

ESCUELA POLITÉCNICA NACIONAL

ESCUELA DE FORMACIÓN DE TECNÓLOGOS

“CONSTRUCCIÓN DE UN CIRCUITO DE CONTROL PARA TRES CARGAS MEDIANTE MENSAJES SMS (SERVICIO PARA MENSAJES CORTOS), EMITIDAS DESDE UN TELÉFONO CELULAR”

PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE TECNÓLOGO EN ELECTRÓNICA Y TELECOMUNICACIONES

ANTONIO KLEBER NAGUA SANDOVAL

antonionagua@hotmail.com

DIRECTOR: ING. CARLOS ROMO HERRERA

carlos.romo@epn.edu.ec

Quito, agosto, 2013

DECLARACIÓN

Yo, Antonio Kleber Nagua Sandoval, declaro bajo juramento que el presente trabajo desarrollado es de mi autoría, que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración, cedo mis derechos de propiedad intelectual correspondiente a este trabajo a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su reglamento y por la normativa institucional vigente.

.....

ANTONIO KLEBER NAGUA SANDOVAL

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Antonio Kleber Nagua Sandoval, bajo mi supervisión.

.....
Ing. Carlos Romo Herrera
DIRECTOR DE PROYECTO

DEDICATORIA

A mis padres, Mauro Filiberto Nagua Gonzales (†) y Rosa Celinda Sandoval Nagua, por todo el apoyo incondicional que me supieron dar, Dios sabrá retribuirles llenándoles de bendiciones.

A mi sobrina, Dayana Mishel Silva Nagua, por la ayuda espiritual y moral que me ha brindado durante el periodo de estudios hasta culminar mi carrera.

A mis hermanas y hermano que siempre estuvieron prestos a darme todo su apoyo.

Antonio Kleber Nagua Sandoval

AGRADECIMIENTO

Mi eterno agradecimiento:

A Dios, ser supremo que en ningún momento me ha abandonado.

A mi padre (†), Mauro F. Nagua G., que en tiempos buenos y malos siempre estuvo a mi lado, guiándome por el camino del bien.

A mi madre, Rosa C. Sandoval N., quien con sus oraciones ante Dios estuvo en todo momento pidiendo por mi bienestar y superación personal.

A todos mis maestros, quienes supieron impartir sus sabios conocimientos de manera incondicional.

A mi tutor, Sr. Ing. Carlos Romo quien puso todo su empeño para que este proyecto culmine con éxito.

Antonio Kleber Nagua Sandoval

ÍNDICE

CAPÍTULO 1.....	1
FUNDAMENTOS TEÓRICOS.....	1
1.1 OBJETIVO GENERAL.....	1
1.2 TELEFONÍA CELULAR O MÓVIL.....	2
1.2.1 Introducción.....	2
1.2.2 Breve historia de la telefonía celular	3
1.3 LAS GENERACIONES DE LA TELEFONÍA INALÁMBRICA.....	3
1.3.1 Primera generación (1G).....	3
1.3.2 Segunda generación (2G).....	4
1.3.3 Generación 2.5 G.....	4
1.3.4 Tercera generación (3G).....	4
1.4 TELÉFONO MÓVIL O CELULAR.....	5
1.4.1 Definición	5
1.4.2 Funcionamiento de un teléfono móvil.....	6
1.5 TECNOLOGÍAS DE ACCESO CELULAR.....	7
1.5.1 Tecnología FDMA o Acceso Múltiple por División de Frecuencia.....	8
1.5.2 Tecnología TDMA o Acceso Múltiple por División de Tiempo.....	8
1.5.3 Tecnología CDMA o Acceso Múltiple por División de Código.....	8
1.6 SISTEMA GLOBAL PARA COMUNICACIONES MÓVILES O GSM.....	9
1.6.1 Definición.....	9
1.6.2 Canal de radio GSM – TDMA.....	10
1.6.3 Movilidad GSM.....	10
1.6.4 Niveles de comunicación	11
1.6.5 Servicios GSM.....	11
1.6.5.1 Control de llamadas.....	11
1.6.5.2 Servicios suplementarios.....	11
1.6.5.3 Servicio de mensajes cortos.....	12
1.6.6 Características de seguridad	12
1.6.7 Arquitectura o Red GSM.....	12
1.6.7.1 Definición de Red GSM.....	13
1.6.7.2 Subsistemas de la Red GSM.....	13
1.6.7.3 Tarjeta SIM	14
1.6.8 Canales de la Red GSM	15

1.6.8.1 Canal de Control de Frecuencia (FCCH).....	15
1.6.8.2 Canal de Sincronización (SCH).....	15
1.6.8.3 Paginación y Canal de Acceso Concedido (PAGCH).....	16
1.6.8.4 Canal de Acceso Aleatorio (RACH).....	16
1.6.8.5 Canales de Difusión Celular (CBCH).....	16
1.6.8.6 Canal de Control Broadcast (BCCH).....	16
1.6.8.7 Canal de Control Asociado Rápido (FACCH).....	16
1.6.8.8 Canal de Tráfico (TCH).....	16
1.6.9 <u>Bandas de frecuencia GSM</u>	16
1.7 TELÉFONO CELULAR NOKIA 3220.....	17
1.7.1 Especificaciones técnicas	18
1.7.2 Puerto de comunicación del teléfono GSM (NOKIA 3220).....	18
1.8 SERVICIO DE MENSAJES CORTOS O SMS.....	19
1.8.1 Definición.....	19
1.8.2 Elementos de la red SMS.....	19
1.8.2.1 Entidad de Envío de Mensajes Cortos (SME, Short Messaging Entity).....	20
1.8.2.2 Centro de Servicio de Mensajes Cortos (SMSC, Short Message Service Center).....	20
1.8.2.3 Centro de Conmutación Móvil (MSC, Mobile Switching Center).....	20
1.8.2.4 SMS – IWMSC (Interworking).....	20
1.8.2.5 SMS – GMSC (Gateway).....	20
1.8.2.6 Sistema de Estación Base o BSS.....	21
1.8.3 Operaciones para el envío de mensajes SMS.....	21
1.8.4 Aplicaciones comunes del SMS.....	22
1.9 CABLE DE COMUNICACIÓN SERIAL USB CA-42.....	22
1.9.1 Definición.....	22
1.9.2 Características.....	23
1.9.3 Lista de compatibilidad.....	23
1.10 MÓDEM GSM.....	23
1.10.1 Modulación.....	24
1.10.1.1 Modulación Analógica.....	24
1.10.1.2 Modulación Digital.....	25
1.10.2 Demodulación	26
1.10.3 Definición de Módem GSM.....	26
1.10.4 Características más importantes de un Módem GSM.....	26

1.11 COMANDOS AT.....	27
1.11.1 Definición	27
1.11.2 Presentación de los comandos AT.....	27
1.11.3 Funciones de los comandos AT.....	28
1.11.4 Módem y comandos AT.....	28
1.12 COMUNICACIÓN SERIAL.....	29
1.12.1 Definición	29
1.12.2 Características de la comunicación serial.....	29
1.12.2.1 Transmisión modulada en amplitud.....	29
1.12.2.2 Estándares con formato marca/espacio.....	30
1.12.2.3 Consideraciones en la comunicación serial.....	31
1.12.2.4 Velocidad de transmisión	31
1.12.2.5 La base de reloj.....	32
1.12.2.6 Líneas o canales de comunicación.....	32
1.12.2.7 Modos de transmisión.....	33
1.13 EL MICROCONTROLADOR.....	40
1.13.1 Introducción.....	40
1.13.2 Definición.....	41
1.13.3 Estructura de un microcontrolador.....	42
1.13.4 Núcleo de un microcontrolador.....	43
1.13.5 Tipos de arquitecturas de microcontroladores.....	43
1.13.5.1 Arquitectura Von Neumann.....	43
1.13.5.2 Arquitectura Harvard.....	44
1.13.6 Procesador en detalle del microcontrolador.....	45
1.13.7 Características de un microcontrolador.....	45
1.13.7.1 Registros	45
1.13.7.2 Unidad de control.....	46
1.13.7.3 Unidad aritmética-lógica.....	46
1.13.7.4 Buses.....	47
1.13.7.5 Conjunto de instrucciones	47
1.13.7.6 Memoria	48
1.13.7.7 Interrupciones	50
1.13.7.8 Periféricos	51
1.14 EL MICROCONTROLADOR ATMEL AVR.....	54
1.14.1 Introducción.....	54
1.14.2 El microcontrolador ATMEGA8A.....	55

1.14.2.1	Características principales del microcontrolador ATMEGA8A.....	55
1.14.2.2	Descripción de los pines del ATMEGA8A.....	57
1.14.2.3	La Unidad Central de Procesos (CPU).....	58
1.14.2.4	El Registro de estados.....	59
1.14.2.5	Registros de propósito general.....	61
1.14.2.6	Registros X, Y, Z.....	62
1.14.2.7	Puntero de Pila.....	63
1.14.2.8	La arquitectura Harvard y RISC.....	63
1.14.2.9	Memoria de programa y memoria de datos.....	64
1.15.	RELÉ DE SRD-06VDC-SL-C.....	65
1.15.1	Definición.....	65
1.15.2	Distribución de pines del relé de SRD-06VDC-SL-C.....	66
1.16	PRINCIPALES TÓPICOS DEL LENGUAJE C++.....	67
1.16.1	Función delay.....	67
1.16.2	Operadores de comparación.....	67
1.16.3	Sentencias de decisión.....	68
1.16.3.1	Ejecución de las instrucciones.....	68
1.16.4	Sentencias de iteración	70
1.16.4.1	Sentencias For.....	70
1.16.4.2	Sentencia while.....	71
	CAPÍTULO 2.....	72
	PLANIFICACIÓN Y CONSTRUCCIÓN DEL HARDWARE.....	72
2.1	INTRODUCCIÓN.....	72
2.2	PLANIFICACIÓN Y CONSTRUCCIÓN DEL CIRCUITO.....	72
2.2.1	Etapa de alimentación	73
2.2.1.1	Fuente de alimentación continua.....	73
2.2.1.2	Proceso para obtener los 12, 5 y 3.8 voltios DC.....	73
2.2.1.3	Circuito de alimentación continua de 12, 5 y 3.8 voltios.....	74
2.2.2	Etapa del usuario remoto	76
2.2.2.1	Monitoreo de mensajes SMS.....	76
2.2.3	Etapa de la red telefónica conmutada	77
2.2.4	Etapa del teléfono celular receptor.....	78
2.2.4.1	Descripción del cable de datos USB CA-42.....	78
2.2.4.2	Descripción del puerto de comunicación del teléfono celular receptor.....	78

2.2.4.3	Conexión y configuración del cable de datos USB CA-42 entre el teléfono celular receptor y el microcontrolador.....	79
2.2.5	Microcontrolador Atmega8A.....	81
2.2.5.1	Circuito de Reset.....	81
2.2.5.2	Secuencia en el hardware para el envío de mensaje.....	82
2.2.5.3	Diseño del software utilizado en el microcontrolador Atmega8A.....	82
2.2.5.4	Funciones entre teléfono celular receptor y microcontrolador.....	87
2.2.5.5	Principales funciones del microcontrolador Atmega8A.....	88
2.2.5.6	Puertos y pines del microcontrolador Atmega8A utilizados.....	88
2.2.6	Etapas de optoacoplamiento.....	89
2.2.6.1	Circuito para la activación de tres leds.....	90
2.2.6.2	Circuito para la activación/desactivación de las cargas.....	91
2.2.7	Etapas de activación.....	93
2.2.7.1	Activación del foco.....	94
2.2.7.2	Activación de la alarma.....	94
2.2.7.3	Activación del interruptor.....	94
2.2.8	Circuito general del proyecto.....	95
	CAPÍTULO 3.....	96
	CONCLUSIONES Y RECOMENDACIONES.....	96
3.1	CONCLUSIONES.....	96
3.2	RECOMENDACIONES	98
3.3	BIBLIOGRAFÍA.....	99
	TABLAS	
	Tabla 1.1: Banda de frecuencias GSM.....	17
	Tabla 1.2: Especificaciones técnicas del teléfono celular Nokia 3220.....	18
	Tabla 1.3: Bits de paridad par.....	38
	Tabla 1.4: Bits de paridad impar.....	38
	Tabla 1.6: Nombre y sintaxis de algunos operadores en C++.....	68
	Tabla 2.1: Funciones que cumplen los diferentes mensajes SMS.....	76
	Tabla 2.2: Nombre y función de los pines utilizados en el microcontrolador.....	89

FIGURAS

Figura 1.1: Elementos de la telefonía móvil.....	5
Figura 1.2: Antena de telefonía Móvil.....	6
Figura 1.3: Conjunto de celdas o células contiguas.....	7
Figura 1.4: Formas de dividir la frecuencia, según los diferentes estándares.....	9
Figura 1.5: Esquema general de una Red GSM.....	12
Figura 1.6: Puerto serial del teléfono Nokia 3220.....	18
Figura 1.7: Estructura básica de la red SMS.....	19
Figura 1.8: Envío del mensaje.....	20
Figura 1.9: Recepción del mensaje.....	21
Figura 1.10: Cable de datos USB CA-42.....	22
Figura 1.11: Ejemplo de modulación en AM y FM.....	24
Figura 1.12: Formato serie marca/espacio.....	30
Figura 1.13: Comunicación asincrónica.....	34
Figura 1.14: Formato básico de transmisión asincrónica.....	34
Figura 1.15: Comunicación sincrónica.....	36
Figura 1.16: Formato básico de transmisión sincrónica.....	36
Figura 1.17: Generadores y detectores de paridad.....	37
Figura 1.18: Primer microprocesador, INTEL 4004.....	41
Figura 1.19: Microcontrolador <u>Motorola</u> 68HC11 y chips de soporte.....	41
Figura 1.20: Estructura básica de un microcontrolador.....	42
Figura 1.21: Arquitectura Von Neumann.....	43
Figura 1.22: Arquitectura Harvard.....	44
Figura 1.23: Diagrama de Bloques del Atmega8A.....	56
Figura 1.24: Pines del microcontrolador Atmega8A	57
Figura 1.25: Arquitectura AVR.....	59
Figura 1.26: Registro de estados.....	60
Figura 1.27: Registros de propósito general.....	61

Figura 1.28: Los Registros X, Y, Z.....	62
Figura 1.29: Puntero de pila.....	63
Figura 1.30: Memoria de programas.....	64
Figura 1.31: Mapa de la memoria de datos.....	65
Figura 1.32: Relé (SRD-06VDC-SL-C)	66
Figura 1.33: Distribución de pines del relé de SRD-06VDC-SL-C.....	67
Figura 2.1: Diagrama de bloques del hardware del circuito.....	73
Figura 2.2: Circuito de 12, 5 y 3.8 voltios DC.....	74
Figura 2.3: Esquema de una red telefónica conmutada.....	77
Figura 2.4: Puerto de comunicación del Nokia 3220.....	79
Figura 2.5: Conexión entre el teléfono NOKIA 3220 y microcontrolador Atmega8A con el cable de datos USB CA-42.....	80
Figura 2.6: Circuito reset.....	81
Figura 2.7: Circuito del led indicador de una comunicación en proceso.....	82
Figura 2.8: Circuito para la activación de tres leds.....	90
Figura 2.9: Circuito para la activación/desactivación de las cargas.....	93
Figura 2.10: Circuito general del proyecto.....	95

CAPÍTULO 1

FUNDAMENTOS TEÓRICOS

1.1 OBJETIVO GENERAL

La elaboración del presente Proyecto de Titulación tiene como objetivo general construir un circuito de control para tres cargas mediante mensajes SMS (Servicio para Mensajes Cortos), emitidas desde un teléfono celular.

El criterio principal de este circuito consiste en que al enviar un mensaje SMS desde un teléfono celular (emisor) debe ser receptado por otro teléfono celular (receptor) para posteriormente, a través de un microcontrolador (Atmega8A), descargar el mensaje SMS y ejecutarlo; esto consiste en activar/desactivar un relé y su(s) respectiva(s) carga(s) que en nuestro caso son un foco, una alarma y un interruptor eléctrico. Si el mensaje SMS no corresponde a ninguna orden que el microcontrolador debe realizar, dicho mensaje será ignorado por el microcontrolador.

Cabe anotar que el teléfono celular receptor, es un Nokia 3220, el mismo que tiene como característica principal el funcionamiento mediante tecnología GSM (Sistema Global para Comunicaciones Móviles), característica necesaria para poder llevar a cabo esta aplicación, como se describirá más adelante. Es posible utilizar otro teléfono receptor, siempre y cuando su funcionamiento posea tecnología GSM.

Ya en la práctica, se puede aplicar a diversas necesidades cotidianas como controlar las seguridades del hogar, controlar las seguridades de un automóvil, manipular las instalaciones eléctricas en alguna empresa, dar un aviso de alarma en el caso un intruso quiera forzar las seguridades de su hogar, etc.

En este capítulo se hará un análisis de la parte teórica de los temas relacionados con el proyecto, ya que esto nos permitirá tener un mejor entendimiento dentro del proceso de elaboración y descripción de este proyecto.

1.2 TELEFONÍA CELULAR O MÓVIL¹

1.2.1 Introducción

Las tecnologías inalámbricas y particularmente la telefonía celular han tenido mucho auge y desarrollo en estos últimos años.

Desde sus inicios a finales de los 70 ha revolucionado enormemente las actividades que realizamos diariamente. Los teléfonos celulares se han convertido en una herramienta primordial para la gente común y de negocios, satisfaciendo así muchas de las necesidades de comunicación que diariamente tenemos. Además, cada vez podemos apreciar teléfonos móviles con mejores características, como son: su tamaño y peso, el desarrollo de baterías más pequeñas y de mayor duración, pantallas más nítidas y de colores, la incorporación de software más amigable, hacen del teléfono móvil un elemento muy apreciado en la vida moderna.

A pesar de que la telefonía móvil o celular fue concebida estrictamente para la voz, el avance de la tecnología ha hecho que estos aparatos incorporen funciones que hasta no hace mucho parecían futuristas, como juegos, reproducción de música MP3 y otros formatos, como correo electrónico, servicio de mensajes cortos (SMS), agenda electrónica PDA, fotografía digital y video digital, videollamada, navegación por Internet, GPS, y hasta Televisión digital; sin embargo, la telefonía inalámbrica del mañana hará posible aplicaciones que requieran un mayor consumo de ancho de banda.

¹ http://es.wikipedia.org/wiki/Telefon%C3%ADa_m%C3%B3vil

1.2.2 Breve historia de la telefonía celular

Martin Cooper fue el pionero en esta tecnología, a él se le considera como "el padre de la telefonía celular" al introducir el primer radioteléfono en Estados Unidos, en 1973, mientras trabajaba para Motorola pero no fue hasta 1979 cuando aparecieron los primeros sistemas comerciales en Tokio, Japón.

En 1981, los países nórdicos introdujeron un sistema celular similar al Sistema Telefónico Móvil Avanzado (AMPS, por sus siglas en inglés). Por otro lado, en Estados Unidos, gracias a que la entidad reguladora de ese país adoptó reglas para la creación de un servicio comercial de telefonía celular, en 1983 se puso en operación el primer sistema comercial en la ciudad de Chicago.

En varios países se diseminó la telefonía celular como una alternativa a la telefonía convencional inalámbrica, la misma que tuvo gran aceptación, por lo que a los pocos años de implantarse se empezó a saturar el servicio; en ese sentido, hubo la necesidad de desarrollar e implantar otras formas de acceso múltiple al canal y transformar los sistemas analógicos a digitales, con el objeto de darle cabida a más usuarios.

1.3 LAS GENERACIONES DE LA TELEFONÍA INALÁMBRICA²

1.3.1 Primera generación (1G): La primera generación (1G) de la telefonía móvil hizo su aparición en 1979 y se caracterizó por ser analógica y estrictamente para voz. La calidad de los enlaces era muy baja, tenían baja velocidad (2400 bauds). En cuanto a la transferencia entre celdas, era muy imprecisa ya que contaban con una baja capacidad basadas en Acceso Múltiple por División de Frecuencia (FDMA, por sus siglas en inglés); además, la seguridad no existía. La tecnología predominante de esta generación es AMPS.

²[http://evdoplus.blogspot.com/2007/04/las-generaciones de la telefonía .html](http://evdoplus.blogspot.com/2007/04/las-generaciones-de-la-telefon%C3%ADa.html)

1.3.2 Segunda generación (2G): La segunda generación (2G) arribó hasta 1990 y a diferencia de la primera se caracterizó por ser digital; utiliza protocolos de codificación más sofisticados y se emplea en los sistemas de telefonía celular actuales. Las tecnologías predominantes en esta generación, son: Sistema Global para Comunicaciones Móviles o GSM ; IS-136 (conocido también como TIA/EIA136 o ANSI-136); Acceso Múltiple por División de Código o CDMA y PDC (Personal Digital Communications), éste último utilizado en Japón.

Los protocolos empleados en los sistemas 2G soportan velocidades de información más altas por voz, pero limitados en comunicación de datos. Se pueden ofrecer servicios auxiliares como datos, fax y Servicio de Mensajes Cortos o SMS. La mayoría de los protocolos de 2G ofrecen diferentes niveles de encriptación. En Estados Unidos y otros países se le conoce a la segunda generación como PCS (Personal Communication Services).

1.3.3 Generación 2.5 G: La generación 2.5G es más rápida, y más económica para actualizar a 3G. La generación 2.5G ofrece características extendidas, ya que cuenta con más capacidades adicionales que los sistemas 2G, como: GPRS (General Packet Radio System), HSCSD (High Speed Circuit Switched), EDGE (Enhanced Data Rates for Global Evolution), IS-136B e IS-95Bm, entre otros.

1.3.4 Tercera generación (3G): La tercera generación (3G) se caracteriza por contener a la convergencia de voz y datos con acceso inalámbrico a Internet; en otras palabras, es apta para aplicaciones multimedia y altas transmisiones de datos.

Los protocolos empleados en los sistemas 3G soportan altas velocidades de información (384 Kbps) y están enfocados para aplicaciones más allá de la voz como audio (MP3), video en movimiento, videoconferencia y acceso rápido a Internet.

Así mismo, en un futuro próximo los sistemas 3G alcanzarán velocidades de hasta 384 kbps, permitiendo una movilidad total a usuarios, viajando a 120 kilómetros por hora en ambientes exteriores. También alcanzará una velocidad máxima de 2 Mbps, permitiendo una movilidad limitada a usuarios.

1.4 TELÉFONO MÓVIL O CELULAR³

1.4.1 Definición

El **teléfono móvil** es un dispositivo inalámbrico electrónico que permite tener acceso a la red de telefonía celular o móvil. Se denomina celular en la mayoría de países latinoamericanos debido a que el servicio funciona mediante una red de celdas, donde cada antena repetidora de señal es una célula, si bien también existen redes telefónicas móviles satelitales. Su principal característica es su portabilidad, que permite comunicarse desde casi cualquier lugar. La principal función es la comunicación de voz, como el teléfono convencional.

En la figura 1.1, se puede apreciar dos elementos básicos que conforman la telefonía móvil: una red de comunicaciones (o red de telefonía móvil) y los terminales (o teléfonos móviles) que permiten el acceso a dicha red.



a) Antena



b) Teléfono móvil o celular

Figura 1.1: Elementos de la telefonía móvil

³ http://es.wikipedia.org/wiki/Telefon%C3%ADa_m%C3%B3vil

1.4.2 Funcionamiento de un teléfono móvil

La comunicación telefónica es posible gracias a la interconexión entre centrales móviles y públicas; según las bandas o frecuencias en las que opera el móvil, podrá funcionar en cualquier parte del mundo. La telefonía móvil consiste en la combinación de una red de estaciones transmisoras-receptoras de radio (repetidores, estaciones base o BTS) y una serie de centrales telefónicas de conmutación de 1er y 5º nivel (Centro de Conmutación de Servicios Móviles y Estación Transceptora de Base, respectivamente) que posibilita la comunicación entre terminales telefónicos portátiles (teléfonos móviles) o entre terminales portátiles y teléfonos de la red fija tradicional.

En su operación el teléfono móvil establece comunicación con una estación base, y a medida que se traslada, los sistemas computacionales que administran la red van cambiando la llamada a la siguiente estación base, en forma transparente para el usuario, es por eso que se dice que las estaciones base forman una red de celdas, cual panal de abeja, sirviendo cada estación base a los equipos móviles que se encuentran en su celda.

En la figura 1.2, se presenta una antena de telefonía móvil que es una estación base, de instalación fija, conectada con los teléfonos móviles mediante ondas electromagnéticas de radiofrecuencia, así mismo dichas antenas se comunican con la central de su propia red.



Figura 1.2: Antena de telefonía Móvil

La gran idea del sistema celular es la división de la ciudad en pequeñas celdas; esta idea permite la re-utilización de frecuencias a través de la ciudad, con lo que miles de personas pueden usar los teléfonos al mismo tiempo. Cada celda tiene una estación base que consiste de una torre y un pequeño edificio que contiene el equipo de radio.

En la figura 1.3, se puede apreciar un arreglo de celdas, cada una de ellas contiene una estación base y el equipo de radio.

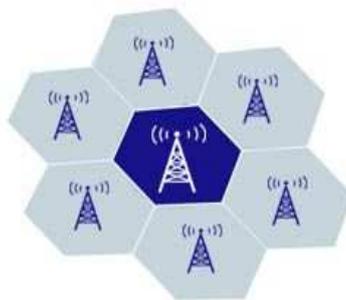


Figura 1.3: Conjunto de celdas o células contiguas

En un sistema análogo, cada celda utiliza un séptimo de los canales de voz disponibles; esto es, una celda más las seis celdas que la rodean en un arreglo hexagonal, cada una utilizando un séptimo de los canales disponibles para que cada celda tenga un grupo único de frecuencias y no haya colisiones.

En otras palabras, en cualquier celda pueden hablar 56 personas en sus teléfonos celulares al mismo tiempo. Con la transmisión digital, el número de canales disponibles aumenta; por ejemplo, el sistema digital TDMA (Acceso Múltiple por División de Tiempo) puede acarrear el triple de llamadas en cada celda, alrededor de 168 canales disponibles simultáneamente.

1.5 TECNOLOGÍAS DE ACCESO CELULAR⁴

Las tecnologías utilizadas actualmente para la transmisión de información en las redes son denominadas de Acceso Múltiple (MA, por sus siglas en inglés),

⁴ <http://www.monografias.com/trabajos14/celularhist/celularhist.shtml>

debido a que más de un usuario puede utilizar o accederá cada una de las celdas de información. Actualmente existen tres diferentes tecnologías, que difieren en los métodos de acceso a las celdas, estas son las siguientes.

1.5.1 Tecnología FDMA o Acceso Múltiple por División de Frecuencia:

Accede a las celdas dependiendo de las frecuencias. Básicamente, separa el espectro en distintos canales de voz, al dividir el ancho de banda en varios canales uniformemente según las frecuencias de transmisión. Los usuarios comparten el canal de comunicación, pero cada uno utiliza uno de los diferentes subcanales particionados por la frecuencia. Mayormente es utilizada para las transmisiones analógicas, aún cuando es capaz de transmitir información digital (no recomendada).

1.5.2 Tecnología TDMA o Acceso Múltiple por División de Tiempo:

Divide el canal de transmisión en particiones de tiempo. Comprime las conversaciones digitales y luego las envía utilizando la señal de radio por un período de tiempo. En este caso, distintos usuarios comparten el mismo canal de frecuencia, pero lo utilizan en diferentes intervalos de tiempo. La compresión de la señal de voz es posible debido a que la información digital puede ser reducida de tamaño por ser información binaria (unos y ceros); debido a la compresión de la información digital, esta tecnología permite tres veces la capacidad de un sistema analógico utilizando la misma cantidad de canales.

1.5.3 Tecnología CDMA o Acceso Múltiple por División de Código:

Esta tecnología, luego de digitalizar la información la transmite a través de todo el ancho de banda del que se dispone, a diferencia de TDMA y FDMA. Las llamadas se sobreponen en el canal de transmisión, diferenciadas por un código de secuencia único; esto permite que los usuarios compartan el canal y la frecuencia. Como es un método adecuado para la transmisión de información encriptada, se comenzó a utilizar en el área militar. Esta tecnología permite comprimir de 8 a 10 llamadas digitales para que ocupen lo mismo que ocupa una llamada analógica.

En la figura 1.4, se muestra un gráfico comparativo del funcionamiento de las mencionadas tecnologías.

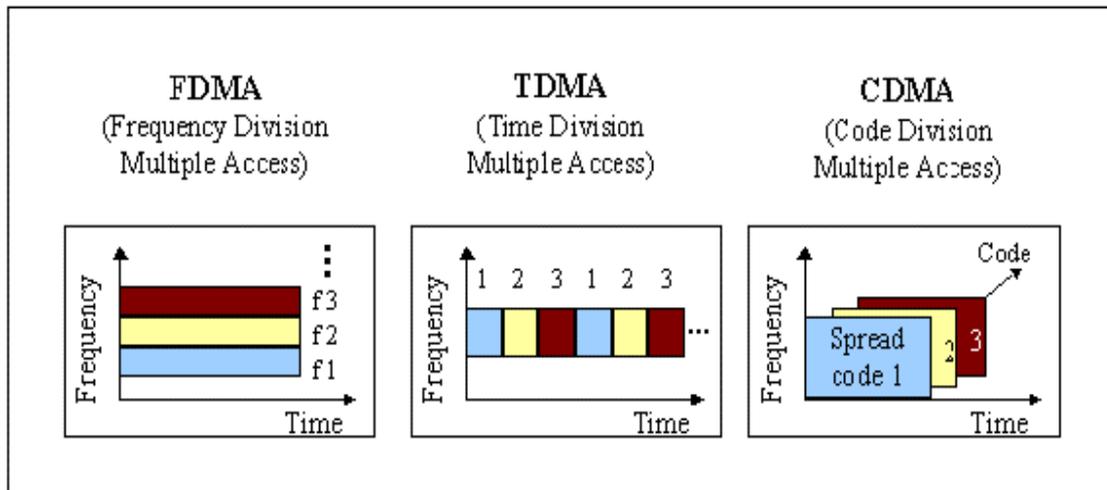


Figura 1.4: Formas de dividir la frecuencia, según los diferentes estándares

1.6 SISTEMA GLOBAL PARA COMUNICACIONES MÓVILES⁵

1.6.1 Definición

El Sistema Global para Comunicaciones Móviles (GSM, por sus siglas en inglés), es un sistema de identificación usado por algunos teléfonos móviles, que determina cómo tienen que establecer una conexión entre ese terminal y otro a través de una red de telecomunicaciones.

En **GSM** se utiliza un algoritmo de codificación de voz, modulación digital GSMK (Gaussian Minimum ShiftKeying), lento salto de frecuencia y arquitectura de ranuras de tiempo TDMA, lo que hace que la tecnología GSM ofrezca seguridad y confiabilidad a los usuarios del servicio. GSM ha sido un sistema diseñado con los niveles rigurosos de la seguridad incorporada y está en constante evolución.

⁵ http://es.wikipedia.org/wiki/Sistema_global_para_las_comunicaciones_m%C

1.6.2 Canal de radio GSM – TDMA

El estándar GSM especifica las bandas de frecuencia de 890 MHz a 915 MHz para la banda del enlace saliente y de 935 MHz a 960 MHz para la banda del enlace entrante; cada banda se divide en canales de 200 KHz.

Otras características del interface de canal de radio son:

- La alineación de tiempo adaptativa: Permite a la estación móvil corregir su ranura de tiempo de transmisión para retardos de propagación.
- La modulación GMSK: Proporciona eficiencia espectral e interferencia fuera de banda baja requerida en el sistema GSM.
- La transmisión y recepción discontinua: Se refiere a la caída de potencia de la estación móvil durante periodos de inactividad y sirve al doble propósito de reducir la interferencia entre canales y aumentar el tiempo de vida de la batería de la unidad portable.
- El salto de frecuencias lento: El salto de frecuencias lento es una característica adicional del interfaz de canal de radio GSM que ayuda a contrarrestar efectos de desvanecimiento Rayleigh y de la interferencia entre canales.

