Simulation of Microcontroller Behavior for Hardware Backdoor Detection

Olebra Comfort[1], Osuagwu Oliver[2] and Chukwudebe Gloria[3]

[1,2]Department of Computer Science, ImoStateUniversity, Owerri
Department of EEE, Federal University of Technology, Owerri

Abstract

This paper presents the concept of unauthorized access through equipment hardware termed "Hardware backdoor", which is the act of remotely gaining access to a computerized equipment by planting a device that gives remote access to unauthorized persons. The objective of this work is to analyze the hexadecimal codes in the microcontroller of a newly procured equipment and verify that it operates at expected frequency. I was motivated by the recent discovery of remote access in United States Military grade equipment purchased from China, which enable the later perform e-espionage activities on countries using their equipments. The methodology adopted is Prototyping, in which the Proteus Simulation software will be used to simulate the behavior of the hexadecimal codes in the microcontrollers, extracted using hardware programmer. The expected result will be a digital oscilloscope displayed frequency that either shows similarity or not to the stored expected frequency.
Keywords: Microcontrollers, Microcontroller Based Backdoor, Hardware Backdoor

1.0 Introduction

Boko Haram conducted its operations more or less peacefully during the first seven years of its existence. That changed in 2009 when the Nigerian government launched an investigation into the group's activities following reports that its members were arming themselves. When the government came into action, several members of the group were arrested in Bauchi, sparking deadly clashes with Nigerian security forces which led to the deaths of an estimated 700 people [1]. During the fighting with the security forces Boko Haram fighters reportedly "used fuel-laden motorcycles" and "bows with poison arrows" to attack a police station. The group's founder and then leader Mohammed Yusuf was also killed during this time while still in police custody. After Yusuf's killing, Abubakar Shekau became the leader and held this position in January 2015.[5]
The unending war with terrorist group Boko Haram is a major cause for concern in Nigeria. Since 2009, the group has launched terrorist attacks on worship centers, schools, etc. The inability of Nigerian Military to bring a final end to the operations of this group prompted quest into what the Nigerian Army could be doing wrong.

Then an eye opener came through researchers CambridgeUniversity computing Laboratory. In his report, Cambridge researcher Sergei Storobogatov wrote that a microchip used by US military and manufactured by China contain a secret "backdoor" that it is "wide open" intellectual property theft, fraud, reverse engineering of the chip to allow introduction of Trojan.[2]

The problem therefore is:
1.Computer manufacturers such as China embed hardware-backdoors on non-volatile memory in chips like BIOS on PCs or in firmware of routers, and export these to target nations for sale at cheap prices:
The threat posed by this discovery is lethal for the following reasons

Military: Nigeria's war against the terrorist group "Boko Haram" could have lingered as a result of backdoors in military surveillance systems, compromised PCs and phones of top military officer, or a compromised network, and computerized equipment microcontrollers containing codes which carry out more functions than meet the eye, resulting in incessant siege laying and killing of Nigerian soldiers.

2.0 Literature Review

This review looks at the role computerized military combat and combat services equipment play in giving away its owners. A look of the Microcontroller:

A **microcontroller** (or **MCU** for microcontroller unit) is a small computer on a single integrated circuit. In modern terminology, it is a System on a chip or SoC. A microcontroller contains one or more CPUs (processor cores) along with memory and programmable

`

input/output peripherals. Program memory in the form of Ferroelectric RAM, NOR flash or OTP ROM is also often included on chip, as well as a small amount of RAM. Microcontrollers are designed for embedded applications, in contrast to the microprocessors used in personal computers or other general purpose applications consisting of various discrete chips.

Microcontrollers are used in automatically controlled products and devices, such as automobile engine control systems, implantable medical devices, remote controls, office machines, appliances, power tools, toys and other embedded systems. By reducing the size and cost compared to a design that uses a separate microprocessor, memory, and input/output devices, microcontrollers make it economical to digitally control even more devices and processes. *Mixed signal* microcontrollers are common, integrating analog components needed to control non-digital electronic systems.

