

Una propuesta teórica sobre agricultura de precisión en el ámbito del Internet of Things



Manuel Guerrero Moñús

Máster en Ingeniería Informática; Arquitectura del Nodo IoT

Índice

1 Introducción.....	2
2 Objetivo del proyecto.....	2
3 Descripción del proyecto.....	2
4 Elementos hardware necesarios.....	3
4.1 Placas de desarrollo.....	3
Placas consideradas.....	3
Placas escogidas para formar el nodo IoT.....	4
Placa escogida como nodo broker.....	4
4.2 Sensores, actuadores y otros dispositivos.....	4
Selección realizada.....	4
Justificación de la elección.....	4
5 Elementos software necesarios.....	5
5.1 Sistema operativo.....	5
5.2 Entorno de desarrollo.....	6
5.3 Influxdb Tick Stack para el análisis de datos en el nodo broker.....	6
6 Comunicación.....	7
6.1 Comunicación entre las placas que conforman el nodo IoT.....	7
6.2 Comunicación entre los nodos IoT.....	7
6.3 Comunicación con el nodo broker y de procesamiento de datos.....	8
7 Energía.....	8
7.1 Consumo energético.....	8
7.2 Mecanismo de alimentación.....	8
8 Coste del proyecto.....	9
9 Referencias.....	9
Enlaces a los elementos hardware.....	9

1 Introducción

Desde finales los años 90 hemos visto como a medida que ha ido transcurrido el tiempo se ha popularizado el paradigma IoT. Esta gran aceptación por parte de la sociedad del siglo XXI viene dada por muchas cuestiones inherentes a ella, principalmente: el deseo de prevalecer conectado para intercambiar u obtener información en tiempo real, la necesidad apostar por un progreso sostenible y económicamente asequible, y el creciente compromiso con los medios urbanos y rurales en lo que respecta al consumo, la producción y el reparto de recursos de toda clase.

Este campo que nace de la unión del hardware de bajo coste, la computación de alto rendimiento, el análisis y la visualización de datos, las comunicaciones, los sistemas empujados y distribuidos, y la inteligencia artificial, ha dado como resultado la aparición de una gran cantidad de proyectos que resuelven o mejoran las respuestas a los problemas de la sociedad actual.

En el caso de este trabajo se pondrá el foco del uso del IoT en el medio rural, concretamente en un proyecto de agricultura de precisión.

2 Objetivo del proyecto

Este trabajo consiste en la elaboración de un marco teórico para la construcción de una aplicación IoT que sirva para automatizar el durísimo trabajo de los jornaleros de una plantación frutícola, mejorando así: la calidad de vida de los agricultores, la calidad de sus productos, la ecología, sus beneficios, eficiencia, el control del estado de los cultivos y de la producción, la gestión de recursos e insumos de toda clase, etc.

3 Descripción del proyecto

Se va a describir el proyecto en base a sus necesidades, por lo que vamos a suponer que después de una conversación con unos presuntos clientes, el resumen a partir del cual pueden extraerse a groso modo los requerimientos del proyecto ingenieril, es el siguiente:

- Disparar de alertas ante ciertas situaciones, por ejemplo ante la aparición de plagas animales o vegetales, pero también ante situaciones climatológicas adversas como lluvias, heladas, etc.
- Obtención de los mapas de datos de los terrenos para obtener información que sirva para ayudar en la toma de decisiones.
- Generación de informes en los que se pueda visualizar la evolución de los cultivos.
- Elaborar predicciones basadas en datos sobre la evolución cosecha y la producción que puede llegar a alcanzarse.
- Introducción de medidores y sensores para la obtención de datos en tiempo real.
- Transformar el sistema de riego actual en un sistema inteligente.

4 Elementos hardware necesarios

4.1 Placas de desarrollo

Placas consideradas

En esta sección se ha realizado un estudio de mercado de placas para el desarrollo de proyectos en el ámbito del paradigma IoT, prestándose una atención especial al precio de estas para poder lograr abaratar los costes del nodo, aunque también se tienen en cuenta otros factores como: sus interfaces de comunicación, GPIO (pines analógicos, digitales y bits de los conversores), cantidad de memoria disponible, frecuencia del procesador, número de núcleos de este, unidades y módulos adicionales que incorpora el microcontrolador y, finalmente, el consumo eléctrico.