1.6.3 Movilidad GSM

Una de las características principales utilizadas en todas las redes GSM y satélite, es la capacidad para soportar el "roaming" (poder cambiar de un país a otro...) de los usuarios. Utilizando la red de señalización de control, los Centros de Estación Móvil (MSCs) interactúan para localizar y conectar a los usuarios en toda la red. Los Registros de Localización Base (HLR) se encuentran incluidos en las Bases de Datos del Centro de Estación Móvil (MSC) para ayudar a la función de determinar como y si las conexiones deben realizarse para los usuarios itinerantes (usuarios Roaming). Cada usuario de una estación móvil GSM tiene asignado un HLR que se utiliza para contener la localización del usuario y los servicios del abonado en cuestión. Un registro

separado, denominado Registro de Localización de Visitante (VLR) se utiliza para seguir la pista de localización de un usuario. Cuando el usuario cruza el área cubierta por el HLR, la estación móvil notificará una nueva VLR de su paradero actual. El VLR a su vez utiliza la red de control para señalar la HLR de la nueva localización de la estación móvil. Utilizando esta información, las llamadas terminadas en el móvil se pueden encaminar al usuario utilizando la información de localización contenida en el HLR del usuario.

1.6.4 Niveles de comunicación

GSM necesita la utilización de varios protocolos para poder controlar las llamadas, transferir información y proporcionar gestión global del sistema. Desde la MS (Estación Móvil) existen los siguientes niveles para la comunicación:

- Interface RF (Radio Frequency) a la BTS.
- Nivel de gestión de Recursos de Radio (RR) al BSC.
- Gestión de la movilidad (MM).
- Gestión de las comunicaciones (CM) al registro VLR del MSC.

1.6.5 Servicios GSM

El nivel de gestión de comunicaciones proporciona 3 clases de servicios primarios; estos son:

1.6.5.1 Control de llamadas: Los servicios de control de llamadas son responsables del encaminamiento de llamadas; esto es, determinar cual es el responsable de los costos de la llamada y la organización que tiene que recibir el pago.

1.6.5.2 Servicios suplementarios: Los servicios suplementarios son el reenvío de llamadas, llamada en espera, aviso de cargo, passwords, entre otras.

1.6.5.3 Servicio de mensajes cortos: El nivel de gestión de las comunicaciones incluye servicios para manipular servicios de mensajes cortos, que son más eficientemente manipulados utilizando transferencias orientadas a paquetes que las conexiones tradicionales de conmutación de circuitos soportadas por el sistema GSM principal.

1.6.6 Características de seguridad

La seguridad en GSM consta de los siguientes aspectos:

- Autenticación de la Identidad del Abonado
- Confidencialidad de la Identidad del Abonado
- Confidencialidad de los Datos de Señalización
- Confidencialidad de los Datos del Usuario

1.6.7 Arquitectura o Red GSM

La figura 1.5, representa la estructura de una red GSM con sus principales elementos.

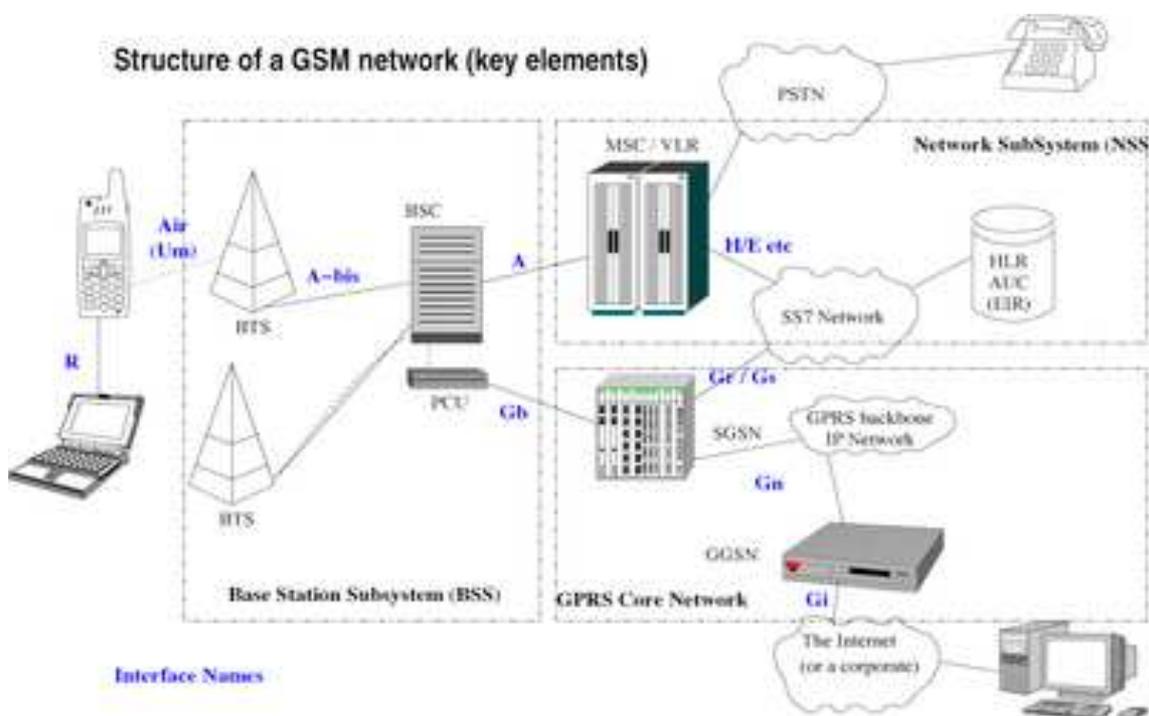


Figura 1.5: Esquema general de una Red GSM.

1.6.7.1 Definición de Red GSM

Se define la Red del Sistema Global de Telefonía GSM como aquel servicio portador constituido por todos los medios de transmisión y conmutación necesarios que permiten enlazar a voluntad dos equipos terminales móviles, mediante un canal digital que se establece específicamente para la comunicación y que desaparece una vez que se ha completado la misma.

1.6.7.2 Subsistemas de la Red GSM⁶

La arquitectura GSM consta de varios Subsistemas:

1.6.7.2.1 Estación Móvil (MS): Son los teléfonos digitales que tienen un dispositivo denominado SIM (Módulo de Identidad del Suscriptor) que es básicamente la típica tarjeta que proporciona la información de servicios e identificación en la Red.

1.6.7.2.2 Subsistema de Estación (BSS): Es una colección de dispositivos que soportan el interface de radio de redes de conmutación. Los principales componentes del BSS son:

1.6.7.2.2.1 Estación Transceptora de Base (BTS): Consta de los módems de radio y el equipo de antenas.

1.6.7.2.2.2 El Controlador de Estaciones Base (BSC): Denominado también Handover, gestiona las operaciones de radio de varias BTS y conecta a un único NSS.

1.6.7.2.3 Subsistema de Conmutación y Red (NSS): Proporciona la conmutación entre el subsistema GSM y las redes externas (PSTN, PDN) junto

⁶ http://www.gsmspain.com/info_tecnica/sms/pagina3.php

con las bases de datos utilizadas para la gestión adicional de la movilidad y de los abonados; sus componentes son:

1.6.7.2.3.1 Centro de Conmutación de Servicios Móviles (MSC): Se encarga de iniciar, terminar y canalizar las llamadas a través del BSC y BS correspondientes al abonado llamado.

1.6.7.2.3.2 Registro de Ubicación Base (HLR): Es una base de datos que almacena la posición del usuario dentro de la red, si está conectado o no y las características de su abono (servicios que puede y no puede usar, tipo de terminal, etc.).

1.6.7.2.3.3 Registro de Ubicación de Visitante (VLR): Es una base de datos más volátil que almacena, para el área cubierta por un MSC, los identificativos, permisos, tipos de abono y localizaciones en la red de todos los usuarios activos en ese momento y en ese tramo de la red.

1.6.7.2.4 El Centro de Autenticación del Usuario (AUC): Se encarga del cifrado de las señales y de la identificación de usuarios dentro del sistema.

1.6.7.2.5 El Registro de Identificación de Equipo (EIR): Guarda listas de permiso de acceso al terminal, al que identifica unívocamente mediante su número de serie o IMEI, para evitar que los terminales robados y denunciados puedan usar la red.

1.6.7.2.6 Subsistema de Operaciones (OSS): Responsable del mantenimiento y operación de la Red, de la gestión de los equipos móviles y de la gestión y cobro de cuota.

1.6.7.3 Tarjeta SIM

Una de las características principales del estándar GSM es el Módulo de Identidad del Suscriptor, conocida comúnmente como tarjeta SIM (por sus

siglas en inglés), la misma que es una tarjeta inteligente desmontable que contiene la información de suscripción del usuario, parámetros de red y directorio telefónico; esto permite al usuario mantener su información después de cambiar su teléfono. Una tarjeta SIM contiene la siguiente información:

- El número telefónico del abonado (MSISDN).
- El número internacional de abonado (IMSI, Identificación internacional de abonados móviles).
- El estado de la tarjeta SIM.
- El código de servicio (operador).
- La clave de autenticación.
- El PIN (Código de identificación personal).
- El PUK (Código personal de desbloqueo).

1.6.8 Canales de la Red GSM⁷

Para establecer y mantener las comunicaciones entre las terminales móviles y las estaciones bases (BS) de la red, GSM utiliza un sistema TDMA para cada una de las frecuencias de que dispone. La comunicación en una determinada frecuencia se realiza a través de tramas temporales de 4,615 ms, divididas en 8 slots cada una. En esos slots se alojan los canales lógicos de GSM, que agrupan la información a transmitir entre la estación base y el móvil de la siguiente manera:

1.6.8.1 Canal de Control de Frecuencia (FCCH): Es la Información de frecuencias. Comunica al móvil (desde la BS) la frecuencia portadora de la BS.

1.6.8.2 Canal de Sincronización (SCH): Sigue a la ráfaga FCCH, proporciona una referencia para todas las ranuras de una frecuencia dada.

⁷<http://www.monografias.com/trabajos75/tecnologias-gsm-cdma-dma-gprs/tecnologias-gsm-cdma-tdma-gprs2.shtml/>

1.6.8.3 Paginación y Canal de Acceso Concedido (PAGCH): Transmisión de información de paginación que se pide en el establecimiento de una llamada a una estación móvil (MS).

1.6.8.4 Canal de Acceso Aleatorio (RACH): Canal no limitado utilizado por la MS para pedir conexiones desde la red terrestre.

1.6.8.5 Canales de Difusión Celular (CBCH): Transmisión no frecuente de difusiones.

1.6.8.6 Canal de Control Broadcast (BCCH): Información de estado de acceso a la MS.

1.6.8.7 Canal de Control Asociado Rápido (FACCH): Control de los "Handovers" (Paso de un usuario móvil de una célula a otra).

1.6.8.8 Canal de Tráfico (TCH): Albergan las llamadas en proceso que soporta la estación base. Dentro de éste canal se considera otros que son de importancia:

1.6.8.8.1 Canal de Tráfico full-rate(TCH/F): Este canal transporta información a una velocidad de 22,8 Kbps.

1.6.8.8.2 Canal de Tráfico half-rate (TCH/H): Este canal transporta información a una velocidad de 11,4Kbps.

1.6.9 Bandas de frecuencia GSM⁸

El interfaz de radio GSM se ha implementado en diferentes bandas de frecuencia.

⁸ http://es.wikipedia.org/wiki/Bandas_de_frecuencia_GSM

La tabla 1.1, muestra las bandas de frecuencia utilizadas por GSM con sus respectivos nombres, canales, enlace ascendente (uplink), enlace descendente (downlink), y los países en los canales son utilizadas.

Tabla 1.1: Bandas de frecuencias GSM

Banda	Nombre	Canales	Uplink (MHz)	Downlink (MHz)	Notas
GSM 850	GSM 850	128 - 251	824,0 - 849,0	869,0 - 894,0	Usada en los EE.UU. Sudamérica y Asia.
GSM 900	P-GSM 900	0-124	890,0 - 915,0	935,0 - 960,0	La banda con que nació GSM en Europa y la más extendida.
	E-GSM 900	974 - 1023	880,0 - 890,0	925,0 - 935,0	<i>E-GSM</i> , extensión de GSM 900
	R-GSM 900	n/a	876,0 - 880,0	921,0 - 925,0	<i>GSM ferroviario</i> (GSM-R).
GSM 1800	GSM 1800	512 - 885	1710,0 - 1785,0	1805,0 - 1880,0	
GSM 1900	GSM 1900	512 - 810	1850,0 - 1910,0	1930,0 - 1990,0	Usada en Norteamérica.

1.7 TELÉFONO CELULAR NOKIA 3220⁹

Existen algunos teléfonos celulares que poseen tecnología GSM, como se podrá apreciar más adelante en el listado de compatibilidad, de los cuales se ha seleccionado el Nokia 3220 para la elaboración de nuestro proyecto, por ser económico y fácil de encontrarlo en el mercado.

⁹ http://es.wikipedia.org/wiki/Nokia_3220

1.7.1 Especificaciones técnicas más importantes:

En la tabla 1.2, se resume las especificaciones técnicas más importantes que posee el teléfono celular en mención.

Tabla 1.2: Especificaciones técnicas

Marca	Nokia.
Modelo	3220.
Tecnología	GSM.
Redes	850 / 1800 / 1900.
Tipo de pantalla	Color (65536 colores).
Resolución de pantalla	128 x 128 pixeles.
Tiempo de conversación	Hasta 3 horas.
Tiempo de espera	Hasta 350 horas.
Conectividad	Cables USB y WAP, PC Suite.
Características de los mensajes	SMS, Texto predictivo, MMS.

1.7.2 Puerto de comunicación serial del teléfono NOKIA 3220¹⁰

El puerto de comunicación del teléfono GSM (Nokia 3220) posee 14 pines, cada uno de ellos cumplen con una función específica.

La figura 1.6, representa el puerto mencionado; los pines utilizados en la elaboración del proyecto serán descritos de manera detallada en el capítulo 2.



Figura 1.6: Puerto serial del teléfono Nokia 3220

¹⁰ <http://dspace.espace.edu.ec/bitstream/123456789/331/1/38T00174.pdf>

1.8 SERVICIO DE MENSAJES CORTOS O SMS¹¹

1.8.1 Definición

El Servicio de Mensajes Cortos (SMS, por sus siglas en inglés) disponible en redes digitales GSM, es una tecnología que permite el envío y recepción de mensajes cortos (conocidos también como mensajes de texto) entre teléfonos celulares, teléfonos fijos y otros dispositivos de mano. Un mensaje SMS es una cadena alfanumérica de hasta 160 caracteres; estos mensajes viajan sobre un canal dedicado a señalización independiente de los de tráfico.

SMS fue diseñado originariamente como parte del estándar de telefonía móvil digital GSM, pero en la actualidad está disponible en una amplia variedad de redes, incluyendo las redes 3G.

1.8.2 Elementos de la red SMS

En la figura 1.7, se presenta los elementos que conforman la red SMS, los mismos que están ubicados conservando un orden secuencial para que cada uno de ellos cumplan con su función de manera apropiada.

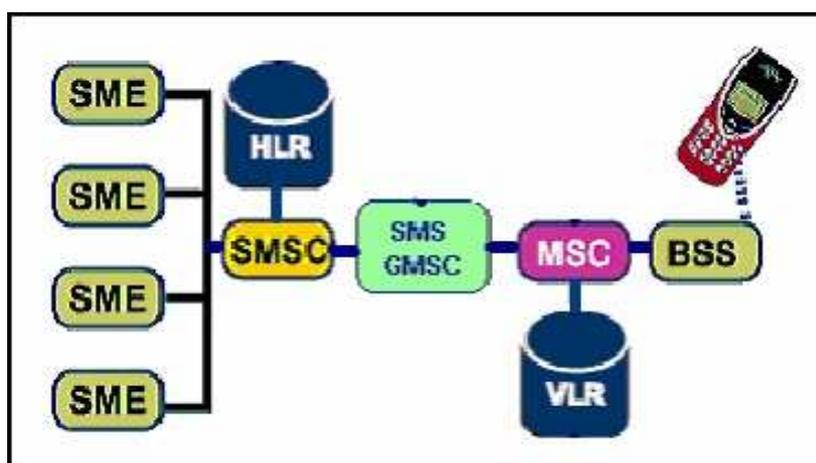


Figura 1.7: Estructura básica de la red SMS

¹¹ <http://ingeniatic.euitt.upm.es/index.php/tecnologias/item/600-sms-servicio-de-mensajes-cortos>

1.8.2.1 Entidad de Envío de Mensajes Cortos (SME, Short Messaging Entity): Entidad que puede enviar o recibir mensajes cortos, pudiendo estar localizada en la red fija, una estación móvil, u otro centro de servicio.

1.8.2.2 Centro de Servicio de Mensajes Cortos (SMSC, Short Message Service Center): Es el responsable de la transmisión y almacenamiento de un mensaje corto, entre el SME y una estación móvil.

1.8.2.3 Centro de Conmutación Móvil (MSC, Mobile Switching Center): Lleva a cabo funciones de conmutación del sistema y el control de llamadas a y desde otro teléfono y sistema de datos.

1.8.2.4 SMS – IWMSC (Interworking): Es originado por el móvil y va hacia el SMSC (servicio SMS-MO).

En la figura 1.8, se presenta la secuencia para realizar el envío de un mensaje.

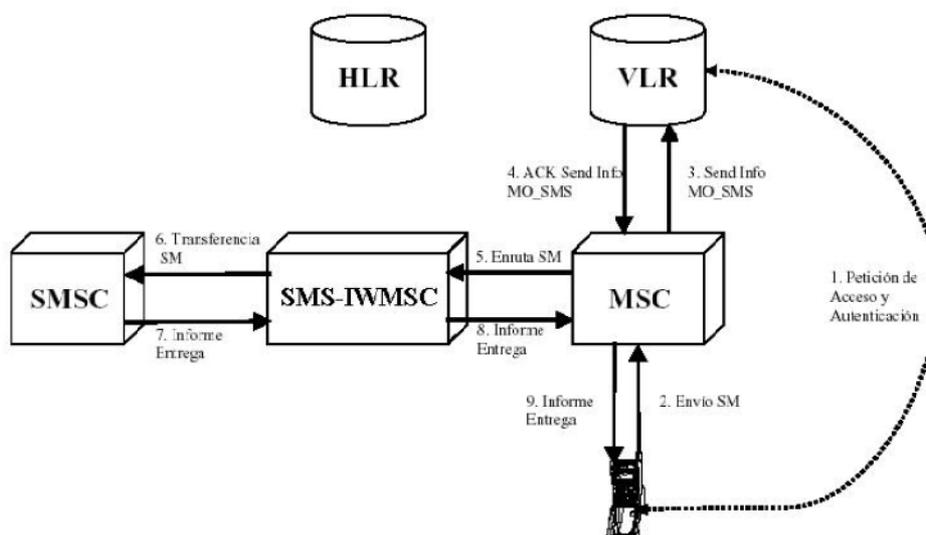


Figura 1.8: Envío del mensaje

1.8.2.5 SMS – GMSC (Gateway): Establecido entre MSC y la red GSM, es el que permite que el SMS llegue a su destino (servicio SMS-MT).

En la figura 1.9, se presenta la secuencia como se realiza la recepción de un mensaje.

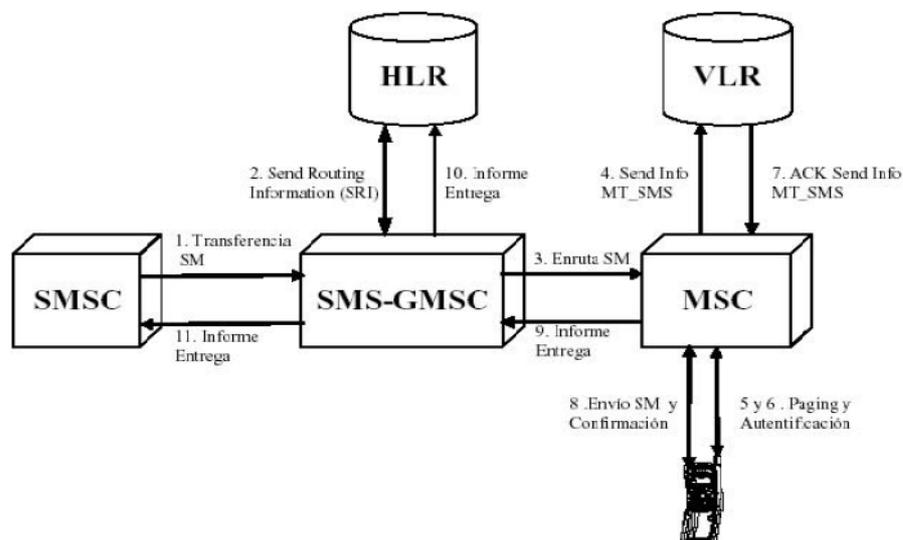


Figura 1.9: Recepción del mensaje

1.8.2.6 Sistema de Estación Base o BSS: Agrupa las partes específicas relativas a los aspectos de radio y celulares de GSM; además, está en contacto directo con las estaciones móviles mediante la interfaz de radio.

1.8.3 Operaciones para el envío de mensajes SMS

Para que un mensaje de texto SMS llegue a su destino se realiza el siguiente procedimiento:

- El mensaje corto es enviado del SME al SMSC.
- Después de completar su proceso interno, el SMSC pregunta al HLR y recibe del mismo información de encaminamiento del usuario móvil.
- El SMSC envía el mensaje corto hacia el MSC.
- El MSC extrae la información del usuario del VLR.
- El MSC transfiere el mensaje corto al MS.
- El MSC devuelve al SMSC el resultado de la operación que se está llevando a cabo. Si lo solicita el SME, el SMSC retorna un informe indicando la salida del mensaje corto.

1.8.4 Aplicaciones comunes del SMS

Debido a que los mensajes SMS son recibidos prácticamente de inmediato por el destinatario y son un medio de comunicación muy personal, muchos los están utilizando como el mejor medio para comunicarse con una comunidad para invitar a eventos, dar avisos, enviar alarmas, coordinar evacuaciones, etc.

1.9 CABLE DE COMUNICACIÓN SERIAL USB CA-42¹²

1.9.1 Definición

El Cable de Conectividad Nokia CA-42 nos permite la transferencia y sincronización bidireccional de datos, información y archivos en general entre el computador con los celulares compatibles; esto es, pasar la información de tu celular al computador y archivos que tengas en el PC puedes colocarlos en tu celular. Se puede transferir Imágenes, Fotografías, Videos, Aplicaciones, Temas, Animaciones, Juegos y, otras utilidades que dependen del celular.

La figura 1.10, representa el cable en mención el cual realiza la transferencia de datos entre el circuito y el teléfono celular receptor.



Figura 1.10: Cable de datos USB CA-42

¹²<http://apreciosderemate.com/cables-de-datos-nokia/71-usb-data-cable-ca-42-nokia-3220-6200-5100-6220-720.html>

1.9.2 Características

- El cable en cuestión lo que hace es una conversión USB <-> COM (o puerto de serie).
- Estos cables usan interfaz USB y no serie.
- El cable CA-42 lleva el convertidor TTL a USB incorporado en la parte del USB.
- El CA-42 solo es detectado por el ordenador si el teléfono está conectado al otro extremo.

1.9.3 Lista de compatibilidad

Nokia 2115i, Nokia 2270, Nokia 2285, Nokia 3100, Nokia 3120, Nokia 3200, Nokia 3205, Nokia 3220, Nokia 3570, Nokia 3585, Nokia 3585i, Nokia 3586i, Nokia 3587i, Nokia 3588i, Nokia 3589i, Nokia 5100, Nokia 5140, Nokia 6015i, Nokia 6016i, Nokia 6019i, Nokia 6020, Nokia 6200, Nokia 6225, Nokia 6560, Nokia 585, Nokia 6610, Nokia 6800, Nokia 6820, Nokia 6822, Nokia 7210, Nokia 7250i.

1.10 MÓDEM GSM¹³

Inicialmente el término inglés módem, es un acrónimo de “modulador/demodulador”; se trata de un equipo externo o interno (tarjeta módem), utilizado para la comunicación de computadoras a través de líneas analógicas de transmisión de voz y/o datos.

Para poder dar una definición clara de lo que un módem de lo que significa un módem GSM, analizaremos los conceptos de modulación y demodulación de señales.

¹³ <http://es.wikipedia.org/wiki/M%C3%B3dem>

1.10.1 Modulación

Modulación engloba el conjunto de técnicas que se usan para transportar información sobre una onda portadora, típicamente una onda sinusoidal. Estas técnicas permiten un mejor aprovechamiento del canal de comunicación lo que posibilita transmitir más información en forma simultánea además de mejorar la resistencia contra posibles ruidos e interferencias. Según la American National Standard for Telecommunications, la modulación es el proceso, o el resultado del proceso, de variar una característica de una portadora de acuerdo con una señal que transporta información. El propósito de la modulación es sobreponer señales en las ondas portadoras.

Básicamente, la modulación consiste en hacer que un parámetro de la onda portadora cambie de valor de acuerdo con las variaciones de la señal moduladora, que es la información que queremos transmitir.

La figura 1.11, presenta una onda de baja frecuencia (portadora, las dos de abajo) que puede modularse en amplitud (AM, varía la amplitud) o en frecuencia (FM, varía la frecuencia).

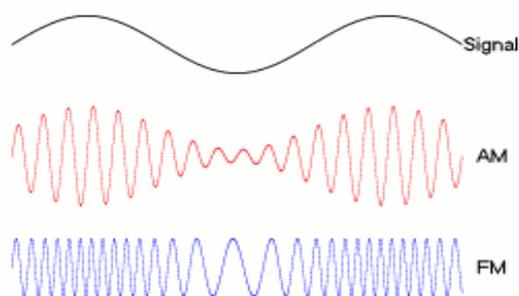


Figura 1.11: Ejemplo de modulación en AM y FM

Hay muchos tipos de modulación y demodulación analógicas y digitales:

1.10.1.1 Modulación Analógica

Dentro de las modulación analógica se conocen tres técnicas de modulación básica:

- Modulación de la amplitud (AM o amplitud modulada).
- Modulación de la frecuencia (FM o frecuencia modulada).
- Modulación de la fase (PM o fase modulada).

La mayoría de los sistemas de comunicación utilizan alguna de estas tres técnicas de modulación básicas, o una combinación de ellas.

1.10.1.2 Modulación Digital

Los siguientes son algunos de los casos extremos de estas técnicas:

- Modulación por desplazamiento de amplitud(ASK, *Amplitude Shift Keying*): Desactiva la amplitud durante toda la trayectoria.
- Modulación por desplazamiento de frecuencia(FSK, *Frecuency Shift eying*): Salta a una frecuencia extrema.
- Modulación por desplazamiento de fase(PSK, *Phase Shift Keying*): Desplaza la fase 180 grados.

Dentro de la modulación digital, tenemos también la modulación GMSK.

1.10.1.2.1 GMSK (Gaussian Minimum Shift Keying): La modulación GMSK es el tipo usado en modulación GSM, este tipo de modulación deriva de la modulación MSK (es un tipo especial de FSK, en el que la desviación de frecuencia pico es igual a la mitad de la tasa de bits) que es importante en los tipos de modulaciones digitales. Tiene tal importancia porque muchos de los datos pasan a través de un filtro gaussiano (como lo dice su acrónimo), esto a su vez estabiliza las variaciones de las frecuencias instantáneas a través del tiempo y reduce sus lóbulos laterales en el espectro que se transmite, esta forma de modulación es usada en radio comunicaciones digitales.

En la variedad de modulaciones digitales, resalta la GMSK por que permite el mejor uso de poder eficiente de amplificadores de Radio Frecuencia (RF).

1.10.2 Demodulación

En telecomunicaciones el término demodulación engloba el conjunto de técnicas utilizadas para recuperar la información transportada por una onda portadora, que en el extremo transmisor había sido modulada con dicha información. Este término es el opuesto a modulación.

En conclusión en cualquier telecomunicación normalmente existirá al menos una pareja modulador-desmodulador. Un modulador traduce un mensaje de señal a una señal de portador que opera dentro de la banda de frecuencia de los medios de comunicación. Un demodulador recibe el mensaje de la compañía y lo traduce en información. Un módem, que toma su nombre de modulación, o "mod" y la desmodulación o "dem", transfiere información de un ordenador a otro sobre una línea telefónica.

1.10.3 Definición de Módem GSM

Es un tipo especializado de módem que acepta una tarjeta SIM, y funciona a través de una suscripción a un operador de telefonía móvil. Desde la perspectiva del operador de telefonía móvil, un módem GSM se ve como un teléfono móvil. El término módem GSM se usa como un término genérico para referirse a cualquier módem que soporta uno o más de los protocolos en la familia evolutiva GSM, incluyendo la 2.5G tecnologías GPRS y EDGE, así como la 3G, entre otras.

1.10.4 Características más importantes de un Módem GSM

- Un módem GSM también podría ser un teléfono móvil GSM estándar con el cable adecuado y el controlador de software para conectarse a un puerto serie o puerto USB de su ordenador.
- Cuando un módem GSM se conecta a un ordenador, permite que el equipo utilice el módem GSM para comunicarse a través de la red móvil. Si bien

estos módems GSM se utilizan con mayor frecuencia para proporcionar conectividad móvil a Internet, muchos de ellos también se puede utilizar para enviar y recibir SMS y mensajes MMS.

- Un módem GSM expone una interfaz que permite a las aplicaciones tales como SMS enviar y recibir mensajes a través de la interfaz de módem. Los operadores móviles de este mensaje a enviar y recibir, realizan su trabajo como si se tratara directamente en un teléfono móvil. Para realizar estas tareas de enviar / recibir mensajes SMS, un módem GSM debe estar apoyado de un "extendido conjunto de comandos".
- Módems GSM son una forma rápida y eficiente para empezar con SMS, porque una suscripción especial a un proveedor de servicios de SMS no es necesario. En la mayoría de las partes del mundo, los módems GSM son una solución rentable para la recepción de mensajes SMS, ya que el emisor paga por la entrega de mensajes.

1.11 COMANDOS AT¹⁴

1.11.1 Definición

Los comandos AT son instrucciones codificadas que conforman un lenguaje de comunicación entre el hombre y un terminal módem. Los comandos AT se denominan así por la abreviatura de *attention*.

1.11.2 Presentación de los comandos AT

El software de comunicaciones se comunica con el módem en un idioma de comando especial que a menudo se conoce como el juego de comandos AT, es el único que el módem comprende.

¹⁴ http://es.wikipedia.org/wiki/Conjunto_de_comandos_Hayes

Por lo general, el software de comunicación le permite controlar el módem sin esfuerzo y de manera conveniente. Puede apenas seleccionar las opciones y operaciones requeridas desde menús en el programa de software de comunicaciones y el programa de comunicaciones transmite estas selecciones al módem en el formato de comando requerido. De inmediato, el módem procesa los comandos y realiza la tarea en particular. Existe diversos comandos AT, los mismos que cumplen diferentes funciones de acuerdo a sus aplicaciones.

1.11.3 Funciones de los comandos AT

Entre las principales funciones de los comandos AT tenemos los siguientes:

- Configuración del teléfono para una conexión inalámbrica, a través de infrarrojos, bluetooth o por cable por el puerto serial.
- Configurar el modem interno del teléfono.
- Solicitar información sobre la configuración actual o estado operativo del teléfono celular.
- Probar las disponibilidades que ofrece el teléfono celular.

1.11.4 Módem y comandos AT

- La mayoría de los módems se controlan y responden a caracteres enviados desde la computadora.
- El lenguaje de comandos para módem más extendido es el de los comandos Hayes que fue inicialmente incorporado a los módems de este fabricante.
- Casi todos estos comandos comienzan con AT (ATtention) y terminan con un retorno del carro (enter). Por lo primero es que son comúnmente llamados comandos AT, cada módem utiliza una serie de ordenes "AT" comunes y otras específicas.

