

UNIVERSITÀ DEGLI STUDI DI PADOVA
FACOLTÀ DI INGEGNERIA

—
DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

—
TESI DI LAUREA IN INGEGNERIA ELETTRONICA

PROGETTAZIONE DI UN SISTEMA
DI MONITORAGGIO AMBIENTALE
TRAMITE RETE DI SENSORI
WIRELESS

RELATORE: CH.MO PROF. LUCA SCHENATO

TUTOR AZIENDALE: ING. RICCARDO TALLON

LAUREANDO: FEDERICO SPAGNOLO

ANNO ACCADEMICO 2010-2011

a mamma e papà...

*“Qualcosa è nascosto. Vai a cercarlo. Vai e guarda dietro i monti.
Corri! È perso e aspetta proprio te. ...”*

RUDYARD KIPLING

Indice

Introduzione	IX
1 Monitoraggio e Reti Wireless	1
1.1 Monitoraggio tramite WSN	1
1.2 Le WSNs nel corso degli anni	2
1.2.1 Scelte tecnologiche	2
1.2.2 Un pò di storia	5
1.3 Applicazioni	6
1.4 Sfide progettuali di una WSN	9
1.5 Architettura di un rete Wireless (modello OSI)	12
1.5.1 Livello fisico (tecnologie a confronto)	13
1.5.2 Livello Data Link	16
1.5.3 Livello Network	17
1.5.4 Livello Trasporto	19
1.5.5 Livello Applicativo	20
1.6 Sviluppi futuri	20
2 Monitoraggio Ambientale	23
2.1 Obiettivi del monitoraggio	23
2.2 Monitoraggio indoor e concetto di Comfort	23
2.2.1 Comfort	24
2.3 Concetto di ‘Occupancy’	26
2.3.1 Tecnologie per l’ occupancy sul mercato	26
2.4 Monitoraggio tramite CO2	28
2.4.1 State of the art	30
2.5 Monitoraggio in Inthebra s.r.l.	32
2.6 Risultati ottenuti	33
2.6.1 Locale A	35
2.6.2 Locale B	35
2.6.3 Locale C	37
2.6.4 Locale D	37
2.6.5 Locale E	39
2.6.6 Locale F	39

3	WSN e protocollo ZigBee	43
3.1	Il protocollo IEEE 802.15.4	43
3.1.1	Trasmissione dei bit a livello fisico (PHY)	43
3.1.2	Livello di accesso al canale (MAC)	45
3.1.3	Livello di rete (NTW)	47
3.1.4	Livello di Applicazione (APL)	48
3.1.5	Tipi di nodo	48
3.1.6	Topologie di rete	50
3.2	ZigBee PRO	53
3.3	Hardware utilizzato	54
3.4	Programmazione dei nodi della WSN	56
3.4.1	Ambiente di sviluppo	56
3.4.2	Coordinatore (SW)	57
3.4.3	Nodo Sensore (Router e EndDevice)	58
4	Raccolta dati e Visualizzazione	61
4.1	Hardware	61
4.1.1	Sensor Board (Nodo sensore)	61
4.1.2	Sensore di CO2	62
4.1.3	MiniPC ARM	64
4.2	Software	64
4.3	Dalla rete al database	66
4.3.1	Protocollo UART implementato	66
4.3.2	Scelta e caratteristiche del Database	68
4.3.3	Utilizzo del Database	70
4.4	Dal database al client	71
4.4.1	Web Server	72
4.4.2	Scelta del Web Server	72
4.4.3	Programma in php	73
4.4.4	Flex Builder	74
5	Conclusioni	79
5.1	Altri risultati ottenuti	79
5.1.1	Aula di scuola elementare	79
5.1.2	Aula A	80
5.1.3	Aula B	83
5.2	Consumi misurati	86
5.2.1	Sistema utilizzato	86
5.2.2	Risultati ottenuti	87
5.3	Possibili Ottimizzazioni	88
5.4	Conclusioni sugli obiettivi raggiunti	91
A	Planimetria completa di Inthebra s.r.l.	93
B	Tabella specifiche sensore di CO2	95

Introduzione

Negli ultimi anni, molti studiosi hanno portato la loro attenzione sullo studio del comfort negli ambienti di lavoro e di vita quotidiana scoprendo la presenza di una forte corrispondenza tra i dati ambientali ottenuti ed il rendimento cognitivo e il livello di attenzione delle persone occupanti i luoghi indagati. Il qui presente lavoro di tesi, riguarda appunto la progettazione e lo sviluppo di un sistema per il monitoraggio ambientale per ambienti indoor: è stato pensato e progettato essenzialmente per poter stabilire la presenza di persone all'interno di un ambiente chiuso. La novità portata da questo lavoro sta nel fatto che, l'indice di presenza o meno, viene deciso in base ad alcune grandezze fisiche tipiche del monitoraggio ambientale dell'aria; infatti, si è cercato di raggiungere lo scopo voluto raccogliendo semplicemente dati di temperatura, umidità, luce e CO₂. Un sistema che riesca a determinare la presenza di persone o meno, riuscirebbe in questo caso a dare delle informazioni importanti ai sistemi di comfort installati (impianti di areazione , tapparelle automatizzate, ecc. . .) per migliorare le condizioni di lavoro in uffici o altro.

Non sempre l'installazione di un sistema elettronico nell'ambito delle infrastrutture (uffici, scuole, ecc. . .) si rivela di facile attuazione: molte volte, infatti, non è possibile agire nell'impianto elettrico dei locali per aggiungere cavi di alimentazione, di rete e quant'altro. Alcune applicazioni cercano di ovviare a tale questione utilizzando sistemi wireless in cui non è necessaria alcuna modifica all'impianto già esistente.

Anche in questo progetto di tesi si è utilizzata una rete di sensori wireless per il monitoraggio di più nodi contemporaneamente. La rete utilizzata è di tipo Mesh con protocollo ZigBee; tutti i nodi comunicano le proprie informazioni con un nodo centrale, un miniPC, collegato a quest'ultimo riesce ad immagazzinarli e a pubblicarli su una pagina web diventando disponibili agli utenti della rete.

Capitolo 1

Monitoraggio e Reti Wireless

In questo primo capitolo, verrà affrontato il tema delle reti di sensori wireless per scopi di monitoraggio ambientale; in particolare, verranno discusse le principali caratteristiche di tali tipi di sistema partendo dai punti di forza, vedandone le possibili applicazioni sino ad arrivare ai possibili sviluppi per il futuro.

1.1 Monitoraggio tramite WSN

Con il termine ‘Wireless Sensor Network’ (o WSN) si indica una determinata tipologia di rete che, caratterizzata da una architettura distribuita, è realizzata da un insieme di dispositivi elettronici autonomi in grado di prelevare dati dall’ambiente circostante e di comunicare tra loro.

Una WSN può quindi essere definita come un insieme di nodi wireless interconnessi (anche detti *mote*, *sensor node*), aventi poca RAM e una CPU con prestazioni relativamente basse. La struttura di una Wireless Sensor Network prevede solitamente diversi nodi wireless sparsi in un’area, che inviano periodicamente dati rilevati tramite sensori ad un punto di raccolta, detto *base station* o *gateway* oppure *coordinatore*, il quale gestisce la rete, raccoglie i dati dei nodi e li inoltra ad un altro sistema remoto per ulteriori elaborazioni.

Come precedentemente accennato, una *Sensor network* offre una vasta gamma di utilizzi per esempio potrebbero essere la raccolta dati, la sorveglianza, il monitoraggio, la telemetria e molti altri ancora.

In base alle prestazioni del mini-processore montato su ogni nodo è possibile costruire una piccola applicazione con la quale, per esempio, si possono gestire ed azionare degli attuatori o dei sistemi di controllo.

Le componenti basilari di una rete per un sistema di questo tipo sono:

1. un insieme di sensori distribuiti
2. una rete di interconnessione (wireless nel caso in questione)
3. un punto di raccolta dei dati

4. un insieme di risorse computazionali con prestazioni medio elevate nel punto di arrivo dei dati della rete al fine di effettuare *datalogging*, correlazioni dei dati, elaborazione, monitoraggio dello stato ecc. . .

Uno schema semplificato della struttura di una rete di questo tipo è mostrato in figura 1.1.

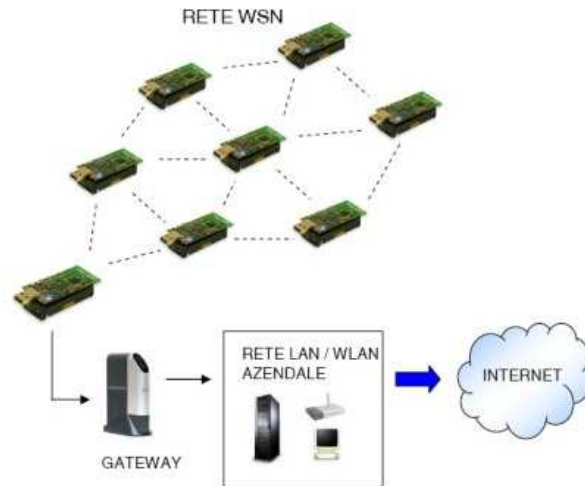


Figura 1.1: Struttura tipica di una WSN

Parte dell'elaborazione dei dati può essere già svolta all'interno della rete, in modo da alleggerire il carico di lavoro dell'unità centrale (che altrimenti dovrebbe effettuare calcoli su molti dati). Le architetture di elaborazione e di comunicazione dei dati sono specifiche del tipo di applicazione che si sta realizzando; quindi, un sistema di monitoraggio ambientale sarà diverso da uno di sorveglianza militare.

1.2 Le WSNs nel corso degli anni

1.2.1 Scelte tecnologiche

I sensori di una rete WSN possono avere varie funzioni, scopi e possibilità e, grazie alla continua evoluzione tecnologica, si scoprono sempre più ambiti di utilizzo. Le reti di sensori esistono già da parecchio tempo: le centraline meteorologiche, i radar di controllo del traffico aereo e la rete elettrica nazionale rappresentano alcuni esempi. Tuttavia esse richiedono hardware specializzato e specifici protocolli, che rendono queste reti particolarmente costose, l'esatto contrario delle WSN.

Lo sviluppo delle Wireless Sensor Network è iniziato molto recentemente ed è un ambito multidisciplinare poiché richiede conoscenze di radio e networking, signal processing, gestione di database, intelligenza artificiale, ottimizzazione delle risorse, algoritmi di risparmio energetico, architetture di sistemi per l'utente e piattaforme tecnologiche (hardware e software). Inoltre, proprio il continuo sviluppo della tecnologia, spinge sempre più lontano eventuali limiti progettuali.

Ad esempio, oggi esistono molti tipi di sensori, in grado di valutare diverse grandezze, di dimensioni veramente ridotte, dall'ottima affidabilità e dal basso consumo energetico. I sensori attualmente sul mercato possono rilevare elettricità, campo magnetico, onde radio, raggi infrarossi, coordinate geografiche, suoni, pressione, parametri ambientali (luminosità, temperatura, umidità, vento, anidride carbonica...), parametri biologici, parametri biochimici e altri ancora.

Un obiettivo commerciale che è stato prefissato è quello di realizzare sistemi con sensori basati sui micro sistemi elettromeccanici (MEMS) della dimensione totale di 1mm^3 [1]. I dati rilevati dai sensori vengono spediti all'interno della rete tramite dei collegamenti wireless a bassa potenza al punto di raccolta, che è spesso connesso ad Internet e che a sua volta può inoltrare i dati ad un punto di analisi. Il meccanismo di accesso al canale wireless è solitamente di tipo *contention-oriented* ad accesso casuale, come definito nello standard IEEE 802. Le WSN sono caratterizzate da alcuni aspetti chiave come, per esempio, i vincoli di potenza elaborativa, la durata limitata della batteria, un basso duty cycle e connessioni *many-to-one* (*molti-a-uno*), che creano delle sfide progettuali nell'ambito di trasporto dei dati, gestione della rete, disponibilità, confidenzialità, integrità e *in-network* processing.

Per di più, lo stack del protocollo realizzato deve essere quanto più leggero possibile, per poter soddisfare i vincoli imposti dai limiti hardware dei nodi. Il superamento di queste difficoltà, ha comportato un primo passo verso la creazione di uno standard per le WSN: già nei primi anni del 2000 furono definiti degli standard, che tuttavia erano proprietari. Solamente in seguito si è cercato di crearne uno aperto.

Dapprima sono stati esaminati quelli già disponibili ma son risultati non adatti a tale scopo, in particolare:

- **IEEE 802.11:** richiede troppe risorse hardware e offre una banda ben oltre le necessità di un nodo della WSN (che trasmette pochissimi byte alla volta)
- **Sistemi ad infrarossi:** richiedono un allineamento visivo tra i nodi.
- **IEEE 802.15.1 (Bluetooth):** risulta troppo complesso e oneroso in termini pecuniari.

Così si è arrivati a definire un nuovo standard, l'**IEEE 802.15.4** e lo **ZigBee** (ZigBee definisce solo i layer software sopra l'802.15.4 e supporta diverse applicazioni). Come si vedrà meglio nei prossimi capitoli, l'IEEE 802.15.4 opera nella banda radio ISM a 2.4 Ghz, permette data rate fino a 250 kbps e un range tipico tra i 10 e i 75 metri (in ambiente di utilizzo).

Ben prima della definizione di questi standard, furono create delle reti che potrebbero sembrare molto simili alle WSN, le MANET, ma che in realtà presentano sostanziali differenze. Le MANET, ovvero Mobile Ad Hoc Network, sono reti costituite ad un preciso scopo e per soddisfare una necessità comunicativa immediata. Solitamente le MANET sono reti wireless multihop e i nodi sono mobili.

Una tipica MANET è quella che viene creata tra le varie squadre di soccorso durante una situazione di *disaster-recovery*. Le sfide principali di una MANET sono quelle di riorganizzare la rete quando i nodi si muovono e gestire i problemi derivanti dalla comunicazione wireless (perdita di connessione, perdita di pacchetti, distanza massima di trasmissione,...); in parte queste sono anche le sfide di una WSN, ma i punti dove le due reti si differenziano veramente sono i seguenti:

1. **Equipaggiamento e applicazioni:** nelle MANET i terminali sono laptop, alimentati da batterie grosse e vi è un utente umano; le applicazioni tipiche sono le chiamate vocali o l'accesso a computer remoti (Web Server,...).
2. **Specificità d'uso:** le MANET non hanno la stessa versatilità d'uso delle WSN, e sono progettate solo a specifici scopi ed usi, nonché numero di nodi.
3. **Interazione con l'ambiente:** le WSN interagiscono con l'ambiente. Ciò significa, ad esempio, che per un periodo in cui non vi sono cambiamenti ambientali particolari, il traffico di rete è molto basso, mentre durante una situazione di allarme, il traffico può essere molto elevato. Le MANET invece supportano un tipo di traffico classico, consumer, caratterizzato da voce e dati.
4. **Scalabilità:** le WSN possono gestire migliaia di nodi, le MANET molti meno.
5. **Autoconfigurazione:** entrambi i tipi di rete richiedono questa funzionalità, anche se è più difficile implementarla sulle WSN, a causa dei molti vincoli già presenti. In ogni caso, l'autoconfigurazione è l'aspetto in cui le due reti più si assomigliano.
6. **Energia:** le reti di sensori hanno vincoli di consumo energetico molto più restrittivi delle MANET, per le quali la sostituzione di una batteria non presenta particolari problemi.
7. **Affidabilità e QoS:** in una MANET ogni nodo deve garantire una certa affidabilità; in una rete di sensori, la perdita di un nodo non deve destare preoccupazione. Per quanto riguarda il Quality of Service (QoS), una MANET può implementare delle politiche di QoS (ad esempio per il traffico voce), mentre una WSN possiede una tipologia di QoS totalmente diversa, dettata da altre regole (risparmio energetico in primis).
8. **Semplicità e risorse limitate:** il software di un nodo di una WSN deve essere il più leggero possibile e occupare poca RAM; in una MANET i tipi di protocolli adoperati non soddisfano nessuna di queste condizioni.
9. **Mobilità:** in entrambi le reti ci possono essere nodi o gruppi di nodi in movimento, ma nelle WSN ciò può implicare complicazioni maggiori.

Quindi, il fatto che le Sensor Network supportino vari tipi di applicazione, interagiscano con l'ambiente e debbano ponderare vari tradeoff, permette di considerarle un sistema diverso dalle MANET. Dunque, anche a causa di queste differenze tra le WSN e le MANET, è stato necessario definire un nuovo standard, ovvero il già citato IEEE 802.15.4.

1.2.2 Un pò di storia

La storia di queste reti può essere divisa in quattro fasi principali:

1. **Sensor Network dell'epoca della guerra fredda:** durante la guerra fredda furono sviluppate negli Stati Uniti delle reti acustiche di sorveglianza sottomarina, che vengono usate ancora oggi per rilevare attività sismiche sottomarine. Inoltre, nel Nord America, furono create delle reti di radar per la difesa aerea, i cui sensori erano costituiti da degli aerei *Airborn Warning and Control System (AWACS)*.
2. **Iniziative del DARPA:** Il maggior stimolo allo sviluppo delle reti di sensori fu nei primi anni 80, quando il DARPA (Defence Advanced Research Projects Agency) sponsorizzò la creazione di Distributed Sensor Network (DSN) per determinare se protocolli TCP/IP, insieme alla rete ARPA, potessero essere usati come reti di sensori. Le DSN prevedevano diversi nodi autonomi equamente spazati, in grado di collaborare tra loro, il cui obiettivo era quello di raccogliere i dati in nodi che potessero utilizzare al meglio quelle informazioni. Le applicazioni delle DNS erano il calcolo distribuito, il signal processing e il tracking. I componenti di queste reti erano sensori acustici, protocolli di comunicazione di alto livello, algoritmi di calcolo e software distribuito.
3. **Applicazioni militari degli anni 80 e 90:** in questo periodo furono sviluppati i primi prodotti commerciali derivanti dalla tecnologia delle DSN che, proprio grazie alla commercializzazione, erano caratterizzati da un costo minore e dall'implementazione dei primi standard. Tali reti di sensori erano adoperati in situazioni di *network-centric warfare*, ovvero in ambienti in cui i sistemi d'arma devono cooperare tra di loro scambiandosi informazioni sull'obiettivo. I sensori infatti erano in grado di tracciare obiettivi multipli, anche molto distanti e con ottimi tempi di risposta. Esempi di questa tecnologia erano le reti di radar per rilevare obiettivi aerei, reti di sensori acustici negli oceani per rilevare sottomarini e reti di sensori schierate sul campo di battaglia.
4. **Ricerca attuale:** dopo l'evoluzione tecnologica degli anni 90 e dei primi anni del 2000, si è giunti ad una nuova generazione di sensori, che può essere definita come seconda generazione di prodotti commerciali. Il progresso ha consentito lo sviluppo di tecnologie ad alta densità come i MEMS e i recenti *NEMS (Nanoscale Electromechanical Systems)*. La standardizzazione è

diventata un fattore determinante per la diffusione delle WSN e la nascita di sistemi Bluetooth, WiFi, Wimax e ZigBee è un fattore che abilita la connettività totale. Tutto ciò, insieme a nuovi processori a basso costo e basso consumo energetico, consente oggi di adoperare le reti di sensori per molteplici applicazioni. La ricerca nelle soluzioni commerciali si sta muovendo verso la definizione di topologie di rete *mesh*, *peer-to-peer* e *cluster-tree*, di standard di sicurezza e di profili di applicazioni comuni.

1.3 Applicazioni

Le reti di sensori fanno parte di un'area di studio molto attiva dal punto di vista dell'innovazione tecnologica; grazie alle loro potenzialità riescono ad essere impiegate in molti campi d'utilizzo. Per l'analisi delle varie applicazioni, una prima suddivisione può essere fatta in base alla finalità della rete. Di seguito si riportano quattro macro-settori in cui è possibile dividere il mondo delle WSN.

- **Monitoraggio.** Questo tipo di rete viene utilizzata per tracciare in maniera continuativa una certa grandezza. La sorgente da monitorare può essere un singolo sensore o una sottorete da cui proviene un aggregato di dati. Una rete di questo genere richiede un campionamento solitamente isocrono e fisso con un consumo energetico considerevole.
- **Riconoscimento di eventi.** Un'altro importante scenario applicativo risulta essere l'*event detection*: la rete deve accorgersi di situazioni di 'allarme', ossia di quando una determinata grandezza esce dai livelli stabiliti. Questo monitoraggio è potenzialmente meno pesante del precedente a livello energetico, in quanto il nodo entra in uno stato attivo solo in casi eccezionali.
- **Classificazione di oggetti.** L'obiettivo di queste reti è il riconoscimento di alcune grandezze tra un insieme di prototipi noti. Questo implica un carico computazionale superiore agli altri tipi di rete.
- **Tracciamento di oggetti.** In questo caso la rete funge da sistema di sorveglianza, riconoscendo e stimando la posizione di alcuni oggetti in una determinata area geografica.

Oltre a questa prima suddivisione si possono descrivere i vari scenari in cui le WSN sono presenti. Inizialmente le Sensor Network erano adoperate solo in contesti particolari come la rilevazione di radiazioni, tracciamento di obiettivi e sorveglianza militare, rilevazione di dati biomedici, monitoraggio di un'area e rilevazione di attività sismiche. Più recentemente, l'attenzione è stata rivolta verso reti di sensori biologici e chimici per applicazioni di sicurezza nazionale, nonché verso lo sviluppo di soluzioni commerciali alla portata di molti.

Un piccolo elenco di possibili applicazioni è il seguente:

- **Settore Militare:**

- monitoraggio di forze nemiche
- monitoraggio di forze alleate ed equipaggiamenti
- sorveglianza di campi di battaglia
- stima dei danni
- rilevazione di attacchi nucleari e biochimici

Come precedentemente detto, le WSN son nate, assieme a molte altri sistemi e tecnologie oggi presenti nel mercato, da sviluppi per applicazioni belliche e militari.

- **Settore ambientale:**

- rilevamento di incendi boschivi
- rilevamento di inondazioni
- monitoraggio di smottamenti e frane
- monitoraggio di microclimi
- agricoltura di precisione

Uno dei campi di maggior utilizzo delle WSN è il monitoraggio ambientale. Con l'avanzare dell'industria e degli agenti inquinanti diventa essenziale controllare aree geografiche molto estese, soprattutto a ridosso dei centri abitati, per valutare l'impatto ambientale di alcune scelte umane.

Un altro ambito d'interesse è lo studio di zone pericolose, come vulcani o zone a rischio sismico.

Gli studi scientifici su flora e fauna si possono avvalere di una rete di monitoraggio ambientale: questo porta anche nell'agricoltura un'ulteriore rivoluzione tecnologica, permettendo di controllare un certo tipo di coltura e intervenire in tempo reale con trattamenti opportuni per debellare insetti o situazioni critiche.

- **Settore biomedico:**

- monitoraggio a distanza di parametri fisiologici
- tracking di medici e pazienti all'interno di un ospedale
- gestione dei medicinali

L'utilizzo di WSN all'interno di ospedali può essere un valido supporto per seguire costantemente alcuni parametri fisiologici dei pazienti e rendere più flessibili e portabili alcuni macchinari. Ma questo scenario è solo il primo

passo verso una rete di sensori molto piccoli e distribuiti all'interno dei tessuti che possano misurare la temperatura, la pressione sanguigna e molti altri parametri di rilievo. Le prime applicazioni in questo senso si hanno nei dispositivi di personal healthcare (per attività ginniche soprattutto), magari integrati in orologi da polso o bracciali appositi che rilevino il battito cardiaco o le calorie bruciate durante l'attività.

- **Settore domestico**

- Home Automation
- lettura contatori (Smart metering)

Tutti gli elettrodomestici più moderni hanno un'elettronica molto evoluta e il futuro consiste nell'interazione tra i vari elettrodomestici. L'unità di controllo di questi dispositivi permette funzioni sempre più avanzate, aprendo uno scenario in cui i vari elettrodomestici possano collaborare autonomamente per gestire in maniera efficiente le faccende domestiche. La chiave per questo è la possibilità di comunicare: sicuramente la tecnologia dovrà essere di tipo wireless. Un'altra applicazione importante è la gestione del riscaldamento, della ventilazione e dell'illuminazione. L'uso di sensori wireless riduce i costi di installazione e rende molto meno invasiva la modifica di sistemi esistenti (si pensi alla necessità di forare le pareti: con la tecnologia wireless questo non è più necessario).

- **Settore commerciale (automotive).**

- controllo ambientale in costruzioni industriali ed edifici
- controllo di inventari
- tracking di veicoli
- monitoraggio della *supply chain*¹

In questa categoria rientrano molti scenari diversi. Sicuramente alcuni fra i più interessanti sono: ambiente automotive (utilizzo delle WSN su automobili), antifurto intelligenti e versatili, servizi all'interno di musei o scuole (localizzazione di persone o oggetti). Nell'ambito automobilistico si intende sia un monitoraggio del traffico sia strumenti di navigazione autonoma, anche se per questi ultimi il livello di affidabilità necessario è molto elevato, superiore a quello ottenibile attualmente.

¹Le reti utilizzate in questi ambienti sono relativamente statiche e gerarchiche, ma hanno dei vincoli di affidabilità e prestazioni molto più stringenti che in altri scenari. Gli utilizzi sono molteplici, ma tutti finalizzati al controllo dei processi produttivi. Spesso si intende realizzare una sottorete di monitoraggio wireless da interfacciare mediante opportuni gateway (ossia dei traduttori di protocolli) con bus di campo tipici degli ambienti industriali (ossia bus di tipo cablato).

1.4 Sfide progettuali di una WSN

Non è possibile creare un prototipo di WSN in grado di gestire tutte le applicazioni sopra elencate. Tuttavia, anche se ognuna di esse avrà sfide particolari, molte saranno condivise. Tali sfide comuni si possono riassumere nelle seguenti caratteristiche:

- **Type of Service.** Le WSN non sono delle reti tradizionali che si limitano a comunicare dei bit, bensì devono provvedere informazioni utili o agire ad eventi particolari. Per questo motivo servono nuovi paradigmi, nuove interfacce utente e nuovi modi di pensare ai servizi della rete.
- **Quality of Service (QoS).** I parametri QoS adottati tradizionalmente come ritardo, banda minima e jitter, non servono se si stanno utilizzando software che tollerano i ritardi e se i nodi scambiano pochi pacchetti alla volta. Inoltre, in alcuni casi basta ricevere qualche dato ogni tanto, mentre in altri tutti i dati devono essere ricevuti oppure devono essere ricevuti entro un certo tempo. Ciò che conta è quindi la quantità e la qualità dei dati ricevuti da un nodo di raccolta di una certa area. Ad esempio, delle metriche valide possono essere l'affidabilità nel rilevamento di certi eventi o l'approssimazione di certe misure.
- **Tolleranza ai guasti.** Un nodo potrebbe esaurire la batteria, subire danni fisici o perdere in qualunque altro modo il collegamento wireless con gli altri nodi (presenza di un ostacolo di disturbo non previsto tra i nodi). Una simile situazione non deve però danneggiare il resto della rete e un altro nodo dovrebbe possibilmente prendere il posto del nodo perso.
- **Lifetime.** Nella maggior parte degli scenari, i nodi sono alimentati da batterie che non è possibile o non è conveniente sostituire. In ogni caso, l'obiettivo di una WSN è quello di rimanere attiva più a lungo possibile o, almeno, per la durata della sua missione. Quindi, il **risparmio energetico** assume un ruolo fondamentale in una Wireless Sensor Network. Anche nelle applicazioni in cui è previsto l'ausilio di un piccolo pannello solare per ricaricare la batteria dei nodi, il risparmio energetico rimane un fattore critico e in queste situazioni l'obiettivo è quello di mantenere la rete attiva per un tempo indeterminato. Tuttavia, l'implementazione di meccanismi di risparmio energetico richiede dei compromessi con la qualità del servizio: la soluzione è ovviamente quella di trovare un giusto bilanciamento tra le due caratteristiche. La definizione di lifetime non è univoca, nel senso che dipende dall'applicazione che si vuole misurare: a volte infatti si indica con lifetime il tempo entro il quale il primo nodo della rete finisce la propria energia (o comunque non funziona più); altre volte invece corrisponde al momento in cui il 50% dei nodi vengono persi; oppure indica la prima volta in cui una regione sotto controllo non è più monitorata da alcun nodo.

- **Scalabilità.** Una WSN può contenere fino a migliaia di nodi e la sua architettura deve essere in grado di supportarli tutti.
- **Densità di nodi non uniforme.** In una rete di sensori possono esserci zone molto affollate (con molti nodi) e zone con pochissimi nodi sparsi. La densità dei nodi può variare nello spazio e nel tempo (ad esempio perché i nodi finiscono la batteria) e la rete deve essere in grado di adattarsi a queste variazioni.
- **Programmabilità.** I nodi devono essere in grado di poter cambiare i propri compiti in qualunque momento, ovvero devono poter essere riprogrammabili. A tal proposito, son molti gli studi relativi a reti che riescono a riprogrammarsi autonomamente senza dover programmare manualmente ogni singolo nodo VIRAL.
- **Auto-mantenimento.** Dato che sia la WSN sia l'ambiente in cui si trova sono in continuo mutamento, la rete deve essere in grado di adattarsi, monitorando il proprio stato di salute, aggiornando i propri parametri, decidendo tra nuovi compromessi (ad esempio diminuendo la qualità del servizio quando l'energia sta per terminare).

Per risolvere tutte queste sfide progettuali, nel corso degli anni, sono stati sviluppati diversi meccanismi per la comunicazione, architetture di sistema e protocolli di varie tipologie. Con tali sviluppi si riescono ad ottenere le seguenti importanti funzionalità:

- **Connessioni wireless multihop.**

La comunicazione diretta tra due nodi non è sempre possibile, poiché potrebbero esserci ostacoli oppure perché i nodi sono molto distanti tra loro e l'utilizzo di una potenza trasmittiva elevata comporterebbe un rapido esaurimento della batteria. Quindi la soluzione è quella di adoperare dei nodi che fungano da relay verso altri nodi (Fig.1.2).

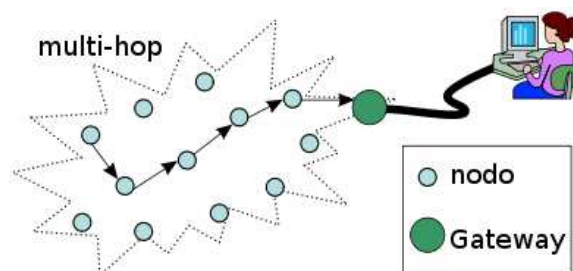


Figura 1.2: Rappresentazione del multi-hop

- **Operazioni energeticamente efficienti.**

E' importante che tutte le operazioni compiute tengano in considerazione il risparmio energetico e bisogna, possibilmente, evitare la formazione di

hotspot, ovvero regioni o gruppi di nodi che esauriscono la propria energia molto più rapidamente degli altri.

- **Autoconfigurazione.**

La rete deve essere in grado di configurare automaticamente tutti i suoi parametri vitali. Per esempio deve gestire autonomamente l'ingresso di un nuovo nodo oppure l'aggiornamento delle tabelle di routing dopo la perdita di alcuni nodi.

- **Collaborazione e in-network processing.**

In alcune applicazioni, un singolo nodo non è in grado di capire se si è verificato un evento. Per questo motivo è necessario che i nodi collaborino tra di loro e compiano *in-network processing*, ovvero eseguano alcuni calcoli sui dati, tipicamente *data-aggregation* (ad esempio il calcolo della temperatura media di una zona) in modo da ridurre la quantità di dati trasmessa attraverso la rete; oppure sfruttando la correlazione tra misurazioni di più sensori.