La siguiente tabla muestra las placas que se ha considerado que tienen las mejores prestaciones, se encuentran ordenadas por su precio, de más baratas a más caras:

Board	Price €	Interfaces	GPIO	RAM	Flash	Processor	Clock	Cores	Units and Modules	Power Consumption
Raspberry Pi Pico	4,20	SPI I2C UART	3A ; 26D ADC 12 b DAC NO	264 kb	2 mb	Cortex M0+ (ARM)	133 MHz	2	Temperature sensor Math libs	25 mA
ESP32 DevKitC 32E	10,00	UART I2C SPI SDIO	10A ; 10D ADC 12 b DAC 10 b	520 kb	4 mb	Xtensa LX6 (Tensilica)	240 MHz	2	Wifi Bluetooth Hall Sensor Math unit	50 mA
Omega2	10,60	MicroSD UART I2C SPI USB Ethernet	18D ADC NO DAC NO	128 mb	32 mb	MIPS24KEc (MIPS)	580 MHz	1	Wifi	180 mA
Teensy-LC	12,13	SPI I2C UART USB	10A ; 10D ADC 12 b DAC 12 b 10 Touch	8 kb	62 kb	Cortex M0+ (ARM)	48 MHz	1	NO	12 mA
Arduino Nano 33 IoT	15,88	UART SPI I2C	8A ; 10D ADC 12 b DAC 10 b	32 kb	256 kb	Cortex M0 (ARM)	48 MHz	1	Wifi Bluetooth Acelerometer	19 mA
Raspberry Pi Zero 2 W	16,50	MicroSD I2C USB UART Ethernet	30D ADC NO DAC NO	512 mb	NO (SD)	Cortex A53 (ARM)	1 GHz	4	Wifi Bluetooth	120 mA
Raspberry Pi 4 Model B	61,50	MicroSD I2C USB UART Ethernet	30D ADC NO DAC NO	4 gb	NO (SD)	Cortex A72 (ARM)	1,5 GHz	4	Wifi Bluetooth	2,5 A

NOTAS:

- La placa de desarrollo ESP32-DevKitC-32E está en la tabla como una placa de referencia, y no se considerará para montar un nodo IoT. La intención de esta decisión es aumentar la variabilidad y originalidad del proyecto.
- Las especificaciones de las placas han sido extraídas de las páginas sus fabricantes. Los precios han sido obtenidos de páginas de proveedores como: Postscapes, Arrow, Adafruit, Allaboutcircuits, SparkFun, Digikey, Mouser, Bosch, Arduino, Raspberry Pi, Kubii, Brico Geek y Electrónica Embajadores. No se han tenido en cuenta los precios o las ofertas de las plataformas de comercio electrónico: Amazon, eBay o AliExpress.

Placas escogidas para formar el nodo IoT

- Raspberry Pi Pico: una placa barata, de bajo consumo, dual core, con capacidad de memoria suficiente, un procesador con la frecuencia de trabajo necesaria, e interfaces para poder comunicarse con otras placas, sin embargo, no posee conversores de analógico a digital, por lo que habrá que comprar alguno que realice la conversión de los sensores ambientales.
- Arduino Nano 33 IoT: barata, de bajo consumo, single core, con la capacidad de memoria y procesamiento necesaria, interfaces para la comunicación con otras placas, módulos Wi-Fi y Bluetooth, y conversores ADC y DAC con el suficiente número de bits para codificar y decodificar señales analógicas.

Placa escogida como nodo broker y de procesamiento

Raspberry 4 Model B: un computador barato y de bajo consumo, quad core, con una frecuencia de trabajo elevada, cuenta con una buena memoria RAM de varios gigabytes, gran cantidad de espacio en la tarjeta Micro-SD, y un módulo para poder comunicarse por Wi-Fi o Bluetooth.