1.12 COMUNICACIÓN SERIAL¹⁵

1.12.1 Definición

La comunicación serial consiste en el envío de un bit de información de manera secuencial; esto es, un bit a la vez y a un ritmo acordado entre el emisor y el receptor. La comunicación serial es un protocolo muy común para comunicación entre dispositivos que se incluye de manera estándar en prácticamente cualquier computadora.

Típicamente, la comunicación serial se utiliza para transmitir datos en formato ASCII. Para realizar la comunicación se utilizan 3 líneas de transmisión, que son:

- 1) Tierra (o referencia).
- 2) Transmitir o transmisión.
- 3) Recibir o recepción.

1.12.2 Características de la comunicación serial: Entre las principales características, tenemos las siguientes:

1.12.2.1 Transmisión modulada en amplitud

Es la comunicación serie a través de la interpretación de dos niveles lógicos de tensión o corriente, denominado *formato marca/espacio*.

El nivel lógico "1" representa un estado de tensión o corriente denominado marca, el nivel lógico "0" representa un estado de tensión o corriente denominado espacio.

¹⁵ <http://digital.ni.com/public.nsf/allkb/039001258CEF8FB686256E0F005888D1>

La figura 1.12, representa los niveles lógicos, sean estos de tensión o corriente del formato serie denominado marca/espacio.

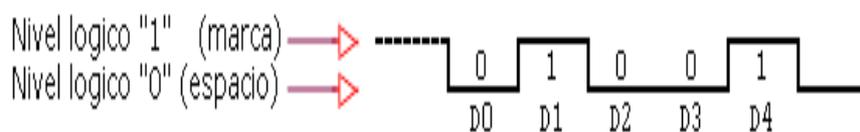


Figura 1.12: Formato serie marca/espacio

1.12.2.2 Estándares con formato marca/espacio

Uno de los estándares más comunes que usa el formato marca/espacio es la “norma RS232”.

1.12.2.2.1 Norma RS232: Es una de las normas más populares empleadas en la comunicación serie; además esta norma RS232 es la que se utiliza en los PC's, si bien hoy día está ampliamente superada por la transmisión serie a través de USB, de manera que está remitiendo su uso. Está patrocinada por la EIA (Asociación de Industrias Eléctricas). La norma RS232 resuelve tres aspectos en la comunicación que se establece entre el Equipo Terminal de Datos o **DTE** y el Equipo para la Comunicación de Datos o **DCE**, estos aspectos son:

1.12.2.2.1.1 Características eléctricas: Se establece que la longitud máxima del cable no debe ser superior a los 15 metros y la velocidad máxima de transmisión es, en principio, 128.000 bps. Los niveles lógicos no son compatibles TTL, considerando:

- 1 lógico entre -3V y -15V.
- 0 lógico entre +3V y +15V.

1.12.2.2.1.2 Características mecánicas de los conectores: Se utiliza un conector de 25 patillas, DB 25, o de 9 patillas, DB 9, donde el conector macho identifica al DTE y el conector hembra al DCE.

1.12.2.2.1.3 Descripción funcional de las señales usadas: Las señales están básicamente divididas en dos grupos:

- Señales primarias, que son normalmente utilizadas para la transferencias de datos.
- Señales secundarias, utilizadas para el control de la información que será transferida.

La norma RS232 está definida tanto para la transmisión síncrona como para la asíncrona, pero cuando se utiliza esta última, sólo un conjunto de terminales es utilizado.

1.12.2.3 Consideraciones en la comunicación serie

Cuando se transmite información a través de una línea serie es necesario utilizar un sistema de codificación que permita resolver los siguientes problemas:

1.12.2.3.1 Sincronización de bits: El receptor necesita saber dónde comienza y dónde termina cada bit en la señal recibida para efectuar el muestreo de la misma en el centro del intervalo de cada símbolo (bit para señales binarias).

1.12.2.3.2 Sincronización del carácter: La información serie se transmite por definición bit a bit, pero la misma tiene sentido en palabras o bytes.

1.12.2.3.3 Sincronización del mensaje: Es necesario conocer el inicio y fin de una cadena de caracteres por parte del receptor para por ejemplo, detectar algún error en la comunicación de un mensaje.

1.12.2.4 Velocidad de transmisión

La velocidad de transmisión de datos es expresada en bits por segundo o baudios. El baudio es un concepto más general que bit por segundo. El

primero queda definido como el número de estados de la señal por segundo, si sólo existe dos estados (que pueden ser representados por un bit, que identifica dos unidades de información) entonces baudio es equivalente a bit por segundo. Baudio y bit por segundo se diferencian cuando es necesario más de un bit para representar más de dos estados de la señal.

La velocidad de transmisión está estandarizada según la norma RS232C en baudios: 75, 110, 150, 300, 600, 1200, 2400, 4800, 9600 y 19200.

1.12.2.5 La base de reloj

Cuando se establece la comunicación es necesario implementar una base de tiempo que controle la velocidad. En un microcontrolador, se utilizaría la base de tiempos del reloj del sistema; en términos genéricos, se utilizaría uno de los siguientes métodos:

- Mediante la división de la base de reloj del sistema; por ejemplo, mediante un contador temporizador programable.
- A través de un oscilador TTL; para cambiar frecuencia hay que cambiar el cristal.
- Generador de razón de baudios; existen diferentes dispositivos especializados que generan diferentes frecuencias de reloj.

1.12.2.6 Líneas o canales de comunicación

Se pueden establecer canales para la comunicación de acuerdo a tres técnicas, siempre tomando al microprocesador o microcontrolador como referencia (transmisor) y al periférico como destino (receptor), estos canales son:

1.12.2.6.1 Simplex: En ella la comunicación serie, usa una dirección y una línea de comunicación. Siempre existirá un transmisor y un receptor, no ambos. La ventaja de este sistema consiste en que es necesario sólo un

enlace a dos hilos; y la desventaja, radica en que el extremo receptor no tiene ninguna forma de avisar al extremo transmisor sobre su estado y sobre la calidad de la información que se recibe. Esta es la razón por la cual, generalmente, no se utiliza.

1.12.2.6.2 Semiduplex: La comunicación serie se establece a través de una sola línea, pero en ambos sentidos. En un momento el transmisor enviará información y en otro recibirá, por lo que no se puede transferir información en ambos sentidos de forma simultánea .

Este modo de transmisión, permite la transmisión desde el extremo receptor de la información, sobre el estado de dicho receptor y sobre la calidad de la información recibida por lo que permite así la realización de procedimientos de detección y corrección de errores.

1.12.2.6.3 Full duplex: Se utilizan dos líneas (una transmisora y otra receptora) y se transfiere información en ambos sentidos. La ventaja de este método es que se puede transmitir y recibir información de manera simultánea.

1.12.2.7 Modos de transmisión

Existen dos modos básicos para realizar la transmisión de datos, estos son:

1.12.2.7.1 Modo asíncrono: Las transmisiones asíncronas son aquellas en que los bits que constituyen el código de un carácter se emiten con la ayuda de impulsos suplementarios que permiten mantener en sincronismo los dos extremos.

En la figura 1.13, se presenta la forma como se realiza la comunicación asincrónica; esto es, la velocidad de envío de los datos es acordada a priori entre el emisor y el receptor.

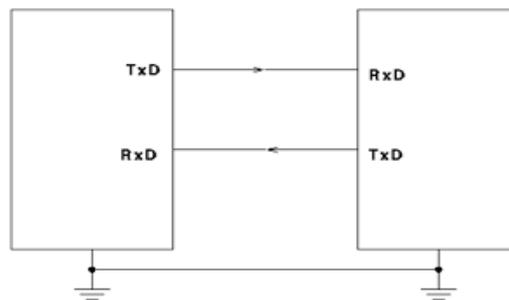


Figura 1.13: Comunicación asincrónica

Dentro de la comunicación asincrónica, debemos considerar, los siguientes aspectos:

1.12.2.7.1.1 Bit de inicio y bit de parada: En la transmisión asíncrona un carácter a transmitir es encuadrado con un indicador de inicio y fin de carácter, de la misma forma que se separa una palabra con una letra mayúscula y un espacio en una oración. La forma estándar de encuadrar un carácter es a través de un bit de inicio y un bit de parada. Durante el intervalo de tiempo en que no son transferidos caracteres, el canal debe poseer un "1" lógico. Al bit de parada se le asigna también un "1". Al bit de inicio del carácter a transmitir se le asigna un "0". Por todo lo anterior, un cambio de nivel de "1" a "0" lógico le indicará al receptor que un nuevo carácter será transmitido.

La figura 1.14, presenta un formato básico de como se realiza la transmisión asincrónica, considerando el bit de inicio, nuevo carácter y el bit de parada.

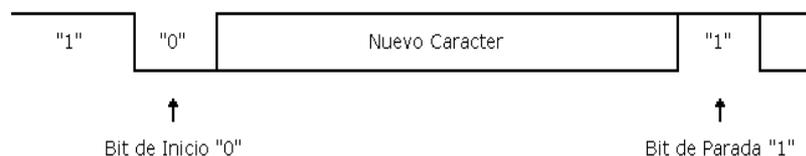


Figura 1.14: Formato básico de transmisión asincrónica

1.12.2.7.1.2 Reglas de transmisión asíncrona: La transmisión asíncrona que vamos a ver es la definida por la norma RS232, la cual se basa en las siguientes reglas:

- a. Cuando no se envían datos por la línea, ésta se mantiene en estado alto (1).
- b. Cuando se desea transmitir un carácter, se envía primero un bit de inicio que pone la línea a estado bajo (0) durante el tiempo de un bit.
- c. Durante la transmisión, si la línea está a nivel bajo, se envía un 0 y si está a nivel alto se envía un 1.
- d. A continuación se envían todos los bits del mensaje a transmitir con los intervalos que marca el reloj de transmisión. Por convenio se transmiten entre 5 y 8 bits.
- e. Se envía primero el bit menos significativo, siendo el más significativo el último en enviarse.
- f. A continuación del último bit del mensaje se envía el bit (o los bits) del final que hace que la línea se ponga a 1 por lo menos durante el tiempo mínimo de un bit. Estos bits pueden ser un bit de paridad para detectar errores y el bit o bits de stop, que indican el fin de la transmisión de un carácter.

Los datos codificados por esta regla, pueden ser recibidos siguiendo los pasos siguientes:

- a. Esperar la transición 1 a 0 en la señal recibida.
- b. Activar el reloj con una frecuencia igual a la del transmisor.
- c. Muestrear la señal recibida al ritmo de ese reloj para formar el mensaje.
- d. Leer un bit más de la línea y comprobar si es 1 para confirmar que no ha habido error en la sincronización.

1.12.2.7.1.3 Velocidad de transmisión en el modo asincrónico: En la transmisión asíncrona por cada carácter se envía al menos 1 bit de inicio y 1 bit de parada así como opcionalmente 1 bit de paridad. Esta es la razón de que los baudios no se correspondan con el número de bits de datos que son transmitidos.

1.12.2.7.2 Modo sincrónico: La transmisión sincrónica, es un método más eficiente de comunicación en cuanto a velocidad de transmisión. Ello viene

dado porque no existe ningún tipo de información adicional entre los caracteres a ser transmitidos.

En la figura 1.15 se presenta la forma en que se realiza la comunicación sincrónica, el envío de los datos es sincronizado por el emisor a partir de un pulso constante de reloj (Clock), con cada pulso envía un nuevo dato.

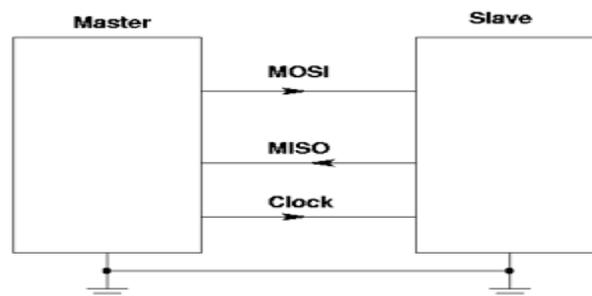


Figura 1.15: Comunicación sincrónica

Cuando se transmite de manera síncrona lo primero que se envía es un octeto de sincronismo ("sync"). El octeto de sincronismo realiza la misma función que el bit de inicio en la transmisión asíncrona, indicando al receptor que va a ser enviado un mensaje. Este carácter, además utiliza la señal local de reloj para determinar cuándo y con qué frecuencia será muestreada la señal; es decir, permite sincronizar los relojes de los dispositivos transmisor y receptor.

La figura 1.16, representa un octeto de sincronismo utilizado en la transmisión sincrónica.

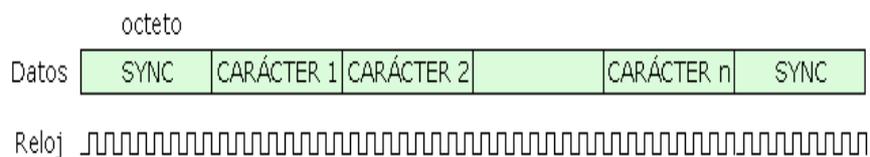


Figura 1.16: Formato básico de transmisión sincrónica

Los caracteres de sincronismo deben diferenciarse de los datos del usuario para permitir al receptor detectar los caracteres "sync". Por ejemplo, el código ASCII utiliza el octeto 10010110.

1.12.2.7.3 Detectar errores en la comunicación: Cuando se escriben o se envían datos, pueden producirse errores, entre otras cosas, por ruidos inducidos en las líneas de transmisión de datos. Es por tanto necesario comprobar la integridad de los datos transmitidos mediante algún método que permita determinar si se ha producido un error. En un caso típico, si al transmitirse un mensaje se determina que se ha producido un error, el receptor solicita de nuevo el mensaje al emisor.

Se pueden detectar errores de acuerdo a los siguientes métodos:

1.12.2.7.3.1 Generadores y detectores de paridad: Como un error en una transmisión serie solamente suele afectar a un bit, uno de los métodos más comunes para detectar errores es el control de la paridad, el cual consiste en añadir un bit, denominado de paridad, a los datos que se envían o escriben.

La figura 1.17, representa la forma en que los generadores y detectores de paridad permite al receptor detectar en un bit cualquier error que haya ocurrido durante la transmisión.

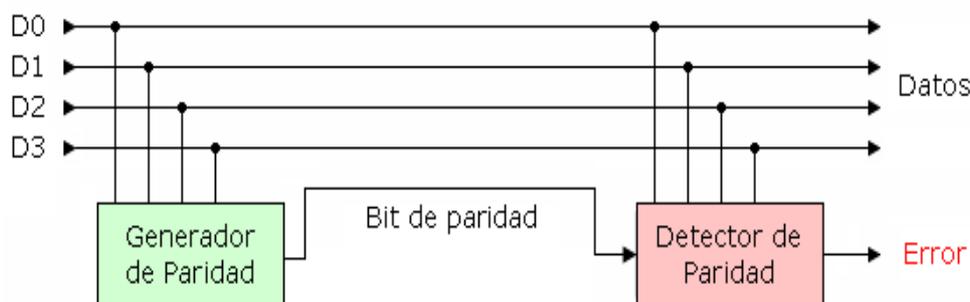


Figura 1.17: Generadores y detectores de paridad

La paridad puede ser par o impar:

1.12.2.7.3.1.1 Paridad par: El bit de paridad será cero, cuando el número de bit "unos" que contienen los datos a transmitir sea un número par, y el bit de paridad será uno cuando los datos que se mandan contienen un número impar de unos.

La tabla 1.3, representa los bits de paridad par; esto es, la suma de los bits que son unos, contando datos y bit de paridad dará siempre como resultado un número par de unos.

Tabla 1.3: Bits de paridad par

Dato	Paridad
0000 0001	1
0101 0001	1
0101 0101	0
0000 0000	0

1.12.2.7.3.1.2 Paridad impar: En el sistema de paridad impar, el número de unos (datos + paridad) siempre debe ser impar.

La tabla 1.4, representa los bits de paridad impar de acuerdo a lo descrito en el párrafo anterior.

Tabla 1.4: Bits de paridad impar

Dato	Paridad
0000 0001	0
0101 0001	0
0101 0101	1
0000 0000	1

1.12.2.7.3.1.3 El error de sobrescritura (overwrite): La UART o Receptor/Transmisor Asíncrono Universal, presenta una estructura interna con doble buffer, ello le permite tener un carácter almacenado en el registro almacenador (buffer) de la recepción mientras que el registro de desplazamiento serie paralelo continúa ensamblando un nuevo carácter. El receptor implementará una rutina que suma todos los bytes de datos recibidos y al resultado se le sumará el último byte (que posee la información en complemento a dos de la suma de los caracteres transmitidos) y si la recepción del mensaje ha sido correcta, el resultado debe ser cero.

1.12.2.7.4 Conversión Serie/Paralelo: Como un sistema microprogramable basado en CPU es un dispositivo que inherentemente maneja los datos de forma paralela, debe realizarse una conversión para obtener el formato de datos serie que requiere los dispositivos periféricos que pueda tener conectados a través de la comunicación serie.

Existen dos formas en que se pueden implementar la conversión serie/paralelo, esta puede ser conversión por software y conversión por hardware.

1.12.2.7.4.1 Conversión por software: Para este método es necesario utilizar un terminal de salida de un puerto conectado al microprocesador o una patilla del microcontrolador. De la misma manera para la recepción se necesitará un terminal de entrada.

1.12.2.7.4.1.1 La transmisión de datos: La rutina de transmisión puede ser llamada por interrupción, cada vez que un dato serie quiera ser transmitido o por que se establezca en una o varias partes del flujo del programa que ejecuta el microcontrolador.

1.12.2.7.4.1.2 La recepción de datos: Una rutina de recepción por software puede ser llamada cada un milisegundo para detectar la presencia de un bit de inicio o por interrupción.

1.12.2.7.4.2 Conversión por hardware: En la conversión por software, el algoritmo se basa simplemente en operaciones de rotación o desplazamiento de registros. Los elementos que permiten la conversión por hardware se basan en registros de desplazamiento. Existen tres tipos de dispositivos que permiten la conversión Serie/Paralelo:

- El Receptor/Transmisor Asíncrono Universal o UART: Usado en la transmisión serie asíncrona.
- El Receptor/Transmisor Síncrono Universal o USRT: Usado en la transmisión serie síncrona, usado en la transmisión a alta velocidad.

- El Receptor/Transmisor Síncrono/Asíncrono Universal o USART: Permite la comunicación serie de forma tanto asíncrona como síncrona.

A la UART se le deben especificar los siguientes parámetros:

- Bits de datos por caracter (usualmente de 5 a 8).
- Bits de parada (1, 1,5 y 2).
- Bit de paridad, para utilizar su capacidad de detección de error.
- Velocidad de transmisión.

1.13 EL MICROCONTROLADOR¹⁶

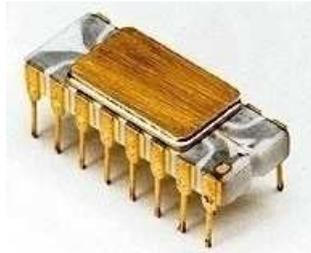
1.13.1 Introducción

Cuando no existían los microprocesadores, los ingenieros y aficionados a la electrónica creaban sus circuitos electrónicos mediante diseños que implicaban muchos componentes individuales y complejos cálculos matemáticos. Un circuito lógico básico a menudo requería de muchos elementos electrónicos basados en transistores, resistencias, circuitos integrados con compuertas lógicas, etc., lo cual implicaba circuitos con muchos ajustes y posibilidades de fallos. En el año 1971 apareció el primer microprocesador (desarrollado por Intel, el 4004) el cual originó un cambio decisivo en las técnicas de diseño de la mayoría de los equipos. A partir de ese momento, fue posible construir un equipo electrónico con un microprocesador, unos pocos circuitos accesorios y un software. Los diseños electrónicos comenzaron a ser mucho más pequeños y simplificados; los diseñadores de equipos electrónicos podían realizar mayor cantidad de tareas en menos tiempo y el tamaño de los equipos se redujo considerablemente; sin embargo, después de cierto tiempo apareció una nueva tecnología, llamada MICROCONTROLADOR que simplificó aún más el diseño

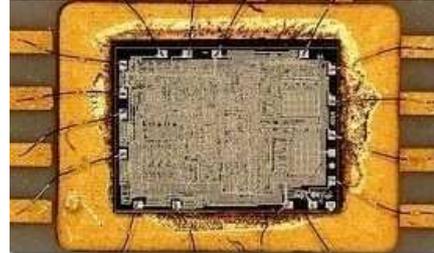
¹⁶ <http://es.wikipedia.org/wiki/Microcontrolador>

electrónico, al incluir en un mismo encapsulado el núcleo del microprocesador, la memoria y las entradas/salidas.

En la figura 1.18, se presenta el primer microprocesador Intel 4004 en forma de encapsulado y como está constituido su parte interior.



Intel 4004, primer microprocesador



Interior del microprocesador 4004

Figura 1.18: Primer microprocesador, INTEL 4004

1.13.2 Definición

Un microcontrolador es un circuito integrado que incluye en su interior las tres unidades funcionales de una computadora: unidad central de procesamiento o CPU, memoria y periféricos de entrada y salida.

En la figura 1.19, se presenta un modelo de microcontrolador (Motorola 68HC11) en forma de encapsulado o chip.



Figura 1.19: Microcontrolador Motorola 68HC11 y chips de soporte

Los microcontroladores son diseñados para reducir el costo económico y el consumo de energía de un sistema en particular. Por eso el tamaño de la unidad central de procesamiento, la cantidad de memoria y los periféricos incluidos dependerán de la aplicación.

Una de las mayores ventajas de los microcontroladores es la posibilidad de realizar modificaciones en el comportamiento de nuestro proyecto simplemente actualizando el software que ejecuta el microcontrolador.

1.13.3 Estructura de un microcontrolador

En la figura 1.20, vemos al microcontrolador en forma de circuito integrado, con su procesador (CPU), buses, memoria, periféricos y puertos de entrada/salida.

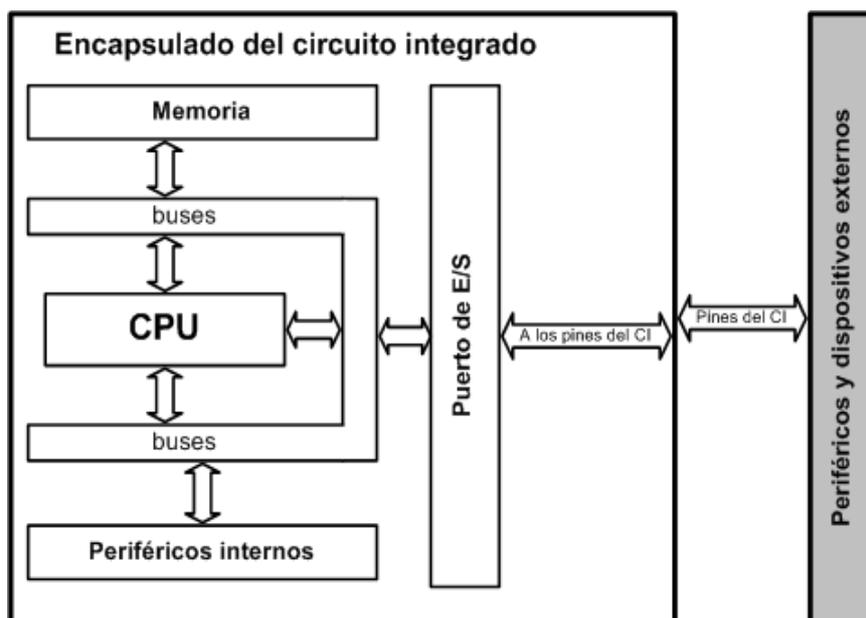


Figura 1.20: Estructura básica de un microcontrolador

Fuera del encapsulado se ubican otros circuitos para completar periféricos internos y dispositivos que pueden conectarse a los pines de entrada/salida. También se conectarán a los pines del encapsulado la alimentación, masa, circuito de complemento del oscilador y otros circuitos necesarios para que el microcontrolador pueda trabajar.

1.13.4 Núcleo de un microcontrolador

Aún cuando el microcontrolador es una computadora embebida dentro de un circuito integrado, se compone de un núcleo y un conjunto de circuitos adicionales. Dentro del núcleo se encuentran el procesador y la memoria, todo ello estructurado de forma tal que conforme una arquitectura de computadora.

1.13.5 Tipos de arquitecturas de microcontroladores

Básicamente existen dos arquitecturas de computadoras: Von Neumann y Harvard. Ambas se diferencian en la forma de conexión de la memoria al procesador y en los buses que cada una necesita.

1.13.5.1 Arquitectura Von Neumann

La arquitectura tradicional de computadoras y microprocesadores está basada en la arquitectura Von Neumann, en la cual la Unidad Central de Proceso (CPU), está conectada a una Memoria única donde se guardan las instrucciones del programa y los datos.

La figura 1.21, representa la Arquitectura Von Neuman en la cual la Unidad Central de Proceso se conecta con la Memoria a través de un Bus de Datos.

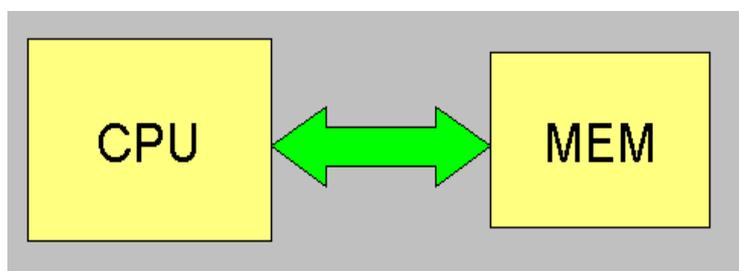


Figura 1.21: Arquitectura Von Neumann

El tamaño de la unidad de datos o instrucciones está fijado por el ancho del bus que comunica la memoria con la CPU.

1.13.5.1.1 Principales limitaciones de la arquitectura Von Neumann: Entre las principales limitaciones de la arquitectura Von Neumann, se cita las siguientes:

- La limitación de la longitud de las instrucciones por el bus de datos, que hace que el microprocesador tenga que realizar varios accesos a memoria para buscar instrucciones complejas.
- La limitación de la velocidad de operación a causa del bus único para datos e instrucciones que no deja acceder simultáneamente a unos y otras, lo cual impide superponer ambos tiempos de acceso.

1.13.5.2 Arquitectura Harvard

La arquitectura **HARVARD** tiene la unidad central de proceso (CPU) conectada a dos memorias por medio de dos buses diferentes. Ambos buses son totalmente independientes y pueden ser de distintos anchos. Una de las memorias contiene solamente las instrucciones del programa (Memoria de Programa), y la otra sólo almacena datos (Memoria de Datos); además, al ser los buses independientes, la CPU puede acceder a los datos para completar la ejecución de una instrucción, y al mismo tiempo leer la siguiente instrucción a ejecutar.

La figura 1.22, representa la Arquitectura Harvard en la cual la Unidad Central de Proceso esta conectada a la Memoria de Programa y a la Memoria de Datos a través de dos buses independientes que pueden ser también de distinto ancho.

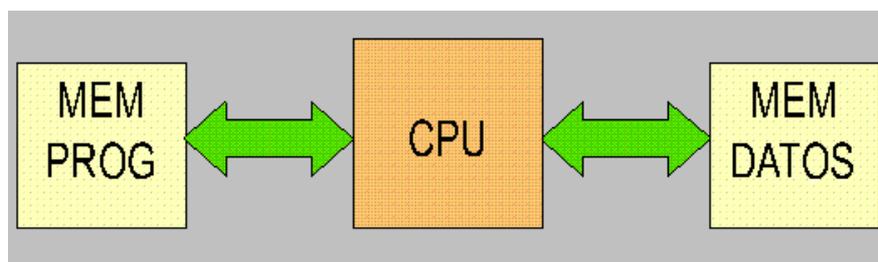


Figura N° 1.22: Arquitectura Harvard

La ventaja fundamental de esta arquitectura es que permite adecuar el tamaño de los buses a las características de cada tipo de memoria; también, el procesador puede acceder a cada una de ellas de forma simultánea, lo que se traduce en un aumento significativo de la velocidad de procesamiento. La desventaja de esta arquitectura, está en que consume muchas líneas de E/S del procesador; por lo que en sistemas donde el procesador está ubicado en su propio encapsulado, solo se utiliza en supercomputadoras.

1.13.6 Procesador en detalle del microcontrolador

Como se señaló anteriormente, el primer microprocesador Intel 4004 que salió al mercado en 1971, es una máquina digital sincrónica compleja, como cualquier otro circuito lógico secuencial sincrónico; sin embargo, la ventaja de este componente está en que aloja internamente un conjunto de circuitos digitales que pueden hacer operaciones corrientes para el cálculo y procesamiento de datos, pero desde una óptica diferente, sus entradas son una serie de códigos bien definidos, que permiten hacer operaciones de carácter específico cuyo resultado está determinado por el tipo de operación y los operandos involucrados.

Lo primordial en un microprocesador está en que la combinación adecuada de los códigos de entrada, su ejecución secuencial, el poder saltar hacia atrás o adelante en la secuencia de códigos en base a decisiones lógicas u órdenes específicas, permiten que la máquina realice diversas operaciones complejas, no contempladas en los simples códigos básicos.

1.13.7 Características principales de un microcontrolador:

1.13.7.1 Registros

Son un espacio de memoria muy reducido pero necesario para cualquier microprocesador, de aquí se toman los datos para varias operaciones que debe realizar el resto de los circuitos del procesador. Los registros sirven también

para almacenar los resultados de la ejecución de instrucciones, cargar datos desde la memoria externa o almacenarlos en ella. Una parte de los registros, la destinada a los datos, es la que determina uno de los parámetros más importantes de cualquier microprocesador, esto significa que mientras mayor sea el número de bits de los registros de datos del procesador mayores serán sus prestaciones, en cuanto a poder de cómputo y velocidad de ejecución, ya que este parámetro determina la potencia que se puede incorporar al resto de los componentes del sistema.

1.13.7.2 Unidad de control

La unidad de control es uno de los elementos fundamentales que determinan las prestaciones del procesador, ya que su tipo y estructura, determina parámetros tales como el tipo de conjunto de instrucciones, velocidad de ejecución, tiempo del ciclo de máquina, tipo de buses que puede tener el sistema, manejo de interrupciones, entre otros elementos, dependiendo del tipo de procesador.

En la Unidad de Control, recae principalmente la lógica necesaria para la decodificación y ejecución de las instrucciones, el control de los registros, la unidad aritmética y lógica (ALU), los buses, etc.

1.13.7.3 Unidad aritmética-lógica

Como los procesadores son circuitos que hacen básicamente operaciones lógicas y matemáticas, se le dedica a este proceso una unidad completa, con cierta independencia. Aquí es donde se realizan las sumas, restas, y operaciones lógicas típicas del álgebra de Boole.

Actualmente este tipo de unidades ha evolucionado mucho y los procesadores más modernos tienen varias ALU, especializadas en la realización de operaciones complejas como las operaciones en coma flotante.

En muchos casos le han cambiado su nombre por el de “coprocesador matemático”, aunque este es un término que surgió para dar nombre a un tipo especial de procesador que se conecta directamente al procesador más tradicional.

1.13.7.4 Buses

Son el medio de comunicación que utilizan los diferentes componentes del procesador para intercambiar información entre sí, eventualmente los buses o una parte de ellos estarán reflejados en los pines del encapsulado del procesador.

En el caso de los microcontroladores, no es común que los buses estén reflejados en el encapsulado del circuito, ya que estos se destinan básicamente a las E/S de propósito general y periféricos del sistema.

Existen tres tipos de buses:

1.13.7.4.1 Buses de Dirección: Se utiliza para seleccionar al dispositivo con el cual se quiere trabajar o en el caso de las memorias, seleccionar el dato que se desea leer o escribir.

1.13.7.4.2 Buses de Datos: Este Bus, trabaja en conjunción con el Bus de Direcciones para transportar los datos a través del computador. El tamaño del Bus de Datos puede ser de 16, 32 o 64 bits.

1.13.7.4.3 Buses de control: Se utiliza para gestionar los distintos procesos de escritura/ lectura y controlar la operación de los dispositivos del sistema.

