

CONTROL NEURODIFUSO DE UN MÓVIL AUTÓNOMO  
ANEXOS

CARLOS ARIEL CORDOBA

WILBER AUGUSTO NAVIA

UNIVERSIDAD DEL CAUCA  
FACULTAD DE INGENIERÍA ELECTRÓNICA Y TELECOMUNICACIONES  
DEPARTAMENTO DE ELECTRÓNICA, INSTRUMENTACIÓN Y CONTROL  
POPAYÁN

2003

# CONTROL NEURODIFUSO DE UN MÓVIL AUTONOMO

CARLOS ARIEL CORDOBA

WILBER AUGUSTO NAVIA

ANEXOS

Director

JUAN FERNANDO FLÓREZ

UNIVERSIDAD DEL CAUCA

FACULTAD DE INGENIERÍA ELECTRÓNICA Y TELECOMUNICACIONES

DEPARTAMENTO DE ELECTRÓNICA, INSTRUMENTACIÓN Y CONTROL

POPAYÁN

2003

## **ANEXO A**

### **Diagramas circuitales**

Se elaboró un circuito esquemático para cada modulo constitutivo del sistema, se tienen los siguientes módulos:

**MICROCONTROLADORES**, Muestra la parte "inteligente" del móvil, los microcontroladores se encargan de manejar los motores y activar los sensores de ultrasonido.

**MULTIPLEXOR\_US**, Corresponde a la parte conformada por un arreglo de switches análogos que permiten usar un solo amplificador para todos los sensores de ultrasonido.

**TRANSMISORES\_US**, Circuito elevador de corriente y voltaje para excitar los transmisores de ultrasonido.

**AMPLIFICADOR\_US**, Amplificador de voltaje de alta ganancia compuesto de 2 etapas en cascada.

**POTENCIA**, Circuito elevador de corriente para controlar los motores, también se muestra el regulador de voltaje para los motores y para los circuitos de los módulos anteriores.

## **ANEXO B**

### **Código embebido en Microcontroladores**

El robot tiene dos Microcontroladores PIC.

Código embebido en el Microcontrolador A: Controla transmisión y recepción de ultrasonido; también las funciones para medir la distancia

Código embebido en el Microcontrolador B: Controla el avance, giro y velocidad de los motores.

## Código embebido en el microcontrolador A

```

;4024us15 ID_PIC="a"
;
;-----
;Programa para controlar los Sensores de USound
;Este Plc se comunica con el PC, mide distancia y retorna la medida
;cuando el PC se lo solicita
;-----Version multiplexada-----
;Se mide la distancia con un detector de nivel
;Entra por el mismo pin, solo se han cambiando
;distancia minima (2cm)
;zona mueerta para este caso inexistente
;PORTA0-5 emisores de Usonido, 6bits

;PORTB1 Rx de usonido
;PORTB7 indicador audible
;PORTB2 indicador online
;PORTB3 indicadores usonido en uso
;PORTB4 bit de descarga de AmOperacional
;PORTB5 bit select 40Khz(H) o 24khz(L)

;PORTC0-6 controla los switches
;-----
LIST      p=16F873      ; Se emplea el PIC 16F873
RADIX    HEX          ; Sistema de numeraciøn hexadecimal

INCLUDE "P16f873.Inc"
;-----
;ENTRADAS
#define bRxUsLarga      btfsc   PORTB,0;in
#define bRxUsCorta     btfsc   PORTB,1;in
#define if_MODO_CORTA  btfss   USCTR,0;
#define if_MODO_LARGA  btfsc   USCTR,0;
;SALIDAS
#define MODO_CORTA_ON   bcf     USCTR,0;
#define MODO_LARGA_ON   bsf     USCTR,0;

#define bOnLine_ON     bcf     PORTB,2;out
#define bOnLine_OFF    bsf     PORTB,2;out
#define bUSound_ON     bcf     PORTB,2;out
#define bUSound_OFF    bsf     PORTB,2;out

#define bINDICADOR_ON  bsf     PORTB,3;out
#define bINDICADOR_OFF bcf     PORTB,3;out
#define bAlarma_ON     bsf     PORTB,3;out
#define bAlarma_OFF    bcf     PORTB,3;out
;-----
;definicion de constantes      HEX      ASCII
constant ID_PIC                =0x61    ;"a" ID de PIC para retorno
constant ID_PIC_IN             =0x00    ;"" IDentificador de PIC
constant INICIO_TR              =0xF0    ;"- " Inicio de trama

constant cINICIO_COM           =0x41    ;"A" Inicio de comunicacion
;ID funciones para ultrasonido
constant cMIN_DISTAN           =0x42    ;"B"
constant cNUM_CICLOS           =0x43    ;"C"
constant cTIM_MUERTO           =0x44    ;"D"
constant cMODO_US              =0x45    ;"E"
constant cUS_MED_CORTA         =0x46    ;"F"
constant cUS_MED_LARGA         =0x47    ;"G"
constant cUS_MED_DIRECTA       =0x48    ;"H"
constant cUS_LEER              =0x49    ;"I"
; "J"

```

```

; "K"
; "L"
; "M"
; ID funciones para motores
constant cMODO_PASO =0x4E ; "N"
constant cFACTOR_VEL =0x4F ; "O"
constant cFIJAR_VEL =0x50 ; "P"
constant cADELANTE =0x51 ; "Q"
constant cATRAS =0x52 ; "R"
constant cZQUIERDA =0x53 ; "S"
constant cDERECHA =0x54 ; "T"
constant cMOV_DIRECTO =0x55 ; "U"
; "V"
; "W"
; "X"
; "Y"
constant cFIN_COM =0x5A ; "Z" Fin de comunicacion
; Constantes para seleccionar modo
constant MODO_CORTA =0x2F ; "/"
constant MODO_LARGA =0x5C ; "\"

;-----
;definicion de tiempos(3*n microsegs)
constant TiempoAswitche =d'33' ;para que reaccione el switche
constant TempRx =d'16' ;Da la resolucione de la medida
;-----
;definicion de registros variables
BASE EQU 0x22
cblock BASE
Aux ;Acumulador auxiliar
TEMP_1 ;temporizadores
TEMP_2
Tiempo
Byte0 ;registros temporales de bytes de la trama
Byte1
Byte2
CONTR_RX ;contador de tramas recibidas
ID_PIC_RX ;id de pic interno
FUNCRX ;id de funcion recibido
VALOR ;datos de la funcion
RECOK ;bandera de inicio de trama
SEHRTR ;bandera de fin de trama
POSTR ;contador de bytes dentro de la trama
SBUF ;registro temporal de byte recibido
CONTR_IN ;contador de tramas interno

USCTR ;registro de control de ultrasonido
Cont ;contador de ciclos Tx por US
Distan ;guarda la ultima distancia medida
DisMin ;minima distancia a capaz de medir
NumCiclos ;# max de ciclos Tx por US
TempMuerto ;tiempo para salvar zona muerta
endc

;PROGRAMA PRINCIPAL

ORG 0 ; El programa comienza en la direcciøn 0 y salta a la
goto INICIO_ ; direcciøn 5 para sobrepasar el vector de interrupciøn
ORG 4
goto INTERRUPCIONES_

;INCLUDE "lcdf8xv1.asm" ;manejar pantallas LCD
;INCLUDE "eeprom.asm" ;W/R de memoria EEprom
;-----
TX_DATO_
clrwdt
bcf PIR1,TXIF
movwf TXREG
bsf STATUS,RP0

```

```

        bcf     STATUS,RP1
TX_DATO_W
        clrwdt
        btfs   RCSTA,TRMT    ;TXSTA
        goto   TX_DATO_W
        bcf     STATUS,RP0
        return
;-----
;
;           ATENCION DE INTERRUPCIONES, Formacion de la TRAMA
;-----
INTERRUPCIONES_
        clrwdt
        btfs   PIR1,RCIF    ;bandera de recepcion de byte completo
        goto   RETINT
        bcf     PIR1,RCIF
        movfw  RCREG        ;BYTE RECIBIDO
        movwf  SBUF
INT0
        MOVfw  POSTR        ;FORMACION DE LA TRAMA COMPUESTA DE 3 BYTES
        sublw  0x00
        BNZ   INT1        ;si es diferente salta, si es igual sigue
        movlw  b'11110000'
        andwf  SBUF,W
        sublw  INICIO_TR    ;nible alto es CARACTER DE INICIO DE TRAMA
        BNZ   INT3
        movlw  0x01
        MOVwf  RECOK        ;Bandera de sincronismo
        movlw  b'00001111'
        andwf  SBUF,W
        movwf  Byte0        ;nible bajo es el contador de tramas
        goto   INT3
INT1
        MOVfw  POSTR        ;CONTADOR DE BYTE DENTRO DE LA TRAMA
        sublw  0x01
        BNZ   INT2
        MOVfw  RECOK
        sublw  0x01
        BNZ   INT2
        MOVfw  SBUF
        MOVwf  Byte1        ;FUNCRX
        goto   INT3
INT2
        MOVfw  POSTR
        sublw  0x02
        BNZ   INT3
        MOVfw  RECOK
        sublw  0x01
        BNZ   INT3
        MOVfw  SBUF
        MOVwf  Byte2        ;VALOR, datos de la funcion
        goto   INT3
INT3
        INCf   POSTR,F      ;Control de recepcion de bytes
        MOVfw  POSTR
        sublw  0x03        ;trama de 3 bytes
        BNZ   INT4
        clrf   POSTR
        clrf   RECOK

        movfw  Byte0        ;Actualizacion de registros
        movwf  CONTR_RX
        movfw  Byte1        ;Actualizacion de registros
        movwf  FUNCRX
        movfw  Byte2
        movwf  VALOR
        movlw  0x01
        MOVwf  SEHRTR      ;BANDERA DE LLEGADA DE TRAMA COMPLETA
INT4
        MOVfw  RECOK        ;BANDERA DE BUENA RX DE TRAMAS