4.2 Sensores, actuadores y otros dispositivos

Selección realizada

La siguiente tabla describe los sensores, actuadores y otros componentes hardware que actúan como complemento del nodo IoT. Se añade además otra información adicional, como su precio, la cantidad de pines digitales y analógicos que necesitan emplear del nodo, y la cantidad de unidades de cada uno que empleará cada nodo.

Name	Description	Price €	Analog Pins Needed	Digital Pins Needed	Uds per IoT node
101020614	Sensor para medir la humedad del suelo.	4,43	1	0	1
BME-280	Medidor de temperatura, humedad, presión atmosférica y altitud.	13,30	2	0	1
BH1750	Sensor de luz ambiental.	4,00	2	0	1
STM32	Sensor para la lluvia.	5,90	1	1	1
28BYJ-48	Motor rotatorio de movimientos discretos.	2,95	0	0	1
ULN2003	Driver Hardware para el motor 28BYJ-48.		0	4	1
OV7675	Cámara de fotos o vídeo.	9,95	6	7	1
MCP3208	Conversor ADC; 8 entradas y 12 bits de precisión.	4,01	0	4	1

Justificación de la elección

- Sensor de humedad capacitivo 101020614: he escogido este sensor ya que ofrece mejores lecturas de la humedad que los sensores de tipo resistivo, los cuales poseen un problema conocido, la corrosión, que se debe a la exposición al aire de sus dos patas conductoras.
- BH1750: este luxómetro se impone en las comparativas de la web a otros chips, en ellas demuestra que es el modelo que menos varía en sus mediciones con respecto a las marcadas por otros luxómetros profesionales. Sensores a los que supera: SI1145, TSL2561, VL6180X y AP3216.

- BME-280: en lo que respecta a la temperatura, este sensor ha sido evaluado frente a los sensores BMP180, DHT22, LM35 y DS18B20. En los estudios realizados, demostró ser el sensor con las lecturas más estables.
- STM32: francamente, los sensores MH-RD, FC-37, YL-83 y el STM32, son prácticamente iguales en lo que respecta a su rendimiento, y por ello no hay un criterio destacable para su elección.
- 28BYJ-48 y ULN2003: la principal razón por la que seleccioné este motor paso a paso es por su popularidad, y por que este tipo de motores son más recomendados que otros como los servomotores o los motores de corriente continua para utilizar una cámara con ellos, esto se debe a que: son más silenciosos, fáciles de controlar, realizan movimientos más precisos, controlan mejor la velocidad de rotación, etc.
- OV7675: he escogido esta cámara debido a su uso en proyectos de Tiny Machine Learning. La cámara es ideal para utilizarla con redes las neuronales profundas de Tensor Flow Lite Micro, que están pensadas para su uso en microcontroladores.
- MCP3208: escogí este conversor de analógico a digital porque permite disponer de varias entradas digitales, hasta 8, y a que la cantidad de bits que emplea para realizar la conversión a digital, 12 bits, es considerable.

5 Elementos software necesarios

5.1 Sistema operativo

Como las placas escogidas poseen un procesador ARM, he decido utilizar un sistema operativo de tiempo real de dicho fabricante, concretamente Mbed OS, que está pensado para ser usado en dispositivos IoT. Sus principales características son:

- Sistema operativo de código abierto; licencia de software libre: Apache License 2.0.
- Incluye soporte para microcontroladores, dispositivos y componentes basados en Cortex-M.
- Pone a disposición del programador: hilos, semáforos, colas, cerrojos, mails, eventos, colas de eventos y temporizadores.
- Posee una amplia comunidad de usuarios.
- Existen muchos ejemplos disponibles para aprender a utilizarlo.
- Cuenta con SSL y TLS para la realización de comunicaciones seguras.
- Dispone de una gran cantidad de opciones para la conectividad: Bluetooth, Wi-Fi, Ethernet, Cellular, NFC, RFID, LoRa LPWAN y 6LoWPAN Sub-GHz Mesh.
- Da soporte a las siguientes interfaces de comunicación: GPIO, I2C, SPI y UART.
- Se integra bien con las plataformas de cómputo en la nube: Amazon Web Services, Google Cloud Platform y Microsoft Azure.

