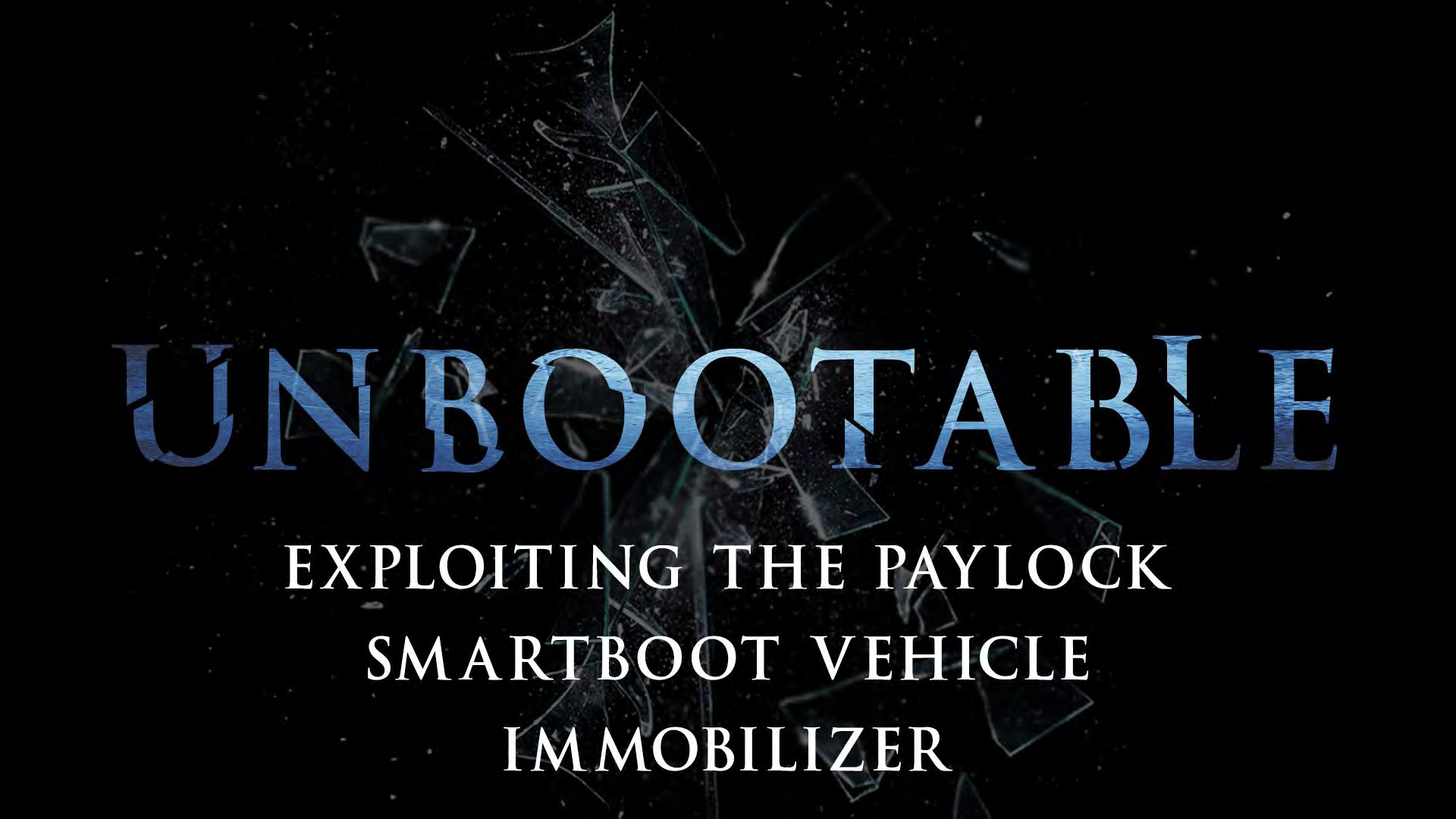

UNBOOTABLE
EXPLOITING THE PAYLOCK
SMARTBOOT VEHICLE
IMMOBILIZER

what is a boot?