```

```

        sublw    0x00
        BNZ     RETINT
        clrf    POSTR

RETINT
        clrwdt
        retfie
;-----
;-----
INICIO_
        clrwdt
        bsf     STATUS,RP0      ;cambio al banco 1
        bcf     STATUS,RP1
;configuracion de I/O puertos

        movlw   b'00000000'
        movwf   PORTA          ;Txs de Usonido
        movlw   b'00000011'    ;estaba 0010 0001
        movwf   PORTB          ;Rx de Usonid leds,descarga,bselect
        movlw   b'10000000'
        movwf   PORTC          ;Activos Low,Enables de Sensores us
;config. comunicaciones seriales
        movlw   b'00100100'    ;config USART
        movwf   RCSTA          ;activa TX TXSTA
;BRGH=1 alta velocidad,
;9600 baudios a 4 mhz,SPBRG=25
        movlw   d'25'
        movwf   TXREG          ;SPBRG
        bsf     PIR1,RCIE      ;PIE1 habil. interrupt en RX
        bcf     STATUS,RP0     ;cambio a banco 0

        movlw   b'10010000'    ;Config USART
        movwf   RCSTA          ;RX continua, puesta en ON
;-----
        bsf     STATUS,RP0     ; Selección del banco 1
        movlw   b'10000011'
        movwf   TMR0          ;Asigna preescaler al WDT 1/32
        movlw   b'00000111'
        movwf   ADCON0        ;config PORTA y PORTE como I/O digitales
        bcf     STATUS,RP0     ;select banco 0
;-----
        clrf    PORTA
        clrf    PORTB
        clrf    PORTC

        clrf    CONTR_RX
        clrf    ID_PIC_RX
        clrf    FUNC_RX
        clrf    VALOR
        clrf    RECOK
        clrf    SEHRTR
        clrf    POSTR
        clrf    CONTR_IN
;configuracion de Tiempos por defecto
;minima distancia capaz de medir+1=1 cm
        movlw   d'0'
        movwf   DisMin
        movlw   d'7'
        movwf   NumCiclos     ;numero de coclos tx por Usonido
        movlw   d'7'
        movwf   TempMuerto   ;retardo para salvar zona muerta
        MODO_CORTA_ON
        bUSound_OFF          ;apaga led US trabajando
;Emite un sonido o hace parpadear un led

        clrwdt
        bINDICADOR_ON
        clrw
        call    DELAY_
        bINDICADOR_OFF
        movlw   b'11000000'
        movwf   INTCON        ;habil. Interrupt en general
;-----
;-----

```

```

;Rutinas de control principal del Programa;
;*****
;-----
FUNCIONES
    bOnLine_OFF
    bcf     INTCON,GIE      ;DESHabilitar INTERRUPCIONES
    MOVfw  SEHRTR          ;si se ha recibido trama, SEHRTR=0x01
    sublw  0x01            ;si no, no procesa ninguna funcion
    BNZ    RETFUN
    clrwdt
    CALL   SINCRONIZAR_
    movfw  ID_PIC_RX       ;revisa si la trama es para mi
    sublw  ID_PIC_IN       ;si no es para mi la descarta
    BNZ    RETFUN
    goto   FUN1            ;si es para mi paso a ejecutar funciones
RETFUN
    clrwdt
    bOnLine_ON
    clrf   ID_PIC_RX
    clrf   FUNCRX
    clrf   SEHRTR
    bsf    RCSTA,CREN
    bsf    INTCON,GIE      ;Habilitar INTERRUPCIONES
    goto   FUNCIONES
FUN1  MOVfw  FUNCRX
    sublw  cNICIO_COM      ;retorna el ID del pic, para gestion
    BNZ    FUN2
    CALL   fInicioCom_
    goto   RETFUN
FUN2  MOVfw  FUNCRX
    sublw  cMIN_DISTAN     ;modificar distancia minima capaz de medir
    BNZ    FUN3
    CALL   fMinDistan_
    goto   RETFUN
FUN3  MOVfw  FUNCRX
    sublw  cNUM_CICLOS     ;modificar numero de ciclos transmitidos
    BNZ    FUN4
    CALL   fNumCiclos_
    goto   RETFUN
FUN4  MOVfw  FUNCRX
    sublw  cTIM_MUERTO     ;modificar tiempo para salvar zona muerta
    BNZ    FUN5
    CALL   fTimMuerto_
    goto   RETFUN
FUN5  MOVfw  FUNCRX
    sublw  cMODO_US        ;elige modo largas o modo cortas
    BNZ    FUN6
    CALL   fModoUS_
    goto   RETFUN
FUN6  MOVfw  FUNCRX
    sublw  cUS_MED_CORTA  ;mide cortas distancias
    BNZ    FUN7
    CALL   fUSMedCorta_
    goto   RETFUN
FUN7  MOVfw  FUNCRX
    sublw  cUS_MED_LARGA  ;mide largas distancias
    BNZ    FUN8
    CALL   fUSMedLarga_
    goto   RETFUN
FUN8  MOVfw  FUNCRX
    sublw  cUS_MED_DIRECTA ;medida de distancia con ultima configuracion

```

```

        BNZ     FUN9
CALL    fUSMedDirecta_
        goto   RETFUN

FUN9    MOVfw  FUNCRX
        sublw  cUS_LEER      ;retorna ultima distancia medida
        BNZ     FUN10
        CALL    fUSLeer_
        goto   RETFUN

FUN10   MOVfw  FUNCRX
        sublw  cFIN_COM      ;beep-beep
        BNZ     RETFUN
        CALL    fFinCom_
        goto   RETFUN
;-----
SINCRONIZAR_
        movlw  b'10000000'
        andwf  FUNCRX,W
        movwf  ID_PIC_RX
        bcf    FUNCRX,7

        INCf   CONTR_IN,F    ;incrementa contador de tramas interno
        movlw  0x0F
        andwf  CONTR_IN,W
        subwf  CONTR_RX,W    ;contador de tramas de llegada
        BNZ     SINCR1
        bAlarma_OFF
        goto  RETSINC

SINCR1  bAlarma_ON          ;alarma de perdida de sincronismo
        movfw  CONTR_RX
        MOVwf  CONTR_IN

RETSINC
        RETurn
;-----
;                                     IMPLEMENTACION de FUNCIONES
;-----
fInicioCom_
        bcf    RCSTA,CREN
        movlw  ID_PIC        ;Se envia al PC Byte de ID del PIC
        call   TX_DATO_      ;Transmite byte al PC
        clrwdt
        bINDICADOR_ON
        movfw  VALOR
        call   DELAY_
        bINDICADOR_OFF
        return
;-----
fMinDistan_
        bcf    RCSTA,CREN
        clrwdt
        movfw  VALOR          ;Minima distancia capaz de medir
        movwf  DisMin
        movfw  FUNCRX        ;Se retorna al PC FUNCRX enviado por PC
        call   TX_DATO_      ;para verificar comunicacion
        return
;-----
fNumCiclos_
        bcf    RCSTA,CREN
        clrwdt
        movfw  VALOR          ;Numero de ciclos a transmitir por US
        movwf  NumCiclos
        movfw  FUNCRX        ;Se retorna al PC FUNCRX enviado por PC
        call   TX_DATO_      ;para verificar comunicacion
        return
;-----
fTimMuerto_
        bcf    RCSTA,CREN

```

```

        clrwdt
        movfw VALOR           ;Retardo para salvar zona muerta
        movwf TempMuerto
        movfw FUNCRIX        ;Se retorna al PC FUNCRIX enviado por PC
        call TX_DATO_       ;para verificar comunicacion
        return
;-----
fModoUS_
        bcf RCSTA,CREN
        clrwdt
        movfw VALOR           ;Parametro de la funcion
        sublw MODO_CORTA
        BNZ FMOD1
        MODO_CORTA_ON        ;Activa modo medida CORTA distancia
        goto FMOD2
FMOD1
        movfw VALOR
        sublw MODO_LARGA
        BNZ FMOD2
        MODO_LARGA_ON        ;Activa modo medida LARGA distancia
FMOD2
        movfw FUNCRIX        ;Se retorna al PC FUNCRIX enviado por PC
        call TX_DATO_       ;para verificar comunicacion
        return
;-----
fUSMedCorta_
        bcf RCSTA,CREN
        movlw d'0'           ;Minima distancia capaz de medir-2=2cm
        movwf DisMin
        movlw d'7'           ;Numero de ciclos transmitidos
        movwf NumCiclos
        movlw d'7'           ;Retardo para salvar zona muerta
        movwf TempMuerto
        MODO_CORTA_ON        ;Activa funcion de medidas cortas
        call UltraSonido_
        movfw Distan         ;Se retorna al PC distancia medida
        call TX_DATO_
        return
;-----
fUSMedLarga_
        bcf RCSTA,CREN
        movlw d'23'          ;Minima distancia capaz de medir-2=25
        movwf DisMin
        movlw d'9'           ;Numero de ciclos transmitidos
        movwf NumCiclos
        movlw d'232'         ;Retardo para salvar zona muerta
        movwf TempMuerto
        MODO_LARGA_ON        ;Activa funcion de medidas largas
        call UltraSonido_
        movfw Distan         ;Se retorna al PC distancia medida
        call TX_DATO_
        return
;-----
fUSMedDirecta_
        bcf RCSTA,CREN
        call UltraSonido_
        movfw Distan         ;Se retorna al PC ultima Distancia medida
        call TX_DATO_
        return
;-----
fUSLeer_
        bcf RCSTA,CREN
        movfw Distan         ;Se envia al PC la ultima medida de ditancia
        call TX_DATO_       ;Transmite medida al PC
        return
;-----
fFinCom_
        bcf RCSTA,CREN
        bsf VALOR,5