Lo qo, las placas en si deben ser mbed. No vale con que sea cortex

5.2 Entorno de desarrollo

El IDE para desarrollar aplicaciones con el RTOS Mbed es Mbed Studio, cuyas características son:

- Licencia de software libre: Apache License 2.0.
- Entorno de desarrollo multiplataforma.
- Permite escribir aplicaciones en C/C++ valiéndose del autocompletado de código.
- Preparado para la organización y depuración de proyectos.
- Incorpora un mecanismo para la detección automática de la placa de desarrollo.
- Emplea sistemas de control de versiones para favorecer el desarrollo de los proyectos.
- Puede realizarse una instalación local del IDE o utilizar la versión web, esta última no requiere la instalación de software.
- Hay disponible buena documentación y vídeos para aprender a manejar el IDE.

5.3 Influxdb Tick Stack para el análisis de datos en el nodo broker

Influxdb es una base de datos NoSQL diseñada para trabajar con series temporales, esta es capaz de realizar la ingesta de datos de un entorno Big Data en tiempo real. La pila tecnológica que rodea a la BDD es:

- Telegraf: el software de Influxdata que se ocupa de recoger múltiples datos de entrada de una fuente y luego enviarlos a otro destino, por ejemplo: una bbdd relacional o NoSQL.
 - Posee plugins de procesamiento para realizar algún tratamiento de las métricas que pasan, por ejemplo: realizar algún cálculo, añadir etiquetas, filtrar por algún campo, etc.
 - También posee plugins para realizar funciones de agregación (máximos, mínimos, medias, cuantiles, desviación estándar, etc), para esta tarea se fija un marco de tiempo, y pasado este se vuelve a calcular otra vez.
- InfluxDB: la BDD que recibirá los datos de Telegraf.
 - InfluxDB es una BDD NoSQL de alto rendimiento creada específicamente para trabajar con series temporales. Permite una ingesta de datos de alto rendimiento, la consulta en tiempo real y la compresión rápida de la información.
 - Emplea un lenguaje de consulta similar al SQL e incluye mecanismos como: consultas continuas y políticas de retención de datos.
- Kapacitor: el software para procesar la información.
 - Kapacitor es un framework de procesamiento de datos que facilita la creación de alertas, la ejecución de tareas ETL y la detección de anomalías.

- Puede procesar los datos según salen de Telegraf, aunque también puede procesarlos una vez son insertados en InfluxDB.
- Permite hacer uso de un lenguaje de scripting para definir transformaciones sobre los puntos de datos, así como definir condiciones booleanas que actúan a modo de filtro.
- Puede integrarse con bibliotecas de aprendizaje automático, motores de concordancia de patrones (patter matching), motores de reglas y otros. Kapacitor trata estos plugins como Funciones Definidas por el Usuario que ceden la ejecución del programa a una función arbitraria y luego reanudan el procesamiento a la salida de esta.
- Kapacitor utiliza un Lenguaje llamado TICKscript para definir tareas que implican la extracción, transformación y carga de datos (ETL), esto permite hacer un seguimiento de los datos, la detección de eventos y crear alertas.
- Chronograf: el visualizador de datos.
 - Chronograf permite visualizar los datos almacenados en InfluxDB, construir consultas, alertas y cuadros de mando para visualizar en tiempo real.
 - Añade un servicio de autenticación y la gestión de roles de usuario para el acceso a recursos (cuadros de mando, conexiones a InfluxDB y Kapacitor, etc).
 - Chronograf también actúa como la GUI de Kapacitor, un motor de procesamiento de datos para procesar tanto datos en streaming como almacenados en InfluxDB.
 - Permite crear alertas con una GUI y ver el historial de estas.
 - Ofrece un editor de TICKscript.

