



Sii-Mobility

Supporto di Interoperabilità Integrato per i Servizi al Cittadino e alla Pubblica Amministrazione

Trasporti e Mobilità Terrestre, SCN_00112

Deliverable ID: DE3.11a

**Titolo: Attuatori integrati - specifica di dettaglio dello
sviluppo**

Data corrente	20/02/2017
Versione (solo il responsabile può cambiare versione)	V 0-11a
Stato (draft, final)	final
Livello di accesso (solo consorzio, pubblico)	Pubblico quando completo
WP	OR3
Natura (report, report e software, report e HW..)	report
Data di consegna attesa	M7 – Luglio 2016
Data di consegna effettiva	M14 – Febbraio 2017
Referente primario, coordinatore del documento	Alessandro Paganone, ECM, alessandro.paganone@ecmre.com
Contributor	Andrea Rindi, UNIFI-DIEF/Mdmlab, andrea.rindi@unifi.it Francesco Chiti, UNIFI-LART, francesco.chiti@unifi.it Argeo Bartolomei, ARGOS, abartolomei@argosengineering.it

	Leonardo Fabbri, ELFI, leonardo.fabbri@elfisrl.net
Coordinatore responsabile del progetto	Paolo Nesi, UNIFI, paolo.nesi@unifi.it

Sommario

1	Executive Summary	6
2	Introduzione ed obiettivi	6
2.1	Acronimi, sigle, terminologia.....	6
2.2	Documenti di riferimento	7
2.2.1	Standards e normative applicabili	7
2.2.2	Documenti Sii-Mobility	7
2.2.3	Altri riferimenti	7
2.3	Contesto.....	8
2.4	Notazioni	8
2.4.1	Schemi di architettura	8
2.4.2	Schemi di dettaglio.....	9
2.5	Obiettivi.....	10
3	Struttura comune a tutti gli Attuatori Integrati.....	11
3.1	Specifica di dettaglio della struttura comune	12
3.1.1	Piattaforma	13
3.1.2	Tablet di servizio.....	17
3.1.3	Sensori e attuatori.....	18
3.1.4	Protocollo base Bluetooth	19
4	Segnale Dinamico	20
4.1	Architettura generale	20
4.2	Piattaforma	21
4.2.1	Meccanica, contenitori, carpenterie, installazione	21
4.2.2	Hardware	21
4.2.3	Software	23
4.3	APP con Procedura di Segnaletica, APP04.....	23
4.4	Sensori / attuatori.....	23
5	Direzionatore.....	25
5.1	Architettura generale	25
5.2	Piattaforma	26
5.2.1	Meccanica, contenitori, carpenterie, installazione	26
5.2.2	Hardware	26
5.2.3	Software	26
5.3	APP con Procedura di Direzioneamento, APP05	26
5.4	Sensori / attuatori.....	26
6	Rastrelliera Intelligente	27
6.1	Architettura generale	27

6.2	Piattaforma	28
6.2.1	Meccanica, contenitori, carpenterie, installazione	28
6.2.2	Hardware	28
6.2.3	Software	30
6.3	APP per gestione rastrelliera, APP06.....	30
6.4	Sensori / attuatori.....	30
6.4.1	Rilevatore RFID	30
6.4.2	Vincolo Meccanico	35
Figura 1	: Attuatori Integrati nel contesto di Sii-Mobility	8
Figura 2	: notazione grafica generale	9
Figura 3	: notazione grafica di dettaglio	9
Figura 4	: Struttura comune degli Attuatori Integrati	11
Figura 5	: Struttura comune - diagramma di contesto.....	12
Figura 6	: Piattaforma – diagramma di contesto	13
Figura 7	: Architettura di base del software	16
Figura 8	: Tablet – diagramma di contesto	18
Figura 9	: Architettura generale del Segnale Dinamico	20
Figura 10	: Schema generale del Segnale Dinamico.....	22
Figura 11	: Architettura del Visualizzatore LED	24
Figura 12	: Architettura generale del Direzioneatore	25
Figura 13	: Architettura generale della Rastrelliera Intelligente.....	27
Figura 14	: Schema generale della Rastrelliera Intelligente-definizioni delle connessioni per la parte di raccolta dati proveniente dai sistemi RFID di ogni stallo(a) e comando degli attuatori a solenoide per ogni stallo (b).....	29
Figura 15	: Sistema locale di identificazione RFID installato sullo stallo della rastrelliera.....	31
Figura 16	: Sistema locale di identificazione RFID - PN 532 RFID/ NFC (Arduino like)	31
Figura 17	: definizione della fase di START/STOP, ACKNOWLEDGE sul bus I2C	32
Figura 18	: comunicazione sul bus I2C (indirizzo slave + data).....	33
Figura 19	: Architettura generale di principio del sistema elettronico, caso operativo di rastrelliera con singolo stallo. La piattaforma TOTEM comunica con i lettori RFID tramite porta seriale I2C	33
Figura 20	: Comunicazione tra RFID Reader e motherboard (da sviluppare per ogni stallo)	34
Figura 21	: CHIP P82B715 per estendere il bus I2C	34

Figura 22 : perno azionamento con intervento esterno (Sx) e elettromagnete dotato di bobina che permette lo sblocco mediante applicazione di una tensione di 12 V (Dx).....	35
Figura 23 : cursore in posizione di blocco mediante compressione a molla (Sx) e cursore in posizione di sblocco mediante applicazione di eccitazione della bobina tramite una tensione di 12 V	35
Figura 24 : schema di principio di controllo tramite pin digitale di Arduino del singolo solenoide responsabile del gancio/sgancio della bicicletta allo stallo.....	36
Figura 25 : relè dedito all'azionamento del perno usato per l'aggancio/sgancio della bicicletta alla rastrelliera.....	36
Figura 26 : PCB ospitante il relè dedito all'azionamento del perno usato per l'aggancio/sgancio della bicicletta alla rastrelliera	37
Figura 27 : Solenoide a D in c.c. A spinta BLP 42-120-621-620, corsa massima 27mm, tensione 12 V c.c., 10W	37

1 Executive Summary

Il documento contiene il progetto dettagliato degli attuatori integrati previsti nel contesto del progetto:

- Segnale Dinamico - per le porte di accesso ai parcheggi e alle ZTL;
- Direzionatore - per la modifica dinamica del senso di percorrenza di una strada o corsia;
- Rastrelliera Intelligente - per realizzare la funzione **AATF** (autonomous anti-theft feature).

Tutti gli attuatori integrati sono realizzati interamente nel contesto del progetto. Per ciascuno di essi sono dettagliati:

1. l'architettura generale, contestualizzata in Sii-Mobility
2. l'architettura di dettaglio, con individuazione delle principali componenti, fino al livello di singola scheda elettronica e/o componente software principale
3. la specifica di dettaglio dell'hardware;
4. la specifica di dettaglio del software;
5. la specifica di dettaglio di contenitori e parti meccaniche
6. la sensoristica utilizzata. Per ciascun sensore sono dettagliate le interfacce fisiche e logiche e l'architettura interna. Per alcuni sensori sviluppati nel contesto della Attività 3.2 si rimanda alla relativa documentazione ([SiiM 4]).

Questo documento (DE3.11a) ha lo scopo di descrivere quanto si intende sviluppare. È pianificata una successiva revisione (DE3.11b) che descriverà quanto effettivamente prodotto, tenendo conto di inevitabili modifiche e variazioni in corso d'opera. Servirà anche a risolvere alcuni "TBD" che a questo stadio del progetto sono ancora presenti.

2 Introduzione ed obiettivi

2.1 Acronimi, sigle, terminologia

API	Application Program Interface
AVL - AVM	Automatic Vehicle Location - Automatic Vehicle Monitoring
CEN	European Committee for Standardization
GPRS	General packet radio service
GPS	Global positioning System
GSM	Global System for Mobile
HDMI	High Definition Multimedia Interface
HTML	Hyper Text Markup Language
ICT	Information and Communication Technologies
ITS	Intelligent Transport Systems
LCD	liquid-crystal display
OD	Open Data
PMS	Private Mobile Systems
RFID	Radio Frequency IDentification o Identificazione a radio frequenza

SII	sistema di interoperabilità integrato
SMS	Short Message Service
SN	social networking, oppure sensor network
SSAMM	Agenzia per la Mobilità Metropolitana strumenti di supporto, TOSCANA
TPL	gestore trasporto pubblico locale
UML	Unified Modeling Language
UMTS	Universal Mobile Telecommunications System
UTC	Urban Traffic Control
V2I	Vehicle-to-Infrastructure
V2V	vehicle-to-vehicle
VWSN	Vehicular Wireless Sensor Networks
W3C	World Wide Web Consortium
WSDL	Web Services Description Language
WSN	Wireless Sensor Networks
XMI	XML Metadata Interchange standard di OMG
XML	Extensible Markup Language
ZTL	Zona a Traffico Limitato

2.2 Documenti di riferimento

2.2.1 Standards e normative applicabili

[Std 1]	TBD

2.2.2 Documenti Sii-Mobility

[SiiM 1]	DE1.1a v3-0 - Analisi dei requisiti e casi d'uso
[SiiM 2]	DE1.2a - Specifica di Integrazione e Casi di Test
[SiiM 3]	DE8.5 - Manuale di qualità del progetto
[SiiM 4]	DE3.6a – Sensori : specifica di dettaglio dello sviluppo
[SiiM 5]	DE3.1a – Kit veicolari : specifica di dettaglio dello sviluppo
[SiiM 6]	Sii-Mobility-meeting-2016-11-28-v7-0 (PPT), verbale della riunione
[SiiM 7]	2016.09.16 Verbale riunione skype OR3

2.2.3 Altri riferimenti

[Ref 1]	Arduino Board Open Source (http://www.arduino.org/)
[Ref 2]	Arduino 101 (https://www.arduino.cc/en/Main/ArduinoBoard101)
[Ref 3]	ARGOS SII-MOBILITY-RAST-BICI-001-R01-15-02-2017 (allegato a questo documento)

2.3 Contesto

Questo documento è prodotto nel contesto dell'Obiettivo Realizzativo *OR3 - Sviluppo di prototipi applicativi verticali, sensori e attuatori* e in particolare della *Attività 3.3 - Studio, definizione e sviluppo di Attuatori integrati per controllo accessi, direzione e velocità*.

Il documento è conforme alle prescrizioni del manuale della qualità del progetto ([SiiM 3]).

La *Figura 1* mostra il ruolo degli Attuatori Integrati nell'architettura generale di Sii-Mobility. Complessivamente, gli attuatori integrati hanno lo scopo di:

- interagire con *Sensori e Attuatori* per acquisirne i dati e comandarli / monitorarli,
- ospitare Applicazioni SW (*Processi*) fornendo loro i servizi per svolgere le loro funzioni, includendo anche le azioni descritte al punto precedente.

I Processi interagiscono con i sistemi informativi dei Gestori di parcheggi e/o ZTL, oppure direttamente con la piattaforma Sii-Mobility.

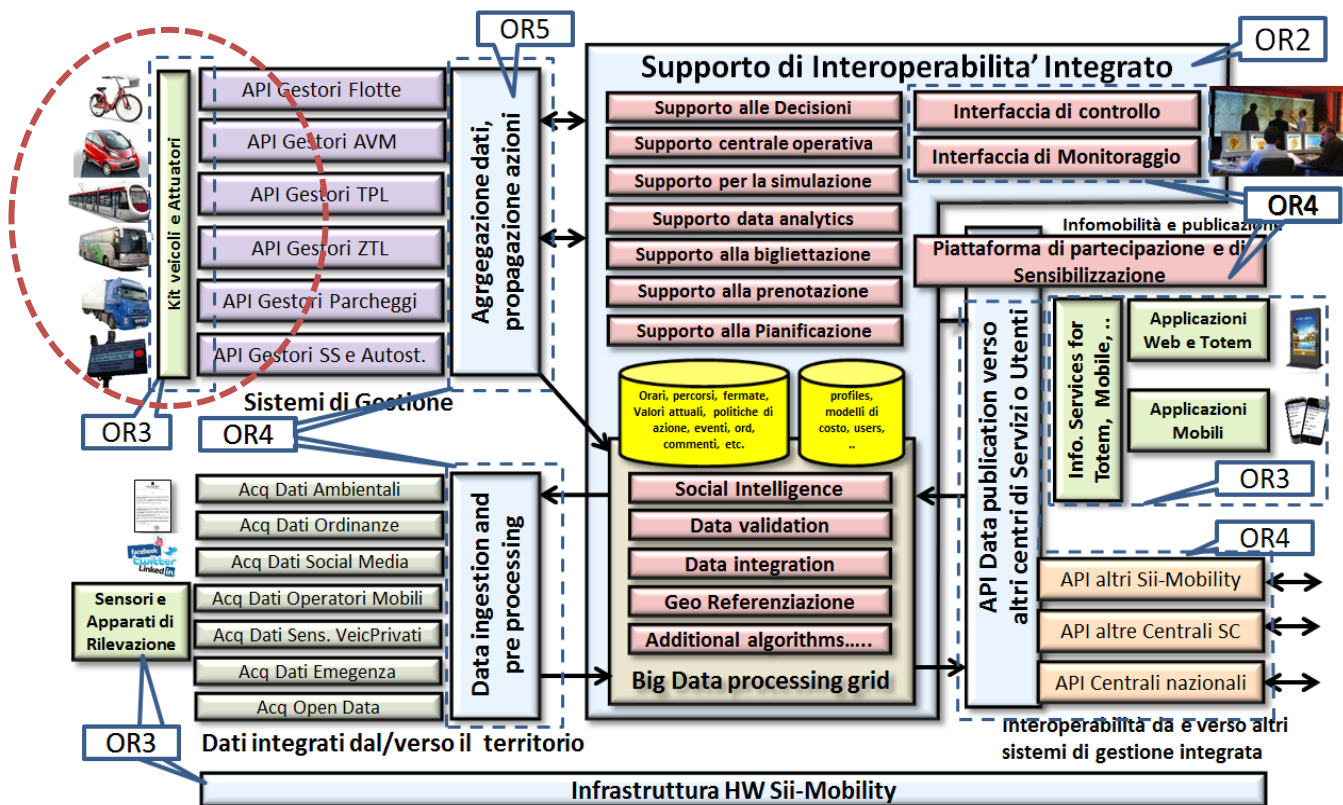


Figura 1 : Attuatori Integrati nel contesto di Sii-Mobility

2.4 Notazioni

2.4.1 Schemi di architettura

Negli schemi di architettura generale è utilizzata la notazione in *Figura 2* per rappresentare i flussi di dati e controlli.