```

```

movfw VALOR ;Se envia al PC el ID de la funcion FinCom
call TX_DATO_ ;Transmite byte al PC
;Emite sonido o parapade un LED

clrwdt
bINDICADOR_ON
movfw VALOR
call DELAY_
bINDICADOR_OFF
clr
call DELAY_
bINDICADOR_ON
movfw VALOR
call DELAY_
bINDICADOR_OFF
return ;si llego fin de trama vuelve a esperar inicio
;*****
;
; RUTINAS para ULTRA SONIDO
;*****
;-----
; Control principal de medicion con Ultrasonido
;-----
UltraSonido_
clrwdt
comf VALOR,W ;Desactiva los sensores innecesarios
movwf PORTC ;activos bajos

movlw 0x40 ;revisa si se desea activar sensor 7
subwf VALOR,W
BNZ US1
bsf PORTB,5
bcf PORTB,4 ;activa switch sensor 7
goto US3
US1
movlw 0x80 ;revisa si se desea activar sensor 8
subwf VALOR,W
BNZ US3
bcf PORTB,5 ;activa switch sensor 8
bsf PORTB,4
US3
movlw TiempoAswitche ;SE DA TIEMPO AL SWITCHE PARA QUE REACCIONE
clrwdt
call TimerW_

movfw NumCiclos
movwf Cont

movlw 0x80
subwf VALOR,W
BNZ US4
call TX_US_40k8_ ;Tx usonido de 40khz para sensor8,PORTB-7
goto US6
US4
movlw 0x40
subwf VALOR,W
BNZ US5
call TX_US_40k7_ ;Tx usonido de 40khz para sensor7,PORTB-6
goto US6
US5
call TX_US_40k_ ;por defecto
;Tx usonido de 40khz, PORTA-0-5
US6
bUSound_ON ;Fija el bit de US trabajando

movfw TempMuerto
clrwdt
call TimerW_ ;espera un tiempo para salvar zona muerta
movfw TempMuerto
clrwdt
call TimerW_ ;espera un tiempo para salvar zona muerta

```

```

movfw  DisMin
movwf  Distan          ;distancia minima capaz de medir
if_MODO_CORTA          ;la distancia medida se almacena en Distan
call   Medir_Dis_Corta_
if_MODO_LARGA
call   Medir_Dis_Larga_

clrf   PORTC           ;activa todos los SWitches de los sensores
bcf    PORTB,5         ;activa SWitche sensor 8
bcf    PORTB,4         ;activa SWitche sensor 7
return

;-----
;-----
;                               Mide la distancia de acuerdo al retardo del eco
;-----
;-----
Medir_Dis_Corta_
incf   Distan,F        ;se incrementa de cm en cm
btfsc  Distan,7        ;distancia maxima capaz de medir 128cm
goto   MDisC0
movlw  TempRx          ;tiempo que da la resolucio de la medida
clrwdt
call   TimerW_         ;calculada para 1 cm
bRxUsCorta          ;revisa si lleo el eco, activos bajos
goto   Medir_Dis_Corta_ ;si no ha llegado sigue contando
bRxUsCorta          ;otra revision para confirmar
goto   Medir_Dis_Corta_
;si ya lleo entonces finaliza

MDisC0
bUSound_OFF          ;Pone a uno bit de medicion
movfw  Distan
sublw  d'129'
movwf  Aux

MDisC1
decfsz Aux,F
goto   MDisC2
goto   MDisC3

MDisC2
movlw  TempRx          ;tiempo de espera de compensacion
clrwdt
call   TimerW_         ;para que el tiempo de medida sea constante
goto   MDisC1          ;sin importar la distancia al obstaculo

MDisC3
movlw  0x20            ;se suma 32 a distan para enviarla al PC
addwf  Distan,F        ;en el PC se resta este valor
return

;-----
;-----
Medir_Dis_Larga_
incf   Distan,F        ;se incrementa de cm en cm
btfsc  Distan,7        ;distancia maxima capaz de medir 128cm
goto   MDisL0
movlw  TempRx          ;tiempo que da la resolucio de la medida
clrwdt
call   TimerW_         ;calculada para 1 cm
bRxUsLarga          ;revisa si lleo el eco, activos bajos
goto   Medir_Dis_Larga_ ;si no ha llegado sigue contando
bRxUsLarga          ;otra revision para confirmar
goto   Medir_Dis_Larga_
;si ya lleo entonces finaliza

MDisL0
bUSound_OFF          ;pone en uno bit de medicion
movfw  Distan
sublw  d'129'
movwf  Aux

MDisL1
decfsz Aux,F
goto   MDisL2
goto   MDisL3

MDisL2

```

```

        movlw   TempRx           ;tiempo de espera de compensacion
        clrwdt
        call   TimerW_          ;para que el tiempo de medida sea constante
        goto   MDisL1          ;sin importar la distancia al obstaculo
MDisL3
        movlw   0x20             ;se suma 32 a distan para enviarla al PC
        addwf  Distan,F         ;en el PC se resta este valor
        return

;-----
;*****
;          RUTINAS para ULTRA SONIDO
;*****
;esta rutina genera pulsos de 40 khz o 25 mics de periodo
TX_US_40k_
        movfw   VALOR           ;12
        movwf  PORTA           ;13      ;puertoA es para tx usonido 6bit de menor peso
        nop    ;1
        nop    ;2
        nop    ;3
        nop    ;4
        clrwdt ;5
        nop    ;6
        nop    ;7
        nop    ;8
        nop    ;9
        nop    ;10
        nop    ;11
        clrf   PORTA           ;12      ;activos altos
        nop    ;1
        nop    ;2
        nop    ;3
        nop    ;4
        nop    ;5
        nop    ;6
        nop    ;7
        nop    ;8
        decfsz Cont,F          ;9      ;cuenta el numero de ciclos emitidos
        goto   TX_US_40k_     ;10-11
        return

;----- Tx para sensor 7
TX_US_40k7_
        nop    ;12
        bsf   PORTB,6         ;13      ;puertoA es para tx usonido 6bit de menor peso
        nop    ;1
        nop    ;2
        nop    ;3
        nop    ;4
        clrwdt ;5
        nop    ;6
        nop    ;7
        nop    ;8
        nop    ;9
        nop    ;10
        nop    ;11
        bcf   PORTB,6         ;12      ;activos altos
        nop    ;1
        nop    ;2
        nop    ;3
        nop    ;4
        nop    ;5
        nop    ;6
        nop    ;7
        nop    ;8
        decfsz Cont,F          ;9      ;cuenta el numero de ciclos emitidos
        goto   TX_US_40k7_     ;10-11
        return

;----- Tx para sensor 8
TX_US_40k8_
        nop    ;12

```



## Código embebido en el microcontrolador B

```

;pmotors3.asm ID_PIC="b"
;Programa para CONTROLAR LOS MOTORES P:P DE LAS LLANTAS
;Este Plc se comunica con el PC
;El PC envia direccion y numero de pasos
;se puede configurar desde el PC la velocidad, modo un paso o medio paso

;PORTA0-3 Control de Motor IZQ

;PORTB1 indicador motores trabajando
;PORTB2 indicador OnLine, supervivencia
;PORTB3 bit de ALARMA, de perdida de sincronismo
;PORTA4 indicador audible, al dar Reset, inicio o fin de com

;PORTC0-3 Control de Motor DER
;-----
LIST      p=16F873      ; Se emplea el PIC 16F873
RADIX     HEX          ; Sistema de numeración hexadecimal

INCLUDE "P16f873.Inc"

;Definicion de Etiquetas
;-----
;ENTRADAS
#define if_MEDIO_PASO  btfss  MOTCTR,0      ;
#define if_UN_PASO     btfsc  MOTCTR,0      ;
#define MEDIO_PASO_ON bcf    MOTCTR,0      ;
#define UN_PASO_ON    bsf    MOTCTR,0      ;

#define bMotores_ON    bcf    PORTB,1 ;out
#define bMotores_OFF   bsf    PORTB,1 ;out

#define bOnLine_ON     bcf    PORTB,2 ;out
#define bOnLine_OFF    bsf    PORTB,2 ;out

#define bINDICADOR_ON bsf    PORTB,3 ;out
#define bINDICADOR_OFF bcf    PORTB,3 ;out estaba portb1
#define bAlarma_ON     bsf    PORTA,4 ;out
#define bAlarma_OFF    bcf    PORTA,4 ;out

;-----
;Definicion de constantes
;-----
constant ID_PIC      =0x62  ;"b" ID de PIC para retorno
constant ID_PIC_IN   =0x80  ;"" IDentificador de PIC
constant INICIO_TR    =0xF0  ;"- " Inicio de trama

constant cNICIO_COM  =0x41  ;"A" Inicio de comunicacion
;ID funciones para ultrasonido
constant cMIN_DISTAN =0x42  ;"B"
constant cNUM_CICLOS =0x43  ;"C"
constant cTIM_MUERTO =0x44  ;"D"
constant cMODO_US     =0x45  ;"E"
constant cUS_MED_CORTA=0x46  ;"F"
constant cUS_MED_LARGA=0x47  ;"G"
constant cUS_MED_DIRECTA =0x48 ;"H"
constant cUS_LEER     =0x49  ;"I"
; "J"
; "K"
; "L"
; "M"
;ID funciones para motores
constant cMODO_PASO  =0x4E  ;"N"
constant cFACTOR_VEL =0x4F  ;"O"
constant cFIJAR_VEL  =0x50  ;"P"
constant cADELANTE   =0x51  ;"Q"
constant cATRAS      =0x52  ;"R"
;SALIDAS