1.13.7.5 Conjunto de instrucciones

Define las operaciones básicas que puede realizar el procesador, que conjugadas y organizadas forman lo que se conoce como SOFTWARE. Existen

dos tipos básicos de repertorios de instrucciones, que determinan la arquitectura del procesador: CISC y RISC.

1.13.7.5.1 Computadora de Conjunto de Instrucciones Complejo (CISC): Los microprocesadores CISC tienen un conjunto de instrucciones que se caracteriza por ser muy amplio y que permiten realizar operaciones complejas entre operandos situados en la memoria o en los registros internos. Este tipo de repertorio dificulta el paralelismo entre instrucciones, por lo que en la actualidad, la mayoría de los sistemas CISC de alto rendimiento, convierten las instrucciones complejas en varias instrucciones simples del tipo RISC, llamadas generalmente microinstrucciones.

1.13.7.5.2 Computadora con Conjunto de Instrucciones Reducido (RISC): Se centra en la obtención de procesadores con las siguientes características fundamentales:

- Instrucciones de tamaño fijo;
- Pocas instrucciones;
- Sólo las instrucciones de carga y almacenamiento acceden a la memoria de datos;
- Número relativamente elevado de registros de propósito general.

Una de las características más destacables de este tipo de procesadores es que posibilitan el paralelismo en la ejecución, y reducen los accesos a memoria. Es por eso que los procesadores más modernos, tradicionalmente basados en arquitecturas CISC implementan mecanismos de traducción de instrucciones CISC a RISC, para aprovechar las ventajas de este tipo de procesadores.

1.13.7.6 Memoria

En los microcontroladores la memoria no es abundante, aquí no encontrará Gigabytes de memoria como en las computadoras personales. Típicamente la

memoria de programas no excederá de 16 K-localizaciones de memoria no volátil (FLASH O EPROM) para contener los programas.

La memoria de acceso aleatorio o RAM, está destinada al almacenamiento de información temporal que será utilizada por el procesador para realizar cálculos u otro tipo de operaciones lógicas. En el espacio de direcciones de memoria RAM se ubican además los registros de trabajo del procesador y los de configuración y trabajo de los distintos periféricos del microcontrolador.

El tipo de memoria utilizada en las memorias RAM de los microcontroladores es memoria de acceso aleatorio estática o SRAM, lo que evita tener que implementar sistemas de refrescamiento como en el caso de las computadoras personales, que utilizan gran cantidad de memoria, típicamente alguna tecnología de memoria de acceso aleatorio dinámica o DRAM. La memoria SRAM poseen pequeñas cantidades de memoria RAM y es el tipo de memoria adecuado para los microcontroladores.

En el caso de la memoria de programas se utilizan diferentes tecnologías, describimos a continuación las mayor utilización:

1.13.7.6.1 ROM de máscara: En este caso no se “graba” el programa en memoria sino que el microcontrolador se fabrica con el programa, es un proceso similar al de producción de los CD comerciales mediante masterización.

1.13.7.6.2 One Time Programmable o OTP: Este tipo de memoria, también es conocida como PROM (memoria solo de lectura programable) o simplemente ROM. Los microcontroladores con memoria OTP se pueden programar una sola vez, con algún tipo de programador.

Se utilizan en sistemas donde el programa no requiera futuras actualizaciones y para series relativamente pequeñas, donde la variante de máscara sea muy

costosa, también para sistemas que requieren serialización de datos, almacenados como constantes en la memoria de programas.

1.13.7.6.3 Erasable Programmable Read Only Memory o EPROM: Los microcontroladores con este tipo de memoria son muy fáciles de identificar porque su encapsulado es de cerámica y llevan encima una ventanita de vidrio desde la cual puede verse la oblea de silicio del microcontrolador.

Se fabrican así porque la memoria EPROM es reprogramable, pero antes debe borrarse, y para ello hay que exponerla a una fuente de luz ultravioleta, el proceso de grabación es similar al empleado para las memorias OTP. Se utilizaban en sistemas que requieren actualizaciones del programa y para los procesos de desarrollo y puesta a punto.

1.13.7.6.4 Electrical Erasable Programmable Read Only Memory o EEPROM: Fueron el sustituto natural de las memorias EPROM, la diferencia fundamental es que pueden ser borradas eléctricamente, por lo que la ventanilla de cristal de cuarzo y los encapsulados cerámicos no son necesarios.

1.13.7.6.5 Memoria Flash: En el campo de las memorias reprogramables para microcontroladores, son el último avance tecnológico en uso a gran escala, y han sustituido a los microcontroladores con memoria EEPROM. A las ventajas de las memorias flash se le adicionan su gran densidad respecto a sus predecesoras lo que permite incrementar la cantidad de memoria de programas a un costo muy bajo; pueden además ser programadas con las mismas tensiones de alimentación del microcontrolador, el acceso en lectura y la velocidad de programación es superior, disminución de los costos de producción, entre otras.

1.13.7.7 Interrupciones

Las interrupciones son esencialmente llamadas a subrutina generadas por los dispositivos físicos, al contrario de las subrutinas normales de un programa en

ejecución. Como el salto de subrutina no es parte del hilo o secuencia de ejecución programada, el controlador guarda el estado del procesador en la pila de memoria y entra a ejecutar un código especial llamado “manejador de interrupciones” que atiende al periférico específico que generó la interrupción.

Al terminar la rutina, una instrucción especial le indica al procesador el fin de la atención de la interrupción. En ese momento el controlador reestablece el estado anterior, y el programa que se estaba ejecutando antes de la interrupción sigue como si nada hubiese pasado. Las rutinas de atención de interrupciones deben ser lo más breves posibles para que el rendimiento del sistema sea satisfactorio, por que normalmente cuando una interrupción es atendida, todas las demás interrupciones están en espera.

Los procesos de atención a interrupciones tienen la ventaja de que se implementan por hardware ubicado en el procesador, así que es un método rápido de hacer que el procesador se dedique a ejecutar un programa especial para atender eventos que no pueden esperar por mecanismos lentos como el de encuesta.

1.13.7.8 Periféricos

Cuando observamos la organización básica de un microcontrolador, señalamos que dentro de este se ubican un conjunto de periféricos, cuyas salidas están reflejadas en los pines del microcontrolador.

A continuación describiremos algunos de los periféricos que con mayor frecuencia encontraremos en los microcontroladores.

1.13.7.8.1 Entradas y salidas de propósito general: También conocidos como puertos de E/S, generalmente agrupadas en puertos de 8 bits de longitud, permiten leer datos del exterior o escribir en ellos desde el interior del microcontrolador, el destino habitual es el trabajo con dispositivos simples como relés, leds, o cualquier otra cosa que se le ocurra al programador.

Algunos puertos de E/S tienen características especiales que le permiten manejar salidas con determinados requerimientos de corriente, o incorporan mecanismos especiales de interrupción para el procesador.

1.13.7.8.2 Temporizadores y contadores: Son circuitos sincrónicos para el conteo de los pulsos que llegan a su para poder conseguir la entrada de reloj. Si la fuente de un gran conteo es el oscilador interno del microcontrolador es común que no tengan un pin asociado, y en este caso trabajan como temporizadores

Por otra parte, cuando la fuente de conteo es externa, entonces tienen asociado un pin configurado como entrada, este es el modo contador.

Los temporizadores son uno de los periféricos más habituales en los microcontroladores y se utilizan para muchas tareas, como por ejemplo, la medición de frecuencia, implementación de relojes, para el trabajo de conjunto con otros periféricos que requieren una base estable de tiempo entre otras funcionalidades.

1.13.7.8.3 Conversor Analógico/Digital: Como es muy frecuente el trabajo con señales analógicas, éstas deben ser convertidas a digital y por ello muchos microcontroladores incorporan un conversor analógico/digital, el cual se utiliza para tomar datos de varias entradas diferentes que se seleccionan mediante un multiplexor.

1.13.7.8.4 Conversor Digital/Análogo: Transforma los datos digitales obtenidos del procesamiento del computador en su correspondiente señal analógica que saca al exterior por una de las patitas de la cápsula.

1.13.7.8.5 Puertos de comunicación: Existen diversos puertos de comunicación en un microcontrolador de los cuales se analizará los que se consideran más importantes.

1.13.7.8.5.1 Puerto serie: Este periférico está presente en casi cualquier microcontrolador, normalmente en forma de UART (Universal Asynchronous Receiver Transmitter) o USART (Universal Synchronous Asynchronous Receiver Transmitter) dependiendo de si permiten o no el modo sincrónico de comunicación.

El destino común de este periférico es la comunicación con otro microcontrolador o con una PC y en la mayoría de los casos hay que agregar circuitos externos para completar la interfaz de comunicación. La forma más común de completar el puerto serie es para comunicarlo con una PC mediante la interfaz EIA-232 (más conocida como RS232).

1.13.7.8.5.2 2SPI: Este tipo de periférico se utiliza para comunicar al microcontrolador con otros microcontroladores o con periféricos externos conectados a él, por medio de una interfaz muy sencilla.

1.13.7.8.5.3 USB: Es un sistema que trabaja por monitoreo (polling) de un conjunto de periféricos inteligentes por parte de un amo, que es normalmente un computador personal. Cada nodo inteligente está gobernado inevitablemente por un microcontrolador.

1.13.7.8.5.4 Comparadores: Son circuitos analógicos basados en amplificadores operacionales que tienen la característica de comparar dos señales analógicas y dar como salida los niveles lógicos '0' o '1' en dependencia del resultado de la comparación.

1.13.7.8.5.5 Modulador de ancho de pulsos: Los Moduladores de ancho de pulsos (o PWM por sus siglas en inglés) son periféricos muy útiles sobre todo para el control de motores, sin embargo hay un grupo de aplicaciones que pueden realizarse con este periférico, dentro de las cuales podemos citar: inversión DC/AC para UPS, conversión digital analógica D/A, control regulado de luz (dimming), etc.

1.13.7.8.5.6 Memoria de datos no volátil: Muchos microcontroladores han incorporado la memoria de datos no volátil, como un periférico más, para el almacenamiento de datos de configuración o de los procesos que se controlan dentro de ellos.

Esta memoria es independiente de la memoria de datos tipo RAM o la memoria de programas, en la que se almacena el código del programa a ejecutar por el procesador del microcontrolador.

1.14 EL MICROCONTROLADOR ATMEL AVR¹⁷

1.14.1 Introducción

ATMEL fabrica los microcontroladores de la familia AVR, esta nueva tecnología proporciona todos los beneficios habituales de arquitectura RISC y Memoria Flash Reprogramable Eléctricamente.

La característica que los identifica a estos microcontroladores de ATMEL es la Memoria Flash y EEPROM que incorpora. El diseño AVR de ATMEL difiere de los demás microcontroladores de 8 bits por tener mayor cantidad de registros (32) y un conjunto ortogonal de instrucciones. La arquitectura AVR es más fácil de programar a nivel de lenguaje ensamblador y es fácil de optimizar con un compilador.

El gran conjunto de registros disminuye la dependencia respecto a la memoria, lo cual mejora la velocidad y disminuye las necesidades de almacenamiento de datos; además, casi todas las instrucciones se ejecutan en 1 ó 2 ciclos de reloj.

Adicionalmente, ATMEL también proporciona en línea el entorno software que permite editar, ensamblar y simular el código fuente. Una vez ensamblado y

¹⁷ <http://www.clubse.com.ar/DIEGO/NOTAS/2/nota18.htm>

depurado el código fuente del programa, se transferirá el código máquina a la Memoria Flash del microcontrolador para esto se debe disponer de otro entorno de desarrollo para programar en forma serial o paralelo la Memoria Flash.

Los tipos de encapsulado del microcontrolador del Atmega presenta desde 28 pines hasta 100 pines en la forma de DIP, TQFP y MLF y su voltaje de alimentación está en el rango de 1.8 a 5.5 voltios.

1.14.2 El microcontrolador ATMEGA8A¹⁸

El microcontrolador ATMEGA8A es de 8 bits, su procesador presenta características avanzadas de tipo **RISC**, segmentado y arquitectura Harvard así como otras que se describen más adelante.

1.14.2.1 Características principales del microcontrolador ATMEGA8A

La tecnología RISC, presentan instrucciones con complejidad reducida, permite una rápida ejecución de las instrucciones que se ejecutan en un solo ciclo de reloj, el ATMEGA8A consigue obtener 1 MIPS por Mhz, permitiendo al diseñador del sistema optimizar su consumo de energía versus la velocidad de procesamiento.

Las instrucciones en la memoria de programas son ejecutados con estructura segmentada (pipelining), al mismo tiempo que una instrucción es ejecutado, se realiza la búsqueda de la próxima instrucción. Este concepto permite habilitar instrucciones para ser ejecutados con cada ciclo de reloj.

En la figura 1.23, se presenta un diagrama de bloques del microcontrolador Atmega8A con sus características principales.

¹⁸http://www.sc.ehu.es/sbweb/webcentro/automatica/web_avr/archivos/Otros%20AVRs/ATmega/ATmega8.htm

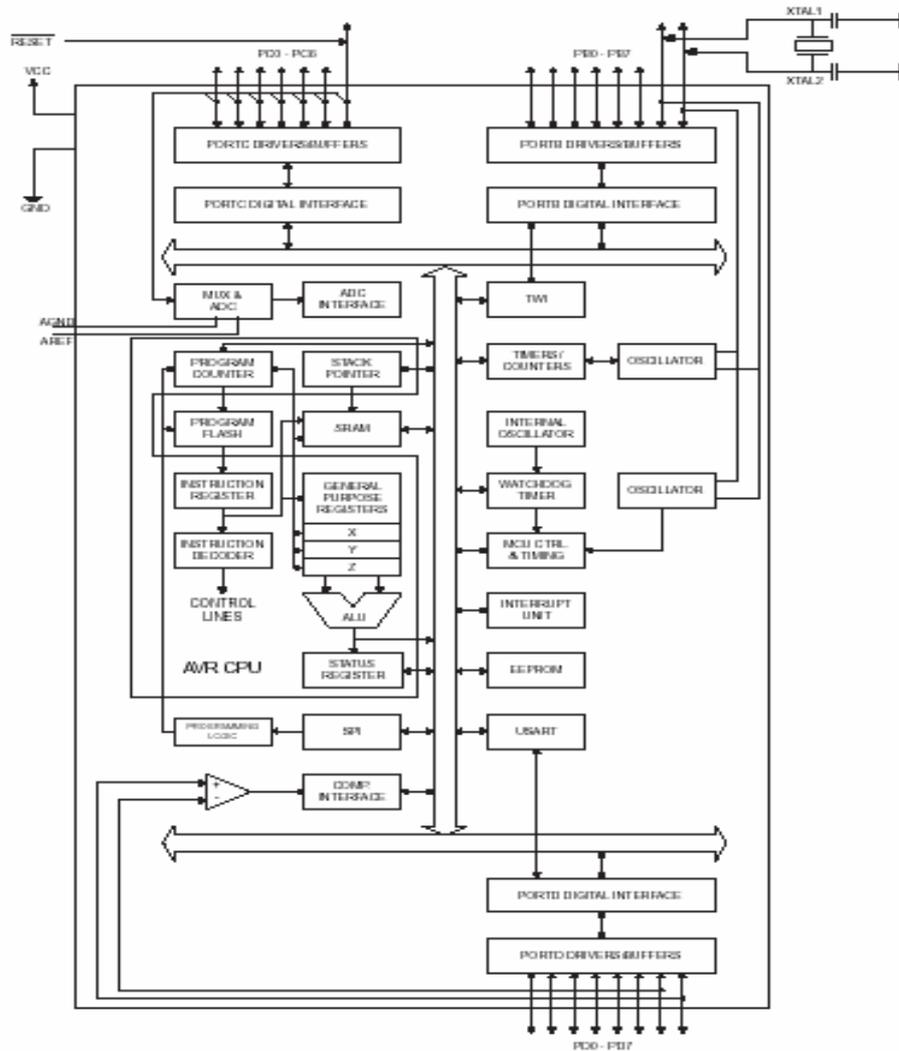


Figura 1.23: Diagrama de Bloques del Atmega8A

- Tiene 32 registros de 8 bits de propósito general. Todos estos registros están conectados a la unidad aritmética lógica (ALU) para un rápido acceso, una instrucción es ejecutada con uno solo ciclo de reloj.
- Tipos de Memoria: Memoria Flash de 8 Kbytes, EEPROM de 512 bytes y SRAM de 1 Kbytes.

1.14.2.1.1 Característica de los periféricos internos:

- Temporizador/contador de 8 bits con pre-escalador y comparador.
- 1 temporizador/contador de 16 bits con pre-escalador, comparador y capturador.

- 8 canales de entrada para cada convertidor A/D (en TQFP y MLF), conformados por: 6 canales A/D de 10 bits y 2 canales A/D de 8 bits.
- 6 canales de entrada para cada convertidor A/D (tipo PDIP), conformados por: 4 canales A/D de 10 bits y 2 canales A/D de 8 bits.
- 1 USART (módulo programable para comunicación serial).
- 1 módulo SPI, para interface serial (master/slave).
- 1 perro guardián.
- 1 comparador analógico.
- Puertos programables de entrada/salida:
 - Puerto B, con 8 líneas.
 - Puerto C, con 7 líneas.
 - Puerto D, con 8 líneas.
- Velocidad de operación: 0-16 MHz.
- Voltaje de alimentación: 4.5 a 5.5 voltios.
- Tipo de empaque: PDIP de 28 pines, TQFP de 32 pines y MLF de 32 pines
- Otras características: fuentes de interrupción internas y externas, oscilador interno de 1, 2, 4 y 8 Mhz, entre otras.

1.14.2.2 Descripción de los pines del ATMEGA8A

En la figura 1.24, se presenta los pines por los cuales está conformado el microcontrolador Atmega8A.

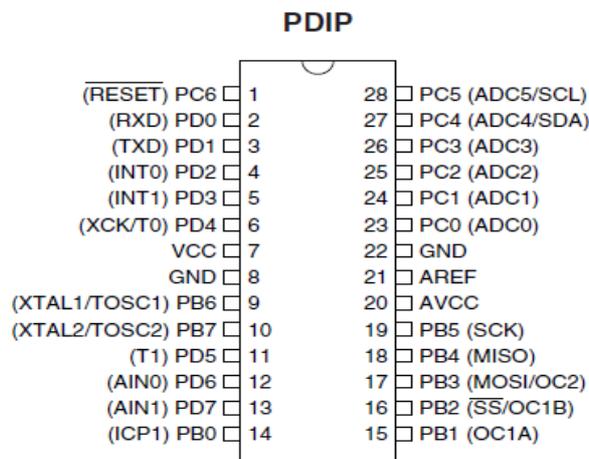


Figura 1.24: Pines del microcontrolador Atmega8A

- **VCC y GND:** Son los pines de alimentación (+5 v) y tierra (0 v).
- **XTAL1 y XTAL2:** Conectores del cristal de reloj externo.
- **RESET:** Corresponde a la línea de reset (entrada).
- **AVCC:** Es el pin para conectar la fuente de alimentación al convertidor A/D.
- **AREF:** Para conectar una tensión de referencia para el conversor A/D interno.
- **Puerto B (PB7 .. PB0):** Compuesto de 8 bits, a cada pin le corresponde un bit, son bidireccionales con resistencia interna “pull up” para cada bit.
- **Puerto C (PC0 .. PC5):** Tiene 7 bits, bidireccionales con resistencias internas pull up, para cada bit.
- **PC6/RESET:** El bit 6, puede ser usado como entrada/salida si los fusibles han sido programados, en caso contrario, PC6 es usado para la entrada Reset. Un bajo nivel en este pin generará un reset.
- **Puerto D (PD0..PD7):** Son 8 líneas bidireccionales de entrada/salida con resistencias internas pull up, para cada uno. Los pines del puerto D, también tiene otras funciones, por ejemplo PD0 y PD1 permiten usar el periférico de comunicación serial USART, PD1 permite la transmisión de datos y PD0, permite la recepción de datos.

1.14.2.3 La Unidad Central de Procesos (CPU)

La función de la CPU es controlar la operación del microcontrolador, permitiendo la ejecución correcta del programa, habilitando el acceso a las memorias, controlar los periféricos y manejar las interrupciones.

Para obtener una mejor performance y paralelismo, el Atmega8A, utiliza la arquitectura Harvard, gracias a ella se puede acceder de forma simultánea e independiente a la memoria de datos y a la memoria de instrucciones.

Los registros de archivo de acceso rápido contienen 32 registros de propósito general de 8 bits y son accesados con un solo ciclo de reloj. Esto permite usar un solo ciclo en una operación aritmética-lógica (ALU).

De los 32 registros de 8 bits 6 pueden ser usados como 3 registros de 16 bits para direccionamiento indirecto apuntando el espacio de memoria de datos. Estos registros adicionales son X, Y, y Z, de 16 bits. La ALU, permite realizar las operaciones aritméticas y lógicas entre registros o entre un registro y una constante. Luego de una operación aritmética, el registro de estados es modificado para reflejar la información acerca del resultado de una operación.

El flujo de un programa es variado por los saltos condicionales e incondicionales y las instrucciones de llamada para dirigirse a la nueva dirección en el espacio de memoria. Las instrucciones del AVR, tiene normalmente un formato compuesto por una palabra de 6 bits. Cada dirección en la memoria de programas contiene instrucciones de 16 ó 32 bits.

La figura 1.25, representa un diagrama de bloques de la Arquitectura AVR.

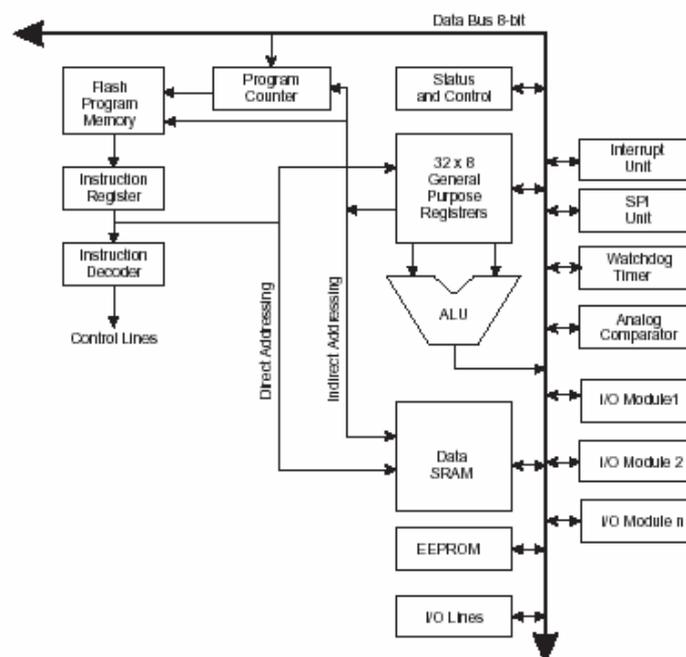


Figura 1.25: Arquitectura AVR

1.14.2.4 El Registro de estados

El registro de estados contiene información acerca del resultado de la más reciente instrucción ejecutada. Esta operación puede ser utilizada para alterar

el flujo de un programa. El registro de estados es sobrescrito luego de ejecutarse una operación en la ALU.

En la figura 1.26, se presenta al Registro de Estados con sus respectivos indicadores, bits y valor inicial.

<i>Bit</i>	7	6	5	4	3	2	1	0
	I	T	H	S	V	N	Z	C
<i>Lectura/escritura</i>	R/W							
<i>Valor inicial</i>	0	0	0	0	0	0	0	0

Figura 1.26: Registro de Estados

El registro de estados, contiene los siguientes indicadores:

1.14.2.4.1 I: Habilitador de interrupciones (Global interrupt enable): El bit I permite habilitar las interrupciones con “1” lógico, si el bit es “0”, no se aceptará ningún pedido de interrupción.

1.14.2.4.2 T: Copia, almacena un bit (Bit copy storage): La posición del bit T, permite ser el destino o la fuente de un bit determinado (es como un lugar temporal para guardar un bit). Para esto se utiliza las instrucciones BLD (bit Load) y BST (Bit Store).

1.14.2.4.3 H: Bandera de acarreo a la mitad (Half carry flag): Es la bandera de medio acarreo, se pone a “1” si en una suma de operandos de 8 bits, hay un acarreo del nibble menos significativo, al más significativo. Se pone a “0” en caso contrario.

1.14.2.4.4 S: Bit de signo (Sign Bit): El bit S siempre es un or-exclusivo entre la bandera negativa “N” y la bandera de desborde “V”.

1.14.2.4.5 V: Bandera de desbordamiento (Two’s complement overflow flag): La bandera V se hace 1 cuando el resultado de una instrucción excede el rango

de las operaciones aritméticas; es decir, en el caso de operaciones de 8 bits cuando el resultado de una operación no está entre -128 y $+127$ decimal y en el caso de 16 bits entre -32768 y $+32767$ decimal. Esto es algo que sucede usualmente en el complemento a dos.

1.14.2.4.6 N: Bandera negativo (Negative flag): La bandera N, se hace 1 cuando el resultado de una operación es un número negativo, o sea el bit más significativo del resultado vale 1. Se pone a 0 en caso contrario.

1.14.2.4.7 Z: Bandera de resultado cero (Zero flag): Se hace "1" cuando el resultado de un operación es cero, de lo contrario es cero.

1.14.2.4.8 C: Bandera de acarreo (Carry flag): La bandera de acarreo (C) se hace 1 cuando existe acarreo en el bit de mayor significado de una operación aritmética; de lo contrario es cero.

1.14.2.5 Registros de propósito general

En la figura 1.27, se presentan los 32 registros de propósito general que Atmega8A, dispone.

7	0 Dirección
R0	0x00
R1	0x01
R2	0x02
.....	
R13	0x0D
R14	0x0E
R15	0x0F
R16	0x10
R17	0x11
.....	
R26	0x1A
R27	0x1B

R28	0x1C
R29	0x1D
R30	0x1E
R31	0x1F

Figura 1.27: Registros de propósito general

Además, a cada registro le corresponde una dirección dentro de las 32 primeras posiciones en el espacio de memoria de datos, conforme se describe en el gráfico anterior.

1.14.2.6 Registros X, Y, Z

Los registros R26 y R27 forman el registro X de 16 bits, los registros R28 y R29 forman el registro Y de 16 bits, los registros R30 y R31 forman el registro apuntador Z de 16 bits.

En la figura 1.28, se presentan los Registros X,Y, Z que Atmega8A, dispone.

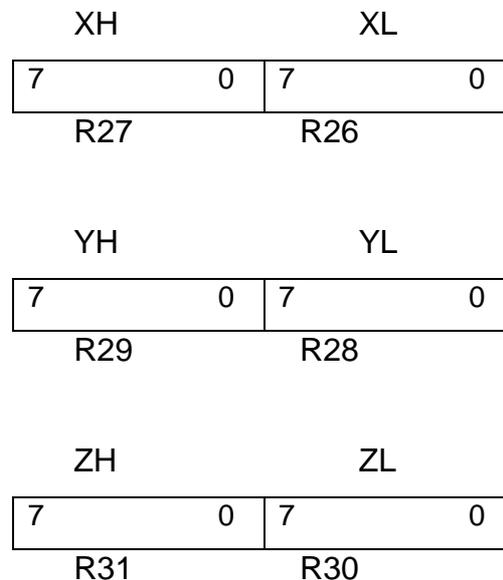


Figura 1.28: Los Registros X, Y, Z

1.14.2.7 Puntero de Pila

La pila es el área de espacio de memoria, utilizado temporalmente para guardar y recuperar datos y/o direcciones cuando la CPU está ejecutando una subrutina programada o una interrupción. El puntero de pila apunta (almacena la dirección) a la próxima dirección libre de la pila y que decrementa su valor en uno cada vez que se almacena un dato (de un byte) en ella, incrementándolo en uno cuando se retira este valor (de un byte). En el caso que se guarda la dirección de retorno cuando se atiende a una subrutina o una interrupción el puntero de pila es decrementado en 2 y luego de ejecutarse la última instrucción de retorno de la subrutina, el puntero de pila es incrementado en 2.

La figura 1.29, representa el puntero de pila en el AVR, el cual es implementado con 2 registros de 8 bits en el espacio de memoria E/S.

Bit	15	14	13	12	11	10	9	8	
	SP15	SP14	SP13	SP12	SP11	SP10	SP9	SP8	SPH
	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	SPL

Figura 1.29: Puntero de pila

1.14.2.8 La arquitectura Harvard y RISC

El microcontrolador Atmega8A utiliza la arquitectura conocida como Harvard; en cuanto a sus instrucciones es de tipo RISC (Reduced Instrucción Set Computer), esto permite que el set de instrucciones y el bus de la memoria de programa pueden diseñarse de manera tal que todas las instrucciones tengan una sola posición de memoria de programa de la misma longitud; además, como los buses son independientes, la CPU del microcontrolador puede estar accediendo a los datos para completar la ejecución de una instrucción, y al mismo tiempo estar leyendo la próxima instrucción a ejecutar.

En resumen Atmega8A, maneja:

- 130 instrucciones, la mayoría se ejecuta en un solo ciclo de reloj.
- 32 registros de propósito general de 8 bits.
- Hasta 16 MIPS si trabaja a 16 Mhz.

1.14.2.9 Memoria de programa y memoria de datos

El microcontrolador Atmega8A define 2 tipos de espacio de memoria, el espacio de la memoria de datos y el espacio de memoria de programas. Además tiene una memoria EEPROM.

1.14.2.9.1 Memoria de programa: Memoria de Programa o Memoria Flash del Atmega8A tiene una capacidad de 8K x 8, las instrucciones son de 16bits o 32 bits, por tanto esta memoria es organizado como 4Kx16 bits. Para mayor seguridad del espacio de la memoria flash (memoria de programas) está dividido en 2 áreas: arranque y de aplicación. Para acceder al espacio comprendido entre 0 y 4095 (\$FFF) direcciones, el tamaño del registro contador de programa (PC) será de 12 bits.

En la figura 1.30, se presenta a la Memoria de Programa del Atmega8A dividida en dos áreas: arranque y aplicación.

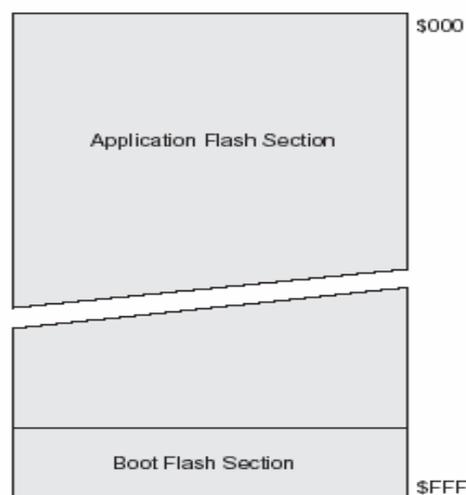


Figura 1.30: Memoria de programas

1.14.2.9.2 Memoria de datos: En la memoria de datos residen los registros de propósito general (R0...R31), los registros de E/S y los registros de la SRAM interna. El espacio de EEPROM consta de 512 bytes, donde opcionalmente se pueden guardar datos que no se pierden al desconectar la alimentación.

La figura 1.31, se presenta la organización de la memoria SRAM del Atmega8A.

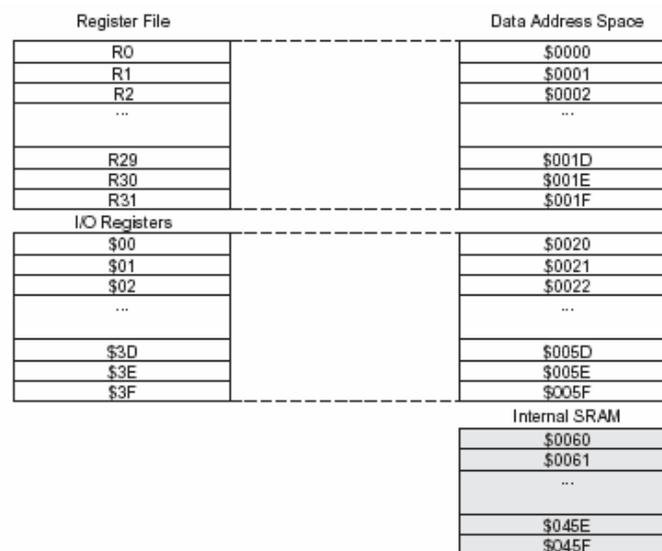


Figura 1.31: Mapa de la memoria de datos

Otros tipos de memorias que posee el microcontrolador Atmega8A, han sido descritos anteriormente cuando se trata del microcontrolador en forma general.

