



Il progetto **OpenDomotica** nasce dopo una prima analisi delle soluzioni domotiche già esistenti e consolidate sul mercato; i **maggiori problemi** emersi sono stati **l'elevato prezzo dei materiali e la scarsa espandibilità** verso altri componenti non direttamente supportati, da qui si è scelto di **cercare innanzitutto una soluzione** a questi due aspetti.

La scelta dell'hardware domotico è ricaduta sulla **piattaforma Arduino**, sia per il suo **basso costo**, sia per la **semplicità nella programmazione** e sia per la **fiorente comunità** che vi ruota intorno (senza trascurare il fatto che si tratta di un **progetto tutto Italiano**), mentre per prese, punti luce e pulsanti a muro si può mantenere quanto già presente in loco, con **piccoli adattamenti** nel cablaggio, in quanto possono essere utilizzati modelli provenienti da **qualsiasi serie commerciale** per impianti domestici, senza la necessità di un nuovo acquisto di materiale ad hoc.

Un'installazione **OpenDomotica** prevede l'impiego di **uno o più nodi domotici**, a seconda delle dimensioni e della disposizione dell'impianto elettrico eventualmente esistente, e di un **pc-server**, collegati tra loro tramite una **comune rete Ethernet**. Ai nodi domotici vengono collegati i pulsanti dell'impianto, che saranno l'interfaccia principale utilizzata in casa, e una serie di relé che comandano le utenze (prese o luci).

Uno dei punti chiave del progetto è la **decentralizzazione della logica di gestione**, per evitare che il malfunzionamento di una parte del sistema infici il funzionamento del resto. Per ottenere questo risultato, è necessario che, in ogni nodo domotico, venga programmata una logica di base locale che gli consente di **gestire autonomamente** ciò che ad esso è direttamente collegato, anche qualora il nodo venga isolato dagli altri o il server non sia raggiungibile, eliminando, di fatto, il single point of failure tipico dei sistemi domotici con logica centralizzata all'interno del pannello di controllo.

Gli sviluppi futuri prevedono **l'integrazione di sensori ambientali** (temperatura, pressione, umidità, pioggia...) che aiuteranno a formulare delle logiche decisionali per il **comfort interno delle persone** e per il **risparmio energetico**.