```

```

constant cIZQUIERDA    =0x53    ;"S"
constant cDERECHA      =0x54    ;"T"
constant cMOV_DIRECTO  =0x55    ;"U"
                                ;"V"
                                ;"W"
                                ;"X"
                                ;"Y"
constant cFIN_COM      =0x5A    ;"Z" Fin de comunicacion
                                ;Constantes para seleccionar modo
constant MODO_CORTA    =0x2F    ;"/"
constant MODO_LARGA    =0x5C    ;"\\"

constant MEDIO_PASO    =0x2F    ;Para configurar desde el PC
constant UN_PASO       =0x5C    ;medio paso o un paso

```

```

;Definicion de tiempos(3*n microsegs)
constant TiempoAswitche =d'33' ;para que reaccione el switch
constant TempRx         =d'16' ;Da la resolucio de la medid
                                ;calibrado para 1 cm

```

```

;Definicion de registros variables

```

```

BASE EQU 0x22
cblock
    BASE
    Aux ;Acumulador auxiliar
    TEMP_1 ;temporizadores
    TEMP_2
    Tiempo
    Byte0 ;registros temporales de bytes de la trama
    Byte1
    Byte2
    CONTR_RX ;contador de tramas recibidas
    ID_PIC_RX ;id de pic interno
    FUNC_RX ;id de funcion recibido
    VALOR ;datos de la funcion
    RECOK ;bandera de inicio de trama
    SEHRTR ;bandera de fin de trama
    POSTR ;contador de bytes dentro de la trama
    SBUF ;registro temporal de byte recibido
    CONTR_IN ;contador de tramas interno

    MOTCTR ;registro de control de motores
    MaxPasos
    FactorTimer
    Velocidad
    NumPasos
    SecMot1
    SecMot2
    ContSecMot1
    ContSecMot2
    ContSec
endc

```

```

ORG 0 ; El prog comienza en la dir 0 y salta a la
goto INICIO_ ; dir 5 para sobrepasar el vector de interrup
ORG 4
goto INTERRUPCIONES_

;INCLUDE "lcdf8xv1.asm" ;manejar pantallas LCD
;INCLUDE "eeprom.asm" ;W/R de memoria EEprom

```

```

-----
; Implementacion de rutinas

```

```

-----
; TRANSMITE UN BYTE POR EL PUERTO SERIAL

```

```

TX_DATO_ ;El byte a Tx se pasa en w
    clrwdt
    bcf PIR1,TXIF
    movwf TXREG

```

```

        bsf     STATUS,RP0
        bcf     STATUS,RP1
TXDAT
        clrwdt
        btfs   RCSTA,TRMT      ;TXSTA
        goto   TXDAT          ;espera bandera de fin de Tx
        bcf     STATUS,RP0
        clrwdt
        return
;-----
;
;          ATENCION DE INTERRUPCIONES, Formacion de la TRAMA
;-----
INTERRUPCIONES_
        clrwdt
        btfs   PIR1,RCIF      ;bandera de recepcion de byte completo
        goto   RETINT
        bcf     PIR1,RCIF
        movfw  RCREG          ;BYTE RECIBIDO
        movwf  SBUF
INT0
        MOVfw  POSTR
        sublw  0x00
        BNZ   INT1           ;si es diferente salta, si es igual sigue
        movlw  b'11110000'
        andwf  SBUF,W
        sublw  INICIO_TR     ;nible alto es CARACTER DE INICIO DE TRAMA
        BNZ   INT3
        movlw  0x01
        MOVwf  RECOK        ;Bandera de sincronismo
        movlw  b'00001111'
        andwf  SBUF,W
        movwf  Byte0        ;nible bajo es el contador de tramas
        goto  INT3
INT1
        MOVfw  POSTR        ;CONTADOR DE BYTE DENTRO DE LA TRAMA
        sublw  0x01
        BNZ   INT2
        MOVfw  RECOK
        sublw  0x01
        BNZ   INT2
        MOVfw  SBUF
        MOVwf  Byte1        ;FUNCRX
        goto  INT3
INT2
        MOVfw  POSTR
        sublw  0x02
        BNZ   INT3
        MOVfw  RECOK
        sublw  0x01
        BNZ   INT3
        MOVfw  SBUF
        MOVwf  Byte2        ;VALOR, datos de la funcion
        goto  INT3
INT3
        INCf   POSTR,F      ;Control de recepcion de bytes
        MOVfw  POSTR
        sublw  0x03        ;trama de 3 bytes
        BNZ   INT4
        clrf  POSTR
        clrf  RECOK

        movfw  Byte0        ;Actualizacion de registros
        movwf  CONTR_RX
        movfw  Byte1        ;Actualizacion de registros
        movwf  FUNCRX
        movfw  Byte2
        movwf  VALOR
        movlw  0x01
        MOVwf  SEHRTR      ;BANDERA DE LLEGADA DE TRAMA COMPLETA

```

```

INT4    MOVfw  RECOK           ;BANDERA DE BUENA RX DE TRAMAS
        sublw 0x00
        BNZ   RETINT
        clrf  POSTR

RETINT
        clrwdt
        retfie

;-----
;
;          INICIALIZACION DEL SISTEMA
;-----
INICIO_
        clrwdt
        bsf   STATUS,RP0      ;cambio al banco 1
        bcf   STATUS,RP1
        ;configuracion de I/O puertos

        movlw b'00000000'
        movwf PORTA          ;Txs de Usonido
        movlw b'00000000'    ;estaba 0010 0010
        movwf PORTB          ;Rx de Usonid leds,descarga,bselect
        movlw b'10000000'
        movwf PORTC          ;Activos Low,Enables de Sensores us
        ;config. comunicaciones seriales
        movlw b'00100100'    ;config USART
        movwf RCSTA          ;activa TX TXSTA
        ;BRGH=1 alta velocidad, SPBRG=25
        movlw d'25'          ;9600 baudios a 4 mhz
        movwf TXREG          ;SPBRG
        bsf   PIR1,RCIE      ;PIE1 habil. interrupt en RX

        bcf   STATUS,RP0      ;cambio a banco 0
        movlw b'10010000'    ;Config USART
        movwf RCSTA          ;RX continua, puesta en ON

;-----
        bsf   STATUS,RP0      ; Seleccin del banco 1
        movlw b'10000011'
        movwf TMR0           ;Asigna preescaler al TMR0 1/16
        movlw b'00000111'
        movwf ADCON0         ;config PORTA y PORTE como I/O digitales
        bcf   STATUS,RP0      ;select banco 0

;-----
        clrf  PORTA
        clrf  PORTB
        clrf  PORTC

        clrf  CONTR_RX
        clrf  ID_PIC_RX
        clrf  FUNC_RX
        clrf  VALOR
        clrf  RECOK
        clrf  SEHRTR
        clrf  POSTR
        clrf  CONTR_IN
        clrf  MOTCTR          ;por defecto medio paso_ON

UN_PASO_ON
        movlw d'4'           ;modo medio paso
        movwf MaxPasos
        movlw d'198'         ;Timer de (256-198)*16=928 mics
        movwf FactorTimer    ;mas el tiempo de procesamiento da 1ms
        movlw d'30'
        movwf Velocidad;PASO/30 ms
        clrf  NumPasos
        clrf  SecMot1
        clrf  SecMot2
        clrf  ContSecMot1
        clrf  ContSecMot2
        ;Emite un sonido o hace parpadear un led

        clrwdt

```

```

        bINDICADOR_ON
        clrw
        call    DELAY_
        bINDICADOR_OFF

        movlw  b'11000000'
        movwf  INTCON          ;Habil. Interrupt de perifericos en general
FIN_INICIO  goto    FUNCIONES ;DEShabil T0IF

;-----
;                               TABLAS DE SECUENCIAS
;-----
TSecuencia8          ;Secuencia para medio paso
        movfw  ContSec
        andlw  0x07          ;acotar el desplazamiento de PCL
        addwf  PCL,F
        retlw  0x04
        retlw  0x06
        retlw  0x02
        retlw  0x0A
        retlw  0x08
        retlw  0x09
        retlw  0x01
        retlw  0x05

;-----
TSecuencia4          ;Secuencia para paso completom, manejo de 2 fases
        movfw  ContSec
        andlw  0x03          ;acotar el desplazamiento de PCL
        addwf  PCL,F
        retlw  0x0A
        retlw  0x09
        retlw  0x05
        retlw  0x06

;-----
;                               RUTINAS DE CONTROL principal del Programa
;-----
FUNCIONES
        bOnLine_OFF
        bcf    INTCON,GIE    ;DESHabilitar INTERRUPCIONES
MOVfw    SEHRTR              ;si se ha recibido trama, SEHRTR=0x01
        sublw  0x01          ;si no, no procesa ninguna funcion
        BNZ   RETFUN
        clrwdt
        CALL  SINCRONIZAR_
        movfw  ID_PIC_RX     ;revisa si la trama es para mi
        sublw  ID_PIC_IN     ;si no es para mi la descarta
        BNZ   RETFUN
        goto  FUN1
RETFUN
        clrwdt
        bOnLine_ON
        clrf  ID_PIC_RX
        clrf  FUNCRX
        clrf  SEHRTR
        bsf   RCSTA,CREN
        bsf   INTCON,GIE    ;Habilitar INTERRUPCIONES
        goto  FUNCIONES

FUN1    MOVfw  FUNCRX
        sublw  cINICIO_COM   ;revisa si se invoca a la funcion InicioCom
        BNZ   FUN2
        CALL  flnicioCom_
        goto  RETFUN

FUN2    MOVfw  FUNCRX
        sublw  cMODULO_PASO
        BNZ   FUN3