Some microcontrollers may use four-bit words and operate at frequencies as low as 4 kHz, for low power consumption (single-digit milliwatts or microwatts). They will generally have the ability to retain functionality while waiting for an event such as a button press or other interrupt; power consumption while sleeping (CPU clock and most peripherals off) may be just nanowatts, making many of them well suited for long lasting battery applications. Other microcontrollers may serve performance-critical roles, where they may need to act more like a digital signal processor (DSP), with higher clock speeds and power consumption.[3][4]

3.0 Material and Methodology

Materials

1. Universal Programmer

Microcontroller

Fig 3.2 Microchip PIC16F887 40 pin

1. MicroC
2. PC
3. Proteus Simulator

Circuit Diagram

ii. The Probe(DC power source)

iii. The Amplifying and Switching Circuit

iv. The Tank or tune circuit

v. Resistors (sensors)

vi. The Antenna

Probe: This provides DC source of 5v. 5v is used because all digital ICs accept maximum of 5v and a minimum of 2.2 which is only used to save power.

Microcontroller: Fig 3.2 showsPIC16F887 which is a medium grade microcontroller.The microcontroller is a small and low-cost microcomputer, which is designed to perform the specific tasks of embedded systems like displaying information, receiving remote signals, etc. The general microcontroller consists of the processor, the memory (RAM, ROM, EPROM), Serial ports, peripherals (timers, counters), etc.

Amplifying and Switching Circuit: The transistor, ZTX321 is a bipolar- junction-transistor (BJT) used for amplifying signal output by the microcontroller. Microcontroller feeds signal into the base of RF transistors. The collector is the +ve lead that carries the amplified output to the tank circuit while the emitter is common to both. Emitter is the –ve lead that increases linearity and stability. This amplifier ensures signal from microcontroller is increased and that sufficient signal is supplied to the tank circuit.

Tank Circuit: Circuit diagram of the capacitor and inductor in parallel form tune circuit or tank circuit. The function of the capacitor is to temporarily store signal and convert electrical signal into electromagnetic signal which will be radiated out. The indicator is an LED(Light Emitting Diode) with the function of giving indication that the system is on.

Resistor R1 limits current going into the LED.

Resistor R2 divides the current between the base, the collector and the emitter.

`

Test signal Generator: Produces sets of patterns according to the programmed algorithm.
Resistors: Used for current management. It limits or regulates the flow of current thereby protecting the transistor and itself.
Analog pipeline: The pipeline delivers its delayed signal output to waveform analyzer.
Multiplexer and Analog/Digital Converter: Converts the analog signal into a digital form. The output from the multiplexer is transferred to the hardware interface.
Hardware Interface: fig 3.1 show the chip programmer. The power supply of the Silicon from DUT (Device Under test) is the hardware interface (the programmer). Each microcontroller has its embedded code produced in the hardware interface. The hardware interface receives the output from the multiplexer and sends it to PC which is the user interface.
PC This is the user interface through which hardware interface (programmer) communicates between microcontroller and user.
Methodology

The Prototyping methodology used for this work is in two phases. First, the hardware programmer is used to read and store codes inside the microcontroller received from the equipment manufacturer, that shows the expected hexadecimal code as well as read the hexadecimal code of the microcontroller a newly procured equipment and these hexadecimal codes are stored in a directory with different names. Next the Proteus Simulation software is used to draw a schematic diagram of Microcontroller. The design is a simulation of the commands output by a microcontroller embedded in military grade computerized equipment such as the T72 and T70 and analysis of simulation result by comparing its frequency based on signal-to-noise ration to a previous expected behavior. Disparity in expected outcome leads to further probe through the use of disassemble which converts the extracted machine code to human readable assembly language so as to investigate embedded code for hardware backdoor. A microcontroller (DUT) is connected to hardware programmer which has been previously installed to its user interface (PC). The programmer environment enables you select the chip you are working with if you have universal programmer) but if it is customized for only a particular chip, then that chip must be used. Available microcontrollers include PIC(Programmable Interrupt Circuit), Atmel, FPGA etc.

