POLITECNICO DI MILANO

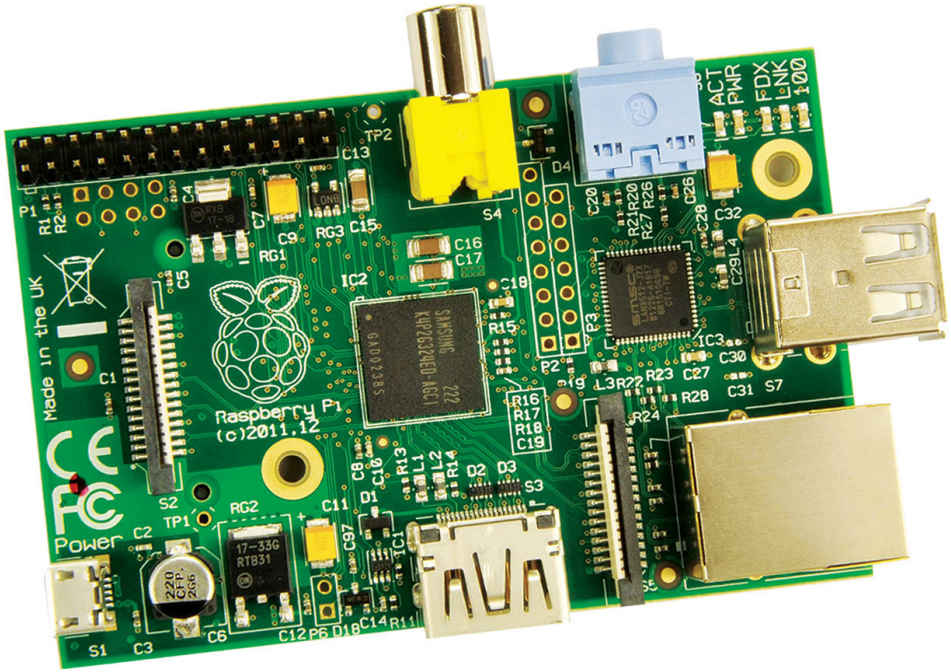
CRYPTOGRAPHIC ALGORITHMS ON BARE METAL

Emanuele De Donatis

Tutors: Prof. Gerardo Pelosi
Ing. Alessandro Barenghi

PROJECT GOALS

- Program on the “bare-metal” of a Raspberry Pi
- Build in C a working implementation of AES algorithm
- Use serial communication



TOOLS USED

HARDWARE

- Raspberry PI board
- USB to TTL RS232 cable (based on PL-2303HX chip)

PROGRAMMING LANGUAGES

- C (Raspberry side)
- Java (PC side)