## Schema a blocchi di un nodo domotico

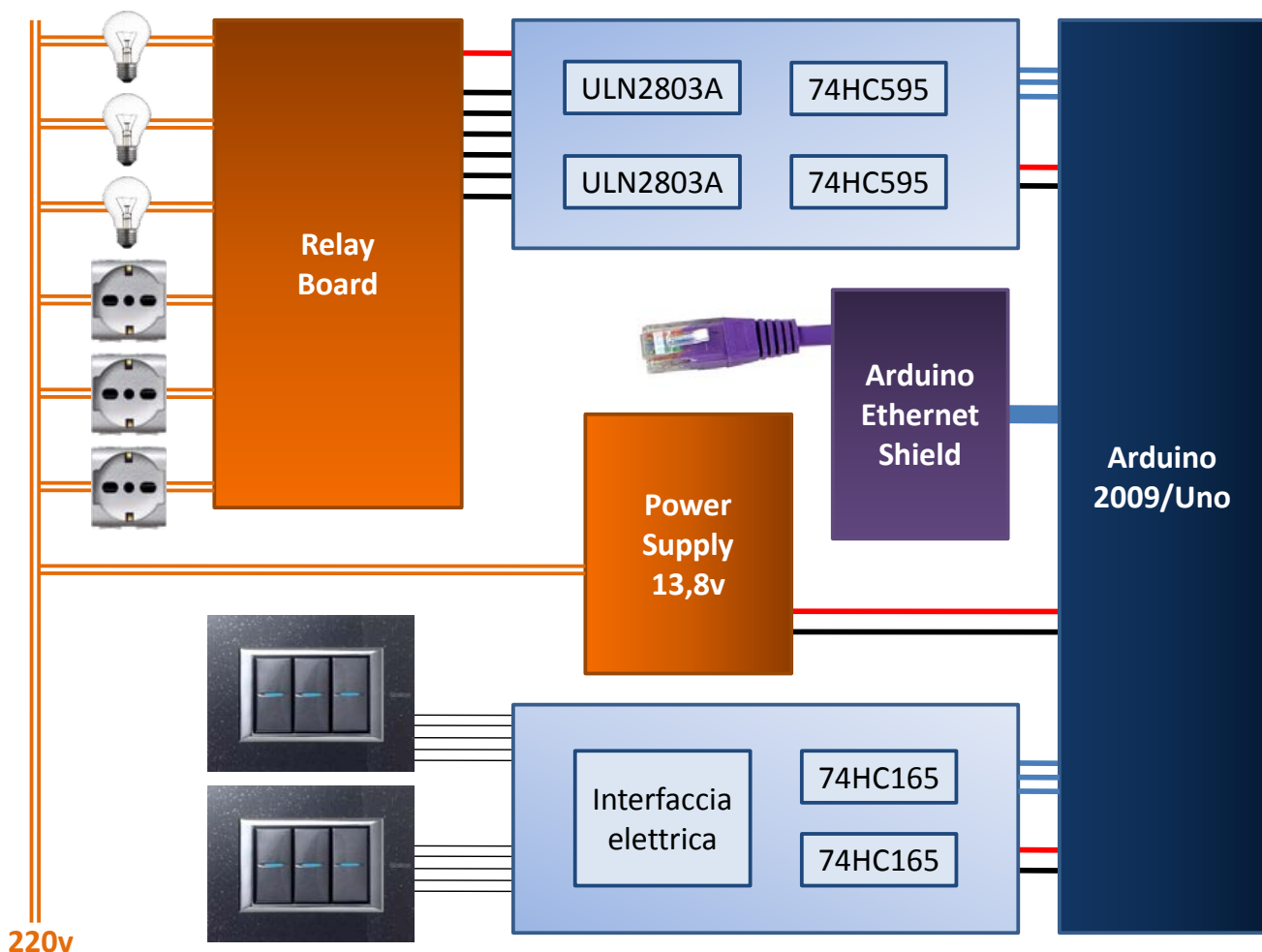
Il cuore del nodo è proprio l'Arduino (Uno/Duemilanove) che incorpora il **microprocessore ATmel ATmega328**. Tale microprocessore viene programmato con l'apposito **firmware OpenDomotica** che mette a disposizione, secondo la **configurazione effettuata sulla EEPROM** (integrata nella CPU), le operazioni basilari da svolgere al verificarsi di un evento locale (es. pressione di un pulsante).

Alla scheda Arduino è connessa la **scheda Ethernet Shield** che consente al nodo di essere connesso ad una rete Ethernet e, quindi, di essere controllato da un altro nodo o dal server, ma anche di controllare un altro nodo remoto qualora la configurazione lo preveda.