old school

PayLock SmartBoot

What Do I Do After My Car Gets Booted?
by NYC Department of Finance

well… how else can I remove it?

the wrong way

the wrong way

the wrong way

Reverse Engineering the PayLock SmartBoot

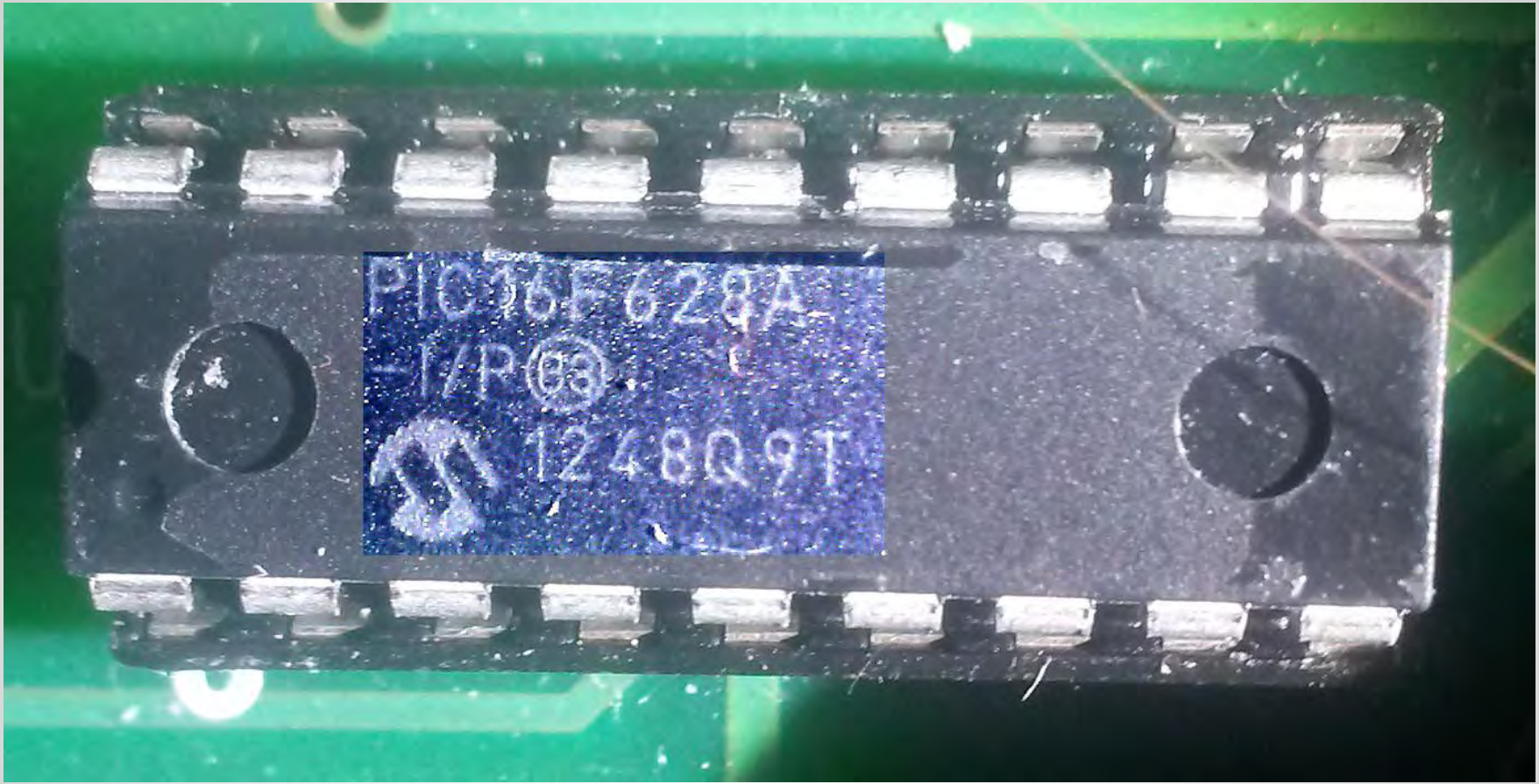
disassembly

[insert disassembly image]


