12 horas). La Dirección y los datos se transfieren en serie a través de un bus bidireccional I2C.

La distribución de sus terminales del DS3231 de acorde a Dallas (2005) se puede observar en la Figura N°13:

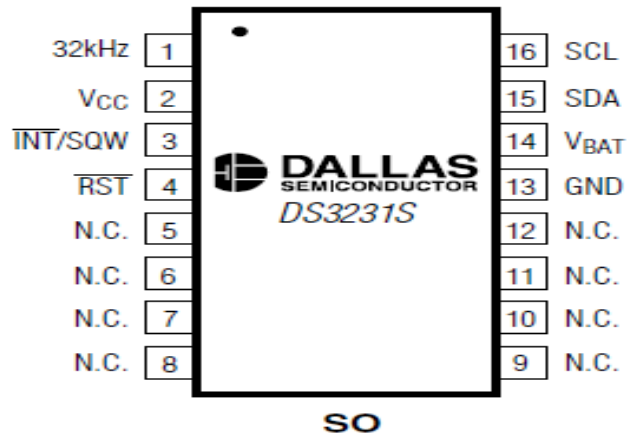


Figura N°13 Encapsulado DS3231

Fuente: Dallas (2005)

Donde aparte de VCC y GND se cuenta con:

- 32kHz. pin de drenaje abierto que requiere una resistencia pullup externa.
- VCC es el pin de alimentación de corriente continua.
- INT/SQW es la salida de onda cuadrada o interrupción baja activa. Este pin de drenaje abierto requiere una resistencia pullup externa cuando se hace uso de su función.
- RST es el pin de restablecimiento activo-bajo. Este pin es una entrada/salida de drenaje abierto.
- N.C. Sin Conexión, este pin debe estar conectado a tierra de acuerdo al fabricante.
- VBAT es la entrada de fuente de alimentación de respaldo.
- SDA Este pin es la entrada/salida de datos en serie para la interfaz serial I2C. Así también el fabricante menciona que este pin requiere una resistencia pull-up externa.
- SCL es la entrada de reloj serie, para la interfaz serial I2C y se usa para sincronizar datos.

Conforme a la hoja de datos de Dallas (2005): El DS3231 admite un bus I2C bidireccional y datos protocolo de transmisión donde el DS3231 opera como esclavo en el bus I2C. Conexiones al bus se hacen a través de la entrada SCL y drenaje abierto SDA.

4.4.3 Tarjetas SD

Son tarjetas de memoria flash de tamaño reducido de gran capacidad de almacenamiento y bajo consumo de energía, diseñadas por lo general para dispositivos portátiles. Estas memorias vienen en tres tamaños como se muestra en la Figura N°14, las cuales se especifican en:

- Tarjeta SD Estándar: Son las más grandes de esta denominación, con medidas de 32 x 24 milímetros, grosor de 2,1 milímetros. Generalmente utilizadas en cámaras fotográficas, grabadoras de audio, y muchos portátiles que son compatibles.
- Tarjeta miniSD: Tienen un tamaño de 21,5x20 milímetros, con un grosor de 1,4 milímetros.
- Tarjeta microSD: Tienen un tamaño de 15x11 milímetros, y un grosor de sólo 1 milímetro. Es la que normalmente utilizan los dispositivos pequeños como los móviles y las tabletas.

Asimismo las tarjetas SD pueden usar adaptadores para utilizarlos en sockets de diferente tamaño.

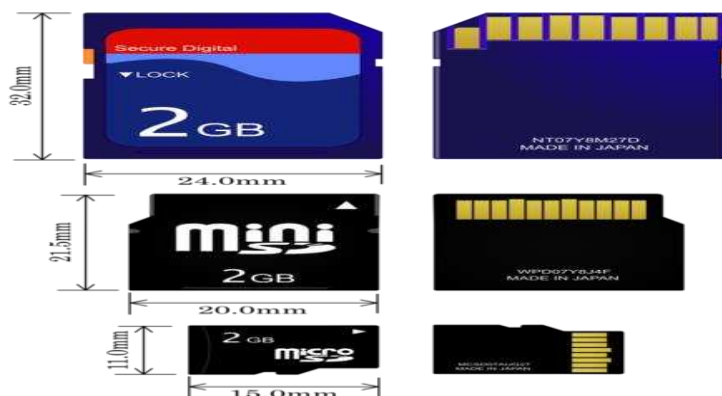


Figura N°14 Tipos de tarjetas SD
Fuente: Fernández Y. (2022).

Tipos de Tarjetas SD por capacidad de almacenamiento

Las tarjetas SD de igual manera se pueden clasificar por su capacidad lo que tampoco limita su uso siendo retro compatibles entre versiones anteriores:

- SDSC: Tarjetas Standard Capacity, con capacidad para almacenar hasta 2 GB de datos.
- SDHC: Tarjetas High Capacity, con capacidad para almacenar hasta 32 GB de datos.
- SDXC: Tarjetas Extended Capacity, con capacidad para almacenar hasta 2 TB GB de datos, que son unos 2.000 GB.
- SDUC, Tarjetas SD Ultra Capacity con capacidad desde los 2 terabytes hasta los 128 terabytes de capacidad de almacenamiento. Estas tarjetas van orientadas a dispositivos que procesan una gran cantidad de información y de datos.

Tipos de Tarjetas SD por su Velocidad (Clase)

Conforme a las velocidades que soporta la tarjeta SD (Velocidad de transferencia de datos) expresado en Bytes por segundo, Fernandez Y.(2022) identificar siete tipos de tarjetas según la velocidad, y se distinguen en diferentes clases:

- Clase 2: Velocidad de escritura mínima de 2 MB/s
- Clase 4: Velocidad de escritura mínima de 4 MB/s
- Clase 6: Velocidad de escritura mínima de 6 MB/s
- Clase 8: Velocidad de escritura mínima de 8 MB/s
- Clase 10: Velocidad de escritura mínima de 10 MB/s
- UHS Speed Class 1 (U1): Tienen una velocidad de escritura mínima de 10 MB/s, pero como tiene un bus mejor que la Clase 10 es mejor para grabar vídeos FullHD a 1080p que son más largos.
- UHS Speed Class 3 (U3): Tienen una velocidad de escritura mínima de 30 MB/s, ideal para grabar vídeos en resoluciones 4K.

Sistemas de Archivo

Un sistema de archivo es el encargado de organizar y gestionar los datos de una unidad de almacenamiento de distintas maneras. Fernandez Y. (2019) menciona que cada sistema de archivos tiene sus propias ventajas y limitaciones entre los cuales están:

- FAT32 es compatible con prácticamente todos los dispositivos y sistemas operativos, razón por la que la mayoría de unidades USB están formateadas en este sistema. Sin embargo está limitado a almacenar archivos de hasta 4 GB. Su lado positivo es que es perfectamente compatible con Windows, macOS y GNU/Linux, y funciona sin problemas en USB 2.0.
- exFAT, es una actualización al FAT32 introducida por Microsoft en Windows Vista con la intención de superar el límite de 4 GB. Es compatible con Windows, macOS o GNU/Linux, a partir de Windows XP SP3 u OS X 10.6.5 Snow leopard. Por lo que es un sistema de archivos muy recomendado para unidades externas como USB o tarjeta SD.
- NTFS, es el sistema de archivos que Windows utiliza por defecto promovida por Microsoft. Sin los límites del tamaño máximo de archivo del FAT32, el NTFS se convierte en una muy buena opción para discos duros y otras unidades externas. En cambio su mayor desventaja es que no es totalmente compatible con todos los sistemas operativos.
- HFS+ De manera similar a NTFS Apple diseñó el sistema HFS+ a su medida. Siendo compatible con los sistemas GNU/Linux y en Windows sólo se puede leer el contenido de los discos formateados con este sistema, pero no escribir en ellos.
- Ext2, ext3 y ext4 Son los sistemas utilizados por las distribuciones GNU/Linux. El principal inconveniente es que no es compatible con otros sistemas operativos.

Módulo lector de memoria SD Card para microcontroladores

Es un dispositivo diseñado para acceder a la memoria SD en modo SPI, Soporta tarjetas micro SD y micro SDHC. El módulo (adaptador de tarjeta SD) es un lector de tarjetas SD

para lectura y escritura a través del sistema de archivos y el controlador de interfaz SPI, SCM El sistema se puede completar dentro de una tarjeta SD de archivos. Sus principales características según su hoja de datos son:

- Admite tarjeta Micro SD, tarjeta Micro SDHC (tarjeta de alta velocidad)
- La placa de circuito de conversión de nivel que puede interconectar el nivel es de 5 V o 3,3 V
- La fuente de alimentación es de 4,5 V ~ 5,5 V, placa de circuito regulador de voltaje de 3,3 V
- La interfaz de comunicaciones es una interfaz SPI estándar
- 4 orificios de posicionamiento de tornillos M2 para una fácil instalación
- Interfaz de control: un total de seis pines (GND, VCC, MISO, MOSI, SCK, CS), GND a tierra, VCC es la fuente de alimentación, MISO, MOSI, SCK para bus SPI, CS es el pin de señal de selección de chip;
- Circuito regulador de 3,3 V: salida del regulador LDO de 3,3 V para chip de conversión de nivel, Micro suministro de tarjeta SD;
- Circuito de conversión de nivel: tarjeta Micro SD para señalar la dirección de conversión de 3,3 V,
- La interfaz de la tarjeta MicroSD para controlar la dirección de la señal MISO también está convertido a 3,3 V, los sistemas de microcontroladores AVR generales pueden leer la señal;
- Conector de tarjeta Micro SD: plataforma autobomba, fácil inserción de tarjeta.
- Orificios de posicionamiento: 4 orificios de posicionamiento de tornillos M2 con un diámetro de 2,2 mm, por lo que el módulo es fácil de instalar el posicionamiento, para lograr la combinación entre módulos.

4.4.4 Pantalla LCD

Las pantallas LCD se utilizan para visualizar los diferentes mensajes por medio de arreglo de caracteres. Vienen en las presentaciones más comunes de 16x2 (16 columnas y 2 filas) y 20x4 (20 Columnas y 4 filas) Sus características técnicas se pueden ver en la Tabla N°5.

Tabla N° 5.
Características eléctricas de la pantalla LCD

Artículo	Símbolo	Condición	Estándar			Unidad
			mínimo	tipo	máx.	
Voltaje de entrada	VDD	+5V	4.7	5.0	5.5	V
		+3,3V	2.7	3.0	5.3	V
Corriente de suministro	IDD	VDD=5V	-----	2.5	5	mamá
Voltaje de conducción LCD recomendado para el módulo de versión de temperatura normal	VDD-V0	-20°C	5.0	5.1	5.7	V
		0°C	4.7	5.0	5.5	
		25°C	4.3	4.5	4.7	
		50°C	4.1	4.3	4.5	
		70°C	3.7	3.9	4.3	
Voltaje directo LED	FV	25°C	-----	4.2	4.6	V
LED de corriente directa	SI	25°C	-----	240	-----	mamá
corriente de alimentación EL	IEL	VEL=110V CA 400 Hz	-----	-----	-----	mamá

Fuente: DZLCM (n.d.).