- **Data-centric.**

In una rete di comunicazione tradizionale, lo scambio di dati avviene tra entità aventi ognuna un indirizzo di rete specifico, quindi si tratta di un'architettura *address-centric*. In una WSN, non importa tanto chi fornisce il dato, ma da quale regione proviene. D'altronde un nodo può essere ridonato da più nodi, e quindi si perde l'individualità dei vari componenti. Ciò che interessa è richiedere una certa informazione ad una certa area monitorata, e non richiedere una certa informazione ad un certo nodo. È un concetto simile alla query di un database 'Visualizza tutte le aree in cui la temperatura è maggiore di X' oppure 'Richiedi i dati di umidità della regione X'.

- **Località.**

Per risparmiare risorse hardware, il nodo deve interessarsi e memorizzare informazioni di routing solo verso i nodi vicini a lui. Così facendo, nel momento in cui la rete dovesse crescere esponenzialmente, le risorse hardware occupate rimarrebbero inalterate. Chiaramente, conciliare località e protocolli di routing efficienti è una delle sfide da affrontare.

- **Bilanciamento dei tradeoff.**

Sia durante la fase di progettazione della Wireless Sensor Network sia durante il suo *runtime*, bisogna ponderare diversi *trade-off*, anche contraddittori tra loro. Alcuni di questi sono già stati accennati: lifetime e qualità del servizio, lifetime della rete e lifetime del singolo nodo, densità della rete ed efficienza del routing, solo per citarne alcuni.

	Data unit	Layer	Function
Host layers	Data	7.Application	Network process to application
		6.Presentation	Data representation, encryption and decryption
		5.Session	Interhost communication
	Segments	4.Transport	End-to-end connections and reliability, flow control
Media Layers	Packet	3.Network	Path determination and logical addressing
	Frame	2.Data Link	Physical addressing
	Bit	Physical	media, signal and binary transmission

Tabella 1.1: Il modello di riferimento OSI (*Open System Interconnection*)

1.5 Architettura di un rete Wireless (modello OSI)

Per descrivere l'architettura di rete di una Wireless Sensor Network, si può ricorrere al modello OSI (*Open Systems Interconnection*), descritto dalla tabella 1.1. Venga tenuto da subito in considerazione che la separazione dei layer in una WSN non è così netta; al contrario, i layer si sovrappongono tra loro, in particolare il livello 2 (*livello MAC*) e il livello 3 (*livello Network*).

Questa è una scelta progettuale dovuta, per motivi che vanno ricondotti alle caratteristiche e ai limiti di una Sensor Network. È insomma una conseguenza dei tradeoff elencati nei paragrafi precedenti: per creare dei protocolli di comunicazione efficienti, leggeri, capaci di supportare avanzate funzionalità di risparmio energetico e flessibili ai cambiamenti della rete stessa, il *livello Mac* e il *livello Network* si trovano spesso a lavorare in stretta collaborazione tra loro, violando così il principio di separazione dei livelli.

I livelli che costituiscono il cuore di una WSN sono il secondo e il terzo, proprio poiché vanno a definire il 'carattere' della rete, ovvero sono questi due livelli che si occupano di risolvere le sfide più difficili di una rete di sensori wireless. I layer rimanenti assumono invece un'importanza minore, specialmente dal quinto in su. È vero che anche il livello fisico svolge un ruolo fondamentale, senza il quale la rete non esisterebbe nemmeno; tuttavia tutte le reti di sensori senza fili a bassa potenza sfruttano apparati radio con caratteristiche simili, ovvero bassa potenza di trasmissione, basso consumo, basso *bit-rate*. Quindi, al di là di fattori come la frequenza o il tipo di modulazione adoperato, il profilo di una WSN a bassa potenza non è definito dal livello Fisico. Il livello del Trasporto, se viene implementato, non è definito da particolari regole e, in ogni caso, non è complicato come nelle reti classiche TCP/IP. I livelli 5, 6 e 7 sono solitamente validi solo per i gateway o per gli host. Fanno eccezioni le reti ZigBee e altre soluzioni OEM, che implementano anche parte del livello Applicazione nei nodi. Verranno ora

	802.11	802.15.1/ Bluetooth	802.15.4/ZigBee
Copertura(metri)	100	10-100	10-75
Throughput(Mbps)	2-30	1-2	0.25
Consumo Energetico	Medio	Basso	Molto basso
Autonomia	Minuti/ poche ore	Diverse ore/ pochi giorni	Giorni/ pochi anni
Dimensioni	Medie	Piccole	Molto Piccole
Rapport costo/ complessità	Alto	Medio	Basso

Tabella 1.2: Tecnologie Wireless a confronto ([4])

analizzati i primi layer del modello.

1.5.1 Livello fisico (tecnologie a confronto)

Esistono varie tecnologie utilizzabili per le Wireless Sensor Network: Wi-Fi, 3G, Wimax, Bluetooth e 802.15.4/ZigBee. Dato che solitamente esse ricorrono all'802.15.4/Zigbee, le altre tipologie di connessione non verranno trattate. Infatti connessioni WiFi, 3G o Wimax vengono per lo più sfruttate per il collegamento del nodo gateway ad Internet o ad un altro host, ma non per creare collegamenti tra i vari nodi. La Figura 1.3 mostra queste tecnologie a confronto per range di trasmissione e data rate, mentre la Tabella 2.1, analizza altri aspetti delle tecnologie nel dettaglio.

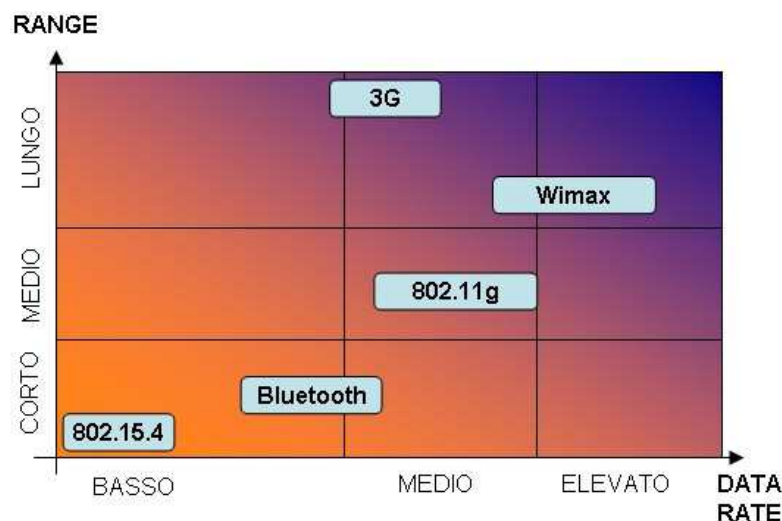


Figura 1.3: Tecnologie a confronto per quel che riguarda rapporto distanza di trasmissione/data-rate

Gli aspetti che devono essere tenuti in considerazione a livello fisico di comunicazione radio sono quelli tipici di una comunicazione senza fili, quali principalmente:

1. **Riflessione.**

Avviene quando un'onda che si sta propagando in un mezzo incontra un oggetto di dimensioni maggiori della sua lunghezza d'onda.

2. **Diffrazione.**

In questo caso la traiettoria tra trasmettitore e ricevitore è ostruita da alcune superfici con contorni irregolari: l'onda si 'piega' attorno all'ostacolo. In alcuni casi, questo fenomeno può risultare vantaggioso, soprattutto se è assente il *line-of-sight*(*LOS*); Fig.1.4

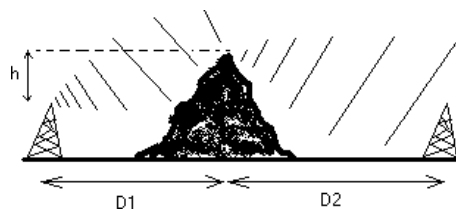


Figura 1.4: Rappresentazione di una comunicazione radio mediante il fenomeno della diffrazione

3. **Diffusione (Scattering).**

Fenomeno che avviene se l'onda incontra oggetti con una dimensione inferiore della propria lunghezza d'onda (foglie, cartelli stradali, ecc).

Questi fenomeni causano distorsione e attenuazione del segnale radio, che risulta essere la somma di varie componenti riflesse, diffratte e diffuse dagli ostacoli incontrati dall'onda. Tale effetto viene denominato *multipath*; infatti, riflessione, diffrazione e diffusione danno origine a traiettorie (path) di propagazione diverse, oltre a quella *LOS* tra mittente e destinatario. Quando ci sono più percorsi di propagazione radio, il segnale finale ricevuto è la somma vettoriale di tutte queste componenti provenienti da ogni direzione e da ogni angolo. Alcune di esse si sommeranno al segnale del path diretto, dando origine ad un'interferenza costruttiva, altre invece si sottrarranno, causando così un'interferenza distruttiva. Un altro fattore critico nella propagazione radio è la *mobilità*: anche quando le due parti comunicanti sono ferme, l'ambiente circostante può essere in movimento (ad esempio i rami delle piante che si muovono con il vento), provocando così improvvise fluttuazioni del segnale, dovute ai fenomeni appena descritti. Se si è all'interno di edifici, la situazione risulta ancora più difficile, a causa della presenza di muri, pavimenti, tubature e persone in movimento che degradano ulteriormente il segnale. Infine, un problema critico che si verifica con i protocolli

MAC di tipo CSMA (Carrier Sense Multiple Access²) è quello del *nodo nascosto* e del *nodo esposto*.

Il nodo nascosto è definito come il nodo entro il raggio d'azione del destinatario ma fuori dalla copertura del nodo trasmettitore. Si faccia riferimento alla Figura 1.5: si supponga che B sia nel raggio di copertura di A e di C, ma A e C siano fuori copertura reciprocamente. A vuole trasmettere a B: effettua il sensing del canale, sente che non è occupato e avvia la trasmissione. Ad un certo punto, C deve spedire un messaggio a B, mentre A sta ancora trasmettendo a B. C effettua il sensing del canale, lo considera libero, perché non sente il segnale di A, e inizia a trasmettere a B, causando così una collisione in B, senza che gli altri nodi se ne accorgano; con il risultato che B perde entrambi i messaggi, di A e di C.

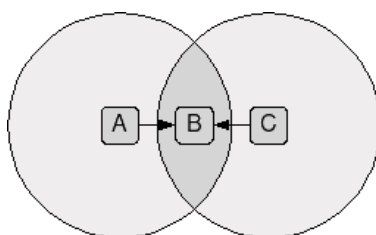


Figura 1.5: Il problema del nodo nascosto

Il nodo esposto è quel nodo nel raggio di trasmissione del mittente ma non del destinatario. L'esempio in questo caso è raffigurato in Figura 1.6: B si trova nel raggio di trasmissione di A e C, ma A e C sono fuori copertura reciprocamente, e D è nel raggio di trasmissione di C. Si supponga che B voglia mandare un pacchetto ad A: effettua il sensing del canale, sente che è libero e trasmette. A questo punto, C vuole trasmettere a D: effettua il sensing del canale, sente che è occupato dalla trasmissione di B e posticipa la sua trasmissione. Tuttavia, C avrebbe potuto trasmettere lo stesso subito, poiché D è fuori dalla portata di B e non si sarebbero verificate collisioni.

Per risolvere questi due problemi sono state proposte due soluzioni: la prima consiste nell'utilizzo di due canali, uno per i dati e uno di controllo. Quest'ultimo serve a segnalare agli altri nodi che è in corso la ricezione di un pacchetto. Tale soluzione però non è quella solitamente implementata nei dispositivi wireless. Infatti, il meccanismo generalmente usato è quello del Collision Avoidance (da cui il protocollo CSMA/CA), che risolve però solo il problema del nodo nascosto. Tale tecnica consiste nella prenotazione del canale, da parte del nodo trasmettitore, tramite un pacchetto RTS (Request to Send). Gli altri nodi nel raggio di coper-

²A differenza di quanto avviene nelle reti LAN su cavo (IEEE802.3), per le reti wireless è stata adottata la tecnica di accesso multiplo con rilevamento della portante ed eliminazione delle collisioni, CSMA-CA (Carrier Sense Multiple Access with Collision Avoidance). In sostanza, significa che ogni dispositivo prima di iniziare una trasmissione deve ascoltare il mezzo e capire se è già in corso una trasmissione. Se c'è già un nodo che sta trasmettendo, allora sarà effettuata la ritrasmissione successivamente con un ritardo casuale. La CSMA-CA viene adottata nelle reti ZigBee semplici di tipo peer-to-peer come per esempio sistemi di sicurezza in cui il dispositivo è in modalità sleep per il 99,99% del tempo.

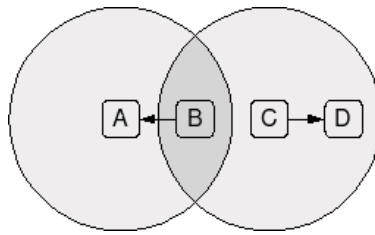


Figura 1.6: Il problema del nodo esposto

tura del trasmettitore confermano la ricezione dell'RTS spedendo un messaggio CTS (Clear to Send) al nodo e non occupano il canale per il tempo specificato nell'RTS. Le caratteristiche del livello fisico dipendono ovviamente dalla tecnologia scelta, ma, come già accennato, per le WSN la tecnologia predominante è l'IEEE 802.15.4/ZigBee.

1.5.2 Livello Data Link

Il livello Data Link è composta da due sottolivelli: *LLC* e *MAC*. Il primo, detto *Logical Link Control* layer, fa da tramite tra i livelli superiori e il livello fisico, consentendo l'interoperabilità tra diversi tipi di rete. Tuttavia, l'uso di questo sublayer è molto limitato, dato che l'interoperabilità è garantita da altri protocolli del network layer. Il livello *MAC* (*Medium Access Control*) riveste invece un ruolo molto importante e si occupa di tre funzioni fondamentali:

1. l'assemblaggio dei dati in frame tramite l'aggiunta di un *header*, contenente informazioni sull'indirizzo, e un *trailer*, contenente informazioni per la correzione degli errori;
2. il disassemblamento dei frame ricevuti per estrarre informazioni sull'indirizzo e la correzione degli errori;
3. la gestione dell'accesso al mezzo di trasmissione condiviso.

Il requisito fondamentale di un MAC per reti WSN è la necessità di risparmiare più energia possibile. Ci sono diversi fattori che contribuiscono allo spreco energetico, come l'eccessivo *overhead*, l'*idle listening*, le collisioni di pacchetti e l'*overhearing*. Inoltre l'accesso regolato al mezzo trasmissivo comporta lo scambio di informazioni di controllo e di sincronizzazione tra i nodi. Il continuo scambio di tali dati può provocare perdite significative di energia. Lo stesso effetto e un *throughput* di rete diminuito possono essere causati anche da un prolungato ascolto del canale in attesa di pacchetti.

La ritrasmissione di pacchetti persi a causa di collisioni è un'ennesima fonte di spreco, nonché di degrado delle prestazioni. Allo stesso modo l'*overhearing*, ovvero la ricezione e la decodifica di pacchetti destinati ad altri nodi, comporta inutili perdite prestazionali ed energetiche.

L'obiettivo di un MAC è quello di impedire queste situazioni appena descritte. Ci sono tre gruppi principali di protocolli MAC: *Schedule-based*, *Contention-based* e *ibridi*. I primi regolano l'accesso al canale tramite una schedulazione, preassegnata ad ogni nodo, affinché un solo nodo alla volta possa occupare il mezzo trasmissivo. I protocolli Contention-based non effettuano invece alcuna preallocazione delle risorse e l'accesso al canale condiviso è garantito *on-demand*. Se due nodi tentano l'accesso contemporaneo, si verifica una collisione. Il compito di questo tipo di MAC è quello di minimizzare, e non di evitare completamente, l'occorrenza di queste collisioni. Per ridurre il consumo energetico, questi protocolli MAC differiscono tra loro nel meccanismo adoperato per minimizzare la probabilità di collisione, l'overhearing e l'overhead per il controllo di accesso al canale. Il problema delle collisioni viene risolto tipicamente ricorrendo ad algoritmi casuali distribuiti per schedulare l'accesso al canale tra i contendenti. Per evitare l'overhearing, si mettono in stato dormiente i nodi (disattivando la radio) il prima possibile. Ciò può però provocare difficoltà di comunicazione tra nodi vicini, che possono essere risolte usando protocolli MAC con scheduling meno restrittivi.

1.5.3 Livello Network

Il livello Network si occupa principalmente del routing. Il routing ha un ruolo importantissimo, dato che lo scopo di una rete di sensori è quello di raccogliere dati da vari punti di un dominio, processarli e farli convogliare tutti in un unico punto (detto sink) dove risiede un'applicazione specifica. Per far sì che questi dati raggiungano la destinazione in maniera efficiente, cioè in maniera affidabile e con un basso costo energetico, è necessario che i protocolli di routing individuino dei cammini ottimali tra i nodi e il sink (o i sink). Le caratteristiche di una WSN come la dinamicità (nodi che perdono il link o esauriscono la batteria, nodi aggiunti,...), la densità (che può variare molto di zona in zona), le risorse hardware limitate e la bassa disponibilità energetica, rendono complicato lo sviluppo di protocolli di routing.

Diffusione e raccolta dei dati

Il routing in una Wireless Sensor Network può essere scomposto in due fasi: diffusione dei dati e raccolta dati. Tipicamente, il processo di raccolta dati viene innescato dal verificarsi di un evento nell'ambiente o dall'applicazione: a questo punto i nodi effettuano le rilevazioni con i propri sensori, li diffondono eventualmente ad altri nodi per effettuare in-network processing (ad esempio aggregazione di dati) e li trasmettono a nodi intermedi, i quali faranno lo stesso finché questo processo di raccolta dati raggiunge il nodo sink. Questo schema di rete appena descritto viene definito *multi-hop*, poiché i nodi possono essere distanti più di un hop dal sink. L'approccio multihop è molto più conveniente rispetto al *single-hop*, sebbene possano esserci scenari in cui quest'ultimo è più indicato. In effetti, un approccio single-hop non consente una grande estensione geografica della rete e

inoltre richiede alte potenze trasmissive, non adatte ad una WSN; inoltre usando MAC di tipo contention-based, i contendenti al canale potrebbero essere numerosi, aumentando così il rischio di collisioni. In ambito multihop, invece, si possono facilmente raggiungere distanze molto lunghe, nonostante i nodi siano posizionati a poche decine di metri. Essendo così vicini tra loro, i nodi necessitano di una bassa potenza trasmissiva e la possibilità di contesa del canale con altri nodi è minore rispetto al caso singlehop.

Purtroppo però ci sono anche aspetti negativi: far sì che il pacchetto spedito da un nodo lontano molti hop dal sink arrivi integro e senza un eccessivo ritardo è una delle sfide che il routing deve affrontare. Il suo compito è proprio quello di trovare quei nodi intermedi, situati tra il nodo lontano diversi hop e il sink, che consentiranno al pacchetto di arrivare a destinazione, tenendo sempre presente i limiti di energia e di banda di una WSN.

Sfide progettuali del routing

Il routing in una Wireless Sensor Network si differenzia particolarmente dal routing di una qualsiasi rete cablata o wireless ad hoc proprio per le caratteristiche peculiari delle WSN:

- **Dinamicità e grandezza della rete:** come già accennato, la densità di una rete può variare molto di zona in zona, nuovi nodi possono entrare e altri possono uscire dalla rete in qualsiasi momento e i link tra nodi potrebbero cadere o degradarsi in seguito a variazioni ambientali. In tutti questi casi, una Sensor Network deve dimostrare la propria capacità di autoconfigurazione, cambiando alcuni parametri e compiendo decisioni in base alla situazione presente. Gli algoritmi di routing sono *self-configuring*, ovvero sono in grado di reagire a questi cambiamenti improvvisi.
- **Risorse limitate:** le risorse limitate sono quelle hardware ed energetiche. Per rientrare nel primo limite, serve che gli algoritmi abbiano una ridotta occupazione di memoria (footprint) e siano facilmente eseguibili da CPU poco potenti. Per quanto riguarda il secondo, il fatto di avere una rete multihop è già di per sé un costo energetico maggiore rispetto ad una a singlehop, poiché nel primo caso più nodi devono essere tenuti accesi, ricevere e inoltrare il pacchetto, mentre in un singlehop, solo due nodi sono coinvolti. Quindi, l'unico modo per diminuire questo costo, è creare dei percorsi di instradamento che bilancino fattori come il numero di nodi intermedi, le risorse energetiche residue di ogni nodo e la potenza trasmissiva richiesta.
- **Data model dell'applicazione:** i data model descrivono il flusso di informazioni tra nodi e sink. Questi modelli sono dipendenti dall'applicazione in rapporto a quali informazioni, come vengono usate e quanto spesso vengono richieste. Sono stati proposti vari modelli di raccolta dati, a seconda delle applicazioni. Alcuni di essi richiedono i dati periodicamente o in base al

verificarsi di certi eventi nell'ambiente. In altri, ogni nodo rileva, salva, elabora e aggrega i dati prima di inviarli al sink. In altri ancora, i dati vengono prelevati interattivamente tramite una comunicazione bidirezionale tra sink e nodi. Un protocollo di routing ideale dovrebbe essere ottimizzato per l'applicazione e al contempo essere in grado di supportare diversi data model, oltre a risultare affidabile, scalabile, efficiente e reattivo. Dal punto di vista pratico ciò non è facilmente realizzabile.

Strategie di routing

Il tradeoff che un protocollo di routing deve bilanciare è quello tra capacità di rispondere ai cambiamenti ed utilizzo efficiente delle risorse. In altre parole, serve il giusto compromesso tra risorse hardware e di banda limitate e overhead necessario per adattarli a queste.

L'overhead in una WSN viene misurato in termini di occupazione di banda, utilizzo energetico e risorse di calcolo. Per valutare se esistessero già protocolli in grado di soddisfare questo tradeoff, furono esaminati quelli adoperati nelle Ad Hoc Wireless Network. In queste reti si usano tre tipi di routing che si differenziano tra loro in base a come l'informazione è acquisita e mantenuta e al modo in cui essa viene utilizzata per calcolare i percorsi ottimali. La prima strategia è detta *routing proattivo* (o *table driven*) e consiste nel disseminare le informazioni di routing aggiornato a tutti i nodi. La rete può avere un'organizzazione *flat* o *gerarchica*. In caso sia flat, il routing sarà ottimale ma l'overhead in seguito ad un cambiamento nella rete sarà enorme. Il routing gerarchico invece è adatto per reti dalle grandi dimensioni.

La seconda strategia è detta di *routing reattivo* e calcola il percorso solo per alcuni nodi *on demand*. Questi protocolli non prevedono il mantenimento di un'informazione globale della rete, perciò, per scoprire nuovi cammini è necessario interrogare tutta la rete tramite un *flooding*, con le risposte che tornano indietro seguendo il percorso inverso. Infine l'ultima strategia è detta di *routing ibrido*, adatta per reti molto grandi. In questo caso, i nodi vengono uniti in gruppi, definiti cluster, adiacenti. All'interno di un cluster il routing è di tipo proattivo, mentre tra i cluster il routing è reattivo. Uno svantaggio di questa tecnica è che mantenere dei cluster provoca un ulteriore overhead. In definitiva, i metodi di routing delle ad hoc network non vanno bene per una WSN, poiché essi non sono adatti a situazioni di rete dinamiche o che crescono in dimensione, dato che overhead e traffico di rete sarebbero in continua crescita. Per questo motivo la ricerca scientifica ha prodotto diversi tecniche di routing, specifiche per le WSN.

1.5.4 Livello Trasporto

La natura data-centric delle WSN combinata con le risorse hardware limitate, rende il protocollo TCP inutilizzabile in questo ambito. Inoltre, le reti di sensori sono caratterizzate da un concetto di affidabilità differente dalle reti classiche. Per di più, i livelli stessi di affidabilità o di controllo di congestione possono

dipendere dal tipo dei dati trasportati. Quindi, le funzionalità di questo livello devono essere progettate trovando il giusto compromesso tra consumo energetico e politiche implementate. Queste ultime sono però dipendenti anche dal percorso seguito dai pacchetti: *forward path* (dai nodi verso il sink) o *reverse path* (dal sink verso i nodi). Nel *forward path* si applica un principio di *event-reliability*, ovvero non ha importanza che tutti i pacchetti arrivino a destinazione, ma serve che arrivino almeno quelli necessari per un corretto monitoraggio dell'evento. Al contrario, nel percorso inverso (*reverse path*), è richiesto che tutti i messaggi spediti dal sink giungano ai nodi destinatari. Questo perché i messaggi mandati dal sink contengono informazioni critiche per il controllo dell'attività del nodo, come query dei dati o istruzioni di programmazione. Quindi, in questo caso, sono necessarie delle regole di consegna dei pacchetti più severe.

1.5.5 Livello Applicativo

Protocolli di questo tipo per una generica WSN non sono ancora diffusi, sebbene in passato ne siano stati proposti alcuni, tra cui Sensor Management Protocol (SMP), Task Assignment e Data Advertisement Protocol (TADAP) e Sensor Query Data Dissemination (SQDDP). SMP fornisce operazioni software per compiti di amministrazione della rete, come data aggregation, clustering, sincronizzazione, localizzazione, accensione sensori, monitoraggio e riconfigurazione dei nodi, autenticazione e distribuzione di chiavi nelle comunicazioni sicure. TADAP gestisce l'assegnazione di 'interessi' riguardo un attributo o un evento e la pubblicizzazione dei dati disponibili. SQDDP fornisce all'utente le interfacce per avviare e rispondere a query e si occupa di assegnare identificativi ai nodi in base ad attributi o alla loro locazione. Il discorso è diverso per le reti ZigBee. Infatti questo standard definisce anche il livello applicativo. Nel capitolo successivo saranno illustrati nel dettaglio le caratteristiche dello Stack ZigBee.

1.6 Sviluppi futuri

Dal punto di vista hardware, l'evoluzione tecnologica consentirà lo sviluppo di microcontrollori sempre più piccoli, efficienti e performanti, che integrano microcontrollore e radio in una piattaforma di dimensioni sempre più ridotte (la figura 1.7 mostra un dispositivo microcontrollore/RF chiamato *Spec* delle dimensioni di una punta di una penna [5]). Anche i sensori subiranno miglioramenti non trascurabili: avranno dimensioni sempre più piccole grazie alle tecnologie MEMS (Figura 1.8) e NEMS, saranno più accurati, meno costosi, più efficienti, affidabili, in grado di sopportare condizioni di utilizzo estreme e potranno rilevare nuove sostanze chimiche, individuando quelle tossiche, esplosivi e agenti biologici.

Da una prospettiva software sarà necessaria l'adozione di standard di comunicazione tra sensori e lo sviluppo di sistemi di *service discovery*, in modo che si possano creare dialoghi tra nodi di diverse reti. Grazie a questi fattori, le WSN diventeranno sempre più diffuse in qualsiasi contesto, in una moltitudine di ap-

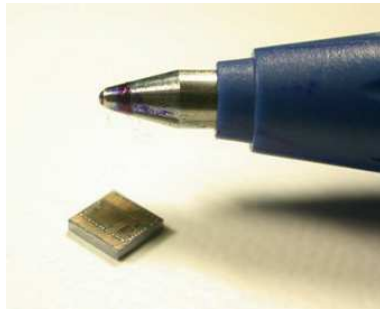


Figura 1.7: Dimensioni di una penna e di uno Spec a confronto

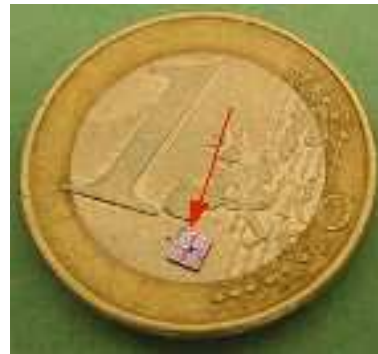


Figura 1.8: Dimensioni di un sensore di pressione MEMS

plicazioni: in ambito domestico, i sensori si troveranno in molti elettrodomestici e dispositivi, compresi quelli di illuminazione e riscaldamento; in ambito biomedico sarà possibile monitorare diversi parametri corporei in tempo reale; in ambito militare i sensori saranno disseminati sui campi di battaglia, sui veicoli e sugli equipaggiamenti delle truppe; in ambito industriale e commerciale si assisterà alla rivoluzione dell'agricoltura di precisione, al tracking dell'inventario e delle catene di montaggio, al monitoraggio degli impianti produttivi. Nell'ambito dei trasporti, le strade saranno delle lunghe reti di sensori per segnalare in tempo reale traffico o condizioni di pericolo e le automobili potranno essere dotate di nuovi meccanismi di prevenzione degli incidenti e dei furti; nell'ambito della robotica, sarà possibile creare microrobot autonomi e in grado di comunicare tra loro.

Tutta questa innovazione però solleva problemi legali e di sicurezza. Infatti serviranno nuove leggi per regolamentare l'utilizzo di queste tecnologie in situazioni particolari di privacy e sarà necessario implementare meccanismi di sicurezza per evitare il tampering dei dati, ovvero la modifica volontaria dei dati rilevati dai sensori, piuttosto che la lettura di dati personali (ad esempio dei propri sensori biologici) o la diffusione di virus nella rete. In definitiva, il potenziale per un'importante rivoluzione della nostra vita quotidiana c'è, ma il lavoro da affrontare è ancora molto.

Capitolo 2

Monitoraggio Ambientale

2.1 Obiettivi del monitoraggio

Monitorare un ambiente significa raccogliere, tramite sensori, informazioni in un'area, salvare e/o visualizzare queste informazioni ed eventualmente passarle ad altri sistemi per analizzarle. Quali debbano essere le informazioni da prelevare dall'ambiente dipende dagli obiettivi. Il sistema tipico di monitoraggio ambientale tramite WSN, è costituito dai nodi, solitamente dotati di sensori di luminosità, temperatura e umidità, distribuiti nell'area interessata. Tali nodi mandano i dati rilevati ad un nodo coordinatore o gateway, il quale è connesso ad un host (un miniPC) atto alla memorizzazione e alla gestione dei dati. Il risultato finale è che l'utente (o un sistema) può consultare (ad esempio tramite browser), anche in tempo reale, le informazioni di un'area remota e agire di conseguenza.

2.2 Monitoraggio indoor e concetto di Comfort

L'espressione ambiente '*indoor*' (o ambiente confinato) è riferita agli ambienti confinati di vita e di lavoro non industriali, ed in particolare, a quelli adibiti a dimora, svago, lavoro e trasporto. Secondo questo criterio, il termine *ambiente indoor* comprende: le abitazioni, gli uffici pubblici e privati, le strutture comunitarie (ospedali, scuole, caserme, alberghi, banche. etc.), i locali destinati ad attività ricreative e/o sociali (cinema, bar, ristoranti, etc.) ed infine i mezzi di trasporto pubblici e/o privati (auto, treno, etc.). Nelle società moderne, la popolazione trascorre una parte molto rilevante del proprio tempo negli ambienti confinati.

