Scuola universitaria professionale della Svizzera italiana
Dipartimento tecnologie innovative

SUPSI

Interfacciamento semi-automatico di una piattaforma microfluidica

Studente/i

Lietti Carlo

Relatore

Tavilla Agatino Christian

Correlatore

Stefanini Igor

Committente

//

Corso di laurea

Ingegneria Elettronica

Modulo

C10081

Anno

2018/2019

Data

10 settembre 2019

STUDENTSUPSI

Indice

1 Abstract **1**

2 Introduzione **3**

3 Cenni teorici **5**

3.1 Cenni storici 5

3.2 Materiali 6

3.2.1 Inorganici 7

3.2.1.1 Silicone 7

3.2.1.2 Vetro 7

3.2.1.3 Ceramici 7

3.2.2 Elastomeri 8

3.2.2.1 PDMS 8

3.2.2.2 TPE 8

3.2.3 Termoplastici 9

3.2.3.1 PMMA 9

3.2.4 Carta 9

3.3 I Lab-On-Chip 10

3.4 Peptidi autoassemblanti 10

4 Il progetto **11**

4.0.1 Alimentazione 13

5 Sistemi d'accesso **15**

5.1 Micro pump 15

5.1.1 Funzionamento Micropump 16

5.1.2 Alimentazione 18

5.2 Syringe pump 18

5.2.1 Funzionamento syringe pump 19

5.2.2 Alimentazione Syringe pump 20

STUDENTSUPSI

6 Sensore di pressione **21**
6.1 Soluzioni possibili . . . 21
6.1.1 LoC personalizzato . . . 21
6.1.2 Sensori intermedi . . . 22
6.2 Sensore di pressione e flusso . . . 23
6.2.1 Tensione di alimentazione . . . 23
6.2.2 Range di pressione . . . 23
6.2.3 Misura di liquidi . . . 24
6.3 MP3V5050DP . . . 25
6.3.0.1 Caratteristiche tecniche . . . 25
6.3.1 Bernoulli e Venturi . . . 27

7 Elettronica e Controllo **31**
7.1 Struttura . . . 31
7.2 Board di alimentazione . . . 32
7.2.1 Schematici . . . 32
7.2.1.1 Foglio 1 BdA . . . 36
7.2.1.1.1 Funzionamento . . . 36
7.2.1.2 Foglio 2 BdA . . . 37
7.2.1.3 Foglio 3 BdA . . . 38
7.2.2 PCB BcA . . . 39
7.2.3 HV9150 . . . 43
7.2.4 Funzionamento e Dimensionamento HV9150 . . . 46
7.2.5 HV513 . . . 50
7.2.6 MP6500GF . . . 52
7.2.7 Funzionamento e dimensionamento MP6500 . . . 53
7.2.7.1 Controllo di passo . . . 53
7.2.7.2 Controllo di corrente . . . 54
7.2.7.3 Segnalazione di errori . . . 56
7.2.8 AP1609 . . . 56
7.2.9 Dimensionamento AP1609 . . . 58
7.3 Board di controllo . . . 60
7.3.1 schematici . . . 60
7.3.1.1 Foglio 1 BdC . . . 64
7.3.1.2 Foglio 2 BdC . . . 64
7.3.1.3 Foglio 3 BdC . . . 64
7.3.2 PCB BdC . . . 65
7.3.3 STM32F407VG . . . 69
7.3.3.1 Programmazione . . . 70

STUDENTSUPSI

7.3.4 AP7173 70
7.3.5 Dimensionamento AP7173 71
7.3.5.1 Tensione d'uscita 71
7.3.5.2 Condensatori di input/output 73
7.3.5.3 Soft-start 73
7.3.6 LM239A-EP 74

8 Interfaccia utente 75
8.1 Il display 75
8.1.1 Tensione di lavoro 75
8.1.2 Tipo di interfaccia 75
8.1.3 Il prezzo 76
8.1.4 ST7565R 77
8.1.4.1 Connettività 78
8.1.4.2 Seriale 79
8.1.4.3 Parallelo 80
8.1.5 Comandi 80
8.1.6 Input esterni 81

9 Software 83
9.1 Diagramma a blocchi 83
9.2 CubeMX 84
9.3 Il programma 85

10 Spesa economica 91

11 Miglioramenti futuri 93

12 Conclusioni 95

13 Bibliografia 97

14 Ringraziamenti 99

STUDENTSUPSI

STUDENTSUPSI

Elenco delle figure

3.1 LoCs 6

4.1 Diagramma a blocchi soluzione 11
4.2 Scatola di plexiglas 12
4.3 Dispositivi per l'alimentazione generale 13

5.1 MIcropompa 16
5.2 Micro pump 17
5.3 Flusso-Pressione 17
5.4 Syringe pump 18
5.5 Funzionamento syringe pump 19
5.6 Syring pump progetto 20

6.1 Chip microfluidico personalizzato 21
6.2 Diagramma di interfacciamento multipli sensori 22
6.3 Modello sistema con sensore e blocco di protezione 23
6.4 Sensore di pressione NXP 25
6.5 Schema interno sensore 26
6.6 Grafico errore di pressione 27
6.7 Gafico errore di temperatura 27
6.8 A sinistra è mostrato lo schema elettrico del sensore, mentre a destra i componenti di cui ha bisogno per una lettura corretta 28
6.9 Tubo in cui variano sezione, altezza, pressione e flusso 28
6.10 Tubo di venturi 29

7.1 Board di Alimentazione e controllo 31
7.2 Foglio n.1 board alimentazione 33
7.3 Foglio n.2 board alimentazione 34
7.4 Foglio n.3 board di alimentazione 35
7.5 Circuito di Boost 36
7.6 Differenti tipologie di DC/DC boost 36

STUDENTSUPSI

7.7 Pcb board di alimentazione, senza piani di alimentazione 39
7.8 Pcb board di alimentazione, TOP layer 40
7.9 Pcb board di alimentazione, BOTTOM layer 40
7.10 Pcb board di alimentazione, 3d 41
7.11 Connettori board di alimentazione 41
7.12 Circuiti di alimentazione micropompe 42
7.13 Driver motori Syringe pump 43
7.14 Connettore per board di controllo 43
7.15 Hv9150 sch 44
7.16 Voltage reference 46
7.17 Grafico "Rfreq-Fgate" 48
7.18 HV513 sch 50
7.19 MP6500 schema interno 53
7.20 Descrizione funzionameto MOSFET 54
7.21 Comportamento I_{OUT} in funzione di $ITRIP$ 55
7.22 Schema elettrico interno AP1609 57
7.23 Schema elettrico AP1609 57
7.24 Interfacciamento AP1609 59
7.25 circuito AP1609 per alte tensioni 59
7.26 Foglio n.1 board controllo 61
7.27 Foglio n.2 board controllo 62
7.28 Foglio n.3 board controllo 63
7.29 Differenza tra due segnali prodotti da un pulsante, con o senza anti-rimbalzo . 65
7.30 Pcb Board di controllo 65
7.31 Pcb Board di controllo, TOP layer 66
7.32 Pcb Board di controllo, BOTTOM layer 67
7.33 Pcb Board di controllo, vista 3d 67
7.34 Alimentazione board di controllo 68
7.35 Connettori per programmazione e pulsanti a sinistra, connettore per il display a destra 69
7.36 Connessione SWD 71
7.37 Schema interno LDO 72
7.38 Applicazione tipica AP7173 72
7.39 Grafico Variazione di Vout a seconda di Css 73
7.40 Comparatore utilizzato 74