WARNING
Do not attempt to move this vehicle.
Your vehicle will be damaged by
this device if driven while attached.
1 6
2 7
3 8
4 9
5 0
* #
FOR REMOVAL 24/7
CALL TOLL FREE:
1-866-404-6373
This device is Municipal Property.
Unauthorized attempts at removal
or vandalism of this device,
is a crime and will be prosecuted
to the full extent of the law.
THIS DEVICE IS ELECTRONICALLY TRACKED


Magnetic Sensor Systems
Van Nuys, CA USA
S-18-100-30LC4PY
PAYLOCK INC
PLK01-0016-5
2006 R
LOCK
UNLOCK
Q1
Q2
J3
12-KEYPAD
J5
R11
R10
R2
R1
ED1
PROGRAM
SLEEP
U1
C2
J4
R3
R9
J2
OUT
C1
BUZ1
BUZZER

the guts

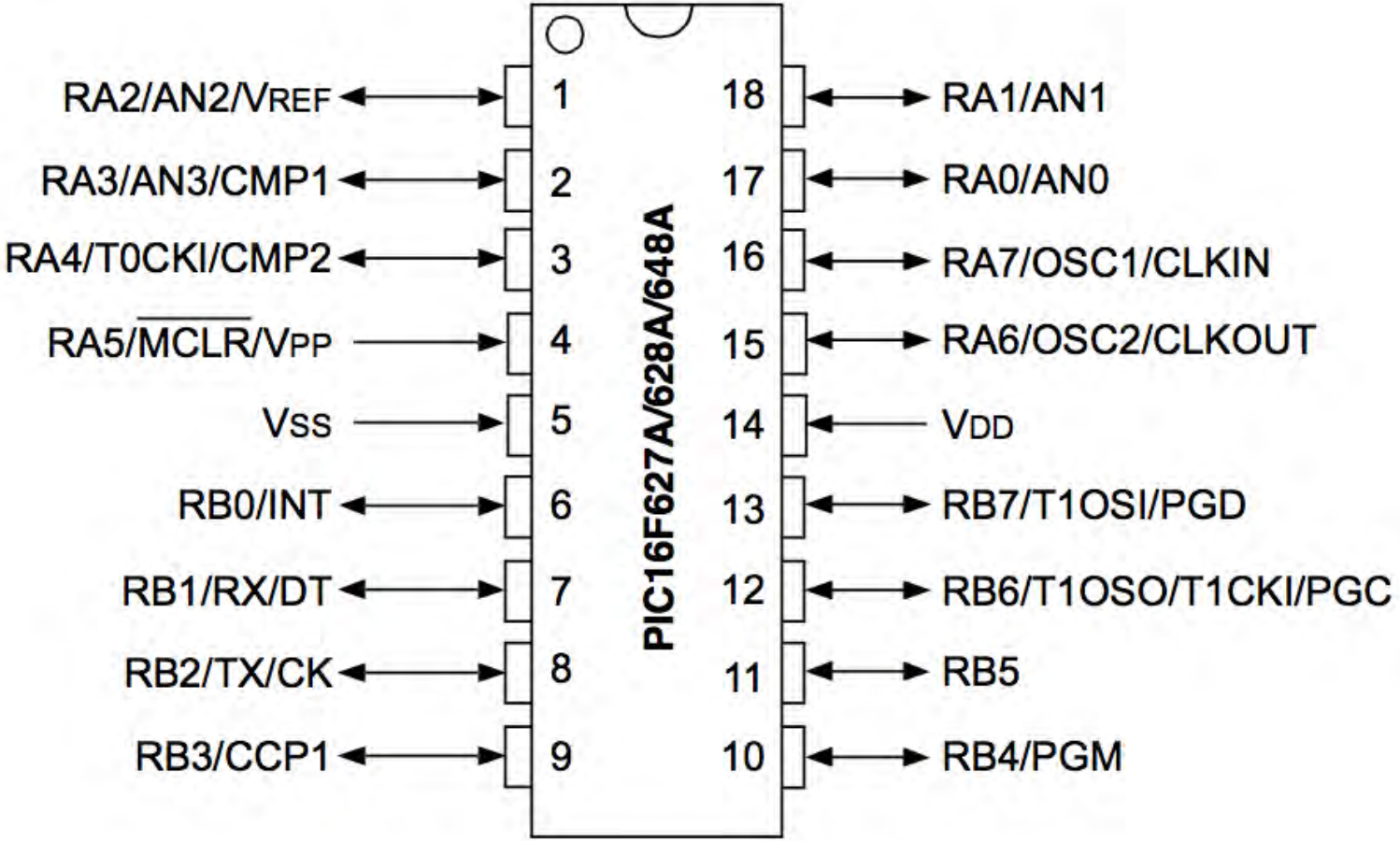


PIC16F628A

PIC16F627A/628A/648A
Data Sheet

Flash-Based, 8-Bit CMOS

Microcontrollers with nanoWatt Technology


PIC16F627A/628A/648A
RA2/AN2/VREF
1
RA3/AN3/CMP1
2
