Office Tag

Author: Lon Glazner
Chico, California
email: lon@solutions-cubed.com

INTRODUCTION:

This design fragment is being used as an entry into the Microchip De$igning for Dollar$ design contest for the month of September. It is based upon a PIC12CXXX 8 bit microcontroller, which is used to control an IR receiver/transmitter circuit. The end result is a short range data transmitter receiver that could be used for remote controls, printer interfaces, and various other data transceiver applications. This design makes you ask, “Shouldn’t every remote control have a PIC12C508 in it”?

Office Tag is similar to laser tag, but these low cost infra-red devices are capable of being used in remote locations to settle a myriad of nasty conflicts. The Office Tag concept grew out of a global need for the peaceful resolution of conflict. It was decided that solving conflicts with guns was reasonable, but the use of bullets was wholly unacceptable. During subsequent testing of the device, it was also determined that office productivity could be enhanced by 247.34567% (rounded up) with Office Tag. The test facility for both the psychological and technical aspects for the Office Tag system was of course SCIETG (pronounced “sight-gee”; stands for Solutions Cubed International Electronics Testing Grounds). In fact, Solutions Cubed has been using Office Tag as a meter for employee work performance. The idea is that if you can win at Office Tag, you’ve got to be a great guy (meant to be a gender-non-specific term). One detrimental aspect of Office Tag has been the increased cost of infra-red components at our purchasing department (located in Bora-Bora). The introduction of Office Tag appears to have been the catalyst for an inter-office arms race. The end result may well be desk mounted hard points bristling with IR emitters and well protected IR receivers, at least in our engineering area.

APPLICATION OPERATION:

Hardware Methodology:

The Office Tag system makes use of an 8 Pin PIC12C508 to do wondrous things with very few parts. The IR emitter consists of a 32.7KHz oscillator, an N-channel FET pass element, an IR LED, and a current limiting resistor. The PIC12C508 sends data that is modulated onto the 32.7KHz carrier frequency. This method of sending data with IR systems, when used in conjunction with a receiver module, can send data out to a range of about 20 feet. The use of multiple IR LEDs or higher current handling LEDs could extend this range. Our office tag system was designed entirely from components available in our stock room. For this reason, the logic components used are displayed on the schematic by the logic function that is implemented. We actually used an MM74C20 (dual-5 input CMOS NAND gate) to generate the logic functions. The 32.7KHz oscillator is based on standard crystals and other low cost components. Be sure to use a CMOS inverter for this oscillator.

The IR receiver is a Lite-On product (LTM-8834-2) that demodulates 32.7Khz carrier frequencies. Lower cost receivers can be designed with discrete components. One of the neat features of this receiver is that it comes as a complete decoder module so digital data can be accurately decoded in a simple manner.

The heart of the Office Tag system is the PIC12C508. The PIC12C508 is operated with internal pull-ups enabled, the internal RC oscillator enabled, and the external MCLR being used.

The PIC12C508 monitors the trigger input (S1 with internal pull-ups enabled), whenever depressed the PIC12C508 fires a bullet (10 consecutive h’F0’ at 2400 baud, 8N1, LSB). The bullet is preceded by an audible tone generated by the PIC12C508 through the piezo electric buzzer (denoted as U5). Like all good guns, the Office Tag system is only a six-shooter. To reload, the user (read gunslinger) must press the reload but-

Microchip Technology Incorporated, has been granted a nonexclusive, worldwide license to reproduce, publish and distribute all submitted materials, in either original or edited form. The author has affirmed that this work is an original, unpublished work and that he/she owns all rights to such work. All property rights, such as patents, copyrights and trademarks remain with author.

ton (S2) which effectively resets the processor. A 2 second delay is instituted on a reset since it takes time to reload. When not firing or reloading, the PIC12C508 is polling the receiver module to see if data is present. For this application, we had the data appear to be a square wave 10 period long (generated by 10 consecutive h'F0' at 2400 baud). If 5 periods of the square wave are monitored, than the Office Tag system is killed. The reload LED will flash and the audible buzzer will beep on a kill. Hitting reload will reset your gun.