8.1 Display LCD 76
8.2 Le due immagini mostrano i due possibili sistemi di interfacciamento del display 77
8.3 Diagramma interno ST7565R 78

STUDENTSUPSI

8.4 a sinistra è mostrato il collegamento per il protocollo SPI half-duplex, mentre a destra è mostrato il diagramma dei tempi per la comunicazione tra MPU e diplay . . . 79
8.5 Piedini da impostare in base all'operazione scelta . . . 80
8.6 Bottoni utilizzati . . . 81

9.1 Flow chart porgramma . . . 83
9.2 Impostazione pinout tramite CubeMX . . . 84
9.3 Interfaccia controllo dei clock CubeMX . . . 85
9.4 Variabili create ed inizializzate . . . 85
9.5 Inizializzazione sistema . . . 86
9.6 funzioni di inizializzazione programma . . . 86
9.7 Prima fase del programma . . . 87
9.8 Funzionamento syringe pump . . . 87
9.9 Funzioni di start e stop per la syringe pump . . . 87
9.10 Controllo micropompa . . . 88
9.11 Funzionamento comando micropompa . . . 88
9.12 Comportamento pulsanti . . . 89
9.13 Funzioni per il passaggio di dati o comandi al display . . . 89

STUDENTSUPSI

STUDENTSUPSI

Elenco delle tabelle

STUDENTSUPSI

STUDENTSUPSI

Capitolo 1

Abstract

Le zanzare sono insetti che vivono nella maggior parte del pianeta. Secondo uno studio sono l'animale che causa il maggior numero di morti l'anno. Studiare le malattie che trasportano è quindi di fondamentale importanza per l'uomo. Purtroppo il processo attualmente utilizzato ad individuare quale virus viene trasportato è molto dispendioso e produce pochissimo materiale da poter analizzare, per questa ragione è necessario trovare una soluzione alternativa per l'individuazione delle malattie trasportate da questi insetti. Questa tesi mira a creare un sistema che permetta a scienziati e ricercatori di interfacciarsi facilmente con un circuito microfluidico, al fine di individuare più velocemente e facilmente il virus trasportato da una zanzara. Per farlo si è voluta realizzare una struttura che permettesse di introdurre all'interno di un chip microfluidico il cDNA sintetizzato dalla saliva di zanzare, tramite diversi sistemi d'accesso. Questa struttura è in grado di monitorare flusso e pressione del liquido introdotto, garantendo costantemente un controllo su di essi. Il progetto realizzato è il primo passo, per cercare una tecnica più efficacie ed efficiente di individuare i virus che vengono trasportati dalle zanzare, con la speranza che questo possa contribuire un giorno a debellarli.

Mosquitoes are insects that live all over the world. A study affirms that they are the major cause of death in the world and therefore studying the diseases that mosquitoes transport is extremely important for human beings. Unfortunately, the process that identifies which virus is transported by mosquitoes is very expensive and produces very little quantity of substance to be analyzed. For this reason, it is necessary to find an alternative way to detect the viruses transported by these insects. This document is about the creation of a system that allows scientists and researchers to better ideintify which virus or viruses are transported by a mosquito. In order to accomplish this goal, it has been created a structure that allows to introduce in a microfluic chip the synthetize mosquitos' cDNA, using different access

STUDENTSUPSI

systems. Such structure is able to monitorize the flux and pression of the introduced fluid, ensuring constant control on them. This project is the first step to find an alternative and also more efficient way to detect virus transported by mosquitos, with the hope that one day, there will be the knowledge to eliminate them all.

STUDENTSUPSI

Capitolo 2

Introduzione

Questa tesi nasce dalla teoria che sia possibile distinguere il virus trasportato da una zanzara, tramite la variazione d'impedenza che del cDNA[1] ricavato dalla saliva di questo animale, produce su una membrana Peptidica, oppure tramite la fluorescenza che scaturisce dall'impatto tra il cDNA ed la membrana stessa. Per dimostrare questa teoria è necessario disporre di un'attrezzatura adatta per il controllo e la gestione del cDNA (che si presenta sotto forma di fluido). In questo documento sono quindi descritti tutti i sistemi utilizzati per creare una piattaforma che permetta a scienziati e ricercatori di gestire il cDNA delle zanzare nel modo più immediato possibile, ammettendo diverse possibilità di accesso ad un chip microfluidico.
Gli obiettivi del progetto sono:

- Creare un sistema che consenta di interfacciarsi in modo semplice tramite una siringa o una provetta, un chip microfluidico.
- Mostrare un feedback all'utilizzatore sulla velocità e pressione del liquido.
- Garantire un flusso di almeno 5μL/minuto
- Creare una struttura che sia in grado di contenere tutta l'attrezzatura utilizzata.

[1] Si chiama DNA complementare la molecola di DNA a filamento singolo che viene prodotta su uno stampo di RNA a opera dell'enzima trascrittasi inversa (ovvero un enzima appartenente alla classe delle transferasi).

STUDENTSUPSI

STUDENTSUPSI

Capitolo 3

Cenni teorici

Con il termine microfluidica si indica la scienza e tecnologia dei sistemi che processano o manipolano piccole quantità di liquidi (10^{-6} a 10^{-18} litri), utilizzando canali di dimensioni di decine o centinaia di micrometri. Questo tipo di approccio è sempre più affermato nel mondo della ricerca, in particolare in campi quali: la Medicina, la Chimica o le Biotecnologie, per le quali la Microfluidica offre una serie di utilissime possibilità in quanto:

- Può funzionare con bassissime concentrazioni di reagenti
- Garantisce un'altissima risoluzione e sensibilità
- Permette la separazione di reagenti con molta accuratezza
- Occupa uno spazio molto ridotto

3.1 Cenni storici

La Microfluidica nasce intorno agli anni 50, quando sono stati sviluppati dei sistemi per erogare bassissime quantità di liquido (micro o nano Litri), questi sistemi hanno preso poi il nome di tecnologie Inkjet, sostanzialmente essi sono sistemi ultracompatti per l'erogazione di fluidi, il loro funzionamento si basa sulla generazione molto rapida di calore tramite impulsi elettrici su di una resistenza, la quale, a contatto con il liquido lo espande spingendolo all'esterno. Successivamente, tra gli anni 80 e 90, sono stati creati i primi esempi di microvalvole e micropompe. Grazie a tali invenzioni i ricercatori hanno pensato di realizzare dei sistemi miniaturizzati per analisi chimiche e biologiche, giungendo infine ai cosìddetti "Lab-On-Chips" (LoCs). Questi sistemi di cui è mostrato un esempio in **??**, uniti ad altre interessanti scoperte fatte negli anni 80, come ad esempio il sequenzionamento del DNA(determinazione dei diversi componenti costitutivi di un DNA), hanno permesso un'enorme avanzamento delle conoscenze in campo biochimico, poiché garantiscono sensibilità e risoluzione che hai tempi non si sarebbero ottenuti con nessun altro metodo. Inoltre, tutte