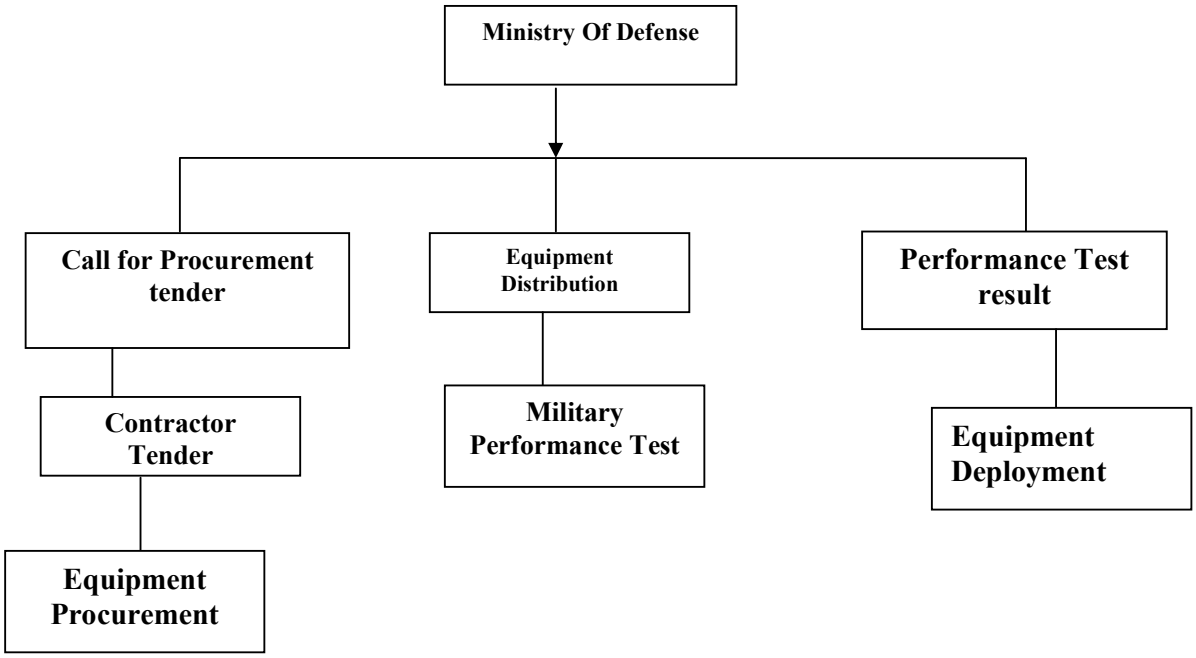

Fig 3.1 High Level Model of the Proposed System

`

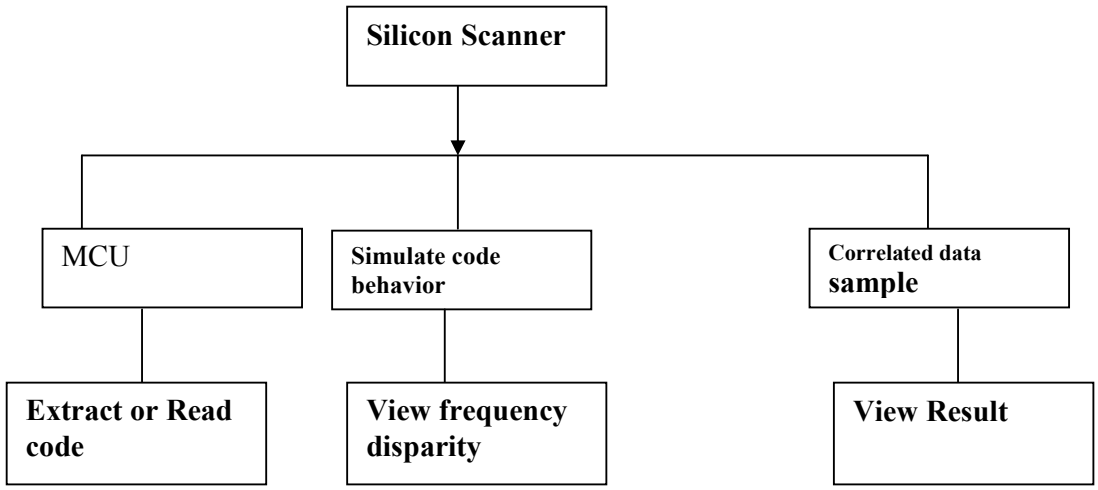

Fig. 3.2 Overall Data Flow Diagram of The New System Experiment

Circuit Diagrams

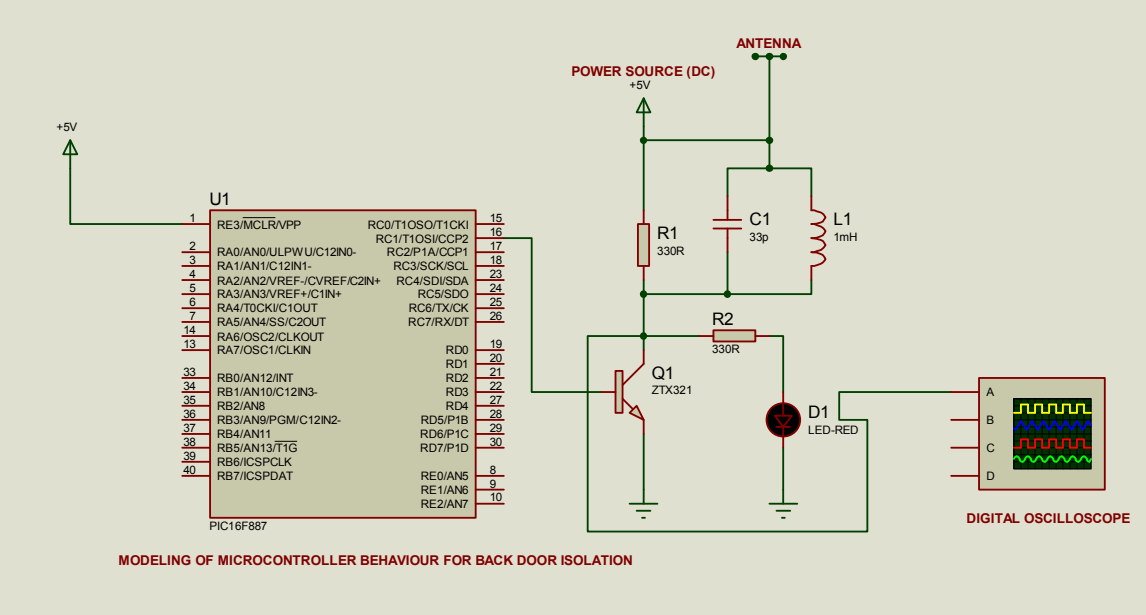

Fig 3.3 Circuit Diagram of PIC16F887 Before test

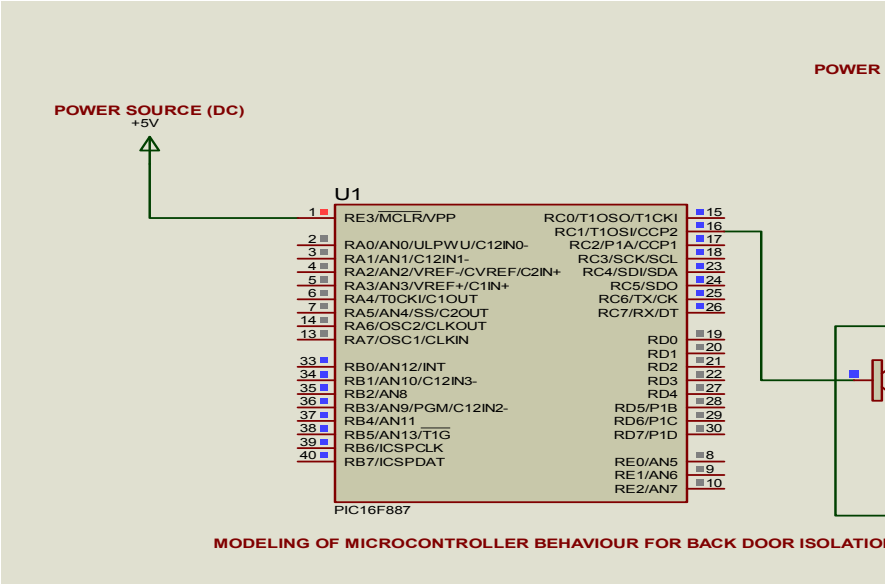

Fig3.4 Circuit Diagram of PIC16F887 During Test: Red LED Blinking

`

Experiment Explanation

Using the programmer IDE, a series of hexadecimal files can be extracted which form input into simulator. The simulator has a probe with DC source of 5v. 5v is used because all digital ICs accept maximum of 5v and a minimum of 2.2 which is only used to save power.

The microcontroller used for simulation is a medium grade (16F887) which is one of the latest products from Microchip. "F" added to the range "16" signifies it is of flash type which means its content can be erased and chip reprogrammed. The chip is a 40 pin, low price, high quality and easily available IC that operates between 0 - 20MHz with internal oscillator frequency range of 8-31MHz.