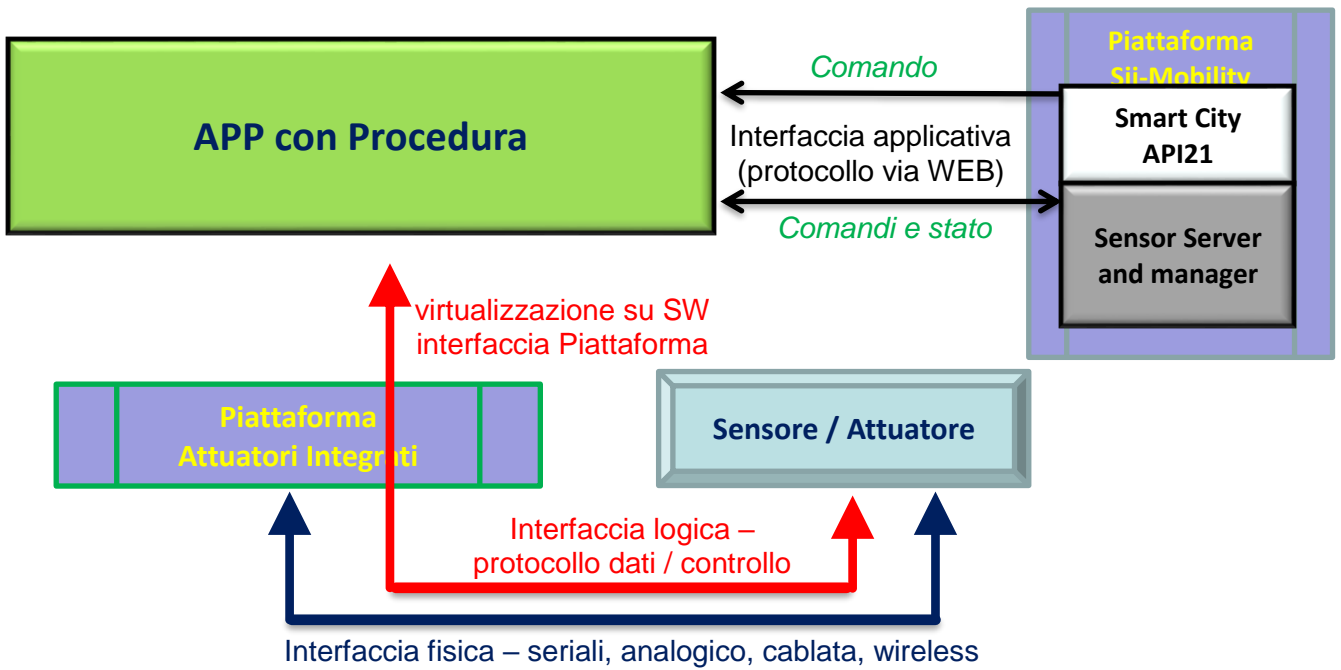


Figura 2 : notazione grafica generale

2.4.2 Schemi di dettaglio

Negli schemi di dettaglio è utilizzata una notazione semi-formale descritta in *Figura 3* ispirata ai diagrammi di Gane-Sarson, a loro volta derivati dalla notazione DFD (Data Flow Diagram).

Interfaccia, elemento esterno		Processo, attività	
Database, archivio, file		Flusso di dati	

Figura 3 : notazione grafica di dettaglio

Negli schemi si rispetta la convenzione sui colori dei flussi dati descritta alla sezione precedente, utilizzando il verde per quei flussi di dati che comprendono più casi di interfacciamento (p.es. sia fisico che logico).

2.5 Obiettivi

Questo documento intende sviluppare completamente l'architettura di dettaglio degli attuatori integrati, senza però duplicare informazioni già presenti in altri documenti di Sii-Mobility. In particolare:

- ✚ alcuni sensori che fanno parte degli attuatori integrati sono descritti in [SiiM 4], essendo prodotti della Attività 3.2. Per questi non si forniscono ulteriori informazioni ma si rimanda al documento indicato;
- ✚ Le App con Procedura (APP04, APP05, APP06) utilizzate per i diversi attuatori integrati sono descritte in altri documenti che al momento non sono determinati. In questo documento sarà inserito un riferimento, al momento "TBD", che verrà risolto nella revisione DE3.11b.

3 Struttura comune a tutti gli Attuatori Integrati

Gli Attuatori Integrati prodotti nel corso del progetto condividono, in toto o in parte, alcune scelte strategiche di base, che sono descritte di seguito.

Includono un apparato HW/SW progettato e sviluppato appositamente per il progetto, che nel seguito viene riferito come *Piattaforma*. Si interfaccia fisicamente con *Sensori* e *Attuatori* (e inoltre ne può contenere altri).

Per lo sviluppo e il supporto della *Procedura* di ciascun attuatore integrato si utilizza un *Tablet* di servizio, in modo da fornire un host espressamente pensato per applicazioni WEB based. Il Tablet è componente sostanziale dell'attuatore integrato.

La Piattaforma e il Tablet comunicano utilizzando gli apparati Bluetooth di cui entrambi sono equipaggiati. Realizzano complessivamente un meccanismo di comunicazione che permette uno scambio di comandi / controlli / informazioni in modalità wireless.

La *Figura 4* mostra la strutturazione descritta.

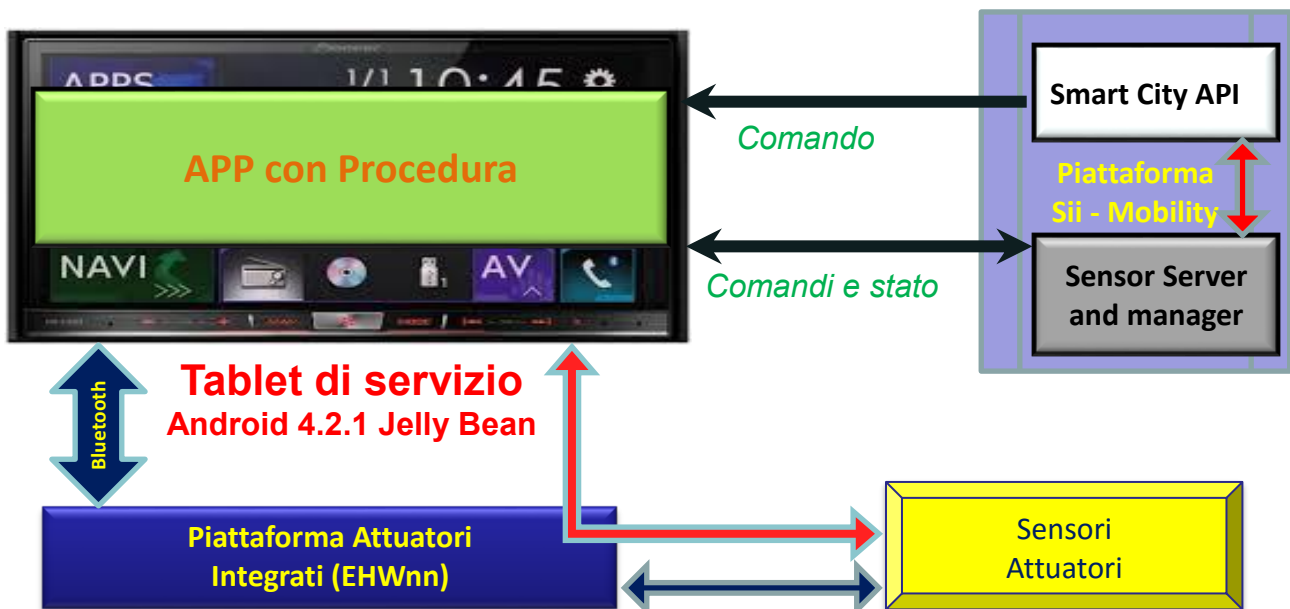


Figura 4 : Struttura comune degli Attuatori Integrati

I vantaggi offerti da questa organizzazione sono molteplici e rilevanti:

- chi sviluppa la Procedura opera in un ambiente “familiare”, costituito da un apparato commerciale, con caratteristiche ben standardizzate. Utilizza una installazione originale del sistema operativo Android, mentre un “porting” di Android sulla Piattaforma esporrebbe a rischi di comportamenti anomali,
- il canale di comunicazione Bluetooth può essere usato efficacemente sia per rendere disponibili alla Procedura i dati dei sensori contenuti nella Piattaforma stessa, sia per virtualizzare l’interazione con i sensori esterni alla piattaforma. Lo stesso vale per i comandi da inviare agli attuatori,
- la Piattaforma risulta essere sostanzialmente una estensione della Procedura, limitando di fatto il numero di partners coinvolti nella definizione e implementazione delle interfacce tra *Procedura* e *Piattaforma* e tra *Procedura* e *Sensori / Attuatori*,

- la Piattaforma si basa su una architettura hardware “open” che costituisce uno *standard de facto* per applicazioni di questo tipo ([Ref 1]).

3.1 Specifica di dettaglio della struttura comune

La *Figura 5* mostra un primo livello di specifica e ripropone lo schema comune di architettura nella notazione utilizzata per la descrizione di dettaglio. A questo livello occorre definire con maggior precisione le interfacce mostrate.

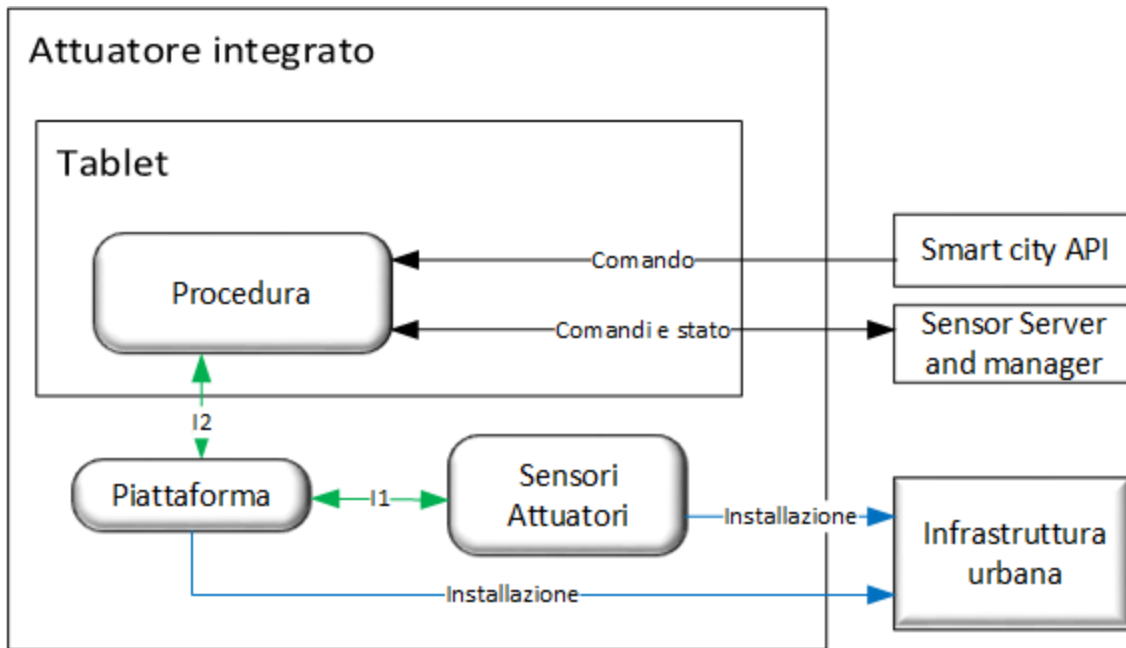


Figura 5 : Struttura comune - diagramma di contesto

In particolare:

- Interfacce esterne:
 - *Comando* e *Comandi a stato* sono interfacce applicative realizzate con scambi di dati su WEB. Non vengono ulteriormente dettagliate in questo documento;
 - *Installazione* della Piattaforma e dei Sensori / Attuatori è un'interfaccia fisica che coinvolge la collocazione dei vari elementi. Richiede un momento di coordinamento tra i partners che realizzano i contenitori ed i supporti dei vari elementi e quelli che si occupano della installazione nel contesto urbano. L'argomento è sviluppato alle sezioni: 4.2.1 per il Segnale Dinamico, 5.2.1 per il Direzionatore, 6.2.1 per la Rastrelliera Intelligente;
- Interfacce interne:
 - **I1** è l'interfaccia fisica e logica tra piattaforma e sensori / attuatori. È sviluppata nelle sezioni che seguono;
 - **I2** è l'interfaccia fisica e logica tra procedura e piattaforma. È sviluppata nelle sezioni che seguono;

3.1.1 Piattaforma

La *Figura 6* mostra il diagramma di contesto della Piattaforma. Si fa distinzione tra sensori / attuatori:

- *passivi*, per i quali l'interfaccia fisica è costituita da collegamenti elettrici e l'interfaccia logica dai livelli di tensione presenti sui collegamenti;
- *attivi*, che includono una CPU, per i quali l'interfaccia fisica è costituita da linee seriali e l'interfaccia logica dai messaggi scambiati secondo un certo protocollo.

I sensori attivi potranno anche avere collegamenti di comando / controllo analoghi a quelli dei sensori passivi in aggiunta allo scambio di messaggi.