STUDENTSUPSI

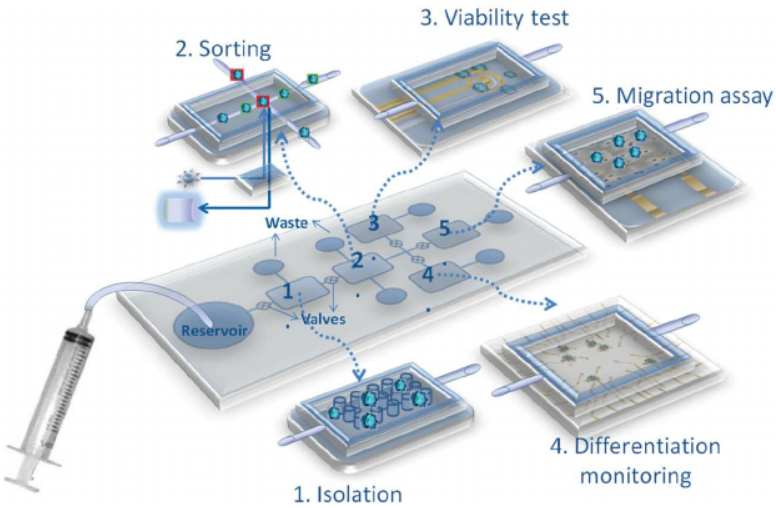


Figura 3.1: Lab-On-Chips schema

le analisi standard di questo genere richiedono costi altissimi e molto personale, i quali con i LoCs sono stati notevolmente ridotti. Nel corso degli anni 90 la ricerca è progredita ed ha permesso di rendere i LoCs, non solo sistemi molto più efficienti, ma ne ha aumentato ulteriormente le potenzialità rendendoli, di fatto, indispensabili per ricerche in ambito biochimico. Oggi i sistemi microfluidici sviluppati sono moltissimi e le pubblicazioni a proposito sono in continua crescita.

3.2 Materiali

A seconda dell'applicazione (LoC, detezione di patogeni, analisi DNA, etc.), la struttura microfluidica varia e deve essere adattata. Nello stesso modo è necessario adattare il materiale con il quale si costruisce suddetta struttura. I primi materiali ad essere usati sono stati Silicone e vetro, ma con l'avanzare delle tecnologie sono emersi nuovi materiali che si dividono in 3 diverse categorie:

- Inorganici
- Polimerici
- Carta

a loro volta i chip a base di polimeri possono essere suddivisi in 3 sotto categorie:

- Elastomeri
- Termoplastici
- Termoindurenti

Elastomeri e Termoplastici sono i più utilizzati per la produzione di chip microfluidici.

STUDENTSUPSI

3.2.1 Inorganici

3.2.1.1 Silicone

Il Silicone è stato il primo materiale utilizzato per i circuiti microfluidici, per poi essere quasi totalmente sostituito da vetro ed altri polimeri.
Fu scelto il Silicone per la sua alta conducibilità termica e stabilità chimica. La fabbricazione del Silicone utilizza sia dei sistemi sottrattivi come il "dry etching"[2], sia dei sistemi additivi ad esempio la deposizione tramite vapore. Il materiale è piuttosto elastico, ciò rende difficile realizzare componenti microfluidici, come ad esempio pompe o valvole e i processi produttivi per questo esso motivo richiedono l'utilizzo di dannosi prodotti chimici. Inoltre, esso ha un costo piuttosto alto rispetto agli altri materiali usati nel settore. Tutte queste caratteristiche rendono il Silicone il materiale meno consigliato per la produzione di chip microfluidici oggi, ma malgrado ciò, viene ancora utilizzato per la produzione di nanocavi, analisi di tessuti cardiaci ed altro. Nonostante ciò esso è ancora utilizzato per la produzione di nanocavi, analisi di tessuti cardiaci ed altro.

3.2.1.2 Vetro

Successivamente, la creazione di microcanali in vetro già conosciuta e utilizzata in altri ambiti come ad esempio nella "elettroforesi"[3], ha portato a preferire l'utilizzo del vetro al silicone. Per produrre il vetro si usano tecniche sottrattive come il "dry etching". Esso è un materiale trasparente, ed isolante elettricamente, inoltre e impermeabile ai gas. Grazie alla sua termo-conduttività e stabilità molecolare i canali di vetro consentono migliori performance rispetto ad altri materiali. Esso è utilizzato principalmente per L'"elettroforesi capillare" ovvero un processo chimico volto alla separazione di cariche ioniche in un determinato fluido. Sue altre applicazioni sono: estrazione di solventi, reazioni su chip, etc. I costi di produzione piuttosto elevati, a causa delle pesanti condizioni con cui il vetro deve essere lavorato per essere modellato correttamente, come ad esempio alte temperatura o alta pressione, e la durezza del materiale, sono i motivi per i quali si sono cercati materiali alternativi per l'ambito microfluidico.

3.2.1.3 Ceramici

Questi tipi di chip si basano sulla LTCC(Co-fired ceramic) ovvero una ceramica fabbricata a strati di ceramica mista a polimeri, che permette la creazione nel chip ceramico, non solo di microcanali, ma anche di sistemi ibridi, e componenti elettronici, come ad esempio resistenze, induttori, condensatori, etc.

[2] Il dry etching consiste nel rimuovere del materiale bombardandolo con degli ioni he possono essere di diversi materiali, uno tra tutti l'Ossigeno

[3] L'elettroforesi è una tecnica analitica e separativa basata sul movimento di particelle elettricamente cariche immerse in un fluido per effetto di un campo elettrico applicato mediante una coppia di elettrodi al fluido stesso.

STUDENTSUPSI

La ceramica viene utilizzata sia in ambito di alta precisione, quindi basso volume ed alti costi, sia in ambito opposto ovvero grandi volumi, ma prezzi ridotti; inoltre questo tipo di materiale ha delle eccellenti proprietà elettriche e meccaniche e consente la creazione sul medesimo chip di sistemi elettrici, ottici e fluidici.
I vantaggi principali dei sistemi ceramici sono il costo basso ed il tempo di produzione rapidi rispetto agli altri materiali della categoria.

3.2.2 Elastomeri

Consistono in un reticolato ordinato di polimeri, che normalmente sarebbe aggrovigliato. Essi possono torcersi e schiacciarsi se sottoposti a forze esterne , per poi ritornare alla forma iniziale una volta che la forza cessa. La loro struttura molecolare non è resistente quanto quella delle controparti inorganiche il che li rende leggermente meno affidabili rispetto ad altri tipi di materiali.