Bill of Materials

Designation	Part	Price/100	Specifications
R1	1K	0.03	5% 1/8 watt
R2	1K ohm	0.03	5% 1/8 watt
R3	20M ohm	0.03	5% 1/8 watt
R4	200K ohm	0.03	5% 1/8 watt
R5	51 ohm	0.03	5% 1/8 watt
R6	56 ohm	0.53	2 watt
C1	22pF	0.18	
C2	22pF	0.18	
C3	0.01uF	0.11	
C4	10uF	0.09	
D1	LED	0.11	RED
D2	IR LED	1.00	150mA max
Q1	2VN4206AV	0.65	N channel FET
U1	LM7805L	0.65	
U2	12C508-4/P	1.13	neato
U3	MM74C20	0.45	CMOS NAND gates
U4	LTM-8834-2	3.35	IR receiver module
U5	piezo alarm	1.00	5Vdc buzzer
	Total:	9.58	

Software Methodology:

The software methodology was explained somewhat at the end of the hardware methodology. Basically this software reacts to two outside stimuli. The first is a trigger (S1) press. The second stimulus is a valid signal received from the IR module (LTM-8834-2). If a valid signal is received, the Office Tag system dies (lights flash, buzzers buzz, etc.). If the trigger is pressed, data is sent out of the IR LED (this is a bullet). The PIC12C508 will only let you fire 6 bullets. If you die or run out of bullets, you must press the reload button (S2) in order for the Office Tag system to reset.

RAM Used:	9 bytes
Program Bytes (as presented):	202
Program Cycles (min, no trigger press, no data in):	24
Program Cycles (max, trigger pressed):	~78,000

Office Tag Rules of Engagement:

None.

Game Hints:

At close range (3-4 feet) you can kill your opponent without actually pointing at the receiver module. You may even score a kill from behind your opponent. At medium range (5-15 feet), a kill can be achieved at angles up to 90 degrees from the front of you opponents Office Tag system. When engaging at long range (16+ feet), actual aiming skill will be necessary, pointing the IR LED directly at your opponents receiver will be required. Kills at extreme ranges of 20 to 30 feet may be possible, but are not guaranteed. Your environment can confuse your opponents Office Tag system. High power fluorescent lights, PC monitors, and other electronic devices can cause noise that can be picked up by your Office Tag system. This noise may prevent

© 1998 Microchip Technology Inc.

your system from registering a kill against you, or it may actually cause your system to die. Beware (or be aware) of this stuff. Office Tag dueling(5 paces, turn and shoot), can hone your targeting skills. Last hint, release the trigger button as soon as you fire. Holding down the button locks your Office Tag system into its reception mode. This will get you killed quick.