The transistor ZTX321 used in the simulation is a bipolar transistor. The choice of the transistor is because chosen design is a radio design that can detect and transmit waveform frequency.

The capacitor used is ceramic capacitor. Schematic diagram of capacitor and inductor in parallel form tune circuit or tank circuit. The function of the capacitor is to convert electrical signal into electromagnetic signal which will be radiated out. The indicator is an LED (Light Emitting Diode) with the function of giving indication that the system is on.

Resistor R1 limits current going into the LED.

Resistor R2 divides the current between the base, the collector and the emitter.

.

Current flows from RC2 into the base of the Transistor Q3. This transistor receives signal through its base, outputs it through the collector (upper part) while the emitter (the lower part earths the transistor). Any of the two transistor types can be used (NPN or PNP) The function of the transistor is to amplify and switch. The initial current flowing from R2 will not be sufficient to power the tank circuit, hence it is fed into transistor ZTX321 which amplifies the current and switches to R3 through the collector, until the tank circuit is fed. At tank circuit electrical signal is then converted to electromagnetic signal that is radiated out in waveform. The frequency of the waveform represents the signal to noise ration that is evaluated and compared to previously stored data. If there is disparity when frequency is compared, further analysis is initialized by decompiling hexadecimal output from the MIC into human readable assembly language. Analysis for different test if suspicious is followed by code investigation.

The simulation procedure involves right clicking on the schematic, selecting edit, selecting hex file of Device Under Test, check if the frequency displayed by the digital oscilloscope is the same as that displayed by the hex file from the microcontroller with expected behavior

`

`

Result

Fig 3.5 Simulation output of PIC16F887 microcontroller based on expected behavior