3.2.2.1 PDMS

Il PDMS (Poly-di-Methyl-Siloxane) è un polimero largamente usato per la fabbricazione ed il prototipaggio di chip microfluidici. É il materiale più comune usato per la microfluidica, grazie ai suoi rapidi tempi di fabbricazione e basso costo. I canali sono creati tramite fotolitografia . Inoltre è possibile creare complessi circuiti sovrapponendo più strutture semplici di PDMS.
Per quanto riguarda le sue proprietà fisiche, questo materiale ha una bassa elasticità rispetto agli altri della sua categoria, che lo rende ottimo per la creazione di microvalvole o pompe, è permeabile ai gas e ciò lo rende utile per il trasporto di ossigeno e anidride carbonica in ambiti come lo studio cellulare, anche se, in altri casi questa caratteristica può favorire la formazione di bolle d'aria all'interno del materiale che potrebbero risultare problematiche nell'applicazione. Purtroppo, questo polimero ha la tendenza ad ingrossarsi quando in contatto con molecole idrofobiche, il che rappresenta il suo maggiore limite, perché ingrossandosi non garantisce il giusto flusso del fluidi al suo interno.

3.2.2.2 TPE

IL TPE(Thermoset Polyester) Questo polimero è insolubile ed ha un'alta resistenza alla rottura, è facile e rapido da fabbricare ed ha il vantaggio di essere trasparente. Inoltre, contrariamente al PDMS, questo materiale non si ingrossa con la maggior parte dei solventi, pur rimanendo permeabile ai gas.
La sua alta resistenza è dovuta alla sua struttura formata dall'incrocio di multiple strutture polimeriche, le quali vengono successivamente scaldate per rendere la struttura ancora più resistente.
Nonostante le sue caratteristiche questo polimero non è molto usata in ambito microfluidico,

STUDENTSUPSI

a causa della sua rigidezza che rende complessa la creazione di valvole e pompe con questo materiale.

3.2.3 Termoplastici

La caratteristica principale di questi polimeri è he essi possono essere rimodellati più volte portandoli alla temperatura di transizione del polimero, ovvero quando passa dal sul stato solido ad uno stato duttile e viscoso.
Questi polimeri sono trasparenti, impermeabili e più duri degli elastomeri, ciò li rende inadatti per colture cellulari e per la creazione di microvalvole o pompe. Il processo di fabbicazione è la termoformazione che tramite una specie di maschera in metallo (per via delle alte temperature), consente la creazione di migliaia di repliche ad alta velocità e basso costo, ciò rende questi polimeri eccellenti per la produzione industriale, ma non per il prototipaggio.

3.2.3.1 PMMA

Il PMMA "Poly-methyl methacrylate" è un polimero molto economico ed è il meno idrofobico tra i comuni materiali plastici. Esso è molto usato per via della sua rigidezza meccanica, trasparenza, compatibilità con l'elettroforesi e prezzo. Inoltre esso è facilmente lavorabile e modificabile, grazie a questa caratteristica esso può essere riutilizzato per la fabbricazione di chip più volte.
Per produrre dei sistemi PMMA si usa principalmente una tecnica chiamata "CO2-laser micromachinig", che consiste nell'intaglio tramite lasers di forme all'interno del materiale, ciononostante si possono usare anche altre tecniche, come ad esempio la "stampa tramite solventi".
Questo materiale viene usato per: chip di analisi, sequenziatori di DNA ed altro.

3.2.4 Carta

La carta è un materiale flessibile basato sulla cellulosa, recentemente è diventato un materiale molto promettente per la produzione di circuiti microfluidici per varie ragioni:

- La sua struttura chimica è facilmente modificabile.
- É facilmente reperibile in tutto il mondo
- Si può smaltire molto semplicemente, senza dover ricorrere a processi chimici particolari
- Il suo colore bianco offre un'ottimo contrasto con qualsiasi fluido colorato diversamente.

Purtroppo, questo materiale può essere usato solo per una quantità limitata di applicazioni per via della sua bassa resistenza meccanica e per la scarsa conoscenza che si ha del

STUDENTSUPSI

materiale in questo ambito. La carta può essere usata per analisi biochimiche, mediche e molecolari. Per far scorrere i fluidi all'interno del materiale ci si affida al suo effetto capillare, il quale permette un flusso sufficiente per le analisi compiute.

3.3 I Lab-On-Chip

Queste strutture hanno avuto un'enorme impatto nella comunità scientifica, ciò è dovuto al fatto che in un singolo LoC possono avvenire migliaia di reazioni al secondo. Inoltre in base alla loro struttura è possibile osservare tutto il processo chimico di una sostanza, senza dover ricorrere a sistemi esterni alla struttura, che potrebbero contaminare la sostanza e rendere il risultato meno affidabile. Per questo progetto è stata prevista l'introduzione di un LoC, esso avrà il compito di incanalare il DNA estratto dalla saliva di zanzare e farlo scorrere all'interno di una membrana di Peptidi, la quale unendo la sua struttura chimica al DNA usato, dovrebbe, in teoria, variare la propria impedenza, fornendo all'osservatore un'idea più o meno precisa della malattia trasportata dal DNA stesso.

3.4 Peptidi autoassemblanti

Un peptide è un segmento di proteina, una catena di aminoacidi uniti tra loro da legami peptidici [4]. Una delle più interessanti classi di biomateriali è costituita dai peptidi autoassemblanti, essi sono in grado di produrre delle particolari strutture bi o tri-dimensionali. Questi peptidi sono diventati molto famosi nell'ultimo periodo per la loro applicazione nel campo medico. Grazie alla loro struttura, infatti, essi consentono di riparare dei tessuti organici, perciò forniscono un'alternativa a vari interventi chirurgici molto invasivi, come ad esempio un trapianto.
Una membrana peptidica è un reticolato di formato dalle diverse connessioni che formano il materiale. Nella tesi che si vorrà dimostrare partendo da questo progetto, si suppone che, grazie alla loro struttura a forma di rete, le membrane peptidiche riescano ad intrappolare la soluzione che gli viene fatta scorrere attraverso e varino di conseguenza le loro caratteristiche elettriche o chimiche.

[4] Il legame peptidico o (giunto peptidico) è il legame chimico responsabile dell'unione degli amminoacidi e della conseguente formazione di peptidi e proteine. Si tratta di un legame di tipo covalente.

STUDENTSUPSI

Capitolo 4

Il progetto

Il progetto ha come scopo principale quello di riuscire ad interfacciare un circuito microfluidico con dei sistemi d'accesso esterni, controllando le soluzioni iniettate e fornendo un feedback all'operatore che utilizza il sistema.
In figura 4.1 è mostrato un diagramma a blocchi del sistema che si è voluto realizzare. Il

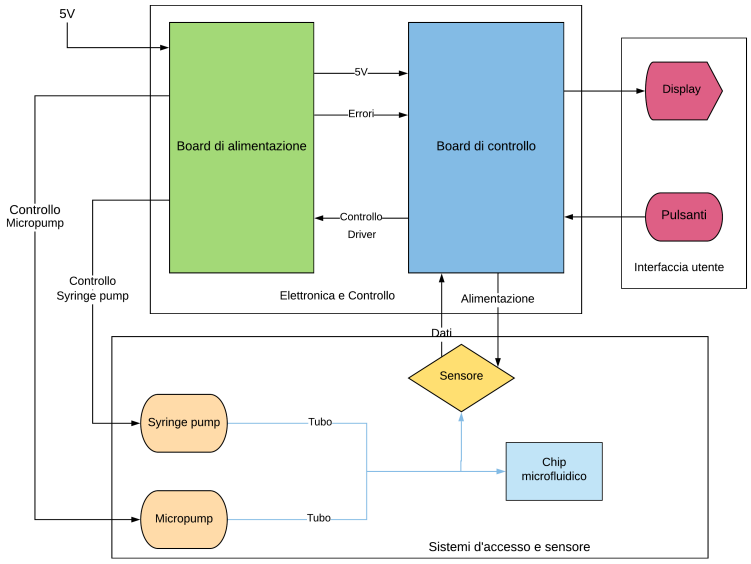


Figura 4.1: Diagramma a blocchi soluzione

sistema è diviso in 3 sezioni principali:

- **Sistemi d'accesso e sensore** Questa sezione consente all'operatore fornire il fluido al sistema, tramite due sistemi differenti: "Syringe pump" e "micro pump". una volta che il fluido viene pompato all'interno del circuito microfluidico, può essere controllato tramite i dati forniti dal sensore

STUDENTSUPSI

- **Elettronica e Controllo** É la sezione principale del progetto, al suo interno è contenuta tutta l'elettronica. Per comodità è stata divisa in due board differenti: "Board di alimentazione" e "Board di controllo".
- **Interfaccia utente** Consente di fornire un feedback all'operatore, su ciò che sta accadendo nel sistema e consente ad esso di interagirci tramite dei pulsanti.

Una volta assemblato tutti i componenti sono stati posti all'interno di una scatola in plexiglas, visibile in figura4.2

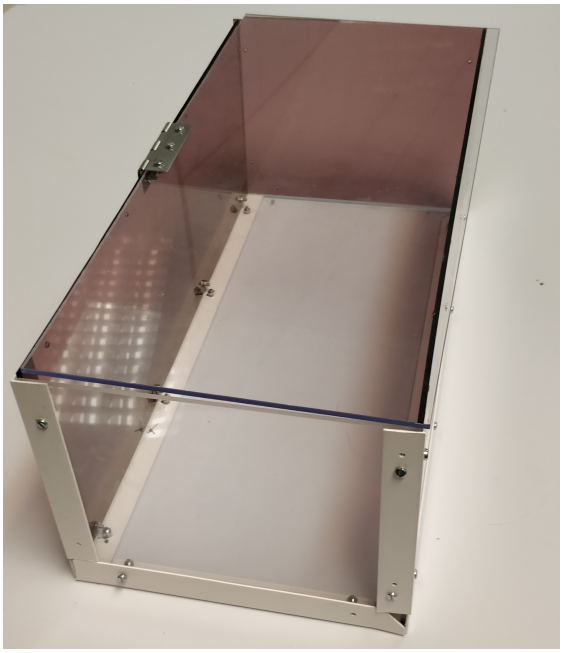


Figura 4.2: Scatola di plexiglas

STUDENTSUPSI

4.0.1 Alimentazione

Per alimentare tutti e tre i blocchi viene utilizzato l'**IRM-60-5ST**, questo componente, visibile in figura 4.3a è un AC/DC converter esso converte la tensione di rete, ovvero 230V in 5V. IRM-60-5ST, può erogare una potenza di 50W, viene connesso da un lato alla board di alimentazione, mentre dall'altro ad un adattatore per la tensione di rete, precisamente il **BVA01/Z0000/02** visibile in figura 4.3b.

(a) Convertitore AC/DC, IRM-60-5ST

(b) L'immagine mostra l'adattatore per per la tensione di rete.

Figura 4.3: Dispositivi per l'alimentazione generale

STUDENTSUPSI

STUDENTSUPSI

Capitolo 5

Sistemi d'accesso

Per questo progetto sono stati pensati due diversi sistemi per fornire del DNA al circuito microfluidico. Il primo consiste nel posizionare direttamente una provetta, all'ingresso del chip microfluidico, il cui contenuto sarà poi pompato all'interno del circuito stesso tramite una "Micro pump". Il secondo consiste nel posizionare una siringa su una particolare struttura chiamata "syringe pump" la quale avrà poi il compito di comprimere automaticamente la siringa fornendo il giusto flusso di DNA al circuito microfluidico.

5.1 Micro pump

Le micro pump o micro pompe, sono MEMS[5] molto utilizzati nel settore microfluidico. Esse permettono, tramite l'uso di due micro-attuatori piezomeccanici di pompare un liquido o un gas in un tubo.
In particolare in figura 5.1 è mostrata la micropompa utilizzata nel progetto, prodotta dall'azienda "Bartels mikrotechnik" è la "mp6-hyb". Tra le svariate micropompe che la ditta produce è stata scelta questa, per due ragioni principali, ovvero: essa può pompare liquidi e garantisce il flusso maggiore tra tutte le micropompe del suo genere.

[5]La tecnologia MEMS o Micro Electro-Mechanical Systems, indica una tecnologia che unisce la costruzione su Silicio con la micromachining. Questa unione consente di realizzare attuatori e sensori dalle dimensioni estremamente ridotte e con un'ottima precisione. Inoltre l'utilizzo di questa tecnologia garantisce un basso consumo di energia, un costo molto basso ed un peso ridotto, grazie a queste caratteristiche i dispositivi MEMS, sono sempre più utilizzati.

STUDENTSUPSI

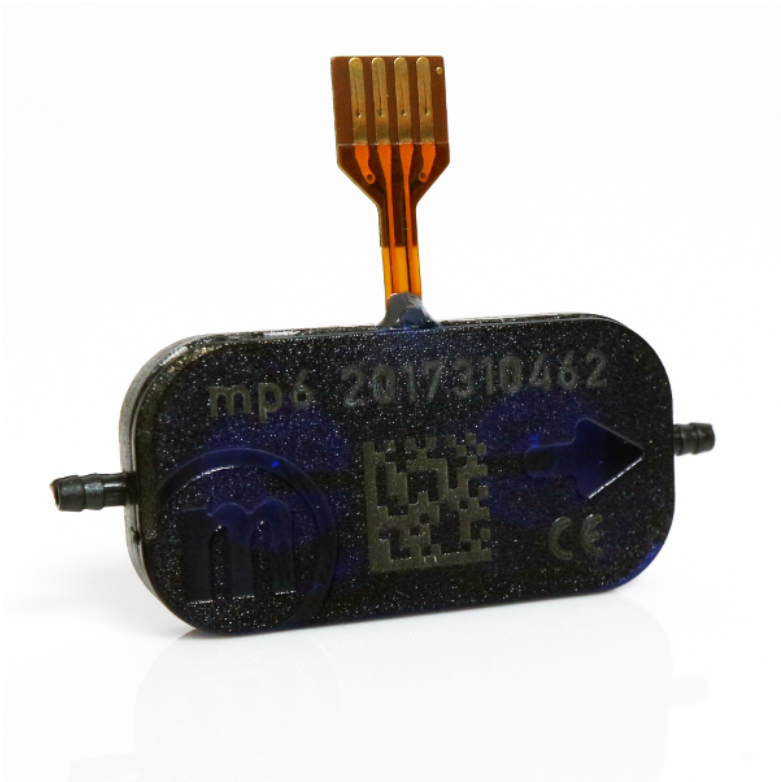


Figura 5.1: Micropompa

5.1.1 Funzionamento Micropump

Come detto in precedenza questi MEMS, si servono di due attuatori piezomeccanici uniti a due interruttori ceramici per svolgere la loro funzione. In figura 5.2 è mostrato il diagramma di funzionamento della pompa diviso in 4 fasi

1. Entrambe le membrane sono rilassate e si aprono le due valvole esterne.
2. La prima membrana si alza così da risucchiare il liquido al suo interno, viceversa la membrana opposta si chiude completamente così da espellere tutto il liquido raccolto, in questa fase le valvole restano come prima.
3. Si chiudono le valvole esterne e si aprono quelle interne, in modo da lasciar scorrere il flusso di liquido da una camera all'altra.
4. La seconda membrana si alza per risucchiare tutto il liquido, mentre quella opposta si schiaccia per espellerlo completamente.