APPENDIX A: SOURCE CODE

```
MPASM 01.40 Released           TAG1.ASM   9-18-1997  10:53:07        PAGE  1


LOC  OBJECT  CODE   LINE SOURCE TEXT
  VALUE

                     ;***************************************************************************
                     ;***************************************************************************
                     ;****                          SOLUTIONS CUBED                          ****
                     ;****               Frank Rossini, Lon Glazner, David Brobst            ****
                     ;***************************************************************************
                     ;***************************************************************************
                     ;
                     ;
                     ;***************************************************************************
                     ;****                    Office Tag Assembly Code Listing               ****
                     ;***************************************************************************
                     ;
                     ;     This assembly code was designed for use with the Office Tag system.
                     ;     The design was generated specifically for the Microchip designing for
                     ;     dollars contest.  The office tag design is a fully fleshed out
                     ;     IR interface system.  The design's cost could be reduced in the
                     ;     receiver section by using discrete components.  This design
                     ;     could be used for IR-printer interfaces, remote control, short range
                     ;     data tranceivers, as well as cool stuff like the office tag stress
                     ;     release system.
                     ;
                     ;***************************************************************************
                     ;
                     ;
                     ;***************************************************************************
                     ;***************************************************************************
                     ;****  Define registers, constants, processor, and assembler directives ****
                     ;***************************************************************************
                     ;***************************************************************************
                     ;
                     ;Processor
                     ;
                             LIST    P=12C508                ;Processor used
                     ;
                     ;       fuses:
                     ;               WDT     - on
                     ;               OSC     - internal RC
                     ;               MCLR    - external MCLR
                     ;               CP      - code protect on
                     ;
                     ;Processor defined registers and bits
                     ;
                             INCLUDE "C:\PIC\HEADERS\P12C508.INC";Microchip include file
                            LIST
                     ; P12C508.INC  Standard Header File, Version 1.01  Microchip Technology, Inc.
                            LIST
                     ;
                     ;Program defined registers
                     ;
00000006             GPIO       EQU        H'06'              ;Output port register
00000007             TEMP0      EQU        H'07'              ;Temporary storage register
00000008             TEMP       EQU        H'08'              ;Temporary storage register
00000009             FLAG0      EQU        H'09'              ;Storage register for flags
0000000A             DATA_REG   EQU        H'0A'              ;Storage register for serial data
0000000B             HALF_BIT   EQU        H'0B'              ;Stores half bit delay
0000000C             FULL_BIT   EQU        H'0C'              ;Stores full bit delay
0000000D             COUNT0     EQU        H'0D'              ;Used to count out delay in comm.
0000000E             COUNT1     EQU        H'0E'              ;Used to count in FIRE_GUN mode
```

© 1998 Microchip Technology Inc.

```
0000000F     00056  BULLETS    EQU         H'0F'              ;Stores number of bullets
             00057  ;
             00058  ;Program defined bits
             00059  ;
             00060  ;GPIO port bits
00000000     00061  TRIGGER    EQU         H'00'              ;GPIO, trigger input
00000001     00062  INPUT      EQU         H'01'              ;GPIO, receiver input pin
00000002     00063  OUTPUT     EQU         H'02'              ;GPIO, output for data sent
00000003     00064  MCLR       EQU         H'03'              ;GPIO, not used as an i/o
00000004     00065  BUZZER     EQU         H'04'              ;GPIO, audible buzzer output
00000005     00066  KILL       EQU         H'05'              ;GPIO, LED flashes when killed
             00067  ;
             00068  ;FLAG0 register bits
00000000     00069  DATA_GOOD  EQU         H'00'              ;Set after good serial data is
received
00000001     00070  DEAD       EQU         H'01'              ;Set when killed
00000002     00071  JUST_FIRED EQU         H'02'              ;Set when trigger was pressed
             00072  ;***************************************************************************
             00073  ;
             00074  ;
             00075  ;***************************************************************************
             00076  ;***************************************************************************
             00077  ;****                           Reset Vector                               ****
             00078  ;***************************************************************************
             00079  ;***************************************************************************
0000         00080             ORG         H'000'
0000 0025    00081             MOVWF       OSCCAL             ;Move internal trim value to osccal
0001 0AA7    00082             GOTO        MAIN
             00083  ;***************************************************************************
             00084  ;
             00085  ;
             00086  ;***************************************************************************
             00087  ;BYTE_OUT:  Sends a byte of data at 2400 baud, 8N1, LSB first.
             00088  ;       The data in the DATA_BYTE register when this routine is
             00089  ;       called is the data that is sent.
             00090  ;
             00091  ;       Called From:            FIRE_GUN
             00092  ;       Modified Registers:     STATUS, TEMP0, TEMP1, DATA_BYTE,
             00093  ;                               GPIO, FLAG, FULL_BIT
             00094  ;       Subroutines Called:     DELAY, OUT_HI(goto), OUT_LO(goto)
             00095  ;       Enabled Interrupts:     NONE
             00096  ;
0002         00097  BYTE_OUT
0002 0004    00098             CLRWDT
0003 0C08    00099             MOVLW       H'08'
0004 0028    00100             MOVWF       TEMP1
0005 020C    00101             MOVF        FULL_BIT,W
0006 0027    00102             MOVW        TEMP0
0007 02A7    00103             INCF        TEMP0
0008 02A7    00104             INCF        TEMP0
             00105  ;
0009 0546    00106             BSF         GPIO,OUTPUT
000A         00107  start_out
000A 02E7    00108             DECFSZ      TEMP0
000B 0A0A    00109             goto        start_out
000C         00110  next_bit_out
000C 020C    00111             MOVF        FULL_BIT,W         ;Baud rate is = 2403baud
000D 0027    00112             MOVWF       TEMP0              ;
000E 032A    00113             RRF         DATA_REG
000F 0603    00114             BTFSC       STATUS,C
0010 0446    00115             BCF         GPIO,OUTPUT        ;Clear GP0
0011 0703    00116             BTFSS       STATUS,C
0012 0546    00117             BSF         GPIO,OUTPUT        ;Set GP0
0013 0000    00118             NOP
0014         00119  waiting
0014 02E7    00120             DECFSZ      TEMP0
```