Así también cuenta con 14 pines de conexión y 16 cuando tiene iluminación integrada como se detalla en la Tabla N°6. Siendo la misma distribución para ambas pantallas 16x2 o 20x4

Tabla N°6
Función de pines pantalla LCD

FUNCIÓN	NRO. PIN	NOMBRE	VALOR	DESCRIPCIÓN
Tierra	1	Vss	-	0V
Alimentación	2	Vdd	-	+5V
Contraste	3	Vee	-	0 - Vdd
Control de funcionamiento	4	RS	0	Register Select (Selección de Registro)
			1	D0 – D7 considerados como comandos D0 – D7 considerados como datos
	5	R/W	0	Read /_Write (Lectura /_Escritura)
			1	Escribir los datos (del microcontrolador al LCD) Leer los datos (del LCD al microcontrolador)
	6	E	Transición de 1 a 0	Enable (Habilitación)
Datos / comandos	7	D0	0/1	Bit 0 LSB
	8	D1	0/1	Bit 1
	9	D2	0/1	Bit 2
	10	D3	0/1	Bit 3
	11	D4	0/1	Bit 4
	12	D5	0/1	Bit 5
	13	D6	0/1	Bit 6
	14	D7	0/1	Bit 7 MSB

Fuente: Guachalla R. (2020)

Las pantallas LCD monocromáticas “Están basados en el microcontrolador HD44780 (Hitachi) integrado y permite visualizar todas las letras de alfabeto (mayúsculas y minúsculas), letras de alfabeto griego, signos de puntuación, símbolos matemáticos etc.” Guachalla R. (2020). La distribución de sus pixeles para formar un carácter se puede observar en la Figura N°15. También es posible visualizar símbolos creados por el usuario.

Higher 4bit Lower 4bit	0000	0010	0011	0100	0101	0110	0111	1010	1011	1100	1101	1110	1111
xxxx0000		0	a	P	`	f		-	ろ	三	α	ρ	
xxxx0001	!	1	A	Q	a	q	μ	ア	チ	4	ā	q	
xxxx0010	"	2	E	R	b	r	┌	イ	ツ	×	β	θ	
xxxx0011	#	3	O	S	c	s	└	ウ	テ	ε	ε	∞	
xxxx0100	\$	4	D	T	d	t	√	エ	ト	ト	μ	Ω	
xxxx0101	%	5	E	U	e	u	•	オ	ナ	ユ	ι	Ω	
xxxx0110	&	6	F	V	f	v	ヲ	カ	ニ	ヨ	ρ	Σ	
xxxx0111	'	7	G	W	g	w	ア	チ	ヌ	ラ	q	π	
xxxx1000	(8	H	X	h	x	イ	ク	ネ	リ	フ	又	
xxxx1001)	9	I	Y	i	y	ウ	ケ	ル	ル	ウ		
xxxx1010	*	:	J	Z	j	z	エ	コ	ノ	ク	イ	〒	
xxxx1011	+	;	K	L	k	l	オ	サ	ヒ	ロ	×	ア	
xxxx1100	,	<	L	¥	l	l	ホ	シ	フ	フ	φ	円	
xxxx1101	-	=	M	I	m	l	ユ	ズ	ハ	ン	ト	÷	
xxxx1110	.	>	N	^	n	→	ヨ	セ	ホ	ハ	ン		
xxxx1111	/	?	O	_	o	←	ウ	ソ	マ	°	ろ	■	

Figura N°15 Correspondencia entre códigos de caracteres y patrones de caracteres
Fuente: Systronix (2000)

4.4.5 Teclado Matricial

Un teclado es un arreglo de pulsadores conectados por filas y columnas (una matriz de pulsadores) como se puede examinar en la Figura N°16, con la intención de reducir el número de pines necesarios para su conexión. “Para poder leer que tecla ha sido pulsada

se debe de utilizar una técnica de barrido y no solo leer un pin de microcontrolador” Canto C. E. (n.d.)

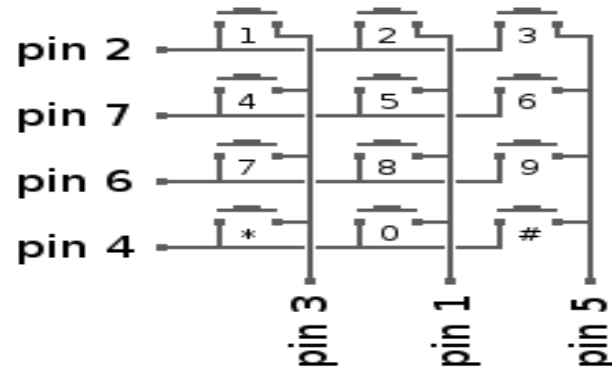


Figura 16. Matriz de pulsadores
Fuente: Castaño S. A. (n.d.).

El procedimiento para detectar la tecla presionada es:

- Inicialmente conectar el teclado matricial en PINES DIGITALES.
- Las Filas del Teclado estarán configurados como SALIDAS.
- Las Columnas del Teclado configurados como ENTRADAS y con el PULLUP activado (por lo tanto, estas entradas siempre estarán recibiendo un 1 lógico, si ningún botón es presionado).
- Se configura todas las salidas (Filas) en 1 lógico o 5v.

Según Castaño S. A. (n.d.).si se Aplica el concepto de la Multiplexación al mandar un 0 lógico por cada fila y vamos a leer todas las columnas, si se detecta que alguna columna recibió el cero lógico, indica que el botón que comparte la fila y la columna fue presionado, en caso contrario vuelvo a colocar la fila en 1 lógico y verifico la siguiente fila.

4.4.6 HC-SR04

HC-SR04 es un sensor de distancia de bajo costo que utiliza ondas de ultrasonido para determinar la distancia de un objeto en un rango de 2 a 450 cm. Sus características Técnicas son sintetizadas por Morgan E. J. (2014) de la siguiente manera

- Fuente de alimentación: +5 V CC
- Corriente de reposo: <2mA
- Corriente de trabajo: 15mA
- Ángulo efectivo: <15°
- Resolución: 0,3 cm
- Ángulo de medición: 30°
- Ancho de pulso de entrada de disparo: 10uS
- Dimensión: 45 mm x 20 mm x 15 mm
- Peso: aprox. 10g

Morgan E. J. (2014). Declara que el sensor HC-SR04 posee dos transductores: un emisor y un receptor piezoeléctricos. Cuando el emisor piezoeléctrico emite 8 pulsos de ultrasonido(40KHz) luego de recibir la orden en el pin TRIG, las ondas de sonido viajan en el aire y rebotan al encontrar un objeto, el sonido de rebote es detectado por el receptor piezoeléctrico, luego el pin ECHO cambia a Alto (5V) por un tiempo igual al que demoró la onda desde que fue emitida hasta que fue detectada como se observa en la Figura N°17, siendo así se puede determinar la distancia con el objeto y el sensor por medio del tiempo del pulso ECO medido por el microcontrolador.

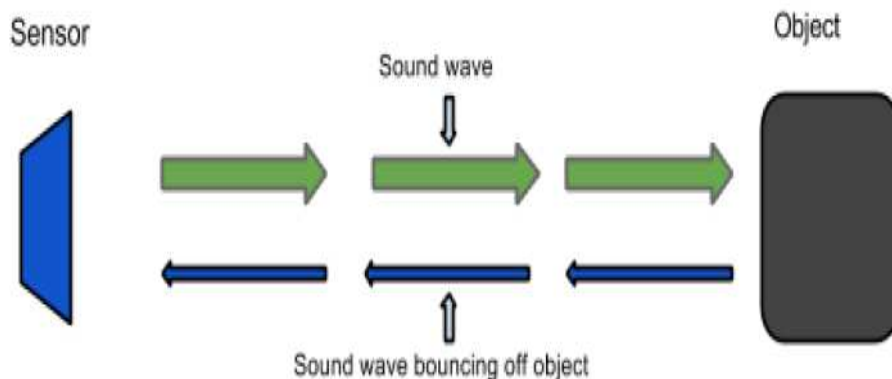


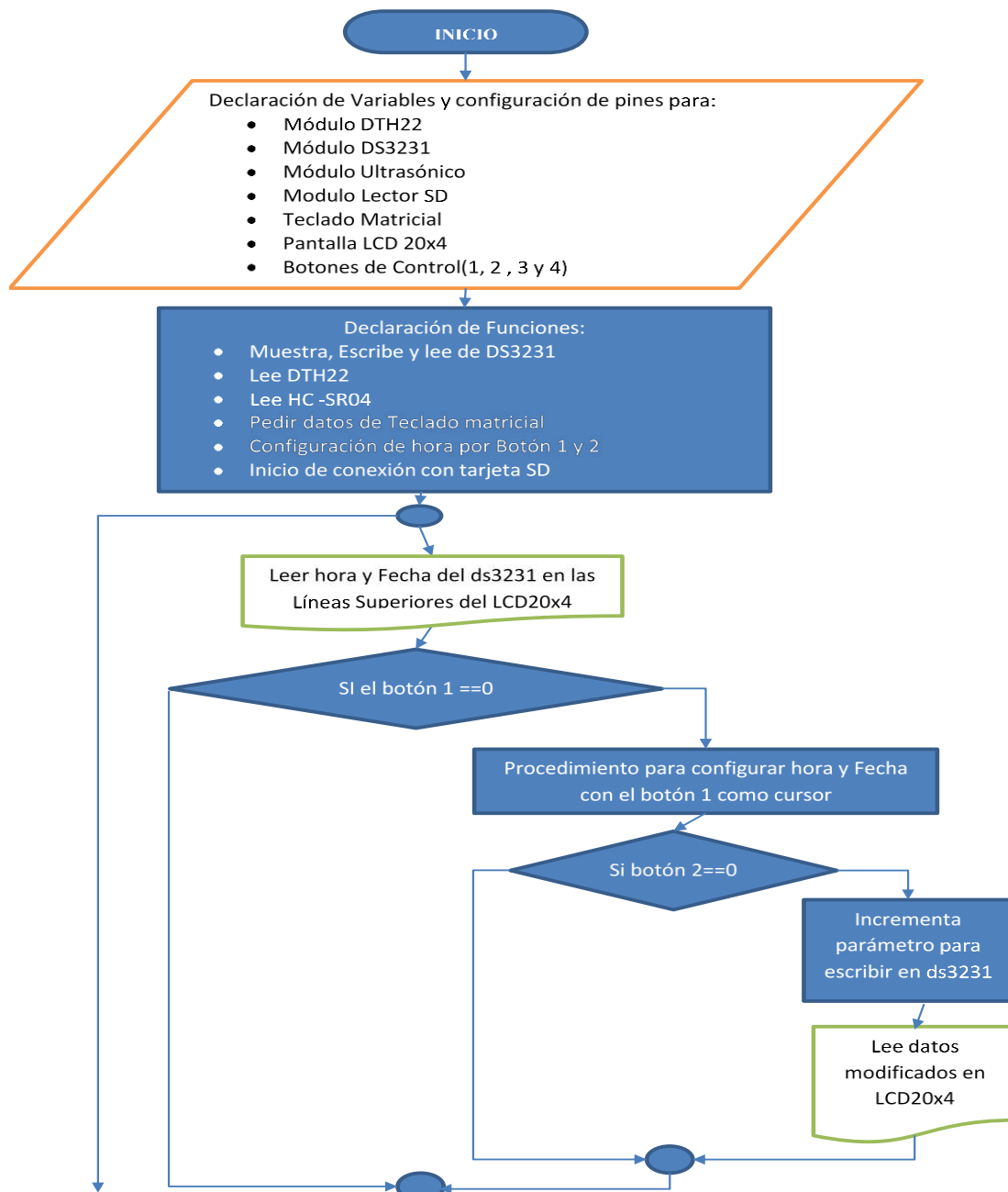
Figura N°17 Funcionamiento del HC-SR04
Fuente: Morgan E. J.(2014).