6 Comunicación

6.1 Comunicación entre las placas que conforman el nodo IoT

La placa Raspberry Pi Pico empleará el ADC MCP3208 para poder leer los valores de los sensores mediambientales 101020614, BME-280, BH1750 y STM32. Una vez recolectados los datos cada 500 milisegundos, se los pasará a la placa Arduino Nano 33 IoT a través del bus I2C, un bus para la comunicación en serie, de carácter síncrono. He escogido este frente a la UART por ser más rápido.

6.2 Comunicación entre los nodos IoT

Cada placa Arduino Nano 33 IoT:

- Averiguará cada 5 minutos mediante técnicas Tiny Machine Learning, la cámara OV7675, el motor rotatorio de movimientos discretos 28BYJ-48 y su driver hardware ULN2003, si la plantación está siendo invadida por alguna especie animal o vegetal que pueda ser peligrosa. La placa habrá de enviar tramas Wi-Fi para informar de este suceso, solo si tiene lugar.
- Recibirá cada 500 milisegundos la información de la placa Raspberry Pi Pico a través del bus I2C, una vez que es recibida, se enviarán tramas Wi-Fi con estos datos para su posterior almacenamiento y estudio.

¿por qué usar las 2 placas? ¿No vale 1 sola?
— ¿cada 500ms? No parece necesario

- Ejecutará un algoritmo de cómputo distribuido para comunicarse con otras placas del mismo tipo en otros nodos IoT de la plantación a través del envío de tramas Wi-Fi, que contendrán la información de los sensores, y también de la vigilancia si procede.

El módulo Wi-Fi NINA-W102 proporciona una conexión inalámbrica de 2,4 GHz y soporta varios estándares, concretamente 802.11 b/g/n, por lo que nos ofrece la posibilidad de conectarnos a cualquier nodo que esté en un radio de 14 – 18 metros.

6.3 Comunicación con el nodo broker y de procesamiento de datos

La comunicación entre los nodos IoT tiene lugar en una dirección concreta dentro de la red que conforman, pues en ella existen, en alguno de sus confines, unos pocos nodos que actúan a modo de cabecillas del clúster. Estos serán los que se comuniquen con el computador Raspberry Pi 4, el cual tendrá instalado el broker MQTT Mosquitto y el Tick Stack de Influxdb.

La misión del nodo broker es, a grandes rasgos:

- Realizar la ingesta y el almacenamiento de la información generada por los nodos.
- Analizar los datos de las series temporales para tomar decisiones y realizar predicciones del crecimiento de la plantación y la producción.
- Ofrecer informes que permitan visualizar el resultado del procesamiento de datos.
- Crear alarmas que hagan saltar las bombas de riego para regar el campo en zonas concretas cuando se detecte sequía, comunicar a los agricultores que la plantación está siendo invadida por alguna especie animal o vegetal no deseada, etc.

7 Energía

7.1 Estimación del consumo eléctrico

La placa Raspberry Pi Pico consume 25 mA, y la Arduino Nano 33 IoT, 19 mA. El objetivo es que funcionen las 24h y tengan una reserva para unos 3 días más. Por lo que se necesitaría una batería de unos 4.400mAh, ya que $(4.400mAh \div (19mA + 25mA)) \div 24h = 4,1$ días

→ con wifi, +

7.2 Mecanismo de alimentación

→ procediendo video,
mucho mas

Para alimentar los nodos IoT utilizaremos los siguientes elementos hardware:

Name	Description	Price €	Uds per IoT node
Solar Lithium Ion/Polymer charger v2 (bq24074)	Cargador para usar con baterías de litio.	19,73	1
Large 6V 3.5W Solar panel	Panel solar de 6 voltios de salida.	39,91	1
Lithium Ion Battery Pack - 3.7V 4400mAh	Batería de litio.	17,65	1

Por un lado, la placa Arduino Nano 33 IoT recibirá la alimentación directamente de la BQ24074, mientras que la Raspberry Pi Pico recibirá su tensión de entrada de la Arduino Nano 33 IoT. Por otra parte, la Raspberry Pi 4 (de 2,5 Amperios) debe ir conectada a la red eléctrica, puesto que es importante que su suministro eléctrico sea continuo y no dependa de la climatología.