© 1998 Microchip Technology Inc.

```
0015 0A14    00121             goto       waiting
0016 02E8    00122             DECFSZ     TEMP1
0017 0A0C    00123             goto       next_bit_out
             00124  ;
0018 020C    00125             MOV        FULL_BIT,W
0019 0027    00126             MOVWF      TEMP0
001A 0C04    00127             MOVLW      H'04'
001B 01E7    00128             ADDWF      TEMP0,F            ;Baud rate = 2398
001C 0446    00129             BCF        GPIO,OUTPUT
001D         00130  stop_bit_out
001D 02E7    00131             DECFSZ     TEMP0
001E 0A1D    00132             goto       stop_bit_out
001F 0800    00133             RETLW      H'00'
             00134  ;***************************************************************************
             00135  ;SIGNAL_IN:  Signal in  receives data from any firing unit.  This routine
             00136  ;       looks consecutiove low pulses, if found then the DEAD flag is set.
             00137  ;
             00138  ;
             00139  ;          Called From:            R_U_TARGETED, FIRE_GUN
             00140  ;          Modified Registers:      STATUS, TEMP0, TEMP1, GPIO, DATA_REG
             00141  ;          Subroutines Called:     DELAY
             00142  ;          Enabled Interrupts:     NONE
             00143  ;
0020         00144  SIGNAL_IN
0020 0004    00145             CLRWDT
0021 0626    00146             BTFSC      GPIO,INPUT         ;Test for a start bit
0022 0800    00147             RETLW      H'00'
0023 0C05    00148             MOVLW      H'05'
0024 002D    00149             MOVWF      COUNT0
0025         00150  test_again
0025 0CC8    00151             MOVLW      H'C8'
0026 0027    00152             MOVWF      TEMP0
0027         00153  low_on
0027 0626    00154             BTFSC      GPIO,INPUT         ;Test for 1000us low signal
0028 0800    00155             RETLW      H'00'
0029 02E7    00156             DECFSZ     TEMP0
002A 0A27    00157             goto       low_on
002B 02ED    00158             DECFSZ     COUNT0             ;Five pulses = dead
002C 0A2F    00159             goto       next_pulse
002D 0529    00160             BSF        FLAG0,DEAD
002E 0800    00161             RETLW      H'00'
             00202  ;***************************************************************************
             00203  ;
             00204  ;
             00205  ;***************************************************************************
             00206  ;FIRE_GUN:  Tests trigger input to see if it has been pressed.  If the button
             00207  ;          is pressed then the routine sends data out of the IR system.  The
             00208  ;          routine also tests the input for data received.  With this routine
             00209  ;          you may be killed while firing the device
             00210  ;
             00211  ;          Called From:            MAIN
             00212  ;          Modified Registers:     DATA_REG, STATUS, FLAG0
             00213  ;          Subroutines Called:     DELAY_XMS, BYTE_IN, BYTE_OUT
             00214  ;          Enabled Interrupts:     NONE
             00215  ;
0045         00216  FIRE_GUN
0045 0606    00217             BTFSC      GPIO,TRIGGER
0046 0800    00218             RETLW      H'00'
0047 0C05    00219             MOVLW      H'05'              ;Debounce trigger for 5ms
0048 093C    00220             CALL       DELAY_XMS
0049 0606    00221             BTFSC      GPIO,TRIGGER
004A 0800    00222             RETLW      H'00'
             00223  ;
004B 0C0A    00224             MOVLW      H'0A'
004C 002E    00225             MOVWF      COUNT1
004D         00226  low_buzz0
```

