Hacking hardware for fun and profit

András Veres-Szentkirályi
vsza@silentsignal.hu

HACKTIVITY
2011

SILENT SIGNAL
VÉSZJELZÉS HELYETT...

WWW.SILENTSIGNAL.HU

Disclaimer

$$\boxed{I = \frac{U}{R}}$$

Because of the numerous generalizations electrical engineers should only watch this talk accompanied by a friend.

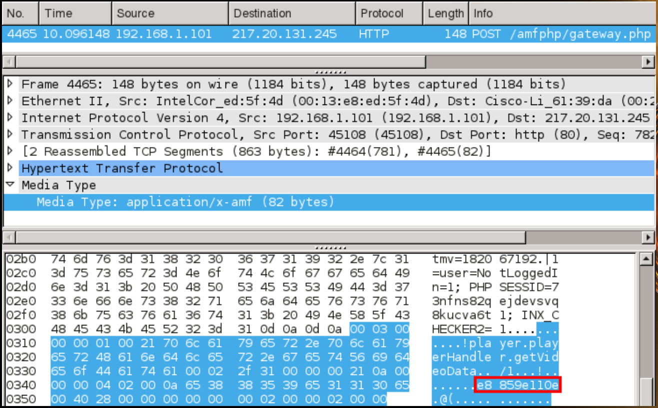
Mission definition

Let's start with something most software hackers would be familiar with:

"I'd like to have a solution that makes me able to download videos from a less-known (read: no ready solution available) site"

Sniffing around

Sniff the traffic with Wireshark, find ID . . .



```
No.  Time      Source         Destination     Protocol  Length Info
4465 10.096148 192.168.1.101  217.20.131.245  HTTP      148 POST /amfphp/gateway.php

▷ Frame 4465: 148 bytes on wire (1184 bits), 148 bytes captured (1184 bits)
▷ Ethernet II, Src: IntelCor_ed:5f:4d (00:13:e8:ed:5f:4d), Dst: Cisco-Li_61:39:da (00:2
▷ Internet Protocol Version 4, Src: 192.168.1.101 (192.168.1.101), Dst: 217.20.131.245
▷ Transmission Control Protocol, Src Port: 45108 (45108), Dst Port: http (80), Seq: 782
▷ [2 Reassembled TCP Segments (863 bytes): #4464(781), #4465(82)]
▷ Hypertext Transfer Protocol
▽ Media Type
    Media Type: application/x-amf (82 bytes)

02b0  74 6d 76 3d 31 38 32 30  36 37 31 39 32 2e 7c 31   tmv=1820 67192.|1
02c0  3d 75 73 65 72 3d 4e 6f  74 4c 6f 67 67 65 64 49   =user=No tLoggedI
02d0  6e 3d 31 3b 20 50 48 50  53 45 53 53 49 44 3d 37   n=1; PHP SESSID=7
02e0  33 6e 66 6e 73 38 32 71  65 6a 64 65 76 73 76 71   3nfns82q ejdevsvq
02f0  38 6b 75 63 76 61 36 74  31 3b 20 49 4e 58 5f 43   8kucva6t 1; INX_C
0300  48 45 43 4b 45 52 32 3d  31 0d 0a 0d 0a 00 03 00   HECKER2= 1.......
0310  00 00 01 00 21 70 6c 61  79 65 72 2e 70 6c 61 79   ....!pla yer.play
0320  65 72 48 61 6e 64 6c 65  72 2e 67 65 74 56 69 64   erHandle r.getVid
0330  65 6f 44 61 74 61 00 02  2f 31 00 00 00 21 0a 00   eoData.. /1...!..
0340  00 00 04 02 00 0a 65 38  38 35 39 65 31 31 30 65   ......e8 859e110e
0350  00 40 28 00 00 00 00 00  00 02 00 00 02 00 00      .@(..... .......
```

SILENT SIGNAL
VÉSZJELZÉS HELYETT...