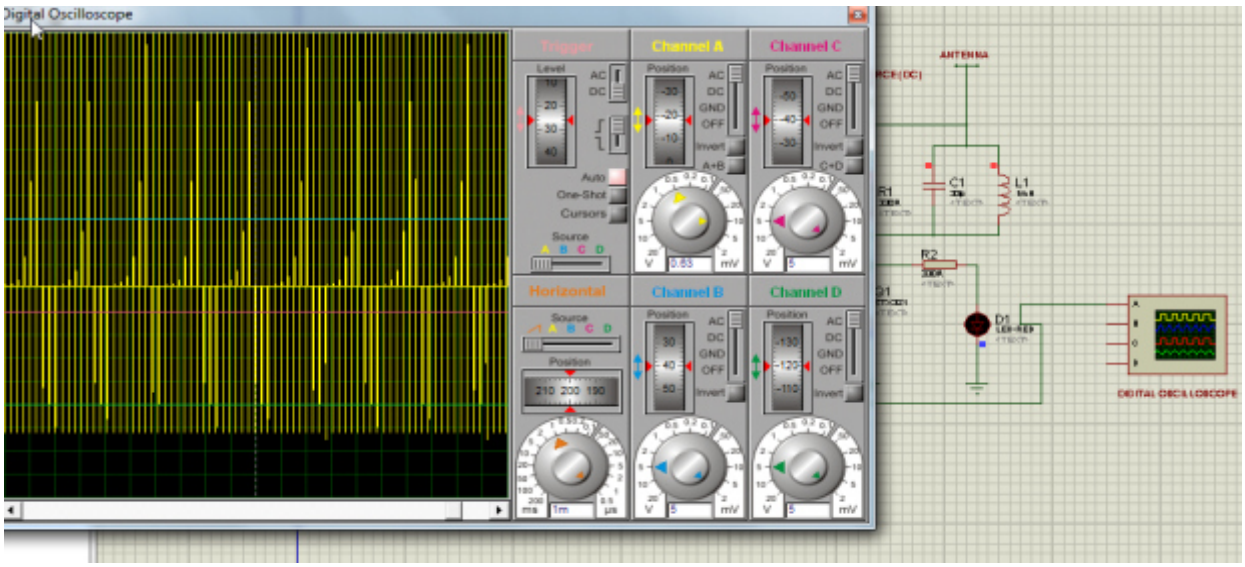

Fig 3.6 Simulation output of PIC16F887 microcontroller based on newly procured equipment

Result Description

The experiment involves the reading of codes embedded in microcontrollers of particular equipment. One of which is accepted as functional and has passed security evaluation test is used to compare to the codes embedded in subsequently purchased equipment microcontrollers. These equipment/microcontrollers carry out the same function at face value, however, the hexadecimal codes embedded are not the same. Also when these codes are simulated using the Proteus simulated, a frequency disparity is noticed. Figures 3.2 and 3.3 show frequencies of the hexadecimal codes simulated using the Proteus software.

Summary

This paper x-rayed the role of Microcontrollers in computerized equipment and how different microcontrollers operate at different frequencies, depending on the level of functions they are built to perform. It also showed that microcontrollers could be embedded with hardware backdoors and still possess produce expected behavior. if the frequency is not as expected, then a remote activity could be going on.

It also showed the insecurity of computerized surveillance equipment procured from unreliable sources, as the medium where they operate such as free space, is vulnerable to attack by their target enemy.

Conclusion

Scanning of a Field Programmable Gate Array chips in equipments procured by the Nigerian Military brings to the fore the need to be security conscious with respect to combat and combat service equipments. The procurement of computerized weapons with all of its advantages over the enemy can have adverse effect if not monitored with utmost caution.

Procurement policy presently practiced by Nigerian Army ensures availability without security considerations. Hence the overall system cannot be said to be dependable.

We conclude that simulation of frequencies of hex files embedded in microcontrollers of with the same grade and function reveal disparity or similarity of the MCU functions.

Recommendation

1. Hardware backdoors are set to be triggered during functional tests of new equipment hence Microcontroller of procured equipment should be tested even before performance tests.
2. The engineering section of the Nigerian army should be equipped with specified software and hardware required to carry out the device single or chain scanning.
3. Devices/equipment that fails the hardware security assurance test must not be put to use until compromised parts are replaced.
4. Computerized military grade equipment with huge advantage of being capable of detecting the enemy must be scanned for backdoor and Trojan and must pass both performance and security assurance tests so that it will not provide remote access for the target.

References

[1] **Adama, N.(2009)**: "Scores Die As Fighteers Battle Nigerian Police" Retrieved August 12, 2016

[2] **Skorobogatov, S.**, **Cowoods, C. (2012):** "Breakthrough in Silicon Scanning Discovers Backdoor in Military Chip" Accessed December 12, 2016,"Boko Haram's four-year Reign of Terror", Channel4 News 8 May, 2014. Retrieved December 2016

[3] **Augarten, S. (1983):** "The Most Widely Used Computer On a Chip"

[4] **Jan A. (1994):** "Basic Microcontroller

[6] Laprie Jean-Claude (1985) "Dependable Computing and Fault tolerance: Concepts and terminologies. In the Proceedings 15th Int' Symposium. Fault-Tolerant computing, pages 2-11, Los Alomitos, California, USA. Computer Society Press.