© 1998 Microchip Technology Inc.

```
004D 0586    00227              BSF        GPIO,BUZZER        ;Turn on buzzer
004E 0C05    00228              MOVLW      H'05'
004F 093C    00229              CALL       DELAY_XMS
0050 0486    00230              BCF        GPIO,BUZZER        ;Turn off buzzer
0051 0C05    00231              MOVLW      H'05'
0052 093C    00232              CALL       DELAY_XMS
0053 02EE    00233              DECFSZ     COUNT1
0054 0A4D    00234              goto       low_buzz0
             00235   ;
0055 0C0A    00236              MOVLW      H'0A'
0056 002E    00237              MOVWF      COUNT1
0057         00238   low_buzz1
0057 0586    00239              BSF        GPIO,BUZZER        ;Turn on buzzer
0058 0C04    00240              MOVLW      H'04'
0059 093C    00241              CALL       DELAY_XMS
005A 0486    00242              BCF        GPIO,BUZZER        ;Turn off buzzer
005B 0C04    00243              MOVLW      H'04'
005C 093C    00244              CALL       DELAY_XMS
005D 02EE    00245              DECFSZ     COUNT1
005E 0A57    00246              goto       low_buzz1
             00247   ;
005F 0C0A    00248              MOVLW      H'0A'
0060 002E    00249              MOVWF      COUNT1
0061         00250   low_buzz2
0061 0586    00251              BSF        GPIO,BUZZER        ;Turn on buzzer
0062 0C03    00252              MOVLW      H'03'
0063 093C    00253              CALL       DELAY_XMS
0064 0486    00254              BCF        GPIO,BUZZER        ;Turn off buzzer
0065 0C03    00255              MOVLW      H'03'
0066 093C    00256              CALL       DELAY_XMS
0067 02EE    00257              DECFSZ     COUNT1
0068 0A61    00258              goto       low_buzz2
             00259   ;
0069 0C0A    00260              MOVLW      H'0A'
006A 002E    00261              MOVWF      COUNT1
006B         00262   low_buzz3
006B 0586    00263              BSF        GPIO,BUZZER        ;Turn on buzzer
006C 0C02    00264              MOVLW      H'02'
006D 093C    00265              CALL       DELAY_XMS
006E 0486    00266              BCF        GPIO,BUZZER        ;Turn off buzzer
006F 0C02    00267              MOVLW      H'02'
0070 093C    00268              CALL       DELAY_XMS
0071 02EE    00269              DECFSZ     COUNT1
0072 0A6B    00270              goto       low_buzz3
             00271   ;
0073 0C0A    00272              MOVLW      H'0A'
0074 002E    00273              MOVWF      COUNT1
0075         00274   low_buzz4
0075 0586    00275              BSF        GPIO,BUZZER        ;Turn on buzzer
0076 0C01    00276              MOVLW      H'01'
0077 093C    00277              CALL       DELAY_XMS
0078 0486    00278              BCF        GPIO,BUZZER        ;Turn off buzzer
0079 0C01    00279              MOVLW      H'01'
007A 093C    00280              CALL       DELAY_XMS
007B 02EE    00281              DECFSZ     COUNT1
007C 0A75    00282              goto       low_buzz4
             00283   ;
007D 0C0A    00284              MOVLW      H'0A'
007E 002E    00285              MOVWF      COUNT1
007F         00286   more_bullets
007F 0CF0    00287              MOVLW      H'F0'              ;Send hexF0
0080 002A    00288              MOVWF      DATA_REG
0081 0902    00289              CALL       BYTE_OUT
0082 02EE    00290              DECFSZ     COUNT1
0083 0A7F    00291              goto       more_bullets
0084 0C19    00292              MOVLW      H'19'
```

```
0085 093C     00293             CALL       DELAY_XMS          ;Wait 25ms
0086 0549    00294              BSF        FLAG0,JUST_FIRED   ;Ensures trigger release in R_U
                                                               routine