RA4/T0CKI/CMP2
3
RA5/MCLR/VPP
4
VSS
5
RB0/INT
6
RB1/RX/DT
7
RB2/TX/CK
8
RB3/CCP1
9
18
RA1/AN1
17
RA0/AN0
16
RA7/OSC1/CLKIN
15
RA6/OSC2/CLKOUT
14
VDD
13
RB7/T1OSI/PGD
12
RB6/T1OSO/T1CKI/PGC
11
RB5
10
RB4/PGM

use the tools at your disposal

Topic: Arduino as a Pic Programmer! (Read 109115 times)

Previous Topic - Next Topic

Soranne

Jr. Member

Posts: 72

Karma: 4 [add]

Arduino rocks

My website

Arduino as a Pic Programmer!

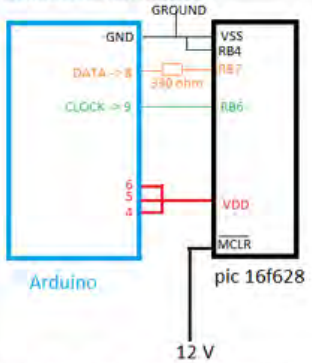
Feb 20, 2012, 07:12 pm Last Edit: May 06, 2012, 06:38 pm by Soranne Reason: 1

Hi everyone!

Here is my first version of my PIC programmer : program your PIC from USB! Cheap and easy!
Feel free to do what you want with this project, modify it, upgrade it publish it,... but just tell me what you've done so that I can share it with everybody