1.15. RELÉ DE SRD-06VDC-SL-C¹⁹

1.15.1 Definición

Este tipo de relé electromecánico, es un interruptor eléctrico que se abre y se cierra bajo el control de otro circuito eléctrico. Cuando una corriente fluye a través de la bobina, el campo magnético resultante atrae una armadura que

¹⁹ <https://en.wikipedia.org/wiki/Relay>

está unido mecánicamente a un contacto en movimiento. El movimiento ya sea que hace o rompe la conexión con un contacto fijo.

Cuando la corriente a la bobina está apagado, la armadura es devuelto por una fuerza de aproximadamente la mitad de fuerte que la fuerza magnética de su posición relajada; por lo general, este es un resorte. La mayoría de los relés se fabrican para operar rápidamente. En una aplicación de bajo voltaje, esto es para reducir el ruido. En una aplicación de alto voltaje o de alta corriente, esto es reducir la formación de arcos. Si la bobina se activa con DC, un diodo es con frecuencia instalado en la bobina, para disipar la energía del campo magnético en el colapso de desactivación, que de lo contrario podría generar un aumento de la tensión y puede causar daños a los componentes del circuito.

En la figura 1.32, se presenta al relé electromecánico (SRD-06VDC-SL-C) con sus características eléctricas en su encapsulado.



Figura 1.32: Relé (SRD-06VDC-SL-C)

1.15.2 Distribución de pines del relé de SRD-06VDC-SL-C

Este tipo de relé tiene la siguiente configuración:

Pin 1 y 2: Bobina.

Pin 3: Normalmente cerrado (NC).

Pin 4: Normalmente abierto (NA).

Pin 5: Común.

En la figura 1.33, se puede apreciar la configuración de los pines que posee el relé mencionado.

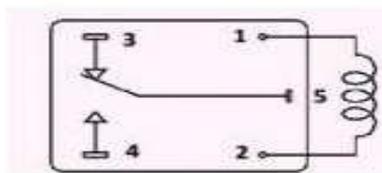


Figura 1.33: Distribución de pines del relé de SRD-06VDC-SL-C

1.16 PRINCIPALES TÓPICOS DEL LENGUAJE C++²⁰

El programa fue elaborado en el lenguaje de programación estructurado, denominado C++; éste dentro de su entorno está conformado por algunos operadores, funciones, sentencias, etc., que a continuación se detalla las más importantes:

1.16.1 Función delay

La función **delay()**, sirve para hacer una pausa de 'n' segundos. En algunos compiladores esta función se incluye en la librería conio.h, pero en nuestro caso lo hacemos nosotros mismos.

Sintaxis:

Delay(variable tipo int);

1.16.2 Operadores de comparación

La función de estos comparadores es justamente comparar dos variables; los comparadores de mayor interés se resume en la siguiente tabla.

²⁰ <https://es.wikipedia.org/wiki/C%2B%2B>

En la tabla 1.6, consta el nombre de los operadores más utilizados en el lenguaje de programación C++ con sus respectivas sintaxis.

Tabla 1.5: Nombre y sintaxis de algunos operadores en C++

Nombre del operador	Sintaxis
Menor que	<code>a < b</code>
Menor o igual que	<code><= b</code>
Mayor que	<code>a > b</code>
Mayor o igual que	<code>a >= b</code>
No igual que	<code>!= b</code>
Igual que	<code>a == b</code>
Negación lógica	<code>!a</code>
AND lógico	<code>a && b</code>
OR lógico	<code>a b</code>

1.16.3 Sentencias de decisión

Las sentencias de decisión o también llamadas de control de flujo son estructuras de control que realizan una pregunta la cual retorna un valor de verdadero o falso (evalúa una condición) y selecciona la siguiente instrucción a ejecutar dependiendo la respuesta o resultado.

1.16.3.1 Ejecución de las instrucciones

La ejecución de las instrucciones incluyendo una estructura de control como el condicional, funcionan de la siguiente manera:

- Las instrucciones comienzan a ejecutarse de forma secuencial (en orden) y cuando se llega a una estructura condicional, la cual esta asociada a una condición, se decide que camino tomar; dependiendo siempre del resultado de la condición, siendo esta falsa o verdadera.

- Cuando se termina de ejecutar este bloque de instrucciones se reanuda la ejecución en la instrucción siguiente a la de la condicional.

1.16.3.1.1 Sentencia if

La instrucción **if** es la más utilizada para construir estructuras de control de flujo, se puede configurar de dos formas:

- Primera Forma

Sintaxis:

if (condición)

```
{
    Set de instrucciones
}
```

Siendo "condición" el lugar donde se pondrá la condición que se tiene que cumplir para que sea verdadera la sentencia y así proceder a realizar el "set de instrucciones" o código contenido dentro de la sentencia.

- Segunda Forma

Ahora se verá la misma sintaxis pero se añadirá la parte "Falsa" de la sentencia:

Sintaxis:

if (condición)

```
{
    Set de instrucciones1 //PARTE VERDADERA
}
else
{
    Set de instrucciones2 //PARTE FALSA
}
```

La palabra “**else**” o “**de lo contrario**” indica al lenguaje que de lo contrario al no ser verdadera o no se cumpla la parte verdadera entonces realizará el "set de instrucciones 2".

1.16.4 Sentencias de iteración

Las Sentencias de **iteración** o **ciclos** son estructuras de control que repiten la ejecución de un grupo de instrucciones. Básicamente, una sentencia de iteración es una estructura de control condicional, ya que dentro de la misma se repite la ejecución de una o más instrucciones **mientras** o **hasta** que una a condición específica se cumpla. Muchas veces se tiene que repetir un número definido o indefinido de veces un grupo de instrucciones por lo que en estos casos se utiliza este tipo de sentencias.

En C++ los ciclos o bucles se construyen por medio de las sentencias **for**, **while** y **do –while**, en nuestro caso se hace uso de los dos primeras sentencias.

1.16.4.1 Sentencias For

La sentencia **for** es útil para los casos en donde se conoce de antemano el número de veces que una o más sentencias han de repetirse.

Sintaxis:

```
for(contador; final; incremento)
{
Codigo a Repetir;
}
```

donde:

- contador, es una variable numérica

- final, es la condición que se evalúa, o sea, el valor final para contador
- Incremento, es el valor que se suma o resta al contador

1.16.4.2 Sentencia while

La sentencia **while** es útil en aquellos casos en donde no se conoce de antemano el número de veces que una o más sentencias se tienen que repetir.

Sintaxis:

```
while(condicion)
{
código a Repetir
}
```

donde:

- **condición**, es la expresión a evaluar.

CAPÍTULO 2

PLANIFICACIÓN Y CONSTRUCCIÓN DEL HARDWARE

2.1 INTRODUCCIÓN

Previo la construcción del hardware, se hizo un estudio de los elementos que intervendrían en el circuito así como de sus características, con el propósito de evitar que sufran averías o que se dañen, sobre todo por una corriente o voltaje que sobrepase los rangos permitidos para algún elemento; además se han tomado también todas las precauciones necesarias en el momento de realizar la interconexión entre ellos, ya que también puede ser la causa para que algunos elementos se dañen o no funcione correctamente, por lo general al no estar bien polarizados.

El proyecto, en su mayoría está conformado por elementos eléctricos y electrónicos; entre ellos, el principal es el microcontrolador Atmega8A, al mismo que se lo considera como el cerebro del circuito ya que de éste dispositivo dependerá que el funcionamiento de las cargas se den de manera correcta; también se puede hacer referencia al relé, al transistor y al regulador de voltaje. Dentro de este mismo capítulo, se describe el software utilizado en el proyecto y por último se representa un esquema general del proyecto.

2.2 PLANIFICACIÓN Y CONSTRUCCIÓN DEL CIRCUITO

Durante los pasos dados para la planificación y construcción del circuito, siempre se ha manteniendo una secuencia ordenada, con el propósito de ir detectando los errores y poderlos corregir con anterioridad, para de esta manera poder obtener un circuito eficiente.

En la figura 2.1, se presenta el diagrama de bloques mediante el cual se indica de manera sintetizada las etapas que conforman el hardware del circuito.

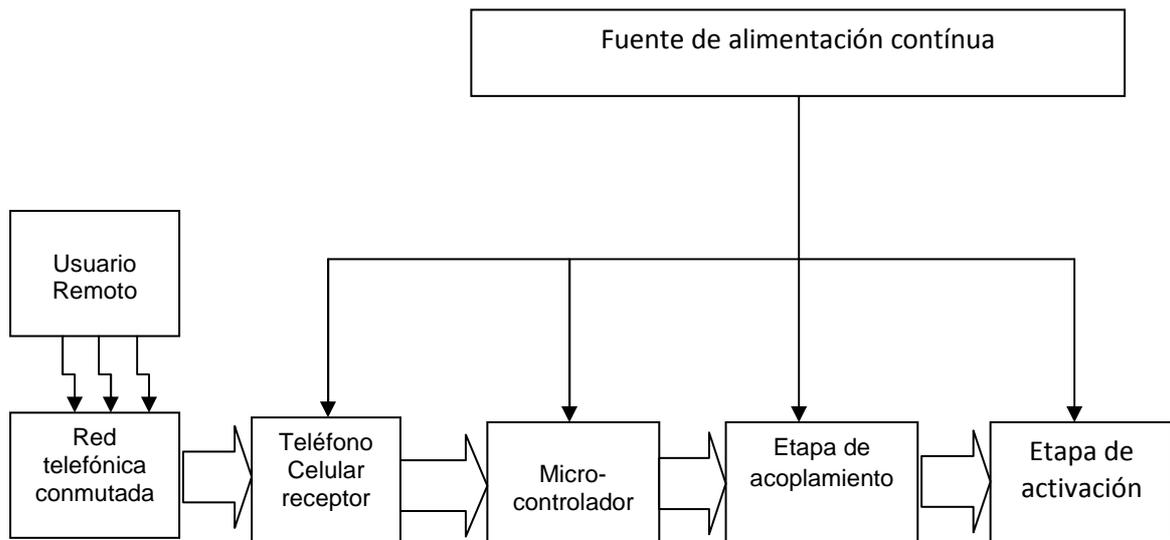


Figura 2.1: Diagrama de bloques del hardware del circuito

Estas etapas son descritas de manera detallada a continuación:

2.2.1 Etapa de alimentación

Para que se dé la comunicación entre las diferentes etapas y poder ver los resultados para el cual fue diseñado el proyecto, es necesario que el circuito esté alimentado con sus respectivos voltajes; por esta razón, el estudio de las etapas que conforman el diagrama de bloques, se lo hace tomando en cuenta el orden de funcionamiento del circuito.

2.2.1.1 Fuente de alimentación continua

Para alimentar el circuito se necesita un voltaje fijo y estable de corriente continua; estos voltajes tienen diferentes valores, dependiendo de las características eléctricas y electrónicas de sus elementos.

2.2.1.2 Proceso para obtener los 12, 5 y 3.8 voltios DC

El proceso que se dió para obtener los voltajes necesarios para hacer funcionar el circuito, es como sigue: se hace uso de la corriente y voltaje alterno que nos

provee la Empresa Eléctrica de Quito; posteriormente y, a través de una fuente de 12 voltios de corriente continua (adaptador), se obtiene el primer voltaje DC, éste voltaje se divide en dos partes de la siguiente manera: Utilizando un regulador de voltaje (LM 7805) colocado inmediatamente después de la fuente de 12 voltios, se logra obtener un segundo voltaje que es de 5 voltios, el mismo que es utilizado para alimentar, a los relés y sus respectivas cargas.

Por último, el voltaje de 3.8 voltios se obtiene conectando en serie dos diodos semiconductores (1N4007) después de la salida del regulador de voltaje. Los diodos nos dan una caída de voltaje de 0.6 voltios cada uno, que sumados nos dan 1.2 voltios; haciendo la diferencia de los 5 voltios menos 1.2 voltios obtenemos los 3.8 voltios; éste voltaje es común para alimentar tanto al microcontrolador como al teléfono celular receptor, tomando en cuenta que los rangos de voltaje para hacer funcionar al microcontrolador, esta entre 2.5V y 5.5V, mientras que el teléfono celular puede soportar un voltaje máximo de 3.8V.

2.2.1.3 Circuito de alimentación continua de 12, 5 y 3.8 voltios

La figura 2.2, presenta gráficamente el proceso seguido para obtener los voltajes descritos anteriormente.

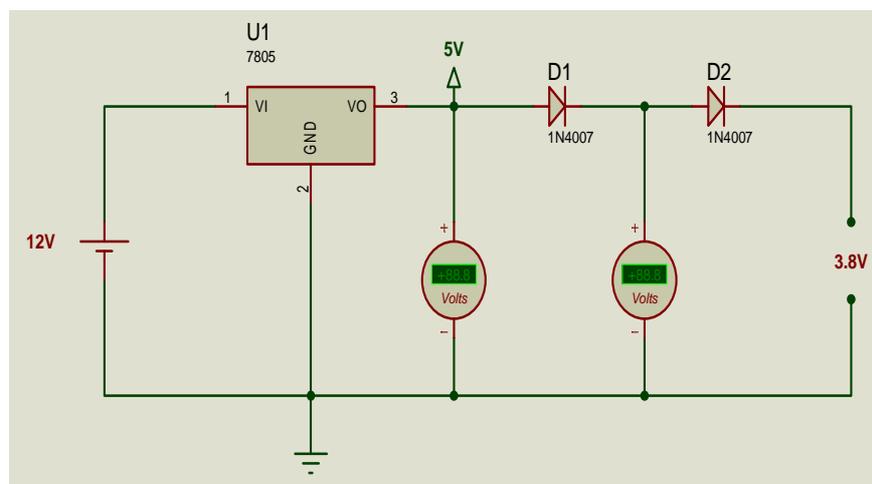


Figura 2.2: Circuito de 12, 5 y 3.8 voltios DC

Para calcular la potencia, se considera lo siguiente:

Datos del fabricante del regulador de voltaje (7805)

$$V_o = 5V$$

$$I_o = 1A$$

Datos del fabricante del relé (SRD-06VDC-SL-C)

$$V_o = 5V$$

$$I_o = 15mA$$

Datos del fabricante del microcontrolador

$$V_o = 3.8V$$

$$I_o = 89,3mA$$

Para los tres relés, será necesario una corriente de:

$$I_{relés} = 89.3mA \times 3 = 267.9mA$$

Sumados los 15mA del microcontrolador, serán:

$$I_t = 267.9mA + 15mA = 282.9mA$$

Como corriente de salida (I_o) del regulador de voltaje es 1A, esto significa que este voltaje es más que suficiente para alimentar a los relés.

Cálculo de la potencia a la salida del regulador de voltaje:

$$P = V \times I$$

$$P = 5V \times 1A = 5W$$

Cálculo de la potencia en el punto de los 3.8V:

$$I = 0.7A \text{ (corriente de salida de la fuente)}$$

$$P = V \times I$$

$$P = 3.8V \times 0.7A = 2.66W$$

2.2.2 Etapa del usuario remoto

Esta etapa trata de la interacción entre el usuario y el circuito, proceso que es imprescindible en el desarrollo del proyecto. La etapa del usuario remoto, constituido principalmente por el teléfono celular emisor, da inicio a la comunicación entre dicho dispositivo y el microcontrolador, por intermedio del teléfono celular receptor.

La función del teléfono celular emisor, consiste en el monitoreo de mensajes; en otras palabras, es el envío de mensajes SMS, los mismos que nos indicarán que relé y por ende su respectiva carga deben activarse o desactivarse, dependiendo del contenido de texto.

2.2.2.1 Monitoreo de mensajes SMS

Consiste en el envío de mensajes SMS, los mismos que realizarán las funciones de activar o desactivar las cargas, dependiendo del contenido de texto del mensaje.

En la tabla 2.1, se describe detalladamente este proceso, indicando el código de cada uno de los mensajes y su función correspondiente.

Tabla 2.1: Funciones que cumplen los diferente mensajes SMS

Código de los mensajes SMS	Función
ON1	Activa relé 1 y su respectiva carga
ON2	Activa relé 2 y su respectiva carga
ON3	Activa relé 3 y su respectiva carga
OF1	Desactiva relé 1 y su respectiva carga
OF2	Desactiva relé 2 y su respectiva carga
OF3	Desactiva relé 3 y su respectiva carga

Si el contenido de los mensajes SMS es diferente a los indicados en la tabla anterior, no se realizará ninguna acción; esto significa que el microcontrolador ignorará cualquier mensaje diferente a los antes indicados.

2.2.3 Etapa de la red telefónica conmutada²¹

La red telefónica conmutada establece la comunicación entre los teléfonos móviles transmisor y receptor, por intermedio de las ondas de radio y, las señales se transmiten a través del aire. Para lograr grandes coberturas será necesario utilizar potencias de transmisión muy elevadas.

La figura 2.3, presenta un esquema general de una red telefónica conmutada, en el cual las flechas nos indican el orden que se debe seguir para que la comunicación se realice de manera apropiada.



Figura 2.3: Esquema de una red telefónica conmutada

Cuando se realiza una llamada o un envío de datos desde un teléfono móvil, en realidad se está realizando una transmisión de radio hacia las antenas receptoras de la empresa con la que hemos contratado el servicio de telefonía.

²¹ http://es.wikipedia.org/wiki/Red_Telef%C3%B3nica_Conmutada

Las antenas están conectadas a una estación base que recoge todas las comunicaciones que reciban las antenas de la célula de cobertura en la que se encuentran. La estación base toma los datos que se emiten desde nuestro teléfono móvil y los procesa para establecer la comunicación con el teléfono al que se desee llamar. Esta información es introducida en la red telefónica conmutada, que procede a la conexión con las centrales digitales o analógicas asociadas al teléfono que recibe la llamada. La red telefónica alertará a la estación base de la célula en la que se encuentre el móvil con el que se quiere comunicar, procediendo a la emisión de la señal de radiofrecuencia a través de sus antenas, que captará el teléfono móvil que se ha marcado.

2.2.4 Etapa del teléfono celular receptor

La función principal de este dispositivo, es precisamente recibir el mensaje SMS. Para establecer la comunicación entre el teléfono celular receptor y el microcontrolador Atmega8A, se hace uso del cable de datos correspondiente al teléfono celular Nokia 3220, denominado USB CA-42.

2.2.4.1 Descripción del cable de datos USB CA-42

El cable en cuestión, está conformado por tres partes: 1) En uno de sus extremos se encuentra el conector que debe ir al puerto de comunicación del teléfono celular Nokia 3220; 2) El cable en sí, que en su interior posee 6 hilos conductores y, 3) En el otro extremo se tiene un conector USB (su gráfico se puede apreciar en el capítulo 1, Figura 1.9, b))

2.2.4.2 Descripción del puerto de comunicación del teléfono celular receptor

El puerto de comunicación del teléfono celular receptor está constituido por 14 pines, de los cuales se utiliza los pines 6, 7 y 8, conforme se ilustra en la siguiente figura.

En la figura 2.4, se presenta el puerto de comunicación del teléfono celular Nokia 3229, con los pines a utilizarse en esta aplicación.

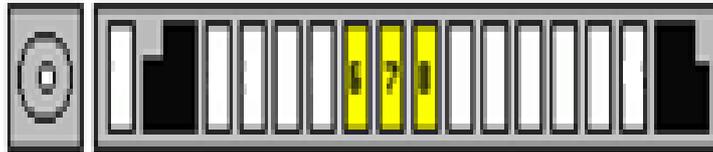


Figura 2.4: Puerto de comunicación del teléfono Nokia 3220.

2.2.4.3 Conexión y configuración del cable de datos USB CA-42 entre el teléfono celular receptor y el microcontrolador

Para realizar esta tarea, se identifica el extremo del cable en el cual se encuentra el conector que pertenece al puerto de comunicación del teléfono Nokia 3220, y se procede a conectarlo; posteriormente, en el extremo donde se encuentra el conector USB fue necesario cortarlo para poder identificar los hilos procedentes de los pines 6, 7 y 8 del puerto de comunicación, donde se encuentra:

- Rx (pin 7) corresponde al cable de color blanco.
- Tx (pin 6) corresponde al cable de color verde.
- GND (pin 8) corresponde al cable de color Naranja.

Una vez realizado la interconexión entre el teléfono celular receptor y el microcontrolador a través del cable de datos USB CA-42, los pines 6, 7 y 8 son los encargados de establecer la comunicación serial tal como se detalla a continuación:

- El pin 6 corresponde a Rx (o receptor).
- El pin 7 corresponde a Tx (o transmisor).
- El pin 8 corresponde a GND (o tierra).

Se debe anotar que para la conexión de los terminales se debe cruzar la conexión; esto quiere decir que el Tx del Nokia se debe conectar con el Rx del microcontrolador Atmega8, y viceversa.

La transferencia de datos se genera con los siguientes parámetros los cuales deberán ser configurados en el dispositivo con el cual se va a comunicar.

- Velocidad de transmisión 9600 Baudios.
- bits de datos.
- Bits de paridad "Ninguna".
- bits de para "1".

Los niveles de voltaje que utiliza este pórtico de conexión, son para el cero lógico (0L) 0 voltios, y para el uno lógico (1L) 5 voltios, es decir trabaja con niveles TTL.

En la figura 2.5, se presenta la forma en que el cable de datos USB CA-42 debe ir conectado a los respectivos dispositivos. En el extremo donde se encuentra el conector USB se debe buscar las líneas que corresponden a los pines 6, 7 y 8 para conectarlos al microcontrolador.

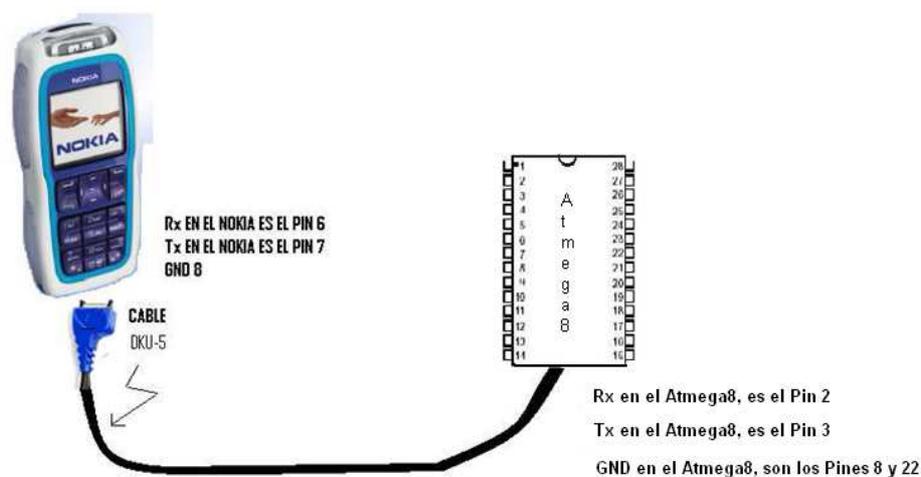


Figura 2.5: Conexión entre el teléfono NOKIA 3220 y microcontrolador Atmega8A con el cable de datos USB CA-42

2.2.5 Microcontrolador Atmega8A

Dispositivo de mayor importancia dentro del circuito, debido a que realiza la mayoría de las funciones para cumplir con el propósito general del proyecto.

2.2.5.1 Circuito de Reset

El circuito Reset es un mecanismo necesario para reiniciar el sistema en caso de que ocurra algún defecto. Este circuito reset utiliza una resistencia y un switch conectados al pin 1 (reset) del microcontrolador, funciona a manera de un disparador, colocando al pin en un nivel bajo al momento de oprimir el pulsador permitiendo un reset instantáneo. Bascom AVR recomienda utilizar una resistencia menor a 40K Ω para no violar las especificaciones técnicas de trabajo requeridas por el micro controlador. Para este proyecto se utilizó una resistencia de 4.7K Ω .

La figura 2.6, representa al circuito reset, en el cual un pulsador va conectado al pin1 (PC6) del microcontrolador a través de una resistencia la cual que limitará la corriente.

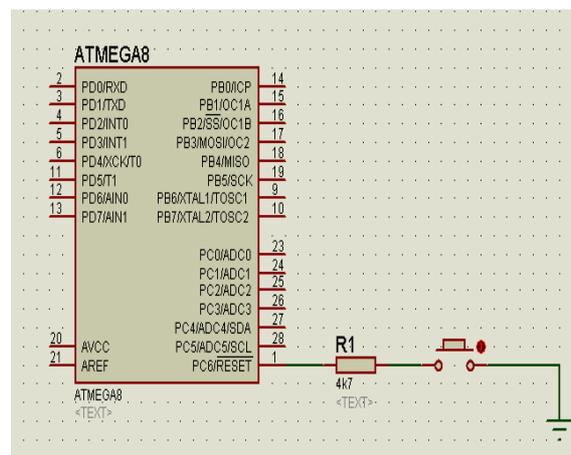


Figura N° 2.6: Circuito reset

Entonces la entrada RESET tiene como función reiniciar el estado del microcontrolador, teniendo como resultado dos acciones importantes:

- Se carga un 0L el en contador de programa, de manera que después de un Reset siempre se ejecuta la instrucción que está en la posición 0 de la memoria del programa.
- Los registros de estado y control toman un estado conocido y determinado.

2.2.5.2 Secuencia en el hardware para el envío de mensaje

Para que el sistema recpte un mensaje SMS, el hardware operará de la siguiente manera:

Una vez enviado el mensaje SMS con el código correspondiente, éste llega a su destino que es el teléfono celular receptor, para posteriormente ser descargado al microcontrolador por medio del puerto serial; luego éste activará el circuito, encendiéndose un led, notificando que una comunicación está en proceso.

La figura 2.7, representa al circuito de una comunicación en proceso; conformados por una resistencia y un led conectados al pin23 (PC0) del microcontrolador.

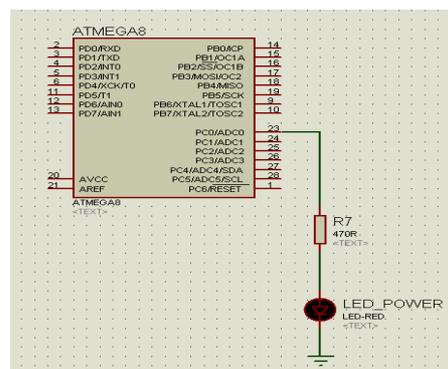


Figura 2.7: Circuito del led indicador de una comunicación en proceso.

2.2.5.3 Diseño del software utilizado en el microcontrolador Atmega8A

Una vez grabado el programa, en el microcontrolador, la tarea de decodificar los mensajes SMS y controlar las cargas, lo realiza el microcontrolador en interacción con el software y el hardware.

Durante la elaboración del programa se tomó en cuenta los siguientes pasos:

2.2.5.3.1 Desarrollo del programa para el microcontrolador Atmega8A: El programa se elaboró de manera ordenada y secuencial de acuerdo a:

2.2.5.3.1.1 Declaración de variables y configuración de registros y puertos:

Declaración de variables y etiquetas:

- Declarar todas las variables que se van a utilizar en el programa.
- Declarar todas las etiquetas que se utilizarán en el programa.
- Fin tarea.

Configuración de registros y puertos:

- Configurar los registros que se utilizarán en el programa.
- Configurar los puertos del microcontrolador, ya sea como salidas o entradas.
- Configurar los puertos de comunicación del microcontrolador.
- Fin de tarea.

Limpieza de registros y puertos:

- Limpiar todos los registros que se utilizarán en el programa
- Fin de tarea

2.2.5.3.1.2 Hacer llamada al celular receptor: Mediante los comandos AT se realiza ésta operación, los mismos que son transmitidos desde el microcontrolador.

Hacer llamada al celular receptor:

- Transmitir el comando AT que ejecuta una llamada de voz (ATD15).

- Bajar la bandera del habilitador a interrupción externa antes de salir de la interrupción, con esto se mantiene lista la interrupción para una nueva activación.
- Regresar a la siguiente instrucción desde donde se saltó a la rutina de atención a la interrupción externa.
- Fin de tarea

2.2.5.3.1.3 Monitoreo de un nuevo mensaje en el celular receptor:

Esta parte del programa se encarga de revisar continuamente el arribo de un nuevo mensaje al celular receptor, haciendo uso de los comandos AT.

Monitorear nuevo mensaje en el celular receptor.

- Transmitir el comando que chequea si existe un mensaje nuevo (AT+CNMI?).
- Si existe un nuevo mensaje el celular responde (+CNMI: <mode>, N° cel, fecha, hora), CLAVE, OK.
- Copiar los datos contenidos en el mensaje en un espacio de la memoria RAM y compara que clave está contenida en el mensaje, para su ejecución.
- Si no existe nuevo mensaje el celular responde (OK).
- Comparar y retornar a la rutina para seguir el monitoreo de nuevo mensaje en el celular receptor.
- Fin de tarea

2.2.5.3.1.4 Decodificar el SMS y verificar qué tipo de comando contiene: En esta tarea se determina el comando que contiene el mensaje escrito, para su posterior ejecución:

Decodificar el SMS y verificar qué tipo de comando contiene.

- Comparar los datos del mensaje guardado en la memoria RAM con los comandos asignados.

- Fin de tarea.

2.2.5.3.1.5 Ejecutar activación de cargas:

- Activar el puerto de salida para activación de carga.
- Enviar el mensaje de confirmación de activación de carga.
- Retornar al monitoreo de nuevo mensaje en el celular receptor
- Fin de tarea.

2.2.5.3.1.6 Ejecutar desactivación de cargas:

- Ejecuta desactivación de carga.
- Activar el puerto de salida para desactivar carga.
- Enviar el mensaje de confirmación de desactivar carga.
- Retornar al monitoreo de nuevo mensaje en el celular receptor.
- Fin de tarea.

2.2.5.3.1.7 Comando no válido:

- Enviar el mensaje notificando un error en el mensaje recibido.
- Retornar al monitoreo de nuevo mensaje en el celular receptor.
- Fin de tarea.

2.2.5.3.2 Comandos AT utilizados

Los comandos AT que fueron necesarios utilizar en la comunicación del microcontrolador Atmega8A son:

2.2.5.3.2.1 Comandos de llamada

- ATD: [Dial Command]
Inicia una llamada telefónica.

Sintaxis:

ATD0969269568 para una llamada de Datos.

ATD0969269568; para una llamada de Voz. (Importante la notación ';')

ATD>"Gospel"; para llamar al contacto almacenado en la agenda con el texto asociado Gospel.

En cuanto se envía el comando ATD0969269568; el celular responde con un OK y ejecuta la llamada al número 0969269568, en cuando termina la llamada el celular receptor responde con un NO CARRIER, este dato es utilizado en el software para salir de la subrutina (hacer llamada).

- AT+CHUP: Colgar llamada

Este comando cuelga la llamada, también cuando un modo de llamar alterna está activo.

Syntaxis:

AT+CHUP Colgar llamada. La respuesta es NO CARRIER.

2.2.5.3.2.2 Comandos de SMS

- AT+CMGF: Seleccionar formato de los mensajes SMS.

Syntaxis:

AT+CMGF=[<mode>] Seleccionar el formato de mensaje.

AT+CMGF? Configuración actual de la consulta.

Response is +CMGF: <modo>.

AT+CMGF=? Mostrar los valores admitidos.

Response is +CMGF: (lista de los <modo>s).

Valores por<modo>:

0 Modo PDU

1 Modo Texto

- AT+CNMI: Nuevas indicaciones en el mensaje a los DTE

Este comando selecciona la forma en la recepción de nuevos mensajes de la red.

Syntax:

AT+CNMI=[<modo>[,<mt>[,<bm>[,<ds>[,<bfr>]]]]] Seleccione procedimiento de indicación.

AT+CNMI? Configuración actual de la consulta.

Response is +CNMI:<mode>,<mt>,<bm>,<ds>,<bfr>.

AT+CNMI=? Mostrar los valores válidos.