0087 02EF    00295              DECFSZ     BULLETS
0088 0800    00296              RETLW      H'00'
0089         00297   no_more_bullets
0089 0004    00298              CLRWDT
008A 0A89    00299              goto       no_more_bullets
             00300   ;***************************************************************************
             00301   ;
             00302   ;
             00303   ;***************************************************************************
             00304   ;R_U_TARGETED:  Tests to see if you are receiving data. If the trigger
             00305   ;          is pressed you cannot exit this routine.
             00306   ;
             00307   ;          Called From:            MAIN
             00308   ;          Modified Registers:     DATA_REG, STATUS, FLAG0
             00309   ;          Subroutines Called:     BYTE_IN, DELAY_XMS
             00310   ;          Enabled Interrupts:     NONE
             00311   ;
008B         00312   R_U_TARGETED
008B 0920    00313              CALL       SIGNAL_IN          ;Read in data
008C 0629    00314              BTFSC      FLAG0,DEAD
008D 0800    00315              RETLW      H'00'
008E         00316   test_trigger
008E 0749    00317              BTFSS      FLAG0,JUST_FIRED   ;Set if entered after firing
008F 0800    00318              RETLW      H'00'
0090 0706    00319              BTFSS      GPIO,TRIGGER       ;Test for trigger release
0091 0A8B    00320              goto       R_U_TARGETED       ;not released
0092 0C05    00321              MOVLW      H'05'              ;Debounce trigger release
0093 093C    00322              CALL       DELAY_XMS          ;5ms delay
0094 0706    00323              BTFSS      GPIO,TRIGGER       ;Test for trigger release
0095 0A8B    00324              goto       R_U_TARGETED
0096 0449    00325              BCF        FLAG0,JUST_FIRED   ;Clear just fired flag
0097 0800    00326              RETLW      H'00'
             00327   ;***************************************************************************
             00328   ;
             00329   ;
             00330   ;***************************************************************************
             00331   ;DEAD_YET:  Tests to see if you have been killed
             00332   ;
             00333   ;          Called From:            MAIN
             00334   ;          Modified Registers:     STATUS, FLAG0, GPIO
             00335   ;          Subroutines Called:     DELAY_XMS
             00336   ;          Enabled Interrupts:     NONE
             00337   ;
0098         00338   DEAD_YET
0098 0729    00339              BTFSS      FLAG0,DEAD
0099 0800    00340              RETLW      H'00'
009A         00341   flash_led
009A 04A6    00342              BCF        GPIO,KILL          ;Toggle LED every 500ms
009B 0586    00343              BSF        GPIO,BUZZER
009C 0CFA    00344              MOVLW      H'FA'
009D 093C    00345              CALL       DELAY_XMS
009E 0CFA    00346              MOVLW      H'FA'
009F 093C    00347              CALL       DELAY_XMS
00A0 05A6    00348              BSF        GPIO,KILL
00A1 0486    00349              BCF        GPIO,BUZZER
00A2 0CFA    00350              MOVLW      H'FA'
00A3 093C    00351              CALL       DELAY_XMS
00A4 0CFA    00352              MOVLW      H'FA'
00A5 093C    00353              CALL       DELAY_XMS
00A6 0A9A    00354              goto       flash_led
             00355   ;***************************************************************************
             00356   ;****                           Main Program                            ****
             00357   ;***************************************************************************
```