```

```

CALL    goto    fFijarPaso_
          RETFUN

FUN3    MOVfw   FUNCRX
          sublw  cFACTOR_VEL
          BNZ    FUN4
CALL    goto    fFactorVel_
          RETFUN

FUN4    MOVfw   FUNCRX
          sublw  cFIJAR_VEL
          BNZ    FUN5
CALL    goto    fFijarVel_
          RETFUN

FUN5    MOVfw   FUNCRX
          sublw  cADELANTE
          BNZ    FUN6
CALL    goto    fAdelante_
          RETFUN

FUN6    MOVfw   FUNCRX
          sublw  cATRAS
          BNZ    FUN7
CALL    goto    fAtras_
          RETFUN

FUN7    MOVfw   FUNCRX
          sublw  cIZQUIERDA
          BNZ    FUN8
CALL    goto    flzquierda_
          RETFUN

FUN8    MOVfw   FUNCRX
          sublw  cDERECHA
          BNZ    FUN9
CALL    goto    fDerecha_
          RETFUN

FUN9    MOVfw   FUNCRX
          sublw  cMOV_DIRECTO
          BNZ    FUN10
CALL    goto    fDirecto_
          RETFUN

FUN10   MOVfw   FUNCRX
          sublw  cFIN_COM           ;si no, no procesa ninguna funcion
          BNZ    RETFUN
CALL    goto    fFinCom_
          RETFUN

;-----
SINCRONIZAR_
movlw   b'10000000'
andwf   FUNCRX,W
movwf   ID_PIC_RX
bcf     FUNCRX,7

        INCf    CONTR_IN,F           ;incrementa contador de tramas interno
movlw   0x0F
andwf   CONTR_IN,W
subwf   CONTR_RX,W                 ;contador de tramas de llegada
BNZ     SINCR1
        bAlarma_OFF
goto    RETSINC

SINCR1  bAlarma_ON                   ;alarma de perdida de sincronismo
movfw   CONTR_RX
MOVwf   CONTR_IN
RETSINC

```

```

RETurn
;-----
;                                     Implementacion de funciones
;-----
fDirecto_      ;modifica el niblo bajo del puerto A y C
               ;directamente desde el PC
    movfw     VALOR
    andlw     0x0F
    movwf     PORTA
    swapf     VALOR,W           ;invierte nibles
    andlw     0x0F
    movwf     PORTC
    return

;-----
fInicioCom_
    bcf       RCSTA,CREN
    clrwdt
    movlw     ID_PIC           ;Se envia al PC el ID del PIC
    call      TX_DATO_        ;para verificar la comunicacion
    bINDICADOR_ON           ;Emite sonido o parapade un LED
    movfw     VALOR
    call      DELAY_
    bINDICADOR_OFF
    return

;-----
fFinCom_
    bcf       RCSTA,CREN
    bsf       VALOR,5
    movfw     VALOR           ;envia al PC el VALOR modificado en un bit
    call      TX_DATO_        ;para verificar la comunicacion
    clrwdt
    bINDICADOR_ON           ;Emite sonido o parapade un LED
    movfw     VALOR
    call      DELAY_
    bINDICADOR_OFF
    clrw
    call      DELAY_
    bINDICADOR_ON
    movfw     VALOR
    call      DELAY_
    bINDICADOR_OFF
    return

;-----
fFijarPaso_
    bcf       RCSTA,CREN
    clrwdt
    movfw     VALOR           ;Parametro de la funcion
    sublw     MEDIO_PASO
    BNZ       FPAS1
    movlw     d'8'
    movwf     MaxPasos
    MEDIO_PASO_ON
    goto      FPAS2

FPAS1
    movfw     VALOR
    sublw     UN_PASO
    BNZ       FPAS2
    movlw     d'4'
    movwf     MaxPasos
    UN_PASO_ON

FPAS2
    movfw     FUNCRIX           ;Se retorna al PC FUNCRIX enviado por PC
    call      TX_DATO_        ;para verificar el fin dl movimiento
    return

;-----
fFactorVel_
    bcf       RCSTA,CREN
    clrwdt
    movfw     VALOR
    movwf     FactorTimer

```

```

movfw  FUNCRX      ;Se retorna al PC FUNCRX enviado por PC
call   TX_DATO_   ;para verificar el fin dl movimiento
return

;-----
fFijarVel_
bcf    RCSTA,CREN
clrwdt
movfw  VALOR
movwf  Velocidad
movfw  FUNCRX      ;Se retorna al PC FUNCRX enviado por PC
call   TX_DATO_   ;para verificar el fin dl movimiento
return

;-----
fAdelante_
bcf    RCSTA,CREN
movfw  VALOR
movwf  NumPasos

ADELO
clrwdt
decfsz NumPasos,F
goto  ADEL1
goto  ADEL2

ADEL1
call   INC_PASO_MOT1_
call   DEC_PASO_MOT2_
movfw  SecMot1
movwf  PORTA
movfw  SecMot2
movwf  PORTC
movfw  Velocidad
CALL   TimerVel_
goto  ADEL0

ADEL2
movfw  FUNCRX      ;Se retorna al PC FUNCRX enviado por PC
call   TX_DATO_   ;para verificar el fin dl movimiento
return

;-----
fAtras_
bcf    RCSTA,CREN
movfw  VALOR
movwf  NumPasos

ATRA0
clrwdt
decfsz NumPasos,F
goto  ATRA1
goto  ATRA2

ATRA1
call   DEC_PASO_MOT1_
call   INC_PASO_MOT2_
movfw  SecMot1
movwf  PORTA
movfw  SecMot2
movwf  PORTC
movfw  Velocidad
CALL   TimerVel_
goto  ATRA0

ATRA2
movfw  FUNCRX      ;Se retorna al PC FUNCRX enviado por PC
call   TX_DATO_   ;para verificar el fin dl movimiento
return

;-----
flzquierda_
bcf    RCSTA,CREN
movfw  VALOR
movwf  NumPasos      ;NUMERO de paso a moverse

IZQ0
clrwdt
decfsz NumPasos,F
goto  IZQ1

```

```

IZQ1      goto      IZQ2
          call      INC_PASO_MOT1_      ;control de la secuencia valida
          call      INC_PASO_MOT2_
          movfw    SecMot1
          movwf    PORTA
          movfw    SecMot2
          movwf    PORTC
          movfw    Velocidad;retardo para control de velocidad
          CALL     TimerVel_
          goto     IZQ0

IZQ2      movfw    FUNCRIX      ;Se retorna al PC FUNCRIX enviado por PC
          call     TX_DATO_      ;para verificar el fin dl movimiento
          return

```

```

;-----
fDerecha_
          bcf      RCSTA,CREN
          movfw    VALOR
          movwf    NumPasos      ;NUMERO de paso a moverse
DER0

```

```

          clrwdt
          decfsz   NumPasos,F
          goto     DER1
          goto     DER2
DER1

```

```

          call     DEC_PASO_MOT1_      ;control de la secuencia valida
          call     DEC_PASO_MOT2_
          movfw    SecMot1
          movwf    PORTA
          movfw    SecMot2
          movwf    PORTC
          movfw    Velocidad;retardo para control de velocidad
          CALL     TimerVel_
          goto     DER0
DER2

```

```

          movfw    FUNCRIX      ;Se retorna al PC FUNCRIX enviado por PC
          call     TX_DATO_      ;para verificar el fin dl movimiento
          return

```

```

;-----
;
;                               RUTINAS para MOVER MOTORES
;
;-----

```

```

;-----
;                               Control principal de MOVIMIENTO de motor PP
;-----

```

```

DEC_PASO_MOT1_
          decf     ContSecMot1,F
          movfw    ContSecMot1
          sublw   0xFF
          BNZ     DPM11
          decf     MaxPasos,W
          movwf    ContSecMot1
DPM11

```

```

          movfw    ContSecMot1
          movwf    ContSec
          if_MEDIO_PASO
          goto     DPM12
          goto     DPM13
DPM12

```

```

          CALL     TSecuencia8
          goto     DPM14
DPM13

```

```

          CALL     TSecuencia4
DPM14

```

```

          movwf    SecMot1
          return
;-----

```

```

INC_PASO_MOT1_

```

```

        incf    ContSecMot1,F
        movfw  ContSecMot1
        subwf  MaxPasos,W
        BNZ   IPM11
        clrf   ContSecMot1
IPM11   movfw  ContSecMot1
        movwf  ContSec
        if_MEDIO_PASO
        goto  IPM12
        goto  IPM13
IPM12   CALL   TSecuencia8
        goto  IPM14
IPM13   CALL   TSecuencia4
IPM14   movwf  SecMot1
        return
;-----
DEC_PASO_MOT2_
        decf   ContSecMot2,F
        movfw  ContSecMot2
        sublw  0xFF
        BNZ   DPM21
        decf   MaxPasos,W
        movwf  ContSecMot2
DPM21   movfw  ContSecMot2
        movwf  ContSec
        if_MEDIO_PASO
        goto  DPM22
        goto  DPM23
DPM22   CALL   TSecuencia8
        goto  DPM24
DPM23   CALL   TSecuencia4
DPM24   movwf  SecMot2
        return
;-----
INC_PASO_MOT2_
        incf   ContSecMot2,F
        movfw  ContSecMot2
        subwf  MaxPasos,W
        BNZ   IPM21
        clrf   ContSecMot2
IPM21   movfw  ContSecMot2
        movwf  ContSec
        if_MEDIO_PASO
        goto  IPM22
        goto  IPM23
IPM22   CALL   TSecuencia8
        goto  IPM24
IPM23   CALL   TSecuencia4
IPM24   movwf  SecMot2
        return
;-----
*****
;
;                               RUTINAS de Temporizacion
;-----
DELAY_  movwf  TEMP_1
        clrf  TEMP_2

