

Table of Contents

1. Introduction.....	2
1.1 Specification blocks – symbols for IoT systems.....	3
1.2 A few simple examples of IoT architectures.....	4
1.3 IoT systems – hardware and software.....	5
1.3.1 The Arduino Hardware.....	5
1.3.2 Arduino IDE.....	5
1.3.2.1 Arduino IDE installation.....	7
1.3.2.2 Arduino IDE libraries and sketches.....	7
1.3.3 C, C++ and Arduino (code) programming.....	9
1.3.3.1 I/O functions.....	9
1.3.3.2 Time functions.....	10
1.3.3.3 Mathematical functions.....	11
1.3.3.4 Interrupts.....	12
2. Connection and communication technologies.....	1
Introduction.....	1
2.1 Essential connection links and buses.....	1
12.1.1 Asynchronous serial communication.....	1
2.1.1.1 UART Receiver.....	1
2.1.1.2 UART Transmitter.....	2
22.1.2 I2C bus.....	4
2.1.2.1 I2C bus structure.....	4
2.1.2.2 I2C bus operational mode.....	4
2.1.2.3 I2C bus with Arduinos and ESPs.....	6
2.1.2.4 Arduino I2C programming (master and slave).....	6
32.1.3 SPI bus.....	9
2.1.3.1 SPI bus connection.....	9
2.1.3.2 SPI bus operation.....	10
2.1.3.3 SPI bus connections and operation on Arduino/ESP boards.....	11
2.1.3.3 Interrupts.....	12
2.2 Essential wired and wireless communication channels.....	13
12.2.1 Ethernet (IEEE 802.3).....	13
2.2.1.1 Ethernet frame structure.....	13
62.2.2 Arduino Ethernet (network) programming.....	14
2.2.2.1 Starting Ethernet.....	14
2.2.2.2 Client-server communication with UDP/IP.....	15
2.2.2.3 An example of Arduino code (Ethernet shield on SPI).....	15
2.2.2.4 Client server communication with connection to the server and data transfer (TCP/IP).....	16
2.2.2.5 Connecting to a Web Server (HTTP/TCP/IP).....	17
2.2.2.6 An example of Web client.....	17
2.2.2.7 An example of Web server, connection and serving Web pages.....	19
2.2.3 WiFi (IEEE 802.11.x).....	21
2.2.3.1 WiFi 802.11.g efficiency and channels.....	21
2.2.3.2 WiFi security.....	21
2.2.3.3 SSID and RSSI.....	22
2.2.3.4 Arduino and ESP8266 WiFi.....	22
2.2.4 RF433 – simple radio channels on 433MHz band.....	30
2.2.4.1 RF433 modules.....	30
2.2.4.2 An example of Arduino code for RF433 transmitter and receiver.....	30
2.2.4.3 An example of Arduino code for RF433 transmitter for switch control.....	32
2.2.5 HC-12–Low power radio channels on 433MHz band.....	33
2.2.5.1 An example of Arduino code for HC-12 (443MHz).....	33

2.2.5.2 HC-12 configuration.....	34
2.2.6 NRF24 – simple radio channels on 2.4GHz band.....	36
2.2.6.1 nRF24 modulation.....	36
2.2.6.2 nRF24 modules.....	36
2.2.6.3 An example of Arduino code with nRF24L01.h and RF24.h libraries.....	36
2.2.6.5 nrf24 server working with Wemos D1 shield and RH_NRF24.h library.....	37
2.2.6.5 nrf24 client working with Wemos D1 shield and RH_NRF24.h library.....	39
2.2.7 Bluetooth link on 2.4 GHz band.....	40
2.2.7.1 Bluetooth modules.....	40
2.2.7.2 An example of Arduino code with SoftwareSerial.h library.....	41
2.2.8 Long Range communication with LoRa (433 MHz, 968MHz).....	42
2.2.8.1 Introduction to LoRa technology.....	42
2.2.8.2 Packet format.....	43
2.2.8.3 Connections and pins.....	43
2.2.8.4 Interrupts.....	44
2.2.8.5 Memory.....	44
2.2.8.6 Range.....	44
2.2.8.7 An example Arduino code with Radio Head RH_RF95.h library.....	45
2.2.9 GSM and SMS messages.....	46
2.2.9.1 GSM circuits, modules and shields.....	46
2.2.9.3 Programming Arduino with GSM and SMS classes.....	47
2.2.9.4 SMS Class.....	48
2.2.9.5 An example of Arduino code with GSM.h and GSM_SMS.h libraries.....	49
2.2.9.6 Programming Arduino shield with SIM900 class.....	50