© 1998 Microchip Technology Inc.

```
00A7         00358  MAIN
             00359  ;
00A7         00360  OPTION_SETUP
00A7 0C8F    00361             MOVLW      H'8F'              ;1000 1111
00A8 0002    00362             OPTION                        ;Pull-up enabled, WDT 1:128
00A9         00363  CLEAR_REGISTERS
00A9 0067    00364             CLRF       TEMP0              ;Clear first RAM location for use
00AA 0C18    00365             MOVLW      H'18'              ;Number of registers to clear
00AB 0027    00366             MOVWF      TEMP0
00AC 0C08    00367             MOVLW      H'08'              ;Start of RAM clearing
00AD 0024    00368             MOVWF      FSR
00AE         00369  clear_loop
00AE 0060    00370             CLRF       INDF               ;Clear register pointed to
00AF 02A4    00371             INCF       FSR,F              ;Go to next RAM location to clear
00B0 02E7    00372             DECFSZ     TEMP0,F            ;Check to see if all clearing done
00B1 0AAE    00373             goto    clear_loop
00B2         00374  PORT_SETUP
00B2 0C2B    00375             MOVLW      H'2B'              ;0010 1011
00B3 0026    00376             MOVWF      GPIO               ;Set output low
00B4 0000    00377             NOP
00B5 0C0B    00378             MOVLW      H'0B'              ;0000 1011
00B6 0006    00379             TRIS       GPIO               ;Set GP2,4,5 direction as an output
00B7         00380  REGISTER_SETUP
00B7 0C87    00381             MOVLW      H'87'
00B8 002C    00382             MOVWF      FULL_BIT           ;Initialize serial communication
00B9 0C29    00383             MOVLW      H'29'              ;for 2400 baud
00BA 002B    00384             MOVWF      HALF_BIT
00BB 0C06    00385             MOVLW      H'06'
00BC 002F    00386             MOVWF      BULLETS            ;Load four bullets
             00387  ;
00BD 04A6    00388             BCF        GPIO,KILL          ;On a reload or power up
00BE 0C08    00389             MOVLW      H'08'              ;turn on LED and wait 2s
00BF 002D    00390             MOVWF      COUNT0             ;before continuing
00C0         00391  reload_delay
00C0 0CFA    00392             MOVLW      H'FA'
00C1 093C    00393             CALL       DELAY_XMS
00C2 02ED    00394             DECFSZ     COUNT0
00C3 0AC0    00395             goto       reload_delay
00C4 05A6    00396             BSF        GPIO,KILL
             00397  ;
             00398  ;***************************************************************************
00C5         00399  MAIN_LOOP
00C5 0004    00400             CLRWDT
00C6 0945    00401             CALL       FIRE_GUN
00C7 098B    00402             CALL       R_U_TARGETED
00C8 0998    00403             CALL       DEAD_YET
00C9 0AC5    00404             goto       MAIN_LOOP          ;Do it all again
             00405  ;***************************************************************************
             00406  ;
             00407  ;End of code indicator
             00408  ;
             00409             END
MEMORY USAGE MAP ('X' = Used,  '-' = Unused)

0000 : XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX
0040 : XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX
0080 : XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX
00C0 : XXXXXXXXXX------ ---------------- ---------------- ----------------

All other memory blocks unused.

Program Memory Words Used:   202
Program Memory Words Free:   309
```

© 1998 Microchip Technology Inc.

NOTES:

© 1998 Microchip Technology Inc.