STUDENTSUPSI

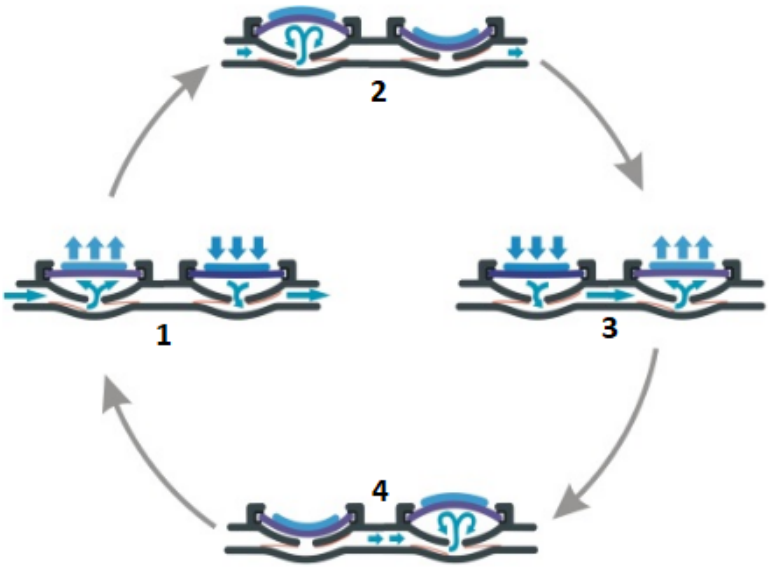


Figura 5.2: Diagramma Micro pump

In base al tipo di pompa scelta si avranno determinati parametri, i più importanti sono flusso e pressione massima, inoltre è interessante notare come per questi MEMS il rapporto tra pressione applicata e flusso di liquido sia lineare. Questa caratteristica si può notare dal grafico 5.3 che mostra il flusso in funzione della pressione esterna applicata alla pompa, maggiore sarà la pressione esterna applicato, minore il flusso di fluido che scorrerà nella pompa. Dal grafico si può notare anche il limite di flusso che una pompa di questo tipo può raggiungere, ovvero circa 7μL al minuto.

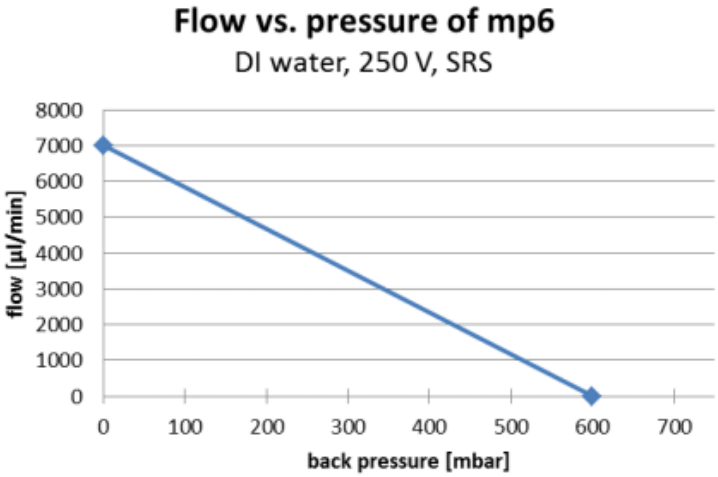


Figura 5.3: Relazione Flusso-Pressione

STUDENTSUPSI

5.1.2 Alimentazione

Per controllare questo tipo di strumenti è necessario inviare due segnali, uno per il controllo del primo attuatore, mentre l'altro per il controllo del secondo. I due segnali devono essere tra loro sfasati di 180° ed il sistema può essere comandato sia con una sinusoide che con un'onda quadra. Nel primo caso si diminuirà al minimo il rumore prodotto dal circuito, mentre nel secondo si avrà un migliore controllo sui fluidi che scorrono all'interno della pompa. Inoltre, la tensione richiesta da questa pompa varia tra i 100 V e i 300 V, poiché per smuovere delle piastre piezoelettriche la differenza di potenziale ai loro capi deve essere molto ampia.
Per questo progetto l'alimentazione è stata affidata a due componenti principali

- **HV513**
- **HV9150**

Questi due componenti consentono di produrre la tensione richiesta e di comandare multiple pompe allo stesso tempo.

5.2 Syringe pump

Questo strumento, di cui si può vedere una foto in figura 5.4 è molto usato in diversi settori, primo tra tutti il medicale, ma anche nell'alimentare e nel settore chimico. Il suo scopo è fornire un flusso controllato di un particolare fluido, per farlo, una volta montatagli sopra una siringa esso procede a schiacciarla lievemente, tramite l'ausilio di un pistone. Il concetto

Figura 5.4: Syringe pump

su cui si basa questo strumento è piuttosto semplice, tuttavia col passare degli anni si è provveduto a produrre "syringe pump" sempre più complesse e con multipli controlli al fine di erogare quantità di fluido più precise.

STUDENTSUPSI

5.2.1 Funzionamento syringe pump

Una siringa viene montata sulla struttura posizionandola su un blocco di fissaggio fatto apposta. A questo punto viene avviato un motore passo-passo (o DC a dipendenza del modello), il quale, fa ruotare una vite senza fine posta sotto la struttura. Alla vite è connesso un pistone che si muove con essa e muovendosi in avanti spinge sempre di più la siringa, finché non viene raggiunto il valore di fluido voluto o si svuota la siringa.
Nei modelli automatizzati è possibile controllare quanto liquido viene erogato anche senza una retroazione, questo poiché un controllore interno conta i passi(nel caso di un motore passo-passo) o il tempo di alimentazione (nel caso di un motore DC) e sapendo che siringa è stata posizionata (alcuni modelli ammettono un solo tipo di siringa dal volume fisso, mentre altri chiedono che siringa si sta utilizzando) è in grado di calcolare quanto fluido è stato erogato e quanto ne è ancora presente. Questa funzione è particolarmente utile in ambito medicale, poiché se deve essere somministrata una soluzione ad un paziente, la "syringe pump" è in grado di segnalare che il fluido sta finendo e che quindi va sostituito. In figura**??** è riportata una rappresentazione del funzionamento della "syringe pump".