5. DESARROLLO DEL TRABAJO

5.1 Desarrollo práctico

5.1.1 Diagrama de flujo

El sistema de Registro de Asistencia del personal se sintetiza en el siguiente diagrama de flujo (Figura N°18)



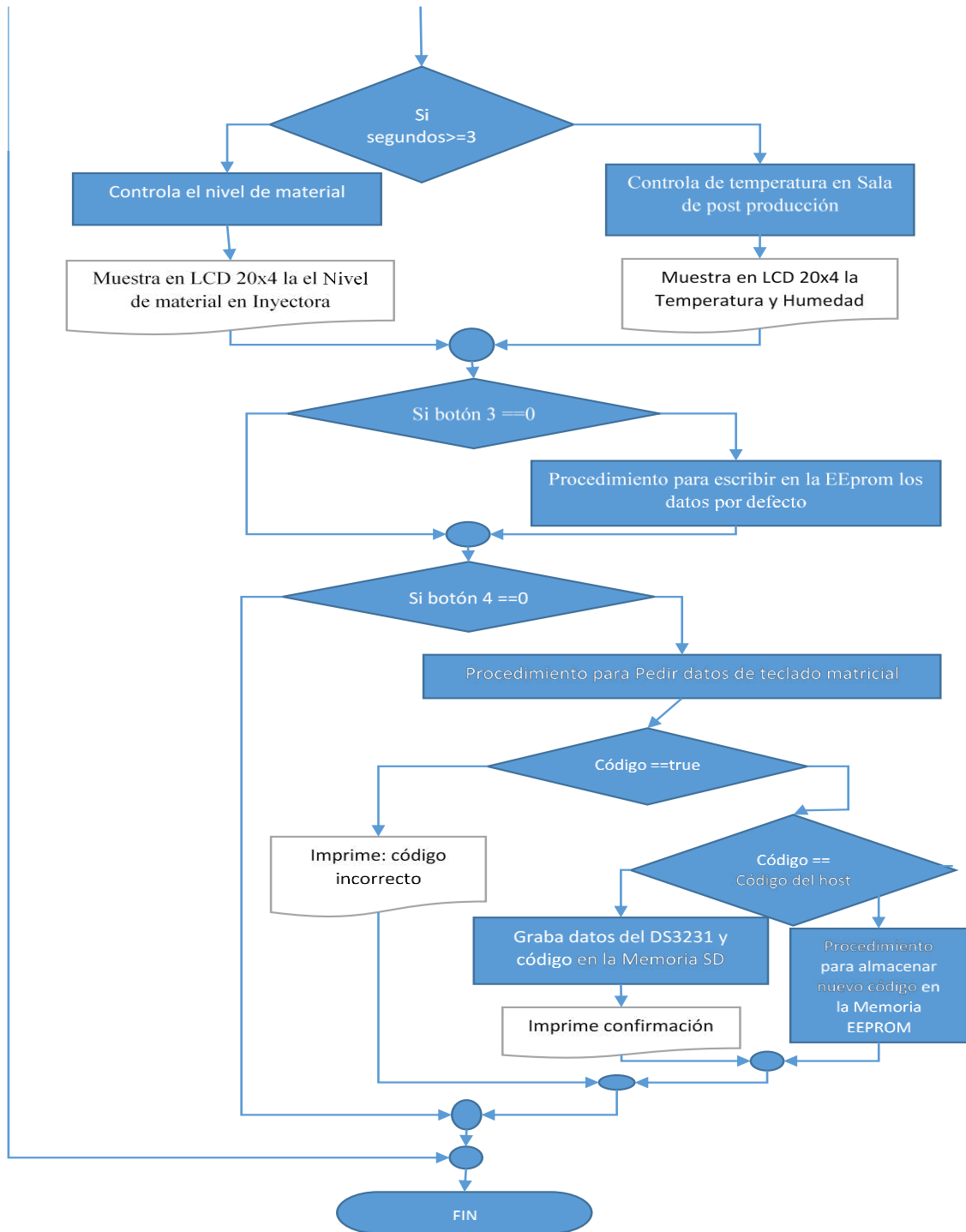


Figura N°18 Flujograma Sistema de Registro de asistencia para la empresa Alvaplas SRL.

Fuente: Elaboración Propia

El sistema de registro de asistencia del personal consiste en una interfaz con el usuario por medio de una pantalla LCD 20x4 y como dispositivos de interacción pulsadores de control y el teclado matricial 4x3 para determinar las funciones de:

- Configuración de Hora y fecha (Botón 1 como cursor + Botón 2 como modificador de parámetros)
- Restablecimiento de Memoria EEPROM (Botón 3)
- Marcado de asistencia (botón 4 + código de acceso mediante el teclado 4x3)

Para hacer uso del registro de asistencia se permite agregar nuevo personal mediante el ingreso del código del administrador (host) donde se puede agregar y/o modificar los datos que se tienen guardados en la memoria EEPROM hasta un límite de 64 usuarios. De otra manera si el código ingresado es correcto se guarda en la memoria SD con los datos de fecha y hora.

De manera por defecto la pantalla LCD 20x4 indica la hora y fecha en las dos primeras filas. Y en las 2 filas de abajo se muestra el nivel de material y la temperatura obtenidas de los sensores correspondiente en tiempo real conmutando cada 30 segundos. De los cuales realizan la activación de Alarma cuando el nivel de material está por debajo del 20% de la capacidad del contenedor de la máquina inyectora, y la activación de un equipo de aire acondicionado.

5.1.2 Diagrama de bloques

El conjunto de Componentes se establecen como Salidas y Entradas para el Microcontrolador PIC18F4550 como se detalla en la Figura N°19:

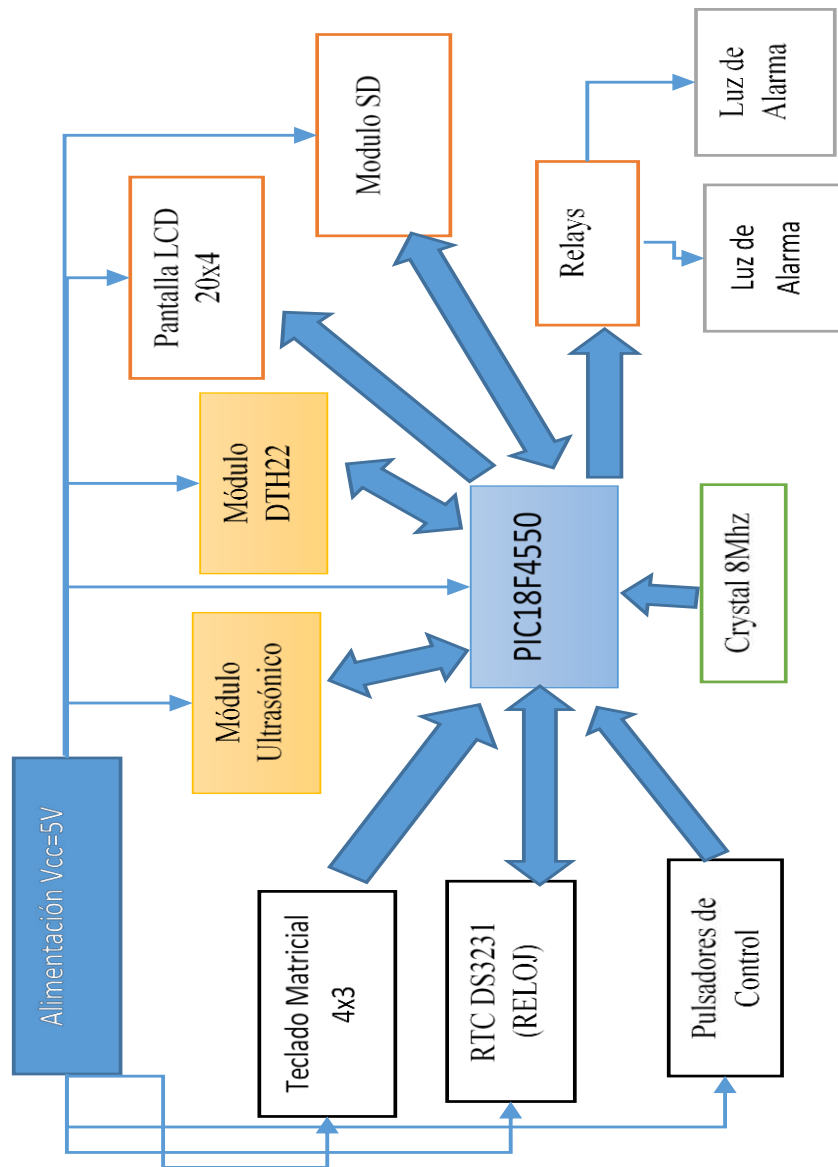


Figura N°19 Diagrama en Bloques Sistema de Registro de Asistencia Empresa Alvaplas SRL.

Fuente: Elaboración Propia

5.2 Desarrollo Experimental

5.2.1 Esquemático

El Circuito Simulado mediante el Software Proteus 8.8 y la simulación de la tarjeta SD con el Programa FAT32 Image se puede apreciar en la FiguraN°20

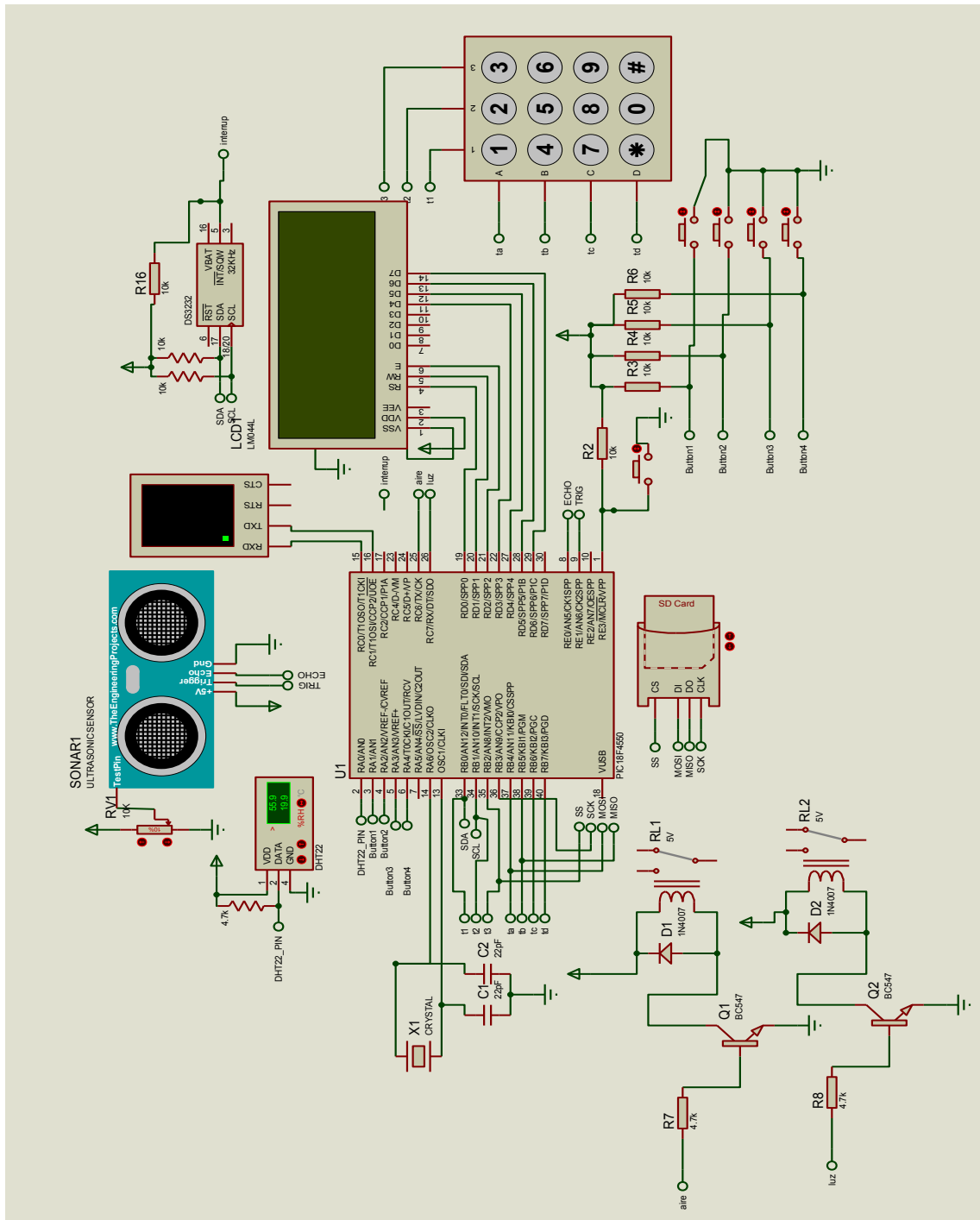


Figura N°20 Esquemático Sistema de Registro de Asistencia Empresa Alvplas SRL.
Fuente: Elaboración Propia