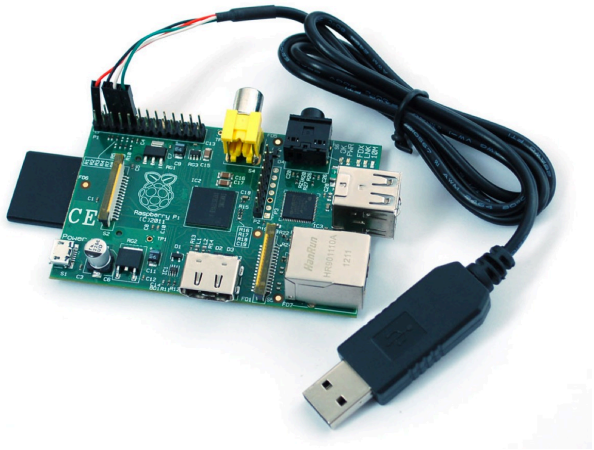
DEVELOPMENT ENVIRONMENT

- NetBeans IDE 7.4
- Processing2
- CoolTerm 1.4

SETUP

- ARM-NONE-EABI gcc cross-compiler

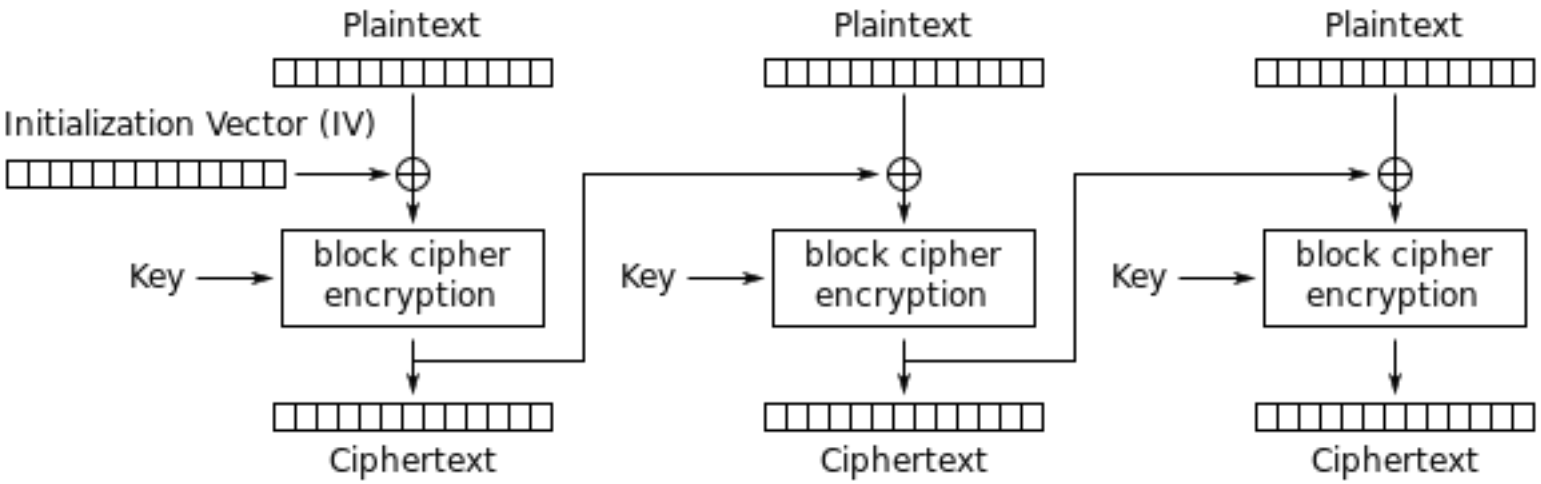
- Raspberry Pi bootstrap



- Serial communication

AES library

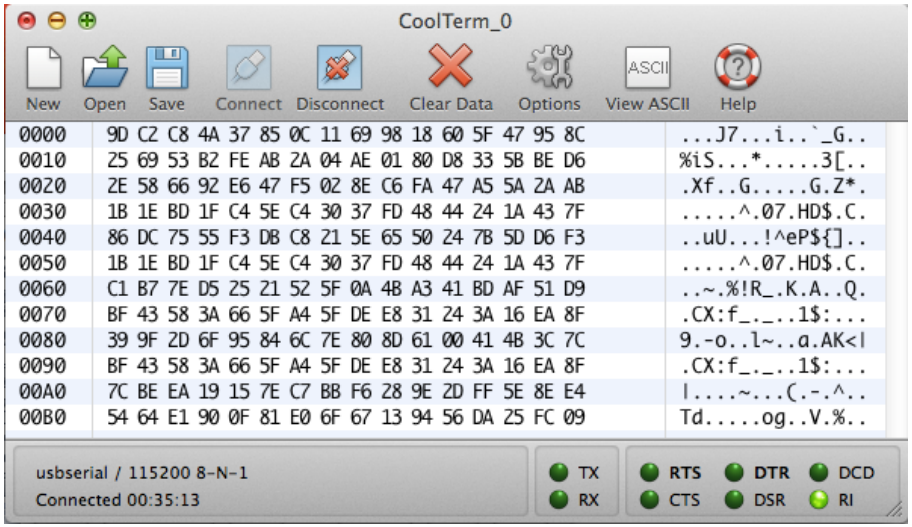
- AES CBC algorithm



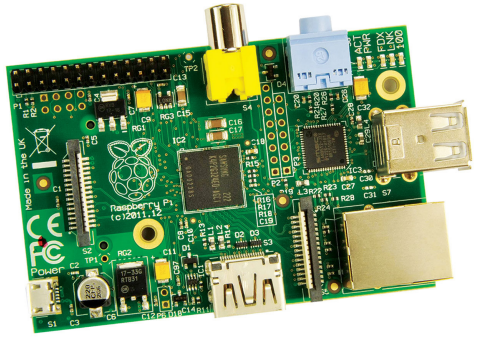
Cipher Block Chaining (CBC) mode encryption

AES Algorithm Validation Suite (AESAVS)
National Institute of Standards and Technology

- Monte Carlo Test - ECB
- Monte Carlo Test - CBC



SERIAL PROTOCOL



MODE (encryption/decryption)

KEY (128-bit)

INITIAL VECTOR

INPUT TEXT

FINISH FLAG

OUTPUT TEXT

PERFORMANCE

	200 MHz	400 MHz
Key + AES *[µs/16B]*	1325.8	929.3
AES *[µs/16B]*	122.5	95.4
AES Throughput	130 KB/s	170 KB/s

RESOURCES

- David Welch‘s repository https://github.com/dwelch67/raspberrypi (bootloader and serial communication code)
- James Snyder‘s repository https://github.com/jsnyder/arm-eabi-toolchain (ARM EABI toolchain)
- PolarSSL https://polarssl.org/ (AES library)
- AES Algorithm Validation Suite (AESAVS) http://csrc.nist.gov/