3. Sensors and controllers (actuators).....	1
3.1 Simple actuators.....	1
3.2 Simple sensors.....	1
Note concerning the interruption enabled pins.....	2
3.3 Motion detection sensors.....	3
3.3.1 PIR sensor.....	3
3.3.2 Microwave sensor – 10.525 GHz Doppler radar motion detector (RMD).....	5
3.3.2.1 An example of Arduino code operating with RMD module.....	5
3.3.3 IR sensor with transmitter.....	6
3.3.3.1 FC-51 sensor.....	7
3.3.3.2 An example of Arduino code for FC-51 sensor.....	7
3.4 Sensors (actuators) operating on I2C bus.....	8
3.4.1 An example of Arduino code for LM75 sensor.....	8
3.4.2 An example of Arduino code for HTU21D and BH1750 sensors.....	9
3.4.3 An example of Arduino code for AM2321 sensor.....	9
3.4.4 An example of Arduino code for BMP180 sensor.....	10
3.4.5 An example of Arduino code for DS3231 (Real Time Clock).....	11
3.4.6 OLED display operating on I2C bus.....	12
3.4.7 IMU (Inertia Measurement Unit) sensor.....	13
3.4.7.1 MPU 6050 sensor board.....	13
3.4.7.2 Two examples of Arduino code for MPU 6050 sensor.....	14
3.4.7.3 The execution results (fragment):.....	16
3.4.8 APDS-9960 , RGB and Gesture Sensor sensor.....	17
3.4.8.1 RGB color sensor.....	17
3.4.8.2 Gesture sensor.....	18
3.4.8.3 Ambient light interrupt.....	20
3.4.9 16-Bit ADC - 4 Channel with Programmable Gain Amplifier.....	23
3.4.10 Digital to Analog Conversion with PCF8591 module.....	24
3.4.10.1 PCF8591 YL-40 board.....	24
3.4.10.2 An example of Arduino code for PCF8591 YL-40 module.....	25
3.4.11 FM Radio Transmitter module.....	26