In questi ultimi decenni sono stati condotti studi più puntuali della qualità dell'aria indoor (*IAQ*) documentando profondi mutamenti quali-quantitativi della stessa, con un progressivo aumento in assoluto delle sostanze inquinanti. Anche in Italia, come conseguenza delle varie crisi energetiche, si sono imposti nuovi criteri tecnico-progettuali per gli edifici ad uso civile. La necessità di contenere i consumi per il riscaldamento e per la ventilazione ha imposto un migliore isolamento degli edifici, con conseguente spinta a sigillare gli ambienti interni ed a sostituire le modalità naturali di aerazione ed illuminazione con mezzi artificiali e

con tecnologia sempre più sviluppata. Alle trasformazioni strutturali degli edifici si sono accompagnate modifiche rilevanti degli arredi (nuovi materiali per mobili, rivestimenti; ecc.) e degli strumenti di lavoro (crescente impiego di fotocopiatrici, videoterminali, stampanti, ecc.).

L'aria interna è ritenuta un importante fattore ambientale da più di un centinaio di anni, ovvero dall'inizio della rivoluzione sull'igiene, avvenuta intorno al 1850, per poi essere approfondita negli studi ambientali, per divenire tematica dominante intorno agli anni '60. L'IAQ svolge un ruolo fondamentale per quanto riguarda la salute pubblica.

Per migliorare la qualità dell'aria sono oggi possibili varie strategie: controllo della fonte dell'aria, pulizia dell'aria (tramite assorbimento e fotocatalisi) e utilizzo della *ventilazione personalizzata*. Quest'ultima tipologia di intervento consiste nel fornire a ciascun occupante aria non inquinata dalle fonti presenti nell'ambiente [6].

2.2.1 Comfort

Il comfort microclimatico è fondamentale in tutti gli ambienti di lavoro e nei luoghi aperti al pubblico. Per ottenere situazioni di benessere microclimatico occorre garantire condizioni accettabili sia dal punto di vista delle grandezze termo-igro-anemometriche che caratterizzano il microclima, sia del livello di illuminazione, sia della qualità dell'aria.

Un microclima confortevole è quello che suscita nella maggioranza degli individui una sensazione di soddisfazione per l'ambiente, identificata col termine 'benessere termoigrometrico' o, semplicemente, 'comfort'. I fattori microclimatici negli ambienti di lavoro unitamente all'intensità dell'impegno fisico svolto, e all'abbigliamento condizionano nell'uomo una serie di risposte biologiche graduate che vanno da sensazioni di benessere termoigrometrico a sensazioni di disagio (discomfort) a vero e proprio impegno termoregolatorio (sudorazione più o meno accentuata), a sindromi patologiche (stress da calore). Un rischio microclima, quale il discomfort, nasce tanto dalla percezione globale del corpo umano quanto da situazioni di disagio localizzate e può essere essenzialmente ricondotto a sensazioni di caldo, di freddo, di eccessive correnti d'aria o sbalzi termici.

Studi sulle esposizioni in ambienti indoor e sugli effetti sulla salute sono stati condotti principalmente in Nord Europa e in Nord America. È emerso un forte legame tra IAQ e cancro ai polmoni, allergie e altre reazioni all'ipersensibilità (comprese le Sick Building Syndrome (*SBS*), e la Sensibilità Chimica Multipla (*MCS*)), e le infezioni respiratorie (*ARI*). Le allergie sono in rapido aumento in tutto il mondo, sia nei paesi sviluppati che in quelli in via di sviluppo, e sono fortemente connesse con le esposizioni in ambienti interni; grande interesse è ora posto sulla relazione fra esposizione agli ambienti e aumento dei sintomi. In molte regioni del mondo, circa la metà della popolazione è colpita, i giovani maggiormente degli anziani, e in molte regioni, l'incidenza è praticamente raddoppiata negli ultimi decenni [7].

PMV	PPD %	Valutazione Ambiente Termico
+3	100	Molto caldo
+2	75.7	Caldo
+1	26.4	Leggermente caldo
$-0.5 < \text{PMV} < +0.5$	< 10	Accettabilità termica
-1	26.8	Fresco
-2	76.4	Freddo
-3	100	Molto freddo

Tabella 2.1: Tabella valori PMV e PPD

Il comfort termico viene definito dalla *ASHRAE* (*American Society of Heating, Refrigerating and Air Conditioning Engineers INC*) come una condizione di benessere psicofisico dell'individuo rispetto all'ambiente in cui vive e opera. La valutazione di tale stato soggettivo può essere oggettivata e quantificata mediante l'utilizzo di indici integrati che tengono conto sia dei parametri microclimatici ambientali (T_a , T_r , V_a , RH), sia del dispendio energetico (dispendio metabolico MET) connesso all'attività lavorativa, sia della tipologia di abbigliamento (isolamento termico CLO) comunemente utilizzato.

Tra gli indici quello che con maggiore precisione rispecchia l'influenza delle variabili fisiche e fisiologiche sopracitate sul comfort termico è il **PMV** (*Predicted Mean Vote*). Esso deriva dall'equazione del bilancio termico il cui risultato viene rapportato ad una scala di benessere psicofisico ed esprime il parere medio (voto medio previsto) sulle sensazioni termiche di un campione di soggetti allocati nel medesimo ambiente.

Dal PMV è derivato un secondo indice denominato **PPD** (*Predicted Percentage of Dissatisfied*) che quantifica percentualmente i soggetti comunque 'insoddisfatti' in rapporto a determinate condizioni microclimatiche. Il PMV risulta un indice particolarmente adatto alla valutazione di ambienti lavorativi a microclima moderato, quali abitazioni, scuole, uffici, laboratori di ricerca, ospedali, ecc; esso è utile nel rilevare anche limitati gradi di disagio termico nei residenti in tali ambienti. Lo stato di comfort termico si raggiunge per valori di PMV compresi tra $+0,5$ e $-0,5$, cui corrisponde una percentuale di insoddisfatti delle condizioni termiche (PPD) inferiore al 10%.

La tabella 2.1 rappresenta i valori dei coefficienti PMV e PPD in base alle condizioni termiche dell'ambiente [8].

In aggiunta al miglioramento della qualità dell'aria, un monitoraggio indoor permette un notevole **risparmio energetico**. Infatti, la corretta e precisa rilevazione dei parametri di un impianto (es. valori di soglia di temperatura, umidità, pressione, . . .), il controllo della regolazione degli stessi in funzione delle stagionalità, delle fasce orarie e della presenza di personale, le logiche di funzionamento appropriate nello spegnimento ed avviamento impianti, oltre a garantire le condizioni di benessere desiderate consentono di ottimizzare il consumo energetico. Il controllo della qualità dell'aria, la segnalazione della presenza di gas quali il biossido o il monossido di carbonio forniscono indicazioni precise per una migliore

ventilazione con un risparmio di energia. E' possibile persino tarare la ventilazione sul numero probabile di personale all'interno dell'edificio basandosi sui livelli di CO2 all'interno di specifiche stanze dell'edificio.

2.3 Concetto di 'Occupancy'

Lo scopo principale di questo lavoro di tesi, come detto precedentemente, è riguardante la progettazione di un sistema di monitoraggio ambientale indoor tramite WSN; in aggiunta alla progettazione, si è cercato di adempiere ad un obiettivo che non è mai stato approfondito a fondo, cioè di verificare se tramite i dati raccolti di temperatura, umidità, luce e CO2 fosse possibile stabilire la presenza o meno di persone all'interno di un ambiente confinato (ottenendo una stima della loro quantità). La presenza di persone prende dall'inglese il termine di '*occupancy*'.

Tale risultato è già stato ottenuto senza l'utilizzo di questi sensori ma mediante l'utilizzo di tecnologie completamente diverse e più costose. Nei prossimi capitoli verranno analizzate nel dettaglio tali tecnologie che raggiungono questo scopo.

2.3.1 Tecnologie per l' occupancy sul mercato

Di seguito sono brevemente illustrate le principali tecnologie presenti sul mercato.

Tecnologia IR

I sensori PIR (Passive InfraRed) sono i sensori di uso più comune e sono in grado di rilevare il 'calore' emesso dagli occupanti della stanza che stanno monitorando. La rilevazione si ha nel momento in cui si verifica un cambiamento nei livelli di raggi infrarossi. I sensori PIR riescono ad ottenere buone prestazioni di rilevazione.

Possiedono un raggio di azione mediamente di qualche metro (4-5 m) principalmente per due motivi:

- Il pattern di monitoraggio presenta delle 'zone morte' che con l'aumentare della distanza diventano via via più ampie.
- Visto che il sensore è di tipo passivo, la sua sensibilità deriva principalmente dalla quantità di calore emanata dal corpo stesso presente nel suo campo di azione. In particolare, il calore umano rilevato decresce con il quadrato della distanza, ciò comporta una notevole diminuzione del range del campo di azione [10].

Un tipico sensori a infrarossi passivo di tipo commerciale è mostrato in figura 2.1.



Figura 2.1: Esempio di rilevatore di presenza da parete a tecnologia PIR (Passive InfraRed)

Tecnologia a ultrasuoni

Tale tecnologia permette la rilevazione della presenza di persone sfruttando il principio Doppler. In particolare, viene emesso un segnale ad alta frequenza ad ultrasuoni per tutto lo spazio, la frequenza del segnale ricevuto in riflessione viene confrontata con quella del segnale originariamente emesso; una variazione di frequenza viene quindi interpretata come presenza di un oggetto in movimento nello spazio. A differenza dei sensori di tipo PIR, questi rilevano il movimento e non la presenza statica di una persona; ciò comporta una serie di vantaggi e svantaggi che rendono tali sensori più adatti per gli spazi aperti, spazi con ostacoli, servizi igienici e spazi con superfici dure [11].

Tecnologia a onde acustiche

I sensori acustici sono dispositivi passivi che rilevano livelli di pressione delle onde sonore in uno spazio. In ambito industriale e commerciale, tali sensori sono spesso utilizzati in combinazione con sensori di altra tecnologia (e.g PIR) in modo da ridurre notevolmente il rischio di 'falso allarme' [12].

La figura 2.2 mostra i tre tipi di sensori appena citati.

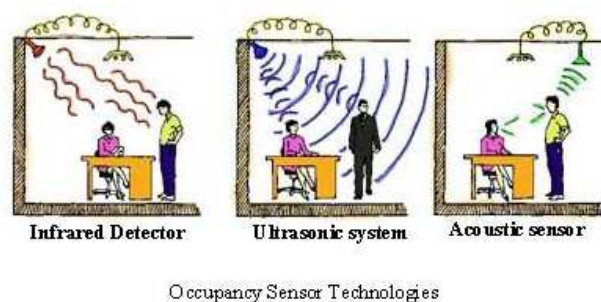


Figura 2.2: Tecnologie per l'occupancy a confronto

2.4 Monitoraggio tramite CO2

L'**anidride carbonica** (nota anche come *biossido di carbonio* o *CO2*), è un gas presente nella miscela d'aria piuttosto stabile. Tale grandezza viene misurata principalmente per il controllo della ventilazione in ambienti confinati (stanze, uffici, aule, ecc...). A seconda delle varie normative e standard, può essere richiesto un tasso di ventilazione di $25.5m^3/h$, oppure una ventilazione continua durante tutto il periodo nel quale il locale è occupato, oppure può essere richiesto una soglia massima di 1000 ppm (parti per milione) di CO2 [15].

La respirazione di una persona genera CO2 e consuma ossigeno in una quantità variabile a seconda di certi parametri quali: grandezza del corpo e livello di attività fisica che la persona sta eseguendo. La relazione tra livello di attività fisica e el tasso di generazione di anidride carbonica è presentato nell'ASHRAE Fundamentals Handbook (ASHRAE 1997). Il tasso di consumo di ossigeno di una persona V_{O_2} , in L/s è dato dalla seguente equazione [17]:

$$V_{O_2} = \frac{0.00276A_D M}{(0.23RQ + 0.77)} \quad (2.1)$$

dove:

- RQ è il quoziente di respirazione, cioè il tasso relativo volumetrico di CO2 prodotto dal consumo di ossigeno.
- M è invece il livello di attività fisica e si misura in mets ($1 \text{ met} = 58.2W/m^2$)
- A_D rappresenta l'area di superficie *DuBois* in m^2 che, considerando H l'altezza del corpo e W la massa in Kg, può essere stimata con l'equazione

$$A_D = 0.203H^{0.725}W^{0.425}$$

Maggiori informazioni sono disponibili in [18]. Il valore del tasso di generazione di anidride carbonica è ottenuto moltiplicando il valore di ossigeno risultato in eq. 2.1 con RQ .

Tali sensori son parte di un sistema di tipo DVC ('Demand-Controlled Ventilation') cioè permettono di utilizzare una strategia che migliora l'efficienza di energia dell'ambiente in cui vengono utilizzati assicurando una qualità dell'aria (IAQ) coerente con le esigenze corporee umane. Il risparmio energetico è conseguente alla limitazione della ventilazione, infatti, i sensori di CO2 forniscono informazioni per l'apertura del dispositivo di ventilazione accendendo il motore di areazione indipendentemente dal controllo sulla temperatura solamente quando le condizioni ambientali dell'aria all'interno della stanza lo richiede.

Di seguito è riportata una tabella (2.2) che mostra per varie concentrazioni di CO2 le '*sensazioni*' provate dal corpo umano.

Concentrazione CO2	Dettagli
$\sim 40000ppm$	Concentrazione del respiro umano (20 l CO2/h)
$5000ppm$	Limite di concentrazione per ambienti di lavoro
$> 1000ppm$	Riduzione di concentrazione ed affaticamento
$1000ppm$	Livello standard per ambienti indoor
$400ppm$	Livello tipico di CO2 nell'atmosfera (ambiente esterno)

Tabella 2.2: Tabella relativa alle varie concentrazioni di CO2

Un buon sistema di ventilazione dovrebbe essere in grado di gestire i flussi di aria in base ai valori di CO2 come segue:

- quando le letture di CO2 sono al di sopra delle soglie previste, il sistema di ventilazione permette il ricircolo d'aria dall'ambiente esterno riducendo i livelli di CO2 e migliorando il IAQ, al contrario
- se la concentrazione di CO2 è molto inferiore al livello di CO2 ottimale, la ventilazione è ridotta mantenendo un appropriato IAQ.

La figura 2.3 mostra un esempio di gestione della ventilazione in un palazzo con percentuale di occupancy diversa di piano in piano [16].

Il rapporto che intercorre tra anidride carbonica e tasso di ventilazione è stato ben studiato negli anni passati [19]. Naturalmente, a parità di altre condizioni, se il tasso di ventilazione in un ambiente confinato diminuisce, la concentrazione di anidride carbonica aumenta e viceversa.

Da [17], il modello che rappresenta la relazione tra CO2 e ventilazione include un sistema di concentrazioni di anidride carbonica a stati stazionari ad un certo tasso di costante cambiamento d'aria ed il tempo necessario per raggiungere queste condizioni stazionarie. Le concentrazioni di CO2 allo stato stazionario possono essere determinate per un dato tasso di ventilazione in una 'singlezone' tramite bilancio di massa.

Assumendo che la concentrazione di CO2 in un certo locale possa essere caratterizzata da un singolo valore C , il bilancio di massa può essere espresso come:

$$V \frac{dC}{dt} = G + QC_{out} - QC \quad (2.2)$$

dove:

- V = volume del locale in esame m^3 (mg)
- C = concentrazione di CO2 indoor mg/m^3 ($ppm(v)$)
- C_{out} = concentrazione di CO2 outdoor mg/m^3 ($ppm(v)$)
- t = tempo in s
- G = tasso di generazione indoor mg/s (m^3/s)

- Q = tasso di ventilazione del locale mg/s (m^3/s)

per un certo tasso di generazione (oppure livello di occupancy) e un costante tasso di ventilazione e di concentrazione outdoor, la concentrazione indoor raggiungerà un valore stazionario di equilibrio C_{ss} dato dalla relazione

$$C_{ss} = C_{out} + \frac{1.8 \times 10^6 G}{Q}$$

con G e Q espressi in L/s ; cosicchè come il tasso di ventilazione aumenta, la concentrazione all'equilibrio diminuisce.

Assunto che un certo locale (si consideri da ora in poi un ufficio) presenti all'inizio della giornata lavorativa una concentrazione di anidride carbonica pari a quella esterna, la concentrazione di CO2 indoor inizierà via via ad aumentare di un tasso dipendente a quello di ventilazione Q diviso il volume del locale stesso V . Questa quantità $\frac{Q}{V}$ può essere vista come un tasso di cambiamento dell'aria esterna del locale mentre il suo inverso $\frac{V}{Q}$ rappresenta la costante di tempo del sistema. Durante il periodo di tempo in cui l'ufficio è occupato dai dipendenti, la concentrazione indoor di CO2 è governata dalla soluzione dell'equazione 2.2 con una soluzione del tipo:

$$C(t) = C_{out} + \frac{G}{Q} \left(1 - e^{-\frac{Qt}{V}}\right) \quad (2.3)$$

si noti che con t molto alto, il valore di C si stabilizza a quello statico calcolato precedentemente. Il tempo necessario perchè il sistema si stabilizzi dipende dal tasso Q/V in quanto con alti valori (ricircolo d'aria alto) si hanno tempi inferiori per raggiungere il valore stazionario di CO2.

2.4.1 State of the art

Il mercato offre una discreta gamma di sensori per la rilevazione di CO2, con principi di funzionamento piuttosto differenti; in particolare, vi sono sensori a semiconduttore allo stato solido e sensori con principio di funzionamento a infrarossi.

Stato dell'arte nella rilevazione di CO2 mediante semiconduttori a stato solido

H.Endres [13] sviluppò un sistema a sensore di CO2 capacitivo con una calibrazione numerica contro l'interferenza provocata dall'umidità. Tale sistema consiste in un condensatore riscaldabile inter-digitale (IDC) con un ricoprimento che lavora come indicatore di CO2 e sensore di umidità capacitivo. Il range di misurazione ottenuto va da 100 a 3000 ppm (parti per milione) con una temperatura ottimale di lavoro di 60°C.

P.Keller [14] ha sviluppato dei sensori allo stato solido basati su stradi di BaTiO3 nanocristallino depositato mediante tecniche di epitassia su un substrato di allumina. Tale sensore lavora alla temperatura di 200-600 °C e sono stati testati con concentrazioni di CO2 di 100-105 ppm.

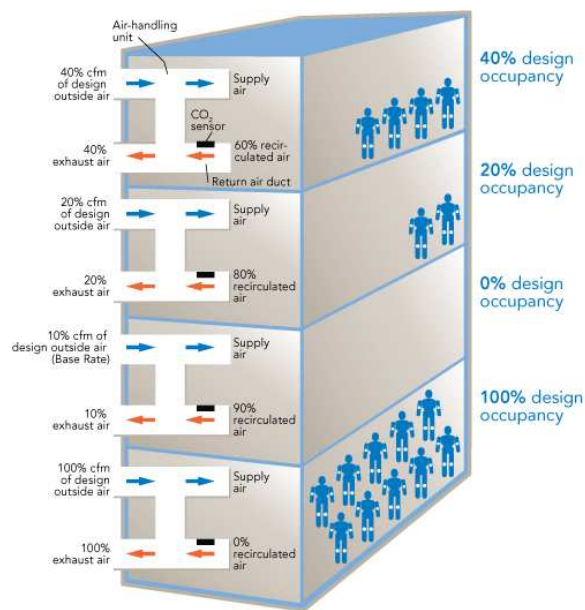


Figura 2.3: Esempio di uno stabile di uffici con gestione dei flussi di ricambio dell'aria proporzionali alla percentuale di occupazione di ogni ambiente

Stato dell'arte nella rilevazione di CO₂ a tecnologia IR

Un'altra tecnologia usata nel mondo del mercato per i sensori di CO₂ è appunto quella che sfrutta il principio di assorbimento ai raggi infrarossi. In particolare, proprio tale tipologia di sensore è stata utilizzata per la rilevazione dell'anidride carbonica in questo lavoro di tesi.

Esistono principalmente due metodi per misurare la concentrazione di CO₂ mediante assorbimento IR, esistono:

- sensori ad assorbimento IR non-dispersivo (NDIR)
- sensori foto-acustici

Il sensore preso in considerazione per il progetto utilizza la prima tecnologia (NDIR) che comparata ai sensori foto-acustici offre i seguenti vantaggi.

- minor sensibilità a variazioni di pressione
- minor sensibilità a vibrazione e interferenza acustica
- design più compatto

Il sensore utilizzato usa un doppio principio di *detection* sfruttando una doppia sorgente a raggio infrarosso per la rilevazione di una certa lunghezza d'onda della luce infrarossa.

Il dispositivo utilizza una procedura di autocalibrazione, che sfrutta una diversa periodicità nella rilevazione della luce emessa dalle due sorgenti sopra citate; in particolare, come si può vedere dalla figura 2.4, una sorgente IR opera per

misurare la concentrazione di CO₂ e genera un segnale IR ogni 30 secondi. La seconda sorgente, invece, è usata come sorgente di riferimento per permettere l'auto-calibrazione. Quest'ultima sorgente è attivata ogni 24 ore e, grazie ad un confronto dei valori precedenti ed una modifica di un offset, permette di ottenere un dispositivo che fornisca delle misurazioni insensibili all'invecchiamento del dispositivo stesso.

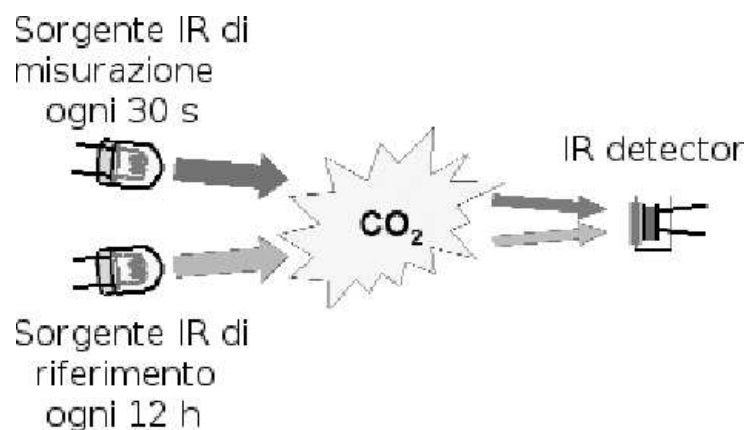


Figura 2.4: Rappresentazione del principio di funzionamento del sensore CO₂ a tecnologia NDIR

Altre caratteristiche tecniche di questo sensore sono analizzate nel corso dei capitoli successivi.

2.5 Monitoraggio in Inthebra s.r.l.

Inthebra s.r.l. è un'azienda padovana nella quale è stato svolto il progetto di tesi qui documentato. In tale ambiente sono state effettuate numerosi test e prove di vario tipo per quel che concerne la verifica del conseguimento dell'obiettivo finale; in particolare, i vari ambienti sono stati monitorati al fine di verificare il corretto funzionamento del sistema finito.

In appendice A è riportata la planimetria dell'ambiente completa di tutte le informazioni riguardanti posizionamento dei sensori, porta finestre ecc. . . mentre di seguito (Figura 2.5) è riportato uno schema più essenziale delle varie stanze ed uffici dell'azienda.

Come si può vedere dalla figura, l'infrastruttura aziendale è composta principalmente da sette ambienti confinati (A,B, . . . ,G) nei quali sono stati svolti i test per l'acquisizione dei dati ambientali. In particolare, la tabella 2.3¹ contiene alcu-

¹I valori dei campi della tabella sono piuttosto approssimativi, sono stati inseriti solamente per dare un'indicazione generica degli spazi più o meno frequentati dai dipendenti aziendali. In particolare, il campo *Numero di persone* stima il numero di dipendenti all'interno della stanza nel periodo in cui è frequentata mentre il campo *Occupazione media giornaliera %* rappresenta la percentuale per la quale l'ambiente è occupato da almeno una persona all'interno della giornata lavorativa (dalle 9:00 alle 18:30 circa)

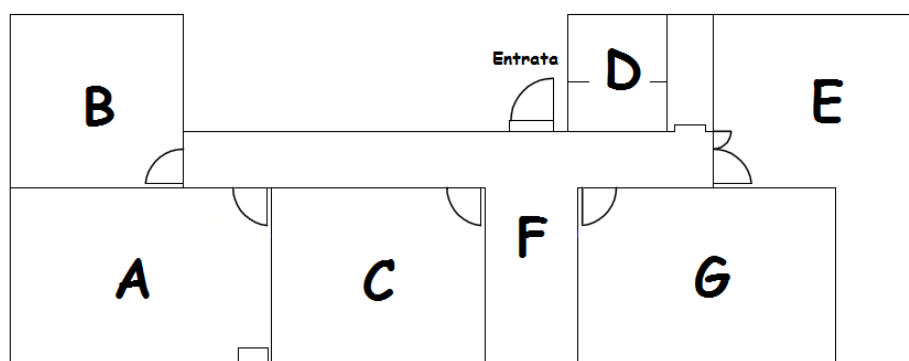


Figura 2.5: Planimetria del campo di test

ni dettagli sull'occupazione media di ognuno di questi ambienti grazie alla quale il lettore può cogliere al meglio i risultati presentati successivamente.

2.6 Risultati ottenuti

Vari nodi sono stati posti all'interno dell'infrastruttura, uno per ogni stanza. Il sistema prevede un periodo di campionamento variabile a scelta del client; nelle prove per le quali saranno mostrati i prossimi risultati, è stato adoperato un periodo di campionamento di 10 secondi. Come risultato finale, si possono ottenere dei grafici che mostrano l'andamento di temperatura, umidità, luce e CO₂ all'interno di ogni stanza per un certo periodo di tempo.

Un periodo di campionamento di 10 secondi è stato utilizzato per riuscire a cogliere le variazioni ambientali più rapide e brusche, soprattutto per quel che riguarda il campionamento dei valori di luce. In questo modo è possibile andare a rilevare con più precisione la presenza di persone in luoghi dove ci sono forti variazioni di luce (per esempio, se veniva impostato un periodo di campionamento maggiore non si poteva riscontrare la presenza di un dipendente che si lava le mani in bagno o un altro che accende la luce in corridoio solo per il tempo necessario a percorrerlo).

Quindi, considerando 6 campioni per ogni minuto, gli n dati raccolti nell'arco di una giornata di 24 ore da un singolo nodo sono al massimo:

$$n = 24h * 60m * 6 = 8640.$$

Questo rappresenta il valore massimo perchè è stato visto che alcuni campioni non riescono a essere immagazzinati per problemi relativi al traffico di rete e all'occupazione di risorse del database.

Si inizi ora ad analizzare i dati raccolti dai sensori posizionati in azienda, suddivisi per locazione. Si vuol premettere che i dati successivi son stati raccolti

	Numero di persone	Occupazione media giornaliera (%)	Descrizione ambiente
A	3-4	80-100%	Ufficio contenente stampanti e 4 PC sempre accesi e ampie vetrate
B	1-2	80-100%	Ufficio di dimensione ridotta con 2 PC accesi
C	1	10-30%	Zona adibita a sala riunioni, ampie vetrate
D	1	5-10%	Bagno cieco (senza fonti naturali di luce)
E	1-2	10-20%	Zona adibita a laboratorio, ampie vetrate rivolte a EST
F	1-2	5-10%	Zona di ingresso e corridoio
G	1	5-10%	Ufficio di un'altra azienda (Non sono stati svolti test in questo ambiente)

Tabella 2.3: Contiene alcune informazione relative all'occupazione dell'ambiente di test

per un periodo temporale di una giornata intera (24 ore). Nei capitoli successivi, si svolgerà un'analisi più dettagliata dei dati raccolti per meglio scoprire le potenzialità di questo sistema utilizzato per stabilire l'occupancy di un certo luogo.

2.6.1 Locale A

La figura 2.6 mostra l'andamento delle grandezze fisiche raccolte (temperatura, umidità, e CO₂) durante 24 h di monitoraggio ². Dal grafico è possibile notare un progressivo aumento durante l'arco della mattinata sia dei valori di temperatura che di anidride carbonica. In tale periodo l'ambiente è stato occupato da circa 3/4 persone. Nel corso della mattinata sono riscontrabili nel profilo della CO₂ due picchi verso il basso, tali sono dovuti all'apertura della porta del locale verso il corridoio.

Un notevole sbalzo delle grandezze in gioco si ha dalle 13:15 alle 14:15 circa, periodo nel quale i locali aziendali vengono lasciati per la pausa pranzo. In questo frangente, le finestre vengono aperte per permettere un ricircolo d'aria con l'esterno; quello che ne consegue è una diminuzione di temperatura da 23°C a 20°C (il test è stato svolto con temperatura esterna di 10°C circa), una diminuzione dei livelli di anidride carbonica da 1120 a 437 ppm (vedi label) ed una brusca variazione, seppur contenuta, dell'umidità.

Con il rientro del personale nei luoghi di lavoro, le finestre sono state chiuse; i livelli iniziano a salire con pendenza e con valori comparabili con quelli della prima mattinata.

Verso le ore 19:00 si nota un progressivo calo dei livelli di CO₂ e temperatura, in corrispondenza del termine della giornata lavorativa, che progressivamente si portano verso i valori riscontrati nel primo mattino.

2.6.2 Locale B

Qui viene di seguito riportato il grafico relativo al monitoraggio dell'ufficio B (Fig. 2.7). Il grafico mostra i valori delle grandezze raccolte per una finestra temporale di 8 ore. In questa analisi, l'ambiente sottoposto a test era occupato da 2 persone. Si vuol precisare che, l'ufficio in questione è quello più di tutti sottoposto a 'via vai' di persone e quindi la porta che lo collega all'ambiente più ampio del corridoio è spesso aperta. Ciò genera molto spesso un cambio d'aria abbastanza importante tra ufficio (B) e corridoio (F).

Dal grafico è possibile notare una serie di aspetti piuttosto interessanti. In particolare si nota un incremento piuttosto costante della temperatura della stanza durante tutto l'arco della giornata lavorativa, approssimativamente da 17°C fino a temperature più confortevoli di 23 °C grazie solo ad un piccolo contributo dovuto all'impianto di riscaldamento ma soprattutto dalle ampie vetrate del locale che permettono di sfruttare al meglio il calore dei raggi solari.