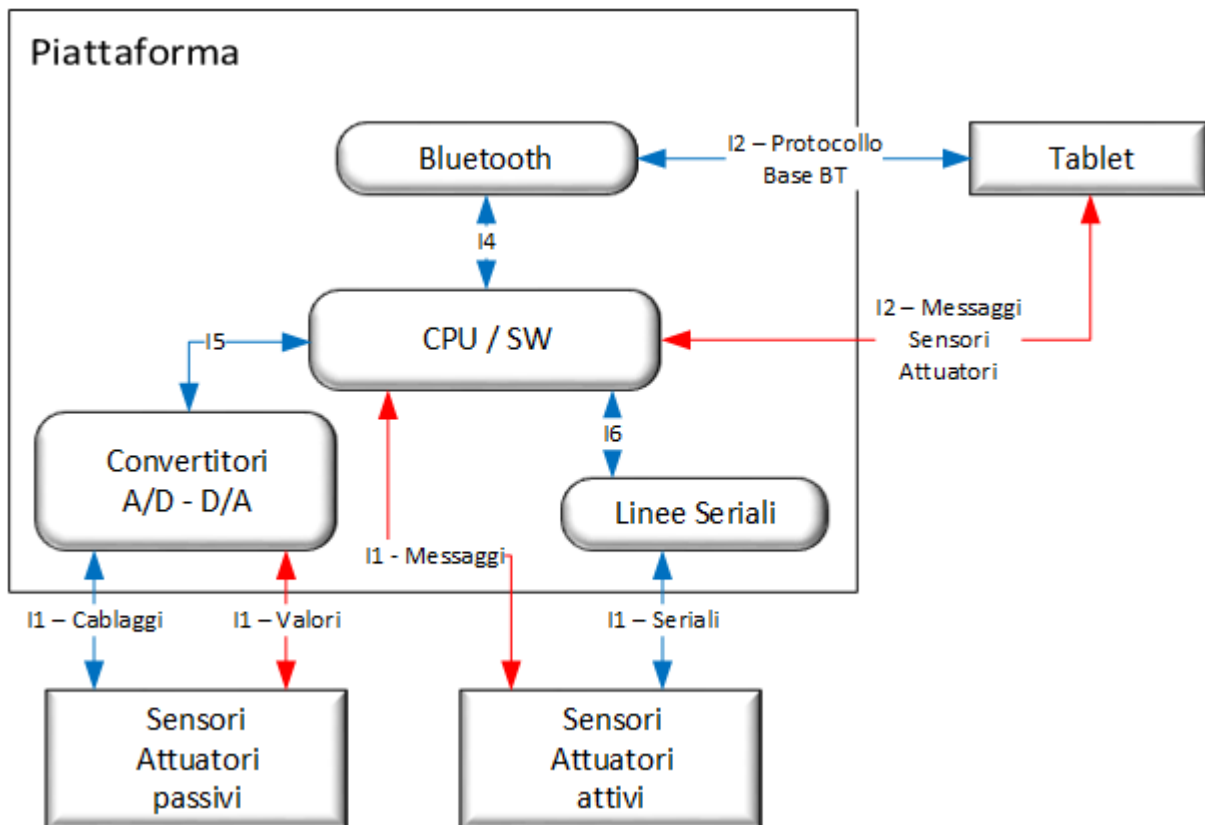


Figura 6 : Piattaforma – diagramma di contesto

La piattaforma include:

- convertitori Analogico/Digitale e Digitale/Analogico per la gestione dei collegamenti elettrici con i sensori e attuatori passivi e attivi. Hanno riscontro nelle porte di I/O della motherboard;
- linee seriali per lo scambio di messaggi con i sensori e attuatori attivi. Hanno riscontro nelle porte di I/O della motherboard;
- un dispositivo Bluetooth per le comunicazioni con il tablet;
- una scheda CPU (motherboard).

Tutti gli elementi qui sopra sono descritti con maggior dettaglio sezione 3.1.1.1

Principali attività e interfacce coinvolte:

- ❖ **gestione sensore / attuatore passivo** : *I1 – Cablaggi* rappresenta i collegamenti cablati con il sensore o attuatore mentre *I1 – Valori* rappresenta il significato della grandezza (tipicamente una tensione) che si acquisisce dal sensore / attuatore, o che si imposta per comandare l'attuatore. Secondo le convenzioni definite per *I2 – protocollo Base BT*, la piattaforma acquisisce i valori, li codifica in messaggi e li invia al tablet oppure decodifica i messaggi dal tablet (*I2 – Messaggi Sensori Attuatori*) e imposta i valori. È importante notare che viene trasmesso il dato grezzo del sensore, il suo significato (interfaccia logica) non è rilevante per la piattaforma, mentre lo è per la Procedura. Lo stesso vale per i comandi agli attuatori. L'interfaccia *I5* rappresenta le convenzioni, locali alla piattaforma (ECM-ELFI), sulla locazione dei convertitori e sulla loro gestione;
- ❖ **gestione sensore attivo** : i messaggi dai sensori / attuatori attivi (*I1 – Messaggi*) sono ritrasmessi senza modifiche al tablet e viceversa (*I2 – Messaggi Sensori Attuatori*) secondo le convenzioni definite per *I2 – protocollo Base BT*. L'interfaccia *I1 – Seriali* rappresenta solo il tipo di linea seriale, mentre *I6* rappresenta le convenzioni, locali alla piattaforma (ECM-ELFI), sulla locazione dei buffer e sulla gestione degli eventi TX/RX. Se un sensore / attuatore attivo ha anche collegamenti cablati, questi sono gestiti come nel caso precedente;
- ❖ **le interfacce fisiche** con i sensori / attuatori sono condivise tra chi sviluppa il sensore / attuatore (che le definisce) e chi sviluppa l'HW della piattaforma (ELFI) che le recepisce;
- ❖ **le interfacce logiche** con i sensori / attuatori sono condivise tra chi sviluppa il sensore / attuatore (che le definisce) e chi sviluppa la Procedura (NEGENTIS) che le recepisce;
- ❖ **gestione del collegamento Bluetooth** : il SW della piattaforma e quello del tablet definiscono e condividono le informazioni necessarie per realizzare un canale di comunicazione Bluetooth (interfaccia *I2 – protocollo Base BT*) con un suo protocollo. Utilizzano, rispettivamente, i servizi della piattaforma, interfaccia *I4*, e di Android (vedere *Figura 8*). Il protocollo comprende il formato dei messaggi (header, footer, etc.), messaggi di controllo e messaggi di dati. Nei messaggi dati sono contenuti (senza modifiche) i messaggi da e per sensori attivi e inoltre i messaggi generati dalla piattaforma per i sensori passivi.

Dunque l'interfaccia logica *I2 – Messaggi Sensori Attuatori* è costituita dall'OR di:

- *I1 – Valori*
- *I1 - Messaggi*

3.1.1.1 Elettronica

3.1.1.1.1 Motherboard

Come scheda madre viene individuata la scheda Arduino 101 (Genuino 101 in Europa), di recente realizzazione, che dispone di un modulo processore Intel Curie e integra un modulo Bluetooth Low Energy (BLE) e un modulo accelerometro/giroscopio a 6 assi.

La piattaforma si presenta con le stesse dimensioni, lo stesso pin-out e periferiche di Arduino Uno (rendendo le shields sviluppate per Arduino Uno compatibili con il 101), ma con architettura sensibilmente diversa. I pin I/O lavorano a 3.3V (rispetto ai 5V di Arduino uno), anche se protetti rispetto ai 5V.

La scheda si basa sul modulo Intel Curie, che contiene due processori, un x86 (Intel Quark) e un 32-bit ARC, entrambi a 32MHZ, diversamente da Arduino uno, basato su ATmega328, fornendo sulla carta un miglioramento delle prestazioni e un minor consumo.

Sul processore Intel Quark gira un sistema operativo RTOS, ancora in fase di sviluppo e per il quale sono previsti periodici aggiornamenti; mentre il processore ARC è dedicato a far girare i programmi di Arduino e a prendersi cura dell'I/O. I due processori operano simultaneamente e condividono la memoria; in questo modo sono disponibili per uso del software esterno 196kB (dei 384 kB) di flash memory, e di 24kB (degli 80 kB) di SRAM.

I due processori comunicano attraverso static mailboxes, con particolare riferimento a alcune operazioni (interfaccia attraverso porta USB; caricare il programma in flash; gestire l'interazione con il Bluetooth; generare PWM).

Il modulo Intel Curie è stato sviluppato in particolare per applicazioni wearable, con particolare attenzione ai bassi consumi, alla connettività (Bluetooth Low Energy) e con il sensore accelerometrico/giroscopio che rende semplice sviluppare analisi del movimento.

L'Arduino 101 ha 14 piedi digitali di input/output (dei quali 4 usabili come PWM), 6 piedi di input analogici, con connettore USB per la comunicazione seriale e per il caricamento degli sketches di Arduino, un connettore ICSP con segnali SPI, e piedi dedicati alla comunicazione I2C. Tutti i piedi operano a 3.3 volt e possono essere usati come interrupt.

Per ulteriori dettagli si veda ([Ref 2]).

3.1.1.1.2 Bluetooth Low Energy

Per quanto riguarda la connettività il protocollo Bluetooth Low Energy (o Bluetooth Smart, parte dello sviluppo Bluetooth 4.0) è stato sviluppato nell'ottica del basso costo e del basso consumo, a parità di distanza di comunicazione, per applicazioni di tipo IoT, ovvero con bassi data rate (non adatto per esempio a streaming audio o video). I sistemi operativi mobile e non, tra cui iOS, Android, BlackBerry, macOS, Linux, Windows8 e 10, lo supportano in modo nativo e tra i vantaggi di questo protocollo c'è quello di poter realmente interagire con una grande varietà di piattaforme mobili, telefoni, tablet e computer.

Il data-rate teorico del BLE è di 1 Mb/s, ma in realtà il massimo data rate realmente raggiungibile è intorno ai 10kB/s, con sensibili differenze a seconda del sistema operativo che lo utilizza (con un throughput da meno di 3kB/s a 13 kB/s circa).

Diversamente dal Bluetooth standard basato su connessione seriale asincrona (UART), il Bluetooth LE agisce in modo diverso. Esistono due funzioni principali che ciascun elemento può compiere: pubblicare notizie (e in questo caso si chiama "peripheral device", che agisce come un server) o leggerle ("central device", che agisce come client). Più central devices possono accedere alle notizie pubblicate da una singola periferica; queste ultime sono presentate come servizi (che possono essere standard o personalizzati), e provvisti di identificativo numerico univoco (UUID), ciascuno suddiviso in caratteristiche. Un meccanismo di notifica informa quando i dati sono cambiati. Questa struttura è chiamata publish-and-subscribe model. (si tende a notificare solo quando ci sono cambiamenti e non su intervalli di tempo regolari). I Peripheral devices forniscono per esempio come caratteristiche i dati di un sensore, oppure permettono di leggere/scrivere su alcune caratteristiche per comandare attuatori.

Il valore di ciascuna caratteristica può essere lunga al massimo 20 byte: questo è un vincolo cruciale del protocollo.

Un central device (che agisce come client) ha 4 funzioni rispetto a una caratteristica: lettura (chiedere alla periferica di fornire il valore della caratteristica), scrittura (modificarne il valore, come con gli attuatori), Indicazione e notifica (chiedere alla periferica di mandare in modo continuo i valori aggiornati della caratteristica, senza chiederli ogni volta).

GAP (General Advertising Profile) è il modo in cui ciascun dispositivo BLE rende nota la sua esistenza; mentre il GATT (General ATtribute Profile) definisce servizi e caratteristiche e abilita le operazioni di lettura/scrittura/notifica.

3.1.1.1.3 Interfacciamento dei sensori

Questo aspetto è sviluppato nel seguito, per ciascuno degli attuatori integrati.

3.1.1.2 Software

3.1.1.2.1 Casi d'uso

Per tutti gli attuatori integrati, la Piattaforma è staticamente associata ad un Tablet di servizio ed inoltre per entrambi si prevede il funzionamento continuo. Di conseguenza il caso di uso corrispondente alla connessione dei due elementi via Bluetooth avviene sporadicamente e sempre tra gli stessi due attori.

Per il resto esiste un unico caso di uso corrispondente al funzionamento nominale, in cui la *Piattaforma* agisce come estensione della *Procedura* per la interazione con sensori e attuatori. Questo caso di uso è realizzato dalla architettura SW definita qui di seguito.

3.1.1.2.2 Architettura di base

La Figura 7 mostra la strutturazione di primo livello del software. Costituisce il dettaglio dell'elemento "CPU / SW" mostrato in Figura 6.

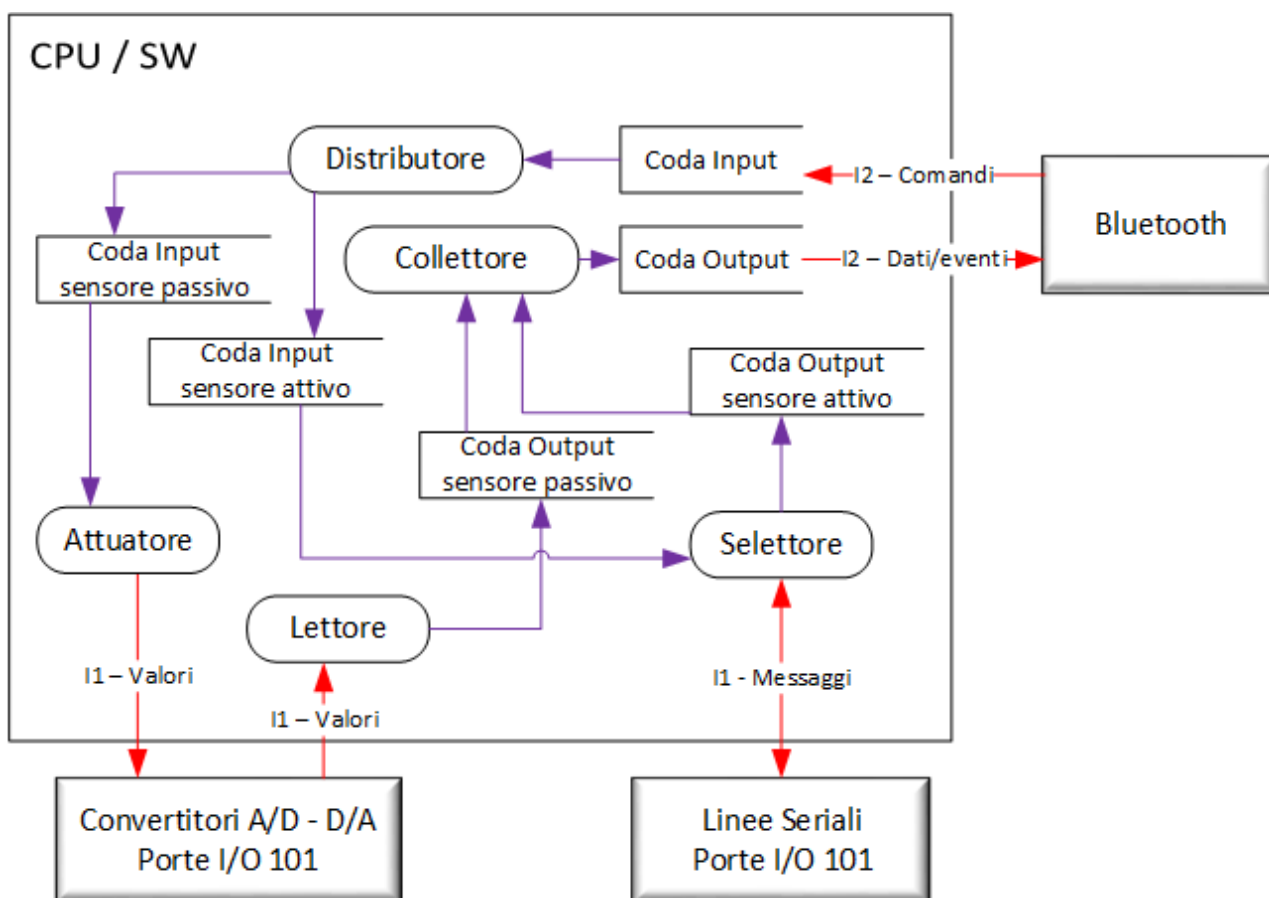


Figura 7 : Architettura di base del software

Esistono un certo numero di code di messaggi e alcuni processi che le gestiscono. In particolare:

- una coda di Input e una coda di Output destinate a contenere messaggi formattati come da Protocollo base Bluetooth. La prima contiene comandi emessi dalla Procedura del Tablet verso i sensori/attuatori, nel seguito indicati genericamente come “periferiche”, la seconda dati ed eventi provenienti dalle periferiche e diretti alla Procedura;
- una coda di Input e una coda di Output dedicate per ogni periferica passiva e per ogni periferica attiva. Una periferica di tipo misto, ovvero con porte seriali e porte analogiche, avrà due coppie di code associate;
- il processo Distributore preleva messaggi dalla coda dei comandi provenienti dal Tablet, analizza il frame del Protocollo base Bluetooth per riconoscere la periferica a cui sono destinati, rimuove il frame e inserisce il testo del messaggio nella coda della periferica. A seconda dei casi esegue elaborazioni aggiuntive, ad esempio in caso di periferica mista può “sviluppare” il messaggio originale e inserire messaggi in più di una coda;
- processi Attuatori che prelevano messaggi dalla coda di input di una periferica passiva (o mista) e analizza il testo del messaggio per selezionare una o più azioni da eseguire su una o più porte di I/O della motherboard. Possono eseguire azioni aggiuntive, ad esempio inserire un messaggio di evento nella coda di output della periferica per riportare l’esito dell’esecuzione del comando. I processi Attuatori possono definire drivers di interrupt collegati alle porte della periferica e sincronizzarsi e cooperare con tali drivers;
- processi Lettori che acquisiscono il valore di tensione su una porta di I/O della motherboard collegata alla periferica, costruiscono un messaggio che codifica la porta e il valore e lo inseriscono nella coda di output della periferica. La lettura di un valore potrà comportare una o più azioni da eseguire su una o più porte di I/O della motherboard e/o la cooperazione con drivers di interrupt appositamente definiti;
- processi Selettori destinati a smistare i messaggi dalle code di input delle periferiche alla opportuna linea seriale della motherboard e ad acquisire messaggi provenienti dalle periferiche sulle linee seriali della motherboard e inserirli nelle code di output corrispondenti.;
- un processo Collettore che preleva messaggi da tutte le code di output delle periferiche secondo criteri di priorità, li codifica con il frame del Protocollo base Bluetooth e li inserisce nella coda di output generale.

In generale, esiste un processo Attuatore e un processo Lettore per ciascuna periferica. Se conveniente, Attuatore e Lettore di una periferica potranno essere realizzati da un singolo processo.

In generale esiste un processo Selettore per ciascuna linea seriale della motherboard. I processi Selettori non eseguono elaborazioni sul testo dei messaggi, dal momento che il relativo protocollo è definito direttamente dallo sviluppatore della periferica e dallo sviluppatore della Procedura del Tablet.

La gestione dei processi è supportata da un opportuno sistema operativo, in corso di selezione. Ulteriori dettagli su questo aspetto saranno forniti

3.1.2 Tablet di servizio

La [Figura 8](#) mostra il diagramma di contesto del Tablet di servizio.

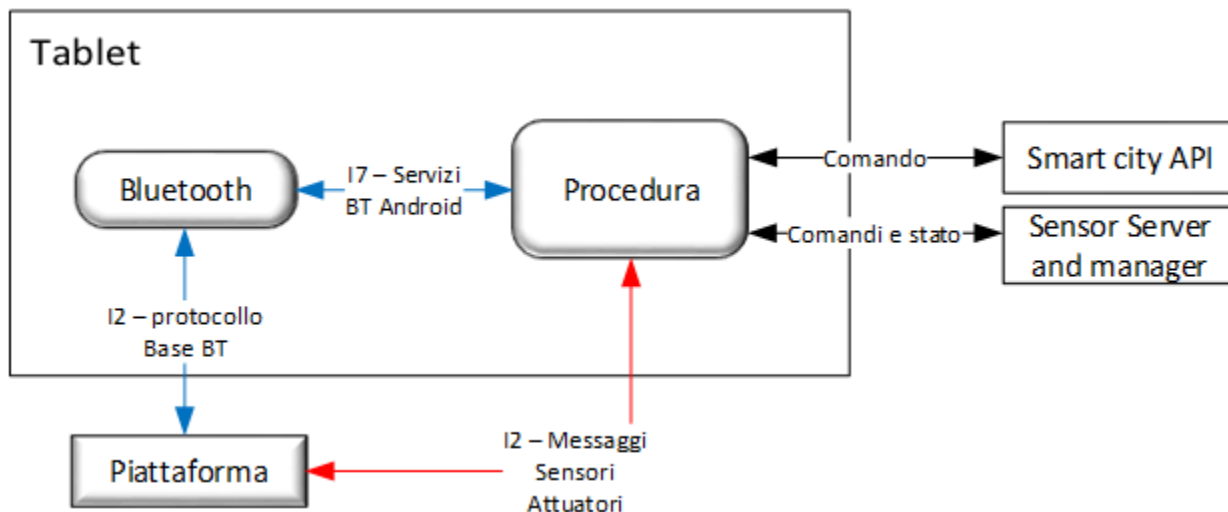


Figura 8 : Tablet – diagramma di contesto

Di seguito sono indicate le caratteristiche minime che il Tablet di servizio deve possedere per supportare efficacemente i Processi che vi sono installati:

- ✚ Sistema Operativo **Android 4.2.1 Jelly Bean** o revisioni successive retro-compatibili;
- ✚ Bluetooth Low Energy

3.1.2.1 Procedura

La *Procedura* ha un comportamento differenziato a seconda del tipo specifico di Attuatore Integrato su cui viene installato. Si rimanda alle relative sezioni 4.3 (Segnale Dinamico), 5.3 (Direzionatore), 6.3 (Rastrelliera Intelligente).

3.1.3 Sensori e attuatori

Gli attuatori integrati hanno una struttura modulare che permette di collegare svariate tipologie di sensori e attuatori. In particolare, la piattaforma consente il collegamento di un certo numero di sensori e attuatori sia di tipo attivo che passivo. Quali poi sono quelli effettivamente utilizzati per ciascun tipo di attuatore integrato dipende dalle funzioni che devono essere realizzate.

Complessivamente, allo stato attuale, sensori e attuatori utilizzati sono:

- Telecamere Intelligenti, sensori di tipo attivo utilizzate per il Segnale Dinamico. Sono realizzate nel contesto dell'Attività 3.2 e descritte in [SiiM 4];
- Visualizzatore LED, attuatore di tipo attivo utilizzato per il Segnale Dinamico e per il Direzionatore. È realizzato nel contesto dell'Attività 3.3 e la sua specifica di dettaglio è sviluppata in questo documento, alla sezione 4.4.1.2;
- Sensori e attuatori utilizzati solo per la Rastrelliera Intelligente. Sono realizzati nel contesto dell'Attività 3.3. La loro specifica di dettaglio è sviluppata in questo documento, alla sezione 6.4.

In corso progetto sarà possibile pianificare il collegamento di altri sensori / attuatori resi disponibili dall'Attività 3.2 oppure internamente all'Attività 3.3, allo scopo di supportare nuove funzioni oppure migliorare l'implementazione di quelle già previste. Si tenga comunque presente che non è

possibile utilizzare barriere mobili nel contesto delle porte ZTL (Segnale Dinamico) in quanto potrebbero creare intralcio al transito dei mezzi di soccorso;

3.1.4 Protocollo base Bluetooth

In questa sezione si sviluppa la specifica del protocollo di base Bluetooth, che viene indicato come *I2 – protocollo Base BT* in *Figura 6* e *Figura 8*.

La specifica deve essere condivisa tra chi sviluppa il software della Piattaforma e della Procedura, in modo da utilizzare in modo consistente le funzionalità Bluetooth presenti sulla Piattaforma e sul Tablet per realizzare un meccanismo di trasporto wireless di messaggi.

In prima istanza, ci si propone di sviluppare il protocollo seguendo lo schema di comunicazione “nativo” Bluetooth Low Energy, descritto in 3.1.1.1.2. Qualora in corso lavori il throughput si rivelasse insufficiente per supportare tutte le funzioni previste, si provvederà ad un riesame generale del meccanismo per migliorarlo o per selezionarne uno alternativo.

Questa sezione sarà completata nella revisione DE3.11**b** di questo documento a cura di DISIT.

4 Segnale Dinamico

4.1 Architettura generale

L'architettura generale, contestualizzata in Sii-Mobility, del Segnale Dinamico è mostrata nella figura seguente.

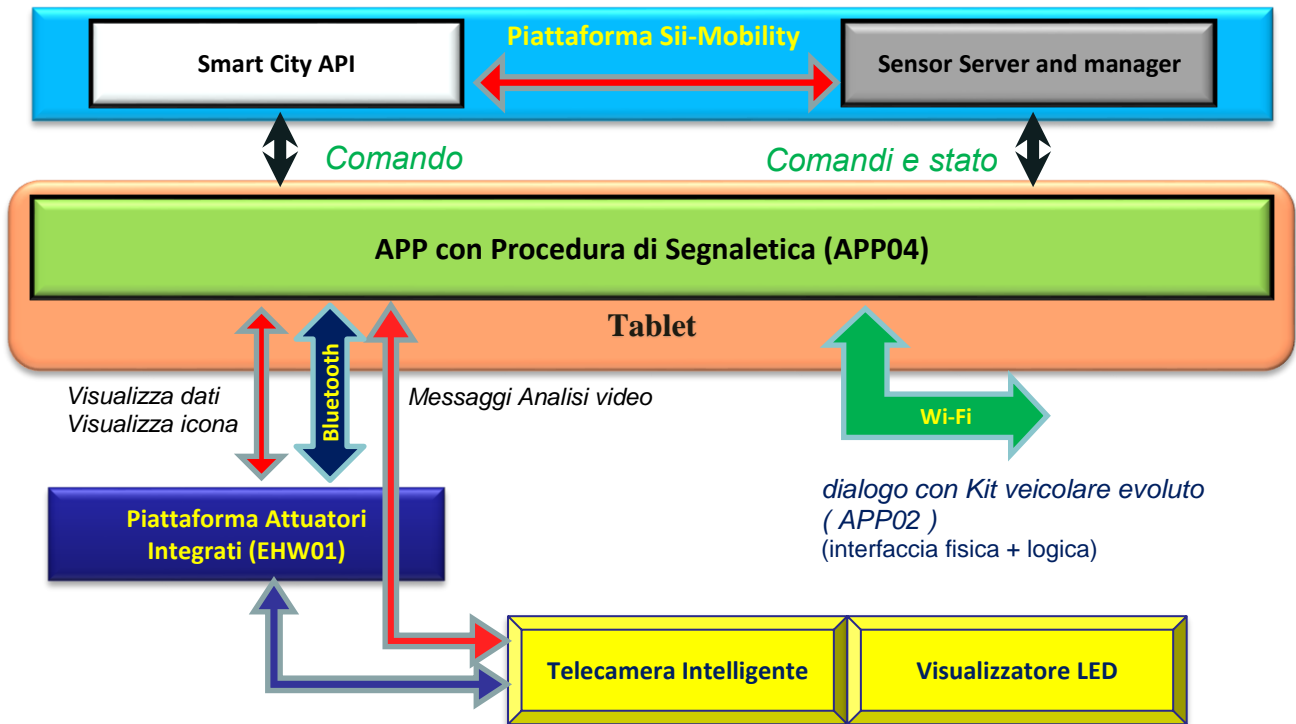


Figura 9 : Architettura generale del Segnale Dinamico

Il segnale dinamico è progettato per essere utilizzato per le porte di accesso ai parcheggi e alle ZTL. È in grado di presentare informazioni grafiche e testuali mediante un attuttore *Visualizzatore LED* per informare l'utente riguardo le autorizzazioni di accesso. La *Procedura* riceve dinamicamente dalla centrale tali informazioni e utilizza i servizi della Piattaforma per visualizzarle sull'attuttore. La variazione dinamica delle condizioni di accessibilità, eseguita in modo coordinato su un certo numero di segnali dinamici permette di realizzare la funzione di riconfigurazione dinamica della ZTL.

Mediante una *Telecamera Intelligente* è in grado di rilevare l'avvicinamento di un veicolo ed identificarlo (numero di targa). La *Procedura* provvede a comunicare con la centrale per verificare eventuali autorizzazioni specifiche per il veicolo. Se il veicolo non risulta autorizzato e supera il segnale dinamico, allora la *Telecamera Intelligente* provvede ad acquisire un'immagine, che viene inviata alla centrale per i provvedimenti del caso.

Una ulteriore *Telecamera Intelligente* è in grado di rilevare lo stato di "strada libera" nel tratto immediatamente successivo al segnale dinamico per ritardare commutazioni dello stato del segnale che renderebbero sanzionabili veicoli appena transitati correttamente.

Le due telecamere sono connesse a un apparato di tipo “*Smart Node*”, descritto in [SiiM 4] alla sezione 5, che a sua volta si interfaccia con la Piattaforma utilizzando la linea seriale I²C.

4.2 Piattaforma

Nelle sezioni che seguono si dettaglia l’insieme dei moduli hardware e software che vengono utilizzati per costruire il Segnale Dinamico.