3.4.11.1	An example of Arduino code with FMTX library.....	26
3.4.12	FM Radio Receiver module.....	28
3.4.12.1	An example of Arduino code for TEA5767 module.....	28
3.4.13	Dot Matrix 8x8 display module - HT16K33.....	29
3.4.13.1	Displaying registered messages.....	29
3.4.13.2	Displaying the messages sent via Web page.....	31
3.4.14	Air quality sensor - CCS811.....	34
3.4.14.1	Arduino code on Wemos D1 (ESP8266) for testing CCS811 sensor.....	34
3.4.14.2	Arduino code on Wemos D1 ESP32 capture and visualization on OLED display.....	36
3.4.15	Displaying temperature from object temperature sensor – GY-906.....	37
3.5	Sensors operating on SPI bus.....	39
3.5.1	RFID sensor on 13.56 MHz.....	39
3.5.1.1	RFID 13.56 MHz ISM Frequency band.....	39
3.5.1.1	An example of Arduino code operating on RC522.....	39
3.5.2	SD card reader.....	41
3.5.2.1	The SD library.....	41
3.5.2.2	An example of Arduino code for micro SD reader.....	44
3.5.3	ArduCam (2MP) operating on I2C and SPI.....	45
3.5.3.1	ArduCam I2C bus operation.....	45
3.5.3.2	ArduCam SPI bus operation.....	46
3.5.4	MP3 Arduino shield with SD card and SPI.....	51
3.5.4.1	An example of programming MP3 shield with Arduino code.....	52
3.5.5	TFT QVGA display operating on SPI.....	53
3.5.5.1	Displaying simple messages sent form a Web client.....	54
3.5.6	TFT QVGA 2.8 inch touch display (ILI9341) operating on SPI.....	57
3.5.6.1	A simple touch screen application – the code.....	58
3.5.6.2	The use of yield() function with ESP8266.....	62
3.6	Serial – UART (tty) bus for sensors - actuators.....	63
3.6.1	Fingerprint scanner.....	63
3.6.1.1	FPM10A Arduino code – enrollment.....	64
3.6.1.2	FPM10A Arduino code – detection (uncomplete).....	66
3.6.2	R301 fingerprint scanner with Wemos D1.....	67
3.6.3	MP3 player modules.....	69
3.6.3.1	DFPlayer Mini.....	69
3.6.3.1	An example of Arduino code for DFPlayer.....	69
3.6.3.2	CATALEX serial mp3 player.....	70
3.6.4	GPS.....	74
3.6.4.1	GPS working principle.....	74
3.6.4.2	NEO6M GPS module.....	75
3.6.4.3	An example of Arduino code to test NEO6M GPS module.....	75
3.6.4.4	Using GPS module with TinyGPS++ library.....	77
3.7	Serial specific protocols on simple logic links (DIO).....	80
3.7.1	DHT11/22 sensors.....	80
3.7.1.1	An example of C code for DHT11 sensor.....	81
3.7.1.2	A simple of Arduino code for DHT11 and DHT22 sensors.....	82
3.7.1.3	Arduino code for DHT11 and DHT22 sensors and Ethernet shield.....	83
3.7.2	segment LED display – TM1637.....	85
3.7.2.1	An example of Arduino code for TM1673 4-digit display.....	86
3.7.3	Ultrasonic sensors.....	86
3.7.3.1	An example of Arduino code for HC-SR04 ultrasonic sensor.....	87
3.8	Simple sensors with analog outputs.....	88
3.8.1	Current measuring sensor.....	88
3.8.1.1	Arduino code for DC current measuring (mode 5 A).....	88
3.8.1.2	Arduino code for AC current measuring (5A mode).....	89
3.9	Sensor calibration and adaptation.....	90
3.9.1	Regression analysis and functions.....	90

3.9.2 Linear regression analysis and functions.....	90
3.9.3 Polynomial (quadratic) regression analysis and functions.....	91

4. MCU and SBC boards – GPIO pins and buses.....	1
Introduction.....	1
4.1 The micro-controllers.....	1
4.1.1 ATMega328.....	1
4.1.2 ATMega2560.....	1
4.1.3 Xtensa – ESP8266 and ESP32.....	1
4.2 Arduino Uno board.....	3
4.2.1 UART – Rx/D and Tx/D (D0 and D1).....	3
4.2.1.1 An example of Arduino code with SoftwareSerial.....	4
4.2.2 Interruption inputs – pins (D2 and D3).....	4
4.2.3 I2C bus pins.....	4
4.2.4 SPI bus pins.....	5
4.3 Arduino Nano board.....	5
4.4 Arduino Mini Pro (3.3V or 5V) board.....	6
4.5 Arduino Mega board.....	7
4.5.1 Arduino Mega pins and buses.....	7
4.5.1.2 SPI bus.....	8
4.5.1.3 I2C bus.....	8
4.6 ESP-12 (8266) based MCU boards.....	9
4.6.1 ESP-12 NodeMCU (V 0.9).....	9
4.6.1.1 UART.....	9
4.6.1.2 I2C.....	9
4.6.1.3 SPI.....	9
4.6.1.4 A simple LoRa gateway with nodeMCU and FM98/FM96.....	10
4.6.2 ESP-12 Wemos D1.....	15
4.6.3 ESP8266 in deep sleep mode.....	15
Hardware Configuration.....	16
4.6.3 Using ESP8266 (Wemos D1) with MQTT protocol.....	18
4.6.4 Using ESP8266 (Wemos D1) with MQTT protocol in deep sleep mode.....	19
4.6.4.1 The mqtt client code:.....	20
4.6.5 Using ESP8266 (Wemos D1) with WebSockets.....	21
4.6.5.1 Keeping the data in EEPROM when in deep sleep mode.....	21
4.6.6 Using ESP8266 with WiFi tools.....	23
4.6.6 Using ESP8266 (Wemos D1) with Amazon Echo.....	30
4.6.6.1 fauxmoESP with ESP8266 based Wemos board.....	31
4.6.7 Using App Inventor to build simple application with ESP8266.....	34
4.6.7.1 The design editor and blocks editor.....	35
4.6.7.2 The Arduino code for Android applicationESP8266.....	36
4.6.8 Using ESP8266 SmartConfig.....	38
4.7 SBC boards and GPIO.....	39
4.7.1 The RPI Hardware.....	39
4.7.1 The RPI Software.....	40
4.7.3 GPIO connections with wiringPi.....	40
4.7.3.1 wiringPi installation.....	40
4.7.3.2 Test wiringPi's installation.....	41
4.7.3.3 wiringPi - gpio utility.....	41
4.7.3.4 Simple logic outputs.....	42
4.7.3.5 Simple logic inputs.....	42
4.7.3.6 The RPI UART and wiringSerial.h.....	43
4.7.4 The RPI I2C: utilities and examples.....	45
4.7.4.1 Changing the Speed of the I2C bus.....	45
4.7.4.2 Testing the I2C device from the command line.....	45
4.7.4.3 Using I2C interface and wiringPiI2C.h to read RTC registers.....	46