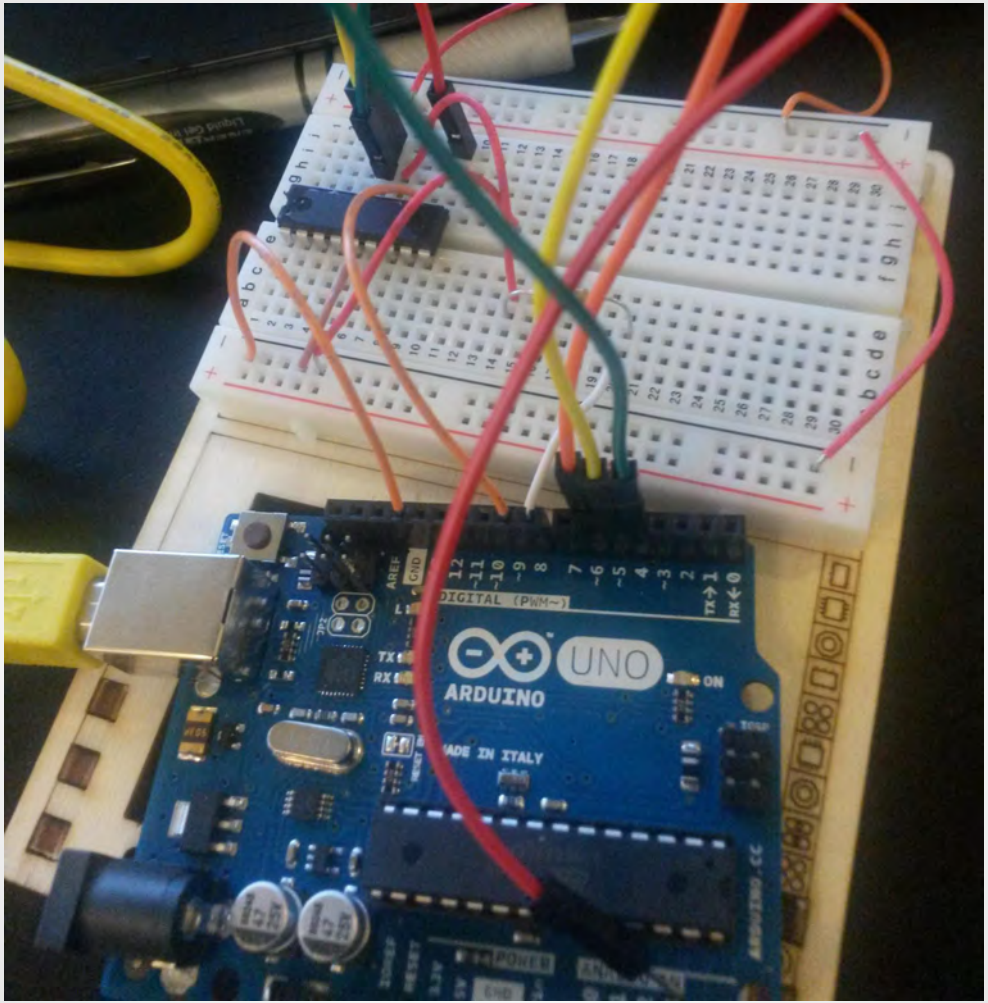
It works with PIC 16F628 but should work with most pic16F; if you can try at home leave a message so that I can list the working PICs here!

Here is the circuit you need to make :



You can change the resistor value between 330 and more.
Always RESET Arduino before putting 12V (I don't know if we need to, but that's just a safe practice)