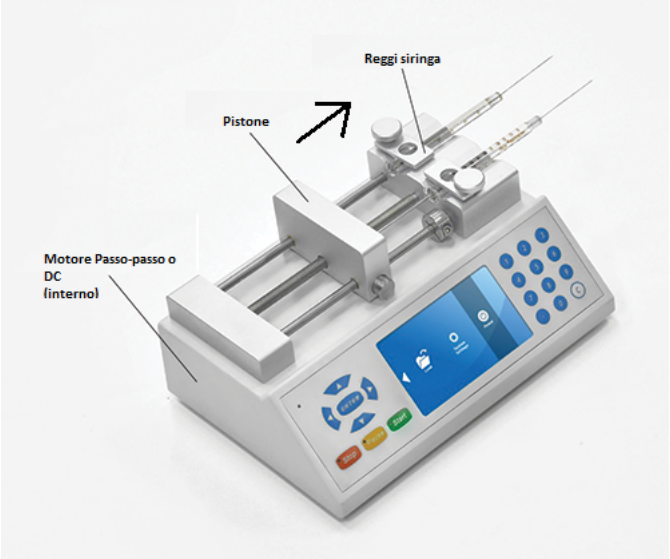


Figura 5.5: Funzionamento syringe pump

STUDENTSUPSI

Il modello usato per questo progetto è una struttura base, essa presenta un aggancio che ammette multiple siringhe, che vanno da un range di 1mL a 150mL, è comandata da un motore passo-passo e può spingere una sola siringa alla volta.
Purtroppo non presenta nessun tipo di controllo ne sul fine corsa, ne sul flusso istantaneo, ragione per cui è stato implementato un'interruttore che permette di segnalare che la siringa è arrivata alla fine. In figura 5.6 è mostrato il modello acquistato, le ragioni per cui è stato scelto questo modello rispetto ad altri o al costruirlo direttamente, sono il prezzo e la versatilità che questa particolare syringe pump offre, inoltre, le dimensioni(290 mm * 130 mm * 130 mm) non eccessivamente grandi hanno permesso un posizionamento nella struttura finale più agiato e comodo.

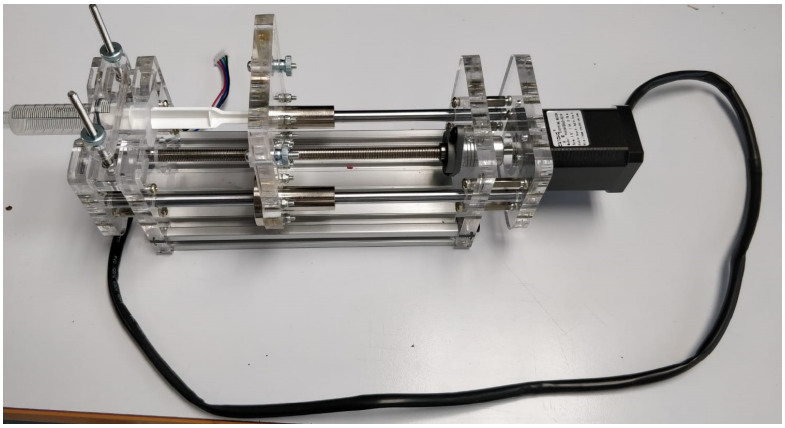


Figura 5.6: Syringe pump progetto

5.2.2 Alimentazione Syringe pump

Per controllare il motore passo-passo della "syringe pump" viene usato un driver prodotto dalla MPS, questo driver è in grado di comandare il motore modificando anche la grandezza di uno step, fino ad arrivare ad 1/8 di passo del motore. Il componente è l'**MPS6500** e sarà analizzato nei prossimi capitoli.

STUDENTSUPSI

Capitolo 6

Sensore di pressione

6.1 Soluzioni possibili

Per garantire un buon controllo sul sistema sono state studiate diverse soluzioni possibili, comprendenti diversi sensori.

6.1.1 LoC personalizzato

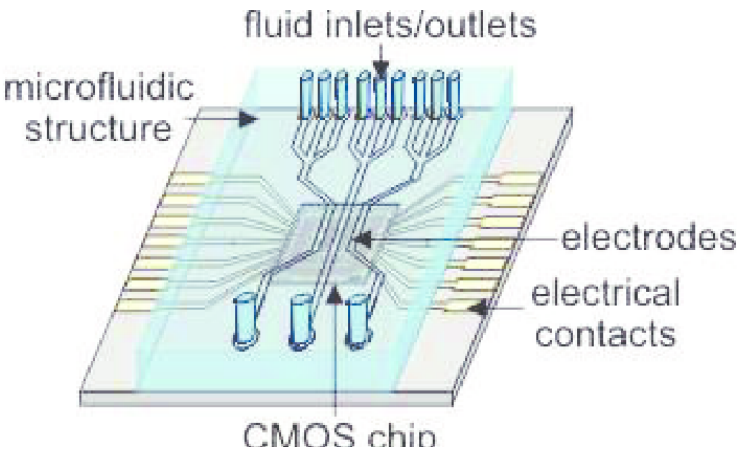


Figura 6.1: Chip microfluidico personalizzato

La prima idea visibile in figura 6.1, scartata immediatamente, era quella di creare un Lab-On-Chip personalizzato che comprendesse al suo interno sensori di: flusso, temperatura e pressione. I vantaggi di questa soluzione erano i seguenti:

- Massima compattezza possibile, creando un chip apposta, la parte di rilevazione del progetto sarebbe stata tutta concentrata in un'area molto ridotta, aumentando la robustezza del sistema(perché tutto integrato), consumi e peso.

STUDENTSUPSI

- Alta sterilità e resistenza alle contaminazioni da parte di agenti esterni, molto importante per via del fatto che la quantità di fluido disponibile è limitata e difficile da sintetizzare.

Ciononostante, questa soluzione presentava un problema non indifferente, ovvero il prezzo. Infatti, sistemi di questo tipo completamente, o anche parzialmente personalizzati arrivano a costare anche decine di migliaia di franchi e visto il budget limitato, questa scelta è stata scartata.

6.1.2 Sensori intermedi

In seconda battuta si era pensato di usare dei sensori esterni, posti tra l'input ed il chip microfluidico 6.2. Per questa soluzione sono stati analizzati molteplici sensori fatti appositamente per apparecchiature microfluidiche, i parametri che si volevano misurare erano: flusso, pressione, umidità e temperatura, anche se questi ultimi risultano complicati da reperire per simili applicazioni.

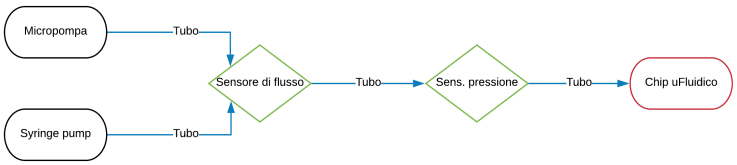


Figura 6.2: Diagramma di interfacciamento multipli sensori

I vantaggi portati da una simile configurazione sono:

- Alta sensibilità e ottima precisione di misura.
- Possibilità di ampliare la quantità di sensori, anche in un secondo momento.