Así también el diseño de la placa PVC en el Software Proteus 8.8 para el circuito impreso como se observa en la Figura N°21

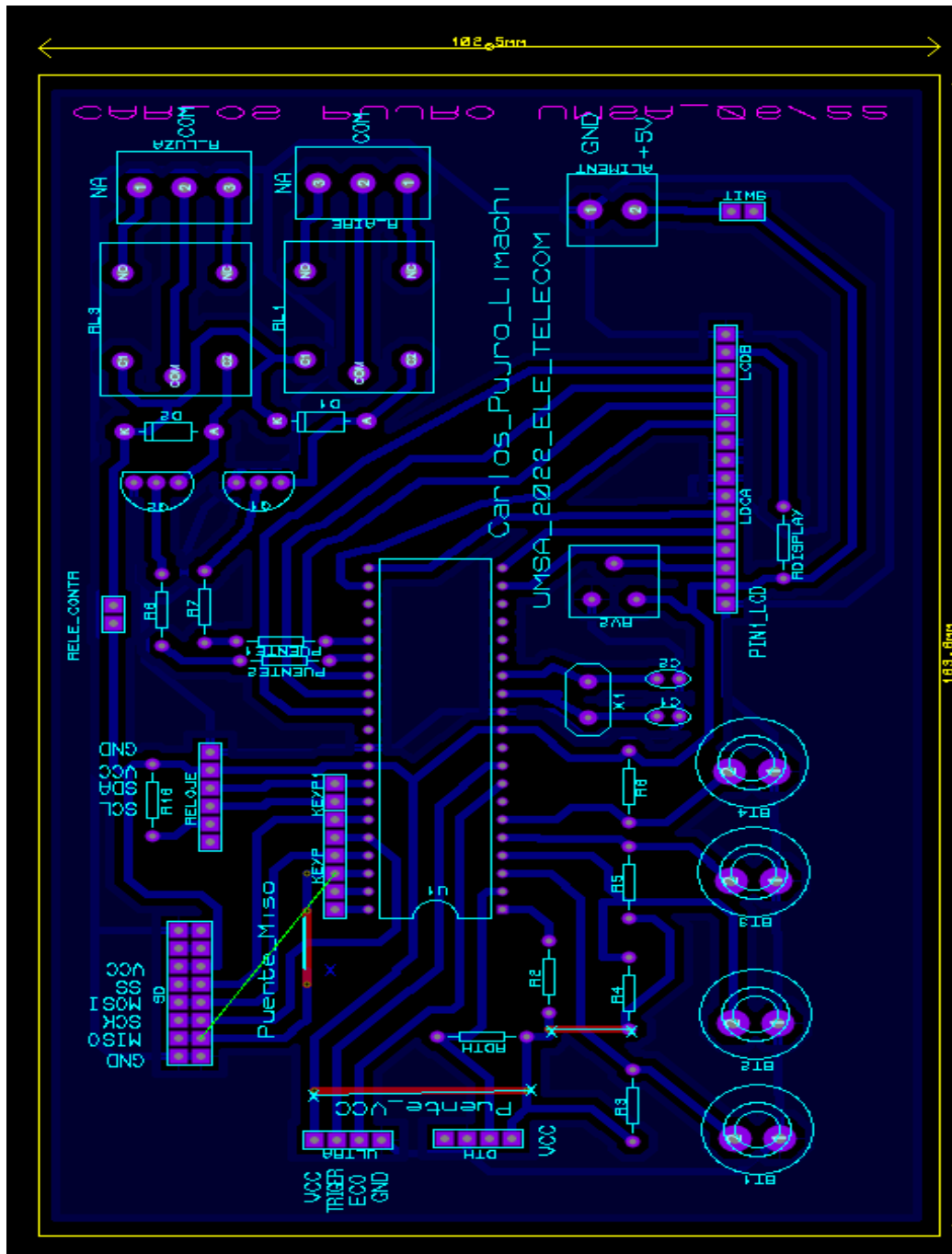


Figura N°21 Modelado para Placa en PVC
Fuente: Elaboración Propia

5.2.2 Código Fuente

El Código Generado mediante el programador CSS PIC Compiler:

```
//Programa: SISTEMA DE REGISTRO DE PERSONAL Y APOYO A FUNCIONES RUTINARIAS
//PARA LA EMPRESA ALVAPLAS SRL
//DISEÑADO POR: CARLOS S. PUJRO LIMACHI
//FECHA: 10/06/2022
#include <18f4550.h> //Incluir libreria de PIC
#fuses HSPLL, PLL2, CPUDIV1 //Habilitar fusibles
#include <lcd.c>
#include <kbd4x4.c> // Libreria para teclado matricial 4x4
#include <mmcsd_m.c> // Incluir libreria MMC/SD card para el manejo de interfaz SD
#include <fat_m.c> // Incluir libreria fat32d para el manejo de interfaz SD
// Definicion de pines para el LCD20x4
#define LCD_RS_PIN PIN_D0
#define LCD_RW_PIN PIN_D1
#define LCD_ENABLE_PIN PIN_D2
#define LCD_DATA4 PIN_D3
#define LCD_DATA5 PIN_D4
#define LCD_DATA6 PIN_D5
#define LCD_DATA7 PIN_D6
//Definicion de Pines para modulo SD
#define MMCSA_PIN_SELECT PIN_B2
#define MMCSA_PIN_SCK PIN_B3
#define MMCSA_PIN_MOSI PIN_B4
#define MMCSA_PIN_MISO PIN_B5
// Definicion de Pines para DTH22
#define DHT22_PIN PIN_A0 // Datos del DTH22 en el PINA0
//Definicion de puerto para teclado Matricial
#define use_portb_kbd //UTILIZAR EL PUERTO B PARA EL TECLADO MATRICIAL
//Definicion de Botones de control
#define button1 PIN_A1 // Button B1 en RA1 pin
#define button2 PIN_A2 // Button B2 en RA2 pin
#define button3 PIN_A3 // Button B3 en RA3 pin para Alarma
#define button4 PIN_A4 // Button PARA ASISTENCIA
//Definicion de pines para HCSR04
#define P_ECHO PIN_E0 //Definir Pines para el Teclado Ultrasónico Eco
#define P_TRIG PIN_E1 ///Definir Pines para el Teclado Ultrasónico Trigger
//Configuraciones
#device PASS_STRINGS = IN_RAM //Primero copie su cadena a la RAM, luego pase la matriz
de RAM a su función
#use delay(clock = 48MHz) //Frecuencia de trabajo
#use fast_io(B) //definir uso de puertoB
#use fast_io(D) //definir uso de puertoD
#use fast_io(C) //definir puerto para salidas
#use standard_io(E) //definir uso de puertoE-para el uso del ULTRASONIC
#use I2C(MASTER, I2C1, FAST = 100000) // Inicializar I2C
#use rs232 (baud=9600, xmit=PIN_C0, rcv=PIN_C1) // Inicializar UART protocol (needed for FAT
library)
```

```

#BYTE INTCON2=0XFF1           // definición PARA LAS PULLUPS directamente del registro

//////////////////// VARIABLES //////////////////////
//variables para teclado
int nu = 0;                    // Contador
int comp;                      //direccionador para comparación de datos con la EEPROM en bucle for
int pos;                       //direccionador para escribir en la EEPROM para nuevo personal
char k;                        // Variable donde se almacena la tecla presionada
int32 acum=0;                  //Variable donde se almacena el total de números (resultado)
int32 num=60;                  //para convertir caracteres a entero
int8 ca0, ca1, ca2, ca3;      //bytes para almacenar en la eeprom
//Variables para el Ultrasonico
int32 fosc = 4800000;         //Variable para el ultrasónico (depende de la frecuencia del uc)
float distancia = 0;          // Variable para almacenar la distancia
//Declaraciones y variables para el modulo SD
int8 fat_status;
FILE LogFile;                 //Establecimiento de OBJETO
int32 host0=6063945;          //número del principal para agregar nuevos
int32 prueb=74845023;         //número de prueba
char clav[]=" ";              //Variable de asignación para mostrar de 9 elementos
char inout[]="ENTRADA";       //Variable de asignación de estado de ingreso o salidas
// Declaracion de variables para el DS3231
char time[] = " : : ";
calendar[] = " / /20 ";
int8 i, second, minute, hour, day, date, month, year;
//Temperatura y humedad variables (DTH22)-----
char Temperature[] = " 00.0 C";
char Humidity[] = " 00.0 %";
int8 T_Byte1, T_Byte2, RH_Byte1, RH_Byte2, CheckSum, previous_second ;
int16 Temp, RH;

////////////////////DECLARACION DE FUNCIONES////////////////////////////////////
//-----Función para el modulo ultrasónico/-----//
float ultra(void)
{
int16 tiempo_eco=0;           //Generación de variables locales
float distancia=0;
float tick = 0;
tick = (4000.0 / fosc) * 8.0; //ajuste mediante la frecuencia del uC
output_high(P_TRIG);         //Comienzo de disparo
delay_us(10);                 //retardo según el fabricante
output_low(P_TRIG);          // final del disparo
while(!input_state(P_ECHO)){ //bucle mientras se recibe el eco
}
set_timer1(0);
while(input_state(P_ECHO)){ //Una vez recibido el Eco
}
tiempo_eco = get_timer1();    //Obtiene el tiempo que se almaceno en el timmer1
distancia = (tiempo_eco/2.0)*tick*34.0; //se obtiene la distancia por medio de la formula
return distancia;
}                               //Final de Rutina ultrasonic

```

```

//-----Función para extraer numero de teclado-----
void Tecladodevuelve(void){

bit_clear(INTCON2, 7); //HABILITAR PULLUPS
acum=0;
nu = 0;
clav=" "; //reiniciar el vector que se manda a la SD
while(TRUE) //bucle temporal
{
k = kbd_getc(); // Lee la tecla presionada
delay_ms(1); //antirebotes
if(k != 0) //Condicion para saber si esta presionada una tecla
{
if(k == '#'){ //Consulta si el carácter es # para borrar
lcd_gotoxy(nu+21, 2); //posiciona en el puntero a borrar
lcd_putc(" "); // Imprime símbolo que demuestra que se borró el anterior carácter
nu--; //se direcciona al elemento a reemplazar
acum=acum/10; //resta el ultimo dígito ingresado
}
else{
if(k=='*'){ // Si se presiona * completa su asistencia
break;
}
else{
clav[nu]=k; //Almacena cada carácter en una cadena
//convertir carácter a número
switch(k){
case '1': num=1;
break;
case '2': num=2;
break;
case '3': num=3;
break;
case '4': num=4;
break;
case '5': num=5;
break;
case '6': num=6;
break;
case '7': num=7;
break;
case '8': num=8;
break;
case '9': num=9;
break;
case '0': num=0;
break;
} //Final de Switch
acum=acum*10; //desplazar de entero a decena a centena a milésima, etc
acum=acum+ num; //acumular valores
nu++;
lcd_gotoxy(nu+21, 2);
}
}
}
}
}
}