```

```

DELAY1  clrwdt                ;1 Maxima temp=255*4*255 micrseg=262 ms
        decfsz    TEMP_2,F    ;2
        goto     DELAY1      ;3,4
        decfsz    TEMP_1,F
        goto     DELAY1
        return

;-----
TimerW_
Timer1  movwf    Tiempo        ;retardo de (Tiempo-1)*3+7 microsegs
        decfsz    Tiempo,F
        goto     Timer1
        return

;-----
TimerVel_
TIV0    movwf    Tiempo        ;retardo de (Tiempo-1)*3+7 microsegs
        bsf      INTCON,T0IE
        bcf      INTCON,T0IF
        movfw    FactorTimer
        movwf    TMR0

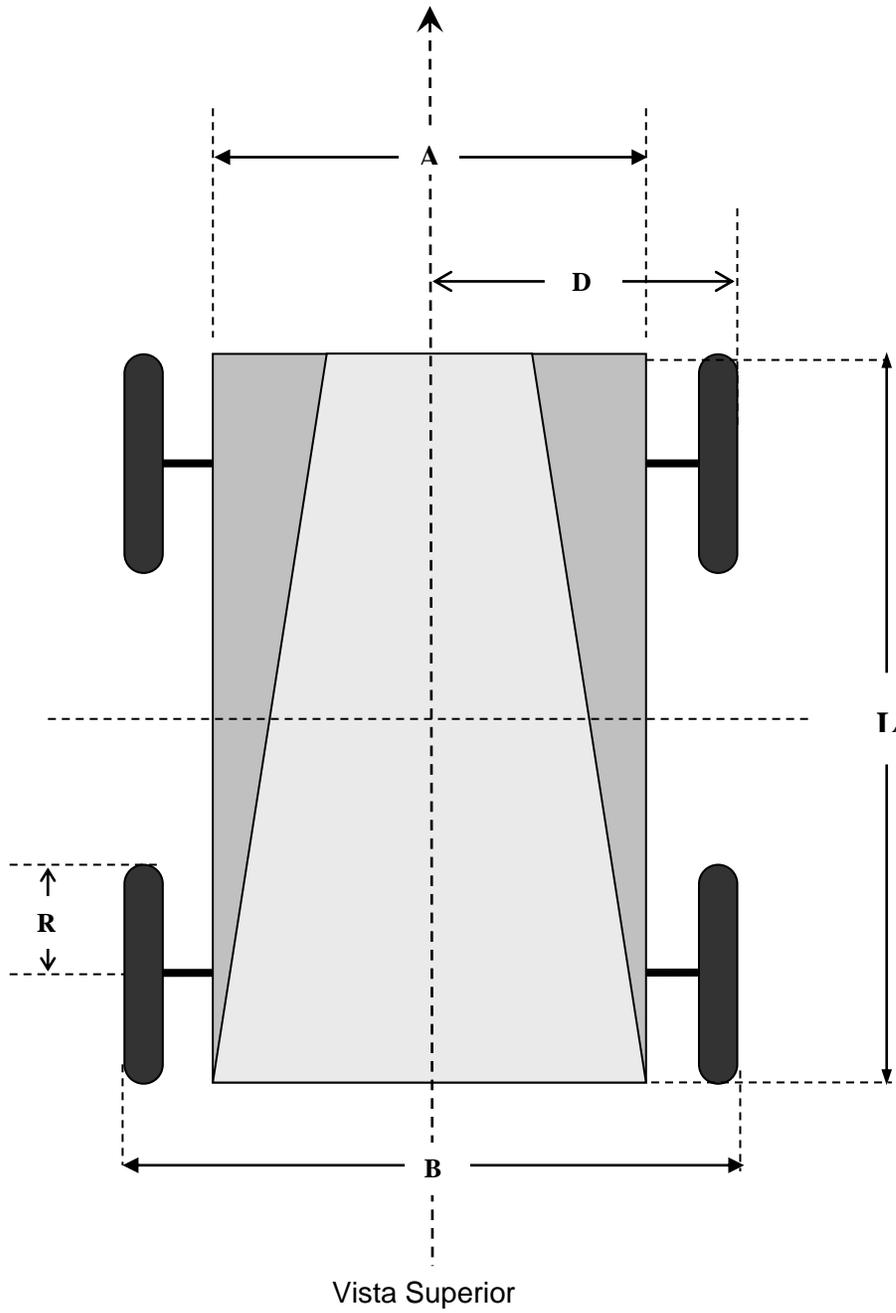
TIV1    clrwdt
        btfsz   INTCON,T0IF
        goto    TIV1
        decfsz  Tiempo,F
        goto    TIV0
        bcf     INTCON,T0IE
        bcf     INTCON,T0IF
        return

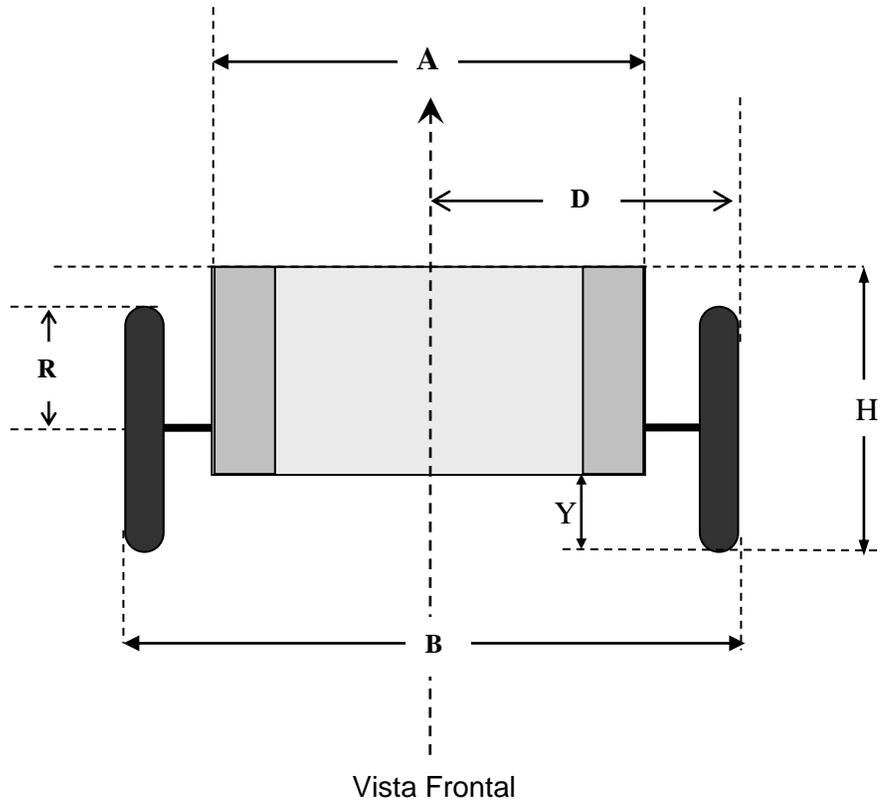
;-----
END

```

# ANEXO C

## Dimensiones Físicas del Móvil





Las dimensiones del móvil son:

- A: 20 cm. Ancho de la estructura de soporte.
- B: 28 cm. Distancia del eje entre llantas.
- L: 28.8 cm. Largo del móvil.
- D: 14 cm. Distancia del eje vertical a la llanta.
- R: 5 cm. Radio de la llanta.
- H: 11.5 cm. Altura de la parte superior del móvil con respecto al piso.
- Y: 3 cm. Altura de la parte inferior del móvil con respecto al piso.

## **ANEXO D**

### **Sensores de ultrasonido utilizados**

Los tipos de sensores que utiliza el robot son:

- MHB40
- MIT40R

El código embebido en el microcontrolador A Controla transmisión y recepción de ultrasonido; también las funciones para medir la distancia

## ANEXO E

### ESPECIFICACIONES TECNICAS

#### Alimentación:

- Sistema Análogo: +12V y -12V
- Sistema digital: 5V
- Motores paso a paso: 8V

#### Sistema de ultrasonido:

- Alcance máximo: 127 Cm
- Alcance mínimo: 2 Cm

#### Motores:

- Tipo: Paso a Paso Unipolares
- Resolución 7.5° por paso
- Consumo de corriente: 500mA por fase.

#### Microcontroladores:

- PIC 16F873

#### Comunicaciones:

- Serial RS-232
- Características comunicación serial: 9600, n, 8, 1.

**ANEXO F**  
**MANUAL DE USUARIO**

## **1. Introducción**

El Robot AMI es un vehículo guiado de forma autónoma; está conectado a un computador que controla sus acciones. En el equipo se encuentran alojados tres programas excluyentes, es decir el robot solo se puede manejar con uno a la vez, estos son TesUsound, TesMotores y Mind.

Se parte de un trabajo de simulación precedente y se desarrolla un programa para el control real de un robot.; la solución implica guiar un vehículo automáticamente empleando redes neuronales para el reconocimiento y un control difusos para la toma de decisiones. Para el usuario, el reconocimiento del ambiente y el controlador difuso permanecen ocultos.

El sistema ha sido desarrollado sobre una plataforma Windows, su interfaz gráfica es de fácil manejo. Este manual no provee información de cómo funciona el programa, es una guía útil para controlar y probar el robot AMI.

## **2. Generalidades**

En ésta sección se realizan recomendaciones para el usuario, descripción de la interfaz gráfica y los pasos a seguir para manipular adecuadamente el robot..

### **2.1 Recomendaciones**

- No exceda los límites de operación dados en las especificaciones técnicas. (Véase anexo D).
- El robot está diseñado para trabajar en ambientes cubiertos y para moverse en superficies planas. No está diseñado para desplazarse en terrenos con irregularidades como prados, gravas, etc.
- No debe transportar más peso que el especificado.