²I dati raccolti relativi alla luce sono stati omessi dal grafico sia perchè di interesse ridotto, sia per semplificare la lettura del grafico

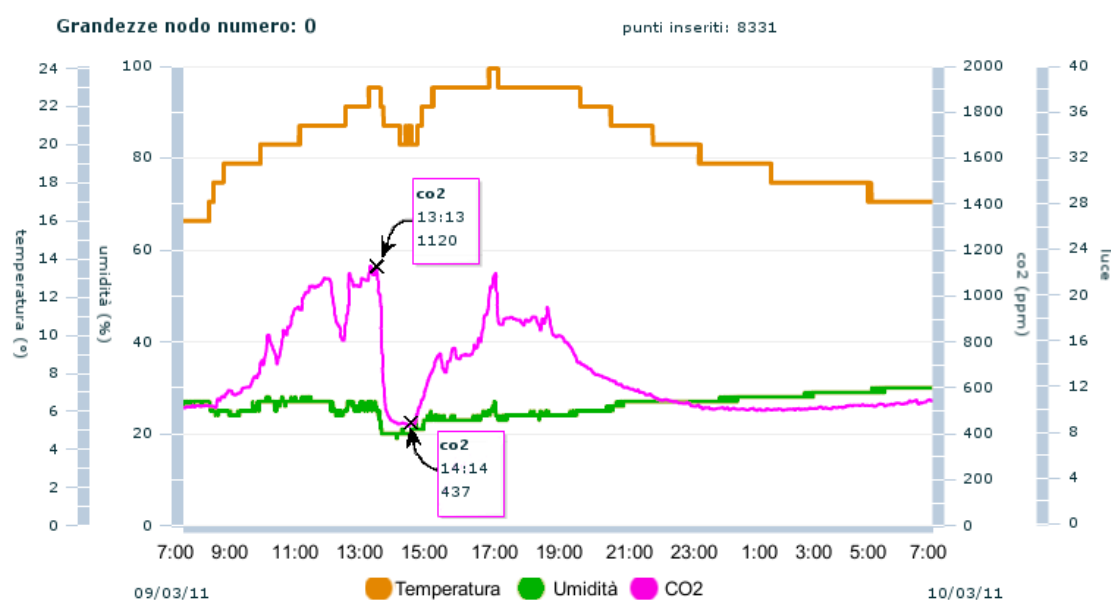


Figura 2.6: Grafico del monitoraggio nel locale A

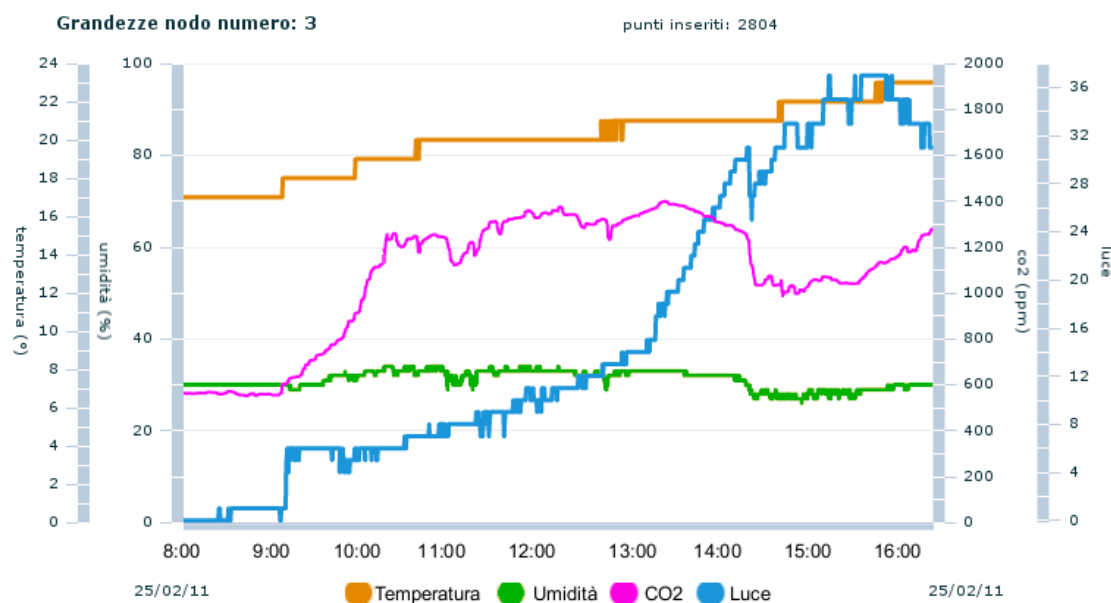


Figura 2.7: Grafico locale B

Un progressivo aumento dei valori di luce si ha a partire dalle ore 8:00 del mattino fino alle 16:00 circa; il gradino riscontrabile alle ore 9:00 è dovuto all'arrivo degli occupanti dell'ufficio che hanno acceso le luci.

Per quel che riguarda l'andamento dell'umidità si riscontra una notevole stabilità dei valori finché l'ambiente non viene occupato; solo successivamente, è riscontrabile una continua variazione dei valori con un sensibile incremento durante il periodo in cui la stanza è occupata ed una leggera diminuzione quando

l'ufficio viene lasciato.

In questo caso, l'andamento della CO₂ fornisce un'importante informazione per quel che riguarda l'occupancy. Si nota fin dall'inizio della giornata lavorativa un progressivo aumento dei valori di anidride carbonica iniziato proprio in corrispondenza del picco di luce dovuto all'accensione della luce; fin tale istante, il livello di CO₂ è costante a 550 ppm. L'aumento durante la mattinata presenta un culmine attorno ai 1400 ppm alle 13:00 circa; dalle ore 10:30 in poi fino a tale momento i valori di CO₂ sono stabili ad un valore medio di 1300 ppm pur presentando notevoli oscillazioni. Tale effetto è dovuto semplicemente allo scambio d'aria con il corridoio come precedentemente accennato.

Dalle 13:00 alle 14:30 circa, si nota una netta diminuzione della concentrazione di CO₂, tale effetto è dovuto al fatto che gli occupanti hanno lasciato il locale per la pausa pranzo.

La fase di discesa è suddivisa in due momenti, dalle 13:10 alle 14:15 si ha la diminuzione della CO₂ con una certa pendenza mentre dalle 14:15 alle 14:25 si nota una diminuzione ancora più brusca. Durante il primo intervallo temporale, la porta è rimasta chiusa e l'ambiente è stato sgomberato, il crollo della CO₂ è dovuto al fatto che all'arrivo degli occupanti, la porta è stata aperta ed uno scambio di aria si ha avuto con il corridoio che è stato arieggiato, con un'apertura della finestra, durante la pausa pranzo.

2.6.3 Locale C

Si tratta di una zona abbastanza ampia con ampie vetrate e con la porta verso la zona più ampia del corridoio. La figura 2.8 mostra l'esito di un monitoraggio di 24 h dalle 16:00 alle 16:00. All'inizio del monitoraggio l'ambiente era occupato da una persona. si può notare lo scalino dei valori di luminosità poco dopo le 18:00; un andamento del genere, significa che la luce è stata spenta e l'inizio della fase calante della concentrazione di CO₂ ne è la prova. I livelli di luce iniziano ad aumentare successivamente dalle 8:00 con i primi raggi di sole fino ad arrivare ad un culmine massimo attorno alle ore 15:00; infatti la presa alla quale il sensore è stato collegato era rivolta verso Ovest.

Da questo grafico è semplice notare che il locale è stato occupato solo per uno dei due giorni di monitoraggio, la seconda parte di questo periodo di monitoraggio non presenta nette variazioni nè di umidità che di CO₂.

Molto interessante è stato notare la gobba che presenta l'andamento della concentrazione di anidride carbonica durante la notte; dopo alcune prove, si riconduce alla presenza di una pianta all'interno della sala che, per il processo di fotosintesi, emette CO₂ durante la notte.

2.6.4 Locale D

Come è descritto in tabella 2.3 tale ambiente è un bagno, per cui viene generalmente occupato per un periodo piuttosto ridotto, normalmente di pochi minuti. In particolare, si tratta di un bagno cieco, senza fonti di luce naturale esterna;

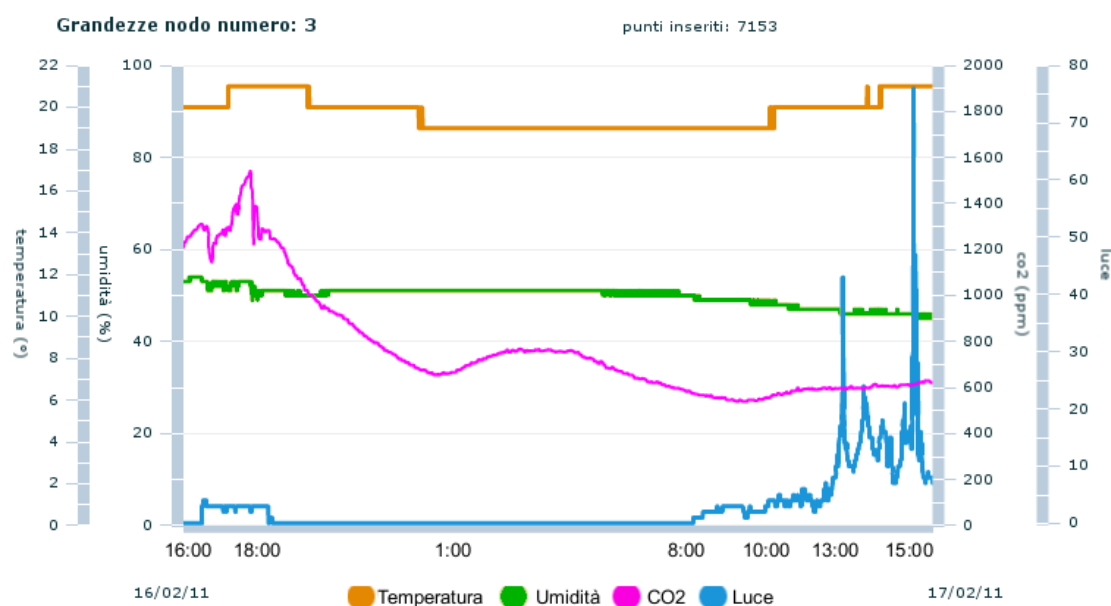


Figura 2.8: Grafico locale C

l'accensione della luce è automatizzata grazie ad un sensore ad ultrasuoni che rileva il movimento dell'occupante. Tutte queste caratteristiche si riscontrano facilmente nell'andamento dei valori di luce rilevati dal sensore. Ogni variazione di luce è quindi a gradino (on/off) e, per come è posto il sensore, garantisce la presenza di una persona all'interno della stanza.

In figura 2.9 è illustrato il risultato di un monitoraggio per un frangente di mattinata nel locale D.

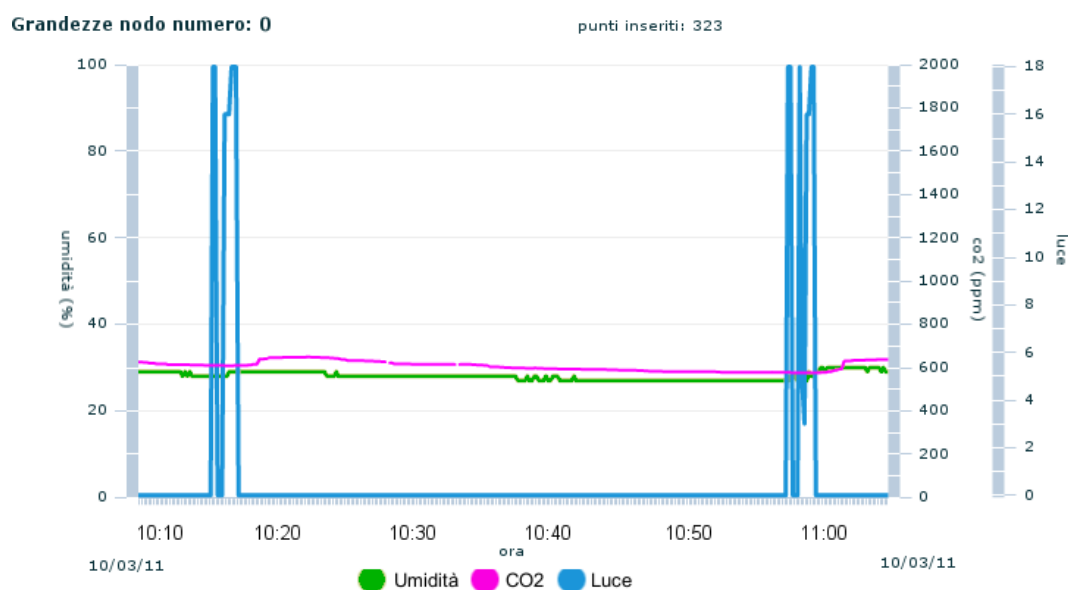


Figura 2.9: Grafico locale D

In questo intervallo temporale, il bagno è stato occupato per 2 volte, la prima volta alle 10:15 e la seconda poco prima delle 11:00. Si notano subito le variazioni brusche dei valori di luce che passa da un valore nullo a quello massimo in modo istantaneo.

Ad ogni utilizzo del bagno, si nota una sensibile variazione dei valori di CO₂. Tale grandezza aumenta di qualche ppm solamente dopo che il locale è stato sgomberato; questo piccolo ritardo è stato rilevato anche in altri test ed è dovuto al sensore utilizzato.

Anche l'umidità subisce una leggera variazione; l'utilizzo dell'asciugamani elettrico porta ad innalzare l'umidità della stanza di qualche punto percentuale.

2.6.5 Locale E

Tale ambiente viene utilizzato in azienda come laboratorio per praticare test, saldature, assemblaggi, ecc. . . Non è quindi occupato perennemente; l'occupazione, infatti, si ha normalmente per un breve periodo (qualche decina di minuti); in particolare la forma del locale non è omogenea ed i sensori non sono stati posti al centro della stanza.

La figura 2.10 è frutto anch'essa di 24 ore di monitoraggio dalle 18:00 alle 18:00 del giorno successivo. Si può notare una certa attività nel laboratorio attorno le 19:30; la luce viene accesa e si nota una notevole variazione dei livelli di CO₂ ed un sensibile aumento dell'umidità, proprio in questo periodo, il laboratorio è stato utilizzato per effettuare delle operazioni proprio in prossimità dei sensori da un dipendente.

Durante la nottata non si notano variazioni, solamente una fase di assestamento della CO₂ con decrescenza esponenziale. Il giorno successivo, è caratterizzato da una notevole aumento di temperatura (da 16° a 25°C) dovuto alla posizione del laboratorio verso Est con un abbassamento del livello di umidità. I livelli di CO₂ indicano che il laboratorio è stato utilizzato solamente dalle ore 11:00 in poi circa.

2.6.6 Locale F

Dalla figura 2.5 si nota come tale ambiente, si sviluppi con una forma a T e si estenda in modo tale da essere occupato dai dipendenti dell'azienda ogni qual volta ci sia un loro spostamento da un locale all'altro; ne consegue che esso sia spesso occupato ma per un breve periodo. Di seguito, la figura 2.11 rappresenta, a differenza di tutte le altre, un monitoraggio d'ambiente durante un giorno non lavorativo.

Tale monitoraggio è stato infatti effettuato durante una giornata intera di sabato; in tale giorno non vi è presente nessun dipendente in ditta ma i locali sono occupati per un breve tempo da un'impresa di pulizie. La raccolta dati ha permesso di stabilire con buona precisione l'orario di arrivo degli addetti alla pulizia.

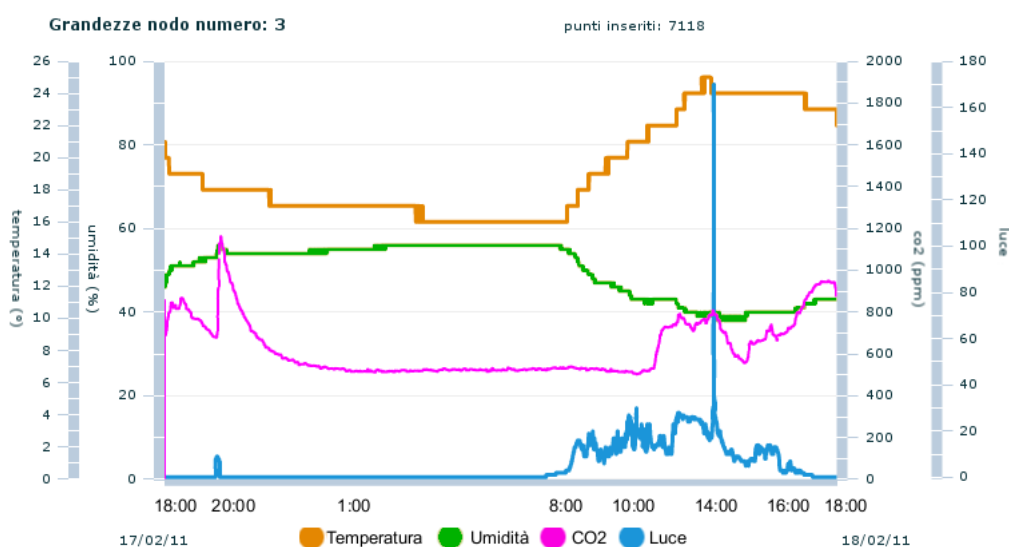


Figura 2.10: Grafico locale E

Il grafico presenta dei valori di CO₂ e umidità piuttosto costanti per tutta la giornata ma proprio in prossimità delle ore 14:40 si ha una brusca variazione di umidità, un calo repentino della temperatura di 2°C (si ipotizza dovuto all'apertura di una finestra) e un piccolo scalino nei valori di anidride carbonica. Tali variazioni sono subito confrontate con altri dati raccolti contemporaneamente in quel periodo da un altro nodo e si è visto che proprio in quel periodo la luce del bagno si è accesa per più di una volta, segno che effettivamente qualcuno stava occupando i locali monitorati. Anche in questo caso è possibile notare un aumento dell'anidride carbonica durante la notte dovuto alla presenza di alcune piante vicine al sensore.

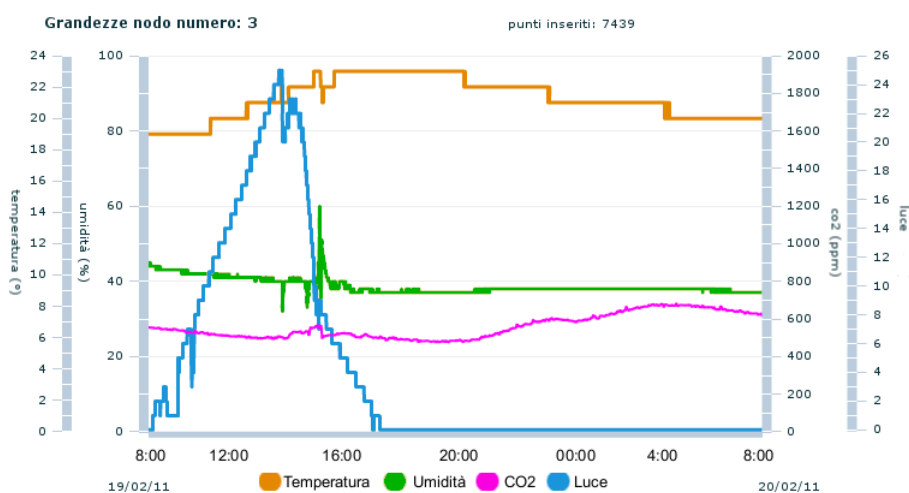


Figura 2.11: Grafico locale F

Nei capitoli successivi

Nei capitoli che seguiranno verrà svolta un'analisi più dettagliata del sistema progettato.

Nel capitolo 3 saranno meglio approfonditi l'impiego e le funzionalità delle reti WSN con protocollo ZigBee; dopo una prima parte teorica sul funzionamento del protocollo, si passerà ad una sezione più applicativa in cui verrà analizzato hardware e software dei moduli ZigBee utilizzati nel progetto.

Nel capitolo 4, si passerà ad illustrare la parte di sistema relativa alla raccolta e visualizzazione dei dati. L'analisi verrà suddivisa in due parti, la prima relativa alla raccolta dei dati dalla rete, la loro interpretazione ed il salvataggio nel database mentre la seconda relativa alla loro visualizzazione nella pagina web a seconda delle richieste del client. Anche in questo capitolo, saranno approfondite le risorse hardware e software utilizzate per ottenere tali funzionalità.

Il capitolo 5, invece, approfondisce più parti distinte tra loro. La prima è relativa ad alcuni test effettuati in aule di scuola primaria, per tale occasione sono state monitorate due aule di due scuole completamente diverse tra loro ottenendo dei risultati piuttosto interessanti.

Nella seconda parte, invece, vengono analizzati e commentati i risultati conseguenti alle misurazioni dei consumi dei moduli wireless adoperati.

Nelle ultime due sezioni, viene discusso il progetto finito; elencando le funzionalità che potrebbero essere ottimizzate e verificando il raggiungimento degli obiettivi preposti grazie all'analisi del prodotto finito.

Capitolo 3

WSN e protocollo ZigBee

Per la progettazione dell'architettura del sistema, si è dapprima svolta una ricerca sulle varie funzionalità (MAC, routing, scheduling, power saving, topology control. . .) presentate in lettura; solo successivamente la ricerca si è indirizzata sui componenti hardware commerciali disponibili sul mercato. Sfruttando le conoscenze apprese, si è creata un'architettura adatta allo scopo di questo progetto di tesi.

3.1 Il protocollo IEEE 802.15.4

3.1.1 Trasmissione dei bit a livello fisico (PHY)

La trasmissione dei bit a livello fisico dipende ovviamente dall'hardware a disposizione. Dato che i nodi utilizzati in questo progetto di tesi (come si vedrà meglio nel seguito) sono dotati di modulo ZigBee, la modulazione radio avviene secondo lo standard IEEE 802.15.4 [20]. Perciò non verranno considerati altri standard. Questo standard, rilasciato nel 2003, adotta una tecnica di *Direct Sequence Spread Spectrum (DSSS)*. Le trasmissioni DSSS moltiplicano i dati da trasmettere per un segnale di 'rumore'. Questo segnale di rumore è una sequenza pseudocasuale di 1 e -1, ad una frequenza molto maggiore di quella del segnale originale, distribuendo così l'energia di quest'ultimo su una banda molto più larga. Il segnale derivante sembra rumore bianco. Tuttavia, questo segnale di 'rumore' viene adoperato dal ricevente per ricostruire il segnale originale, attraverso la moltiplicazione per la stessa sequenza pseudocasuale di 1 e -1. Questo processo è conosciuto con il nome di *despreading*. Affinché il despreading funzioni correttamente, le sequenze di trasmissione e ricezione devono essere sincronizzate. Ciò richiede che il ricevitore sincronizzi la sua sequenza con quella del trasmettitore attraverso un processo di ricerca del timing. Un beneficio derivante dalla sincronizzazione è la possibilità di usare il timing relativo per determinare la posizione del ricevitore se si conosce quella del trasmettitore.

L'effetto causato del despreading è un miglioramento del rapporto *SNR (Signal to Noise Ratio)* ed è definito *process gain*. Se un terzo trasmettitore invia

una sequenza pseudocasuale diversa (o non la trasmette) sullo stesso canale, il despreading non determina alcun process gain per quel segnale. Per questo motivo il DSSS implementa il *Code Division Multiple Access (CDMA)*, che permette a diversi trasmettitori di utilizzare lo stesso canale, all'interno dei limiti delle proprietà di correlazione tra le sequenze pseudocasuali ([21]).

Ritornando allo standard 802.15.4, le bande definite sono tre:

- 868 Mhz per l'Europa
- 915 Mhz per gli Stati Uniti
- 2.4 Ghz in tutto il mondo

Per consentire la coesistenza di più reti nella stessa area, è stato adottato un approccio *Frequency Division Multiplexing*, dividendo le bande disponibili in canali (1 canale per la banda 868 Mhz, 10 per la 915 Mhz con channel spacing di 2 Mhz, 16 per la 2.4Ghz con channel spacing di 5 Mhz). I canali delle prime due bande sono stati definiti per applicazioni a basso bit rate. Quest'ultimo è pari a 20 kbps per la banda 868 Mhz e 40 kbps per 915 Mhz. L'ampiezza dei canali della banda 2.4 Ghz consente invece bit rate maggiori, fino a *250 kbps*.

Le principali funzioni del livello fisico dell'802.15.4 sono le seguenti:

- **Attivazione/Spegnimento** del modulo radio
- **Energy Detection (ED)**: stima la potenza del segnale ricevuto (parametro utilizzato per la selezione del canale)
- **Link Quality Indication (LDI)**: caratterizzazione del segnale ricevuto in base alla potenza e alla qualità del segnale
- **Selezione del canale**
- **Clear Channel Assessment**: parametro per la rilevazione dell'energia del canale per determinare se è già occupato da una stazione che sta trasmettendo.
- **Trasmissione e Ricezione** del bit a livello fisico.

Il *protocol data unit* del livello fisico dell'802.15.4 è mostrato in figura 3.1 ed è costituito dai seguenti campi:

- **SHR**: permette al ricevente di sincronizzarsi con il bit stream.
- **PHR**: contiene informazioni sulla lunghezza del frame.
- **Payload**: è di lunghezza variabile e contiene il MAC frame.

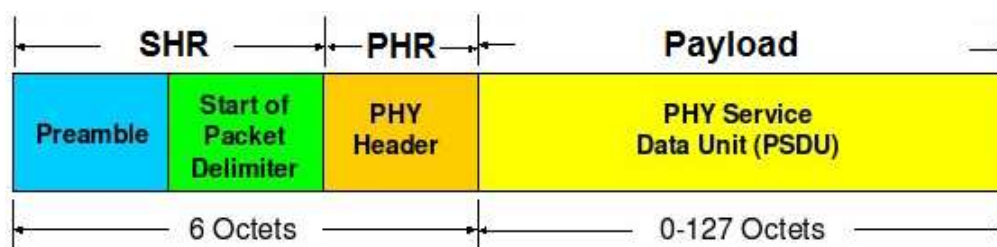


Figura 3.1: Protocol Data Unit del livello fisico dell'802.15.4

3.1.2 Livello di accesso al canale (MAC)

Le principali funzioni che tale layer dello stack va a svolgere sono:

- Creazione di una Personal Area Network (PAN)
- Associazione e dis-associazione ad una rete
- Trasmissione dei pacchetti Beacon
- Gestione del meccanismo del Guaranteed Time Slot (GTS)
- Attuazione dell'algoritmo CSMA-CA per l'accesso al canale
- Calcolo e verifica del Frame Check Sequence (FCS)
- Supporto alla sicurezza basato sull'algoritmo di cifratura AES-128

Esso deve inoltre gestire il processo di rilevamento (Discovery) da parte di un dispositivo, di quelli ad esso vicini. Il tempo richiesto per far ciò è dell'ordine di 30ms, mentre le tecnologie concorrenti, come Bluetooth, possono impiegare fino a 5-6s prima di poter iniziare ad utilizzare completamente il dispositivo. Le principali funzioni del livello MAC sono implementate in software a differenza di quanto avviene per il livello fisico (PHY layer) e scritte generalmente in linguaggio C.

Esistono 4 possibili tipi di frame a livello MAC, come illustrato in fig.3.2:

- Frame di dati
- Frame ACK
- Frame di comando MAC
- Frame di beacon

Il **frame di dati** è costituito al massimo da 128 bytes; esso è numerato per assicurare l'instradamento di tutti i pacchetti. Il campo *Frame Check Sequence* assicura che tutti i pacchetti siano ricevuti senza errori; grazie a ciò, l'affidabilità in condizioni sfavorevoli di trasmissione migliora notevolmente. Un altro frame molto importante è il **frame ACK**. Esso fornisce la conferma che il pacchetto

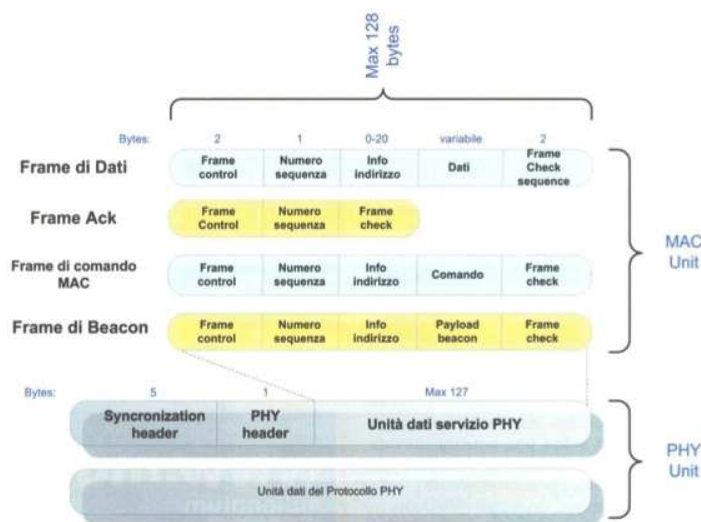


Figura 3.2: I frame del MAC layer

inviato è stato ricevuto correttamente garantendo la consistenza dei dati, ma ovviamente con un aumento di latenza. Il **frame di comando MAC** fornisce un meccanismo per il controllo e configurazione remota dei nodi client. Come ultimo frame, il **frame di beacon** ha il compito di ‘svegliare’ i dispositivi client, i quali sono in ascolto del loro indirizzo e vanno in modalità sleep se non lo ricevono. I *beacon* (che sono in pratica segnali di sincronismo) sono importanti per le reti a maglia e cluster-tree per mantenere tutti i nodi sincronizzati senza la necessità che essi rimangano in ascolto per lunghi periodi di tempo, consumando così le batterie. Trattandosi di una trasmissione in cui il mezzo (radio) è condiviso da tutti i dispositivi, è necessario disporre di qualche metodo di arbitraggio della trasmissione, affinché due dispositivi non inviino pacchetti contemporaneamente. Esistono due tecniche utilizzate: la CSMA-CA ed il beacon.

Ora verrà discussa solamente la seconda tecnica sopra citata in quanto la prima è già stata discussa nel primo capitolo. La seconda tecnica consiste nell’invio da parte del coordinatore di un superframe (modalità beacon) ad intervalli regolari di tempo (multipli di 15.38ms, fino a 252s). Tra un beacon e l’altro ci sono 16 time slot (slot temporali) di pari ampiezza, in ciascuno dei quali è garantita l’assenza di collisione, come illustrato in figura 3.3.

Tutti i dispositivi si contendono i primi 9 time slot, mentre gli ultimi slot temporali sono invece assegnati dal coordinatore ad un nodo specifico e sono detti GTS (Guaranteed Time Slot). Durante la fase di inattività (*Inactive*) tutti i nodi sono nella fase di sleep. Nella fase di attività (*Active*) tutti i nodi possono accedere al canale, in particolare tale intervallo è suddiviso a sua volta in:

- **Contention Access Period (CAP):** tale periodo è diviso in slot temporali lunghi 20 simboli, che costituiscono la base temporale per l’algoritmo Slotted CSMA/CA
- **Contention Free Period (CFP):** tale periodo è formato da gruppi di uno o

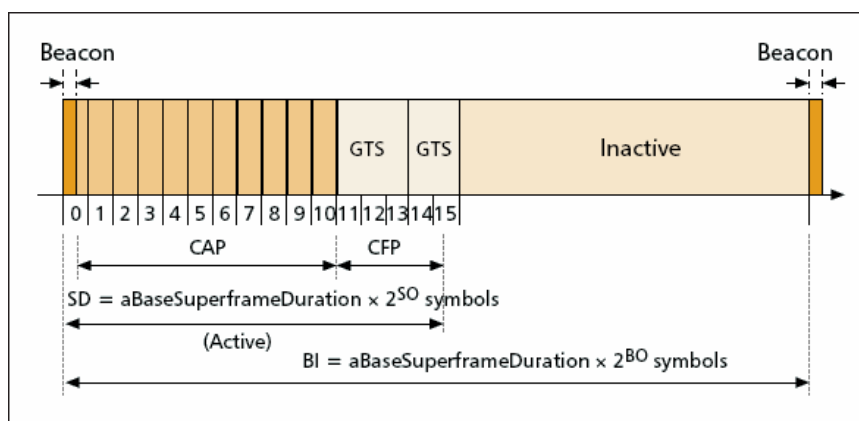


Figura 3.3: Il superframe: accesso al mezzo in modalità beacon

più slot temporali (GTS) che vengono riservati per la fase di trasmissione e ricezione ai dispositivi che necessitano garanzie di accesso al canale.

Nel caso un nodo debba trasmettere una grande quantità di informazione, il coordinatore può assegnarli anche più di un GTS. Tale struttura garantisce una banda dedicata ed una bassa latenza rispetto alla prima tecnica. Inoltre consente di ridurre notevolmente il consumo delle batterie, poiché ciascun dispositivo sa esattamente quando trasmettere ed è sicuro che non ci saranno collisioni.

Esiste anche un'ulteriore modalità di funzionamento, la modalità **non beacon-enabled**. In questo modo di funzionamento, i messaggi Beacon vengono trasmessi solo in risposta alle interrogazioni dei dispositivi che vogliono unirsi ad una rete già esistente o formarne una nuova. In tale modalità ogni dispositivo utilizza l'algoritmo CSMA/CA per accedere al canale.

