SMART GARDEN PROJECT

Pervasive System 2016 – DIAG "La Sapienza"

Stefano Coratti & Massimo Perri

INSTALLATION INSTRUCTIONS

The replication of the system is a complex task that requires at least average electronic skills and , of course, a bit of familiarity with software and Arduino related hardware e.g. board, sensors , modules and shields.

Get the material, both hardware and software, in particular 3x ATmega328P and related components to make an arduino standalone @16Mhz (e.g. XTAL 16MHz, 2x 22pF caps, 10Kohm pullup resistor and the like) for this one can use as reference this web page: https://www.arduino.cc/en/Main/Standalone ; then also an Arduino Mega 2560 is required (mega 1280 can be used as well, and so also other similar boards with at least 3 hardware serials uart; software serials are not suitable for the project). Other support material is required e.g. an USB-TTL converter is highly suggested to have the diagnostic & debug serial communication with the PC; USB-RS232+MAX232 are also cheap and valid alternatives to USB-TTL converters. USBASP programmers can be built using e.g. ATmega8 or ATmega88 plus some other components or bought online or in electronic shops. Sensors needed are included in diagrams and schematics and are: one BMP180 (Bosh) , one DHT11, one 10K photoresistor for weather data; 3x Arduino soil hygrometer (HL-69 or the like) + 3x moisture sensor (HL-69, FC-28 or the like) + 3x 2-wires Dupont lines; 3x small 220v or 12v submersible water pump (e.g. 30l/h flow); as software Arduino IDE , USBASP driver, a terminal program e.g. Putty and libraries (e.g. Blynk and ESP8266) are required.

Build the sensor board and connect it to the Mega: use BMP180, DHT11, photoresistor, ESP8266 and nRF24L01 to build the board; use extra components for voltage level shifting and 3v3 volt line regulation. Program Arduino Mega with the master sketch and connect the sensor board to it.

Build the slave boards (plant computers) and connect to the pumps: make the slaves (you may use 3, or less or more that 3 changing the code accordingly) using ATmega328P and other components required e.g. relay and nRF24L01 module.

Program fuse bits using the ICSP connector, connecting the USBASP programmer and using avrdude or ExtremeAVRBurner: correct fuses bits settings are: low_fuses=0xff high_fuses=0xde extended_fuses=0x05

Program each slave using the ICSP connector and the USBASP programmer & Arduino IDE: to do that add a standalone configuration (as provided in point 4) to your Arduino IDE in the 'boards.txt' file under the installation directory. When all is ready connect the single boards to their water pump to test and use the system.

In '<your-arduino-ide-install-dir>\hardware\arduino\avr\ boards.txt' add the following configuration for the Arduino standalone:

##

```
stand_alone_atmega328.name=ATmega328P (stand-alone @16MHz)

stand_alone_atmega328.build.board=ATmega328P_stand-alone_16MHz

stand_alone_atmega328.upload.using=usbasp

stand_alone_atmega328.upload.protocol=usbasp

stand_alone_atmega328.upload.tool=usbasp

stand_alone_atmega328.upload.params.quiet=true

stand_alone_atmega328.upload.protocol=stk500

stand_alone_atmega328.upload.maximum_size=32768

stand_alone_atmega328.upload.speed=57600

stand_alone_atmega328.bootloader.low_fuses=0xFF

stand_alone_atmega328.bootloader.high_fuses=0xDA

stand_alone_atmega328.bootloader.extended_fuses=0x05

stand_alone_atmega328.bootloader.path=atmega

stand_alone_atmega328.bootloader.unlock_bits=0x3F

stand_alone_atmega328.bootloader.lock_bits=0x0F

stand_alone_atmega328.build.mcu=atmega328p

stand_alone_atmega328.build.f_cpu=16000000L

stand_alone_atmega328.build.core=arduino:arduino

stand_alone_atmega328.build.variant=arduino:standard

##############################################################
```

Use a suitable water tank, put the pumps inside it and connect their output hoses to the plants: when ready for test/run fill the tank with water submersing the pumps

Switch on the Mega and slaves (slaves must be provided with external power source e.g. 5V or 12v powersupply with 5V regulator on board), connect pump power system (e.g. 12V battery or to a wall socket if pumps runs on poweline)

Connect USB cable to the Mega – if debug is required connect the USB-TTL (or RS232+MAX232) converter to another USB port

Open Terminal on Mega com port (es. COM10 if that is the port assigned to Mega): if debug needed launch the terminal on the com port of the usb-ttl converter too; then launch the startup script, this will trigger an automatic reset on the master, then initialization will follow, the master will init the WiFi module, then will get the timestamp from Blynk and proceed with the first broadcast sync message to the slaves. If debug terminal is active you could see messages from the master announcing the various phases transitions

Tweak parameters to suit your needs: switch on or off the email alerts, switch on or off the in cloud logging on ThingSpeak, change the plant computer configuration to be sent, etc. all this in the master sketch; also change idle delay to fix a maximum repetition rate e.g. 30 minutes , 60 minutes or the like as you need