- Evitar golpes sobre los sensores de ultrasonido.
- No derramar líquidos sobre el y jamás lo use en superficies húmedas.
- 

El sistema de simulación ha sido desarrollado bajo el lenguaje de implementación Visual C++. El proceso de simulación puede dividirse en tres partes: La ubicación de obstáculos en el ambiente, la ubicación y orientación inicial del móvil y la ejecución del movimiento.

El tamaño del ambiente posee una escala proporcional directa al tamaño de la ventana, la cual depende de la resolución dada a la pantalla en píxeles. La resolución empleada por defecto es de 800 x 600 y el tamaño del área cliente es aproximadamente de 800 x 500 dando como resultado un ambiente simulado de 8 metros de largo por 5 metros de ancho.

La velocidad de ejecución del movimiento depende intrínsecamente de la velocidad de cómputo de la máquina sobre la cual se ejecuta el sistema de simulación, debido a los requerimientos de cómputo para la red neuronal y el dibujado de la interfaz.

## **2.2 Descripción de la Interfaz del Sistema de Control**

La ventana principal contiene los elementos requeridos para acceder a las diferentes características soportadas por el sistema de simulación, mediante la barra de herramientas o desde el menú. Como se muestra en la figura 1, se distinguen cuatro partes en la interfaz: El menú de opciones, La barra de herramientas, el ambiente constituido por el área cliente de la ventana y la barra de estado para despliegue de mensajes. A continuación se detalla cada una de las partes que componen la ventana principal.

## A. EL MENU DE OPCIONES

Contiene las opciones generales del sistema. Se compone de seis ítems los cuales se describen a continuación:

**Archivo :** El menú **Archivo** (Véase figura 2.a ) contiene las opciones para crear un nuevo ambiente y para cerrar la aplicación.

**Ver :** Permite ver u ocultar la barra de herramientas o la barra de estado ( ver figura 2.b ). La opción *Trayectoria* habilita o inhibe el despliegue de la trayectoria seguida por el móvil. La condición por defecto de esta opción es inhabilitada. La opción *Rejilla*, permite ver o esconder la rejilla que divide el ambiente en secciones cuadradas de 50 cms de lado teniendo en cuenta la escala planteada. La opción *Histograma* presenta la ventana "Histograma Polar" que despliega el histograma normalizado obtenido para cada ciclo de muestreo en forma polar.

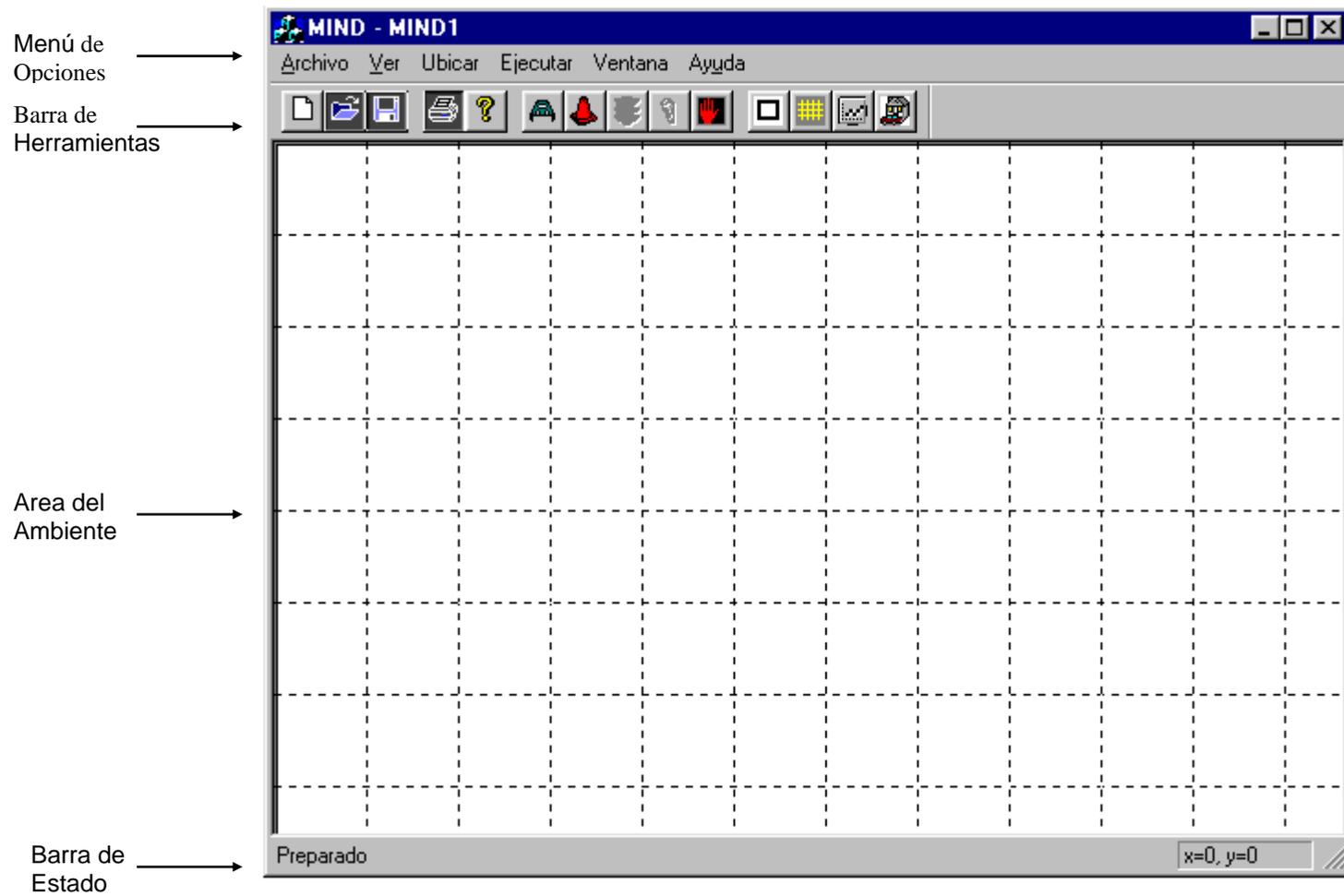


Figura 1. Ventana principal de la interfaz de usuario del sistema de simulación.

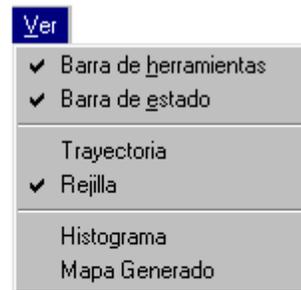
Pueden observarse los lóbulos que se crean como resultado del tratamiento de la información obtenida a partir de la red neuronal y que son empleados por el controlador difuso como medida del grado de ocupación. Una vez desplegada la ventana, se inhibe la opción *Histograma* activándose solamente si ésta ventana es cerrada.

La opción *Mapa Generado* presenta la ventana “ Mapa Reconocido “ que despliega un mapa aproximado obtenido durante el recorrido que el móvil ha realizado en el ambiente. Este mapa es construido a escala 1: 5 y los colores azul, verde y rojo indican el grado de certeza en la ubicación de obstáculos. Esta opción está activa solamente si el móvil se encuentra detenido.

Al desplegar cualquiera de las dos ventanas, el foco cambiará a la última ventana activa. Si cualquiera de éstas dos ventanas ha sido desplegada y el foco cambia a la ventana principal, éstas serán ocultadas y su visibilidad se recuperará a través del menú Ventana.



( a )



( b )

Figura 2. Menús Archivo ( a ) y Ver ( b ).

**Ubicar** : las opciones *Móvil* y *Meta* del menú **Ubicar** ( Ver figura 3.a ) despliegan dos cuadros de dialogo por medio de los cuales el usuario puede fijar la posición y orientación del móvil ( figura 4.a ) y la posición final de llegada ( figura 4.b ) respectivamente. Al pulsar el botón OK se hará efectiva la acción.

Existe una opción adicional nombrada *Activar WFM* encargada de activar o inhibir el algoritmo de seguimiento de pared. El fin de ésta opción es proveer la capacidad de comparar el comportamiento del móvil en condiciones de trampa u otras configuraciones de obstáculos con y sin la heurística de seguimiento de pared.

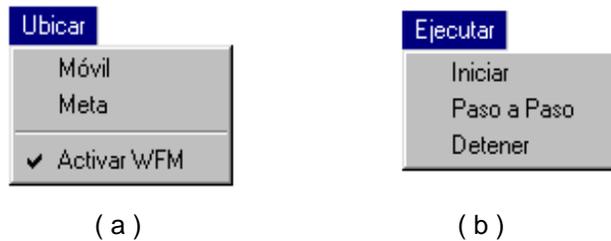
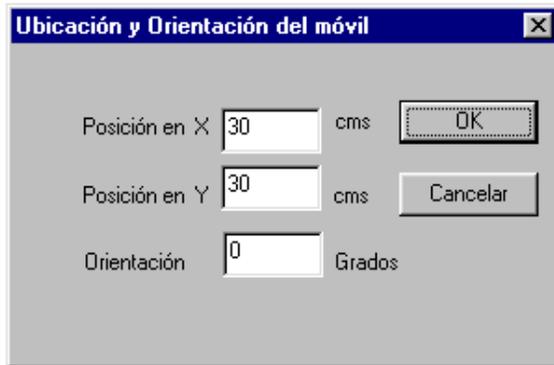


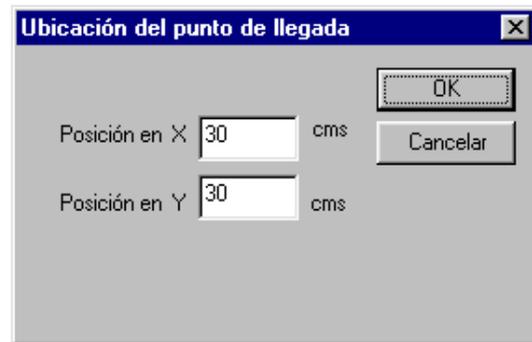
Figura 3. Menús Ubicar ( a ) y Ejecutar ( b ).