Response is +CNMI: (listas de los <modo>s), (<mt>s), (<bm>s), (<ds>s),(<bfr>s).

2.2.5.4 Funciones entre teléfono celular receptor y microcontrolador

Una vez realizado las conexiones apropiadas y grabado el programa en el microcontrolador, éste debe realizar las siguientes tareas:

- El microcontrolador debe establecer un enlace con el celular receptor por medio del pòrtico serial.
- Una vez establecida la comunicación entre los dos dispositivos, el microcontrolador transmitirá los caracteres que forman los comandos AT.

El microcontrolador siempre estará monitoreando el arribo de un nuevo mensaje al celular receptor. Si llega un nuevo mensaje, este es enviado al microcontrolador para ser decodificado y ejecutado dependiendo del contenido del mensaje. Si el contenido del mensajes descargado no corresponde a ninguna operación, éste será ignorado o simplemente lo recibirá y lo guardará en su memoria sin realizar ninguna acción dentro del circuito.

Dependiendo del comando AT recibido, el microcontrolador ejecutará:

- Envía un código de llamada; esta llamada sirve para verificar que la comunicación entre el microcontrolador y el teléfono celular receptor es correcta.
- Finaliza código de llamada.
- Configuración de los mensajes SMS
- El teléfono celular receptor debe enviar los mensajes al microcontrolador por intermedio del puerto serial.
- El puerto de salida PC0 se activa para encender un led, indicando que el microcontrolador está listo para recibir los mensajes.

2.2.5.4 Principales funciones del microcontrolador Atmega8A

- Descargar y leer el contenido de los mensajes recibidos en el celular receptor para ejecutarlos, dependiendo del texto que este contenga. Varias instrucciones.
- Dar una señal para la activación o desactivación de los relés.
- Activar la carga correspondiente al relé activado.

2.2.5.5 Puertos y pines del microcontrolador Atmega8A utilizados

Los puertos o pines del microcontrolador Atmega8A durante la elaboración del circuito son utilizados de la siguiente manera:

- El puerto C (PC6) es utilizado como entrada o reset.
- El puerto serial D (PD0_Rx, PD1_Tx), para la comunicación con el teléfono celular; esto es: PD0, permite la recepción de datos y PD1 permite la transmisión de datos.
- El pin 7 que corresponde a Vcc, es utilizado para alimentar al circuito con un voltaje de 3.8V.
- El pin 8 corresponde a GND.

- El puerto B (PB0, PB1, PB2), asignado como puerto de salida, son utilizados para encender tres diferentes leds que nos indican que relé entra en función.
- El puerto B asignado como puerto de salida (PB3, PB4, PB5) para el control de las cargas.
- Los puerto de salida C (PC0)que activará un led para indicar que la comunicación está en proceso.

En la tabla 2.2, se resume los pines utilizados , nombre y función.

Tabla 2.2: Nombre y función de los pines utilizados en el microcontrolador

PINES	NOMBRE	FUNCIÓN
7, 8	VCC, GND	Fuente.
1	PC6	Reset.
2	PD0	Recepción.
3	PD1	Transmisión.
14	PB0	Salida para encender led indicador del relé1.
15	PB1	Salida para encender led indicador del relé2.
16	▪ ▪ ▪ 辦Salida	Salida para encender led indicador del relé3.
17	PB3	Control de carga1.
18	PB4	Control de carga2.
19	PB5	Control de carga3.
23	PC0	Led indicador de comunicación en proceso.

2.2.6 Etapa de optoacoplamiento

Para alimentar la parte de potencia, se necesita de un voltaje y corriente superiores a los requeridos para alimentar la parte de control, entonces la función de la etapa de optoacoplamiento es aislar las dos partes, tomando en cuenta que el teléfono celular receptor y el microcontrolador (que conforman la

parte de control), pueden sufrir daños con voltajes y corrientes que superen el rango par el cual está diseñado.

2.2.6.1 Circuito para la activación de tres leds

El microcontrolador activará los puertos de salida PB0, PB1 y PB2 para encender 3 diferentes leds, cada uno de los cuales indicarán las siguientes funciones: Al activar el puerto de salida PB0 se enciende el led 1, el mismo que nos indica que la orden que da el microcontrolador es correcta y por lo tanto el transistor que corresponde a este puerto entrará en saturación.

La figura 2.8, representa a un circuito para la activación de tres leds; esto significa que cuando enviamos desde el teléfono celular emisor los mensajes SMS, se encenderá los leds, dependiendo del código del mensaje.

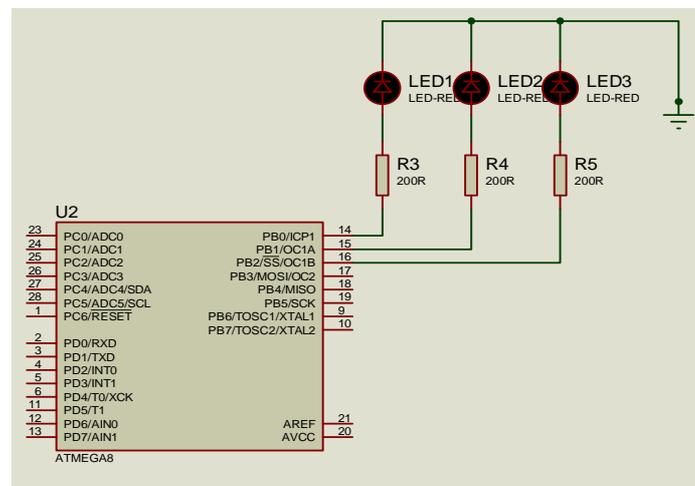


Figura N° 2.8: Circuito para la activación de tres leds

De manera similar se procede para los puertos de salida PB1 y PB2.

Para limitar la corriente máxima entre los puertos de salida PB0, PB1 Y PB2 y los respectivos leds, se hace uso de una resistencia para cada uno de los casos. Esta resistencia la calculamos de la siguiente manera:

Datos del fabricante del microcontrolador:

$V = 5$ voltios (voltaje máximo); pero se alimenta con $V_{cc} = 3.8V$

$I = 25$ mA (corriente máxima); pero tomamos $I = 10mA$

Caída de voltaje en el diodo es:

$$V_d = 1.8V$$

Cálculos:

$$R = \frac{V_{cc} - V_d}{I};$$

$$R = \frac{3.8V - 1.8V}{10mA}$$

$$R = 200\Omega$$

Esto significa que resistencias R1, R2 y R3 tienen el mismo valor que es de 200Ω .

2.2.6.2 Circuito para la activación/desactivación de las cargas

Para realizar esta tarea, se ha utilizado los puertos de salida del microcontrolador PB3, PB4 y PB5, de la siguiente manera: Cuando el puerto de salida PB3 esté en 1L, el transistor 1 entra en saturación y consecuentemente los contacto del relé cambian de posición para activar el transistor 1 y, así se activa la carga correspondiente, que en nuestro caso le asignamos una “lampara incadensecente”.

De igual manera es el caso de los puertos de salida PB4 y PB5, variando únicamente su carga que en el caso de PB4 es una “alarma” y, para el puerto PB5 es un “interruptor eléctrico”.

Entre los puertos PB3, PB4, PB5 y los respectivos transistores, la corriente máxima se limita haciendo uso de una resistencia. El valor de dichas resistencias, la calculamos de la siguiente manera:

Datos del transistor:

$$I_c = 0.1A = 100mA$$

$$\beta = 50 \text{ (considerando el transistor en saturación)}$$

$$V_{cc} = 3.8V$$

Cálculos:

$$I_b = \frac{I_c}{\beta}$$

$$I_b = \frac{100mA}{50}$$

$$I_b = 2mA$$

$$R = \frac{V_{cc} - V_{be}}{I_b}, \text{ donde } V_{cc} \text{ es el voltaje de salida del microcontrolador.}$$

$$R = \frac{3.8V - 0.6V}{2mA}$$

$$R = 1600\Omega$$

Reemplazamos por:

$$R = 1500\Omega$$

Esta resistencia es igual para los puertos PB3, PB4, PB5 y sus respectivos transistores.

El circuito, queda de la siguiente manera:

- La BASE, va conectado al puerto de salida PB3 del microcontrolador por intermedio de la resistencia calculada anteriormente.

- El COLECTOR va conectado al pin 2 del relé, en paralelo con un diodo 1N4007, posteriormente de los pines 3 y 5 del relé hemos hecho uso de una bornera a la cual van conectados los terminales de alimentación para la respectiva carga.
- El EMISOR va directo a tierra.

La figura 2.9, representa al circucito para la activación/desactivación de las cargas, lo que significa que la carga se activará previo al encendido del led correspondiente a dicha carga.

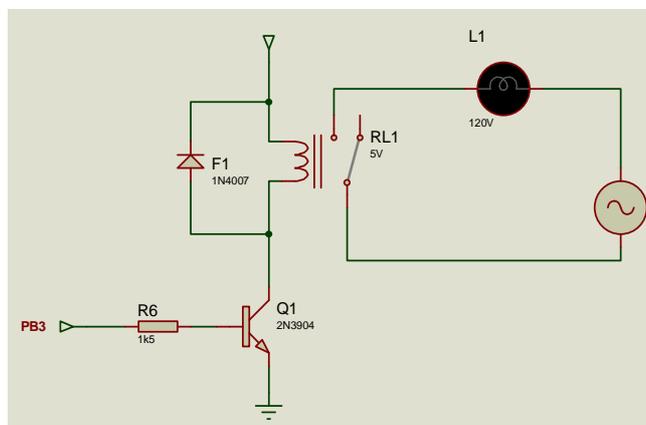


Figura 2.9: Circuito para la activación/desactivación de las cargas

Los pines sobrantes del relé de SRD-06VDC-SL-C van conectados conforme se indica a continuación:

- El pin 1 con conectado a Vcc.
- El pin 4 queda suelto.

Este proceso se repite para los otros dos transistores y relés.

2.2.7 Etapa de activación

Despues de un minucioso trabajo se llega a la etapa de activación, la cual consiste en comprobar el funcionamiento de cada una de las cargas.

2.2.7.1 Activación del foco

Este dispositivo se encuentra alimentado con un voltaje de 110 voltios, pero para manipular su funcionamiento (encender/apagar), en una polaridad del cable de alimentación del foco, conectamos a una bornera, la misma está comunicada con el relé encargado de realizar esta tarea. Al enviar desde el teléfono celular emisor un mensaje SMS con el contenido de texto de ON1, el foco se encenderá; mientras que si el contenido del texto del mensaje SMS es OF1, se apagará.

2.2.7.2 Activación de la alarma

De manera similar al caso del foco, la alarma se encuentra alimentada con un voltaje de 110 voltios y una polaridad del cable de alimentación lo conectamos a una bornera, la cual que se encuentra comunicada con su respectivo relé; entonces al activarse o desactivarse dicho relé, realizará la misma función en la alarma. Al enviar desde el teléfono celular emisor un mensaje SMS con el contenido de texto de ON2, la alarma se activará; mientras que si el contenido del texto es OF2, la alarma se desactivará.

2.2.7.3 Activación del interruptor

El proceso de alimentación y conexión a la bornera es igual que en los dos casos anteriores. Para que se active el interruptor, desde el teléfono celular emisor debe ser enviado un mensaje SMS con el contenido de texto de ON3; y, para que se desactive, desde el teléfono celular emisor debe ser enviado un mensaje SMS con el texto OF3. Cabe anotar que en el momento de realizar las pruebas correspondiente para la activación de las cargas, para el caso de la alarma y el interruptor eléctrico primeramente se lo hizo con un foco, con el propósito de poder comprobar a través del encendido del foco que la señal que da el microcontrolador llega a su destino, obteniendo así un resultado positivo; posteriormente se coloca las cargas reales y, de la misma manera se comprobó su correcto funcionamiento.

2.2.8 Circuito general del proyecto

Una vez analizado y comprendido el funcionamiento de las diferentes etapas, y dentro de ellas los respectivos elementos por los cuales estaría conformado el circuito, se procedió a elaborar el esquema del circuito general del proyecto, el mismo que está diseñado en el programa denominado PROTEUS, siguiendo las normas o reglas a cumplirse para obtener un circuito óptimo. El PROTEUS es un programa fácil de manejar; sin embargo, es importante tomar en cuenta las características que deben tener de cada uno de los elementos que conforman el circuito, caso contrario en el momento de simular o como resultado final se puede obtener resultados diferentes al buscado, lo que indudablemente no cumpliría con el propósito del proyecto.

En la figura 2.10, representa la forma como queda elaborado el circuito general del proyecto con sus respectivos elementos.

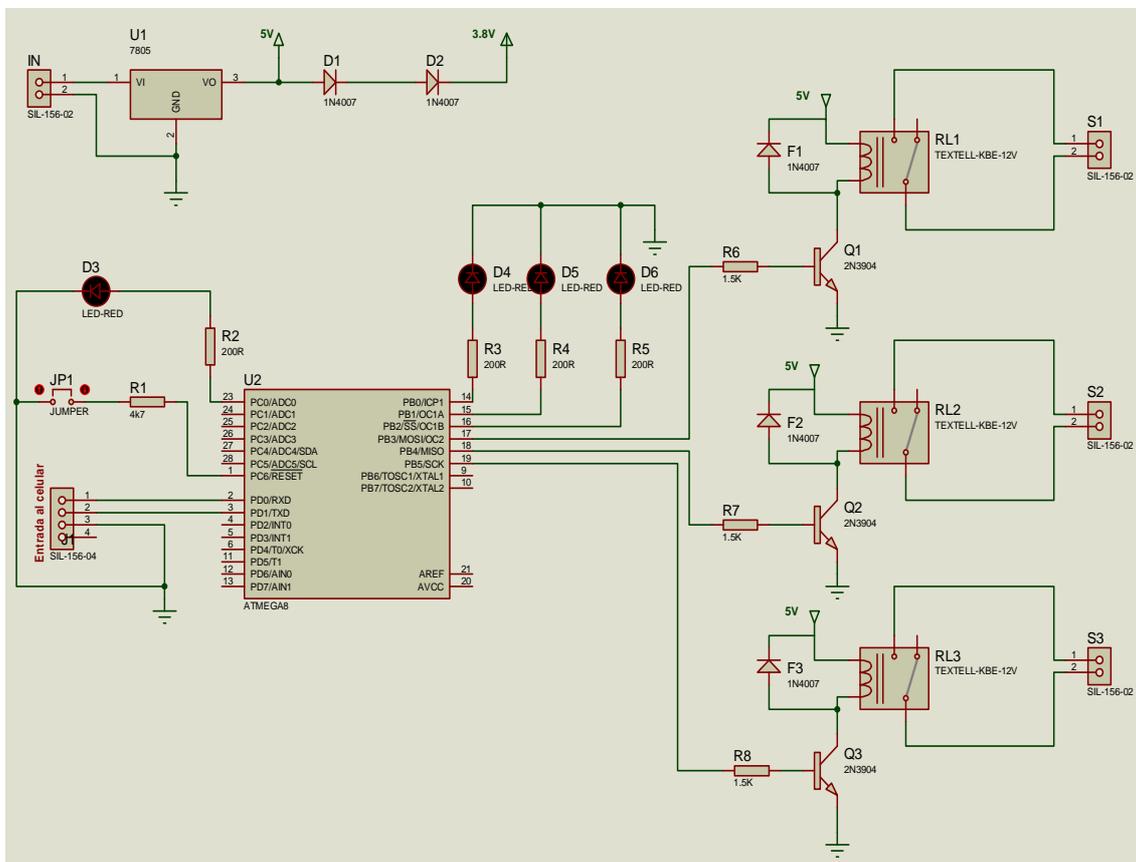


Figura 2.10: Circuito general del proyecto

CAPÍTULO 3

CONCLUSIONES Y RECOMENDACIONES

3.1 CONCLUSIONES

- Mediante este proyecto se realizó la construcción de un circuito de control para tres cargas mediante mensajes SMS (Servicio para Mensajes Cortos), emitidas desde un teléfono celular, utilizando el microcontrolador Atmega8A.
- Se explicó de manera detallada los objetivos específicos para los cuales fue planteado el proyecto; esto es: 1) Para la comunicación entre el teléfono celular (receptor) y el microcontrolador, se utilizó el cable de datos denominado USB CA-42. 2) La comunicación que se da entre los dispositivos es la Comunicación Serial. 3) Sus etapas fueron descritas a través de un diagrama de bloques. 4) Los dispositivos inmersos en el circuito fueron seleccionados de acuerdo a sus características eléctricas y electrónicas, especialmente los rangos de voltaje y corriente para los cuales fueron diseñados. 5) En la influencia del medio ambiente para el funcionamiento del circuito, se utiliza la transmisión de ondas electromagnéticas.
- El microcontrolador Atmega8A, fue seleccionado de acuerdo a las necesidades del proyecto y características del microcontrolador. Se consideró principalmente los puertos de entrada/salida y su capacidad de memoria interna para la programación, evitando así hacer uso de un microcontrolador inadecuado.
- El circuito ha sido alimentado con dos voltajes diferentes (5 y 3.8 voltios), tomando en cuenta las propiedades eléctricas y electrónicas de los

diferentes dispositivos que intervienen en él. Para controlar los voltajes y corrientes, se hizo haciendo uso de un multímetro digital.

- Se consiguió que el microcontrolador envía la señal digital y datos correspondiente a la función que se desee realizar.
- Se realizaron las pruebas necesarias; esto consiste principalmente en el envío de mensajes SMS, los cuales serán interpretados por el microcontrolador dependiendo de su contenido de texto, para posteriormente realizar la acción correspondiente, esto es activar/desactivar las cargas. Se concluye que éste cumple con su propósito general del proyecto.
- Al iniciar el funcionamiento del circuito, es necesario activar el circuito Reset, pues siempre existe algún defecto proveniente de la comunicación entre los dispositivos.
- Para la elaboración del circuito impreso, se utilizó el programa denominado PROTEUS, el cual es un programa fácil de manipular y además provee toda la información que se requiere para diseñar los circuitos electrónicos; sin embargo, se tomó muy en cuenta las características de cada uno de los elementos para poder obtener un resultado eficiente.

3.2 RECOMENDACIONES

- Como se mencionó, el proyecto cumple a cabalidad con el objetivo general, pero esto no significa que éstas sean sus únicas aplicaciones; dependiendo de la programación, se puede hacer uso en: 1) Control de las instalaciones eléctricas de una empresa. 2) Programar al microcontrolador para controlar las seguridades de nuestros hogares y a la vez dar un aviso de alerta en caso de que algún intruso opte por violar las seguridades, etc. 3) Ordenar a un robot las funciones que debe cumplir en alguna actividad previamente organizada.
- El momento de realizar la programación, es aconsejable hacerlo en un lenguaje de programación fácil y que el software necesario para grabar el programa en el microcontrolador se pueda conseguir en el mercado.
- Al elaborar el circuito es muy importante tener en cuenta los rangos de voltaje y corriente de los dispositivos de igual manera su polaridad.
- Tomar en cuenta la vida útil de las cargas (foco, alarma e interruptor) ya que puede darse el caso que debido al daño de uno de estos dispositivos no funcione el circuito.
- Este proyecto por ser económico y por tener mucha importancia en su aplicación, es recomendable que los ciudadanos lo incorporen en sus propiedades, sean estos: sus hogares, empresas, automóviles, etc.
- Es recomendable estar pendiente de la batería del teléfono celular receptor; esto es, cuando se descargue la batería reemplazarla por otra de manera inmediata o a su vez adaptarla con para que su alimentación sea continua, evitando así que el circuito deje de funcionar.

3.3. BIBLIOGRAFÍA

- a) Circuitos Electrónicos, Thomas L. Floyd, Tomo I.
- b) Sistemas Digitales, Principios y Aplicaciones, Ronald J. Tocci, Quinta Edición.
- c) Prácticas con microcontroladores, aplicaciones Industriales, Javier Martínez Pérez, Mariano Barrón Ruiz.
- d) http://es.wikipedia.org/wiki/Telefon%C3%ADa_m%C3%B3vil
- e) http://es.wikipedia.org/wiki/Sistema_global_para_las_comunicaciones_m%C3%B3viles
- f) http://www.google.com.ec/Nºhl=es&gs_rn=9&gs_ri=psy-ab&pq=tecnologia%20gsm&cp=14&gs_id=3c&xhr=t&q=comunicaci%C3%B3n+serial&es_nrs=true&pf=p&scient=psy-ab&oq=comunicacion+se&gs_l=&pbx=1&bav=on.2,or.r_qf.&fp=29fb39727d8b4a2&biw=1366&bih=677
- g) <http://es.wikipedia.org/wiki/Microcontrolador>
- h) http://es.wikipedia.org/wiki/Transistor_de_uni%C3%B3n_bipolar

ANEXO A

DATA SHEET DEL MICROCONTROLADOR

ATmega8A

8-bit with 8K Bytes In-System Programmable Flash ATmega8A Summary

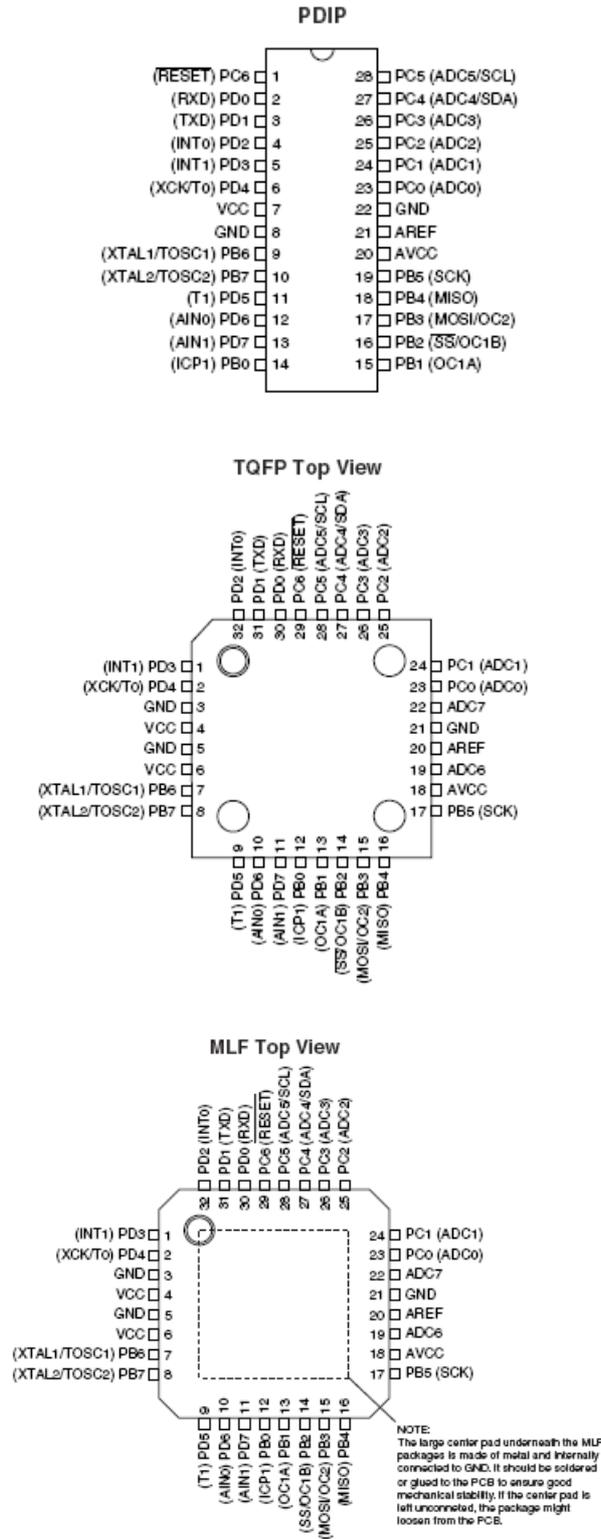
Features

- High-performance, Low-power AVR® 8-bit Microcontroller
- Advanced RISC Architecture
 - 130 Powerful Instructions – Most Single-clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 16 MIPS Throughput at 16 MHz
 - On-chip 2-cycle Multiplier
- High Endurance Non-volatile Memory segments
 - 8K Bytes of In-System Self-programmable Flash program memory
 - 512 Bytes EEPROM
 - 1K Byte Internal SRAM
 - Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
 - Data retention: 20 years at 85°C/100 years at 25° C⁽¹⁾
 - Optional Boot Code Section with Independent Lock Bits
- In-System Programming by On-chip Boot Program
- True Read-While-Write Operation
 - Programming Lock for Software Security
- Peripheral Features
 - Two 8-bit Timer/Counters with Separate Prescaler, one Compare Mode
 - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
 - Real Time Counter with Separate Oscillator
 - Three PWM Channels
 - 8-channel ADC in TQFP and QFN/MLF package
- Eight Channels 10-bit Accuracy
 - 6-channel ADC in PDIP package
- Six Channels 10-bit Accuracy
 - Byte-oriented Two-wire Serial Interface
 - Programmable Serial USART
 - Master/Slave SPI Serial Interface
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparator
- Special Microcontroller Features
 - Power-on Reset and Programmable Brown-out Detection
 - Internal Calibrated RC Oscillator
 - External and Internal Interrupt Sources
 - Five Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, and Standby
- I/O and Packages
 - 23 Programmable I/O Lines
 - 28-lead PDIP, 32-lead TQFP, and 32-pad QFN/MLF
- Operating Voltages
 - 2.7 - 5.5V for ATmega8A
- Speed Grades
 - 0 - 16 MHz for ATmega8A
- Power Consumption at 4 Mhz, 3V, 25°C
 - Active: 3.6 mA
 - Idle Mode: 1.0 mA
 - Power-down Mode: 0.5 µA

ATmega8A

1. Pin Configurations

Figure 1-1. Pinout ATmega8A

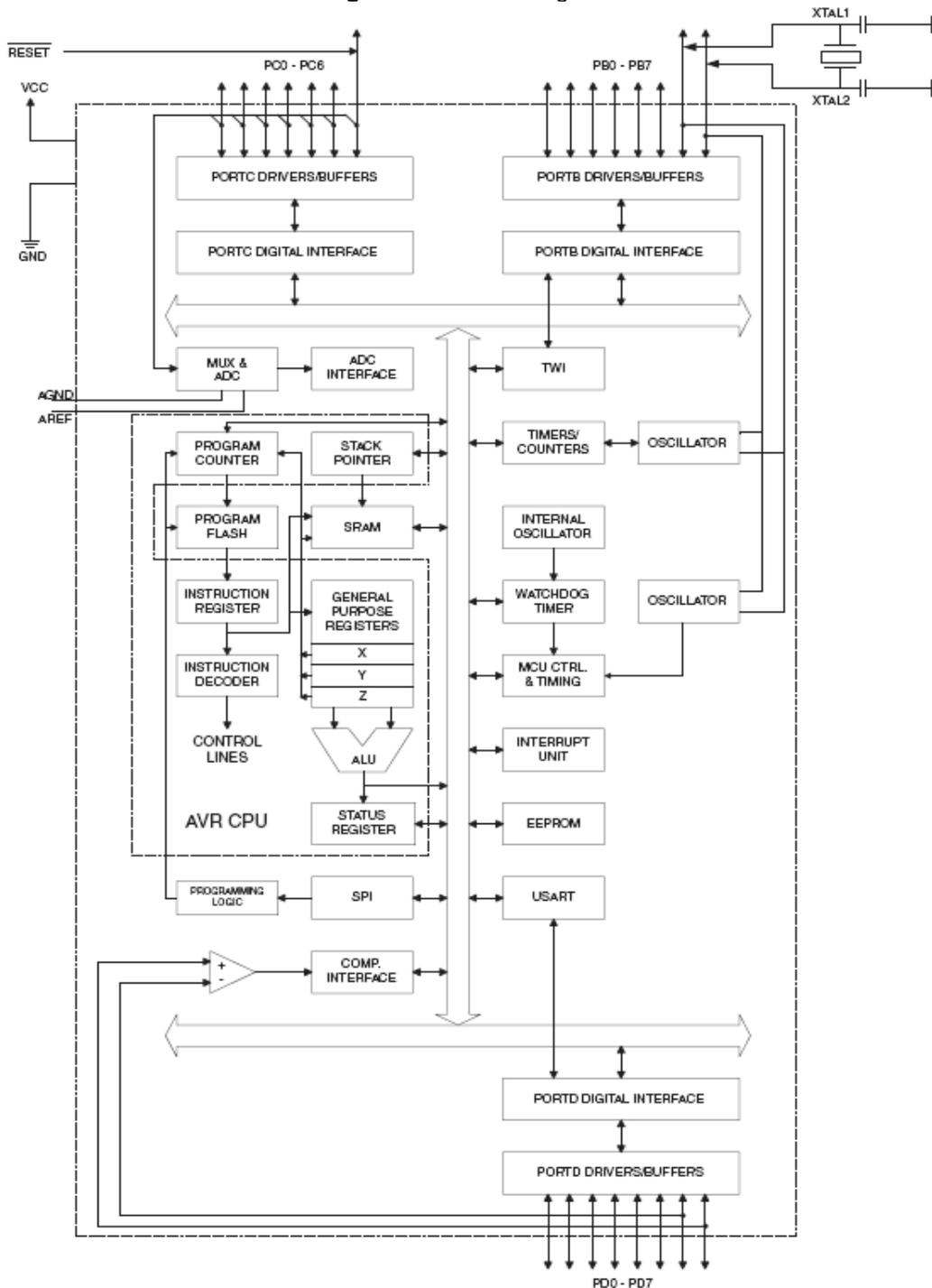


2. Overview

The ATmega8A is a low-power CMOS 8-bit microcontroller based on the AVR RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega8A achieves throughputs approaching 1 MIPS per MHz, allowing the system designer to optimize power consumption versus processing speed.

2.1 Block Diagram

Figure 2-1. Block Diagram



ATmega8A

The AVR core combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers.

The ATmega8A provides the following features: 8K bytes of In-System Programmable Flash with Read-While-Write capabilities, 512 bytes of EEPROM, 1K byte of SRAM, 23 general purpose I/O lines, 32 general purpose working registers, three flexible Timer/Counters with compare modes, internal and external interrupts, a serial programmable USART, a byte oriented Two-wire Serial Interface, a 6-channel ADC (eight channels in TQFP and QFN/MLF packages) with 10-bit accuracy, a programmable Watchdog Timer with Internal Oscillator, an SPI serial port, and five software selectable power saving modes. The Idle mode stops the CPU while allowing the SRAM, Timer/Counters, SPI port, and interrupt system to continue functioning. The Power-down mode saves the register contents but freezes the Oscillator, disabling all other chip functions until the next Interrupt or Hardware Reset. In Power-save mode, the asynchronous timer continues to run, allowing the user to maintain a timer base while the rest of the device is sleeping. The ADC Noise Reduction mode stops the CPU and all I/O modules except asynchronous timer and ADC, to minimize switching noise during ADC conversions. In Standby mode, the crystal/resonator Oscillator is running while the rest of the device is sleeping. This allows very fast start-up combined with low-power consumption.

The device is manufactured using Atmel's high density non-volatile memory technology. The Flash Program memory can be reprogrammed In-System through an SPI serial interface, by a conventional non-volatile memory programmer, or by an On-chip boot program running on the AVR core. The boot program can use any interface to download the application program in the Application Flash memory. Software in the Boot Flash Section will continue to run while the Application Flash Section is updated, providing true Read-While-Write operation. By combining an 8-bit RISC CPU with In-System Self-Programmable Flash on a monolithic chip, the Atmel ATmega8A is a powerful microcontroller that provides a highly-flexible and cost-effective solution to many embedded control applications.

The ATmega8A AVR is supported with a full suite of program and system development tools, including C compilers, macro assemblers, program debugger/simulators, In-Circuit Emulators, and evaluation kits.

2.2 Pin Descriptions

2.2.1 VCC Digital supply voltage.

2.2.2 GND Ground.

2.2.3 Port B (PB7:PB0) – XTAL1/XTAL2/TOSC1/TOSC2

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

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

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

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

The various special features of Port B are elaborated in "Alternate Functions of Port B" on page

58 and “System Clock and Clock Options” on page 24.