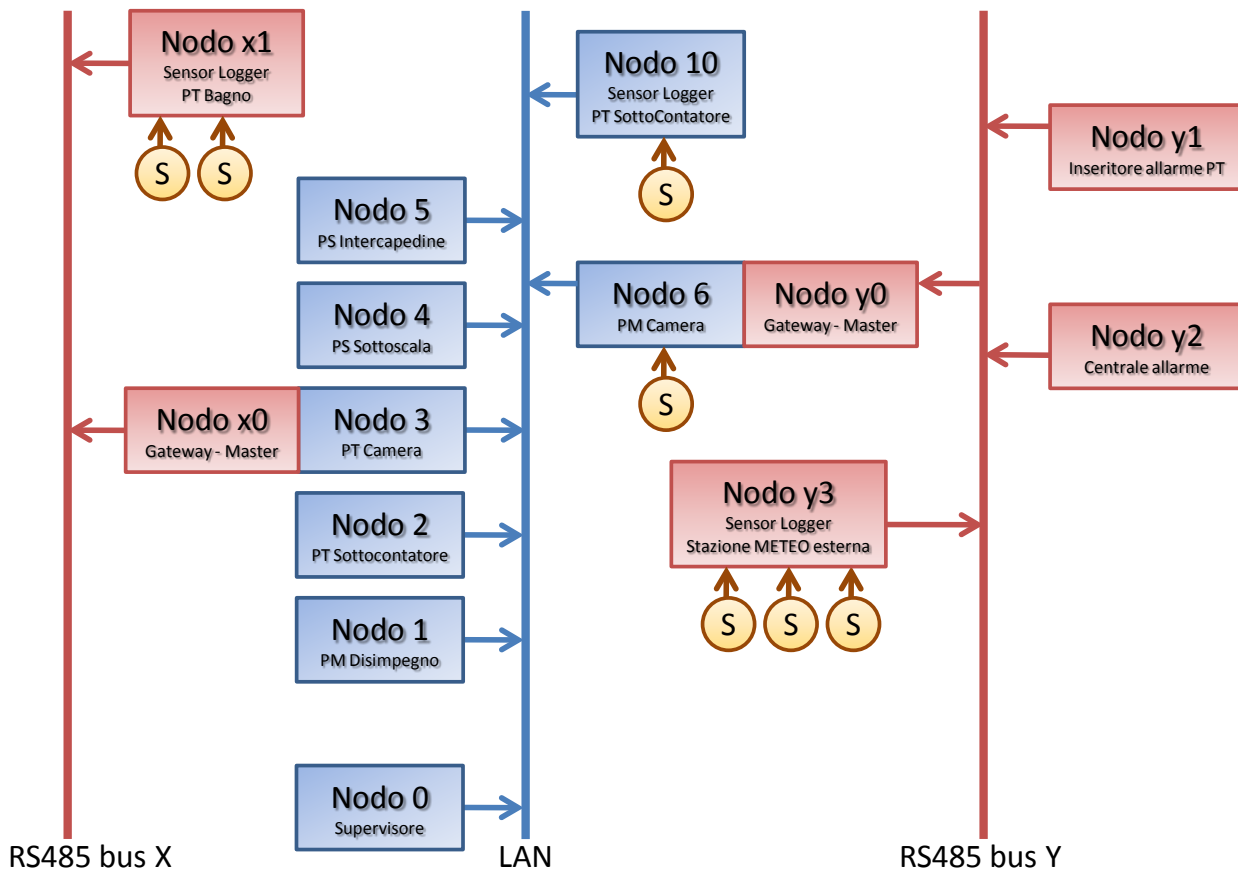
Sempre alla scheda Arduino è connessa una **scheda di multiplexing** che consente di collegare molteplici utenze/attuatori utilizzando tre soli pin di input/output del microprocessore ATmel, al fine di aumentare il numero di uscite previste. Il multiplexing viene effettuato utilizzando la **tecnica ShiftRegister** che opera una conversione seriale/parallelo dei dati trasmessi, consentendo di espandere la catena di output secondo le esigenze, potenzialmente senza limiti.

Il controllo delle varie utenze (luci, prese, motori, caldaia, irrigazione, etc.) viene effettuato attraverso relè meccanici ad alta potenza. Tali relè vengono pilotati direttamente dalla scheda di multiplexing.

Con la stessa tecnica di ShiftRegister, questa volta in modalità parallelo/seriale, vengono aumentati i pin di ingresso del microprocessore ATmel per poter acquisire i comandi da un maggior numero di pulsanti.



## Esempio di scenario di installazione di un network OpenDomotica



In questo scenario sono installati:

- una serie di Nodi (Nodo 1 – 5) per il controllo di tutte le utenze previste e l'acquisizione di tutti i pulsanti atti a comandarle
- un PC/Server (Nodo 0) che svolge la funzionalità di supervisore dell'impianto
- un nodo rilevatore di dati ambientali (Nodo 10). I dati raccolti da tale nodo vengono riportati al supervisore che ne farà uso per le logiche di funzionamento avanzate che non sono gestibili dai singoli nodi
- un bus dati separato (RS485) per lo scambio di dati prioritari tra moduli che necessitano di elevata affidabilità nella comunicazione