**Ejecutar** : El menú **Ejecutar** ( figura 3.b ) permite una ejecución continua o paso a paso del movimiento del móvil y detener su movimiento a voluntad del usuario. Las opciones *Iniciar* y *Paso a paso* están habilitadas solamente si el móvil y el punto de meta han sido previamente ubicados. Cuando se emplean las opciones *Iniciar* o *Paso a paso*, las opciones *Móvil* y *Meta* del menú **Ubicar** ( figura 3.a ) son deshabilitadas, es decir, no se puede ubicar el móvil o la meta mientras se realiza el movimiento. La opción *Detener* habilita nuevamente éstas opciones del menú **Ubicar**.

**Ventana :** El menú **Ventana** ( figura 5.a ) permite cambiar el foco entre las diferentes ventanas, ya sea la ventana principal, la ventana Histograma Polar ó la ventana Mapa Reconocido. Adicionalmente permite crear un nuevo Ambiente.



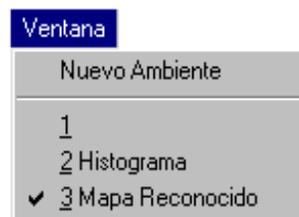
( a )



( b )

Figura 4. Cuadros de dialogo para ubicar el Móvil ( a ) y la Meta ( b ).

**Ayuda :** El menú **Ayuda** ( figura 5.b ) despliega el cuadro de dialogo Acerca de MIND...



( a )



( b )

Figura 5. Menús Ventana ( a ) y Ayuda ( b ).

## B. BARRA DE HERRAMIENTAS:

Permite acceder a las principales opciones del menú directamente. La figura 6 describe sus funciones.

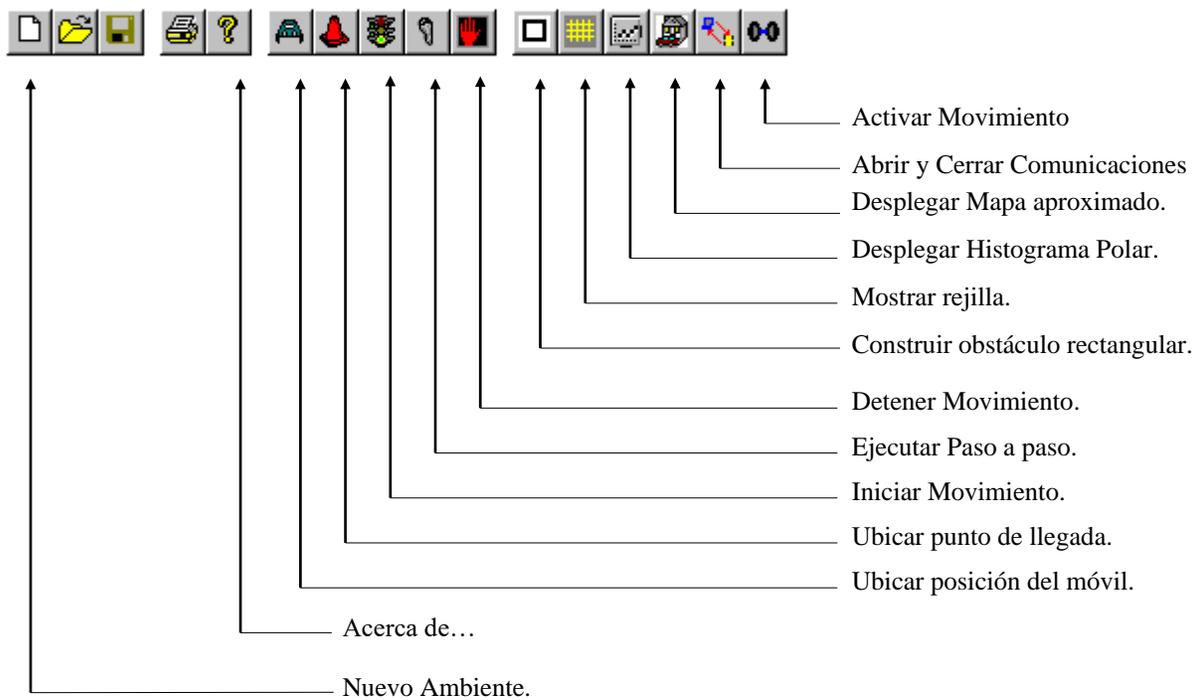


Figura 6. Funciones de la barra de herramientas.

## C. AREA DE MENSAJES Y COORDENADAS:

Constituida por la barra de estado, presenta el mensaje que da información sobre la función que realiza el botón de comando sobre el cual se encuentre ubicado el Mouse. Además despliega las coordenadas de la posición del puntero sobre el área del ambiente.

### 3. Descripción de las funciones de la barra de Herramientas.



Ubicar el móvil en un punto de la pantalla, esta posición debe corresponder a la posición del móvil en el ambiente real.



Ubicar el punto de llegada de el móvil en la pantalla, este debe coincidir a escala con el punto real en el ambiente en el que se mueve el robot.

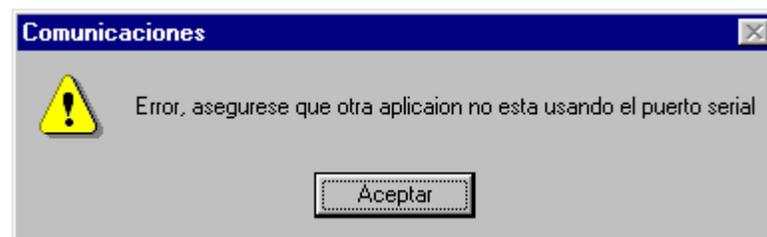


Abrir comunicaciones con el robot, si se logra una apertura de comunicación exitosa se muestra el siguiente cuadro de dialogo:



Esto significa que el robot se ha enlazado correctamente con el servidor y está listo para recibir ordenes.

Si por el contrario no se puede abrir la comunicación se muestra el siguiente cuadro de dialogo:



Si las comunicaciones no se establecen de manera adecuada entonces el móvil queda incapacitado para moverse.

Si las comunicaciones se han iniciado correctamente se puede cerrar el enlace presionando el mismo icono, si se cierra adecuadamente la sesión se muestra el siguiente cuadro de dialogo:



Activa o desactiva el movimiento del robot real, pero la representación virtual que se muestra en pantalla si se puede mover.

Cuando se activa se muestra la siguiente pantalla:



Cuando se desactiva el movimiento se muestra el siguiente dialogo:



Si la comunicación fue abierta y los motores se activaron se puede proceder a iniciar la navegación autónoma ya sea paso a paso pulsando el icono:



O de manera continua pulsando el icono:

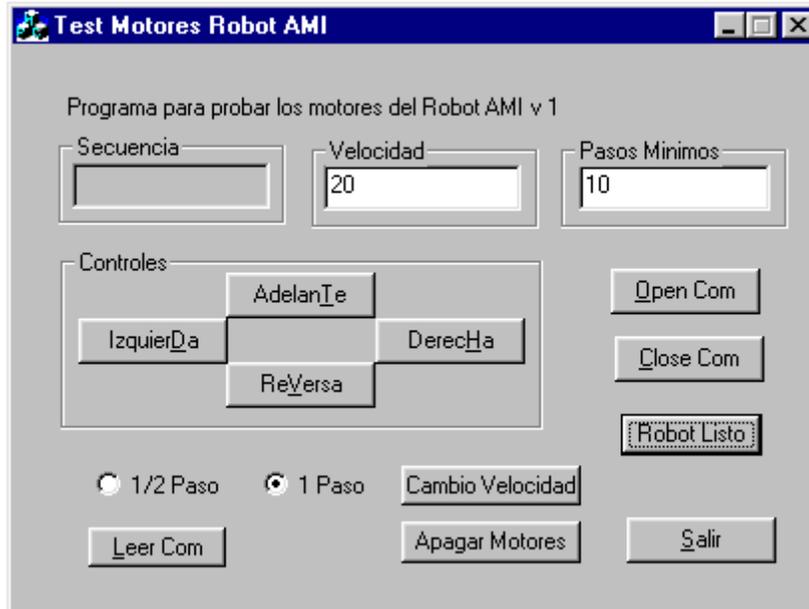


#### **4. Programas adicionales para control manual y pruebas del robot**

Los siguientes programas se construyeron como herramientas de ayuda al usuario del robot con el fin de hacer un test manual de cada una de las partes importantes de la plataforma móvil como son los sensores de ultrasonido y los motores. Sus interfaz de usuario son muy simples e intuitivamente se puede aprender a manejarlas. Igual que en programa principal de control, MIND, estos dos programas deben conectarse al robot por un puerto serial y solamente uno a la vez puede estar en comunicación con el robot por lo tanto si se pretende usarlos de forma simultanea los programas no lo permiten.

##### 4.1 Programa Test Robot

Este programa permite controlar de forma manual los motores del robot, de este modo se puede mover el robot al frente, derecha, izquierda y atrás, cambiar la velocidad y la configuración de los pasos ya sea medio paso o paso completo, la interfaz gráfica es la siguiente:



#### 4.2 Programa Test Usound

Este programa permite comprobar el funcionamiento de cada uno de los sensores de ultrasonido, así como configurar los tiempos para zona muerta y demás parámetros relevantes para un óptimo funcionamiento de los sensores.