2.2.4 Port C (PC5:PC0)

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

2.2.5 PC6/RESET

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

If the RSTDISBL Fuse is unprogrammed, PC6 is used as a Reset input. A low level on this pin for longer than the minimum pulse length will generate a Reset, even if the clock is not running. The minimum pulse length is given in Table 25-3 on page 247. Shorter pulses are not guaranteed to generate a Reset. The various special features of Port C are elaborated.

2.2.6 Port D (PD7:PD0)

Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running. Port D also serves the functions of various special features of the ATmega8A as listed.

2.2.7 RESET

Reset input. A low level on this pin for longer than the minimum pulse length will generate a reset, even if the clock is not running. The minimum pulse length is given in Table 25-3 page 247. Shorter pulses are not guaranteed to generate a reset.

2.2.8 AVcc

AVcc is the supply voltage pin for the A/D Converter, Port C (3:0), and ADC (7:6). It should be externally connected to Vcc, even if the ADC is not used. If the ADC is used, it should be connected to Vcc through a low-pass filter. Note that Port C (5:4) use digital supply voltage, Vcc.

2.2.9 AREF

AREF is the analog reference pin for the A/D Converter.

2.2.10 ADC7:6 (TQFP and QFN/MLF Package Only)

In the TQFP and QFN/MLF package, ADC7:6 serve as analog inputs to the A/D converter. These pins are powered from the analog supply and serve as 10-bit ADC channels.

3. Resources

A comprehensive set of development tools, application notes and datasheets are available for download on <http://www.atmel.com/avr>.

4. Data Retention

Reliability Qualification results show that the projected data retention failure rate is much less than 1 PPM over 20 years at 85°C or 100 years at 25 °C.

5. Register Summary

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Page
0x3F (0x6F)	SREG	I	T	H	S	V	N	Z	C	8
0x3E (0x6E)	SPH	-	-	-	-	-	SP10	SP9	SP8	11
0x3D (0x6D)	SPL	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	11
0x3C (0x6C)	Reserved									
0x3B (0x6B)	GICR	INT1	INT0	-	-	-	-	IVSEL	IVCE	48, 68
0x3A (0x6A)	GIFR	INTF1	INTF0	-	-	-	-	-	-	69
0x39 (0x69)	TIMSK	OCIE2	TOIE2	TICIE1	OCIE1A	OCIE1B	TOIE1	-	TOIE0	73, 104, 124
0x38 (0x68)	TIFR	OCF2	TOV2	ICF1	OCF1A	OCF1B	TOV1	-	TOV0	74, 104, 104
0x37 (0x67)	SPMCR	SPMIE	RWWSB	-	RWWSRE	BLBSET	PGWRT	PGERS	SPMEN	224
0x36 (0x66)	TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE	191
0x35 (0x65)	MCUCR	SE	SM2	SM1	SM0	ISC11	ISC10	ISC01	ISC00	38, 67
0x34 (0x64)	MCUCSR	-	-	-	-	WDRF	BORF	EXTRF	PORF	43
0x33 (0x63)	TCCR0	-	-	-	-	-	CS02	CS01	CS00	73
0x32 (0x62)	TCNT0	Timer/Counter0 (8 Bits)								73
0x31 (0x61)	OSCCAL	Oscillator Calibration Register								31
0x30 (0x60)	SFIOR	-	-	-	-	ACME	PUD	PSR2	PSR10	57, 77, 126, 196
0x2F (0x4F)	TCCR1A	COM1A1	COM1A0	COM1B1	COM1B0	FOC1A	FOC1B	WGM11	WGM10	99
0x2E (0x4E)	TCCR1B	ICNC1	ICES1	-	WGM13	WGM12	CS12	CS11	CS10	101
0x2D (0x4D)	TCNT1H	Timer/Counter1 – Counter Register High byte								102
0x2C (0x4C)	TCNT1L	Timer/Counter1 – Counter Register Low byte								102
0x2B (0x4B)	OCR1AH	Timer/Counter1 – Output Compare Register A High byte								103
0x2A (0x4A)	OCR1AL	Timer/Counter1 – Output Compare Register A Low byte								103
0x29 (0x49)	OCR1BH	Timer/Counter1 – Output Compare Register B High byte								103
0x28 (0x48)	OCR1BL	Timer/Counter1 – Output Compare Register B Low byte								103
0x27 (0x47)	ICR1H	Timer/Counter1 – Input Capture Register High byte								103
0x26 (0x46)	ICR1L	Timer/Counter1 – Input Capture Register Low byte								103
0x25 (0x45)	TCCR2	FOC2	WGM20	COM21	COM20	WGM21	CS22	CS21	CS20	121
0x24 (0x44)	TCNT2	Timer/Counter2 (8 Bits)								123
0x23 (0x43)	OCR2	Timer/Counter2 Output Compare Register								123
0x22 (0x42)	ASSR	-	-	-	-	AS2	TCN2UB	OCR2UB	TCR2UB	123
0x21 (0x41)	WDTCSR	-	-	-	WDCE	WDE	WDP2	WDP1	WDP0	43
0x20 ⁽¹⁾ (0x40 ⁽¹⁾)	UBRRH	URSEL	-	-	-	-	UBRR[11:8]			160
	UCSRC	URSEL	UMSEL	UPM1	UPM0	USBS	UCSZ1	UCSZ0	UCPOL	159
0x1F (0x3F)	EEARH	-	-	-	-	-	-	-	EEAR8	19
0x1E (0x3E)	EEARL	EEAR7	EEAR6	EEAR5	EEAR4	EEAR3	EEAR2	EEAR1	EEAR0	19
0x1D (0x3D)	EEDR	EEPROM Data Register								19
0x1C (0x3C)	EEDR	-	-	-	-	EERIE	EEMWE	EEWE	EERE	19
0x1B (0x3B)	Reserved									
0x1A (0x3A)	Reserved									
0x19 (0x39)	Reserved									
0x18 (0x38)	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	66
0x17 (0x37)	DDRB	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	66
0x16 (0x36)	PINB	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	66
0x15 (0x35)	PORTC	-	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0	66
0x14 (0x34)	DDRC	-	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0	66
0x13 (0x33)	PINC	-	PINC6	PINC5	PINC4	PINC3	PINC2	PINC1	PINC0	66
0x12 (0x32)	PORTD	PORTD7	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	66
0x11 (0x31)	DDRD	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	66
0x10 (0x30)	PIND	PIND7	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	66
0x0F (0x2F)	SPDR	SPI Data Register								135
0x0E (0x2E)	SPSR	SPIF	WCOL	-	-	-	-	-	SPI2X	134
0x0D (0x2D)	SPCR	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0	133
0x0C (0x2C)	UDR	USART I/O Data Register								156
0x0B (0x2B)	UCSRA	RXC	TXC	UDRE	FE	DOR	PE	U2X	MPCM	157
0x0A (0x2A)	UCSRB	RXCIE	TXCIE	UDRIE	RXEN	TXEN	UCSZ2	RXB8	TXB8	158
0x09 (0x29)	UBRRL	USART Baud Rate Register Low byte								160
0x08 (0x28)	ACSR	ACD	ACBG	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	196
0x07 (0x27)	ADMUX	REFS1	REFS0	ADLAR	-	MUX3	MUX2	MUX1	MUX0	208
0x06 (0x26)	ADCSRA	ADEN	ADSC	ADFR	ADIF	ADIE	ADPS2	ADPS1	ADPS0	209
0x05 (0x25)	ADCH	ADC Data Register High byte								210
0x04 (0x24)	ADCL	ADC Data Register Low byte								210
0x03 (0x23)	TWDR	Two-wire Serial Interface Data Register								193
0x02 (0x22)	TWAR	TWA6	TWA5	TWA4	TWA3	TWA2	TWA1	TWA0	TWGE	194
0x01 (0x21)	TWSR	TWS7	TWS6	TWS5	TWS4	TWS3	-	TWPS1	TWPS0	193
0x00 (0x20)	TWBR	Two-wire Serial Interface Bit Rate Register								191

6. Instruction Set Summary

Mnemonics	Operands	Description	Operation	Flags	#Clocks
ARITHMETIC AND LOGIC INSTRUCTIONS					
ADD	Rd, Rr	Add two Registers	$Rd \leftarrow Rd + Rr$	Z,C,N,V,H	1
ADC	Rd, Rr	Add with Carry two Registers	$Rd \leftarrow Rd + Rr + C$	Z,C,N,V,H	1
ADIW	Rd,K	Add Immediate to Word	$RdH:RdL \leftarrow RdH:RdL + K$	Z,C,N,V,S	2
SUB	Rd, Rr	Subtract two Registers	$Rd \leftarrow Rd - Rr$	Z,C,N,V,H	1
SUBI	Rd, K	Subtract Constant from Register	$Rd \leftarrow Rd - K$	Z,C,N,V,H	1
SBC	Rd, Rr	Subtract with Carry two Registers	$Rd \leftarrow Rd - Rr - C$	Z,C,N,V,H	1
SBCI	Rd, K	Subtract with Carry Constant from Reg.	$Rd \leftarrow Rd - K - C$	Z,C,N,V,H	1
SBIW	Rd,K	Subtract Immediate from Word	$RdH:RdL \leftarrow RdH:RdL - K$	Z,C,N,V,S	2
AND	Rd, Rr	Logical AND Registers	$Rd \leftarrow Rd \& Rr$	Z,N,V	1
ANDI	Rd, K	Logical AND Register and Constant	$Rd \leftarrow Rd \& K$	Z,N,V	1
OR	Rd, Rr	Logical OR Registers	$Rd \leftarrow Rd \vee Rr$	Z,N,V	1
ORI	Rd, K	Logical OR Register and Constant	$Rd \leftarrow Rd \vee K$	Z,N,V	1
EOR	Rd, Rr	Exclusive OR Registers	$Rd \leftarrow Rd \oplus Rr$	Z,N,V	1
COM	Rd	One's Complement	$Rd \leftarrow 0xFF - Rd$	Z,C,N,V	1
NEG	Rd	Two's Complement	$Rd \leftarrow 0x00 - Rd$	Z,C,N,V,H	1
SBR	Rd,K	Set Bit(s) in Register	$Rd \leftarrow Rd \vee K$	Z,N,V	1
CBR	Rd,K	Clear Bit(s) in Register	$Rd \leftarrow Rd \& (\sim K)$	Z,N,V	1
INC	Rd	Increment	$Rd \leftarrow Rd + 1$	Z,N,V	1
DEC	Rd	Decrement	$Rd \leftarrow Rd - 1$	Z,N,V	1
TST	Rd	Test for Zero or Minus	$Rd \leftarrow Rd \& Rd$	Z,N,V	1
CLR	Rd	Clear Register	$Rd \leftarrow Rd \& 0$	Z,N,V	1
SER	Rd	Set Register	$Rd \leftarrow 0xFF$	None	1
MUL	Rd, Rr	Multiply Unsigned	$R1:R0 \leftarrow Rd \times Rr$	Z,C	2
MULS	Rd, Rr	Multiply Signed	$R1:R0 \leftarrow Rd \times Rr$	Z,C	2
MULSU	Rd, Rr	Multiply Signed with Unsigned	$R1:R0 \leftarrow Rd \times Rr$	Z,C	2
FMUL	Rd, Rr	Fractional Multiply Unsigned	$R1:R0 \leftarrow (Rd \times Rr) \lll 1$	Z,C	2
FMULS	Rd, Rr	Fractional Multiply Signed	$R1:R0 \leftarrow (Rd \times Rr) \lll 1$	Z,C	2
FMULSU	Rd, Rr	Fractional Multiply Signed with Unsigned	$R1:R0 \leftarrow (Rd \times Rr) \lll 1$	Z,C	2
BRANCH INSTRUCTIONS					
RJMP	k	Relative Jump	$PC \leftarrow PC + k + 1$	None	2
IJMP		Indirect Jump to (Z)	$PC \leftarrow Z$	None	2
RCALL	k	Relative Subroutine Call	$PC \leftarrow PC + k + 1$	None	3
ICALL		Indirect Call to (Z)	$PC \leftarrow Z$	None	3
RET		Subroutine Return	$PC \leftarrow STACK$	None	4
RETI		Interrupt Return	$PC \leftarrow STACK$	I	4
CPSE	Rd,Rr	Compare, Skip if Equal	if $(Rd = Rr)$ $PC \leftarrow PC + 2$ or 3	None	1/2/3
CP	Rd,Rr	Compare	$Rd - Rr$	Z, N, V, C, H	1
CPC	Rd,Rr	Compare with Carry	$Rd - Rr - C$	Z, N, V, C, H	1
CPI	Rd,K	Compare Register with Immediate	$Rd - K$	Z, N, V, C, H	1
SBRC	Rr, b	Skip if Bit in Register Cleared	if $(Rr(b)=0)$ $PC \leftarrow PC + 2$ or 3	None	1/2/3
SBR	Rr, b	Skip if Bit in Register is Set	if $(Rr(b)=1)$ $PC \leftarrow PC + 2$ or 3	None	1/2/3
SBIC	P, b	Skip if Bit in I/O Register Cleared	if $(P(b)=0)$ $PC \leftarrow PC + 2$ or 3	None	1/2/3
SBIS	P, b	Skip if Bit in I/O Register is Set	if $(P(b)=1)$ $PC \leftarrow PC + 2$ or 3	None	1/2/3
BRBS	s, k	Branch if Status Flag Set	if $(SREG(s) = 1)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRBC	s, k	Branch if Status Flag Cleared	if $(SREG(s) = 0)$ then $PC \leftarrow PC + k + 1$	None	1/2
BREQ	k	Branch if Equal	if $(Z = 1)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRNE	k	Branch if Not Equal	if $(Z = 0)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRCS	k	Branch if Carry Set	if $(C = 1)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRCC	k	Branch if Carry Cleared	if $(C = 0)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRSH	k	Branch if Same or Higher	if $(C = 0)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRLO	k	Branch if Lower	if $(C = 1)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRMI	k	Branch if Minus	if $(N = 1)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRPL	k	Branch if Plus	if $(N = 0)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRGE	k	Branch if Greater or Equal, Signed	if $(N \oplus V = 0)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRLT	k	Branch if Less Than Zero, Signed	if $(N \oplus V = 1)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRHS	k	Branch if Half Carry Flag Set	if $(H = 1)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRHC	k	Branch if Half Carry Flag Cleared	if $(H = 0)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRTS	k	Branch if T Flag Set	if $(T = 1)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRTC	k	Branch if T Flag Cleared	if $(T = 0)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRVS	k	Branch if Overflow Flag is Set	if $(V = 1)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRVC	k	Branch if Overflow Flag is Cleared	if $(V = 0)$ then $PC \leftarrow PC + k + 1$	None	1/2

BRIE	k	Branch if Interrupt Enabled	if (I = 1) then PC ← PC + k + 1	None	1 / 2
BRID	k	Branch if Interrupt Disabled	if (I = 0) then PC ← PC + k + 1	None	1 / 2
DATA TRANSFER INSTRUCTIONS					
MOV	Rd, Rr	Move Between Registers	Rd ← Rr	None	1
MOVW	Rd, Rr	Copy Register Word	Rd+1:Rd ← Rr+1:Rr	None	1
LDI	Rd, K	Load Immediate	Rd ← K	None	1
LD	Rd, X	Load Indirect	Rd ← (X)	None	2
LD	Rd, X+	Load Indirect and Post-Inc.	Rd ← (X), X ← X + 1	None	2
LD	Rd, -X	Load Indirect and Pre-Dec.	X ← X - 1, Rd ← (X)	None	2
LD	Rd, Y	Load Indirect	Rd ← (Y)	None	2
LD	Rd, Y+	Load Indirect and Post-Inc.	Rd ← (Y), Y ← Y + 1	None	2
LD	Rd, -Y	Load Indirect and Pre-Dec.	Y ← Y - 1, Rd ← (Y)	None	2
LDD	Rd, Y+q	Load Indirect with Displacement	Rd ← (Y + q)	None	2
LD	Rd, Z	Load Indirect	Rd ← (Z)	None	2
LD	Rd, Z+	Load Indirect and Post-Inc.	Rd ← (Z), Z ← Z + 1	None	2
LD	Rd, -Z	Load Indirect and Pre-Dec.	Z ← Z - 1, Rd ← (Z)	None	2
LDD	Rd, Z+q	Load Indirect with Displacement	Rd ← (Z + q)	None	2
LDS	Rd, k	Load Direct from SRAM	Rd ← (k)	None	2
ST	X, Rr	Store Indirect	(X) ← Rr	None	2
ST	X+, Rr	Store Indirect and Post-Inc.	(X) ← Rr, X ← X + 1	None	2
ST	-X, Rr	Store Indirect and Pre-Dec.	X ← X - 1, (X) ← Rr	None	2
ST	Y, Rr	Store Indirect	(Y) ← Rr	None	2
ST	Y+, Rr	Store Indirect and Post-Inc.	(Y) ← Rr, Y ← Y + 1	None	2
ST	-Y, Rr	Store Indirect and Pre-Dec.	Y ← Y - 1, (Y) ← Rr	None	2
STD	Y+q, Rr	Store Indirect with Displacement	(Y + q) ← Rr	None	2
ST	Z, Rr	Store Indirect	(Z) ← Rr	None	2
ST	Z+, Rr	Store Indirect and Post-Inc.	(Z) ← Rr, Z ← Z + 1	None	2
ST	-Z, Rr	Store Indirect and Pre-Dec.	Z ← Z - 1, (Z) ← Rr	None	2
STD	Z+q, Rr	Store Indirect with Displacement	(Z + q) ← Rr	None	2
STS	k, Rr	Store Direct to SRAM	(k) ← Rr	None	2
LPM	Rd, Z	Load Program Memory	R0 ← (Z)	None	3
LPM	Rd, Z+	Load Program Memory and Post-Inc	Rd ← (Z)	None	3
LPM	Rd, Z+	Load Program Memory and Post-Inc	Rd ← (Z), Z ← Z + 1	None	3
SPM		Store Program Memory	(Z) ← R1:R0	None	-
IN	Rd, P	In Port	Rd ← P	None	1
OUT	P, Rr	Out Port	P ← Rr	None	1
PUSH	Rr	Push Register on Stack	STACK ← Rr	None	2
POP	Rd	Pop Register from Stack	Rd ← STACK	None	2
BIT AND BIT-TEST INSTRUCTIONS					
SBI	P, b	Set Bit in I/O Register	I/O(P, b) ← 1	None	2
CBI	P, b	Clear Bit in I/O Register	I/O(P, b) ← 0	None	2
LSL	Rd	Logical Shift Left	Rd(n+1) ← Rd(n), Rd(0) ← 0	Z, C, N, V	1
LSR	Rd	Logical Shift Right	Rd(n) ← Rd(n+1), Rd(7) ← 0	Z, C, N, V	1
ROL	Rd	Rotate Left Through Carry	Rd(0) ← C, Rd(n+1) ← Rd(n), C ← Rd(7)	Z, C, N, V	1
ROR	Rd	Rotate Right Through Carry	Rd(7) ← C, Rd(n) ← Rd(n+1), C ← Rd(0)	Z, C, N, V	1
ASR	Rd	Arithmetic Shift Right	Rd(n) ← Rd(n+1), n=0:6	Z, C, N, V	1
SWAP	Rd	Swap Nibbles	Rd(3:0) ← Rd(7:4), Rd(7:4) ← Rd(3:0)	None	1
BSET	s	Flag Set	SREG(s) ← 1	SREG(s)	1
BCLR	s	Flag Clear	SREG(s) ← 0	SREG(s)	1
BST	Rr, b	Bit Store from Register to T	T ← Rr(b)	T	1
BLD	Rd, b	Bit load from T to Register	Rd(b) ← T	None	1
SEC		Set Carry	C ← 1	C	1
CLC		Clear Carry	C ← 0	C	1
SEN		Set Negative Flag	N ← 1	N	1
CLN		Clear Negative Flag	N ← 0	N	1
SEZ		Set Zero Flag	Z ← 1	Z	1
CLZ		Clear Zero Flag	Z ← 0	Z	1
SEI		Global Interrupt Enable	I ← 1	I	1
CLI		Global Interrupt Disable	I ← 0	I	1
SES		Set Signed Test Flag	S ← 1	S	1
CLS		Clear Signed Test Flag	S ← 0	S	1
SEV		Set Twos Complement Overflow	V ← 1	V	1
CLV		Clear Twos Complement Overflow	V ← 0	V	1
SET		Set T in SREG	T ← 1	T	1
CLT		Clear T in SREG	T ← 0	T	1
SEH		Set Half Carry Flag in SREG	H ← 1	H	1
CLH		Clear Half Carry Flag in SREG	H ← 0	H	1
MCU CONTROL INSTRUCTIONS					
NOP		No Operation		None	1
SLEEP		Sleep	(see specific descr. for Sleep function)	None	1
WDR		Watchdog Reset	(see specific descr. for WDR/timer)	None	1

7. Ordering Information

Speed (MHz)	Power Supply (V)	Ordering Code	Package ⁽¹⁾	Operation Range
16	2.7 - 5.5	ATmega8A-AU ⁽²⁾ ATmega8A-PU ⁽²⁾ ATmega8A-MU ⁽²⁾	32A 28P3 32M1-A	Industrial (-40°C to 85°C)

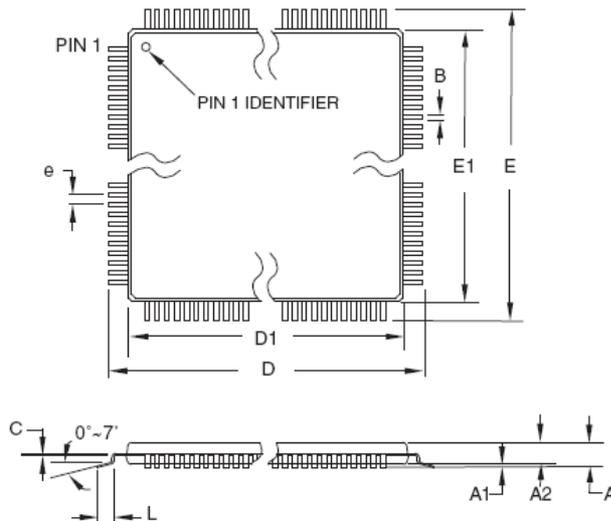
Notes:

1. This device can also be supplied in wafer form. Please contact your local Atmel sales office for detailed ordering information and minimum quantities.
2. Pb-free packaging alternative, complies to the European Directive for Restriction of Hazardous Substances (RoHS directive). Also Halide free and fully Green.

Package Type	
32A	32-lead, Thin (1.0 mm) Plastic Quad Flat Package (TQFP)
28P3	28-lead, 0.300" Wide, Plastic Dual Inline Package (PDIP)
32M1-A	32-pad, 5 x 5 x 1.0 body, Lead Pitch 0.50 mm Quad Flat No-Lead/Micro Lead Frame Package (QFN/MLF)

8. Packaging Information

8.1 32A



COMMON DIMENSIONS
(Unit of Measure = mm)

SYMBOL	MIN	NOM	MAX	NOTE
A	-	-	1.20	
A1	0.05	-	0.15	
A2	0.95	1.00	1.05	
D	8.75	9.00	9.25	
D1	6.90	7.00	7.10	Note 2
E	8.75	9.00	9.25	
E1	6.90	7.00	7.10	Note 2
B	0.30	-	0.45	
C	0.09	-	0.20	
L	0.45	-	0.75	
e	0.80 TYP			

- Notes:
1. This package conforms to JEDEC reference MS-026, Variation ABA.
 2. Dimensions D1 and E1 do not include mold protrusion. Allowable protrusion is 0.25 mm per side. Dimensions D1 and E1 are maximum plastic body size dimensions including mold mismatch.
 3. Lead coplanarity is 0.10 mm maximum.

10/5/2001



2325 Orchard Parkway
San Jose, CA 95131

TITLE

32A, 32-lead, 7 x 7 mm Body Size, 1.0 mm Body Thickness, 0.8 mm Lead Pitch, Thin Profile Plastic Quad Flat Package (TQFP)

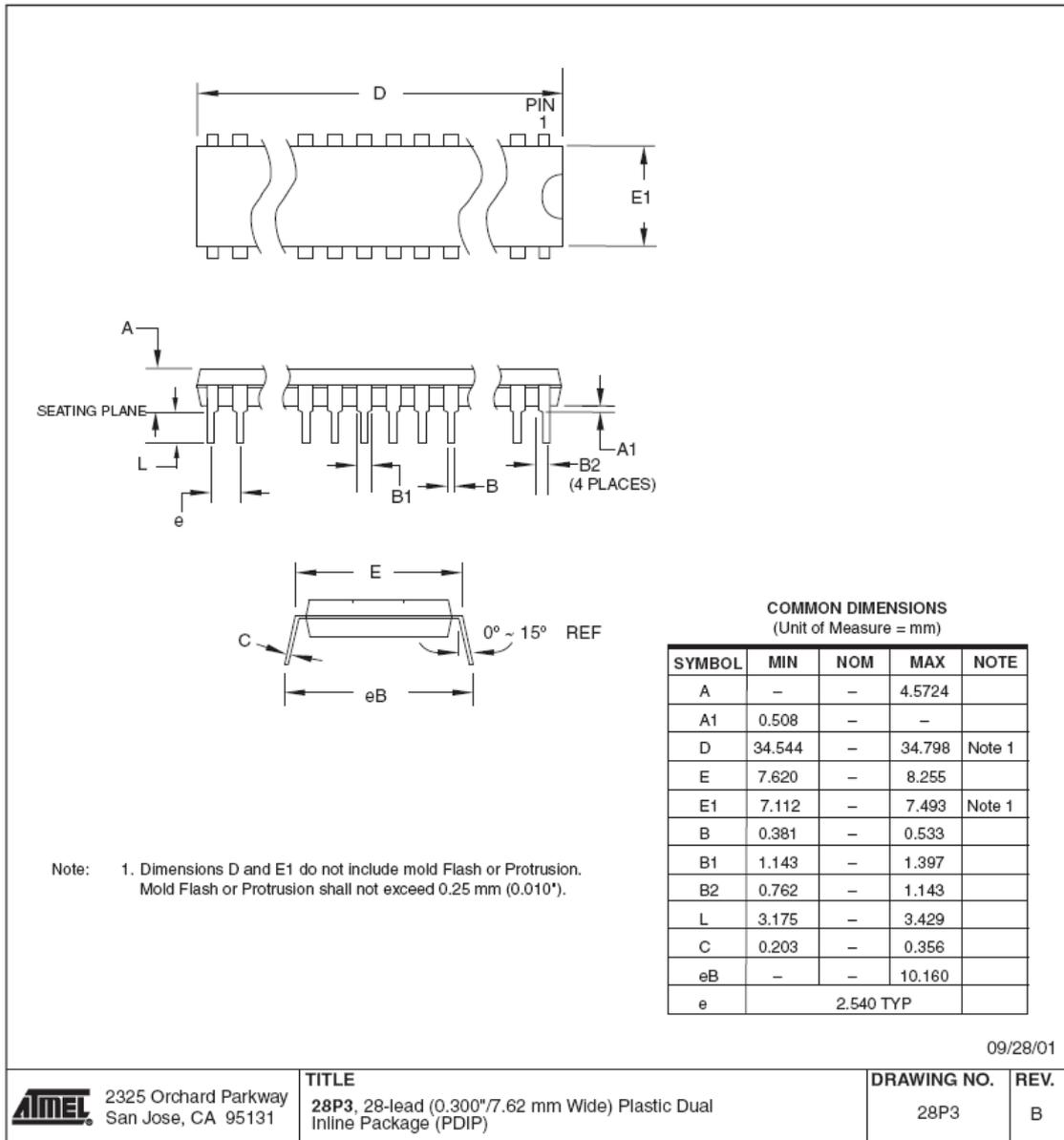
DRAWING NO.

32A

REV.

B

8.2 28P3



09/28/01



2325 Orchard Parkway
San Jose, CA 95131

TITLE

28P3, 28-lead (0.300"/7.62 mm Wide) Plastic Dual
Inline Package (PDIP)

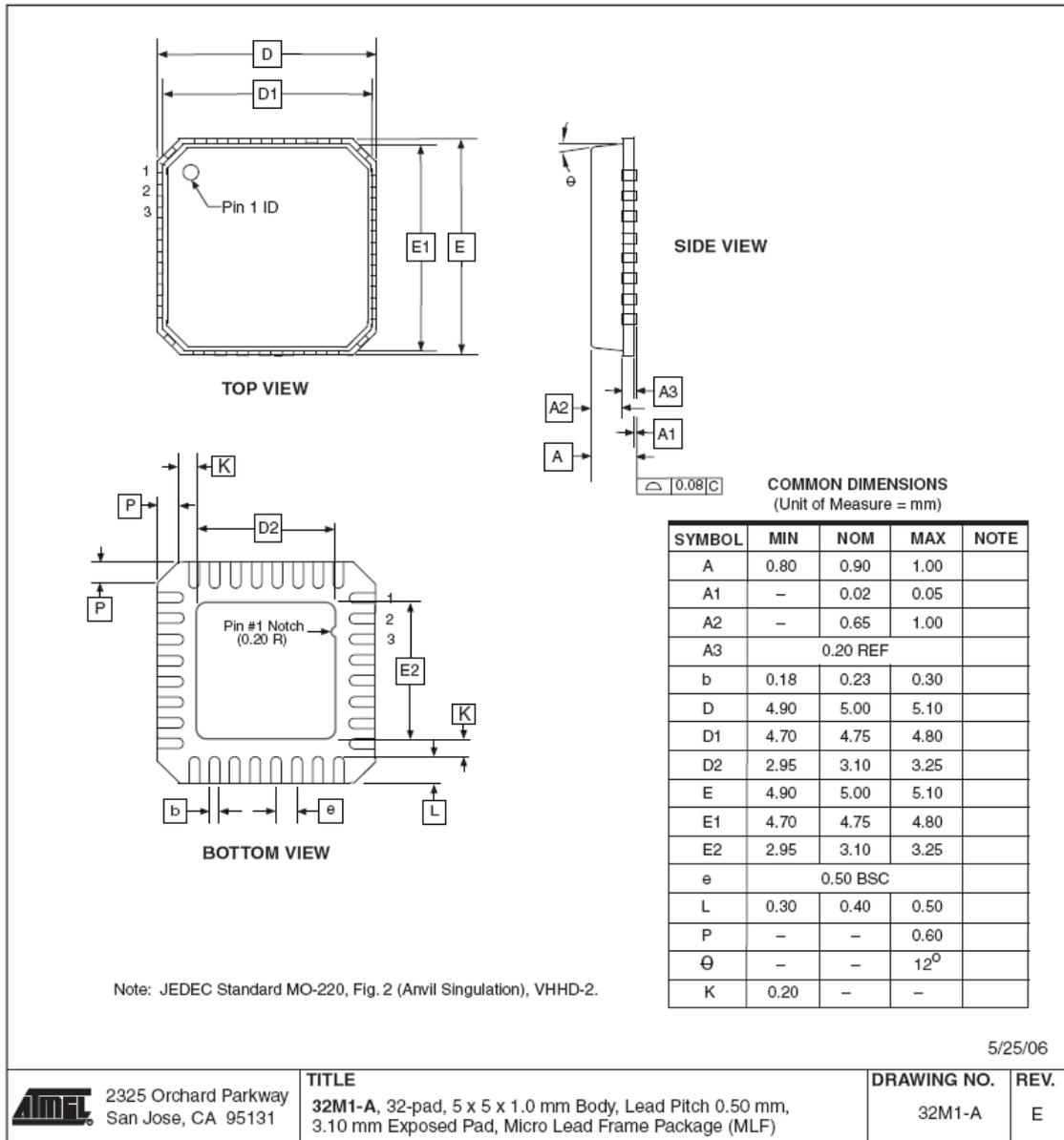
DRAWING NO.

28P3

REV.

B

32M1-A



2325 Orchard Parkway
San Jose, CA 95131

TITLE
32M1-A, 32-pad, 5 x 5 x 1.0 mm Body, Lead Pitch 0.50 mm,
3.10 mm Exposed Pad, Micro Lead Frame Package (MLF)

DRAWING NO.
32M1-A

REV.
E

9. Errata

The revision letter in this section refers to the revision of the ATmega8A device.