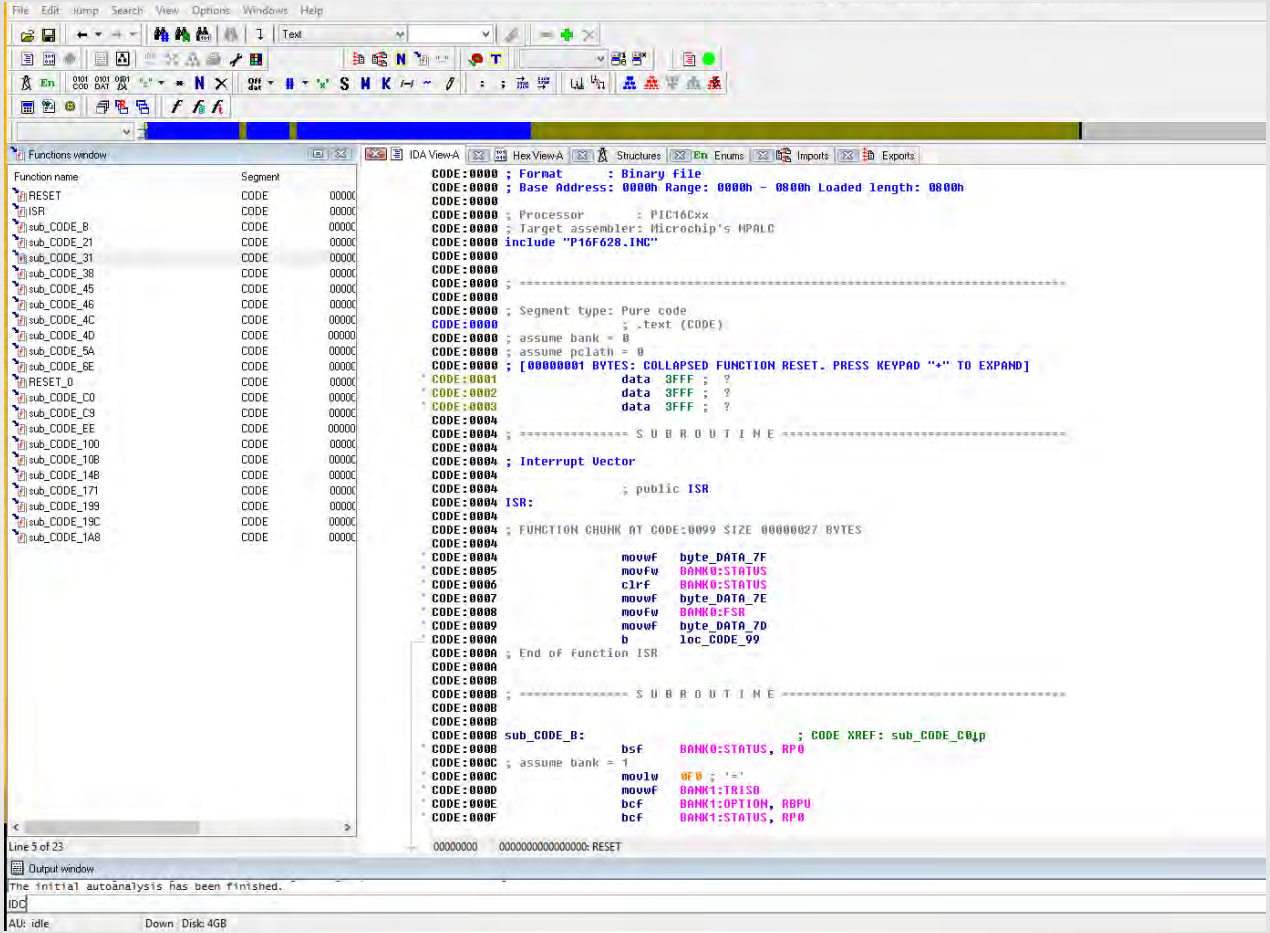
And here is the code :


UNO
ARDUINO
ON
DIGITAL (PWM~)
POWER

we have output!

```
10100010001111  00000110101110  10100010001010
11111111111111  11000000001110  01010000000011
11111111111111  10000000100001  00000010000110
11111111111111  01100000000011  01111000000110
00000011111111  10100000011111  00000000001000
00100000000011  11000000001101  00101010101110
00000110000011  10000000100001  01111010000110
00000011111110  01100000000011  00000000001000
00100000000100  10100000011111  00101010101110
00000011111101  11000000001011  01111100000110
10100010011001  10000000100001  00000000001000
01011010000011  01100000000011  00101010101110
11000011110000  10100000011111  01111110000110
00000010000110  11000000000111  00000000001000
01001110000001  10000000100001  00101010101110
01001010000011  00100000101110  01000000000011
```

...



hexify and disassemble

looks nice. what are we missing?

```
CODE:0031
CODE:0031 ; =============== S U B R O U T I N E =======================================
CODE:0031
CODE:0031
CODE:0031 sub_CODE_31:                            ; CODE XREF: sub_CODE_14B+5↓p
CODE:0031                 bsf     BANK0:STATUS, RP0
CODE:0032 ; assume bank = 1
CODE:0032                 movwf   BANK1:EEADR
CODE:0033                 bsf     BANK1:STATUS, RP0
CODE:0034                 bsf     BANK1:EECON1, RD
CODE:0035                 movfw   BANK1:EEDATA
CODE:0036                 incf    BANK1:EEADR, f
CODE:0037                 b       loc_CODE_8A
CODE:0037 ; End of function sub_CODE_31
CODE:0037
CODE:0038 ; assume bank = 0
CODE:0038
```

EE... EE... EEPROM?