4.7.5 The RPI SPI: utilities and examples.....	49
4.7.5.1 Using SPI with and wiringPiSPI.h to setup and read/write operations.....	49
4.7.5.2 Using SPI with and wiringPiSPI.h to program an EEPROM module (25LC010A).....	50
4.7.5.3 Using SPI with and wiringPiSPI.h to program to receive the data via an nRF24 module.....	51
4.7.6 Interfacing RPI with Arduino and/or ESP boards.....	53
4.7.6.1 Interfacing via UART (/dev/ttyS0).....	53
4.7.6.2 Interfacing via I2C bus with Arduino Pro mini 3.3V.....	55
4.7.6.3 Interfacing via SPI bus in slave mode.....	56
4.7.7 Raspberry Pi and Long Range communication with LoRa (SPI bus).....	58
4.7.7.1 RFM95/96/97/98 circuits.....	58
4.7.7.2 Chistera-Pi board.....	59
4.7.7.3 LoRA GPS HAT.....	60
4.7.7.4 A simple extension board for RFM95/98 module.....	60
4.7.7.5 Simple RFM95/98 module and breakout.....	61
4.7.7.6 Simple sender receiver code.....	62
4.7.7.7 Wemos LoRa gateway communication with RPI LoRa module.....	65
4.7.8 Raspberry Pi as a WiFi Access Point with bridge to Ethernet.....	69
4.7.9 Raspberry Pi (2/3) with Neo-6M GPS modules-actuators.....	72
4.7.9.1 Connection on UART pins.....	72
4.7.9.2 Serial console turn-off.....	72
4.7.9.3 Preparing the GPS library.....	72
4.7.9.4 Use the GPS library.....	73
4.7.9.5 Programming example.....	73
4.7.9.6 Visualizing the position.log.....	74
4.8 Communication security - encrypting and decrypting the messages.....	75
4.8.1 Encryption and decryption with XXTEA algorithm.....	75
4.8.2 Creating CRC16 checksum.....	77