[7] Gray Jim and Daniel Siewiorek(1991) "High Availability Computer Systems. IEEE Computer, 24(9): 39-48

Appendix A

```
//Expected Normal Code in Microcontroller PIC16F887
unsigned short current_duty, old_duty, current_duty1, old_duty1;
void InitMain() {
ANSEL  = 0;                    // Configure AN pins as digital
  ANSELH = 0;
  C1ON_bit = 0;                  // Disable comparators
C2ON_bit = 0;

  PORTA = 255;
  TRISA = 255;                   // configure PORTA pins as input
PORTB = 0;                     // set PORTB to 0
  TRISB = 0;                     // designate PORTB pins as output
  PORTC = 0;                     // set PORTC to 0
  TRISC = 0;                     // designate PORTC pins as output
  PWM2_Init(1500000000);           // Initialize PWM2 module at 1.5GHz
}

void main() {
InitMain();
  current_duty1 = 16;              // initial value for current_duty1

  PWM2_Start();                  // start PWM2
  PWM2_Set_Duty(current_duty1);      // Set current duty for PWM2

while (1) {                    // endless loop

if (RA2_bit) {                 // button on RA2 pressed
     Delay_ms(40);
current_duty1++;             // increment current_duty1
     PWM2_Set_Duty(current_duty1);
    }

if (RA3_bit) {                 // button on RA3 pressed
     Delay_ms(40);
current_duty1--;             // decrement current_duty1
     PWM2_Set_Duty(current_duty1);
    }

   Delay_ms(5);                  // slow down change pace a little
  }
}
```

APPENDIX B

```
//prototype backdoor code embedded on microcontroller PIC16F887

unsigned short current_duty, old_duty, current_duty1, old_duty1;

void InitMain() {
ANSEL  = 0;                    // Configure AN pins as digital
  ANSELH = 0;
  C1ON_bit = 0;                  // Disable comparators
C2ON_bit = 0;

  PORTA = 255;
  TRISA = 255;                   // configure PORTA pins as input
PORTB = 0;                     // set PORTB to 0
  TRISB = 0;                     // designate PORTB pins as output
  PORTC = 0;                     // set PORTC to 0
  TRISC = 0;                     // designate PORTC pins as output
  PWM1_Init(945000000);            // Initialize PWM1 module at 945MHz
  PWM2_Init(1500000000);           // Initialize PWM2 module at 1.5GHz
}

void main() {
InitMain();
  current_duty  = 16;              // initial value for current_duty
  current_duty1 = 16;              // initial value for current_duty1

  PWM1_Start();                  // start PWM1
  PWM2_Start();                  // start PWM2
  PWM1_Set_Duty(current_duty);       // Set current duty for PWM1
  PWM2_Set_Duty(current_duty1);      // Set current duty for PWM2

while (1) {                    // endless loop
if (RA0_bit) {                 // button on RA0 pressed
    Delay_ms(40);
current_duty++;                // increment current_duty
    PWM1_Set_Duty(current_duty);
   }

if (RA1_bit) {                 // button on RA1 pressed
    Delay_ms(40);
current_duty--;                // decrement current_duty
    PWM1_Set_Duty(current_duty);
   }

if (RA2_bit) {                 // button on RA2 pressed
    Delay_ms(40);
current_duty1++;               // increment current_duty1
    PWM2_Set_Duty(current_duty1);
```

```
    }

if (RA3_bit) {                  // button on RA3 pressed
     Delay_ms(40);
current_duty1--;                // decrement current_duty1
     PWM2_Set_Duty(current_duty1);
    }

   Delay_ms(5);                 // slow down change pace a little
  }
}
```