Sono inoltre specificati nel dettaglio sensori e attuatori utilizzati connessi al Segnale Dinamico.

4.2.1 Meccanica, contenitori, carpenterie, installazione

Al momento non è possibile sviluppare questo aspetto, che richiede analisi congiunta con i gestori del traffico, dei parcheggi e di ZTL per individuare soluzioni funzionali al progetto, in linea con leggi e regolamenti vigenti e compatibili con le locazioni in cui potrà essere installato il contenitore.

Si pianifica in ogni caso un unico contenitore, in grado di contenere il Visualizzatore LED, la Piattaforma, il Tablet di servizio e lo Smart Node. Le Telecamere Intelligenti potranno essere posizionate all’interno dello stesso contenitore oppure esternamente, sul contenitore stesso o su altro idoneo supporto.

Occorrerà invece concordare coi gestori pubblici posizionamento e fissaggio più opportuni, trovando contestualmente un valore ragionevole per le sue dimensioni.

Questa sezione sarà completata nella revisione DE3.11**b** di questo documento a cura di ARGOS.

4.2.2 Hardware

4.2.2.1.1 Motherboard

Il Segnale Dinamico utilizza la Piattaforma selezionata per la struttura comune a tutti gli attuatori integrati, descritta alla sezione 3.1.1.1.

4.2.2.1.2 Interfacciamento dei sensori e attuatori

La motherboard interagisce con i sensori descritti nelle sezioni seguenti, ovvero una telecamera intelligente (4.4.1.1) e un Visualizzatore LED (4.4.1.2).

Le alimentazioni dei sensori saranno fornite tutte da linea esterna e non dalle porte della piattaforma: questa è una soluzione che garantisce maggiore robustezza e diminuisce il rischio di interferenze nel funzionamento di tutto il kit. La Figura 10 e la Tabella 1 mostrano uno schema generale e il dettaglio del collegamento delle interfacce fisiche alle porte di Arduino 101.

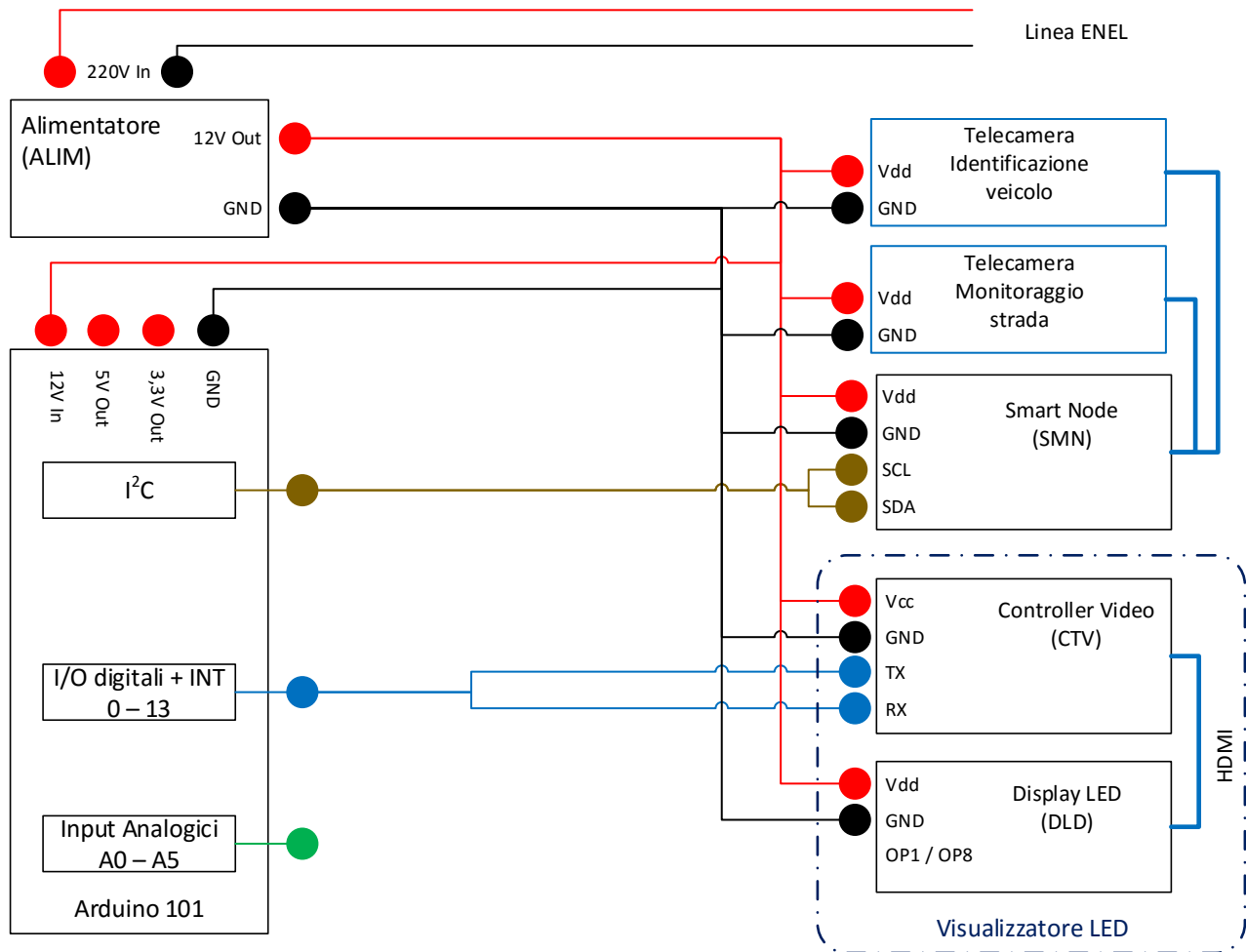


Figura 10 : Schema generale del Segnale Dinamico

Segnale Dinamico - Connessioni fisiche tra piattaforma e sensori							
Analog IN	A0	A1	A2	A3	A4	A5	
I/O/INT Ports	0/RX	1/TX	2°	3*	4	5*°	6*°
I/O/INT Ports	7	8°	9*	10°	11°	12°	13°
Power	ATN/SS	IOREF	RESET	3,3	5	GND	Vin
						Ground	12V
Other	SCL	SDA	AREF	ICSP/ MISO	ICSP/ MOSI	ICSP/ SCK	ICSP
	SMN SCL	SMN SDA					

Tabella 1: Segnale Dinamico - connessioni fisiche tra motherboard e sensori

*: porte che supportano PWM

°: porte che supportano interrupt sui cambiamenti di stato

4.2.3 Software

La struttura di base del SW del Segnale Dinamico corrisponde a quella comune a tutti gli attuatori integrati, descritta alla sezione 3.1.1.2.

4.3 APP con Procedura di Segnaletica, APP04

La APP con Procedura APP04 è descritta in altro documento di progetto, che al momento non è determinato. Il riferimento, attualmente “TBD”, verrà risolto nella revisione DE3.11b di questo documento a cura di DISIT.

4.4 Sensori / attuatori

Nel seguito una sezione è dedicata a ciascuno dei sensori e attuatori connessi al Segnale Dinamico.

4.4.1.1 Telecamera Intelligente, SN07

Di seguito i riferimenti nel deliverable dedicato ai sensori innovativi.

Si noti che alla Piattaforma sono collegate due telecamere, la prima rivolta verso i veicoli in avvicinamento per identificarli, la seconda rivolta verso il tratto di strada (parte della ZTL) che è controllato dal Segnale Dinamico per rilevare se ci sono veicoli in transito e/o fotografare i veicoli in caso di infrazione.

Le due telecamere sono connesse a un apparato di tipo “Smart Node”, descritto in [SiiM 4] alla sezione 5, che a sua volta si interfaccia con la Piattaforma utilizzando la linea seriale I²C.

4.4.1.1.1 Interfacce

Questo sensore è descritto nel documento [SiiM 4]. In particolare:

- l’interfaccia fisica è descritta alla sezione 3.1.1.2
- l’interfaccia logica è descritta alla sezione 3.1.1.3

4.4.1.2 Visualizzatore LED

Il Visualizzatore LED è composto da due sezioni principali:

- un Controller Video che ha il compito di gestire dinamicamente un archivio di descrittori HTML, generare le immagini corrispondenti e inviarle su una uscita HDMI. Il Controller Video è quindi fornito di un opportuno motore di rendering. È connesso con la Piattaforma, da cui riceve messaggi che richiedono la visualizzazione di un descrittore HTML e, per alcuni descrittori, indicazioni sul contenuto di campi testuali dinamici.
- un Display con tecnologia LED connesso all’uscita HDMI del Controller Video che visualizza le immagini generate. Le proprietà del display e opportuni accorgimenti della struttura che lo contiene concorrono ad ottenere una visibilità ottimale

Le interfacce fisiche e logiche verso la Piattaforma sono quindi tutte e solo quelle del Controller Video e sono descritte qui di seguito.

4.4.1.2.1 Interfaccia fisica

È costituita da una singola linea seriale TTL utilizzata in modo unidirezionale, ovvero sono previsti messaggi dalla Piattaforma al Controller Video ma non vice-versa.

4.4.1.2.2 Interfaccia logica

I messaggi dalla Piattaforma contengono l'identificatore unico del descrittore HTML da visualizzare e, qualora questo comprenda dei campi testuali dinamici, il contenuto da visualizzare per ciascun campo.

4.4.1.2.3 Architettura

L'architettura di base del Visualizzatore LED è mostrata nella Figura 11 seguente:

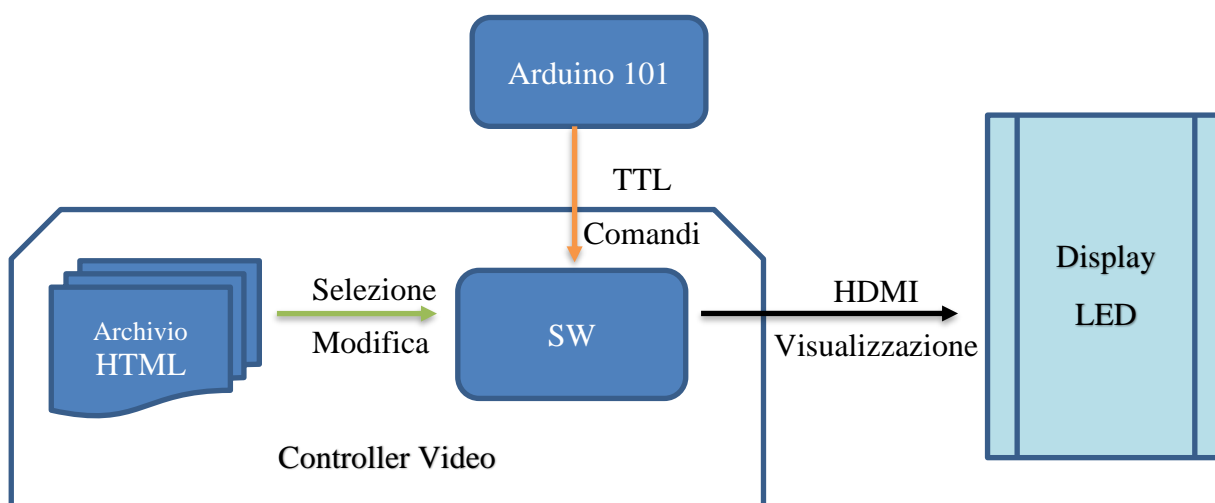


Figura 11 : Architettura del Visualizzatore LED

4.4.1.2.4 Specifica di dettaglio

Il Controller Video è realizzato programmando una board Raspberry o similare con adeguata potenza di calcolo e disponibilità di:

- porta (mini) HDMI
- porta seriale TTL
- software di rendering HTML
- (micro) SD slot per archivio di descrittori HTML

Al momento il miglior candidato sembra essere Raspberry Pi Zero, ma una scelta definitiva non è ancora stata fatta.

Per quanto riguarda il Display LCD, dal momento che l'unico requisito fondamentale è un ingresso HDMI esiste una vasta scelta tra i prodotti in commercio. La selezione definitiva sarà fatta in corso lavori a seguito di sperimentazioni per valutare la visibilità in varie condizioni di illuminazione una volta messi a punto gli accorgimenti che si intende utilizzare nel supporto / contenitore del display.

Questa sezione sarà aggiornata e completata nella revisione DE3.11b di questo documento a cura di ELFI.

5 Direzione

5.1 Architettura generale

L'architettura generale, contestualizzata in Sii-Mobility, del Direzione è mostrata nella figura seguente.

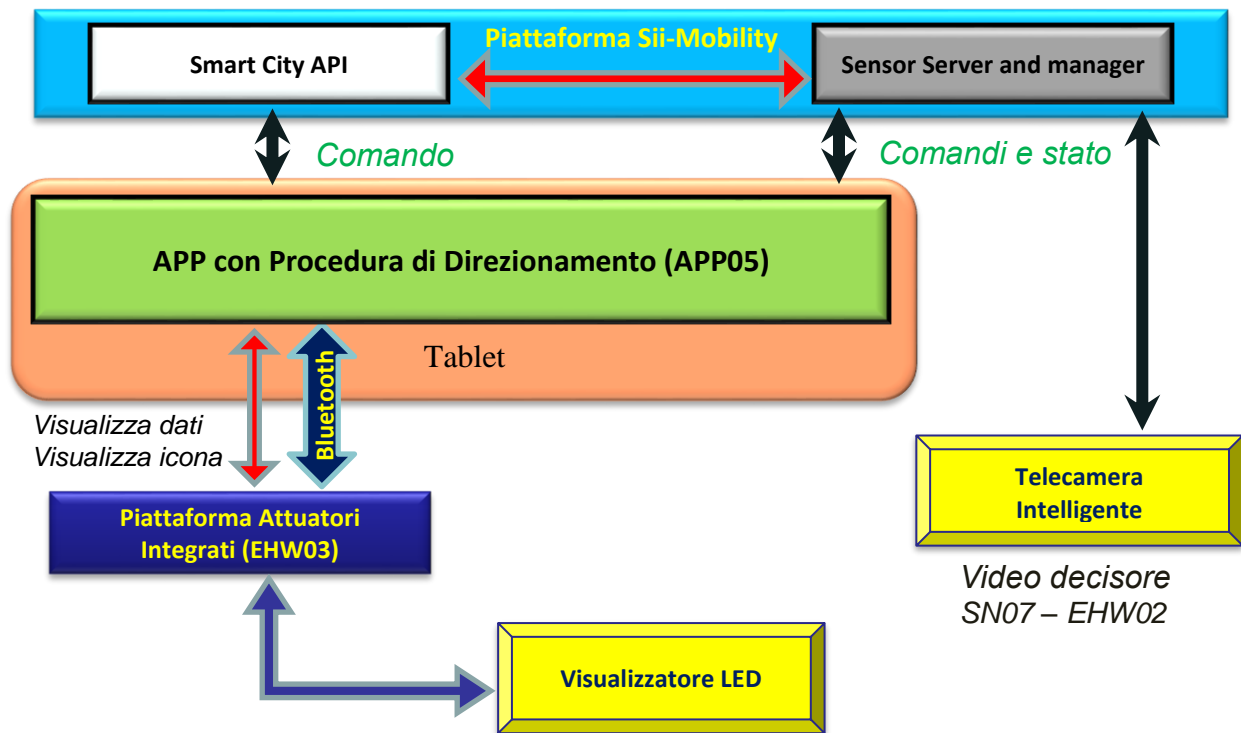


Figura 12 : Architettura generale del Direzione

Il Direzione è progettato per essere utilizzato nel contesto della modifica dinamica del senso di percorrenza di una strada o corsia. Ha la funzione di presentare icone / scritte che forniscono informazioni di libero accesso oppure di accesso vietato / limitato mediante l'attuatore *Visualizzatore LED*.