WWW.SILENTSIGNAL.HU

Development

. . . develop and debug Python script with ease!

```
__amftpl__ = ('\0\x03\0\0\0\x01\0!player.'
'playerHandler.getVideoData\0\x02/1\0\0\0'
'!\n\0\0\0\x04\x02\0\n{vid}\0@(\0\0\0\0\0'
\0\x02\0\0\x02\0\0')

amfreq = __amftpl__.format(vid=url2vid(url))
amfresp = urlopen(
'http://videosite/gateway.php', amfreq).read()

return re.findall(
r'http://[a-zA-Z0-9/._]+\.(?:mp4|webm)', amfresp)
```

SW conclusion

- Speed: blazing fast cycles
  - edit – run – debug in seconds
- Cost: practically nothing
  - you already have a PC
- Scope: limited to existing underlying (mostly hardware) infrastructure

First barrier: cost

So, you'd like to hack HW, right? You just need . . .

- a development board
- a programmer
- an (in-circuit) debugger
- an oscilloscope
- a logic analyzer
- various incompatible cabling
- bloated, closed, proprietary software packages

Remember DVD Jon?

Second barrier: knowledge

Oh, you got money?

Then you have a comprehensive knowledge about . . .

- fscked up languages
  - WTF BASIC, abnormal C-variants
- matching compilers
- the internals of the target hardware
  - 400 pg. datasheet
- proprietary peripheals and drivers

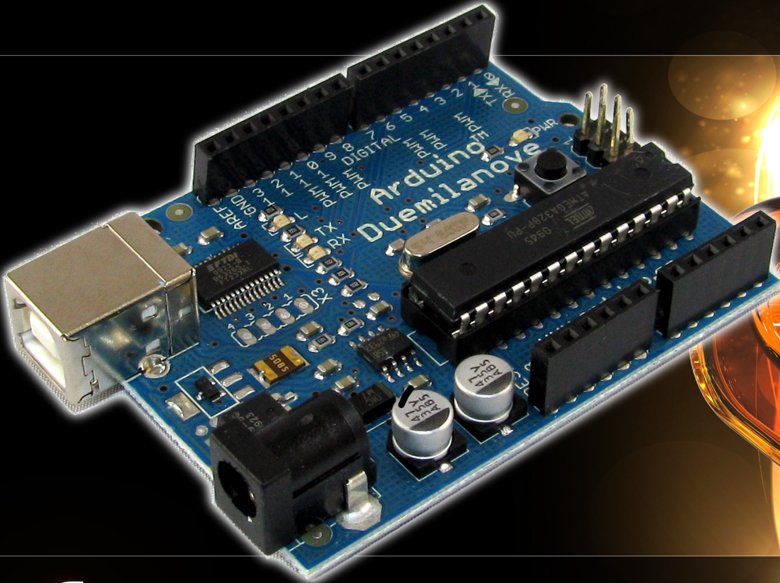
. . . right?

Solutions in the software world

We already had these problems in the software development world and solved it:

- free quality compilers are available for great languages
- abstraction available in frameworks hides the underlying layers
- you don't have to use a debugger, when a `print` is enough

Aren't these applicable to hardware as well?


Arduino
Duemilanove
DIGITAL
AREF
GND
TX
RX
PWR
ATMEGA328P-PU

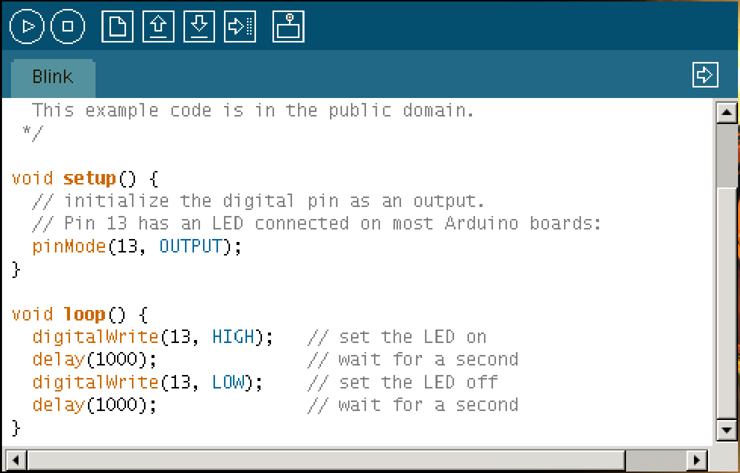
What is it?

- Full-fledged general purpose microcontroller board
- Think of it as a computer
- It has a CPU: it's called a microcontroller
- It has memory: 1k RAM
- It has storage
  - 16k Flash (code)
  - 512b EEPROM (data)
- It has ports: USB, RS-232, I^{2}C, SPI, . . .