```

```

        lcd_putc(k);    // Imprime cada tecla que se presiono
    }
}
}
} //Final de Loop temporal de consulta
delay_ms(150);
bit_set(INTCON2, 7); //desHABILITAR PULLUPS
delay_ms(150);
}
//-----Funciones para el módulo ds3231-----////////////////
//FUNCION PARA LEER DEL DS3231
void DS3231_read(){
    // Función de lectura de ds3231
    i2c_start();           // comienzo de protocolo
    i2c_write(0xD0);       // DS3231 dirección
    i2c_write(0);          // Envía dirección de registro (seconds registro)
    i2c_start();           // Reinicia I2C
    i2c_write(0xD1);       // comienza lectura de datos
    second = i2c_read(1);   // lectura de segundos del register 0
    minute = i2c_read(1);  // lectura de minutos del register 1
    hour = i2c_read(1);    // lectura de horas del register 2
    day = i2c_read(1);     // lectura de días del register 3
    date = i2c_read(1);    // lectura de fecha del register 4
    month = i2c_read(1);   // lectura de mes del register 5
    year = i2c_read(0);    // lectura de año del register 6
    i2c_stop();            // detener I2c
}
//-----Función de Calendario-----
void calendar_display(){
    switch(day){ // Dependiendo del numero del dia devuelve el dia
        case 1: strcpy(calendar, "Dom / /20 "); break;
        case 2: strcpy(calendar, "Lun / /20 "); break;
        case 3: strcpy(calendar, "Mar / /20 "); break;
        case 4: strcpy(calendar, "Mie / /20 "); break;
        case 5: strcpy(calendar, "Jue / /20 "); break;
        case 6: strcpy(calendar, "Vie / /20 "); break;
        case 7: strcpy(calendar, "Sab / /20 "); break;
        default: strcpy(calendar, "*** / /20 "); break;
    }
    calendar[13] = year % 10 + 48;
    calendar[12] = year / 10 + 48;
    calendar[8] = month % 10 + 48;
    calendar[7] = month / 10 + 48;
    calendar[5] = date % 10 + 48;
    calendar[4] = date / 10 + 48;
    lcd_gotoxy(1, 2); // Posición en el display
    printf(lcd_putc, calendar); // muestra en el display
}
//-----Función para mostrar en display la hora y fecha-----
void DS3231_display(){
    // conversion BCD a decimal
    second = (second >> 4) * 10 + (second & 0x0F);
    minute = (minute >> 4) * 10 + (minute & 0x0F);
}

```

```

hour = (hour >> 4) * 10 + (hour & 0x0F);
date = (date >> 4) * 10 + (date & 0x0F);
month = (month >> 4) * 10 + (month & 0x0F);
year = (year >> 4) * 10 + (year & 0x0F);

time[7] = second % 10 + 48;
time[6] = second / 10 + 48;
time[4] = minute % 10 + 48;
time[3] = minute / 10 + 48;
time[1] = hour % 10 + 48;
time[0] = hour / 10 + 48;
calendar_display(); // Llamado a la función Calendario para mostrar en display
lcd_gotoxy(1, 1); // Posicionamiento en display
printf(lcd_putc, time); // Mostrar en display
}
//-----Función de salto de carácter-----
void blink(){ //Salta de posición cuando el se presiona las teclas 1, 2 o 3
int8 j = 0;
while(j < 10 && (input(button1) || i >= 5) && input(button2) && (input(button3) || i < 5)){
j++;
delay_ms(25);
}
}
//-----Función para editar parámetros
int8 edit(parameter, x, y){
while(!input(button1) ); // espera a que se presiones la tecla 1
delay_ms(10); //para antirebotes
while(TRUE){

while(!input(button2)){ // Si la tecla 2 es presionada entonces procede a configurar
delay_ms(100); //para antirebotes
parameter++;
if(((i == 0) || (i == 5)) && parameter > 23) // Limite si las horas son mayor a 23
parameter = 0;
if(((i == 1) || (i == 6)) && parameter > 59) // Limite si los minutos son mayor a 59
parameter = 0;
if(i == 2 && parameter > 31) // Limite si la fecha pasa los 31 días
parameter = 1;
if(i == 3 && parameter > 12) // Limite si los mese pasan de 12
parameter = 1;
if(i == 4 && parameter > 99) // Limite si los años pasan de 100 no probable
parameter = 0;

DS3231_read(); // Llamado a la función de lectura del DS3231
DS3231_display(); // Llamado a la función para mostrar en display

delay_ms(200); // retardo

if(!input(button1) && i < 5 ){
i++; // Variable que determina el desplazamiento al siguiente parámetro
return parameter; // Devuelve parámetro para configurar al DS3231
}
}
}

```



```

}
}
}

//----- DHT22 FUNCIONES-----
//Funcion comienzo de señal
void Start_Signal(void){
    output_drive(DHT22_PIN);           // Configurar el pin como salida
    output_low(DHT22_PIN);             // posiciona el pin como un 0 logico
    delay_ms(25);                      // tiempo de espera
    output_high(DHT22_PIN);            // Salida de Pin activada
    delay_us(30);                      // tiempo de espera
    output_float(DHT22_PIN);           // Configurar el pin como entrada
}
//Funcion revision de respuesta
int1 Check_Response(void){
    set_timer1(0);                    // activa el timmer en 0
    while(!input(DHT22_PIN) && get_timer1() < 150); // Espera hasta que DHT22_PIN sea alto
                                                //(verificación de respuesta de tiempo bajo de 80 µs)
    if(get_timer1() > 149)              // Si se demora mas tiempo muestra Response error
        return 0;                      // Return 0 (Device has a problem with response)
    else {
        set_timer1(0);                // activa el timmer en 0
        while(input(DHT22_PIN) && get_timer1() < 150); // Espera hasta que DHT22_PIN sea bajo
        if(get_timer1() > 149)         // Si se demora mas tiempo muestra Response error
            return 0;                  // en ese caso devuelve 0 que significa que hubo un problema
        else
            return 1;                  // Devuelve 1 si la todo esta normal
    }
}
//Funcion de lectura de dato del dht22-----
int1 Read_Data(int8 *dht_data) {
    int8 j;
    *dht_data = 0;
    for(j = 0; j < 8; j++){
        set_timer1(0);                // Reinicia el Timmer
        while(!input(DHT22_PIN))      // Espera a que el pin sea alto
            if(get_timer1() > 150) {   // Si el tiempo bajo > 100 µs muestra error
                return 1;
            }
        set_timer1(0);                // Reinicia el Timmer
        while(input(DHT22_PIN))       // Espera a que el pin sea bajo
            if(get_timer1() > 150) {   // Si tiempo alto > 100 µs significa error (normalmente tarda 26-
                return 1;              // Devuelve 1 si hay error
            }
        if(get_timer1() > 70)         // Si tiempo alto > ~46µs ==> Sensor enviado 1
            bit_set(*dht_data, (7 - j));
    }
    return 0;                          // devuelve 0 (si esta normal)
}
}

```

```

//////////////////////////////////MAIN//////////////////////////////////
void main()
{
    lcd_init();                // Inicializa la LCD
    kbd_init();                // Inicializa el teclado matricial
    setup_timer_1(T1_INTERNAL | T1_DIV_BY_8); //Activación de Timer uno a preescala 8
    set_tris_c(0x00);          //configuración como salida el puerto c

    output_low(PIN_C7);        // inicio de pin para Luz de emergencia
    output_low(PIN_C6);        // inicio de pin desactivado para aire acondicionado
    //INICIO DE SD
    printf("Initializing FAT library ... ");
    fat_status = fat_init();    //Comprueba si la SD esta conectada (estado 0 si esta conectada)
    if( fat_status == 0 ) {
        // INICIALIZA CON UN OK
        printf("OK");
        ////////////////////////////////////Descomentar cuando se requiera un nuevo documentos de Registro////////////////////////////////
        //mk_file("/Registro.xls");          // Crea el archivo SD
        // fopen("/Registro.xls", "w", &LogFile); // Abrir el archivo con permiso de escritura 'w'
        // fatputs("UMSA 2022 Proyecto de aplicacion para ALVAPLAS SRL.\r\n" , &LogFile); //Generar
TITULO
        //fatputs("Aplicante: Carlos S. Pujro Limachi.\r\n" , &LogFile); //Generar TITULO
        //fatputs("UMSA 2022 Proyecto de aplicacion para ALVAPLAS SRL.\r\n\r\n" , &LogFile); //Generar
TITULO
        //fatputs("\r\n FECHA   ; HORA   ; Nro. CI   ; DETALLE \r\n", &LogFile); //Generar orden
de datos
        // fatclose(&LogFile);                //Cerrar el Archivo
    }

    else // Si fat_status=1 significa error
        printf("Error!");

    ////////////////////////////////////LOOP PRINCIPAL-----
    while(TRUE)
    {
        // LLAMADAS DE FUNCIONES PARA EL DISPLAY
        DS3231_read();                // Llamado de la función del lectura del DS3231
        DS3231_display();            // Llamado de función para mostrar en display los datos de ds3231
        lcd_gotoxy(11,1);            //Posición para LOGO
        lcd_putc("UMSA_2022");        //Mostrar a la esquina sup. Derecha logo

        //BOTON PARA CONFIGURAR LA HORA Y FECHA----
        if(!input(button1)){         // Cuando el botón 1 está presionado entonces edita
            i = 0;
            hour = edit(hour, 1, 1);
            minute = edit(minute, 4, 1);
            while(!input(button1));   // espera a que el botón 1 sea nuevamente presionado
            while(TRUE){
                while(!input(button2)){ // rutina para modificar parametros con boton 2
                    day++;              // incrementar día
                    if(day > 7) day = 1;
                }
            }
        }
    }
}

```

```

    calendar_display();           // Llamado de la función para mostrar calendario
    lcd_gotoxy(1, 2);           // posición en display
    printf(lcd_putc, calendar);   // muestra calendario
    delay_ms(200);
}
lcd_gotoxy(1, 2);
lcd_putc(" ");
blink();
lcd_gotoxy(1, 2);
printf(lcd_putc, calendar);     // muestra calendario
blink();                         // llamado de la función de salto
    if(!input(button1))         // una vez concluido configurado el parametro salta
        break;
}
date = edit(date, 5, 2);        // Configurar fecha
month = edit(month, 8, 2);      // Configurar mes
year = edit(year, 13, 2);      // Configurar Año
// Conversion BCD a decimal
minute = ((minute / 10) << 4) + (minute % 10);
hour = ((hour / 10) << 4) + (hour % 10);
date = ((date / 10) << 4) + (date % 10);
month = ((month / 10) << 4) + (month % 10);
year = ((year / 10) << 4) + (year % 10);
// Escribir datos en el ds3231
i2c_start();                   // Inicio de i2c
i2c_write(0xD0);               // Direccion de DS3231
i2c_write(0);                  // Envia la dirección del registro de segundos)
i2c_write(0);                  // reinicia segundos
i2c_write(minute);             // Envia datos de minutos al DS3231
i2c_write(hour);              // envia datos de horas al DS3231
i2c_write(day);               // Envia datos de dias al DS3231
i2c_write(date);              // Envia datos de fecha al DS3231
i2c_write(month);             // Envia datos de mes al DS3231
i2c_write(year);              // Envia datos de año al DS3231
i2c_stop();                   // detiene i2c
delay_ms(200);                 // retardo de establecimiento
}
// -----Activacion de datos de Host0-----
if(!input(button3)){           // si el boton 3 es presionado
    //grabacion de la Eeprom con el dato del host (principal) DIRECCION 0
    ca0=make8(host0, 0);
    ca1=make8(host0, 1);
    ca2=make8(host0, 2);
    ca3=make8(host0, 3);
    write_eeprom(0, ca0); //grabacion de byte LSB en EEprom direccion 0 para el host0
    write_eeprom(1, ca1); //grabacion de byte LSB en EEprom direccion 1 para el host0
    write_eeprom(2, ca2); //grabacion de byte LSB en EEprom direccion 2 para el host0
    write_eeprom(3, ca3); //grabacion de byte LSB en EEprom direccion 3 para el host0

    ca0=make8(prueb, 0);
    ca1=make8(prueb, 1);
    ca2=make8(prueb, 2);