Chiaramente, la decisione di invertire il senso unico di una strada può essere attuata in sicurezza solo se in quel momento non ci sono veicoli in transito. Per questo motivo, una *Telecamera Intelligente* viene utilizzata per rilevare lo stato di "strada libera".

La telecamera **non** è collegata direttamente all'attuatore integrato ma alla centrale Sii-Mobility per motivi di sicurezza. Infatti, non è possibile affidarsi al giudizio di un dispositivo elettronico intelligente per acquisire un'informazione che, qualora errata, ha un impatto diretto sulla sicurezza delle persone. L'informazione deve essere necessariamente confermata da un operatore umano.

Per questi motivi, la telecamera dialoga direttamente con la centrale, trasmettendo anche un'informazione visiva in base alla quale l'operatore può confermare e autorizzare la commutazione. La telecamera non fa quindi parte dell'attuatore integrato e viene mostrata solo per completezza.

La procedura complessiva, realizzata dalla centrale Sii-Mobility, consiste in:

- impostazione di entrambi gli attuatori posti agli estremi della strada in stato di "accesso vietato",

- attendere conferma della avvenuta commutazione,
- attendere l'informazione di "strada libera" dalla telecamera,
- chiedere conferma da parte dell'operatore,
- commutare allo stato "accesso consentito" uno dei due attuatori integrati.

Il ruolo della *Procedura* è ricevere le richieste di cambio di stato dell'attuatore integrato, attuarle, verificare l'attuazione e fornire conferma alla centrale.

5.2 Piattaforma

Nelle sezioni che seguono si dettaglia l'insieme dei moduli hardware e software che vengono utilizzati per costruire il Direzionatore.

Sono inoltre specificati nel dettaglio sensori e attuatori utilizzati connessi al Direzionatore.

5.2.1 Meccanica, contenitori, carpenterie, installazione

Al momento non è possibile sviluppare questo aspetto, che richiede analisi congiunta con i gestori del traffico per individuare soluzioni funzionali al progetto, in linea con leggi e regolamenti vigenti e compatibili con le locazioni in cui potrà essere installato il contenitore.

Si pianifica in ogni caso un unico contenitore, in grado di contenere il Visualizzatore LED, la Piattaforma e il Tablet di servizio. Occorrerà invece concordare coi gestori pubblici posizionamento e fissaggio più opportuni, trovando contestualmente un valore ragionevole per le sue dimensioni.

Questa sezione sarà completata nella revisione DE3.11b di questo documento a cura di ARGOS.

5.2.2 Hardware

5.2.2.1 Motherboard

Il Direzionatore utilizza la Piattaforma selezionata per la struttura comune a tutti gli attuatori integrati, descritta alla sezione 3.1.1.1.

5.2.2.1.2 Interfacciamento dei sensori e attuatori

La motherboard interagisce unicamente con il Visualizzatore LED, descritto alla sezione 4.4.1.2. La Figura 10 e la Tabella 1 valgono anche per il caso del Direzionatore per descrivere le connessioni con il Visualizzatore LED.

5.2.3 Software

La struttura di base del SW del Direzionatore corrisponde a quella comune a tutti gli attuatori integrati, descritta alla sezione 3.1.1.2.

5.3 APP con Procedura di Direzionamento, APP05

La APP con Procedura APP05 è descritta in altro documento di progetto, che al momento non è determinato. Il riferimento, attualmente "TBD", verrà risolto nella revisione DE3.11b di questo documento a cura di DISIT.

5.4 Sensori / attuatori

Il Direzionatore utilizza unicamente il Visualizzatore LED, descritto alla sezione 4.4.1.2.

6 Rastrelliera Intelligente

6.1 Architettura generale

L'architettura generale, contestualizzata in Sii-Mobility, della Rastrelliera Intelligente è mostrata nella figura seguente.

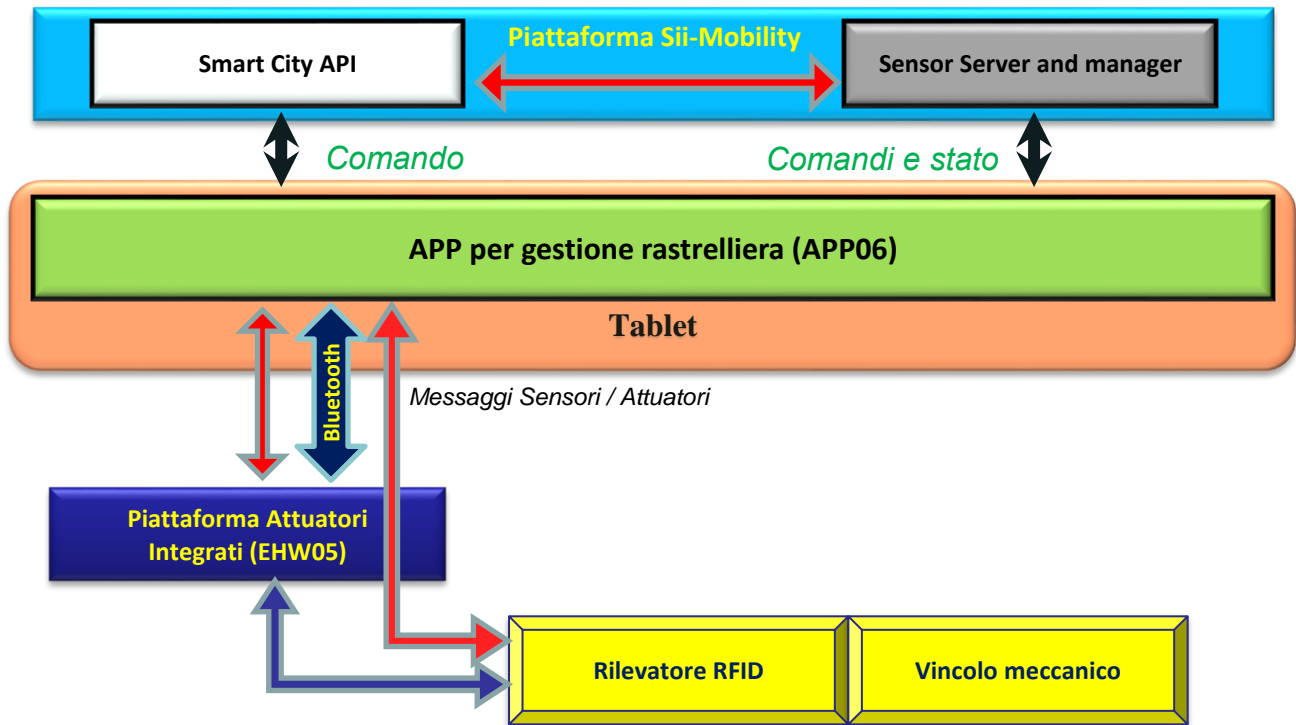


Figura 13 : Architettura generale della Rastrelliera Intelligente

La Rastrelliera Intelligente è progettata per realizzare la funzione **AATF** (autonomous anti-theft feature) principalmente o totalmente sull'infrastruttura di terra, aggiungendo elementi anche complessi all'infrastruttura ed eventuali elementi molto semplici sul kit bike.

La Rastrelliera Intelligente è in grado di comunicare autonomamente con la centrale Sii-Mobility. L'architettura è in linea con la struttura generalizzata comune a tutti gli attuatori integrati, descritta alla sezione 3.

Per quanto riguarda il meccanismo per rilevare o impedire il furto si hanno almeno due possibilità:

- i. il veicolo è equipaggiato con un tag RFID e la Piattaforma della Rastrelliera Intelligente è in grado di rilevarne la (mancata) presenza. Questo provoca un allarme a meno che dalla centrale Sii-Mobility non sia stata inviata un'autorizzazione, cosa che avviene quando la Mobile App dello Smartphone Utente ha completato l'associazione con il veicolo ed informa di questo la centrale;
- ii. la Rastrelliera Intelligente è dotata di vincoli meccanici che impediscono il prelievo del veicolo. Questi sono comandati e controllati dalla Piattaforma, che a sua volta è pilotata dalla Procedura del Tablet. Il comando di sblocco viene inviato dalla centrale Sii-Mobility alla Procedura quando la Mobile App dello Smartphone Utente ha completato l'associazione con il veicolo ed informa di

questo la centrale. Quando l'Utente restituisce la bici, il vincolo meccanico viene ripristinato (occorre verificare il buon esito dell'operazione) e solo allora si chiude l'associazione con l'Utente.

6.2 Piattaforma

Nelle sezioni che seguono si dettaglia l'insieme dei moduli hardware e software che vengono utilizzati per costruire la Rastrelliera Intelligente.

Sono inoltre specificati nel dettaglio sensori e attuatori utilizzati connessi alla Rastrelliera Intelligente.

6.2.1 Meccanica, contenitori, carpenterie, installazione

Al momento non è possibile sviluppare operativamente questo aspetto, che richiede analisi congiunta con i gestori del traffico, dell'urbanistica e dei servizi di bike-sharing per individuare soluzioni funzionali al progetto, in linea con leggi e regolamenti vigenti e compatibili con le locazioni in cui potrà essere installata la Rastrelliera Intelligente.

Al momento è stata fatta una proposta di progetto di una rastrelliera con la prerogativa di essere flessibile a fronte di varie categorie di biciclette e dotata di un meccanismo di blocco/sblocco della bicicletta efficace ed efficiente. Una descrizione dettagliata della (possibile) infrastruttura è contenuta nell'allegato "ARGOS SII-MOBILITY-RAST-BICI-001-R01-15-02-2017" ([Ref 3]).

Questa sezione sarà completata nella revisione DE3.11b di questo documento a cura di DIEF e ARGOS.

6.2.2 Hardware

6.2.2.1.1 Motherboard

Il Segnale Dinamico utilizza la Piattaforma selezionata per la struttura comune a tutti gli attuatori integrati, descritta alla sezione 3.1.1.1.

NOTA: è in fase di studio un sistema di gestione della rastrelliera che differisce per la comunicazione della quantità di dati raccolti dalla motherboard attraverso non più la catena BT/TABLET/SSM (*Service Server Manager*), ma mediante una shield GSM/GPRS che colloquia direttamente al SSM tramite richieste RES e ricezione di messaggi JSON (protocollo HTTP).

Tale studio potrebbe avere ripercussioni sulla scelta della motherboard, in quanto non sarebbe più di interesse usare il modulo BT Low Energy, ma piuttosto il modulo GSM/GPRS.

Una decisione su questo punto sarà presa in corso lavori e riportata e motivata nella revisione DE3.11b di questo documento a cura di DIEF.

6.2.2.1.2 Interfacciamento dei sensori e attuatori

La motherboard interagisce con i sensori / attuatori descritti nelle sezioni seguenti, ovvero Rilevatore RFID (6.4.1), e Vincolo Meccanico (6.4.2).

Le alimentazioni dei sensori / attuatori saranno fornite tutte da linea esterna e non dalle porte della piattaforma: questa è una soluzione che garantisce maggiore robustezza e diminuisce il rischio di interferenze nel funzionamento di tutto il kit.

Di seguito viene definita l'architettura di sistema per il monitoraggio di N stalli della bike che avviene secondo protocollo di comunicazione I^2C (vedi Figura 14 a). In Figura 14 b invece è definito l'insieme di connessioni per il comando dell'attuatore lineare a solenoide che permette il

gancio/sgancio della bike al singolo stallo: anche in questo caso lo schema è fatto per rappresentare uno scenario di N stalli che equipaggiano la rastrelliera. La Tabella 2 riassuntiva dei collegamenti invece è compilata prendendo in esame un solo stallo della rastrelliera.