What's the big deal?

- It's cheap ($30 UNO, clones under $20)
- The software runs on Linux, Mac and Windows
- The drivers **actually work** on these platforms
- The software **and** hardware is free (as in free speech)
- The leaning curve is really smooth with the dead simple IDE (see next slide)
- It is **not** made by a chip maker (important!)
- "It is the Apple][of the open source prototyping movement – the first successful device that was able to build a significant following."

Hello World: blink



```
  This example code is in the public domain.
 */

void setup() {
  // initialize the digital pin as an output.
  // Pin 13 has an LED connected on most Arduino boards:
  pinMode(13, OUTPUT);
}

void loop() {
  digitalWrite(13, HIGH);    // set the LED on
  delay(1000);               // wait for a second
  digitalWrite(13, LOW);     // set the LED off
  delay(1000);               // wait for a second
}
```

Simplest I/O: serial port

- **Arduino**: `Serial.begin(9600);`
- **Pure C (from the official AVR datasheet)**

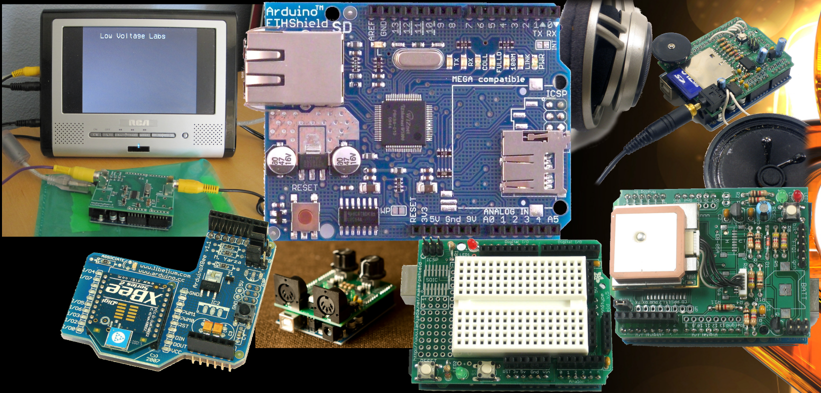
$$ubrr = \frac{F_{osc}}{16 \cdot (9600 - 1)}$$

```
/* Set baud rate */
UBRR0H = (unsigned char)(ubrr >> 8);
UBRR0L = (unsigned char)ubrr;
/* Enable receiver and transmitter */
UCSR0B = (1 << RXEN0) | (1 << TXEN0);
/* Set frame format: 8 data, 2 stop bits */
UCSR0C = (1 << USBS0) | (3 << UCSZ00);
```

Hardware addons: shields

“A product transcends being a mere product and becomes the core of an ecosystem when it’s easy to add things onto it and when the interface between the two stays stable enough that people feel comfortable committing resources to it over the long term. **With the early PCs, it was their slot connectors. With the iPod it’s the dock connector. With the Arduino, it’s the shield connector.** I think we all know there are problems with its current design, but the Arduino team is wise not to change it in an incompatible way because that hurts the Arduino ecosystem.”

Some of the more than 200 available shields



comms (Ethernet, WiFi, GSM, GPS, RFID), storage, multimedia (wave, MIDI), special control (motor), HID (displays, buttons)

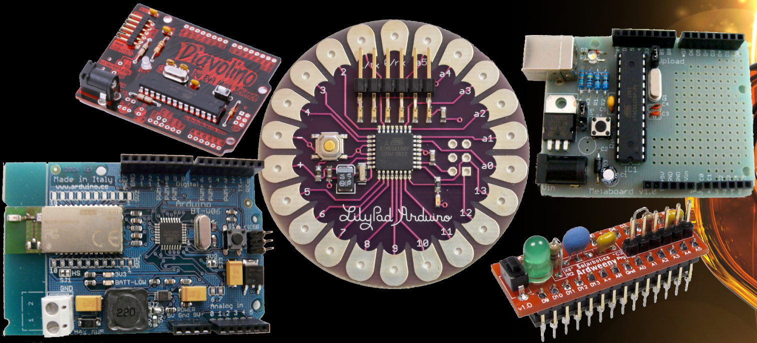
Software addons: libraries

Just like Python:

- batteries included (no pun indended)
- ready-made libraries for every task
- communcation: DMX, RS-485, PS/2, HTTP
- hardware I/O
  - stepper motors, button debouncing
  - LCD, LED, TFT displays
- access to internal peripheals: timers, I^{2}C
- because of C/C++, it's easy to contribute
- sane licenses

Close relatives

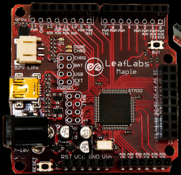
The design is CC-BY-SA – expand the ecosystem!