3.1.3 Livello di rete (NTW)

In questo livello s'inserisce il protocollo ZigBee il quale, come già detto in precedenza, fornisce i servizi di rete e livello applicativo. Brevemente, le funzioni ZigBee del livello di rete riguardano:

- Esecuzione dei comandi MAC come livello superiore
- Individuazione delle figure di rete coordinatore, router ed end-device, ognuna con funzioni e compiti man mano decrescenti all'interno della rete
- Auto-formazione e gestione delle connessioni di rete
- Supporto dei pacchetti data e comando
- Possibilità d'implementazione di un 'Trust Centre' che gestisca le mansioni di sicurezza dell'intera rete ZigBee
- Gestione delle chiavi network create con algoritmo di cifratura AES-128

- Assegnazione di indirizzi a 16 bit per l'utilizzo nelle politiche di routing
- Realizzazione dell'instradamento dei pacchetti.

Proprio quest'ultimo punto necessita un piccolo approfondimento, in quanto, l'instradamento può avvenire per due modi diversi:

1. **Tree Routing:** i pacchetti destinati ad altri dispositivi nella rete vengono inviati al 'padre' o al 'figlio' secondo le associazioni effettuate con le primitive IEEE802.15.4.
2. **Table Routing:** un ciclo di '*discovery*' consente di conoscere quali sensori sono stati raggiunti da un pacchetto inviato in broadcast. Le risposte dei singoli dispositivi (pesati a seconda della funzione prescelta) consentono di realizzare ad ogni hop la tabella di routing.

Il protocollo utilizzato, prevede la formazione di tre tipi differenti di topologie di reti (stella, albero, mesh) che saranno discusse nei paragrafi seguenti.

3.1.4 Livello di Applicazione (APL)

Il layer applicazione è costituito dai driver e dal codice, contenuti nella ROM del microcontrollore. Schematicamente di un nodo ZigBee si possono evidenziare, oltre al blocco relativo all'alimentazione, anche quelli inerenti il transceiver, l'antenna, il microcontrollore e l'interfaccia utente (rappresentata da Input/Output). Il transceiver implementa il layer fisico, ossia si occupa della modulazione del segnale come descritto in precedenza. All'interno della ROM del microcontrollore è presente l'implementazione del livello MAC, NWK e applicazione.

Il protocollo ZigBee, a differenza di altri protocolli di comunicazione, definisce anche il livello applicativo; in particolare esso è composto dall'Application Support Sublayer (APS), dal Zigbee Device Object (ZDO) e da oggetti applicativi definiti dal produttore. Nella Figura 3.4, si può osservare come è strutturato tutto lo stack di ZigBee.

L'**APS** si occupa di mantenere le tabelle delle associazioni tra dispositivi caratterizzati da servizi ed esigenze proprie, e di inoltrare i messaggi tra dispositivi associati. Un altro suo compito è quello del discovery, cioè della ricerca di altri dispositivi che stanno operando nella zona. **ZDO** definisce il ruolo dei nodi, inizializza e risponde a richieste di connessione e crea connessioni sicure tra nodi. Gli oggetti applicativi definiti dal produttore implementano le funzionalità scelte in conformità alle regole dello standard ZigBee.

3.1.5 Tipi di nodo

Una WPAN ZigBee può includere due diversi tipi di dispositivi: **FFD** (Full Function Device) e **RFD** (Reduced Function Device). Un dispositivo del primo tipo può operare all'interno della rete secondo tre modalità: funzionando da coordinatore della rete, da coordinatore semplice o da terminale di comunicazione. Un

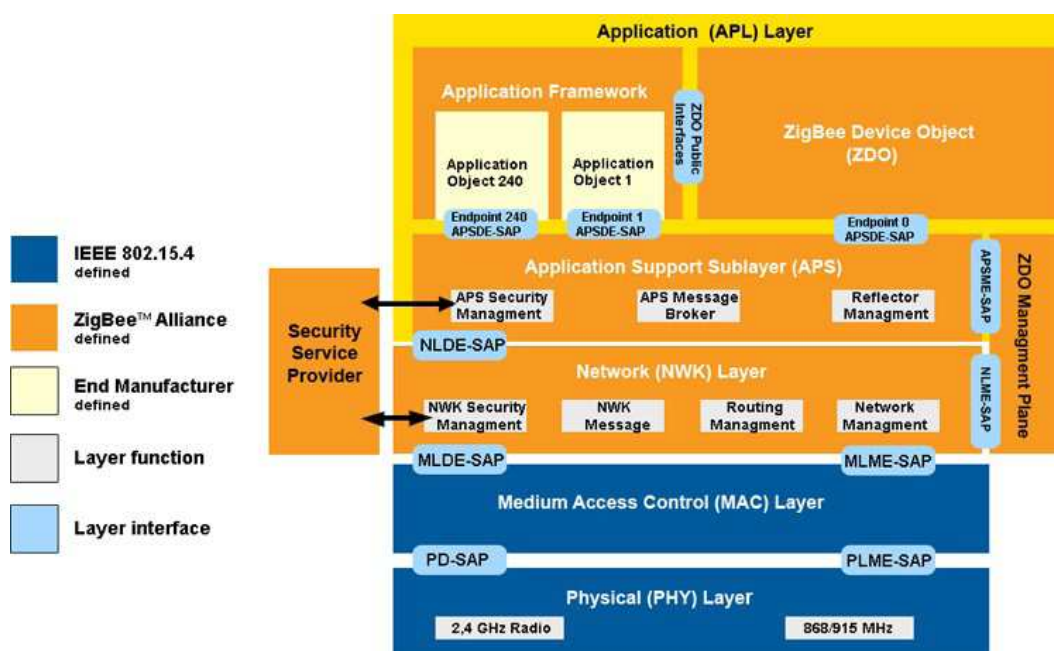


Figura 3.4: Architettura dello stack ZigBee

dispositivo FFD (Coordinatore o Router) può dialogare con altri dispositivi di entrambe le categorie, mentre un RFD (EndDevice) può comunicare direttamente solo con un FFD. L'inclusione di terminali RFD all'interno della rete è orientata per applicazioni estremamente semplici, come interruttori di luce o sensori ad infrarossi, che non necessitano dell'invio di grosse quantità di dati e possono quindi, essere supportate attraverso minime risorse energetiche e limitate capacità di memoria.

I dispositivi possono quindi essere di tre tipi:

- Coordinatore
- Router
- EndDevice (nodo foglia)

Coordinatore

Tutte le reti di tipo ZigBee possiedono un nodo Coordinatore indipendentemente dalla tipologia di rete utilizzata (come vedremo nel prossimo paragrafo).

A livello di rete (NTW) il coordinatore svolge importanti funzionalità per l'inizializzazione della rete infatti:

- seleziona la frequenza di canale da utilizzare (quella con l'attività più bassa),
- inizializza la rete,
- permette ad altri dispositivi di connettersi alla rete da esso gestita,

- fornisce messaggi di routing gestione della sicurezza e altri servizi.

Router

I Router sono dei nodi che adempiono alle seguenti funzionalità:

- ritrasmettono i messaggi da un nodo all'altro,
- permettono ai nodi figli di connettersi ad essi.

Come vedremo nel prossimo paragrafo, i Router compiono queste attività in modo diverso a seconda si tratti di una rete ad albero oppure mesh; infatti, in una rete ad albero, i router sono posti in modo tale da passare i dati *'su e giù'* per l'albero mentre in una rete mesh, un router può essere posizionato dappertutto.

Da notare il fatto importante che i router non possono dormire, in quanto devono essere sempre in ascolto dei propri figli.

EndDevice

Sono dispositivi sempre collocati alle estremità della rete utilizzata. Tali nodi non sono in grado di ritrasmettere messaggi e non possono permettere ad altri nodi di aggiungersi alla rete.

La caratteristica principale di questi nodi è quella che svolgono la loro fase di trasmissione e ricezione dati alternata con una fase di *sleep* in cui il processore viene posto in uno stato dormiente a basso consumo energetico. In questo modo sono spesso attrezzati con un'alimentazione a batteria facilmente integrabile con una sorgente d'energia rinnovabile (pannello fotovoltaico, cella di Peltier, ecc...)

3.1.6 Topologie di rete

Come precedentemente accennato, le tipologie di rete supportate dallo standard ZigBee e quindi dallo standard IEEE 802.15.4 sono le seguenti:

1. Stella
2. Albero
3. Mesh

Rete a Stella

La rete a stella è la più semplice delle reti ZigBee. Tale semplicità non garantisce grosse potenzialità in quanto essa è composta solamente da un Coordinatore e più nodi EndDevice. Ogni EndDevice può comunicare solamente con il Coordinatore; in questo modo, perchè avvenga una comunicazione tra un EndDevice ed un altro, tale messaggio dovrà passare per il Coordinatore (fig 3.5).

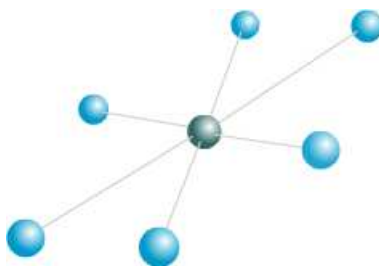


Figura 3.5: Rappresentazione di rete a stella

Rete ad Albero

Tale topologia consiste in un nodo Coordinatore al quale gli altri nodi sono connessi con le seguenti due regole:

1. Il Coordinatore può essere connesso con una serie di nodi Router o EndDevice loro figli
2. Ogni Router può essere connesso a più Router e/o EndDevice.
3. Gli EndDevice non possono aver figli.

Tale gerarchia può essere rappresentata come un albero con il Coordinatore posto nel top come mostrato in fig. 3.6

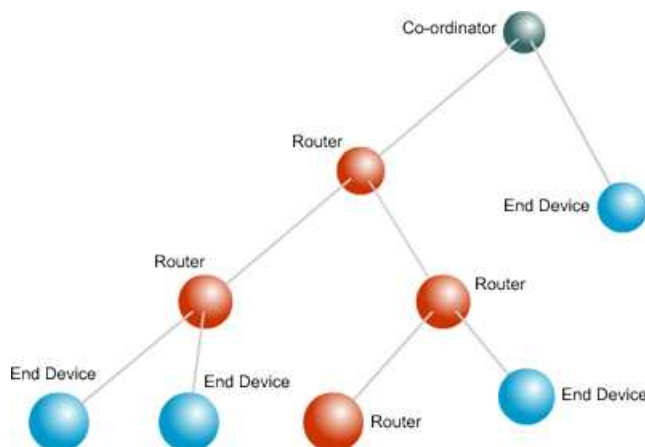


Figura 3.6: Rappresentazione di rete ad albero

la comunicazione avviene nel seguente modo: un nodo può comunicare direttamente solamente con il proprio padre o con i propri figli (se esistono). Per spedire un messaggio attraverso la rete, esso dovrà viaggiare dal nodo sorgente risalendo l'albero finché trova il più vicino nodo antenato del nodo destinatario, a questo punto il messaggio potrà scendere attraverso l'albero fino a destinazione.

Uno svantaggio di questa topologia di rete è che non esistono percorsi alternativi di comunicazione se un collegamento tra nodi dovesse fallire.

Rete Mesh

Tale rete offre alcune potenzialità che le altre topologie non riescono ad offrire. Proprio questa topologia è stata utilizzata nel progetto di tesi per cui si cercherà di approfondirla più nel dettaglio.

Come in una rete ad albero, il Coordinatore può connettersi con una serie di nodi Router e EndDevice; anche un Router può connettersi con altri Router e EndDevice.

A differenza delle reti ad albero, il modo di comunicazione è leggermente differente, in particolare:

- Un Router può comunicare direttamente con un suo figlio, con un suo padre ma anche con qualsiasi altro Router o Coordinatore all'interno del range radio.
- Il Coordinatore può direttamente comunicare con i suoi figli ma anche con qualsiasi Router all'interno del range radio.

La topologia ottenuta è rappresentata in figura 3.7.

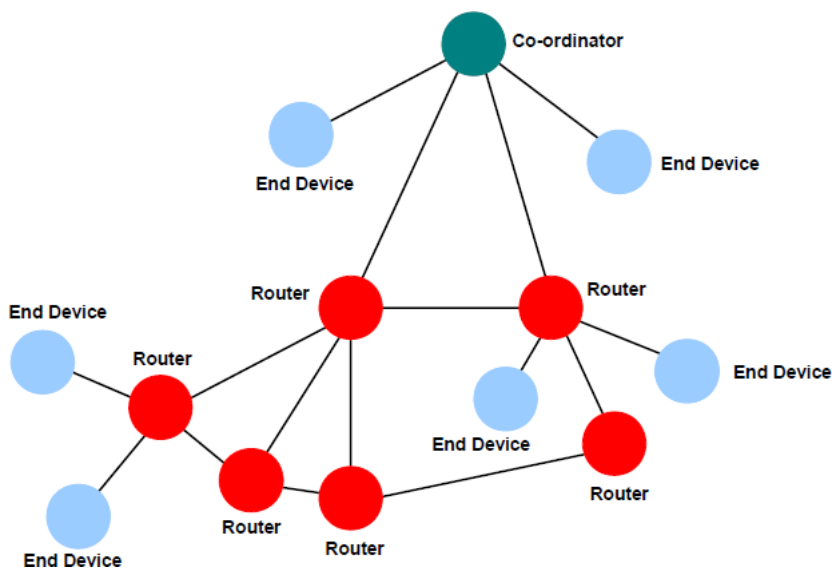


Figura 3.7: Rappresentazione di rete mesh

L'abilità di un nodo Router di comunicare direttamente con altri nodi vicini (entro il raggio d'azione del segnale wireless) è la proprietà principale che caratterizza le reti mesh rispetto a quelle ad albero. Sfruttando questa proprietà, si ottengono comunicazioni molto flessibili che permettono di identificare percorsi alternativi all'interno della rete nel caso in cui un collegamento dovesse saltare.

Se un EndDevice è impostato in modalità di sleep, non è in grado di ricevere messaggi direttamente; il protocollo permette che i dati destinati ad un nodo EndDevice vengano bufferizzati al suo nodo padre; solamente una volta che il nodo foglia è in fase non-dormiente, esso può chiedere l'informazione ricevuta al proprio padre.

In questo tipo di reti, è prevista una proprietà di ‘route discovery’ che permette di trovare il percorso migliore disponibile nella rete per raggiungere un certo nodo.

Si vuol ricordare nuovamente, che la propagazione dei messaggi è gestita direttamente dal network layer (NWL) dello stack ed è quindi trasparente all’applicazione caricata nel nodo.

3.2 ZigBee PRO

Dal momento in cui i dispositivi utilizzati per il progetto utilizzano questa nuova versione di protocollo ZigBee, in questo breve capitolo esso sarà introdotto e le varie differenze con la versione standard. Attualmente, ZigBee PRO è l’ultima versione rilasciata del protocollo ZigBee. Tale configurazione fornisce caratteristiche aggiuntive alle specifiche del protocollo ZigBee 2007 e si differenzia per molti aspetti dalle specifiche Zigbee 2006. L’intento di ZigBee PRO è quello di migliorare le funzionalità del protocollo ZigBee, rendendolo più robusto ed efficiente in scenari applicativi con elevato numero di nodi e possibili interferenti [22].

Le principali caratteristiche che contraddistinguono tale protocollo sono le seguenti:

- **Creazione di reti Beaconless:** in tale maniera, non è presente un meccanismo di sincronizzazione fra padre e terminale figlio in sleeping mode. In aggiunta, ogni dispositivo terminale (foglia), è libero di configurare un proprio duty cycle. In questo caso l’interazione tra padre e figlio è gestita con modalità polling, Il dispositivo padre deve memorizzare nella cache le informazioni destinate al figlio dormiente.
- **Assegnamento degli indirizzi.** Non viene più gestito in maniera gerarchica, ma si usa una distribuzione stocastica. Ciò permette l’implementazione di procedure di gestione per i conflitti d’indirizzo e consente la possibilità di riutilizzo degli indirizzi
- **Routing.** Implementazione di tecniche che permettono di minimizzare il traffico di routing e la dimensione delle rispettive tabelle, specialmente in presenza di nodi catalizzatori del traffico (*aggregators*), in particolare:
 - **Many-to-one routing:** evita che ogni dispositivo della rete attivi separatamente la procedura di discovery del nodo catalizzatore, demandando al nodo aggregator il compito di comunicare in broadcast la propria presenza.
 - **Source routing.** Inserisce le informazioni di routing necessarie all’instradamento all’interno del pacchetto stesso. In questo modo solo il nodo sorgente deve possedere una entry relativa al percorso nella tabella di routing. È necessario però che il nodo sorgente reperisca, in precedenza, informazioni sulla topologia della rete e conosca per intero il percorso che lo collega alla destinazione.

- **Frequency agility.** Offre la possibilità di cambiare il canale di lavoro mentre la rete è operativa garantendo maggior protezione dalle interferenze.
- **Sicurezza.** Dà la possibilità di scegliere fra due differenti profili di sicurezza:
 1. *standard*: non richiede nessun tipo di autenticazione fra dispositivi padre-figlio e fra neighbor; per il trust center è opzionale supportare le link key; il cambio della network key avviene utilizzando la network key corrente.
 2. *high*: richiede l'autenticazione fra dispositivi padre-figlio e fra neighbor; i pacchetti provenienti da dispositivi non presenti nelle tabelle dei neighbor sono scartati; tutti i dispositivi supportano le link key; non è possibile cambiare la network key utilizzando la network key corrente.
- **Compatibilità con ZigBee.** I dispositivi sono in grado di distinguere le reti ZigBee da quelle ZigBee PRO attraverso un identificatore che le distingue nel payload del pacchetto Beacon. In particolare, i dispositivi ZigBee PRO possono unirsi ad una rete ZigBee solo come dispositivi terminali, anche i dispositivi ZigBee possono unirsi ad una rete ZigBee PRO solo come dispositivi terminali ma solo se la rete in questione utilizza il profilo di sicurezza standard.

3.3 Hardware utilizzato

Il mondo del mercato offre una vasta scelta per quel concerne l'hardware utilizzabile per implementare una WSN. Dopo un periodo di ricerca dei prodotti disponibili, la scelta è ricaduta su un prodotto molto innovativo, ad alte prestazioni e a ridotti consumi, si tratta del modulo JN148 della casa produttrice Jennic. Il datasheet integrale del dispositivo è mostrato in appendice. Di seguito, in figura 3.8 è riportato lo schema a blocchi del dispositivo.

Da tale schema, non è difficile notare i principali blocchi del nodo wireless in questione. In particolare, i componenti hardware dei nodi di una WSN comprendono CPU, memoria, radio, sensori, attuatori e fonte di alimentazione.

Componenti hardware

Di seguito son riportate alcune delle caratteristiche del nodo JN148 scelto¹:

- **Controller:** è la CPU del nodo e si occupa di rilevare i dati tramite i sensori, elaborarli, decidere dove e quando spedirli, ricevere dati da altri nodi, gestire gli attuatori e i sensori. Il dispositivo utilizzato monta un microcontrollore, ovvero un processore per sistemi embedded. Poiché richiede poca

¹Queste e molte altre caratteristiche sono disponibili nel datasheet del dispositivo [23] e del microcontrollore [24]

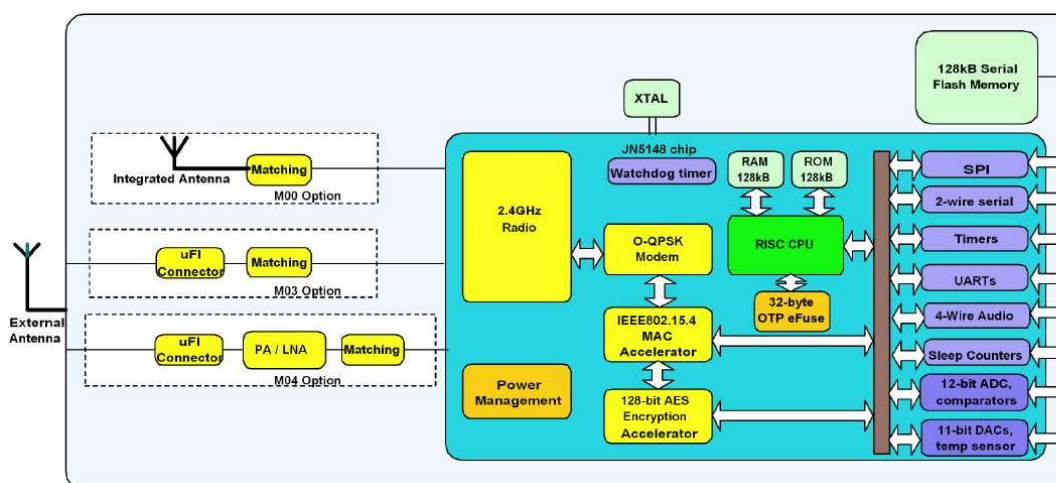


Figura 3.8: Schema a blocchi del modulo wireless JN148

energia, ha delle memoria integrata e senza controller di memoria; è facilmente programmabile e si integra bene con altri componenti come sensori e attuatori e ha delle modalità di risparmio energetico.

Si tratta di un microcontrollore JN5148 proprio della casa costruttrice Jennic; è un dispositivo RISC (*Reduced Instruction Set Computer*) a 32 bit con frequenza di clock fino a 32 MHz, diversi convertitori analogico/digitale a 12bit, dei convertitori digitale/analogico e molte altre interfacce di comunicazione con l'esterno. Si voglia far notare i consumi di corrente di tale dispositivo, caratteristica fondamentale per renderlo efficace come nodo in una rete WSN ZigBee:

- $100nA$ in stato dormiente profondo (Deep sleep current)
- $1.25\mu A$ in stato dormiente con Timer attivo.
- $17.5mA$ in ricezione
- $15mA$ in trasmissione

- **Memoria:** la memoria è di tipo RAM per le variabili temporanee, mentre per il codice si usano le EEPROM o le flash. Il microcontrollore dispone di 128 KB di memoria RAM, altrettanti di memoria ROM, e 4Mbit di flash per il codice di programmazione e dati.
- **Modulo di comunicazione:** è costituito da un transceiver ad onde radio che opera sui 2.4Ghz (protocollo IEEE 802.15.4).
- **Fonte di alimentazione:** per dare energia ad un nodo si potrebbe ricorrere a batterie, che possono essere di qualsiasi tipo, a seconda dell'hardware e dell'applicazione da realizzare. Le reti che devono avere un lifetime molto lungo potrebbero ricorrere a fonti energetiche ausiliare per ricaricare le batterie dei nodi (celle fotovoltaiche, sistemi di ricarica a vibrazioni meccaniche, ecc. ...).

3.4 Programmazione dei nodi della WSN

3.4.1 Ambiente di sviluppo

Per la programmazione dei nodi wireless è stato utilizzato l'ambiente di sviluppo Eclipse. Si tratta di un ambiente di sviluppo integrato multi-linguaggio e multiplatforma sviluppato da una comunità strutturata sullo stile dell'open source.

Eclipse può essere utilizzato per la produzione di software di vario genere, si passa infatti da un completo IDE per il linguaggio Java (JDT, Java Development Tools) a un ambiente di sviluppo per il linguaggio C++ (CDT, C/C++ Development Tools) e a plug-in che permettono di gestire XML, Javascript, PHP [25].

La piattaforma di sviluppo è incentrata sull'uso di plug-in; questi plug-in permettono la programmazione dei moduli JN148 e sono progettati ad hoc e forniti dalla casa costruttrice Jennic.

I plug-in utilizzati per la programmazione sono due:

- **JenOS Config.Editor.** Questo editor fornisce un'interfaccia grafica per la gestione delle risorse ad alto livello come *timer*, *task*, *mutex*, ecc. . .
- **ZPS Config.Editor.** Tale editor garantisce una facile via per impostare i parametri della rete ZigBee come per esempio le proprietà del Coordinatore, Router e EndDevice. Per un approfondimento delle proprietà configurabili, vedere [26].

Metodi di programmazione

I dispositivi vengono programmati grazie a due metodologie differenti ma collegate tra loro.

- Il microcontrollore montato (JN5148) riesce a lavorare con metodologia **multitasking** e riesce a gestire fino a 32 task contemporaneamente. L'utilizzo dei vari task (passaggio dati, interrupt, ecc. . .) viene gestito dal plug-in grafico di Eclipse.
- Ogni singolo task viene definito mediante uno script in linguaggio C. L'utilizzo delle *API* (*Application Programmable Interface*) fornite da alcune librerie permette di programmare facilmente il microcontrollore sfruttando tutte le sue potenzialità sia a livello ZigBee sia a livello di applicazione².

La figura 3.9 mostra il processo di build utilizzato dal motore di compilazione che permette di ottenere i file binari da utilizzare a partire da i file in linguaggio C dall'utente, dal ZPS Configuration Editor e dal JenOS Configuration Editor.

²La parte relativa agli script di codice utilizzato per il progetto sarà omessa in questo documento

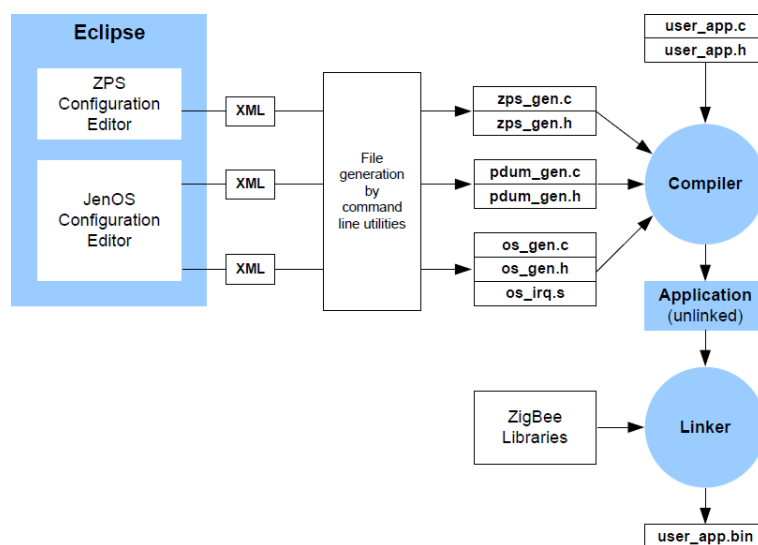


Figura 3.9: Processo di 'Build' (dai file .c ai .bin)

Ora si procederà ad analizzare più o meno dettagliatamente le strutture dei programmi progettati per ognuno dei tipi di nodi della rete. In realtà si dovrebbe seguire una suddivisione tra quello che è il programma relativo la gestione della rete (caratteristiche tipiche del protocollo ZigBee: routing, binding dei nodi, ecc. . .) e quello relativo l'applicazione (raccolta dei dati dei sensori, gestione dell'UART, ecc. . .). In particolare, essendo lo ZigBee uno standard ben definito, la casa costruttrice mette a disposizione dell'utilizzatore la parte di programma relativa all'implementazione della rete. Si farà una suddivisione tra software (SW) progettato per il nodo coordinatore e software progettato per il nodo sensore.

3.4.2 Coordinatore (SW)

L'architettura software utilizzata per il nodo coordinatore è riportata in figura 3.10 in cui sono messi in relazione vari blocchi, in particolare, si possono notare i blocchi *messaggio*, *task* e *software timer*. I primi sono degli oggetti che sono utilizzati (e così si possono pensare) come dei mezzi di comunicazione dei dati, i secondi sono rappresentati da alcuni blocchi di codice che vengono eseguiti se attivati da un qualsiasi evento mentre gli ultimi sono dei veri e propri timer implementati via software.

Nel dettaglio i vari task sono così utilizzati:

- **App_taskControllerNode.** Rappresenta il task principale nel programma applicativo del controllore. E' attivato da un evento di Stack generato dal ZigBee Protocol Stack (ZPS) che gli fornisce un messaggio contenente l'evento di stack. Tale task contiene il codice che permette l'inizializzazione della rete mediante una macchina a stati.
- **App_taskLogData.** Tale task rappresenta il SENSOR endpoint, in particolare gestisce i dati di misurazione provenienti dai nodi sensore (Router e

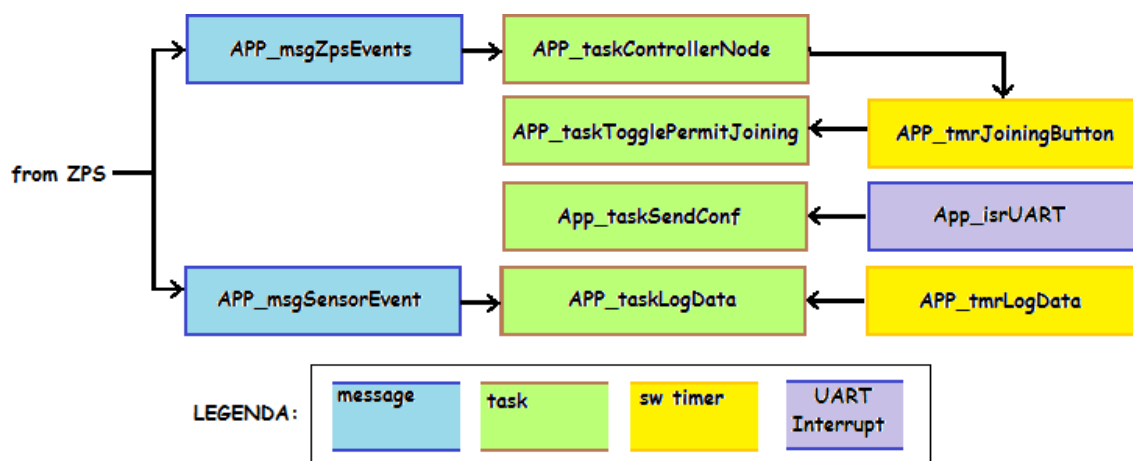


Figura 3.10: Rappresentazione delle relazioni intercorrenti tra i blocchi di software nel nodo coordinatore

EndDevice). Quando un nodo sensore spedisce le sue informazioni raccolte al coordinatore, l'indirizzo del nodo che ha spedito il dato viene registrato e i dati vengono salvati. Utilizzando questa tecnica di log è possibile vedere la cronologia dei dati ricevuti (questo modo di lavorare non è stato utilizzato visto che, in tale progetto, è il miniPC ad effettuare il log dei dati). Il task è attivato dal task ZPS che gli passa un messaggio contenente le informazioni relative al data-frame. E' anche attivato periodicamente da un software timer per effettuare una specie di controllo temporale sui dati.

- **APP_taskSendConf.** Rappresenta il task che permette di impostare un nuovo periodo di campionamento della WSN, tale valore gioca un ruolo importante nella mole di dati che dovrà essere gestita dal coordinatore. Tale task è attivato da un *interrupt* proveniente da una delle due UART del dispositivo; in particolare, quando il processore ARM (miniPC) invierà il nuovo dato per il periodo di campionamento tramite seriale UART, il task si attiverà subito e invierà attraverso tutti i nodi della rete il nuovo dato.
- **APP_taskTogglePermitJoining.** Questo task è gestito da un timer software che è attivato a sua volta dallo stato di Permit-Joining. Tale funzionalità è gestita da un tasto posto nella board del coordinatore che se è premuto per più di 2 secondi, questo task viene attivato. Se il tasto viene premuto per meno di 2 secondi, il task del coordinatore principale blocca il timer prima che esso scada evitando che questo task venga attivato.