5. SBC configuration and network/media programming..... 1

5.1 Network interfaces, configurations and softAP.....	1
5.1.1 RPI3 Interfaces.....	1
5.1.2 RPI Zero W Interfaces.....	2
5.1.3.1 Adding Ethernet interface.....	2
5.1.3.2 Using USB-OTG port as a virtual Ethernet interface.....	3
5.1.4 Building SoftAP.....	4
5.1.4.1 Stopping network-manager.....	4
5.1.4.2 hostapd configuration.....	4
5.1.4.3 dnsmasq configuration.....	5
5.1.4.4 Set up IPv4 forwarding.....	5
5.2 “socket” API for communication with Ethernet and WiFi.....	6
5.2.1 Working with UDP datagrams.....	6
5.2.1 “socket” address structure.....	7
5.2.2 Client-server message exchange application in datagram mode.....	8
5.2.2.1 The client side.....	8
5.2.2.2 The server side.....	8
5.2.3 Broadcasting.....	9
5.2.4 Working with TCP connections.....	10
5.2.4.1 A client/server TCP application – echo.....	11
5.2.4.2 Writing a “parallel” server.....	12
5.2.4.3 Using TCP connection to send an e-mail via SMTP service.....	13
5.2.4.5 Reading WEB page.....	15
5.2.5 Reading and writing from/to thingspeak.com server.....	17
5.2.5.1 Reading and writing directly from/to thingspeak.com with HTML page.....	17
5.2.5.2 Reading and writing from/to thingspeak.com server using curl library.....	19
5.2.6 Installing and Testing Mosquitto MQTT on the Raspberry Pi.....	22
5.2.6.1 JSON Data Payload.....	22

5.2.6.2	Installing Mosquitto.....	22
5.2.6.3	Configuring and Starting the Mosquitto Server.....	23
5.3.3.4	Testing the Mosquitto server.....	23
5.2.6.4	Using the Mosquitto RPI3 with an ESP on Wemos D1.....	23
5.3.4	Installing and Testing WebSocket server – tornado on the Raspberry Pi.....	23
5.2.7	Running the Tornado server with the Wemos D1 based client.....	25
5.3	Using Bluetooth communication with RPIs.....	26
	Introduction.....	26
8.3.1	Bluetooth transport protocols.....	26
5.3.1.1	RFCOMM.....	26
5.3.1.2	L2CAP.....	26
5.3.2	Installation.....	26
8.3.2.1	bluetoothctl.....	26
5.3.2.2	Python code on Raspberry Pi.....	27
5.3.3	Programming transmission with Bluetooth.....	28
5.3.3.1	Bluetooth stack Bluez.....	28
5.3.3.2	Bluetooth device address.....	28
5.3.3.3	RFCOMM socket address.....	28
5.3.3.5	Scanning for devices.....	29
5.3.3.5	Sending (client) and receiving messages (server) with RFCOMM.....	30
5.3.3.6	Sending (client) and receiving messages (server) with L2CAP.....	33
5.4	Installation and configuration for nRF24.....	35
5.4.1	Downloading and installing software.....	35
5.2.2	Connections and pin configuration.....	35
5.5	Using GSM modem to send/receive SMS messages.....	37
8.5.1	Basic C programming with AT commands.....	38
5.5.1.1	The SMS sender.....	38
5.5.1.2	The SMS receiver.....	39

6. Examples of simple IoT architectures.....	1
6.1 First example: a simple Arduino Nano based architecture.....	1
5.1.1 The client front-end part (sensors).....	1
6.1.2 The client back-end part (network connection).....	2
6.1.3 The UDP server on RPI/RPIZ.....	4
6.1.4 To do.....	5
6.2 Second example: A simple ESP8266 NodeMCU based architecture.....	6
6.2.1 Front-end of the server side.....	6
6.2.2 Back-end of the server side.....	7
6.2.3 The client side on RPI/RPIZ.....	9
6.2.4 To do.....	10
6.3 Third example: A simple communication system for IoT with Arduino Nano and NRF24L01 modules.....	11
5.3.1 Introduction to NRF24L01.....	11
6.3.2 NRF24 server.....	12
6.3.3 NRF24 client side with an RTC clock.....	13
6.3.4 To do.....	15
6.4 Fourth example: A simple communication system for IoT with Arduino Nano and LoRa SX1276.....	16
5.4.1 Simple experimental architecture LoRa technology.....	16
6.4.2 Server side (with selection of the clients).....	16
6.4.3 Client side with LM75 sensor.....	18
6.4.4 To do.....	20
6.5 Fifth example: A simple communication system for IoT with Arduino Nano ESP8266 and LoRa SX1276 serial modems.....	21
6.5.1 Introduction.....	21
6.5.2 GPS to LoRa gateway (client).....	21
6.5.3 LoRa to WiFi gateway (server) on ESP-12 MCU.....	24
6.5.4 WiFi receiver on SBC.....	25