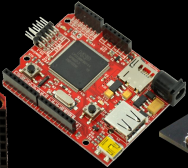
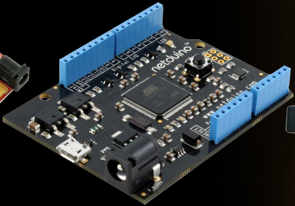


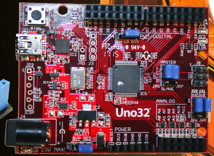
same microcontroller, same software stack, different hardware

Distant relatives

- compatible only in form factor (shields)
- different hardware and software stacks:
  - ARM (Cortex), .NET micro, PIC (ChipKit)
- interesting concept

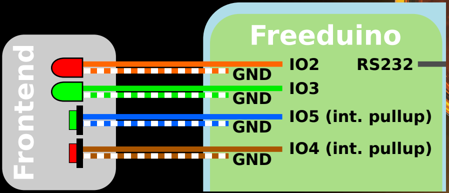


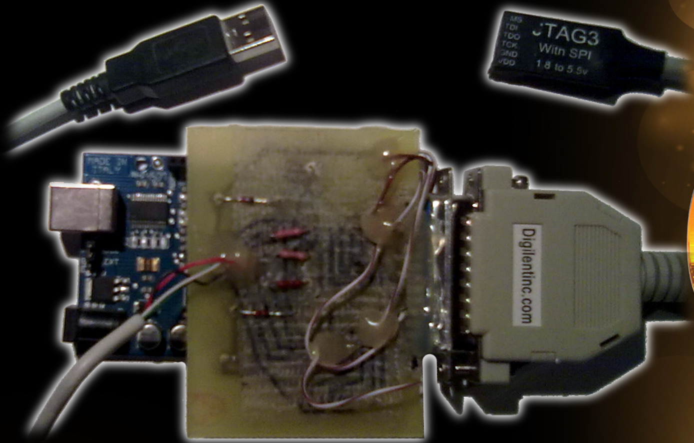


HackSense

- reports presence of hackers @ H.A.C.K.
- primitive frontend uses Arduino (LEDs and buttons)
- connected to a Linksys WRT54GL running OpenWRT using RS-232 and a simple serial protocol (one byte packets)

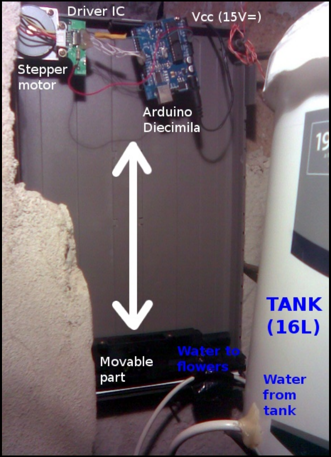


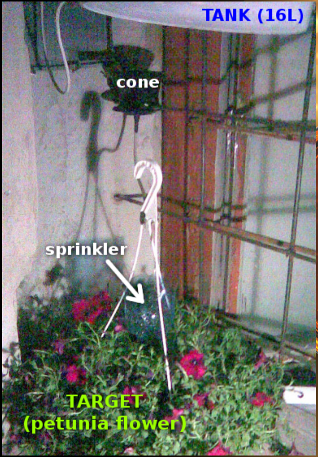
Free USB JTAG interface (FUJI)



Driving a CGA display

Hack$_2$O


Driver IC
Vcc (15V=)
Stepper motor
Arduino Diecimila
TANK (16L)
Movable part
Water to flowers
Water from tank


TANK (16L)
cone
sprinkler
TARGET (petunia flower)

Workshop with H.A.C.K.

Hungarian
Autonomous
Center for
Knowledge

Thanks for your attention!