GENERATED HEX CODE

FOR APPENDIX A

```
:02000000A62830
:0E0006008312031322088A00210882000800DA
:0C00140083120313 9B011D129D120800B3
:1000200083120313FB01FA01F9018030F800F10C8F
:10003000F00C031C24287408F90775080318750FC1
:10004000FA070318FB0A0310F01F2C287408FA079C
:1000500075080318750FFB07FB0CFA0CF90CF80C6C
:10006000031C17287B08F3007A08F2007908F100D6
:060070007808F000080012
:1000760083160317880189018312871388 13FF30BB
:100860003138500FF308316850083128601831 6CD
:10096008601831287018316870183121210921 03C
:0A00A600FF30831692000A200800C4
:1000B000831603131208013EF000F1010318F10A40
:1000C00083122308F400F50110200630F2007008B6
:1000D000F4007108F500720803197328F50CF40C8C
:1000E000F513FF3E6C280430F1007408F00071082D
:1000F00003197E28F00D7010FF3E7828303070050F
:10010000F3007408F0007508F100F10CF00CF11325
:10011000F10CF00CF11370089B000F309D05730873
:040120009D04080032
:1001240003208A110A128000840AA10A0319A20A70
:08013400F003031D92280800EE
:10013C00831603138710831212150C309D040800CC
:10014C003B201030A0009E202008A3005820051D45
:10015C00BC286830FC00E430FD00FD0BB328FC0B20
:10016C00B3280000A00A2008A3005820851DCB2826
:10017C006830FC00E430FD00FD0BC228FC0BC228EB
:10018C000000A0032008A30058200D30FC00FB3019
:10019C00FD00FD0BCF28FC0BCF2800000000AD2884
:0201AC00D62853
:04400E00F22F000786
:00000001FF
```

GENERATED HEX CODE FROM APPENDIX B

:02000000F428E2
:0C0006008312031395011712971208 00D3
:0E0012008312031322088A00210882000800CE
:1000200083120313FB01FA01F9018030F800F10C8F
:10003000F00C031C24287408F90775080318750FC1
:10004000FA070318FB0A0310F01F2C287408FA079C
:1000500075080318750FFB07FB0CFA0CF90CF80C6C
:10060000031C17287B08F3007A08F2007908F100D6
:060070007808F000080012
:0C00760083120313 9B011D129D12080051
:1000820083160313120801 3EF000F1010318F10A6E
:1000920083122408F400F50110200630F2007008E3
:1000A200F4007108F500720803195C28F50CF40CD1
:1000B200F513FF3E55280430F1007408F000710872
:1000C20003196728F00D7010FF3E6128303070056B
:1000D200F3007408F0007508F100F10CF00CF11354
:1000E200F10CF00CF11370089B000F309D057308A2
:0400F2009D04080061
:1000F6008316031312080 13EF000F1010318F10AFA
:1001060083122408F400F50110200630F20070086E
:10011600F4007108F500720803199628F50CF40C22
:10012600F513FF3E8F280430F1007408F0007108C3
:10013600031 9A128F00D7010FF3E9B283030700582
:10014600F3007408F0007508F100F10CF00CF113DF
:10015600F10CF00CF113700895000F309705730839
:040166009 7040800F2
:10016A0083160313071183121215 0C309704080023
:10017A0009208A110A128000840AA10A0319A20A14
:08018A00F003031DBD2808006D
:1001920083160313871083121215 0C309D04080076
:1001A2008316031788018901831287138813FF308E
:1001B20003138500FF308316850083128601831 6A0
:1001C2008601831287018316870183121210921 00F
:1001D200FF308316920003201210921 0FF30831614
:0601E20092003B20080022
:1001E800D1201030A3001030A000B520C92023086A
:1001F800A4007B202008A4004120051C1029683099
:10020800FC00E430FD00FD0B0729FC0B072900006A
:10021800A30A2308A4007B20851C1F296830FC0042
:10022800E430FD00FD0B1629FC0B16290000A30382
:100238002308A4007B20051D2E296830FC00E4302B
:10024800FD00FD0B2529FC0B25290000A00A20082C
:10025800A4004120851D3D296830FC00E430FD00E4
:10026800FD0B3429FC0B34290000A0032008A4004E
:1002780041200D30FC00FB30FD00FD0B4129FC0B3B
:0A028800412900000000129482967
:04400E00F22F000786:00000001FF