9.1 ATmega8A, rev. L

- **First Analog Comparator conversion may be delayed**
- **Interrupts may be lost when writing the timer registers in the asynchronous timer**
- **Signature may be Erased in Serial Programming Mode**
- **CKOPT Does not Enable Internal Capacitors on XTALn/TOSCn Pins when 32 KHz Oscillator is Used to Clock the Asynchronous Timer/Counter2**
- **Reading EEPROM by using ST or STS to set EERE bit triggers unexpected interrupt request**

1. First Analog Comparator conversion may be delayed

If the device is powered by a slow rising V_{CC} , the first Analog Comparator conversion will take longer than expected on some devices.

Problem Fix / Workaround

When the device has been powered or reset, disable then enable the Analog Comparator before the first conversion.

2. Interrupts may be lost when writing the timer registers in the asynchronous timer

The interrupt will be lost if a timer register that is synchronous timer clock is written when the asynchronous Timer/Counter register (TCNTx) is 0x00.

Problem Fix / Workaround

Always check that the asynchronous Timer/Counter register neither have the value 0xFF nor 0x00 before writing to the asynchronous Timer Control Register (TCCRx), asynchronous Timer Counter Register (TCNTx), or asynchronous Output Compare Register (OCRx).

3. Signature may be Erased in Serial Programming Mode

If the signature bytes are read before a chip erase command is completed, the signature may be erased causing the device ID and calibration bytes to disappear. This is critical, especially, if the part is running on internal RC oscillator.

Problem Fix / Workaround:

Ensure that the chip erase command has exceeded before applying the next command.

4. CKOPT Does not Enable Internal Capacitors on XTALn/TOSCn Pins when 32 KHz Oscillator is Used to Clock the Asynchronous Timer/Counter2

When the internal RC Oscillator is used as the main clock source, it is possible to run the Timer/Counter2 asynchronously by connecting a 32 KHz Oscillator between XTAL1/TOSC1 and XTAL2/TOSC2. But when the internal RC Oscillator is selected as the main clock source, the CKOPT Fuse does not control the internal capacitors on XTAL1/TOSC1 and XTAL2/TOSC2. As long as there are no capacitors connected to XTAL1/TOSC1 and XTAL2/TOSC2, safe operation of the Oscillator is not guaranteed.

Problem Fix / Workaround

Use external capacitors in the range of 20 - 36 pF on XTAL1/TOSC1 and XTAL2/TOSC2. This will be fixed in ATmega8A Rev. G where the CKOPT Fuse will control internal capacitors also when internal RC Oscillator is selected as main clock source. For ATmega8A Rev. G, CKOPT = 0 (programmed) will enable the internal capacitors on XTAL1 and XTAL2. Customers who want compatibility between Rev. G and older revisions, must ensure that CKOPT is unprogrammed (CKOPT = 1).

5. Reading EEPROM by using ST or STS to set EERE bit triggers unexpected interrupt request.

Reading EEPROM by using the ST or STS command to set the EERE bit in the EECR register triggers an unexpected EEPROM interrupt request.

Problem Fix / Workaround

Always use OUT or SBI to set EERE in EECR.

ANEXO B

DATA SHEET DEL REGULADOR DE VOLTAJE (LM7805)



www.fairchildsemi.com

MC78XX/LM78XX/MC78XXA

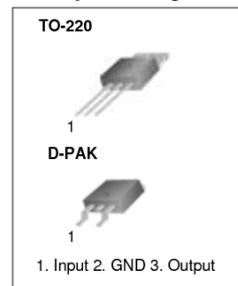
3-Terminal 1A Positive Voltage Regulator

Features

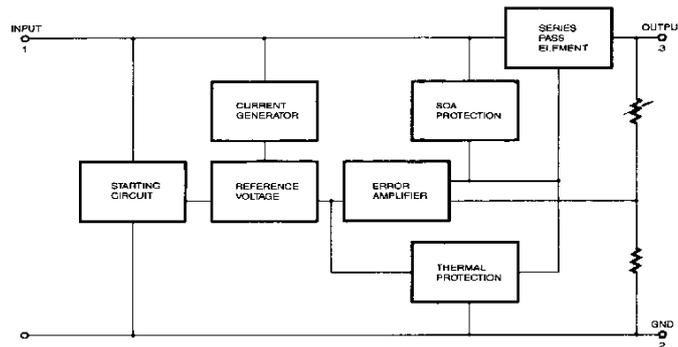
- Output Current up to 1A
- Output Voltages of 5, 6, 8, 9, 10, 12, 15, 18, 24V
- Thermal Overload Protection
- Short Circuit Protection
- Output Transistor Safe Operating Area Protection

Description

The MC78XX/LM78XX/MC78XXA series of three terminal positive regulators are available in the TO-220/D-PAK package and with several fixed output voltages, making them useful in a wide range of applications. Each type employs internal current limiting, thermal shut down and safe operating area protection, making it essentially indestructible. If adequate heat sinking is provided, they can deliver over 1A output current. Although designed primarily as fixed voltage regulators, these devices can be used with external components to obtain adjustable voltages and currents.



Internal Block Diagram



Rev. 1.0.1

Electrical Characteristics (LM7805)

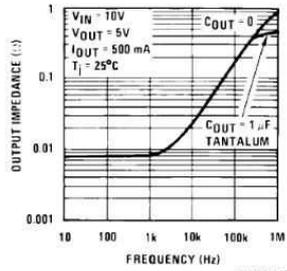
Refer to the test circuits. $-40^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$, $I_O = 500\text{mA}$, $V_I = 10\text{V}$, $C_I = 0.1\mu\text{F}$, unless otherwise specified.

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Unit	
V_O	Output Voltage	$T_J = +25^{\circ}\text{C}$	4.8	5.0	5.2	V	
		$5\text{mA} \leq I_O \leq 1\text{A}$, $P_D \leq 15\text{W}$, $V_I = 7\text{V to } 20\text{V}$	4.75	5.0	5.25		
Regline	Line Regulation ⁽¹⁾	$T_J = +25^{\circ}\text{C}$	$V_O = 7\text{V to } 25\text{V}$	-	4.0	100	mV
			$V_I = 8\text{V to } 12\text{V}$	-	1.6	50.0	
Regload	Load Regulation ⁽¹⁾	$T_J = +25^{\circ}\text{C}$	$I_O = 5\text{mA to } 1.5\text{A}$	-	9.0	100	mV
			$I_O = 250\text{mA to } 750\text{mA}$	-	4.0	50.0	
I_Q	Quiescent Current	$T_J = +25^{\circ}\text{C}$	-	5.0	8.0	mA	
ΔI_Q	Quiescent Current Change	$I_O = 5\text{mA to } 1\text{A}$	-	0.03	0.5	mA	
		$V_I = 7\text{V to } 25\text{V}$	-	0.3	1.3		
$\Delta V_O/\Delta T$	Output Voltage Drift ⁽²⁾	$I_O = 5\text{mA}$	-	-0.8	-	mV/ $^{\circ}\text{C}$	
V_N	Output Noise Voltage	$f = 10\text{Hz to } 100\text{kHz}$, $T_A = +25^{\circ}\text{C}$	-	42.0	-	$\mu\text{V}/V_O$	
RR	Ripple Rejection ⁽²⁾	$f = 120\text{Hz}$, $V_O = 8\text{V to } 18\text{V}$	62.0	73.0	-	dB	
V_{DROPP}	Dropout Voltage	$I_O = 1\text{A}$, $T_J = +25^{\circ}\text{C}$	-	2.0	-	V	
r_O	Output Resistance ⁽²⁾	$f = 1\text{kHz}$	-	15.0	-	m Ω	
I_{SC}	Short Circuit Current	$V_I = 35\text{V}$, $T_A = +25^{\circ}\text{C}$	-	230	-	mA	
I_{PK}	Peak Current ⁽²⁾	$T_J = +25^{\circ}\text{C}$	-	2.2	-	A	

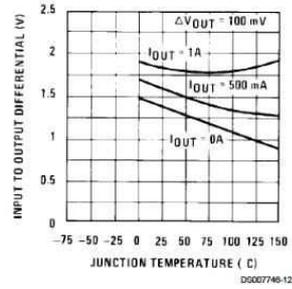
LM78XX

Typical Performance Characteristics (Continued)

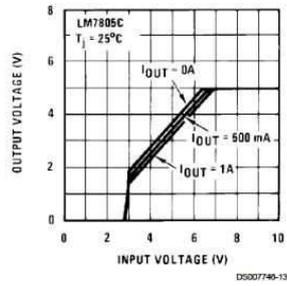
Output Impedance



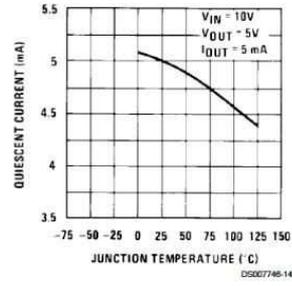
Dropout Voltage



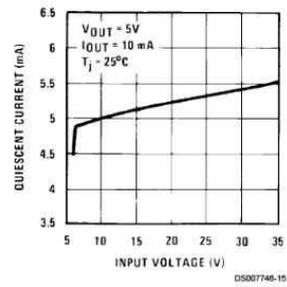
Dropout Characteristics



Quiescent Current



Quiescent Current



ANEXO C

TRANSISTOR 2N3904



2N3904

SMALL SIGNAL NPN TRANSISTOR

Ordering Code	Marking	Package / Shipment
2N3904	2N3904	TO-92 / Bulk
2N3904-AP	2N3904	TO-92 / Ammopack

- SILICON EPITAXIAL PLANAR NPN TRANSISTOR
- TO-92 PACKAGE SUITABLE FOR THROUGH-HOLE PCB ASSEMBLY
- THE PNP COMPLEMENTARY TYPE IS 2N3906



APPLICATIONS

- WELL SUITABLE FOR TV AND HOME APPLIANCE EQUIPMENT
- SMALL LOAD SWITCH TRANSISTOR WITH HIGH GAIN AND LOW SATURATION VOLTAGE

2N3903, 2N3904

General Purpose

Transistors

NPN Silicon

ELECTRICAL CHARACTERISTICS ($T_A = 25^\circ\text{C}$ unless otherwise noted)

Characteristic	Symbol	Min	Max	Unit	
OFF CHARACTERISTICS					
Collector–Emitter Breakdown Voltage (Note 2) ($I_C = 1.0\text{ mAdc}$, $I_B = 0$)	$V_{(BR)CEO}$	40	–	Vdc	
Collector–Base Breakdown Voltage ($I_C = 10\text{ Adc}$, $I_E = 0$)	$V_{(BR)CBO}$	60	–	Vdc	
Emitter–Base Breakdown Voltage ($I_E = 10\text{ Adc}$, $I_C = 0$)	$V_{(BR)EBO}$	6.0	–	Vdc	
Base Cutoff Current ($V_{CE} = 30\text{ Vdc}$, $V_{EB} = 3.0\text{ Vdc}$)	I_{BL}	–	50	nAd	
Collector Cutoff Current ($V_{CE} = 30\text{ Vdc}$, $V_{EB} = 3.0\text{ Vdc}$)	I_{CEX}	–	50	nAd	
ON CHARACTERISTICS					
DC Current Gain (Note 2) ($I_C = 0.1\text{ mAdc}$, $V_{CE} = 1.0\text{ Vdc}$)	h_{FE}	2N3903	20	–	–
		2N3904	40	–	–
($I_C = 1.0\text{ mAdc}$, $V_{CE} = 1.0\text{ Vdc}$)		2N3903	35	–	–
		2N3904	70	–	–
($I_C = 10\text{ mAdc}$, $V_{CE} = 1.0\text{ Vdc}$)		2N3903	50	150	–
		2N3904	100	300	–
($I_C = 50\text{ mAdc}$, $V_{CE} = 1.0\text{ Vdc}$)		2N3903	30	–	–
		2N3904	60	–	–
($I_C = 100\text{ mAdc}$, $V_{CE} = 1.0\text{ Vdc}$)		2N3903	15	–	–
		2N3904	30	–	–
Collector–Emitter Saturation Voltage (Note 2) ($I_C = 10\text{ mAdc}$, $I_B = 1.0\text{ mAdc}$) ($I_C = 50\text{ mAdc}$, $I_B = 5.0\text{ mAdc}$)	$V_{CE(sat)}$	–	0.2 0.3	Vdc	
Base–Emitter Saturation Voltage (Note 2) ($I_C = 10\text{ mAdc}$, $I_B = 1.0\text{ mAdc}$) ($I_C = 50\text{ mAdc}$, $I_B = 5.0\text{ mAdc}$)	$V_{BE(sat)}$	0.65 –	0.85 0.95	Vdc	
SMALL–SIGNAL CHARACTERISTICS					
Current–Gain – Bandwidth Product ($I_C = 10\text{ mAdc}$, $V_{CE} = 20\text{ Vdc}$, $f = 100\text{ MHz}$)	f_T	2N3903	250	–	MHz
		2N3904	300	–	–
Output Capacitance ($V_{CB} = 5.0\text{ Vdc}$, $I_E = 0$, $f = 1.0\text{ MHz}$)	C_{obo}	–	4.0	pF	
Input Capacitance ($V_{EB} = 0.5\text{ Vdc}$, $I_C = 0$, $f = 1.0\text{ MHz}$)	C_{ibo}	–	8.0	pF	
Input Impedance ($I_C = 1.0\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$, $f = 1.0\text{ kHz}$)	h_{ie}	2N3903	1.0	8.0	k
		2N3904	1.0	10	–
Voltage Feedback Ratio ($I_C = 1.0\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$, $f = 1.0\text{ kHz}$)	h_{re}	2N3903	0.1	5.0	$\times 10^{-4}$
		2N3904	0.5	8.0	–
Small–Signal Current Gain ($I_C = 1.0\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$, $f = 1.0\text{ kHz}$)	h_{fe}	2N3903	50	200	–
		2N3904	100	400	–
Output Admittance ($I_C = 1.0\text{ mAdc}$, $V_{CE} = 10\text{ Vdc}$, $f = 1.0\text{ kHz}$)	h_{oe}	–	40	–	
Noise Figure ($I_C = 100\text{ Adc}$, $V_{CE} = 5.0\text{ Vdc}$, $R_S = 1.0\text{ k}$, $f = 1.0\text{ kHz}$)	NF	2N3903	–	6.0	dB
		2N3904	–	5.0	–
SWITCHING CHARACTERISTICS					
Delay Time	$(V_{CC} = 3.0\text{ Vdc}$, $V_{BE} = 0.5\text{ Vdc}$, $I_C = 10\text{ mAdc}$, $I_{B1} = 1.0\text{ mAdc}$)	t_d	–	35	n
Rise Time		t_r	–	35	n
Storage Time	$(V_{CC} = 3.0\text{ Vdc}$, $I_C = 10\text{ mAdc}$, $I_{B1} = I_{B2} = 1.0\text{ mAdc}$)	2N3903	–	175	N
		2N3904	–	200	s
Fall Time	t_f	–	50	n	

2. Pulse Test: Pulse Width 300 μ s; Duty Cycle 2%.

ANEXO D

RELÉ SRD-06VDC-SL-C

SONGLE RELAY

	RELAY ISO9002	SRD
--	----------------------	------------



1. MAIN FEATURES

- Switching capacity available by 10A in spite of small size design for highdensity P.C. board mounting technique.
- UL,CUL,TUV recognized.
- Selection of plastic material for high temperature and better chemical solution performance.
 - Sealed types available.
- Simple relay magnetic circuit to meet low cost of mass production.

2. APPLICATIONS

- Domestic appliance, office machine, audio, equipment, automobile, etc.
- (Remote control TV receiver, monitor display, audio equipment high rushing current use application.)

3. ORDERING INFORMATION

SRD	XX VDC	S	L	C
Model of relay	Nominal coil voltage	Structure	Coil	Contact form
SRD	03 05 06 09 12 14 18 VDC	S:Sealed type	L:0.36W	A:1 form A B:1 form B
		F:Flux free type	D:0.45W	C:1 form C

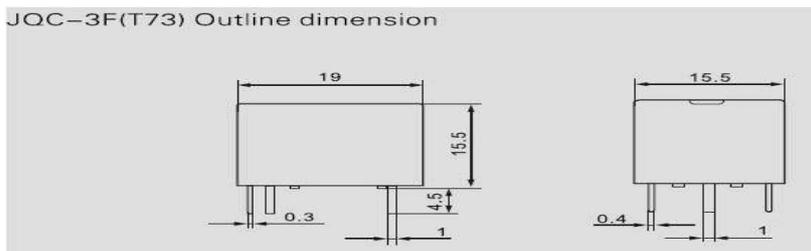
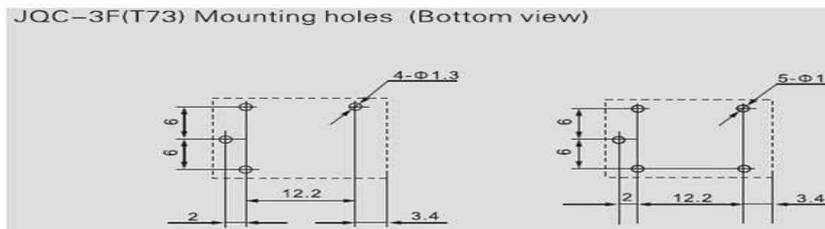
4. RATING

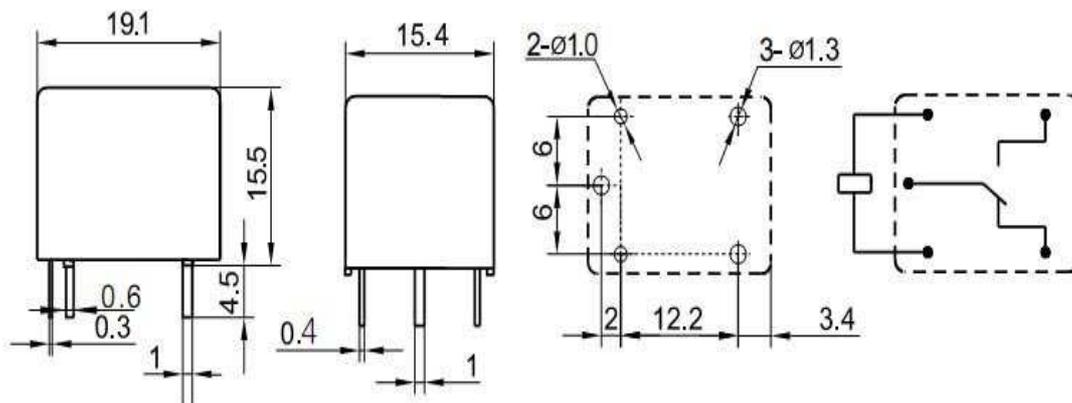
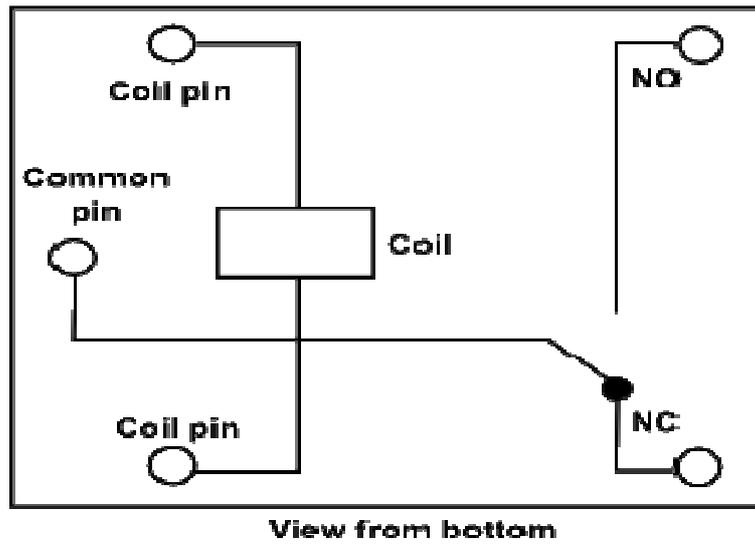
CCC	FILE NUMBER:CQC03001003729	7A/240VDC
CCC	FILE NUMBER:CQC03001003731	10A/250VDC
UL/CUL	FILE NUMBER: E167996	10A/125VAC 28VDC
TUV	FILE NUMBER: R50056114	10A/250VAC 30VD

5. DIMENSION(unit:mm)

DRILLING(unit:mm)

WIRING DIAGRAM





- Remark:**
1. When the tolerance is not indicated in dimensions,
 - If the dimension is ≤ 1 mm, the tolerance is ± 0.2 mm;
 - If the dimension is between 1-5 mm, the tolerance is ± 0.3 mm;
 - If the dimension is ≥ 5 mm, the tolerance is ± 0.4 mm;
 2. The tolerance of mounting holes is ± 0.1 mm.

6. COIL DATA CHART (AT20°C)

Coil Sensitivity	Coil Voltage Code	Nominal Voltage (VDC)	Nominal Current (mA)	Coil Resistance (Ω) $\pm 10\%$	Power Consumption (W)	Pull-In Voltage (VDC)	Drop-Out Voltage (VDC)	Max-Allowable Voltage (VDC)
SRD (High Sensitivity)	03	03	120	25	abt. 0.36W	75%Max.	10% Min.	120%
	05	05	71.4	70				
	06	06	60	100				
	09	09	40	225				
	12	12	30	400				
	24	24	15	1600				
SRD (Standard)	03	03	150	20	abt. 0.45W	75% Max.	10% Min.	110%
	05	05	89.3	55				
	06	06	75	80				
	09	09	50	180				
	12	12	37.5	320				
	24	24	18.7	1280				
	48	48	10	4500	abt. 0.51W			

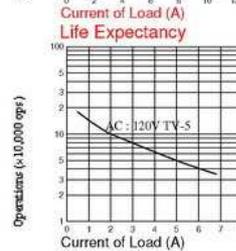
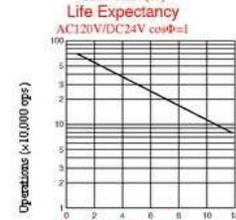
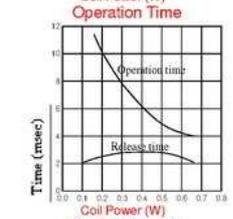
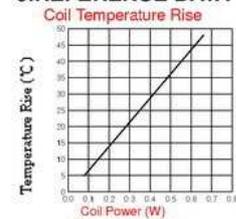
7. CONTACT RATING

Item	Type	SRD	
		FORM C	FORM A
Contact Capacity Resistive Load ($\cos\Phi=1$)		10A 125VAC	10A 30VDC 10A 250VAC
Inductive Load ($\cos\Phi=0.4$ L/R=7msec)		3A 120VAC 3A 28VDC	5A 120VAC 5A 28VDC
Max. Allowable Voltage		250VAC/110VDC	250VAC/110VDC
Max. Allowable Power Force		800VAC/240W	1200VA/300W
Contact Material		AgCdO	AgCdO

8. PERFORMANCE (at initial value)

Item	Type	SRD
Contact Resistance		100m Ω Max.
Operation Time		10msec Max.
Release Time		5msec Max.
Dielectric Strength		
Between coil & contact		1500VAC 50/60HZ (1 minute)
Between contacts		1000VAC 50/60HZ (1 minute)
Insulation Resistance		100 M Ω Min. (500VDC)
Max. ON/OFF Switching		
Mechanically		300 operation/min
Electrically		30 operation/min
Ambient Temperature		-40°C to +85°C
Operating Humidity		45 to 85% RH
Vibration		
Endurance		10 to 55Hz Double Amplitude 1.5mm
Error Operation		10 to 55Hz Double Amplitude 1.5mm
Shock		
Endurance		100G Min.
Error Operation		10G Min.
Life Expectancy		
Mechanically		10 ⁷ operations. Min. (no load)
Electrically		10 ⁵ operations. Min. (at rated coil voltage)
Weight		abt. 10grs.

9. REFERENCE DATA



ANEXO E

LISTADO DEL PROGRAMA

//PROGRAMA

```

#include <avr/io.h>                // INCLUYE LIBRERIAS
#define F_CPU 1000000UL           // CONFIGURA CRISTAL PARA 1MZ
#include <util/delay.h>           // INCLUYE LIBRERIA DE RETARDO

char caracteruno, caracterdos, caractertres,retardo; // VARIABLES

int main(void)
{
    DDRB=255;                      // CONFIGURA PUERTO B COMO SALIDAS
    DDRC=255;                      // CONFIGURA PUERTO C COMO SALIDAS
    UBRR = 12;                     // CONFIGURA VELOCIDAD DE COMUNICACION SERIAL A 9600 BAUDIOS
    UCSRB = 0b00011000;           // ACTIVA RECEPCION Y TRANSMISION
    UCSRC = 0b10000110;          // CONFIGURA COMUNICACION A 8 BIT, UN BIT DE PARADA
    _delay_ms(250); _delay_ms(250); _delay_ms(250); // PAUSA
    UDR ='A';                      // CARGA LA LETRA A PARA ENVIAR
    while(!(UCSRA & (1 << UDRE))) ; // ESPERA HASTA ACABAR DE ENVIAR BYTE
    UDR ='T';

```

```
while(!(UCSRA & (1 << UDRE))) ;
UDR ='D';
while(!(UCSRA & (1 << UDRE))) ;
UDR ='1';                                // ENVIA CODIGO DE LLAMADA
while(!(UCSRA & (1 << UDRE))) ;
UDR ='5';
while(!(UCSRA & (1 << UDRE))) ;
UDR =',';
while(!(UCSRA & (1 << UDRE))) ;
UDR =13;                                  // ENTER
while(!(UCSRA & (1 << UDRE))) ;
_delay_ms(250); _delay_ms(250); _delay_ms(250); _delay_ms(250);    // PAUSA
_delay_ms(250); _delay_ms(250); _delay_ms(250); _delay_ms(250);
UDR ='A';
while(!(UCSRA & (1 << UDRE))) ;
UDR ='T';
```

```
while(!(UCSRA & (1 << UDRE))) ;
UDR ='+';
while(!(UCSRA & (1 << UDRE))) ;
UDR ='C';
while(!(UCSRA & (1 << UDRE))) ;                               // FINALIZA LLAMADA
UDR ='H';
while(!(UCSRA & (1 << UDRE))) ;
UDR ='U';
while(!(UCSRA & (1 << UDRE))) ;
UDR ='P';
while(!(UCSRA & (1 << UDRE))) ;
UDR =13;
while(!(UCSRA & (1 << UDRE))) ;
_delay_ms(250); _delay_ms(250); _delay_ms(250); _delay_ms(250); _delay_ms(250); _delay_ms(250); // PAUSA
UDR ='A';
while(!(UCSRA & (1 << UDRE))) ;
```

```
UDR ='T';
while(!(UCSRA & (1 << UDRE))) ;
UDR ='+';
while(!(UCSRA & (1 << UDRE))) ;
UDR ='C';
while(!(UCSRA & (1 << UDRE))) ;
UDR ='M';
while(!(UCSRA & (1 << UDRE))) ;
UDR ='G'; //CONFIGURA SMS
while(!(UCSRA & (1 << UDRE))) ;
UDR ='F';
while(!(UCSRA & (1 << UDRE))) ;
UDR ='=';
while(!(UCSRA & (1 << UDRE))) ;
UDR ='1';
while(!(UCSRA & (1 << UDRE))) ;
```

```
UDR =13;
while(!(UCSRA & (1 << UDRE))) ;
_delay_ms(250); _delay_ms(250); _delay_ms(250); _delay_ms(250); _delay_ms(250); _delay_ms(250); // PAUSA
UDR ='A';
while(!(UCSRA & (1 << UDRE))) ;
UDR ='T';
while(!(UCSRA & (1 << UDRE))) ;
UDR ='+';
while(!(UCSRA & (1 << UDRE))) ;
UDR ='C';
while(!(UCSRA & (1 << UDRE))) ;
UDR ='N';
while(!(UCSRA & (1 << UDRE))) ;
UDR ='M';
while(!(UCSRA & (1 << UDRE))) ;
UDR ='I';
```

```
while(!(UCSRA & (1 << UDRE))) ;           // COMANDO AT, INDICA AL CELULAR QUE
UDR ='=';                                 // ENVIE LOS SMS POR EL PUERTO SERIAL
while(!(UCSRA & (1 << UDRE))) ;
UDR ='1';
while(!(UCSRA & (1 << UDRE))) ;
UDR =',';
while(!(UCSRA & (1 << UDRE))) ;
UDR ='2';
while(!(UCSRA & (1 << UDRE))) ;
UDR =',';
while(!(UCSRA & (1 << UDRE))) ;
UDR ='0';
while(!(UCSRA & (1 << UDRE))) ;
UDR =',';
while(!(UCSRA & (1 << UDRE))) ;
UDR ='0';
```

```

while(!(UCSRA & (1 << UDRE))) ;
UDR =',';
while(!(UCSRA & (1 << UDRE))) ;
UDR ='0';
while(!(UCSRA & (1 << UDRE))) ;
UDR =13;
while(!(UCSRA & (1 << UDRE))) ;
_delay_ms(250); _delay_ms(250); _delay_ms(250); _delay_ms(250); _delay_ms(250); // PAUSA
PORTC=1; // ACTIVA PINC0, INDICA QUE EL CIRCUITO ESTA LISTO PARA RECIBIR SMS
for(;;){
    uno:
    while(!(UCSRA & (1 << RXC))) ; // ESPERA HASTA RECIBIR CARACTER
        caracteruno = UDR;
        if(caracteruno != 'O') {goto uno;} // DISCRIMINA SI EL CARACTER ES LA LETRA O
    dos:
        while(!(UCSRA & (1 << RXC))) ; // ESPERA HASTA RECIBIR CARACTER

```

```

caracterdos = UDR;
if(caracterdos == 'O') {goto dos;}           // SI SEGUNDO CARACTER ES DIFERENTE DE O SALTA A tres
if(caracterdos == 'F') {goto tres;}        // SI SEGUNDO CARACTER ES DIFERENTE DE F SALTA A tres
if(caracterdos != 'N') {goto uno;}         // SI SEGUNDO CARACTER ES N ENTONCES SALTA A uno
tres:
while(!(UCSRA & (1 << RXC))) ;             // RECIBE TRECER CARACTER
caractertres = UDR;
if(caracteruno == 'O') {if(caracterdos == 'N'){ if(caractertres == '1'){PORTB |= 1 << PB0;PORTB |= 1 << PB3;};};}
// SI CARACTERE RECIBIDOS = ON1 ENTONCES ACTIVA PINB0 Y PINB3
if(caracteruno == 'O') {if(caracterdos == 'N'){ if(caractertres == '2'){PORTB |= 1 << PB1;PORTB |= 1 << PB4;};};}
// SI CARACTERE RECIBIDOS = ON2 ENTONCES ACTIVA PINB1 Y PINB4
if(caracteruno == 'O') {if(caracterdos == 'N'){ if(caractertres == '3'){PORTB |= 1 << PB2;PORTB |= 1 << PB5;};};}
// SI CARACTERE RECIBIDOS = ON3 ENTONCES ACTIVA PINB2 Y PINB5
if(caracteruno == 'O') {if(caracterdos == 'F'){ if(caractertres == '1'){PORTB &= ~(1 << PB0);PORTB &= ~(1 << PB3);};};}
// SI CARACTERE RECIBIDOS = ON1 ENTONCES DESACTIVA PINB0 Y PINB3
if(caracteruno == 'O') {if(caracterdos == 'F'){ if(caractertres == '2'){PORTB &= ~(1 << PB1);PORTB &= ~(1 << PB4);};};}

```

```
// SI CARACTERE RECIBIDOS = ON2 ENTONCES DESACTIVA PINB1 Y PINB4
```

```
if(caracteruno == 'O') {if(caracterdos == 'F'){ if(caractertres == '3'){PORTB &= ~(1 << PB2);PORTB &= ~(1 << PB5);};};}
```

```
// SI CARACTERE RECIBIDOS = ON3 ENTONCES DESACTIVA PINB2 Y PINB5
```

```
}
```

```
}
```