3.4.3 Nodo Sensore (Router e EndDevice)

La struttura software del Nodo Sensore³ è rappresentata in figura 3.11

³D'ora in poi si parlerà di Nodo Sensore intendendo quel tipo di dispositivo che permette l'acquisizione delle grandezze fisiche dai vari sensori. In questo contesto si tratterà dei nodi Router e EndDevice

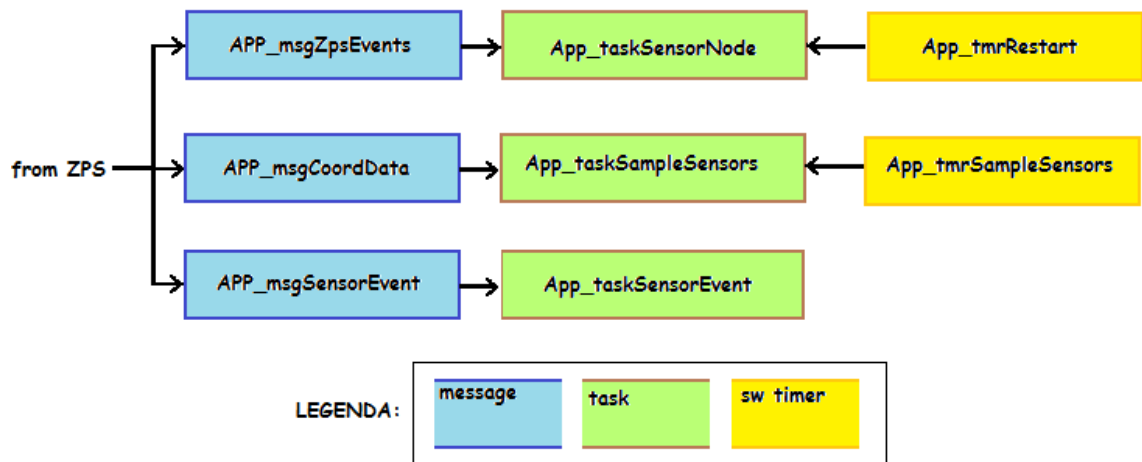


Figura 3.11: Rappresentazione delle relazioni intercorrenti tra i blocchi di software nel nodo sensore

L'applicazione del nodo sensore è implementata dai seguenti task:

- **APP_taskSensorNode.** Si tratta del task principale per questa tipologia di nodi. E' attivato da un evento dello stack generato dallo ZPS dal quale riceve un messaggio contenente i dettagli dell'evento. Il taskSensorNode contiene la macchina a stati necessaria per individuare una rete alla quale aggregarsi.
- **APP_taskSampleSensors.** Tale task coordina la macchina a stati per la lettura dei dati di luce, temperatura, umidità e CO2. A lettura terminata, tale task invia un frame di dati contenente tali valori appena letti al nodo coordinatore. Nel caso in cui il nodo sensore sia un router, questo task è periodicamente attivato da un timer software mentre nel caso in cui si tratti di un EndDevice, il task è attivato da una chiamata di risveglio dallo stato di sleep. In aggiunta, tale task gestisce i messaggi provenienti dal nodo coordinatore (nel caso in questione, l'impostazione del tempo del periodo di campionamento).
- **APP_taskSensorEvent.** Questo è un task aggiuntivo presente solamente nei nodi router e non negli EndDevice. In entrambi queste tipologie di nodo, il led 2 posto sulla scheda indica l'attività del nodo ma, mentre nel nodo EndDevice esso si accende con il risveglio e si spegne con la dormienza del nodo stesso, in un router, il LED si illumina quando un frame di dati è passato allo stack per la trasmissione e si spegne quando lo stesso stack ritorna un valore di conferma. Questo task implementa il SENSOR_DATA endpoint ed ha dunque il compito di ricevere la conferma APS generata dallo ZPS dopo che il frame SENSOR_DATA è stato spedito. Lo ZPS invia

un messaggio contente la conferma e tale messaggio attiva il task. Il task procederà a questo punto con lo spegnimento del LED.

La differenza principale dell'applicazione caricata in un *nodo sensore router* e quella di un *nodo sensore EndDevice*, sta semplicemente nel fatto che mentre il router sfrutta un timer software per attivare il processo di campionamento dei dati dai sensori e procedere con il loro invio, il nodo EndDevice, non essendo sempre acceso come il router, utilizza un timer interno a 32KHz progettato appropriatamente per il risveglio del microcontrollore.

Capitolo 4

Raccolta dati e Visualizzazione

In questo capitolo si andranno ad analizzare più approfonditamente le risorse hardware e i componenti software utilizzati per il sistema di monitoraggio.

4.1 Hardware

La parte fisica del progetto è composta principalmente da 3 piattaforme hardware distinte e con compiti diversi. Tali componenti sono: la *sensor board* (*nodo sensore*), il *nodo Coordinatore* ed il *miniPC*.

4.1.1 Sensor Board (Nodo sensore)

Questa unità hardware risulta essere sostanzialmente una scheda elettronica di dimensioni piuttosto contenute con integrati a bordo i sensori necessari per l'acquisizione delle grandezze fisiche richieste ed il modulo JN148 programmato come Router o EndDevice. I sensori ricoprono un ruolo di notevole importanza nel raggiungimento dello scopo finale, le loro caratteristiche di precisione, sensibilità e fondo scala sono un aspetto molto importante da tenere in considerazione in fase di progetto¹.

Sensore di Temperatura e Umidità

La sensor board monta un sensore *SHT11* della *sensirion* per la lettura dei valori di umidità relativa e temperatura. Il dispositivo è composto da due microsensori calibrati, un convertitore ADC a 14 bit e un minicircuito per la conversione in seriale (I2C). Tale sensore viene interrogato mediante le API dei microcontrollori JN148 che permettono una lettura della temperatura in un range da 0 124 °C e una lettura di umidità da 0 al 100% [37].

¹In tale progetto la sensor board non è stata assemblata direttamente dal progettista del sistema ma è stata acquistata con i sensori di temperatura, umidità e luce già montati; solamente il sensore di CO2 è stato comperato a parte dopo un' attenta analisi delle offerte del mercato per quel tipo di dispositivo e interfacciato per lavorare sulla sensor board.

Sensore di Luce

La scheda del nodo sensore monta un sensore TAOS TSL2550 per la lettura dei valori di luce. Si tratta di un sensore con uscita digitale seriale I2C; esso combina due fotodiodi, uno sensibile alla luce visibile ed uno sensibile all'infrarosso, accompagnati da un convertitore analogico/digitale con una precisione di 12-bit ottenendo una risposta simile a quella dell'occhio umano [38]. Il valore restituito dal sensore rappresenta la luce assorbita dai fotodiodi espressa in *lux*.

La figura 4.1 mostra una fotografia della sensor board completa di modulo RF, sensori di luce, umidità, temperatura e anidride carbonica. Tale circuito può essere alimentato sia da due batterie AAA che da un alimentatore esterno.

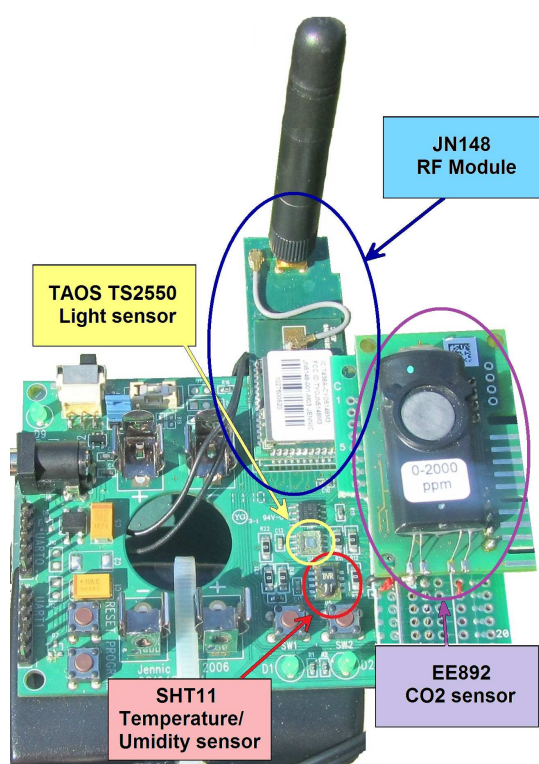


Figura 4.1: Sensor Board utilizzata

4.1.2 Sensore di CO2

A differenza degli altri sensori, la scelta del sensore di CO2 è stata il risultato di una ricerca piuttosto approfondita dei prodotti già presenti sul mercato. L'anidride carbonica, infatti, rappresenta la grandezza d'ambiente più significativa sia per determinare il grado di comfort sia per valutare l'occupancy di un ambiente indoor. A tal punto la scelta è ricaduta su un prodotto innovativo, di buona precisione e dalle grandezze piuttosto contenute; si tratta del sensore *EE892* della ditta *E+E electronic*.

Il principio di misurazione utilizzato (già discusso nel paragrafo 2.4.1) dal dispositivo è basato sulla tecnologia ad infrarossi (NDIR). Una caratteristica importante del dispositivo è che non richiede alcuna manutenzione grazie alla brevettata autocalibrazione $E + E$, che sopperisce agli effetti dell'usura e garantisce un'eccellente stabilità a lungo termine. In aggiunta, il sensore ha un bassissimo consumo e, grazie all'intervallo di misurazione regolabile, il consumo medio può essere ridotto a meno di $60\mu A$ offrendo la perfetta soluzione per dispositivi alimentati a batteria. Il sensore in questione è mostrato in figura 4.2.

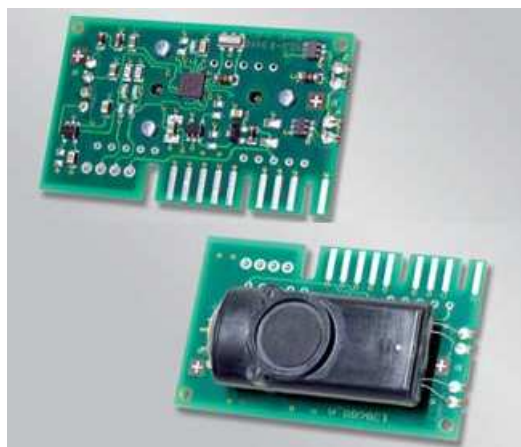


Figura 4.2: Sensore di CO2 utilizzato

Tale dispositivo permette la lettura della concentrazione di CO2 da 0 fino ad un massimo di 2000 ppm con una discreta precisione; le altre caratteristiche tecniche sono illustrate in appendice B.

Presenta un'interfaccia di comunicazione digitale con protocollo di comunicazione non propriamente open source; si tratta del protocollo chiamato *E2 Interface*.

Protocollo digitale E2

Il protocollo di comunicazione E2 è basato sui principi tipici della comunicazione I2C. La differenza più significativa rispetto al più conosciuto protocollo I2C è rappresentata dalla più bassa velocità di trasmissione, da un sistema di addressing modificato e da un diverso meccanismo di gestione dell'errore (checksum byte). L'interfaccia E2 è usata principalmente per la trasmissione digitale bi-direzionale tra un modulo *master* (per esempio un microcontrollore) e un modulo *slave*, nel caso citato il sensore di CO2. La trasmissione avviene in modo sincrono e seriale con il master che è responsabile della generazione del clock; in questo modo, lo slave non può inviare dati in modo indipendente.

Il driver per la lettura del sensore è stato scritto in linguaggio C propriamente per questo sensore (API relative alla comunicazione in I2C non sono state utilizzate in quanto la comunicazione sarebbe stata troppo veloce).

4.1.3 MiniPC ARM

Il modulo hardware più significativo e più potente del sistema è rappresentato dal miniPC; ad esso spettano numerosi compiti: dal salvare i dati provenienti dal coordinatore nel database sino alla loro visualizzazione nella pagina web. E' stato utilizzato un processore ARM9 progettato appositamente per applicazioni embedded; si tratta del *TS-7500* della ditta *Technologic Systems Inc.*[39].

Il TS-7500 è composto da un processore ARM9 Cavium a 250MHz e di una FPGA 5000 LUT; esso rappresenta una soluzione embedded potente e flessibile grazie ad una RAM da 64 MB, una flash on-board per il boot da 4 MB e uno slot per una carta Micro SD per il salvataggio dei dati.

Nel miniPC, sono disponibili parecchi modi diversi per interfacciarsi con l'esterno; in particolare esso dispone di: porta Ethernet, porte USB, porte a I2C, seriali UART e SPI (implementate dall'FPGA).

Caratteristica non meno importante del dispositivo è rappresentata dalla bassa potenza consumata; esso infatti consuma 400mA a 5V.

Il TS-7500 ARM Single Board Computer (SBC) è compatibile con un ampio range di sistemi operativi; in questo caso è stato utilizzato con una distribuzione Unix in modo tale da garantire un completo approccio open-source. Nel dispositivo è installato un Kernel Linux 2.6 con distribuzione *Debian* (version Lenny Arm) operante *out-of-the-box*; tale Kernel include una serie di driver per l'utilizzo dell'hardware on-board in modo da essere utilizzato per più applicazioni possibili.

Il compatto filesystem iniziale è basato su Busybox e uClibc ed è progettato per migliorare performance e flessibilità. I TS-7500 SBC sono configurati per caricare il sistema operativo Debian dalla carta SD durante la fase di boot; le grosse potenzialità nell'utilizzo di un sistema operativo di questo tipo si riflettono nella possibilità di disporre di un completo ambiente di sviluppo embedded GNU in C/C++. In aggiunta a questo, altri servizi di notevole importanza sono disponibili dal sistema quali: Apache Web Server, FTP, SSH, Telnet and Samba.

Per una maggior consultazione delle caratteristiche di questo prodotto, ci si riferisca a [39].

L' ARM è montato su un supporto meccanico provvisto di un pò di elettronica per offrire, oltre alla copertura del processore, altre utili funzionalità come per esempio: un interfaccia RS232, 3 relè, una porta RS485, 8 Buffered Inputs e 3 Buffered Outputs. La figura 4.3 mostra dunque il miniPC TS-7500 integrato nella sua copertura.

4.2 Software

In questo paragrafo verrà analizzato tutto il software progettato e funzionante nel processore ARM9 del minipc; partendo dapprima da una visione più estesa dell'architettura per passare successivamente ad una analisi più dettagliata.



Figura 4.3: Immagine del miniPC (con scatola semi-aperta)

Architettura

Il progetto di tesi è stato sviluppato utilizzando più linguaggi di programmazione, partendo da un livello di astrazione piuttosto basso fino ad arrivare a un linguaggio basato su oggetti ed altre strutture astratte.

Il sistema può così essere pensato suddiviso in più parti: la prima che comprende tutto il codice e le risorse per l'immagazzinamento dei dati proveniente dalla rete mentre la seconda implementa tutte le funzioni necessarie per visualizzare tali dati in una pagina web.

La parte in comune tra queste due appena accennate è rappresentata dal database utilizzato per il salvataggio dei dati e dei parametri di configurazione.

La figura 4.4 mostra una rappresentazione a blocchi dell'architettura del software implementata dal miniPC.

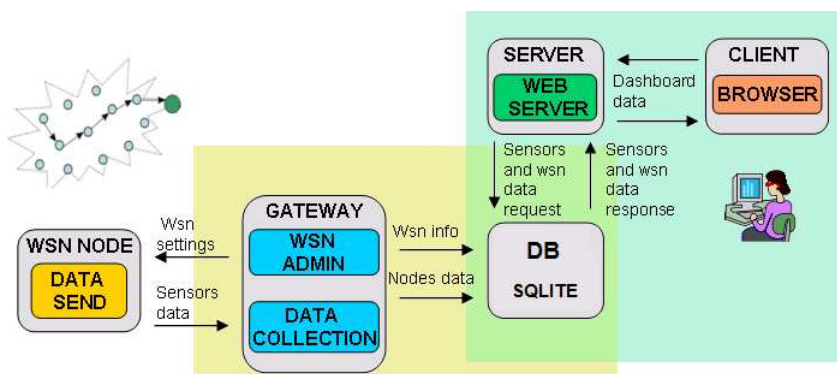


Figura 4.4: Architettura software del sistema

Un'analisi più dettagliata delle varie componenti del software sarà effettuata nei prossimi due paragrafi descrivendo dapprima la parte di software relativa all'immagazzinamento dei dati provenienti dalla rete (4.3) e successivamente quella relativa alla visualizzazione dei dati nella pagina web (4.4).

4.3 Dalla rete al database

Come anticipato, in questo paragrafo si andranno ad esporre le varie caratteristiche e funzionalità della raccolta e immagazzinamento dati provenienti dalla rete. Il collegamento fisico tra il nodo coordinatore e il miniPC è rappresentato da un cavo USB con un convertitore seriale racchiuso al suo interno; la comunicazione, grazie alle potenzialità del sistema operativo Unix, avviene dunque per via seriale mediante protocollo UART [34]. La scrittura e lettura avvengono infatti per ‘caratteri’ che verranno interpretati dal programma come dati provenienti da determinati nodi e immagazzinati nel database.

Per fare in modo di eseguire le operazioni appena elencate, si è pensato di sviluppare un programma scritto in linguaggio C; linguaggio che bene si adatta alle caratteristiche dell’hardware e del sistema operativo utilizzato.

Le funzioni principali di tale programma sono le seguenti:

- Apertura del database
- Inizializzazione delle periferiche USB e del buffer UART
- Interpretazione dati provenienti dall’ UART
- Salvataggio dei dati nel database

Per analizzare con più facilità il modo di funzionamento del programma si può far riferimento al diagramma di flusso presentato in figura 4.5. Si è pensato di utilizzare un sistema di tipo ‘*polling*’ per la lettura del dato proveniente dall’UART; con questo sistema si ha un controllo del dato in arrivo dal coordinatore ad ogni secondo dopodichè il processore si addormenta. Questo sistema è realizzabile grazie al fatto che il protocollo UART abbinato all’hardware scelto, permette che i dati, una volta giunti in ingresso, vengano memorizzati in un buffer di memoria finchè non vengono letti, così che se arriva più di un dato nello stesso istante non vi siano problemi di concorrenza.

4.3.1 Protocollo UART implementato

Per il passaggio dei dati dal coordinatore al miniPC è stato implementato un protocollo ad hoc per questo tipo di applicazione. Come già detto, i dati arrivano sotto forma di singoli caratteri che devono essere interpretati e salvati; quando il coordinatore riceve un dato da un certo nodo, le sue informazioni vengono spedite sotto forma di stringa strutturata nel seguente modo:

“t--u--l--C---a-- ...”

in cui i trattini sono sostituiti dai valori delle varie grandezze. In particolare quelli successivi al carattere:

- t rappresentano il valore di temperatura,

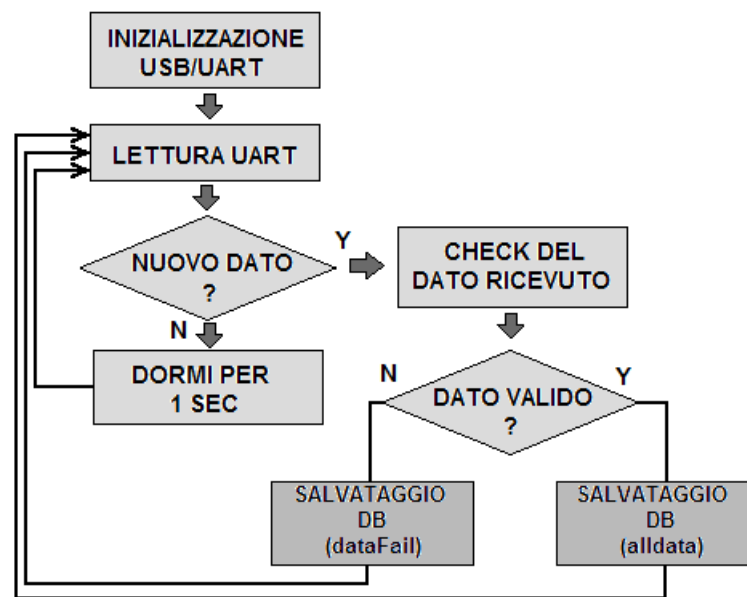


Figura 4.5: Diagramma di flusso del programma in C per l’acquisizione e il salvataggio dei dati dalla rete

- u rappresentano l’umidità,
- l rappresentano la luce,
- C rappresentano la concentrazione di CO₂,
- a rappresentano il numero identificativo del nodo al quale appartengono i valori in questione.

Il numero di cifre della stringa può variare da caso a caso, a seconda del numero di cifre di ogni singolo valore di grandezza; si riporta di seguito un esempio di una stringa inviata tramite UART dal coordinatore al miniPC:

“t21u48l8C1287a4”

la quale rappresenta le grandezze campionate dal nodo numero 4 (temperatura = 21°C, umidità=48% , luce=8 CO₂ = 1287 ppm).

Il programma in C legge la stringa carattere per carattere riuscendo ad estrapolare i vari dati e salvarli in una ‘struttura’ contenente tutti i campioni raccolti in un dato istante da un nodo. Dopo un primo utilizzo di tale protocollo in una trasmissione seriale ci si è accorti dell’esistenza di una piccola probabilità di una non corretta interpretazione nella lettura della stringa. L’utilizzo di una funzione di check è stato necessario per evitare inconvenienti nella lettura; solamente dopo questo controllo è possibile il salvataggio del dato nel data base.

4.3.2 Scelta e caratteristiche del Database

Il primo passo per l'implementazione del database è stato scegliere quale DBMS(Database management system) utilizzare. Il database ideale avrebbe dovuto avere le seguenti caratteristiche (oltre a quelle tipiche di un database):

- **leggerezza:** è assolutamente importante che il database occupi pochissime risorse hardware, sia per l'installazione sia durante il suo funzionamento;
- **buone performance:** dato il contesto in cui il database viene adoperato, le tabelle assumono abbastanza velocemente grandi dimensioni. Il database deve essere in grado di rispondere comunque in tempi ragionevoli alle query che riceve;
- **disponibilità API:** l'esistenza di API complete e ben documentate, preferibilmente per il linguaggio C e php, consente uno sviluppo più rapido dell'applicazione;
- **open source:** ovviamente, come quasi tutto il software in questo progetto, il DBMS deve essere open source e deve essere installabile su un sistema Linux.

Dunque, caratteristiche come la gestione avanzata di più utenti o funzioni di sicurezza particolari non vengono richieste in quanto sono funzioni svolte dai livelli gerarchici di sistema più alti (webserver). Dati quindi questi vincoli, sono stati scartati i più diffusi DBMS come PostgreSQL e MySQL, poiché contengono tante funzionalità avanzate che non servono al progetto in questione e, inoltre, occupano molto spazio in memoria in fase di installazione. In particolare, MySQL occupa quasi 100 Mb, mentre PostgreSQL circa 20 Mb ma è più lento di MySQL. Alla fine la scelta è caduta su SQLite [31], che è proprio il database con le caratteristiche ideali per il sistema realizzato.

SQLite

SQLite è un database relazionale embedded open source. È stato sviluppato con lo scopo di fornire un modo comodo di gestire dati senza l'overhead introdotto dai classici database relazionali. SQLite è famoso per la sua alta portabilità, la facilità di utilizzo, la compattezza, l'efficienza e l'affidabilità. Esattamente le caratteristiche del database cercato.

SQLite è di tipo embedded, ovvero non gira come un'applicazione indipendente, bensì coesiste all'interno dell'applicazione che serve. Quindi, il suo codice è integrato come una parte del programma che lo ospita. Il vantaggio che ne deriva è che server e client sono nello stesso processo e non richiedono dunque configurazioni di rete. SQLite offre molte funzionalità, nonostante le sue dimensioni di soli 250 KB. Supporta un gran numero di query ANSI SQL92 (transazioni, viste, controlli sui dati, query composte), trigger, indici, colonne con autoincremento, clausole LIMIT e OFFSET.

I principi seguiti da sqlite sono principalmente:

- **Nessuna configurazione:** è assente qualsiasi forma di Database Administration; tutto ciò che serve è incluso nel nucleo del programma.
- **Portabilità:** SQLite gira sotto qualsiasi piattaforma che sia Linux, Windows, Mac, Solaris, Symbian o Palm Os. Inoltre funziona su architetture a 16, 32 e 64 bit. La portabilità poi riguarda non solo il software ma anche il file del database. Infine, SQLite può contenere fino a 2 TB di dati (se il sistema operativo lo permette).
- **Compattezza:** SQLite è stato progettato per essere piccolo e leggero: infatti comprende solo un file header e una libreria e il tutto non occupa più di mezzo megabyte. Altrettanto compatti sono i database creati, che sono dei normali file del sistema operativo: tutti gli oggetti, come tabelle, trigger, schemi, indici e viste sono contenute all'interno di un singolo file. Inoltre, i record sono a lunghezza variabile, quindi viene allocato solo lo spazio necessario per mantenere il dato del campo.
- **Semplicità:** l'API di SQLite è semplice e facile da utilizzare, ben documentata e intuitiva; grazie a ciò, una vasta community sviluppa in continuazione nuove librerie.
- **Nessun costo di licenza:** tutto il codice è di pubblico dominio.
- **Affidabilità:** tutto il codice in SQLite viene ampiamente testato.
- **Convenienza:** SQLite ha una serie di caratteristiche, come la possibilità di aggiungere diversi database ad una singola sessione, che lo rendono adatto a molti scopi.

In termine di prestazioni, SQLite è tanto veloce, se non più veloce, dei più diffusi DBMS, nell'effettuare operazioni tipo INSERT, UPDATE o SELECT su una tabella. Se invece si utilizzano query molto complicate, allora SQLite inizia a mostrare i suoi limiti e i normali database relazionali rispondono più agilmente.

I limiti di SQLite sono essenzialmente tre:

- **Concorrenza:** il sistema permette più entità in lettura, ma solo una in scrittura; le operazioni di scrittura bloccano il database.
- **Dimensione del database:** anche se la dimensione massima è di 2 TB, avere database molto grandi costa in termine di occupazione della RAM. Ad esempio, un database di 100 GB richiede l'allocazione di 25 MB di RAM prima di ogni transazione.
- **Networking:** il supporto di rete è assente. Sebbene i database possano essere condivisi su Networking File System, le prestazioni con questi sistemi possono essere basse a causa della latenza introdotta da tali file system.

Le specifiche di tale progetto di tesi, fanno sì che tali limiti non risultino essere un problema.

4.3.3 Utilizzo del Database

Il database creato in questo progetto di tesi, viene utilizzato sia per il salvataggio dei dati che per i parametri per la configurazione. Per tale motivo sono state create più tabelle all'interno dello stesso database con compiti ed impostazioni diverse; esse sono elencate di seguito e verranno analizzate di seguito per capirne le funzionalità principali:

- **dataNow**
- **alldata**
- **dataFail**
- **config**

Tabella: dataNow

Rappresenta la tabella nella quale vengono salvati tutti gli ultimi dati raccolti dai vari nodi, contiene dunque solamente i dati più aggiornati provenienti dai nodi della rete; i dati sono inseriti mediante funzione di UPDATE. Le informazioni raccolte riguardano *numero identificativo del nodo, temperatura, umidità, luce, CO2 e ora del campionamento*. Di seguito è riportata la struttura del database dataNow:

```
CREATE TABLE dataNow(  
    id INTEGER PRIMARY KEY,  
    num_nodo INT,  
    temp INT,  
    umid INT,  
    luce INT,  
    co2 INT,  
    ora INT  
)
```

Per meglio gestire date e orari di campionamento, si è utilizzato il formato tipico dei sistemi UNIX per il salvataggio dell'ora. Si tratta di un formato standard in cui l'orario viene espresso sottoforma di numero intero (INTEGER); esso rappresenta un *offset* in secondi rispetto alla mezzanotte (UTC) del 1° gennaio 1970 (detta *epoca*) [40].

Questo tipo di rappresentazione, oltre che ad essere compatta, è indipendente dai fusi orari, ed è quindi direttamente confrontabile anche tra calcolatori situati a grandi distanze geografiche tra loro, ed evita di dover effettuare aggiustamenti nel caso ad esempio di dati trasmessi da un fuso orario all'altro.

Tabella: alldata

Contiene le stesse informazioni della tabella *dataNow* ma i dati non vengono aggiornati di volta in volta bensì vengono inseriti mediante una funzione di INSERT. Tale tabella contiene dunque tutta la cronologia dei dati raccolti e risulta essere di dimensioni notevoli (dopo pochi giorni di campionamento presenta decine di migliaia di record salvati). È stata utilizzata l'impostazione di CREATE INDEX sul campo relativo l'ora, come verrà descritto in seguito ciò renderà le operazioni di SELECT (necessarie per la visualizzazione del grafico) nettamente più veloci. La struttura della tabella *alldata* è la stessa di quella della tabella *dataNow*.

Tabella: dataFail

Come già accennato alla fine di 4.3.1, prima del salvataggio dei dati nel database, la loro validità è verificata onde evitare che dati letti non correttamente siano salvati nel database. Se il dato letto è considerato valido, viene salvato nelle due tabelle precedentemente descritte (*dataNow* e *alldata*) in caso contrario, viene salvato in un'altra tabella chiamata appunto *dataFail*. In questo modo, l'utilizzatore del sistema può, in caso di mal funzionamento, controllare l'istante in cui è avvenuta l'errata lettura e quale campo è stato male interpretato. La struttura della tabella *dataFail* è la stessa di quella della tabella *dataNow*.

Tabella: config

Questa tabella è stata creata per contenere tutti i parametri relativi alla configurazione della rete, e all'operatività del miniPC. In questo progetto è stata utilizzata per contenere solamente il periodo di campionamento dei nodi; tale valore è aggiornato dal client e successivamente spedito ai vari nodi della rete dal coordinatore.

La struttura di tale tabella è del tipo

```
CREATE TABLE config(  
    id INTEGER PRIMARY KEY,  
    sample_time INT,  
    ...  
)
```

4.4 Dal database al client

Una volta che i dati sono salvati nel database, è necessario l'utilizzo di una serie di strumenti per renderli disponibile all'utilizzatore (lato *client*). Si è pensato di caricarli su una pagina web visualizzabile dai più comuni web browser. Per il progetto in questione si è utilizzata una pagina web dinamica cioè il cui contenuto è, in tutto o in parte, generato sul momento dal server e può essere quindi diversa

ogni volta che viene richiamata. Questo significa che non si utilizza direttamente il linguaggio HTML, ma che si ricorre a dei linguaggi di programmazione (i linguaggi di scripting) che si occupano della creazione della pagina nel momento in cui questa viene visitata, anche interagendo con i visitatori, e possono variare l'output HTML successivo dopo calcoli eseguiti con apposite strutture messe a disposizione dal linguaggio di scripting che si usa.

In questo progetto è stato utilizzato uno script in *php* per la gestione delle query al database; richieste utilizzate sia per ottenere i dati dal database che per aggiornare il valore di periodo di campionamento. Una volta effettuata la query, i dati vengono convertiti in una struttura XML per poi essere elaborati da un programma scritto in Adobe Flex per la visualizzazione finale sul browser.

L'analisi di questo sistema è l'argomento di questo paragrafo partendo dalla scelta del web server sino alla visualizzazione finale al client.

4.4.1 Web Server

Il Web Server funge da intermediario tra l'utente (client) e il database del miniPC. Quindi il suo compito è di prelevare i dati dal database, elaborarli e mandarli all'utente. Più precisamente, si occupa di:

- richiedere al database i dati delle rilevazioni più recenti o quelle specificate dall'utente;
- salvare nel database alcuni parametri stabiliti dall'utente come un nuovo periodo di campionamento dei sensori;
- generare dati XML;
- autenticare gli utenti.