8 Coste del proyecto

El proyecto tiene una parte de coste fijo y una parte de coste variable:

- Coste fijo (CF): independientemente del número de nodos IoT.
 - 61,50 € de la Raspberry Pi 4 como nodo broker y de procesamiento de datos.
 - 36,99 € por una tarjeta Micro-SD de 256 GB que actúa como disco duro del broker.
- Coste variable (CV): según el número de nodos IoT.
 - 121,83 € por el coste de los sensores, actuadores, paneles solares y otro HW.
 - 20,24 € por el coste de las placas Raspberry Pi Pico y Arduino Nano 33 IoT.

El coste final del proyecto viene determinado por la siguiente ecuación, donde 1,3 es el factor para obtener un margen de beneficios: $1,3 * (CF + CV * N^{\circ} \text{Nodos})$

Suponiendo que tengamos un terreno que mida la mitad de un campo de fútbol posesional, más o menos unos 2.025 m², que vamos a utilizar la mitad de la superficie para cultivar y que cada árbol frutal necesita un área propia de 5 m² para poder crecer sin que lo perjudiquen otros árboles frutales, necesitamos un total de 202 nodos IoT en la plantación.

El precio total del proyecto sería: $1,3 * ((98,49 \text{ €}) + (142,07 \text{ €} * 202 \text{ nodos})) = 37.435,62$

9 Referencias

Enlaces a los elementos hardware

- Placas que conforman el nodo IoT:
 - Raspberry Pi Pico: <https://www.kubii.es/raspberry-pi-3-2-b/3205-raspberry-pi-pico-0728886755172.html?src=raspberrypi>
 - Arduino Nano 33 IoT: <https://store.arduino.cc/products/arduino-nano-33-iot?selectedStore=eu>
- Placa que actúa como nodo broker y para el procesamiento de datos:
 - Raspberry Pi 4: <https://www.kubii.es/raspberry-pi-3-2-b/2772-nuevo-raspberry-pi-4-modelo-b-4gb-0765756931182.html>
- Sensores, actuadores y otros dispositivos utilizados por los nodos:
 - Sensor de temperatura, humedad y presión BME280: <https://www.adafruit.com/product/2652>

- Medidor de la humedad del suelo 101020614101020614:
<https://www.mouser.es/ProductDetail/Seeed-Studio/101020614?qs=u16ybLDytRZLXly%252BMAAdDNQ%3D%3D>
- Sensor de luz BH1750: <https://www.adafruit.com/product/4681>
- Sensor de lluvia STM32:
<https://www.electronicaembajadores.com/es/Productos/Detalle/SSHULL1/sensores/sensores-de-humedad-agua/stm32-sensor-de-lluvia-compatible-con-arduino-y-raspberry-pi>
- Motor rotatorio de movimientos discretos 28BYJ-48 y su driver hardware ULN2003:
https://tienda.bricogeek.com/motores-paso-a-paso/969-motor-paso-a-paso-28byj-48-5v-con-driver-uln2003.html?search_query=28BYJ-48&results=1
- Módulo de cámara OV7675: https://www.kubii.es/recherche?controller=search&orderby=position&orderway=desc&search_query=OV7675&submit_search=
- Conversor ADC MCP3208: <https://www.digikey.es/es/products/detail/microchip-technology/MCP3208-CI-P/305928>
- Componentes para el montaje de los circuitos de alimentación:
 - Solar Lithium Ion/Polymer charger – v2 (BQ24074):
<https://www.adafruit.com/product/390>
 - Large 6V 3.5W Solar panel: <https://www.adafruit.com/product/500>
 - Lithium Ion Battery Pack - 3.7V 4400mAh: <https://www.adafruit.com/product/354>

Muy buen trabajo ; muy exhaustivo
 No me queda claro como tenía el posicionamiento
 de los nodos : parte debería estar en el suelo (humedad)
 y parte en altura (cámara). ¿tenías algo pensado?