```

```

ca3=make8(prueb, 3);

write_eeprom(8, ca0); //grabación de byte LSB en EEprom dirección 0 para el usuario prueba
write_eeprom(9, ca1); //grabación de byte LSB en EEprom dirección 1 para el usuario prueba
write_eeprom(10, ca2); //grabación de byte LSB en EEprom dirección 2 para el usuario prueba
write_eeprom(11, ca3); //grabación de byte LSB en EEprom dirección 3 para el usuario prueba
}
//-----RUTINA PARA ESTABLECER ASISTENCIA-----
if(input(button4)==0){ //Si se presionó el botón 4
  while(!input(button4)); // ESPERA QUE SE SUELTE EL BOTON
  lcd_putc("\f"); //LIMPIA DISPLAY
  lcd_gotoxy(5,1); //Posición
  lcd_putc("_UMSA_2022"); //Muestra logo
  lcd_gotoxy(3,2); //Posición
  lcd_putc(" ELECTR_TELECOM");
  lcd_gotoxy(21,1); //Posición
  lcd_putc(" INGRESE USUARIO");
  Tecladodevuelve(); //llamada de la función para el teclado matricial
//-----convirtiendo el dato ingresado en bytes separados
ca0=make8(acum, 0);
ca1=make8(acum, 1);
ca2=make8(acum, 2);
ca3=make8(acum, 3);
//comprobar el valor ingresado con las dirección de la EEPROM
for(int ii=0; ii<244; ii+=4){
  comp=0;
  if(ca0==read_eeprom(ii)) comp++; //Ajuste para comprobación de los datos Guardados
  if(ca1==read_eeprom(ii+1)) comp++;
  if(ca2==read_eeprom(ii+2)) comp++;
  if(ca3==read_eeprom(ii+3)) comp++;
  delay_ms(10); //retardo para q compruebe bien
  if(comp==4){ // if datos ingresados coinciden con alguna dirección de la EEPROM
    if(ii==0){ //Cuando se ingresa con el host0 en la dirección 0
      lcd_gotoxy(21,1);
      lcd_putc(" NUEVO_PERSONAL ");
      delay_ms(2000);
      while(TRUE) //bucle temporal
      {
        lcd_gotoxy(21,1); //FILA 1
        lcd_putc("INGRESE_ITEM_(1-60)");//Mensaje para introducir posición
        lcd_gotoxy(21,2); //FILA 2
        lcd_putc(" ");//Mensaje de limpieza
        delay_ms(500);
        Tecladodevuelve(); //llamada de la función para obtener la posición
        pos=acum; //Guarda la posición
        if(pos>0 && pos<61){ //Si la posición está en el Rango de la EEPROM procede...
          //lcd_clear();
          lcd_gotoxy(21,1); //FILA 1
          lcd_putc(" NRO_USUARIO_NUEVO ");//Mensaje para introducir posición
          lcd_gotoxy(21,2); //FILA 2
          lcd_putc(" ");//Mensaje de limpieza

```



```

        fclose(&LogFile);                //Cierra el archivo
    } // if( fat_status == 0 )
    else {                               // Si fat_status=1 significa error
        lcd_gotoxy(21,2);
        lcd_putc("  error_  ");
        delay_ms(10000);
    }
    break;                               //Salir del ciclo FOR
}
} //Final de if datos ingresados coinciden con alguna dirección de la EEPROM
} //Final de For de verificación de valores con EEPROM
if(comp!=4) { //Condición en la que los datos ingresador NO coinciden con alguna dirección de
la EEPROM
    lcd_gotoxy(21,1);
    lcd_putc("  USUARIO  ");
    lcd_gotoxy(21,2);
    lcd_putc("  INCORRECTO  ");
    delay_ms(1000);
}
    lcd_putc("\n"); //LIMPIA DISPLAY
} // FINALIZACION DE IF DE PARA ESTABLECER ASISTENCIA

//-----Condional para establecer que mostrar cada 30 segundos-----
if(second>=30){
    if(((Temp / 100) % 10)>=2){ //condicional cuando la temperatura esta menor igual a 20 grados
//lógica directa se activa con nivel BAJO el relay
        output_high(PIN_C6); //pin c6 como salida para habilitar aire acondicionado
    }
    else{
        output_low(PIN_C6); //se activa aire acondicionado cuando se excede los 30 grados
    }
}
//-----Rutina DTH22-----
if( second != previous_second ) {
    previous_second = second;
    Start_Signal(); // Envía comienzo de señal
    if(Check_Response() { // Verifique si hay una respuesta del sensor (si está bien,
comience a leer los datos de temperatura y humedad)
        // si hay Respuesta entonces leer (y guardar) datos del sensor DHT22 y verificar errores de tiempo
de espera
        if(Read_Data(&RH_Byte1) || Read_Data(&RH_Byte2) || Read_Data(&T_Byte1) ||
Read_Data(&T_Byte2) || Read_Data(&Checksum)) {
            lcd_gotoxy(21, 1); // posicion en LCD
            lcd_putc("  Time Out!  ");
            lcd_gotoxy(21, 2); // posicion en LCD
            lcd_putc("  "); //limpieza de fila
        }
        else { // Si no hay error de tiempo de espera
            if(CheckSum == ((RH_Byte1 + RH_Byte2 + T_Byte1 + T_Byte2) & 0xFF)) {
                // Si no hay un error de suma de comprobación
                RH = RH_Byte1; //Asigna el primer BYTE
                RH = (RH << 8) | RH_Byte2; //realiza una mascara
                Temp = T_Byte1; //Asigna el valor a Temp
            }
        }
    }
}

```

```

Temp = (Temp << 8) | T_Byte2; //realiza una mascara
if( RH >= 1000) //si el dato sobrepasa los 1000 entonces la humedad es 100
  Humidity[0] = '1';
else
  Humidity[0] = '-';
  if(Temp > 0x8000) { //Si la temperatura sobrepasa los 8000 entonces la temperatura esta por
debajo de los 0 grados
    Temperature[0] = '-';
    Temp = Temp & 0x7FFF;
  }
  else
    Temperature[0] = '-'; //Escritura de los datos de temperatura
    Temperature[1] = (Temp / 100) % 10 + 48;
    Temperature[2] = (Temp / 10) % 10 + 48;
    Temperature[4] = Temp % 10 + 48;
    Temperature[5] = 223; // de acuerdo al LCD es el número que corresponde al
carácter °
    Humidity[1] = (RH / 100) % 10 + 48; //Escritura de los datos de Humedad
    Humidity[2] = (RH / 10) % 10 + 48;
    Humidity[4] = RH % 10 + 48;
    lcd_gotoxy(21, 1); lcd_putc("Temperatura ="); //Muestra etiquetas
    lcd_gotoxy(21, 2); lcd_putc(" Humedad =");
    lcd_gotoxy(34, 1); // Posiciona en el display
    printf(lcd_putc, Temperature); // envía los datos de temperatura
    lcd_gotoxy(34, 2); // Posiciona en el display
    printf(lcd_putc, Humidity); // envía los datos de humedad
  }

else { // EN caso de Falla
  lcd_gotoxy(21, 1);
  lcd_putc(" Revise_Error! "); //Muestra mensaje de error
  lcd_gotoxy(21, 1); // Posiciona en el display
  lcd_putc(" "); // Limpia toda la fila
} // Fin de else

} // Fin de else

} // Fin de If

else { // En caso de no haber respuesta del sensor
  lcd_gotoxy(21, 1); // mostrar el mensaje
  lcd_putc(" No responde ");
  lcd_gotoxy(21, 2);
  lcd_putc(" desde el sensor ");
}
}

delay_ms(50);
}
else{ //Instrucciones del Ultrasonico
  distancia = ultra(); // Obtiene la distancia y la almacena
  distancia = (150-distancia)*100/150; //Ajuste para 150cm que es lo que mide el contenedor

```

```

delay_ms(200);
lcd_gotoxy(21,1);           //posición en LCD
lcd_putc(" NIVEL DE MATERIAL: "); //Etiqueta
lcd_gotoxy(21,2);           //posición en LCD
lcd_putc("      %      "); //Etiqueta
lcd_gotoxy(26,2);
printf(lcd_putc," %0.2f ", distancia); // Imprime la distancia
delay_ms(500);
if(distancia<=20){
  output_high(PIN_C7);      // Activa el pin para luz de advertencia
}
else { //
  output_low(PIN_C7);      // Desactiva luz de advertencia si el nivel es superior al 20%
}
}
} //Final de bucle principal
} //Final de Void Main

```

5.3 Análisis de Costos y Duración

5.3.1 Costo de Materiales

Para la implementación del Sistema de Registro de personal para la empresa Alvaplas SRL. Se tiene una estimación detallada en la Tabla N°7 de los costos de materiales utilizados

Tabla N°7
Costos de Materiales (Expresado en Bolivianos)

DETALLE	CANT.	PRECIO	TOTAL
PIC18F4550	1,00	60,00	60,00
LCD 4X20	1,00	45,00	45,00
TECLADO MATRICIAL 4X3	1,00	30,00	30,00
HSR04	1,00	20,00	20,00
DTH22	1,00	45,00	45,00
DS3231	1,00	25,00	25,00
MODULO SD	1,00	20,00	20,00
RELAY	2,00	5,00	10,00
CRISTAL 8MHZ	1,00	5,00	5,00
CONDENSADORES 22PF	2,00	0,50	1,00
RESISTENCIAS (10K,1K,4,7K Y 330) DE 1/4 (W)	10,00	0,20	2,00
POTENCIOMETRO 10K	1,00	2,00	2,00
DIODOS 1N4007	2,00	0,20	0,40
PULSADORES	4,00	1,00	4,00
ZOQUET 40PINES	1,00	2,00	2,00

ESPADINES MACHO Y HEMBRA	3,00	3,00	9,00
PLACA PVC	1,00	10,00	10,00
TABLERO Y TUBOS	1,00	15,00	15,00
PIEZAS EN 3D	18,00	2,00	36,00
TOTAL			341,40

Fuente: Elaboración propia

5.3.2 Costo de Mano de Obra

La estimación de costo de mano de obra se detalla en la Tabla N°8 donde se divide en dos columnas de costos imprescindibles (Real) y los costos adicionales que no afecta en el funcionamiento del Sistema de Registro de Asistencia y Apoyo En Operaciones Rutinarias.

Tabla N°8
Costos de Mano de obra (Expresado en Bolivianos)