4.4.2 Scelta del Web Server

La scelta del web server è dipesa dai vincoli di leggerezza, compattezza, performance e tipo di licenza. Avere un web server leggero e compatto significa poterlo installare su macchine molto vecchie o addirittura su processori ARM come quello utilizzato, il che vuol dire risparmiare sul costo totale del progetto.

Esistono diverse soluzioni web server, di cui quella open source più famosa è Apache. Apache offre molte potenzialità ed è stato quindi il candidato per questo progetto. Le funzionalità disponibili da questo tipo di web server sono:

1. Autenticazione
2. Negoziazione dei contenuti in base alle capacità del client
3. Virtual hosting (più siti Web sullo stesso server)
4. Personalizzazione di logfile e messaggi di errore

5. Possibilità illimitata di URL rewriting, aliasing e redirecting. Con tale strumento web si può ottenere una manipolazione e riscrittura flessibile dell'URL.

Molte di queste funzioni sono completamente trasparenti al client una volta che il web server è configurato correttamente.

Per queste caratteristiche, Apache ha costruito attraverso gli anni la sua fama di web server d'eccellenza. Il vasto numero di moduli ad estensione delle funzionalità di base del demone, la grande affidabilità e la facilità di implementazione di infrastrutture equipaggiate con Apache, ne hanno fatto una scelta affidabile.

Apache presenta tra i vari punti di forza la modularità, con un numero di estensioni veramente molto vasto, utilizzabili per qualsiasi tipo di esigenza, che sia il proxying, l'autenticazione, il caching, il virtual hosting, ecc. . . Apache viene utilizzato da più di dieci anni sui web server di tutto il mondo e si è costruito una reputazione di affidabilità e solidità, oltre ad essere facilmente configurabile, con una sintassi chiara ed esplicativa. Non ultimo è inoltre accompagnato da un'ottima e ben fornita documentazione.

Quello che potrebbe essere un punto debole di questo web server, è il notevole consumo di memoria del demone; Apache è infatti basato su un'architettura multithread nella quale, ogni singolo thread rappresenta un singolo processo che viene diviso ed eseguito in modo parallelo e concorrente ad ogni richiesta effettuata. E' proprio questo tipo di esecuzione che coinvolge l'utilizzo di un certo quantitativo di memoria².

Per un approfondimento sul principio di funzionamento di Apache vedere [32].

4.4.3 Programma in php

PHP (PHP Hypertext Preprocessor) [33] è un linguaggio di scripting interpretato con licenza open source adatto a sviluppare applicazioni web lato server, anche se può essere usato per scrivere script a linea di comando o applicazioni *standalone* con interfaccia grafica.

Nel sistema realizzato è servito per costruire diverse applicazioni necessarie per far interagire il client con i dati memorizzati nel miniPC. Di seguito sono riportate le funzioni principali svolte dal programma php nel progetto:

- Lettura dei dati del database
- Salvataggio dei dati
- Conversione dei dati in formato XML
- Gestione delle scadenze delle pagine

²Altri web server come Ngin e Lighttpd sono stati sviluppati negli ultimi anni per andare in contro a sistemi che esigono un consumo di memoria minore. Laddove Apache è basato su un'architettura process-based e necessità di 1 thread per ogni richiesta da soddisfare, Nginx e Lighttpd sono basati su un'architettura che consente di servire più richieste con uno solo o con pochissimi thread sfruttando le loro caratteristiche di asincronicità.

4.4.4 Flex Builder

Per la visualizzazione dei dati su una pagina web è stato utilizzato come strumento di design e di gestione dei dati della pagina un'applicazione in Flex; esso un framework open source gratuito ad alta produttività per la creazione di applicazioni web interattive implementabili uniformemente su tutti i principali browser, desktop e sistemi operativi, con l'ausilio dei runtime Adobe, Flash Player e Adobe AIR. L'idea fondamentale è quella di ottenere applicazioni molto gradevoli sotto il profilo dell'esperienza utente, sfruttando la potenza del formato SWF, minimizzando il lavoro di sviluppo e fornendo un modello di programmazione noto agli sviluppatori di applicazioni classiche.

Flex utilizza un approccio allo sviluppo basato su componenti già pronti. I componenti si possono personalizzare o creare ex novo con ActionScript 3 e si possono utilizzare dichiarandoli in documenti MXML, un formato proprietario Adobe di documenti XML per la definizione di interfacce utente.

L'MXML viene usato per definire anche aspetti non visuali: accesso a dati in remoto, data bindings e comportamenti (behaviors). Costruire un'applicazione Flex con MXML presenta delle similitudini con la progettazione con HTML, attraverso l'uso di tag che descrivono componenti e comportamenti. La differenza tra i due linguaggi sta nel fatto che l'HTML è interpretato dal browser mentre l'MXML viene compilato per generare un swf che a sua volta viene eseguito dal Flash Player.

I file MXML seguono le regole dei file XML ben formati: presenza di un elemento radice, corretto annidamento degli elementi, chiusura obbligatoria dei tag vuoti. Lo scheletro di un file MXML contiene: una dichiarazione di un documento XML, un root tag Application che funge da contenitore di più alto livello.

Le applicazioni sviluppate con Flex vengono eseguite all'interno di un comune browser ma sfruttano la potenza del runtime environment del Flash Player 9 per eseguire la logica client side, effettuare il rendering degli elementi grafici e visuali, riprodurre animazioni, audio e video.

Per questo progetto è stato utilizzato uno strumento di sviluppo per Flex basato sul noto Eclipse: *Adobe Flex Builder*. Flex Builder consente agli sviluppatori di creare in modo rapido anche le più complesse applicazioni che integrano XML, Web Service o LiveCycle Data Service. Gli avanzati strumenti di progettazione e layout aiutano gli sviluppatori a creare interfacce utente personalizzate, semplici da utilizzare e arricchite da elementi multimediali come audio e video.

I compiti principali del programma sviluppato con Flex Builder riguardano:

- la *progettazione del layout* della pagina web,
- la *gestione delle richieste* alla pagina php.

Layout e richieste sono sviluppate all'interno del programma in modo piuttosto indipendente; in particolare mentre il layout è scritto in MXML, la parte di programma relativa le richieste è scritta in Action Script.

Progettazione del layout

La pagina web è stata pensata per offrire il massimo della semplicità da parte dell'utente di reperire le informazioni dei dati della rete. La figura 4.6 mostra la pagina web e la sua struttura; per garantire un facile utilizzo della tale si è utilizzata una struttura a schede (tab). La pagina presenta cinque schede, nel particolare³:

1. Grafici. Dopo aver selezionato un nodo della rete, permette la visualizzazione dei dati di quest'ultimo sul grafico in due modalità diverse: in tempo reale (*'Tempo Reale'*) e per un certo intervallo temporale (*'Intervallo Temporale'*); nel primo caso è possibile scegliere quanti minuti devono essere mostrati mentre nel secondo va impostato un intervallo temporale selezionando data, ore e minuti.
2. Ultimi dati. Contiene una tabella con i dati più aggiornati dalla rete mostrando il numero del nodo, temperatura, umidità, luce, CO2 e l'orario di acquisizione.
3. All data. Contiene anch'essa una tabella ma a differenza della scheda precedente, in questa vengono caricati tutti i dati mostrati nel grafico relativi ad un singolo nodo. La dimensione di questa tabella dipende dunque principalmente dall'intervallo temporale specificato.
4. Data Fail. Contiene una tabella che raccoglie tutti i dati letti non correttamente dal miniPC.
5. Configurazione. Rappresenta l'interfaccia grafica per modificare alcuni parametri relativi la rete, il database e non solo. In particolare, da qui si può modificare il periodo di campionamento dei nodi della rete, il periodo del timer per la visualizzazione del grafico e si può andare a cancellare il contenuto dei database.

Gestione delle richieste

Come già detto, i dati sono estrapolati dalle tabelle del database grazie ad uno script in linguaggio php; è sufficiente effettuare delle richieste a tale script per ottenere i risultati da visualizzare nella pagina web. In particolare, le richieste al programma in php sono accompagnate da alcuni parametri passati tramite metodo GET [36] necessari per far eseguire al web server compiti diversi.

Per come è stato progettato il sistema, il programma php può ricevere 6 tipi diversi di richiesta da parte del programma swf di flex; ogni richiesta viene interpretata in modo diverso e viene seguita da una query specifica al database, i dati estrapolati sono ritornati in formato XML al programma swf. Nel dettaglio:

³Non si vuole ora entrare nel dettaglio allo sviluppo di questi tab in quanto la loro progettazione non ha avuto grosse richieste dal punto di vista ingegneristico.

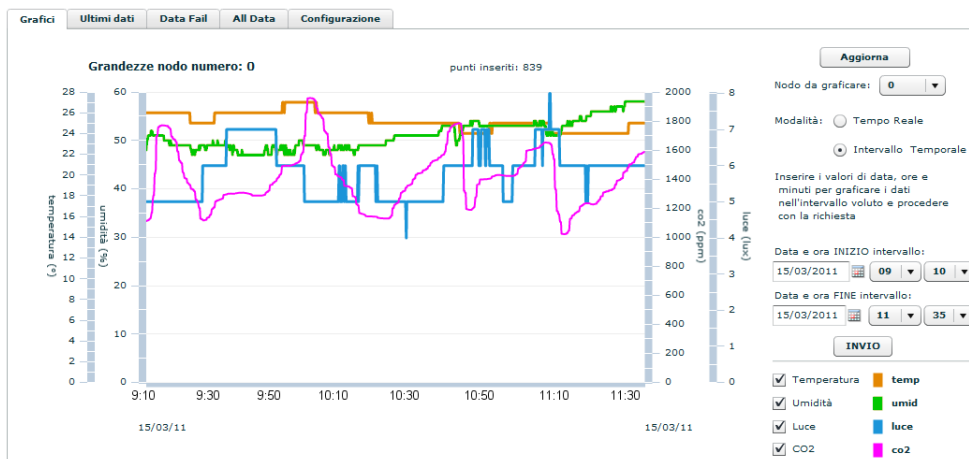


Figura 4.6: Design della pagina web progettata

1. **Aggiornamento tabella Ultimi dati.** Tale richiesta permette la visualizzazione del contenuto della tabella del database *dataNow* nella griglia della scheda *Ultimi dati*; in questo caso la pagina php quando riceverà tale richiesta effettuerà una query al database richiedendo con un `SELECT` tutto il contenuto della tabella *dataNow*. Dalla riga del browser, la richiesta è del tipo:

`http:// ... /filephp.php?first=sort&second=num12`

con *first* e *second* le due variabili per il metodo GET. Il valore *num* può essere sostituito da altri campi (*temp,umid,luce,co2*) che rappresentano il principio di ordinamento dei dati della tabella.

2. **Grafico (Tempo reale).** Per ottenere la visualizzazione del grafico in modalità *Tempo reale* (precedentemente discussa) è necessario effettuare una richiesta alla pagina php con due campi aggiuntivi: il primo che rappresenta il nodo della rete preso in esame mentre il secondo che indica il valore di data in formato UTC del primo dato del grafico. Quest'ultimo valore è calcolato dal valore della data espressa in formato UTC del dato più recente sottratto il numero di minuti impostati dal browser per la visualizzazione.

La pagina php effettuerà una query nella tabella *alldata* richiedendo i dati del nodo voluto e con orario di campionamento non inferiore a quello passato dal terzo campo del POST. In questo caso la richiesta è del tipo:

`http:// ... /filephp.php?first=graph_req&second=1&third=1297848200`

in cui è richiesta la visualizzazione dei dati campionato dal nodo numero 1 dall'ora 1297848200 sino a quella del dato più recente salvato nel database.

3. **Grafico (Intervallo Temporale).** Tale richiesta richiede l'aggiunta di un ulteriore campo rispetto a quella precedente in quanto l'intervallo di visualizzazione necessita sia del momento iniziale che di quello finale usando una richiesta del tipo:

.../filephp.php?first=graph_req2&second=1&third=1297848200&fourth=1297850640

NOTA: L'utilizzo del campo *CREATE INDEX* nella colonna della data nella tabella *alldata* è necessario per velocizzare notevolmente le query appena discusse. In questa maniera i dati sono già ordinati nel tempo e le condizioni nell'orario in fase di *SELECT* risultano più agevoli e leggere.

4. **Aggiornamento tabella data Fail.** Per aggiornare la tabella relativa ai dati letti non correttamente viene effettuata una richiesta del tipo:

http:// .../filephp.php?first=dataFail

alla quale il programma php farà seguire una richiesta alla tabella *dataFail* del database. Anche in questo caso i dati sono convertiti in una struttura XML e la tabella nella pagina web verrà aggiornata.

5. **Aggiornamento periodo di campionamento.** Dalla pagina relativa la configurazione del sistema è possibile inserire un nuovo periodo di campionamento dei sensori da parte dei nodi della rete; quando è premuto il tasto *invia*, viene inoltrata una richiesta del tipo:

http:// .../filephp.php?first=set_sample&second=15

in cui il valore *second* rappresenta il nuovo periodo; tale valore andrà a sostituire quello già salvato nella tabella config del database mediante una richiesta di *UPDATE*.

6. **Cancellazione tabelle database.** La possibilità di cancellazione delle tabelle del database è stata pensata per non dover agire direttamente sul miniPC; dalla pagina di configurazione è sufficiente selezionare la tabella che si vuole cancellare e dare conferma. Da ciò partirà una richiesta del tipo:

http:// .../filephp.php?first=delete&second=dataFail

in cui il parametro *second* è una stringa contenente il nome della tabella della quale si vuol effettuare il refresh.

Ogni richiesta relativa ad una lettura nel database, è gestita dall'utilizzo di alcuni timer gestiti via software; nel particolare, il programma in flex gestisce 3 timer uno per l'aggiornamento della tabella degli ultimi dati, uno per i grafici ed uno per la tabella dei data fail. Nel programma implementato, è possibile gestire il periodo del timer relativo l'aggiornamento del grafico; tale opzione è resa necessaria in quanto per quantità di dati piuttosto importanti, qualche migliaia di punti, la visualizzazione del grafico non risulta immediata. Ciò è dovuto al tempo necessario al programma in php di interrogare il database; tempo che può essere anche di qualche secondo. Risulta quindi inutile aggiornare il grafico effettuando una richiesta ogni 10 secondi quando il programma php ne impiega magari 15 per ricevere i dati dal database; in questo caso il problema risulterebbe risolto se il periodo di aggiornamento venisse modificato a 20 secondi.

Capitolo 5

Conclusioni

5.1 Altri risultati ottenuti

In aggiunta alla raccolta dei dati negli uffici di Inthebra, si è pensato di utilizzare il sistema progettato per effettuare alcune prove in alcune aule di scuola primaria de Veneto.

5.1.1 Aula di scuola elementare

Negli ultimi anni, sono stati svolti molti studi riguardanti le problematiche che potrebbe avere il corpo umano dovute ad alti livelli di anidride carbonica. Tale analisi si è focalizzata per buona parte sugli effetti di CO_2 in ambienti confinati, principalmente nelle aule di scuola. In tali locazioni, si ha la presenza di un importante numero di persone, mediamente 15-25 compreso il docente, e spesso i valori volumetrici degli spazi sono piuttosto ristretti. Questo, in aggiunta al fatto che la salute dei bambini di giovane età ha un'importanza fondamentale, ha portato molti gruppi di ricerca a svolgere degli studi a proposito.

Come esempio, l'istituto di Fisiologia Clinica CNR dell'università di Pisa ha svolto uno studio sulle aule europee [29]. Da tale analisi, si è visto come il valore standard suggerito per esposizione a lungo termine (1000 parti per milione) viene superato nel 66 per cento delle aule europee coinvolte, con Italia, Francia e Danimarca prime a quasi 1900 ppm, cioè quasi al doppio della soglia di sicurezza. In aggiunta si è vista una correlazione tra le concentrazioni di PM10 [30] e CO_2 , cioè all'aumentare di un inquinante corrisponde un aumento dell'altro.

Oltre a far diminuire il livello di concentrazione cognitiva degli occupanti dell'aula (studenti e docenti), un alto livello di CO_2 (>1000ppm) è responsabile di alcune patologie delle vie respiratorie; sempre dallo studio svolto dall'università di Pisa [29], l'anidride carbonica aumenta il rischio di tosse secca e rinite.

Dopo aver capito l'importanza dei livelli di CO_2 nelle locazioni scolastiche, si è pensato di utilizzare il sistema progettato in questo lavoro di tesi per monitorare i livelli di anidride carbonica in alcune aule del Veneto. Lo studio si è svolto su due

aule dalle caratteristiche completamente differenti, come si vedrà successivamente, la prima studiata (Aula A) presentava dimensioni piuttosto ridotte con un numero di studenti contenuto; al contrario, la seconda (Aula B), ospitava un numero di studenti più numeroso, tipico per una classe elementare italiana.

L'aula A appartiene ad una scuola categorizzata 'di montagna' posta ad un'altitudine di 900 metri slm; durante il periodo di test la temperatura esterna era di poco meno di 10 °C con un'umidità del 30% circa. Differentemente da A, la scuola in cui è stato effettuato il secondo monitoraggio, si trova in pianura e le condizioni dell'ambiente esterno nel momento in cui il test è stato svolto erano tipicamente 'primaverili' con temperatura di 16°C e umidità di del 30%.

5.1.2 Aula A

Le caratteristiche basilari dell'ambiente in cui il test è stato svolto sono riportate in tabella 5.1.

Dimensioni	Num. occupanti	Ora inizio	Ora fine	Ora intervallo
$6 \times 4 \times 4 = 96m^3$	11 + 1 docente	8:30	12:30	10:30 - 11:00

Tabella 5.1: Caratteristiche dell'aula A

Si può notare dai valori della tabella che il numero di studenti in questione è piuttosto ridotto. Non per questo, dai risultati ottenuti, sono stati riscontrati livelli di anidride carbonica piuttosto elevati.

Le seguenti due immagini, raffigurano i grafici ottenuti durante un giorno intero di monitoraggio. In particolare, la figura 5.1 mostra il risultato del monitoraggio complessivo effettuato dalle ore 8:00 del 28/2/2011 alle ore 8:00 del 1/3/2011.

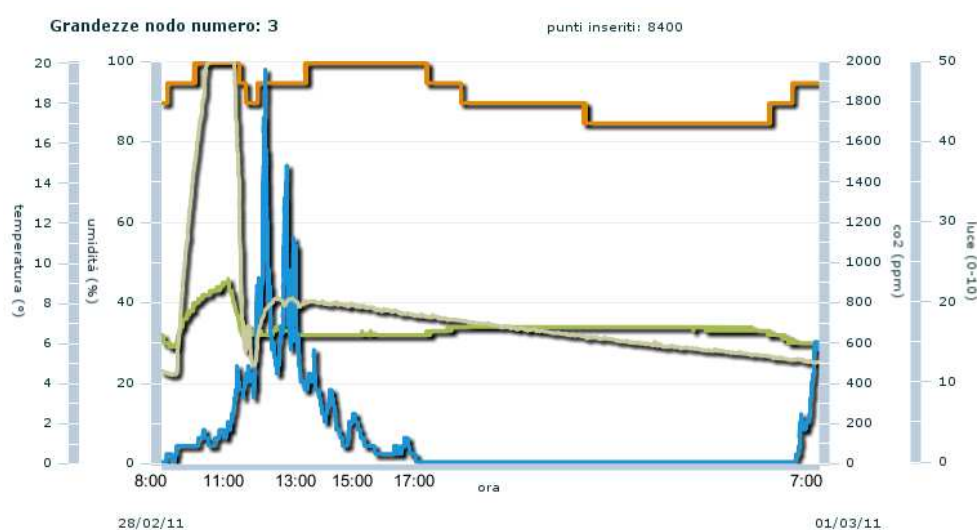


Figura 5.1: Grafico di una giornata intera di monitoraggio nell'aula A

Per analizzare nel dettaglio i dati ottenuti, la figura 5.2 mostra appunto l'andamento dei dati monitorati solamente per l'intervallo di tempo per il quale l'aula è stata occupata (dalle 8:30 alle 12:30).

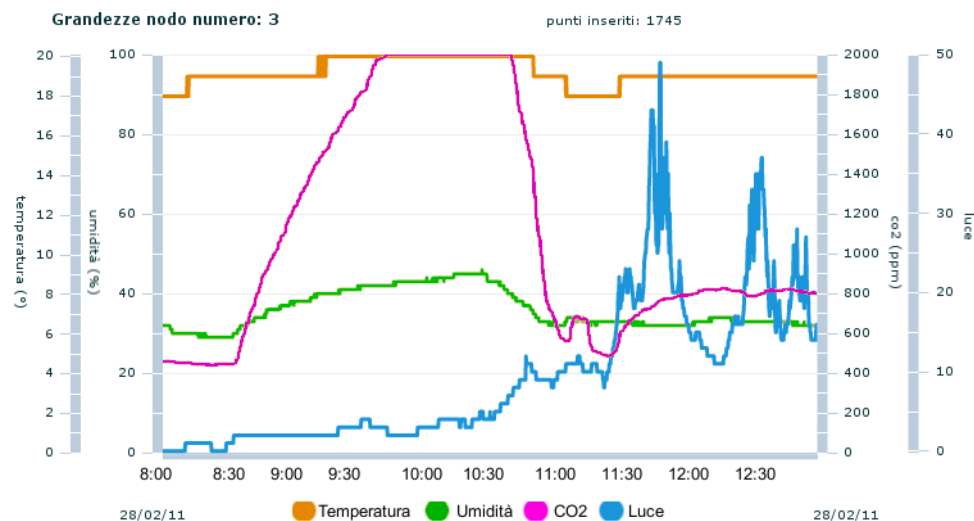


Figura 5.2: Grafico del monitoraggio nell'aula A durante l'orario scolastico (lezione + intervallo)

I risultati ottenuti sono, dal punto di vista del test, piuttosto interessanti. Si può notare come, il livello di CO₂ all'inizio del monitoraggio, quando l'aula non era ancora occupata, è piuttosto basso, paragonabile al livello di anidride carbonica dell'atmosfera (poche centinaia di ppm). Appena l'aula inizia ad essere occupata, attorno alle 8:30, si ha una variazione brusca nell'andamento sia della CO₂ che dell'umidità; anche uno scalino nei livelli di luce campionata rappresenta un'indicazione di occupazione dell'aula in quanto l'illuminazione a neon della stanza è stata attivata con l'inizio delle lezioni.

Durante il proseguo della mattinata, i dati si possono analizzare divisi in 3 finestre temporali:

- **Fase 1:** dalle 8:30 alle 10:30.
- **Fase 2:** dalle 10:30 alle 11:05.
- **Fase 3:** dalle 11:05 alle 12:30.

Per ognuna di queste fasi, le grandezze in gioco si comportano in modo diverso.

Fase 1

Durante questa fase l'aula è occupata con finestre e porte chiuse; si nota che tutti i valori misurati tendono ad aumentare, in particolare:

- Il valore di **temperatura** aumenta di un paio di gradi (informazione poco importante per stabilire l'occupancy)

- Il valore di **umidità** aumenta progressivamente da 30% fino ad un valore di circa 45%.
- Il valore di **luce** aumenta ma l'informazione relativa non è piuttosto affidabile in quanto il valore letto dal sensore dipende da una serie di fattori poco prevedibili ¹
- Il valore di **CO2** offre la più ricca informazione per identificare lo stato di occupancy dell'aula. Si può notare come esso aumenti in maniera esponenziale secondo la relazione 2.3 precedentemente studiata. Attorno le 9:45 del mattino, cioè dopo 1h 15 m dopo l'inizio delle lezioni, i valori di CO2 erano tali da far saturare il sensore utilizzato. Il sensore satura alla soglia dei 2000ppm, soglia sufficientemente alta per degradare il livello di attenzione e di salute degli studenti. Mediante un'interpolazione grafica, si può stabilire in maniera indicativa il massimo valore di CO2 raggiunto; esso si ottiene proprio alla fine della fase 1 attorno le 10:30. Così, se le linee della fase 1 e della fase 2 venissero prolungate, si otterrebbe un punto con ascissa proprio 10:30 e valore di CO2 di addirittura 2700 ppm.

Fase 2

In questo periodo temporale, l'aula viene liberata dagli studenti e le finestre vengono aperte arieggiando l'ambiente. Si nota con facilità:

- una diminuzione della temperatura in quanto la quella esterna era poco superiore ai 10 gradi,
- una diminuzione dei valori di umidità, da 45% a 30%,
- un ulteriore aumento della luce, si può ipotizzare infatti che sia dovuto sia al fatto che il sole durante tale fase è in ascesa ma anche perchè l'apertura delle finestre magari comporta un apertura delle tende,
- un crollo dei valori di CO2 da un ipotetico valore di 2700 ppm a valori tipici per esterni come 570 ppm; il tutto in una trentina di minuti (circa -70 ppm/min).

Fase 3

Gli studenti ritornano in aula dopo l'intervallo e le finestre vengono chiuse. In tale fase si nota subito un aumento dei valore di co2 seguito da un ulteriore calo per poi salire nuovamente. Ciò è dovuto al fatto che la porta dell'aula che dà sul corridoio è stata aperta attorno le 11:30 ed rimasta aperta per il proseguo della giornata; ciò ha comportato un notevole ricircolo d'aria che ha mantenuto

¹Il valore di luce dipende da dov'è posto il sensore, se la luce della stanza è accesa oppure no, dalla luce esterna, ecc. . .

l'anidride carbonica su livelli ottimali. I valori di temperatura e umidità rimangono pressochè costanti mentre si nota un brusco aumento della luce presente in aula dovuto al fatto che le finestre sono rivolte verso sud ed in questa fase il sole raggiunge il culmine massimo di luminosità.

Con il proseguire della giornata, si notano delle leggere variazioni di CO₂ dovute alla presenza saltuaria della bidella all'interno dell'aula.

5.1.3 Aula B

Quest'altro test è stato effettuato su un aula di scuola elementare dalle caratteristiche completamente differenti; si è passati da un aula di quasi una dozzina di alunni occupanti ad un'altra con un numero decisamente maggiore di studenti. Come è presentato nella tabella 5.2 il numero di studenti ora è di 20 unità, quasi il doppio rispetto al caso precedente.

Come si può immaginare, questo sarà frutto di un tasso di generazione di CO₂ maggiore, anche se, in questo caso, i livelli saranno limitati da una notevole dimensione dell'aula in esame.

Dimensioni	Num. studenti	Ora inizio	Ora fine	Ora intervallo
$7 \times 6 \times 3.5 \simeq 150m^3$	20 + 1 docente	8:15	16:30	10:30 - 11:00 13:00 - 13:30

Tabella 5.2: Caratteristiche dell'aula B

Si vuol ora analizzare meglio l'andamento delle grandezze ambientali raccolte durante l'arco della giornata scolastica; in tale test, è stato chiesto alle maestre di annotare dettagliatamente tutte le volte che l'aria all'interno dell'aula potesse subire un cambio (apertura delle finestre e delle porte). In questa maniera, i dati ottenuti possono essere commentati con più sicurezza senza la necessità di effettuare delle ipotesi.

Dalla figura 5.3 si nota l'andamento delle grandezze di temperatura, umidità e CO₂² per tutto l'arco della giornata scolastica³.

- Partendo dall'analisi della **temperatura** si nota come essa sia contenuta tra i 22 ed i 27 °C quest'ultimi raggiunti ad inizio mattinata e a fine giornata. Tali valori sono decisamente troppo elevati per una giornata di fine inverno; come si vedrà successivamente il sistema di riscaldamento dell'aula non è impostato correttamente ottenendo valori di temperatura sempre troppo elevati.

²Il valore di luce è stato tralasciato perchè di poca importanza in quanto il sensore è stato posto in un luogo piuttosto riparato per evitare il rischio di subire dei danni.

³A differenza dell'analisi del caso precedente, qui non si vuole entrare nel dettaglio suddividendo la giornata in più fasi.

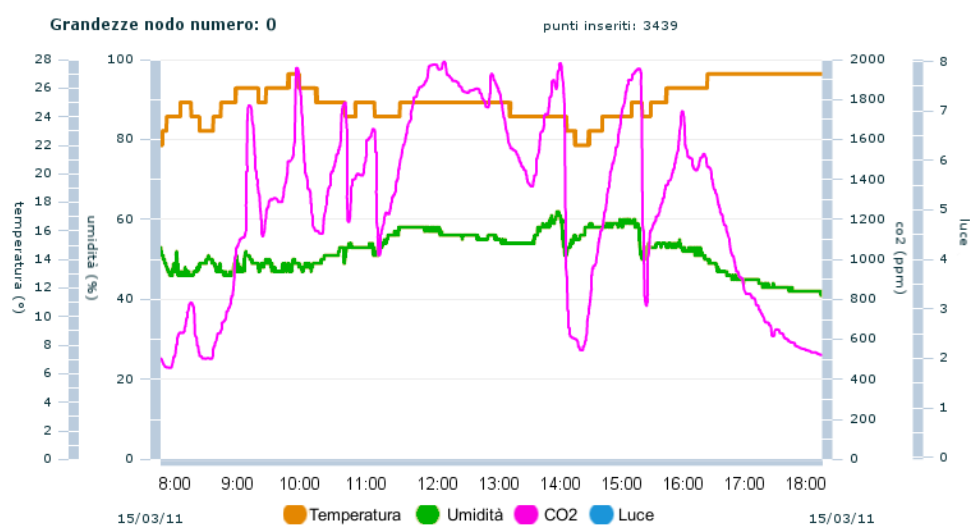


Figura 5.3: Grafico del monitoraggio nell'aula B durante l'orario scolastico (lezione + intervalli)

- Di primo impatto, il lettore può notare notevoli picchi nell'andamento nel grafico 5.3 della **CO₂**. Di primo mattino, è presente in aula una concentrazione di qualche centinaio di ppm per poi subire una grossa variazione con l'inizio delle lezioni (verso le 8:30 circa). Dalle 9:30 alle 11:00 si nota una serie di picchi causati dall'apertura delle finestre verso il cortile della scuola; le finestre vengono chiuse solamente dopo qualche minuto, sufficiente comunque per riportare la concentrazione di CO₂ da livelli alti (da 1600 a 1900 ppm) fino a livelli più accettabili (da 1000 a 1200 ppm).

Per il proseguo della giornata, i picchi si susseguono meno frequentemente con un periodo di apertura delle finestre maggiore, da notare il picco delle ore 14:00 in cui l'anidride carbonica crolla da un valore di quasi 2000 ppm (fondoscala del sensore) sino a 550 ppm; in questo frangente anche la temperatura subisce una diminuzione di un paio di gradi fino a 22°C.

Attorno alle ore 16:00 la giornata scolastica è conclusa e, l'andamento esponenziale decrescente della CO₂ ne dà la conferma.

- L'andamento dell'**umidità** non presenta notevoli variazioni durante la giornata; con un'analisi accurata, però, si notano alcuni gradini discendenti di umidità in corrispondenza dei picchi di CO₂, segno dell'effettiva apertura delle finestre. Questi gradini si notano bene negli ultimi picchi della giornata (picco delle ore 14:00 e picco delle ore 15:30) in cui l'umidità scende dal 60% al 50% circa.

Il test è stato effettuato anche in questo caso per 24h, la figura 5.4 illustra l'andamento delle grandezze del grafico precedente esteso a tutto il periodo di monitoraggio.

Si notano subito i valori esagerati di temperatura che raggiungono addirittura i

28°C alle 22:30, segno di un malfunzionamento o di una non corretta configurazione dei parametri del sistema di riscaldamento. Mentre i valori di CO₂ restano costanti, durante la notte temperatura e umidità variano in maniera ‘speculare’ (lo stesso profilo di ascesa della temperatura si nota come profilo di discesa dell’umidità e viceversa); tale particolarità è dovuta al sistema di riscaldamento ad aria (ventilconvettore) che emana aria calda abbassando l’umidità relativa della stanza.