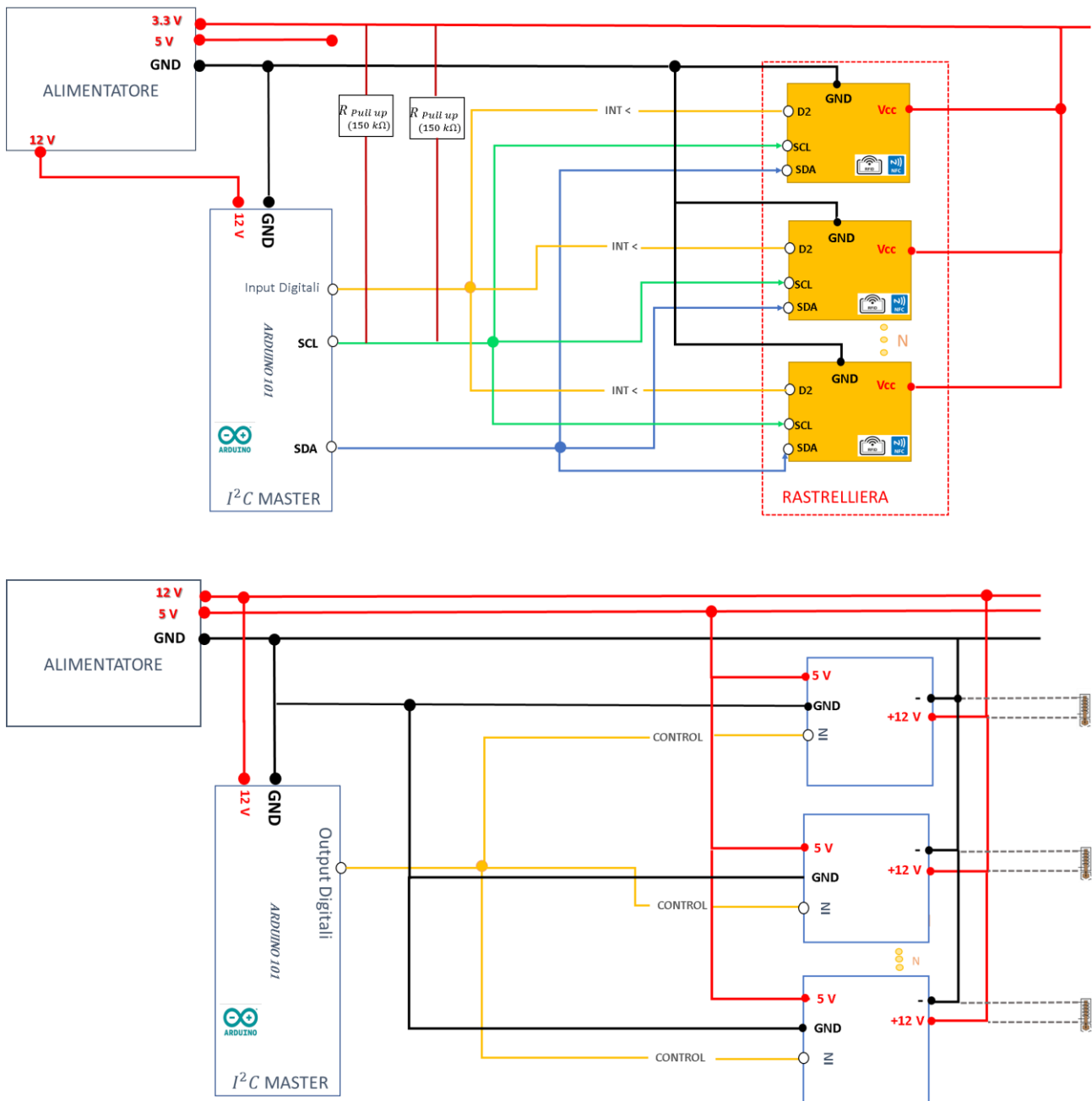


Figura 14 : Schema generale della Rastrelliera Intelligente-definizioni delle connessioni per la parte di raccolta dati proveniente dai sistemi RFID di ogni stallo(a) e comando degli attuatori a solenoide per ogni stallo (b)

Rastrelliera Intelligente - Connessioni fisiche tra piattaforma e sensori							
Analog IN	A0	A1	A2	A3	A4	A5	
I/O/INT Ports	0/RX	1/TX	2°	3*	4	5*	6*
			PN532-D2.OUT	Relè-IN			
I/O/INT Ports	7	8°	9*	10°	11°	12°	13°
Power	ATN/SS	IOREF	RESET	3,3	5	GND	Vin
						Ground	12V
Other	SCL	SDA	AREF	ICSP/MISO	ICSP/MOSI	ICSP/SCK	ICSP
	PN532(SCL)	PN532(SDA)					

Tabella 2: Rastrelliera Intelligente - connessioni fisiche tra motherboard e sensori

*: porte che supportano PWM

°: porte che supportano interrupt sui cambiamenti di stato

6.2.3 Software

La struttura di base del SW della Rastrelliera Intelligente corrisponde a quella comune a tutti gli attuatori integrati, descritta alla sezione 3.1.1.2.

6.3 APP per gestione rastrelliera, APP06

La APP con Procedura APP06 è descritta in altro documento di progetto, che al momento non è determinato. Il riferimento, attualmente “TBD”, verrà risolto nella revisione DE3.11b di questo documento a cura di DISIT.

6.4 Sensori / attuatori

Nel seguito una sezione è dedicata a ciascuno dei sensori e attuatori connessi alla Rastrelliera Intelligente.

6.4.1 Rilevatore RFID

6.4.1.1 Interfaccia fisica

Per quanto riguarda la scelta del lettore RFID la soluzione in prima analisi adottata verte verso una tecnologia di identificazione del TAG non di prossimità (< 4cm), ma di vicinanza (10 cm) in modo tale da non avere vincoli troppo restrittivi tra la distanza fisica tra il rilevatore RFID e il TAG posto sulla bicicletta (vedi Figura 15, in cui la distanza massima tra lettore RFID e TAG è di 65 mm).

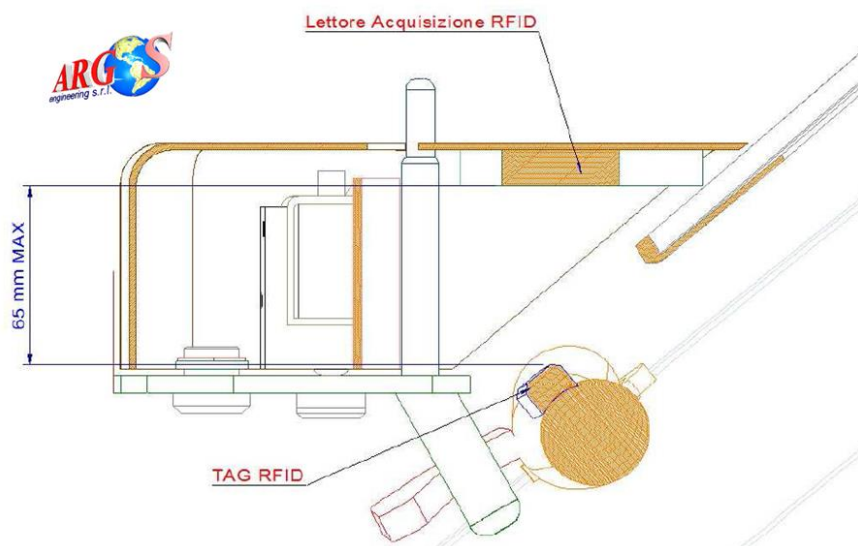


Figura 15 : Sistema locale di identificazione RFID installato sullo stallo della rastrelliera

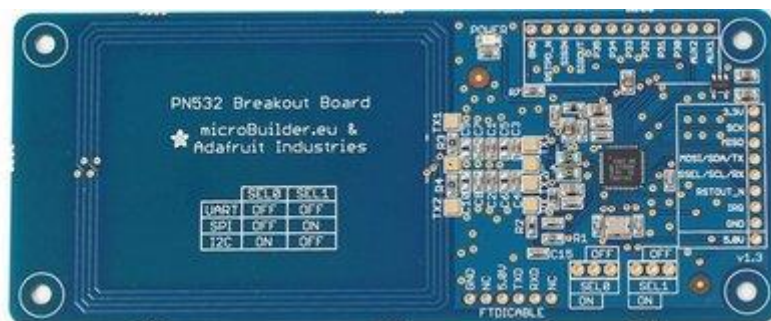


Figura 16 : Sistema locale di identificazione RFID - PN 532 RFID/ NFC (Arduino like)

Il sistema di identificazione RFID scelto è il PN 532 (Figura 16) le cui principali caratteristiche sono:

- Protocolli supportati: I2C(default)/SPI/UART;
- Tensione operativa: 3.3 V;
- Compatibile con Arduino Diecimila, Duemilanove, UNO;
- Adatto per letture NFC & TAG MIFARE;
- Frequenza Operativa: 13.56 MHz
- Range di Lettura Massimo: 10 cm
- Basato su modulo PN532
 - ISO/IEC 14443/MIFARE Reader/Writer
 - Felica Reader/Writer
 - ISO/IEC 14443/MIFARE Reader/Writer

Questo chip set è molto potente, e può praticamente fare tutto, come la lettura e la scrittura di tag e carte, comunicare con i telefoni (ad esempio per l'elaborazione dei pagamenti), ed 'agire' come un tag NFC. La frequenza operativa del modulo di lettura è stata scelta come compromesso tra range di lettura, costo e coefficiente di penetrazione del materiale (parametro interessante in quanto il sito di installazione ha materiali metallici nelle vicinanze).

Il protocollo di comunicazione scelto è I^2C che è preferibile per alcune sue caratteristiche:

- segue la logica *master/slave*, ideale in ottica della gestione del traffico tra la motherboard e più sensori RFID;
- permette di raggiungere distanze di qualche metro, a differenza del protocollo SPI;
- in ottica di avere un sistema flessibile alla struttura della rastrelliera, è possibile usare delle estensioni di bus I^2C che agevolano la comunicazione a distanze dell'ordine di decine di metri (vedi 6.4.1.1.4);
- il numero di pin di interfacciamento è minimizzato, anche lato motherboard: tale aspetto sarebbe stato molto critico facendo uso di comunicazione seriale UART, soprattutto con rastrelliere dotate di un numero di stalli maggiore di 4/5;
- la scheda breakout/shield scelta prevede l'uso di un piedino digitale che fa da notifica di avvenuta identificazione di un TAG RFID.

6.4.1.1.2 Interfaccia logica

Le linee fondamentali previste dal protocollo I^2C sono:

- SDA: Linea di scambio dati
- SCL: Linea di clock

La Figura 17 mette in luce le condizioni operative che permettono di definire la fase iniziale e finale della comunicazione:

- START: avviene per una transizione alto-basso di SDA quando SCL è alto;
- STOP: avviene con una transizione basso-alto di SDA quando SCL è basso.

Avvenuta la condizione di START, è possibile iniziare la comunicazione, per cui ogni byte trasmesso deve essere seguito dal messaggio di ACK: il clock per il segnale di ACK viene generato dal *master*, mentre lo *slave* genera il messaggio di ACK portando bassa la linea SDA per tutto il tempo in cui il CLOCK è a livello alto.

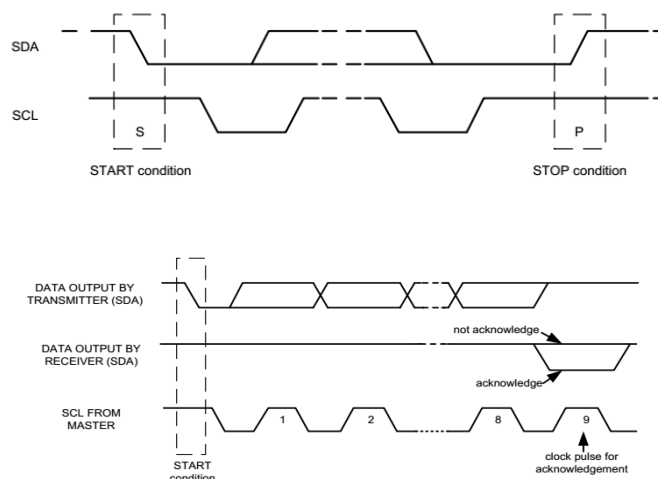


Figura 17 : definizione della fase di START/STOP, ACKNOWLEDGE sul bus I^2C

La comunicazione quindi si compone delle seguenti fasi:

- Trasmissione dal *master* di 7 bit di indirizzo dello *slave* + un ottavo bit (R/ \bar{W}) in cui si specifica se il *master* è in lettura o scrittura ;
- Il master attende il messaggio di ACK da slave che viene segnalato secondo le condizioni descritte in Figura 17;
- La comunicazione si interrompe una volta imposte le condizioni di *stop* descritte in Figura 18.

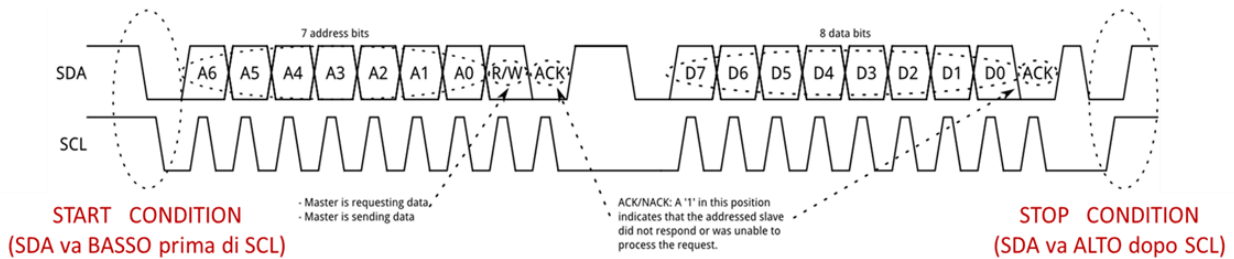


Figura 18 : comunicazione sul bus I²C (indirizzo slave + data)

6.4.1.1.3 Architettura

Di seguito (Figura 19) è riportata l'architettura generica di principio per quanto riguarda l'infrastruttura della rastrelliera, composta da una parte di identificazione delle bike tramite lettori RFID, le cui informazioni saranno trasmesse alla motherboard tramite un protocollo di comunicazione seriale I²C per motivi che sono stati discussi nel paragrafo 6.2.2.1.2.