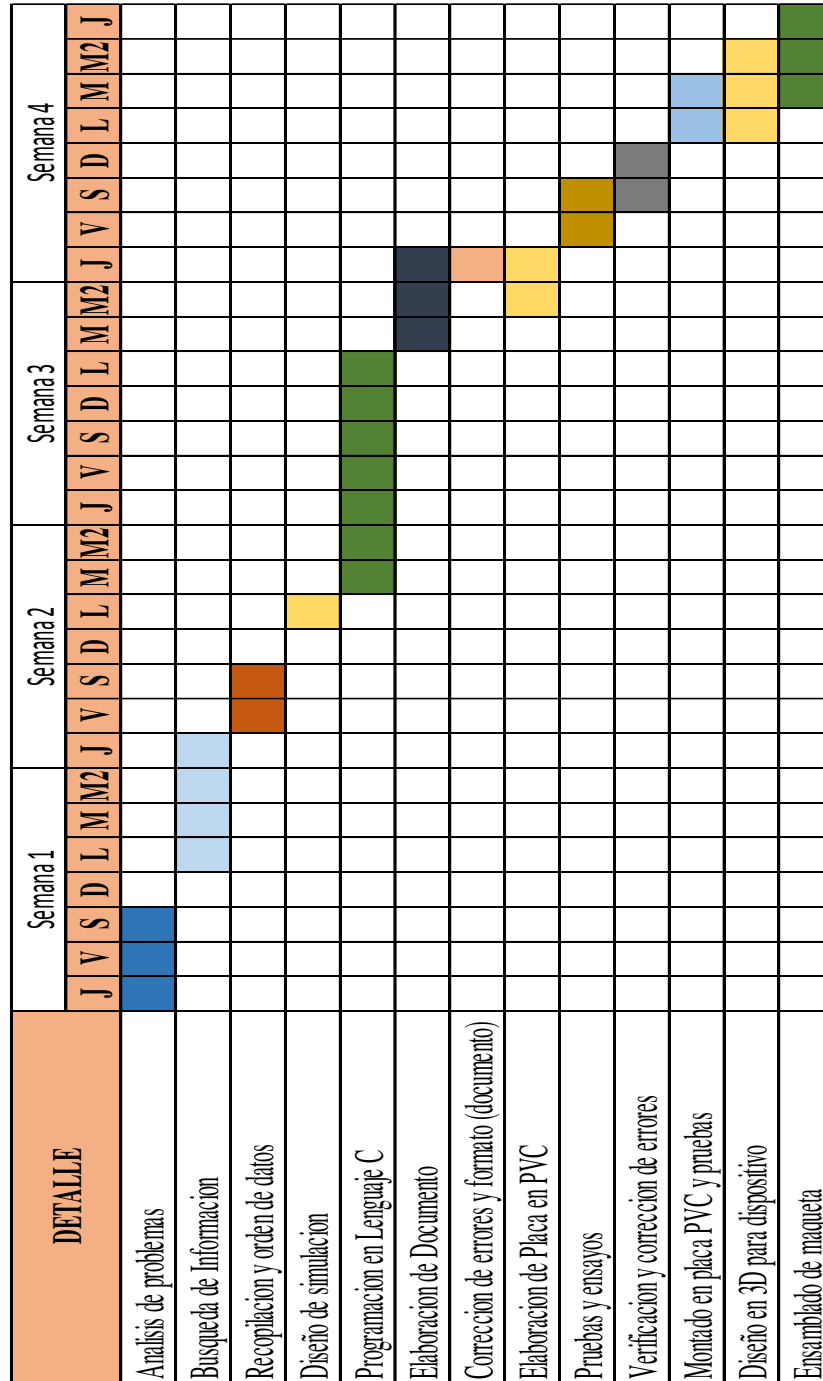
DETALLE	REAL	ADICIONAL	TOTAL
Búsqueda de Información y Recopilación y orden de datos	20,00		20,00
Diseño de simulación	10,00		10,00
Programación en Lenguaje C	50,00		50,00
Elaboración de Placa en PVC	20,00		20,00
Diseño en 3D para dispositivo	30,00		30,00
Ensamblado de maqueta		30,00	30,00
TOTAL			150,00

Fuente: Elaboración propia

5.3.3 Tiempo Empleado

El tiempo estimado para la realización del trabajo de aplicación se considera a partir del 12 de Mayo del 2022 como se detalla en la Tabla N°9, establecido en intervalos de siete días a la semana con una duración de cuatro semanas en total.

Tabla N°9
Diagrama Gantt para tiempo de trabajo de aplicación



Fuente: Elaboración propia

6. CONCLUSIONES

Debido a que Alvaplas SRL. es denominada una PYME en etapa de desarrollo, utiliza un sistema de registro de personal manual mediante uso de hojas de firmas (planillas), el cual no cubre de manera adecuada la necesidad de determinar datos de entradas, salidas, atrasos y faltas como se observó en la definición de las funciones de un sistema de control de personal.

En este sentido, la implementación de un sistema de registro de asistencia de bajo costo para un número limitado de personal de acuerdo al tamaño de una Pyme puede ser proporcionado por dispositivos de registro físico como tarjetas o por contraseña. Para el caso de la Empresa Alvaplas SRL., tales dispositivos están administrados por un microcontrolador de gama mejorada como el PIC18F4550, por su bajo costo y alto rendimiento como se observó en el conjunto de instrucciones, registros especiales y puertos con los que cuenta. Asimismo, el microcontrolador tiene la capacidad de administrar funciones complementarias debido a que cuenta con espacio de memoria de programa útil para tareas específicas adicionales.

El sistema se diseñó específicamente para la Empresa Alvaplas SRL, la cual está dedicada a la fabricación de plantas de Zapato, con la posibilidad de adicionar sensores que ayuden con las funciones complementarias de los operadores de la empresa como ser la medición de nivel de material de la maquinaria inyectora de caucho y el control de temperatura del ambiente de postproducción. Lo anterior significaría alivianar las tareas de los operadores y del personal administrativo de la empresa.

Desde la perspectiva tecnológica, la solución más adecuada para mejorar el deficiente método de control y gestión de asistencia del personal en la Empresa Alvaplas SRL. puede ser también aplicable en empresas de similar estructura (Pyme). Esta solución es el desarrollo un sistema de Registro (control) de asistencia del personal con el empleo de un microcontrolador como base, que hace uso de un dispositivo RTC como reloj independiente conjuntamente memorias SD y EEPROM para un registro de datos que garantiza un sistema estable y confiable.

En este sentido se puede modificar periféricos específicos de acuerdo a las necesidades de la Empresa como ser: teclados numéricos; lectores de tarjetas con RFID; lectores de código de barras o QR. Y finalmente implementar un Sistema de control de personal con equipo biométrico cuando la empresa escale a un tamaño superior.

7. BIBLIOGRAFÍA

- Angulo J.M. & Angulo I, (1999). Microcontroladores PIC: Diseño práctico de aplicaciones (2da ed.). McGrawHill Interamericana.
- Aosong (n.d.) Digital-output relative humidity & temperature sensor/module AM2303. Recuperado de: <https://cdn-shop.adafruit.com/datasheets/DHT22.pdf>
- Canto C. E. (n.d.) El teclado Matricial. Recuperado de: http://galia.fc.uaslp.mx/~cantocar/microcontroladores/SLIDES_8051_PDF/20_MATRI.PDF
- Castaño S. A. (n.d.). Teclado Matricial. Recuperado de: <https://controlautomaticoeducacion.com/arduino/teclado-matricial-keypad/>
- CENAC (2009). SISTEMA DE CONTROL DE PERSONAL. Recuperado de;
- Colegio de Bachilleres del Estado de Baja California (2014). PROCEDIMIENTO PARA EL CONTROL DE ASISTENCIA DEL PERSONAL ADMINISTRATIVO Y DE CONFIANZA. Recuperado de: <http://info.cobachbc.edu.mx/static/cms/docs/procedimientos/05.pdf>
- Cortés N. (2020). Sistemas de control de asistencia laboral: elegir el mejor. Recuperado de: <https://www.geovictoria.com/cl/tecnologia/sistemas-de-control-de-asistencia-laboral-elegir-el-mejor/>
- Dahl N. (2021).La resistencia Pull-up Como funciona y elegir un valor. Recuperado de: <https://www.build-electronic-circuits.com/pull-up-resistor/>
- Dallas (2005). Extremely Accurate I2C-Integrated RTC/TCXO/Crystal. Recuperado de: <https://www.alldatasheet.com/datasheet-pdf/pdf/112132/DALLAS/DS3231.html>
- Delpueche S. (2022). ¿Qué ganas optimizando el sistema de control de asistencia de tu empresa? Recuperado de: <https://factorialhr.es/blog/control-de-asistencia-laboral/>
- DZLCM (n.d.). Módulo de matriz de puntos de caracteres — 2004C. Recuperado de: https://www.dz lcm.com/product/character_dot_matrix_module_2004c?gclid=CjwKCAjwv-GUBhAzEiwASUMm4o7pA_GwLNI-3CC05cGu-O4i4hT9qGWlen3-TFuumFb9lCp-3-tsOxoC49kQAvD_BwE
- EDEBE (2022). Control de Personal. Recuperado de: https://www.edebe.com/ciclosformativos/zona-publica/ut03_830029_la_oarh_cas.pdf

- Fernandez Y. (2019). FAT32, NTFS o exFAT: qué sistema de archivos elegir al formatear tu disco duro o USB. Recuperado de: <https://www.xataka.com/basics/sistemas-de-archivo-como-saber-cual-elegir-al-formatear-tu-disco-duro-o-usb>
- Fernandez Y. (2022). Tipos de tarjetas SD: qué significan sus clases, tipos y numeraciones. Recuperado de: <https://www.xataka.com/basics/tipos-tarjetas-sd-que-significan-sus-clases-tipos-numeraciones>
- Guachalla R. (2020). Modelo de Programación del PIC18F4550. Recuperado de: <https://es.scribd.com/document/446462888/P01-Modelo-de-Programacion-del-PIC18F4550-pdf>
- <https://www.repositoriodigital.ipn.mx/bitstream/123456789/10233/1/Manual2009.pdf>
- Kimaldi(2022). Control de Personal. Recuperado de: https://www.kimaldi.com/blog/control_de_acceso_y_presencia/control_de_personal/
- Microchip (2009). PIC18F2455/2550/4455/4550 Data Sheet. Recuperado de: <https://ww1.microchip.com/downloads/en/devicedoc/39632c.pdf>
- Microcontroladores (n.d.) Microcontroladores PIC. Recuperado de: <https://microcontroladores.com/pic/>
- Morgan E. J. (2014). HCSR04 Ultrasonic Sensor. Recuperado de: <https://pdf1.alldatasheet.com/datasheet-pdf/view/1132203/ETC2/HC-SR04.html>
- Ondasat (2021). Que son los sistemas biométricos? Recuperado de: <https://www.ondasat.net/biometricos/>
- Quiroga D. (2020). El Segmento Stack (pila)Recuperado de: <https://clibre.io/blog/por-secciones/codigo/item/386-el-segmento-stack-la-pila>
- RRHHDigital (2021). Cuatro útiles sistemas de control de acceso de personal. Recuperado de: <http://www.rrhhdigital.com/secciones/tecnologia-e-innovacion/149464/Cuatro-utiles-sistemas-de-control-de-acceso-de-personal>
- Ruoti S. (2002). Cómo funcionan los códigos QR y qué peligros suponen para tu celular. Recuperado de :<https://www.bbc.com/mundo/noticias-61084121>
- Systronix (2000), Systronix 20x4 LCD Brief Technical Data. Recuperado de: http://www.systronix.com/access/Systronix_20x4_lcd_brief_data.pdf
- Tecmikro(2022). Introducción a la programación de microcontroladores PIC. Recuperado de: <https://tecmikro.com/content/8-programacion-microcontroladores-pic>
- Tocci R. J. & Widmer N. S. (2003). Sistemas Digitales Principios y Aplicaciones. (8va ed.) México: Prentice Hall
- Verle M. (n.d.). Microcontroladores PIC – Programación en C. Recuperado de: <https://www.mikroe.com/ebooks/microcontroladores-pic-programacion-en-c-con-ejemplos>

- Wokrmeter (n.d.) ¿Cuáles son las 6 mejores herramientas para el control de empleados? Recuperado de: <https://www.workmeter.com/blog/6-herramientas-concretas-para-el-control-de-empleados/>

8. ANEXOS



Anexo 1. Maquinaria de la Empresa ALVAPLAS SRL.