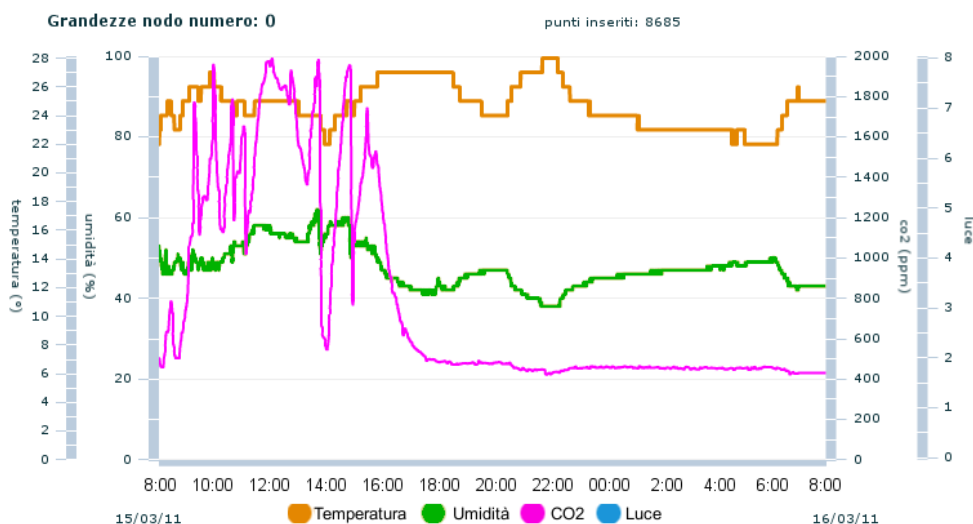


Figura 5.4: Grafico del monitoraggio nell’aula B durante 24h di campionamento)

Differenze tra aula A e aula B

Nel complesso, pur avendo un numero di alunni occupanti molto superiore, nell’aula B si riscontrano livelli di CO₂ piuttosto accettabili, in quanto si può notare un valore medio della giornata di circa 1500 ppm; questo solo grazie al fatto che la stanza è stata spesso arieggiata, anche grazie alle ottime condizioni ambientali esterne.

Tale conclusione non si può di certo affermare per i test svolti nell’aula A in cui i livelli di anidride carbonica sono risultati troppo elevati (un’apertura delle finestre tra l’inizio delle lezioni e l’intervallo sarebbe bastata per migliorare notevolmente il comfort dell’aria).

L’apertura delle finestre ed il range di misurazione del sensore troppo limitato, non ha permesso nei due casi di misurare un valore costante di CO₂; la differenza notevole nel numero di alunni occupanti l’aula è comunque riscontrabile dalle pendenze degli andamenti di anidride carbonica rilevati; si è dunque preso in esame l’andamento della CO₂ nell’aula A dalle ore 8:30 alle 9:30 (fig.5.5) e nell’aula B dalle 14:30 alle 16:30 (fig. 5.6). Linearizzando l’andamento, dalle figure si

può notare nel primo caso una pendenza di circa 30 ppm/minuto (2000 ppm/h) mentre nel secondo, una pendenza di circa 40 ppm/minuto (2400 ppm/h)⁴.

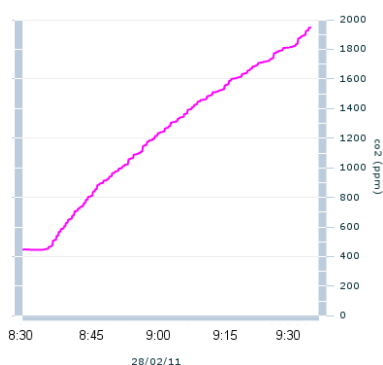


Figura 5.5: Pendenza della CO2 nell'aula A

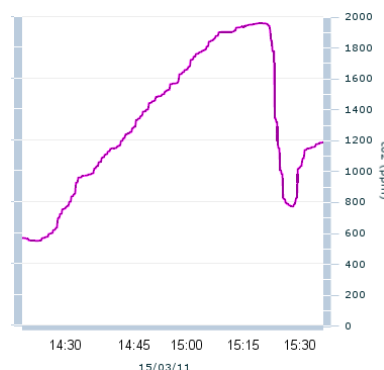


Figura 5.6: Pendenza della CO2 nell'aula B

5.2 Consumi misurati

Per ottenere delle informazioni più dettagliate per quel che riguarda i consumi misurati dai vari nodi, si è deciso di implementare un sistema di misurazione della corrente. Tutto ciò per ottenere un confronto tra i consumi effettivi e quelli dichiarati nel datasheet dalla casa costruttrice.

I valori ottenuti diventano di notevole importanza nel caso in cui si decida di alimentare i nodi con una batteria alimentata da una fonte di energia rinnovabile. In tale paragrafo sarà prima illustrato brevemente il sistema di misura utilizzato e successivamente saranno analizzati i risultati ottenuti.

5.2.1 Sistema utilizzato

La corrente è stata misurata grazie al multimetro *34970a* della casa *Agilent* [27]. Tale strumento permette di essere interfacciato con un PC che effettua periodicamente delle letture ottenendo una serie di valori misurati; il dispositivo è in grado di misurare tensioni, resistenze e correnti. Per rendere il sistema di misura più leggero e veloce, è stato scelto di aggiungere in serie al circuito una resistenza di valore molto basso e di misurarne la caduta di tensione ai propri capi. Il programma per la lettura dei dati è stato scritto in linguaggio *Vee* [28] che periodicamente effettua una richiesta al multimetro che risponde con un valore di tensione misurata. Il sistema è mostrato in figura 5.7.

Nei risultati seguenti la corrente è stata misurata all'ingresso della scheda del nodo completa; ciò vuol dire che essa non è relativa solamente a quella consumata

⁴Nel caso B sono presenti, nella prima parte della mattinata dei picchi di CO2 con pendenza nettamente superiore, ma il loro andamento poco prolungato nel tempo non ha permesso di effettuare un'analisi sufficientemente accurata

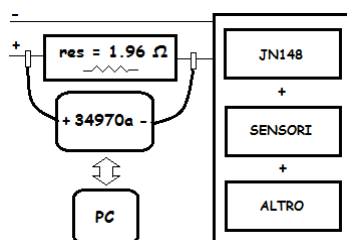


Figura 5.7: Schema per la misurazione dei consumi di corrente della *sensor board*

dal modulo wireless (JN148) ma comprende anche il consumo dei vari sensori montati sulla scheda e di un pò di altra elettronica di contorno (DC/DC, capacità, LED, ecc. . .).

Con il sistema risultante, si riesce a campionare con una frequenza di circa 10 campioni al secondo. Tale valore è sufficiente per ottenere una stima della corrente consumata anche se alcuni picchi abbastanza rapidi saranno definiti solamente da pochi campioni.

5.2.2 Risultati ottenuti

I valori ottenuti sono quindi misurazioni in tensione. Il valore di corrente consumata sarà quindi, il valore di tensione diviso per la resistenza in serie.

Per tale esperimento è stato utilizzato una resistenza con valore misurato pari a $R \simeq 1.96\Omega$.

Le misurazioni sono state effettuate su 3 tipi di nodi:

1. Nodo router privo di sensore di CO2
2. Nodo router con sensore di CO2
3. Nodo EndDevice

Router senza CO2

La figura 5.8 mostra l'andamento della tensione ai capi della resistenza posta in serie ad una delle schedine utilizzate per il progetto con montato un modulo JN148 e i vari sensori (temp.,umidità e luce) escluso quello di CO2. Dal grafico si può notare un andamento della tensione a picchi con periodo di 10 secondi; la caduta di tensione media ai capi della resistenza a riposo è di circa 35.5 mV, corrispondenti a circa 18 mA. In fase di trasmissione dei dati, si nota la presenza di picchi di tensione pari a 42mV cioè 21.4 mA con durata di circa 300 ms.

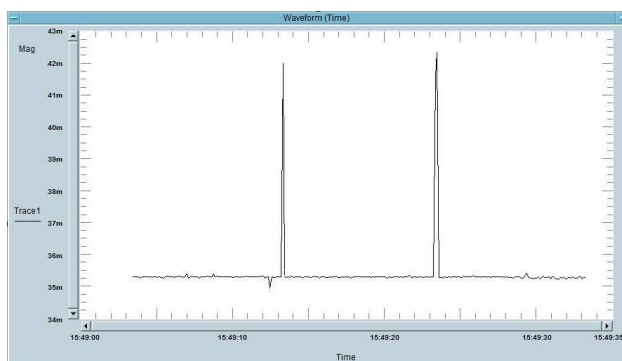


Figura 5.8: Grafico del valore in tensione misurato ai capi della resistenza con collegata una scheda che monta un nodo router priva di sensore di CO2

Router con CO2

A differenza del caso precedente, qui è stato aggiunto il contributo del sensore di CO2 che, come si è visto, presenta dei consumi decisamente importanti e confrontabili con quelli della scheda stessa. Dal grafico di figura 5.9 si possono notare due fasi distinte durante il periodo di 10 secondi:

- **A.** In tale fase si può notare un picco di corrente come nel caso precedente corrispondente all'istante di comunicazione seguito da una 'coda' corrispondente al tempo impiegato per la lettura del sensore di CO2. La scheda consuma circa 21 mA con un picco a 25 mA.
- **B.** Il consumo di corrente che si può notare in questa fase corrisponde solamente a delle operazioni interne del sensore di CO2. Se, infatti, il sensore viene scollegato dal circuito ma con l'alimentazione ancora collegata, tale consumo è ancora ben visibile. La scheda consuma circa 20 mA medi con picchi di qualche mA in più.

EndDevice

A differenza del primo caso in cui si analizzava il consumo di un router, qui i livelli di corrente consumata sono decisamente minori in quanto il dispositivo è normalmente spento e si riaccende solamente per comunicare (figura 5.10).

In questo caso, la corrente consumata dalla scheda quando il dispositivo è in fase di *sleep* è ora di circa 2 mA con picchi in trasmissione di circa 12 mA.

5.3 Possibili Ottimizzazioni

Sin dalle prime fasi di progetto, il sistema è stato pensato e sviluppato in modo da garantire un suo facile impiego come prodotto commerciale grazie alla sua versatilità e al suo semplice utilizzo; l'interfaccia web, infatti, è stata predisposta per assicurare un facile accesso del sistema anche al fruitore più esigente.

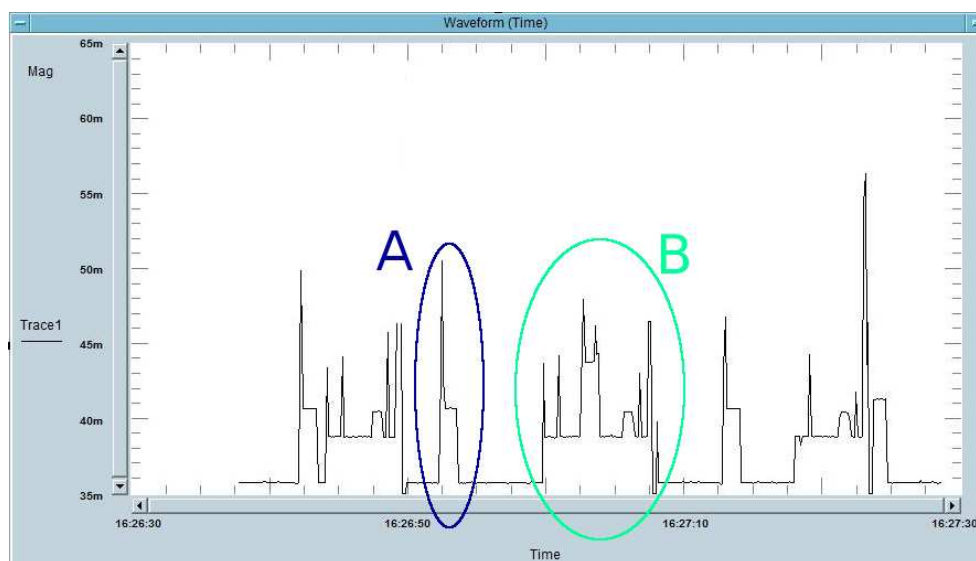


Figura 5.9: Grafico del valore in tensione misurato ai capi della resistenza con collegata una scheda che monta un nodo router ed il sensore di CO2

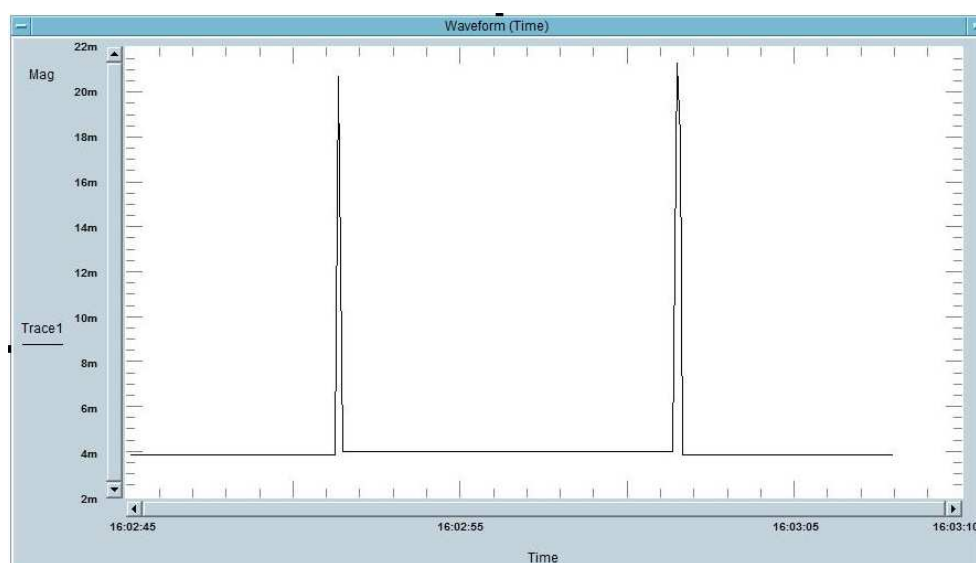


Figura 5.10: Grafico del valore in tensione misurato ai capi della resistenza con collegata una scheda che monta un nodo EndDevice (priva di sensore di CO2)

Il sistema presentato risulta essere ben funzionante, ma come la maggior parte dei progetti commerciali, esso non può essere ritenuto perfettamente concluso; esiste infatti un certo numero di aspetti e funzionalità che possono essere migliorati e ottimizzati.

In particolare, solamente a progetto concluso, si è notato un utilizzo del database non del tutto ottimizzato; tale aspetto si nota sia nelle fasi di lettura che di scrittura.

- **Problemi in scrittura.** Nel momento in cui più dati giungono al miniPC, molte volte può avvenire un conflitto tra di essi in fase di salvataggio nel database; in questi casi, il programma in C si trova a memorizzare il secondo dato finché il database è ancora occupato dalla scrittura del primo. Una soluzione potrebbe essere quella di gestire la scrittura mediante l'utilizzo di *thread*; essi sono essenzialmente dei mezzi usati dal C per gestire più processi che utilizzano risorse comuni (per esempio database). In questo caso potrebbe essere lanciato un *thread* ogniqualvolta un dato debba essere memorizzato; in questo modo, il problema della concorrenza viene eliminato in quanto i *thread* successivi vengono lanciati solamente se quello corrente è concluso.
- **Problemi in lettura.** Qualora si vogliano caricare sulla pagina web dei grafici che richiedono una mole di dati piuttosto elevata (con un numero di punti superiore a qualche migliaio), il sistema presenta alcune difficoltà nell'effettuare il caricamento con una certa rapidità. Per esempio, nel caricare un grafico contenente i valori di una mezza giornata, 3000 punti circa, il sistema ci impiega non meno di una decina di secondi. Questo ritardo temporale è dovuto alle prestazioni del processore del miniPC e del DBMS; è proprio la query del SELECT che richiede un tempo così elevato⁵. Se, in campo applicativo, questo risultasse un problema, esso potrebbe essere ovviato grazie ad una gestione diversa delle tabelle, per esempio creando una tabella diversa per ogni nodo della rete.

Per ottimizzare al meglio il datalogging dei dati, si potrebbe gestire il campionamento in modo personalizzato da sensore a sensore e da nodo a nodo; la grande mole di dati potrebbe, infatti, diventare un problema per la gestione del database. A seconda dei casi, potrebbe accadere che molti campioni non siano necessari e risultino ridondanti o che in alcune aree monitorate sia richiesto un campionamento più o meno frequente.

Per riferirsi ad un esempio concreto, nel caso di Inthebra, potrebbe essere che il nodo sensore posto in bagno, non debba per forza campionare con lo stesso periodo degli altri nodi posti in locazioni più utilizzate; in aggiunta, il nodo potrebbe essere programmato per campionare la luce con un certo periodo mentre i valori di temperatura, umidità e anidride carbonica con un periodo molto inferiore al fine di ottenere una mole di dati di dimensioni decisamente inferiori.

⁵Come già accennato le richieste di SELECT per la visualizzazione del grafico, sono di tipo composto con codizioni sul numero di nodo e sull'orario, tra le limitazioni di SQLite (4.3.2), infatti, era già stata discussa la bassa velocità nell'eseguire query piuttosto complicate.

5.4 Conclusioni sugli obiettivi raggiunti

Dopo un'approfondita analisi dei vari dati raccolti nei vari test effettuati, si vuol ora capire se il sistema implementato sia effettivamente in grado oppure no di stabilire se vi è la presenza di persone all'interno di un locale. I test sono stati effettuati in ambienti e condizioni sempre diverse tra loro ma è stato quasi sempre possibile notare una variazione di almeno una delle grandezze rilevate qualora vi fosse almeno una persona all'interno della stanza.

La maggior parte delle volte, però, le variazioni sono risultate piuttosto lievi e paragonabili ad altre dovute a situazioni indipendenti dal grado di occupazione come per esempio l'apertura di porte, apertura di finestre ecc... Nei casi in cui l'ambiente, magari di una certa dimensione volumetrica, era occupato da un individuo, le variazioni erano sì presenti ma talmente lievi da non garantire la presenza della persona nella stanza; molte volte era sufficiente un'apertura di qualche secondo della porta per permettere ai sensori di rilevare le variazioni dovute al ricircolo d'aria avvenuto.

Sarà presentata ora una breve analisi per capire quali delle 4 grandezze fisiche rilevate siano quelle che meglio si adeguano per stabilire il grado di *occupancy*:

1. **Temperatura:** in tutti i test effettuati non ha mai portato una grossa informazione relativa l'occupazione dell'area di test; in effetti, essa è molto dipendente da fattori non collegati per forza all'*occupancy*, quali per esempio:
 - sistemi di riscaldamento/raffreddamento (ventilconvettori),
 - livello di luce del sole (in un ambiente con ampie vetrate è sufficiente un tiepido sole invernale per riscaldare l'ambiente di qualche grado)
 - altro...
2. **Luce:** dall'analisi dell'andamento di tale grandezza si è visto che il suo monitoraggio riesce a stabilire con precisione la presenza di persone solamente nei casi in cui l'occupazione dell'ambiente comporta una variazione brusca della grandezza stessa (in cui per variazione brusca si intende l'accensione delle luci oppure l'apertura delle tapparelle). Si consideri nuovamente la figura 2.9; in quel caso specifico, grazie al fatto che il locale in questione era un bagno cieco, la luce rappresentava la grandezza che meglio si adeguava a stabilire la presenza di persone o meno. Nella maggior parte degli altri casi, però, era sufficiente che la stanza fosse occupata solamente durante le ore di sole perchè i valori di luce non portassero informazione relativa al grado di *occupancy*.
3. **Umidità:** dai grafici raccolti, rappresenta assieme alla CO₂ una valida grandezza fisica per determinare la presenza di persone; in quasi tutti i casi la presenza di individui comporta un, seppur lieve, aumento dell'umidità.

Ogniqualevolta si sia verificata un'apertura delle finestre durante la mattinata lavorativa (o scolastica), si è notato un ripido crollo di qualche punto percentuale di umidità seguito da una lenta risalita ai valori precedenti.

L'utilizzo dei valori di tale grandezza fisica diventa inefficace nel momento in cui i livelli di umidità dell'aria esterna siano superiori, o peggio uguali, con quelli dell'ambiente di test. In questo modo non sarebbe più possibile interpretare correttamente le variazioni di umidità rilevate.

4. **Anidride Carbonica:** Indubbiamente, tale grandezza riesce meglio di tutte a stabilire la presenza di persone visto che i livelli di CO₂ presenti in un determinato ambiente sono dovuti quasi essenzialmente alla respirazione delle persone occupanti. In tutti i test effettuati, la concentrazione di anidride carbonica parte da livelli di qualche centinaio di ppm per aumentare progressivamente con l'occupazione della stanza fino a raggiungere livelli più o meno elevati a seconda del numero di occupanti e del volume dell'ambiente (dai 1000 ppm in su).

Molte volte, il range di variazione è molto ampio, anche più di un migliaio di ppm (figura 5.2). Una variazione del genere però, non è immediata; essa può durare diversi minuti sino a raggiungere un valore stabile dipendente da quante persone occupano il locale (se la variazione è positiva) o dal tasso di ventilazione dell'aria (se la variazione è negativa). In aggiunta al tempo di assestamento dei valori di CO₂, vi è un ritardo introdotto dal sensore utilizzato; mentre le variazioni di umidità sono rilevate immediatamente quelle di anidride carbonica sono rilevate dopo qualche minuto circa (tale fenomeno è ben riscontrabile nella figura 2.9).

Nel caso in cui si hanno continui circoli d'aria, apertura di porte o finestre, anche questa grandezza fisica subisce delle variazioni notevoli indipendentemente dalla quantità di persone occupanti la stanza.

Si può quindi affermare che **tale sistema è valido per la rilevazione dell'“occupancy” solo in alcuni casi**. I casi a cui ci si riferisce sono quelli in cui l'ambiente sia ben confinato e il rapporto numero di persone su volume dell'ambiente sia sufficientemente alto; ogni cambio d'aria, infatti, è responsabile di variazioni delle grandezze rilevate che non sono bene interpretabili, specie da un algoritmo di calcolo.

Per concludere, si può affermare come tale apparecchiatura non sia di certo adatta al fine di determinare la presenza di individui in un sistema d'allarme perchè presenterebbe dei ritardi e delle incertezze inadeguate al contesto; diventa piuttosto un valido mezzo per la gestione “intelligente” dei sistemi di ricircolo d'aria al fine di limitare i consumi energetici ottimizzando allo stesso tempo la qualità dell'aria ed il benessere delle persone.

Appendice A

Planimetria completa di Inthebra s.r.l.

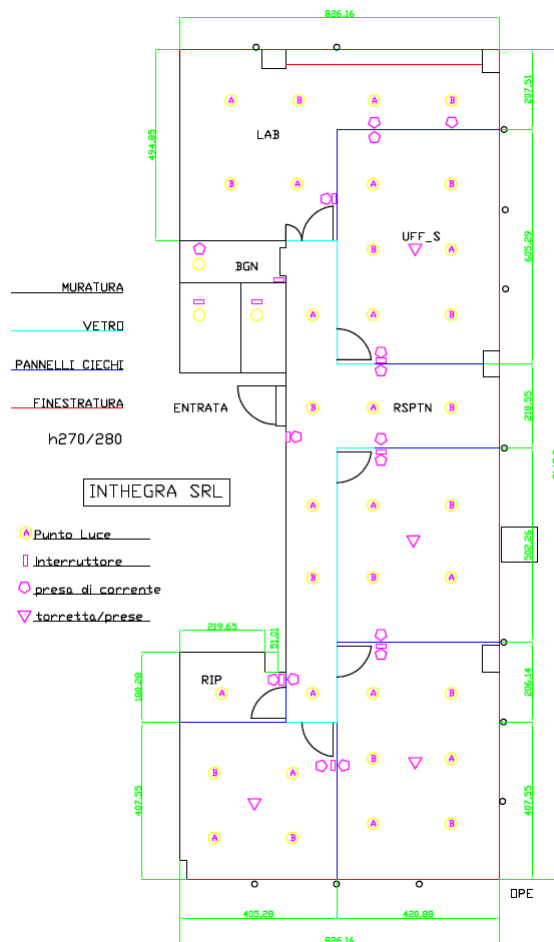


Figura A.1: Planimetria della ditta Inthebra s.r.l. in cui sono stati effettuati i vari test

Appendice B

Tabella specifiche sensore di CO2

La tabella B.1 mostra le principali caratteristiche di misura del sensore di anidride carbonica utilizzato.

1. Range di misurazione	0 – 2000 ppm
2. Errore Assoluto	$\pm 50\text{ppm} + 2\%$ valore misurato
3. Dipendenza dalla temperatura	2ppm CO2/°C(0... 50°C)
4. Intervallo di misura	da 15 sec fino ad 1 ora
5. Condizioni operative:	-40... 60°C 5... 95% RH

Tabella B.1: Caratteristiche di misura del sensore EE892

La tabella B.2 mostra, invece le caratteristiche elettriche del sensore.

1. Tensione di alimentazione:	4.75-7.5 V DC
2. Consumo medio:	3.75 mA (con 15 sec di periodo) 58 μ A (con 1 ora di periodo)
3. Corrente di picco:	max 500mA per 50 ms

Tabella B.2: Caratteristiche elettriche del sensore EE892

Ringraziamenti

Sono veramente molte le persone che in qualche maniera hanno contribuito per la realizzazione di questa tesi.

- *Innanzitutto desidero ringraziare il professore **Luca Schenato** che mi ha appoggiato accademicamente per questa tesi.*
- *Un sincero grazie a tutti i colleghi ed amici di Inthebra che mi hanno supportato (o sopportato) per i mesi di lavoro. In particolare grazie al mio tutor aziendale **Riccardo** che ha sempre risposto alle mie esigenze e dubbi con ottimi consigli; ringrazio anche l'amico **Simone** per tutte le volte che mi ha aiutato quando avevo dei dubbi per la progettazione in Flex.*
- *Grazie a **mamma** e **papà** ai quali ho dedicato questo lavoro perchè se sono arrivato a questo punto è solamente per merito loro. Assieme a **Ari** e **Martina** mi hanno sempre incoraggiato ad andare avanti motivandomi a fare sempre del mio meglio.*
- *Grazie alla direzione didattica di Asiago per avermi permesso di effettuare le prove per il monitoraggio nelle scuole di Rotzo e grazie a **Serena Cunico** per il monitoraggio nelle scuole di Padova.*
- *Un grazie a quella cosa che si chiama **MUSICA** che mi ha accompagnato per i cinque anni di università dal primo esame fino all'ultimo giorno di tesi. Non v'è stato un esame che non è sia stato preparato assieme ad una qualche sinfonia di Beethoven o un qualche concerto di Tchaikovsky (tanto per citarne due di numero)*

Bibliografia

- [1] Kazem Sohraby, Daniel Minoli, Taieb Znati, 'Wireless Sensor Networks: Technology, Protocols, and Applications', WileyInterscience, 2007. ISBN 0471743003, 9780471743002.
- [2] Filippo de Stefani 'Sistema di monitoraggio ambientale tramite WSN', 2008;
- [3] Stephen Brown, National University of Ireland Maynooth Cormac J. Sreenan, University College Cork 'A New Model for Updating Software in Wireless Sensor Networks', 2006;
- [4] B. Peters, 'Sensing Without wires: Wireless Sensing Solves Many Problems, But Introduces a Few of Its Own,' Machine Design, Penton Media, Cleveland, OH,
- [5] 'Spec chip', 2007, http://www.jhlhlab.com/jhill_cs/spec/
- [6] Melikov A.K. (2004). Personalized ventilation. Indoor Air 14, supp.7: pp. 157-167.
- [7] Comfort ambientale, 2011, http://it.wikipedia.org/wiki/Comfort_ambientale
- [8] Elettrosmog Control, 2011, www.elettrosmogcontrol.com/microclima.htm
- [9] Lighting control system with infrared occupancy detector, William Kahl, 1985
- [10] http://www.iklimnet.com/expert_hvac/occupancy-sensors.html, 2011
- [11] Apr 1, 2007 12:00 PM, By Craig DiLouie, Lighting Controls Association
- [12] Reducing barriers to use of high efficiency lighting systems, Marzo 2003 - Gennaio 2004, Lighting Research Center
- [13] H.-E. Endres, R. Hartinger, M. Schwaiger, G. Gmelch, M. Roth, 'A capacitive CO2 sensor system with suppression of the humidity interference.', Sensors and Actuators B 57 (1999) 83-87.

-
- [14] P. Keller, H. Ferkel, K. Zweiacker, J. Naser, J.-U. Meyer, W. Riehemann, The application of nanocrystalline BaTiO₃-composite films as CO₂-sensing layerSensors and Actuators B 57 (1999) 39-46.
- [15] ‘Ventilation Control Options’, the Comfort Zone, April 2003, Maury Tiernan
- [16] ‘Carbon Dioxide (CO₂) Sensors’,
www.hydro.mb.ca/your_business/hvac/ventilation_co2_sensor.shtml
- [17] State-of-the-Art Review of CO₂ Demand Controlled Ventilation Technology and Application, Steven J. Emmerich, Andrew K. Persily, 2001
- [18] EPA Exposure Factors Handbook (EPA 1999).
- [19] Persily Dols 1990; Persily 1997.
- [20] De Nardis, L. Di Benedetto., ‘Overview of the IEEE 802.15.4/4a standards for low data rate Wireless Personal Data Networks’. In Positioning, Navigation and Communication, 2007. WPNC '07. 4th Workshop. Publication Date: 2222 March 2007. On page(s): 285-289.
- [21] ‘Studio del protocollo ZigBee e primo dimostratore’, Technical Report TR-4.4.2.
- [22] ‘ZigBee-PRO Stack Profile: Platform restrictions for compliant platform testing and interoperability’, Revision 05
- [23] 2011,http://www.jennic.com/products/modules/jn5148_modules
- [24] ‘Data Sheet: JN5148-001’,
http://www.jennic.com/files/product_briefs/JN-DS-JN5148-1v6.pdf
- [25] [http://it.wikipedia.org/wiki/Eclipse_\(informatica\)](http://it.wikipedia.org/wiki/Eclipse_(informatica))
- [26] Section 1.3, JN-UG-3065, Revision 1.1, 2-Dec-2009
- [27] datasheet: ‘Agilent 34970A Data Acquisition’ Switch Unit Family
- [28] ‘VEE Pro User’s Guide’, Agilent Technologies,
- [29] School air quality related to dry cough, rhinitis and nasal patency in children, M. Simoni, 2010
- [30] ‘<http://www.ecoage.it/pm10.htm>’
- [31] SQLite, 2011, <http://www.sqlite.org/>
- [32] <http://httpd.apache.org/docs/>
- [33] PHP, 2011, <http://www.php.net>

-
- [34] http://en.wikipedia.org/wiki/Universal_asynchronous_receiver/transmitter
- [35] 2011, <http://www.adobe.com/it/products/flex/>
- [36] 2011, <http://php.html.it/guide/lezione/2663/le-variabili-get-e-post/>
- [37] JN-RM-2003, Revision 2.0, 22 Novembre 2010, LPRF Board API Reference Manual
- [38] Ambient Light Sensor (ALS) : TSL2550D,
www.taosinc.com/ProductDetails.aspx?id=133
- [39] 2011, www.embeddedarm.com
- [40] http://en.wikipedia.org/wiki/Unix_time