13.3 Reading the EEPROM Data Memory

To read a data memory location, the user must write the address to the EEADR register and then set control bit RD (EECON1<0>). The data is available, in the very next cycle, in the EEDATA register; therefore it can be read in the next instruction. EEDATA will hold this value until another read or until it is written to by the user (during a write operation).

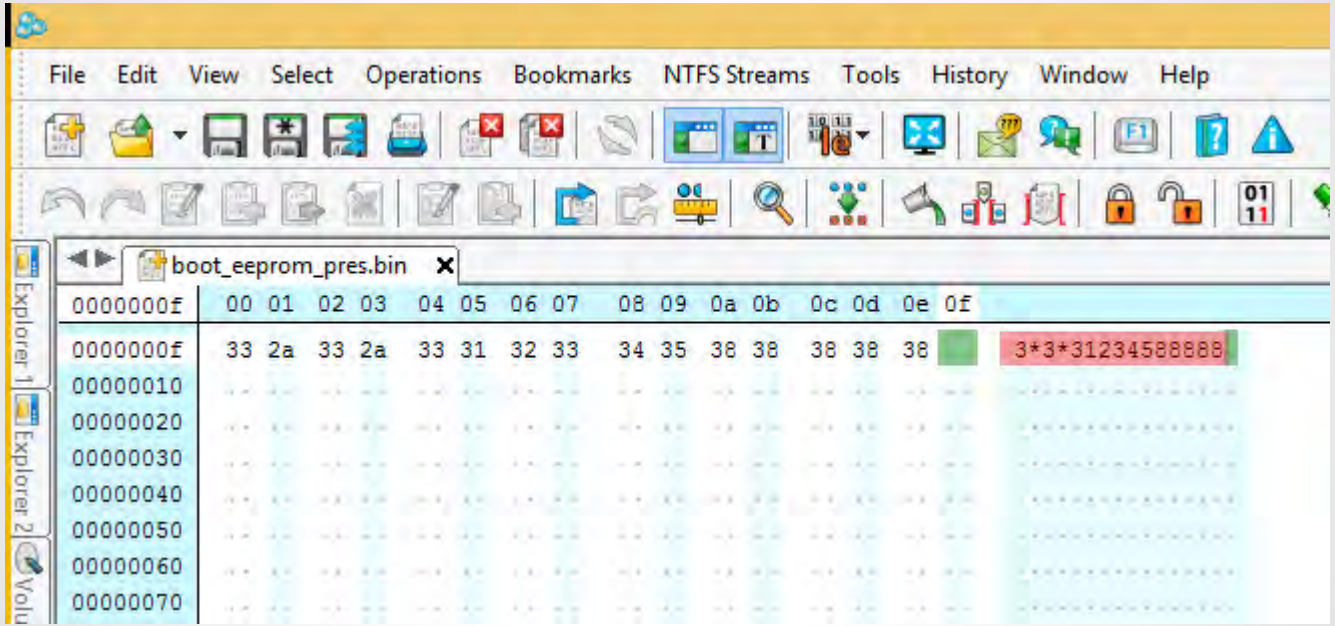
EXAMPLE 13-1: DATA EEPROM READ

```
BSF     STATUS, RP0     ;Bank 1
MOVLW   CONFIG_ADDR     ;
MOVWF   EEADR           ;Address to read
BSF     EECON1, RD      ;EE Read
MOVF    EEDATA, W       ;W = EEDATA
BCF     STATUS, RP0     ;Bank 0
```

EEPROM

update our arduino code

dump the eeprom data


File Edit View Select Operations Bookmarks NTFS Streams Tools History Window Help
boot_eeprom_pres.bin
0000000f 00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f
0000000f 33 2a 33 2a 33 31 32 33 34 35 38 38 38 38 38
3*3*31234588888
00000010
00000020
00000030
00000040
00000050
00000060
00000070
Explorer 1
Explorer 2

et voilà

[insert unlock video]

the right way

UNBOOTABLE

EXPLOITING THE PAYLOCK SMARTBOOT VEHICLE IMMOBILIZER

by fluxist

(and don’t forget to pay your parking tickets)