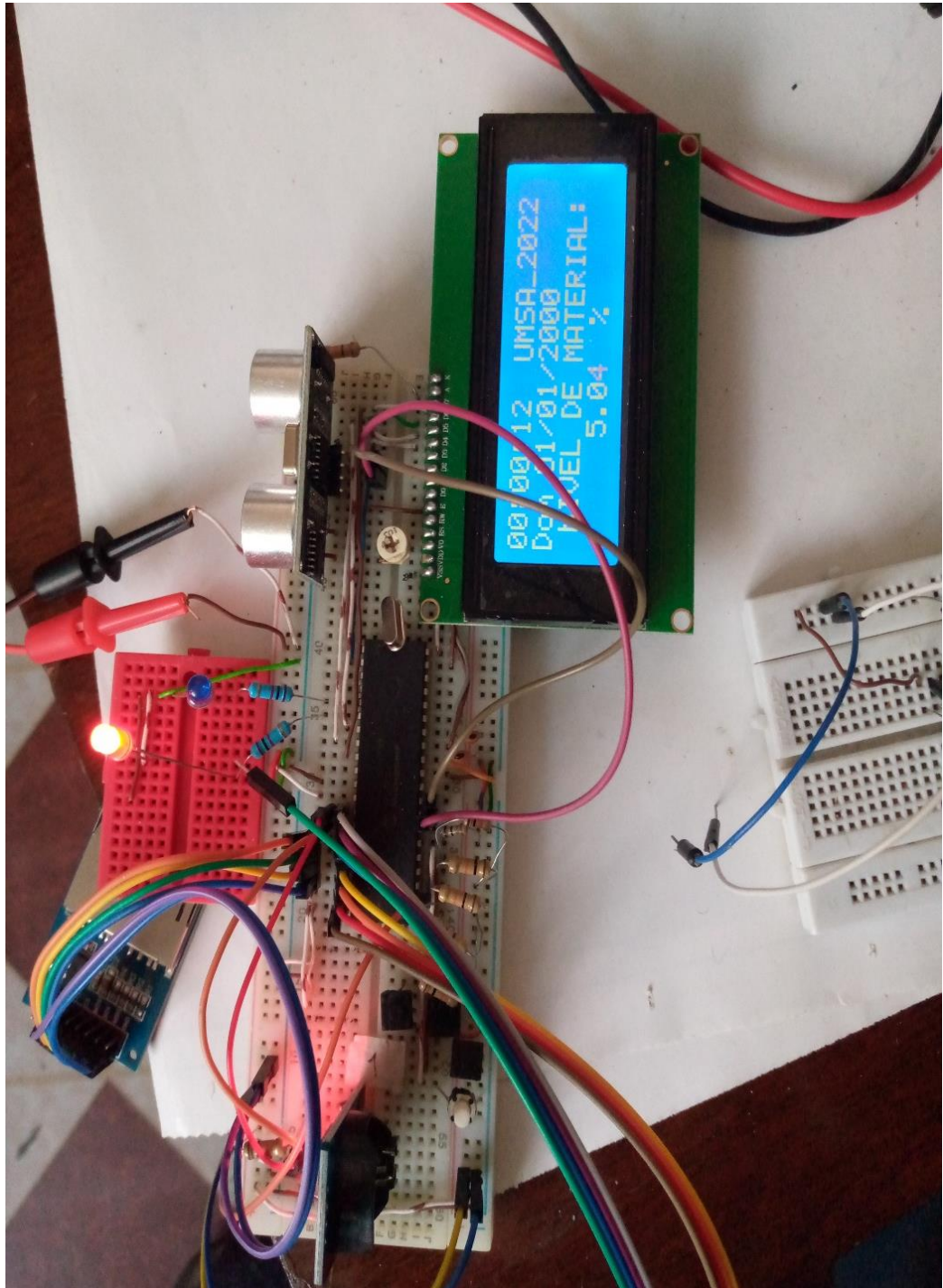
Anexo 2. Ambiente de Post Producción de la Empresa ALVAPLAS SRL.



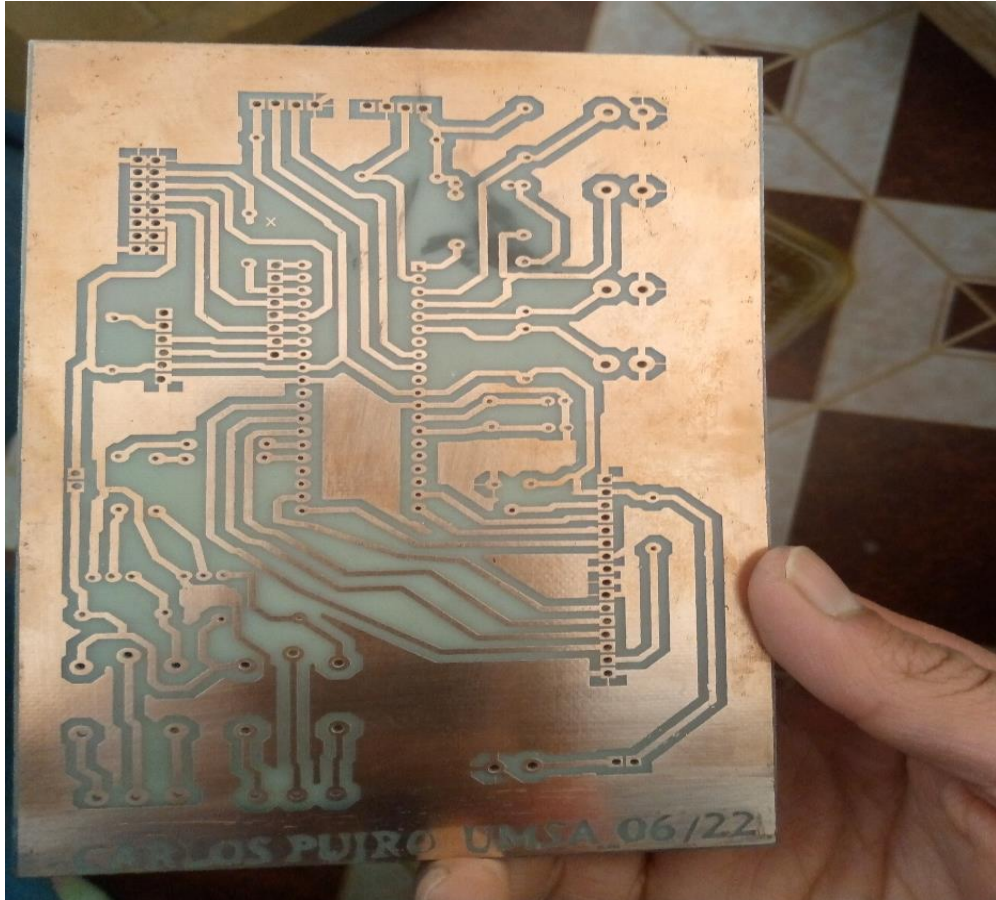
Anexo 3. Operador de la Empresa ALVAPLAS SRL.



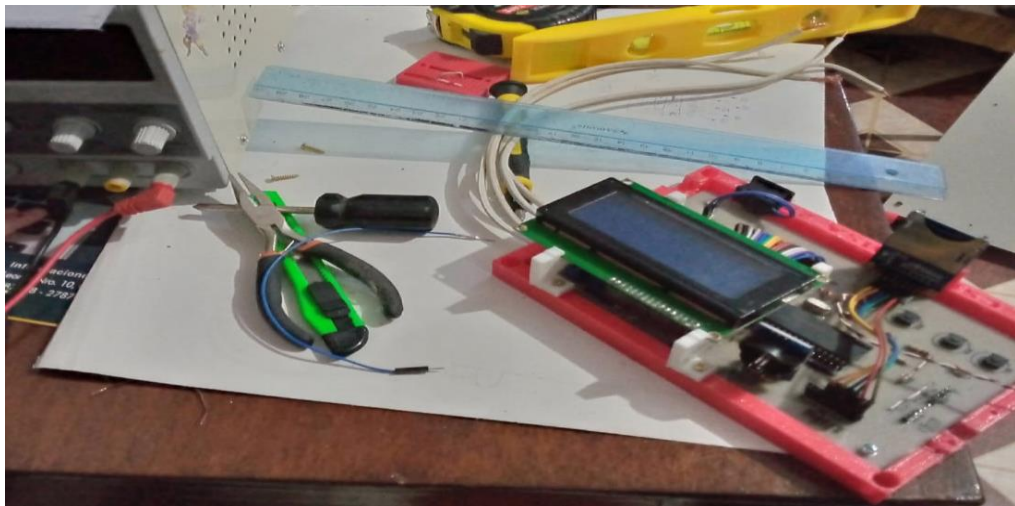
Anexo 4. Contenedor de material para maquina Inyectora de la Empresa ALVAPLAS SRL.



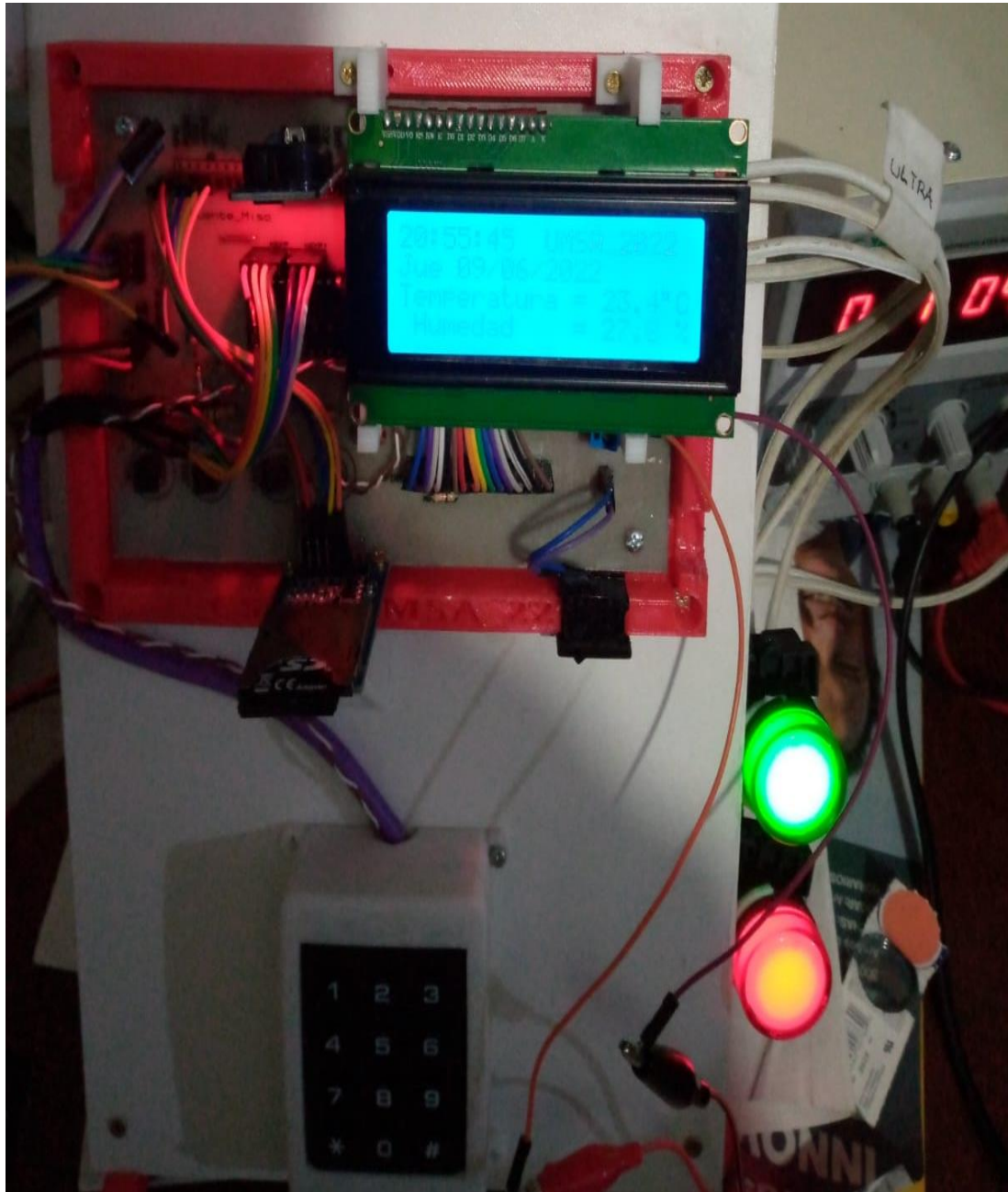
Anexo 5. Pruebas del funcionamiento del circuito a realizar



Anexo 5. Elaboración de la Placa PVC para el circuito.



Anexo 6. Elaboración de tablero y base para el circuito.



Anexo 7. Tablero de exposición para el Sistema de Registro de Asistencia y Apoyo en Operaciones Rutinarias mediante PIC18F4550 para la Empresa ALVAPLAS SRL.