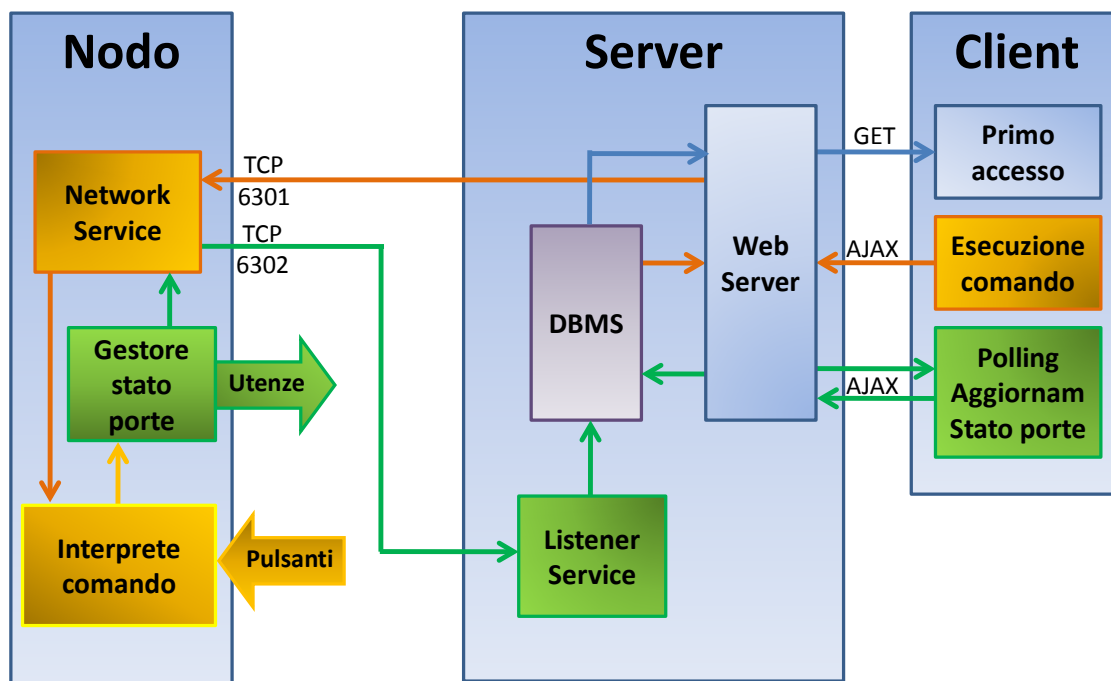
Il **bus RS485** mette in comunicazione i vari elementi di gestione **dell'impianto d'allarme** per garantirne il funzionamento anche nel caso di un black-out elettrico in cui il network Ethernet non sarebbe disponibile, in quanto gli apparati di rete sarebbero spenti. Al contrario, il bus RS485, essendo gestito direttamente dalle schede Arduino connesse ad un alimentatore con batteria tampone, può continuare a funzionare. Tale bus può anche essere utilizzato per veicolare i dati da altre apparecchiature (es. stazioni meteo, sensori ambientali, etc.).

L'interconnessione del bus prioritario RS485 al bus Ethernet viene effettuata mediante un **nodo Gateway** (Nodo 6) che si occupa di trasportare i messaggi da un bus all'altro.

## Architettura di sistema

L'architettura complessiva del sistema prevede:

- Una serie di nodi OpenDomotica
- Un PC/Server con servizi di Web Server, DBMS e OpenDomotica
- Uno o più client (PC, SmartPhone, Tablet, etc.)



Il **controllo remoto** dell'impianto domotico viene effettuato dai client mediante un'**interfaccia Web** che mette a disposizione le funzioni di gestione delle singole utenze, ma anche funzionalità più complesse come: **scenari** d'uso degli ambienti, **monitoraggio** dei consumi e dei dati ambientali, gestione dei profili di **climatizzazione**, programmi di **irrigazione**, interazione con l'**impianto di allarme**, etc.

Il Web Server utilizza il DBMS per recuperare i dati di configurazione e di stato attuale del sistema per costruire l'interfaccia da visualizzare sul client.

L'utente, attraverso i controlli dell'interfaccia, impartisce i comandi da eseguire al Web Server che si occupa di tradurli, attraverso i dati di configurazione presenti sul DBMS, ed inviarli al nodo di competenza.

Il nodo, ricevuto un comando dal network, a sua volta lo interpreta e lo esegue variando lo stato di una o più utenze. Ad ogni cambiamento di stato di un'utenza, sia attraverso comando remoto che locale (pressione di un pulsante), il nodo invia un messaggio al server (servizio Listener) che si occupa di registrare la variazione sul DBMS. In questo modo il dato registrato è sempre allineato con lo stato reale dell'impianto consentendo quindi, anche a tutti i client connessi in quel momento, di visualizzare la situazione sempre aggiornata.