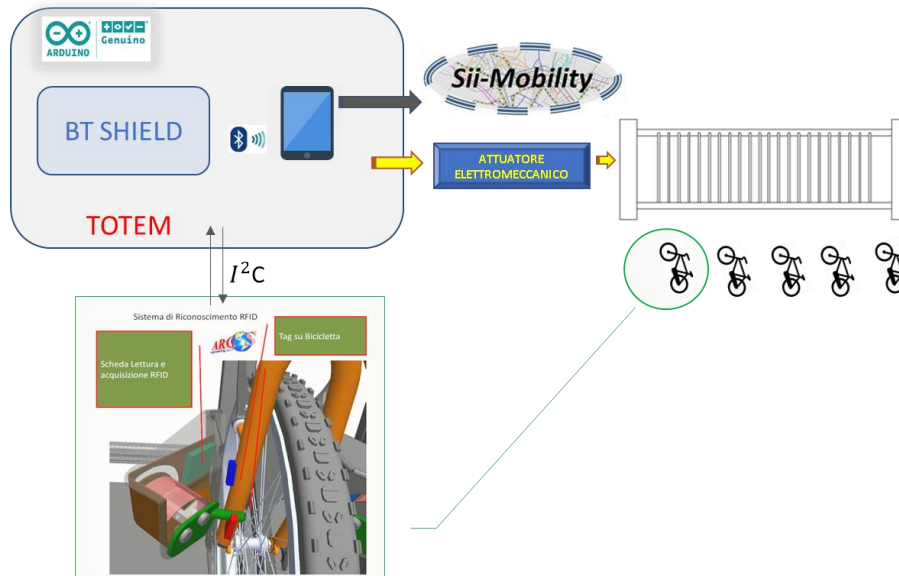


Figura 19 : Architettura generale di principio del sistema elettronico, caso operativo di rastrelliera con singolo stallo. La piattaforma TOTEM comunica con i lettori RFID tramite porta seriale I²C

L'I²C ha un'impostazione predefinita, in quanto utilizza un minor numero di pin: gli analogici 4 e 5 sono utilizzati per la I²C. Il pin digitale 2 è utilizzato per la notifica "interrupt", nel senso che il pin cambierà stato nel momento in cui viene identificato un TAG nel raggio di azione. Di seguito viene definita l'architettura di comunicazione tra motherboard e singolo stallo (Figura 20). La linea dati

SDA e la linea di clock SCL richiedono una resistenza di pull-up di 150 kΩ verso la linea di alimentazione.

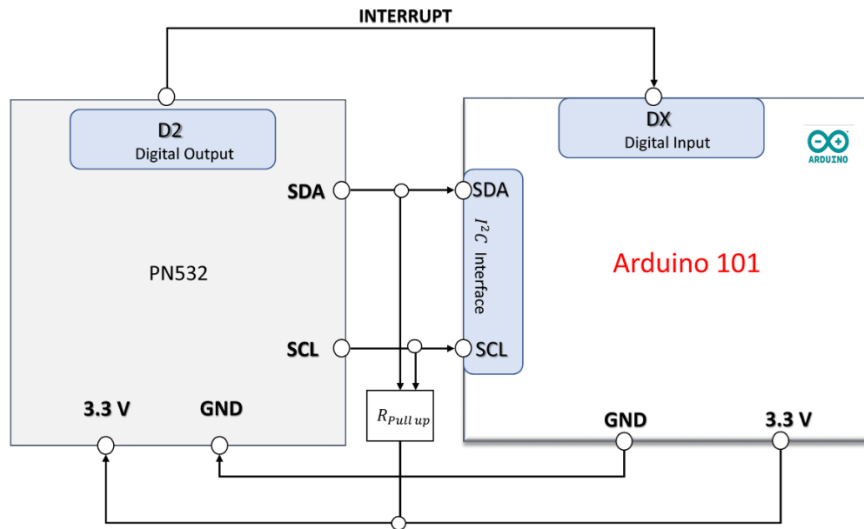


Figura 20 : Comunicazione tra RFID Reader e motherboard (da sviluppare per ogni stallo)

6.4.1.1.4 Specifica di dettaglio

In questo paragrafo di dettaglio si vuole riportare un possibile utilizzo di una estensione del bus dati, da adottare nel caso di rastrelliera di elevata estensione. In particolare, al fine di proporre una soluzione il più possibile flessibile a fronte di una rastrelliera composta da N sensori si può considerare l'applicazione di un'estensione del bus dati I^2C attraverso l'utilizzo del chip P82B715PN su entrambi i "front end" (Figura 21). La distanza massima di comunicazione su bus I^2C è limitata a qualche metro: attraverso l'utilizzo di questo chip è possibile aumentare la distanza fino a decine di metri.

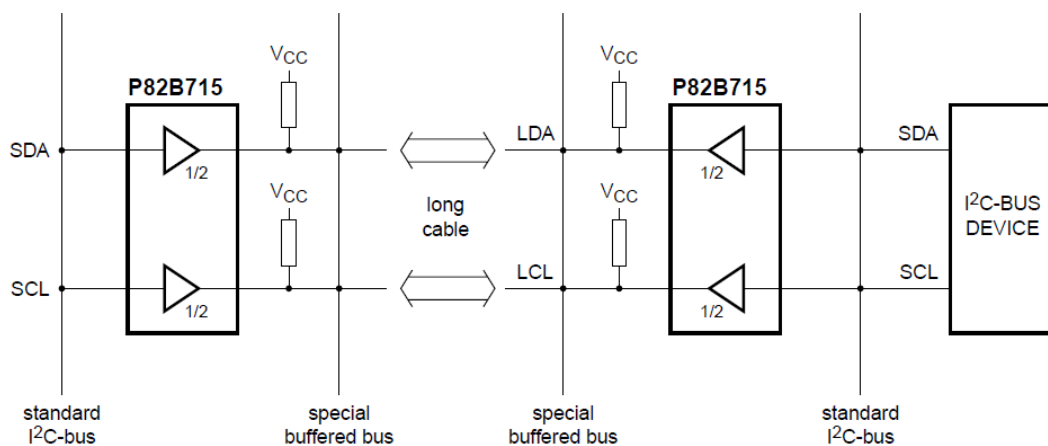


Figura 21 : CHIP P82B715 per estendere il bus I^2C

6.4.2 Vincolo Meccanico

6.4.2.1.1 Interfaccia fisica

Il sistema di Blocco proposto prevede un bloccaggio da un lato della bicicletta e viene attivato facendo ruotare il perno (blu) sull'esterno del supporto: quando la staffa (verde) arriva in posizione all'interno scatta un sistema a molla che blocca la bici sul lato su cui ho eseguito l'operazione (vedi Figura 22 Sx).

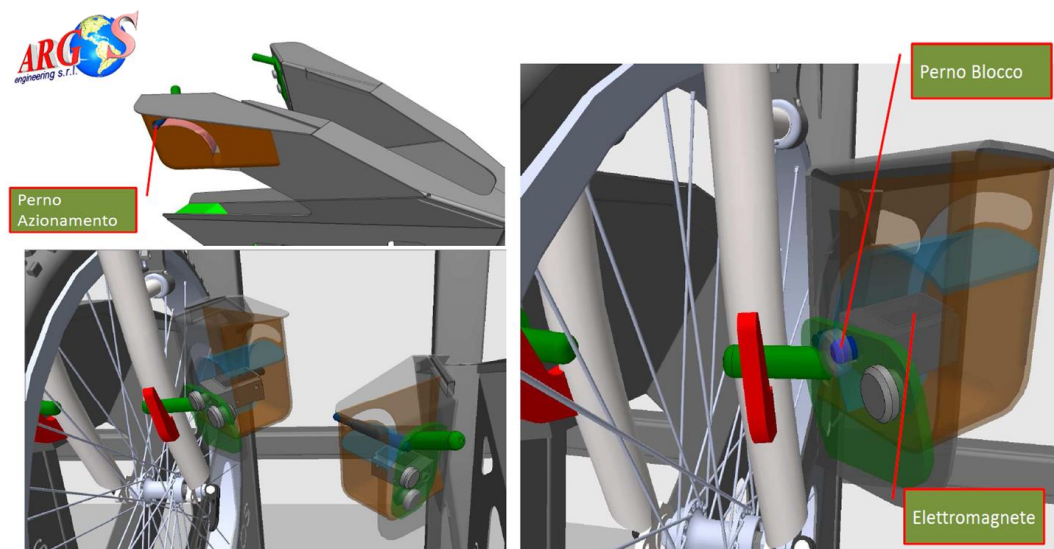


Figura 22 : perno azionamento con intervento esterno (Sx) e elettromagnete dotato di bobina che permette lo sblocco mediante applicazione di una tensione di 12 V (Dx)

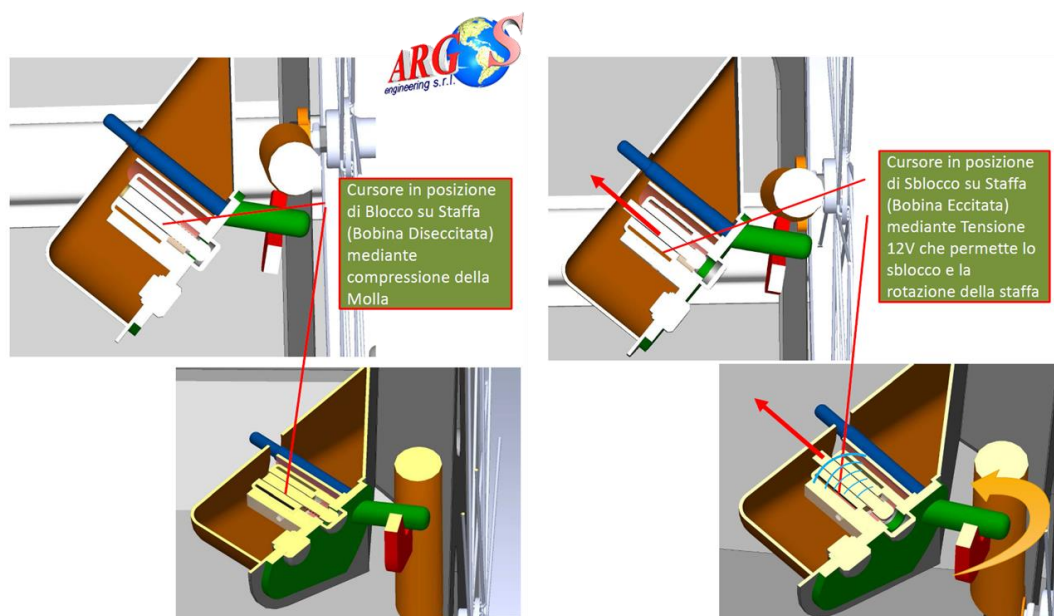


Figura 23 : cursore in posizione di blocco mediante compressione a molla (Sx) e cursore in posizione di sblocco mediante applicazione di eccitazione della bobina tramite una tensione di 12 V

Il perno azionato da una molla è però un cursore di un elettromagnete dotato di bobina elettrica che lavora a trazione e quindi risponde al comando di sblocco soltanto mediante attivazione di una tensione elettrica di fatto generata da una logica di utilizzo del sistema (vedi Figura 23 Dx).

L'attuatore tramite solenoide reagisce ad una applicazione della tensione 12 V e, di conseguenza, ai fini del suo controllo, è sufficiente implementare un relè comandato da un pin digitale del μC a bordo del totem (Figura 24).

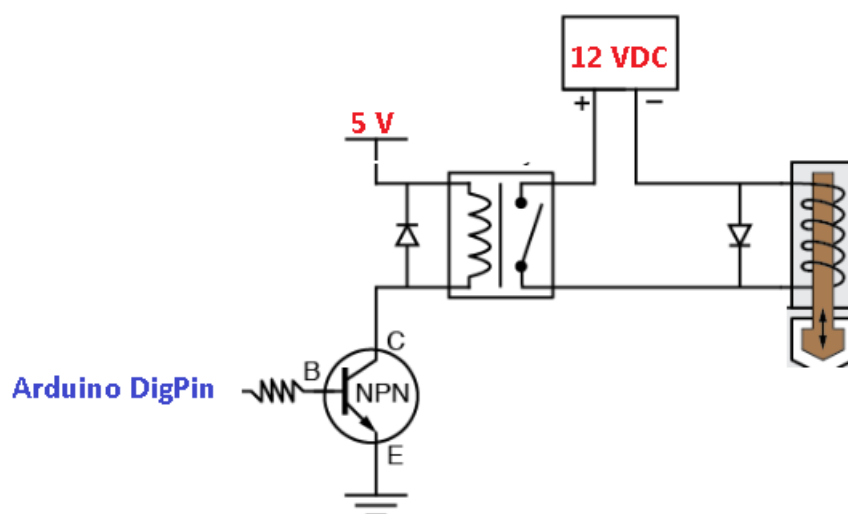


Figura 24 : schema di principio di controllo tramite pin digitale di Arduino del singolo solenoide responsabile del gancio/sgancio della bicicletta allo stallo

6.4.2.1.2 Interfaccia logica

Il relè selezionato per pilotare lo stato di eccitazione/diseccitazione del solenoide è compatibile con i livelli logici alto/basso dei pin digitali di Arduino 101 (LOW=0 V, HIGH: 5 V). Inoltre, è stato scelto di utilizzare un relè che è normalmente aperto NO (di conseguenza il gancio è chiuso) che tramite applicazione dei 12 V comporta l'apertura del gancio.

6.4.2.1.3 Architettura

Per quanto riguarda l'architettura fa fede la (Figura 14 b) alla sezione 6.2.2.1.2 e quanto descritto alla sezione 6.4.1.1.3 e in particolare la Figura 19.

6.4.2.1.4 Specifica di dettaglio



Figura 25 : relè dedito all'azionamento del perno usato per l'aggancio/sgancio della bicicletta alla rastrelliera

Una possibile relè di azionamento del solenoide è illustrato in Figura 25 e le sue principali caratteristiche sono:

- tensione della bobina: 5v dc
- bobina Resistenza: $70 \omega \sim 80 \omega$
- Resistenza di contatto: 100ω max
- tempo di funzionamento: 10msec max

- Tempo di rilascio: 5msec max
- Temperatura ambiente: -25 ° C a 70 ° c
- umidità in funzionamento: 45-85% ur
- corrente di funzionamento: 43mA ~ 46ma
- versione corrente: 15mA ~ 18Ma
- pin: 5pin (permette la configurazione NO (*normal open*), NC (*normal close*))
- resistenza di isolamento: ≥ 100 (ohm)
- rigidità dielettrica:
 - tra bobina & contatti: ac 1500V 50hz ~ 60hz / min
 - tra contatti: ac 1000v 50hz ~ 60hz / min
- carico nominale: 10A 250V ac / 10a 125v ac / dc 10a 30v / 10a 28V dc

Il pcb ospitante il relè e anche la logica di controllo descritta in Figura Y è il “5V 1 Channel Relay Module Shield for Arduino ARM PIC AVR DSP SRD-05VDC-SL-C” che ha tutti i componenti descritti in Figura 26.



Figura 26 : PCB ospitante il relè dedito all’azionamento del perno usato per l’aggancio/sgancio della bicicletta alla rastrelliera

Un possibile attuatore lineare pensato per tale applicazione è il seguente, in Figura 27:



Figura 27 : Solenoide a D in c.c. A spinta BLP 42-120-621-620, corsa massima 27mm, tensione 12 V c.c., 10W