6.5.5 To do.....	26
6.6 Sixth example : sending images/streams from ESP to RPI3.....	27
6.6.1 Arduino code for ESP MCU.....	27
6.6.2 The HTML page on lighttpd server implemented on RPI3.....	31
6.6.3 To do.....	32
6.7 Seventh example : sending – receiving messages on ESP32 Lora boards with OLED screen.....	33
6.7.1 The Arduino code for the sender.....	34
6.7.2 The Arduino code for the receiver:.....	37
7.1 A simple client-server system with an Arduino Nano as client and RPI3 as server.....	1
7.1.1 The client-side sub-system.....	1
7.1.2 The server side.....	3
7.1.3 To do.....	5
7.2 Simple LoRa gateways (servers) for long range communication between tablets or smart-phones (clients).....	6
7.2.1 WiFi gateway code.....	6
7.2.2 To do.....	9
7.3 A client terminal with fingerprint scanner for Internet communication with remote server.....	10
7.3.1 The enrollment of the users (scans).....	10
7.3.1.1 The enrollment program.....	10
7.3.2 The client side of the scanner.....	13
7.3.3 The server side.....	19
7.3.3 To do.....	22
7.4 RPI3 based gateway with RF24 terminals using .json messages.....	23
7.4.1 The server side.....	23
7.4.2 The client side.....	26
7.4.3 To do.....	28
7.5 An GSM (SMS) gateway with LoRa terminals.....	29
7.5.1 The gateway code.....	30
7.5.2 The terminal Arduino code.....	33
7.5.3 To do.....	35
7.6 1 An RPI3 gateway to thingspeak.com for ESP terminals.....	36
7.6.1 Arduino terminal code with ThingSpeak.h library.....	36
7.6.2 Arduino terminal code without ThingSpeak.h library.....	38
7.6.3 To do.....	40
7.7 Building a LoRa-Internet gateway with configurable terminals.....	41
7.7.1 The gateway code on Wemos.....	41
7.7.2 The code running on the RPI3.....	47
7.8 Concurrent LoRa gateway and client with data encryption on Wemos D1 boards.....	51
7.8.1 The operation of the “concurrent” LoRa Gateway.....	51
7.8.1.1 The code of concurrent gateway with encryption :.....	52
7.8.2 The operation of the LoRa client.....	57
7.8.2.1 The “concurrent” client code.....	57

8 Building LoRaTS architecture.....	1
8.1 LoRa modulation.....	1
8.1.1 Principle of spread spectrum.....	2
8.1.2 LoRa Spread Spectrum.....	3
8.2 LoRa RFM95 / 96/97/98 Modems and expansion boards.....	4
8.3 Programming for LoRa RFM95/96/97/98 modems.....	5
8.4 Basic architecture with LoRa modems and Wemos D1 cards.....	6
8.4.1 rf95_client program running on Wemos D1.....	6
8.4.2 rf95_server program running on Wemos D1.....	7
8.5 Basic architecture with LoRa modems on RPIZ (W) and Wemos D1.....	9
8.5.1 C code of LoRa client on RPIZ(W) :.....	9
8.5.2 Code Arduino code on Wemos D1.....	11
8.6 LoRaTS frames between terminals and LoRaTS gateway.....	13
8.6.1 LoRaTS protocol.....	14

8.6.2 Code C code for the gateway RPIZ(W).....	15
8.7 LoRaTS terminal in low power consumption mode (<1mA).....	23
Attention :.....	23
8.7.1 Arduino code for Wemos d1 (ESP8266) and Wemos ESP32.....	23
8.8 Storing and reading data in EEPROM memory.....	29
8.8.1 Writing into EEPROM.....	29
8.8.2 Reading data fromEEPROM.....	30