Purtroppo sono presenti anche degli svantaggi che hanno reso anche questa scelta inapplicabile:

- Resistenza alla contaminazione esterna piuttosto bassa, in quanto nei sensori possono filtrare polveri sottili, che potrebbero entrare a contatto con il fluido usato.
- Il prezzo, pur essendo decisamente minore al prezzo di un LoC personalizzato, dispositivi creati apposta per la microfluidica costano centinaia di franchi.
- Spazio occupato, viste le basse quantità di fluido introdotto in questi sensori, garantire una misura accurata richiede una parte elettronica ed una meccanica piuttosto complesse e, spesso con dimensioni Piuttosto grandi, un sensore di pressione e flusso può arrivare ad occupare uno spazio di 30x30cm.

STUDENTSUPSI

6.2 Sensore di pressione e flusso

A questo punto è risultato ovvio che usare sensori creati apposta per l'ambito microfluidico, non era possibile. Perciò si è trovata una soluzione che garantisse una buona resistenza agli agenti esterni e che potesse fornire sufficienti informazioni sul fluido.
Essa consiste nell'utilizzare un sensore di pressione, sia per misurare la pressione, che(tramite una derivazione della legge di Bernoulli) il flusso. Per garantire la sterilità del sistema, invece, si è pensato di incapsulare il sensore, insieme al chip microfluidico in un blocco chiuso, lasciando il corpo del sensore all'esterno. Il modello di questa soluzione è presentato in figura 6.3.

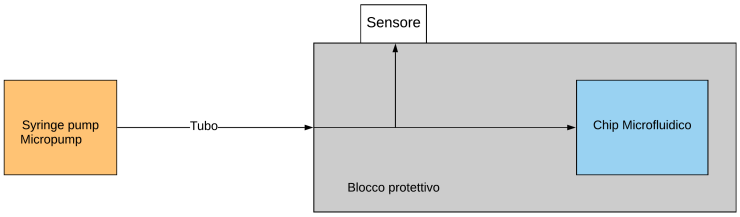


Figura 6.3: Nella figura è mostrato il modello della soluzione con un singolo sensore in grado di calcolare pressione e flusso

La scelta di quale sensore utilizzare si è basata su tre fattori principali, ovvero:

- Tensione di alimentazione
- Range di pressione
- Capacità di misurare pressione di liquidi

6.2.1 Tensione di alimentazione

Questo fattore era molto importante poiché il sensore doveva essere interfacciato con il microcontrollore usato per il controllo di tutte le periferiche del progetto. Tale microcontrollore viene alimentato dalla board tramite un tensione di 3.3V, per cui, onde evitare ulteriori circuiti, che avrebbero aumentato consumi e spazio, si è cercato un sensore che lavorasse con questa tensione di alimentazione.

6.2.2 Range di pressione

Per garantire una misura sia nel caso dell'utilizzo della micropompa, che della syringe pump, si è fissato come pressione minima accettabile per il sensore 500mBar. Il valore scelto non deve essere nè troppo basso, nè troppo alto. Nel primo caso non si potrebbe garantire un

STUDENTSUPSI

flusso rapido, nel secondo, invece, sorgerebbe il rischio di predite nelle giunzioni tra tubi e chip microfluidici.

6.2.3 Misura di liquidi

Lavorando con un liquido è necessario assicurasi, che il sensore utilizzato possa entrare a contatto con liquidi senza danneggiarsi. Ciononostante, è possibile utilizzare anche un sensore per gas, in questo caso, però, il posizionamento del sensore dovrebbe garantire che rimanga dell'aria tra sensore e liquido. Per evitare complicazioni, si è quindi cercato un sensore che resistesse al contatto con i liquidi.

STUDENTSUPSI

6.3 MP3V5050DP

Figura 6.4: Sensore di pressione NXP

Dopo un'analisi di mercato, la scelta è ricaduta sul sensore raffigurato in figura 6.4. Prodotto da "NXP", l'"MP3V5050DP" è un sensore di pressione amplificato e compensato in temperatura, usato per applicazioni che involvono principalmente gas, ma compatibile anche con liquidi. Le sue specifiche sono:

- Tensione di alimentazione 3.3V.
- Range di pressione differenziale(differenza di pressione misurata tra una punta e l'altra) misurato, fino a 500 mBar.
- Accuratezza 2.5$\%$ rispetto al valore massimo.
- Tempo di risposta 1.0 ms.
- Pressione massima sostenuta 2 Bar.

6.3.0.1 Caratteristiche tecniche

La figura 6.5 mostra lo schema di funzionamento del sensore, in pratica esso è composto da un elemento sensibile alla pressione che in base ad essa genera una tensione, essa viene inizialmente amplificata in un primo stadio che ha il compito sia di amplificarla che di compensare l'errore derivato dalla temperatura, a questo punto il segnale passa in un secondo stadio di amplificazione, che, sposta il segnale in base allo '0' e poi lo manda in uscita. Questo sensore, rispetto ad altri, produce un'uscita puramente analogica, quindi varia la tensione sul suo piedino V_{OUT} proporzionalmente alla pressione rilevata (tralasciando gli errori), ciononostante esistono sensori simili che forniscono l'uscita anche con protocolli digitali, come ad esempio l'"ABPDRRT005PG2A5" prodotto da "Honeywell", la cui uscita

STUDENTSUPSI

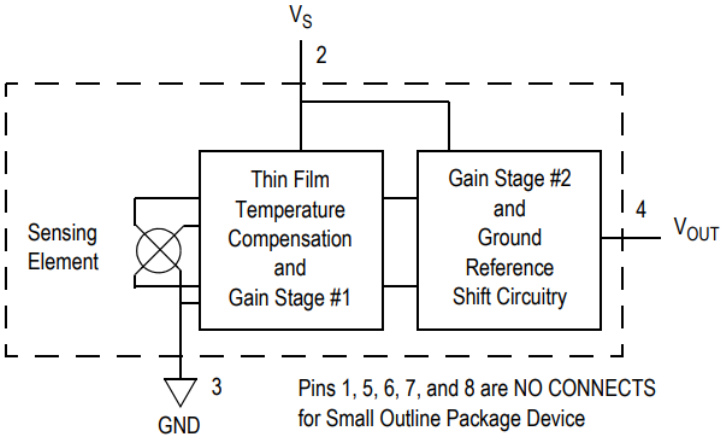


Figura 6.5: Schema interno sensore

viene trasmessa con protocollo I^2C. L'uscita dal sensore dipende da diversi fattori, infatti per calcolarlo esso sfrutta la seguente equazione:

$$V_{OUT} = V_S \cdot (P \cdot 0.018 + 004) \pm (Pressureerror \cdot Temp.factor \cdot 0.018 \cdot V_S) \tag{6.1}$$

da cui si deriva:

$$P = \frac{V_{OUT}}{V_s \cdot 0.018} - \frac{0.04}{0.018} \mp (Pressureerror \cdot Temp.factor) \tag{6.2}$$

dove:

- $V_{OUTSENS}$ è la tensione di uscita dal sensore
- V_S è la tensione di alimentazione + 0.30V
- P è la pressione misurata
- $Pressureerror$ l'errore stimato sulla pressione misurata. Per calcolarlo si usa il grafico 6.6
- $Temp.factor$ l'errore derivato dalla temperatura in cui opera il sensore, il suo valore si può trovare tramite il grafico 6.7

[h]

STUDENTSUPSI

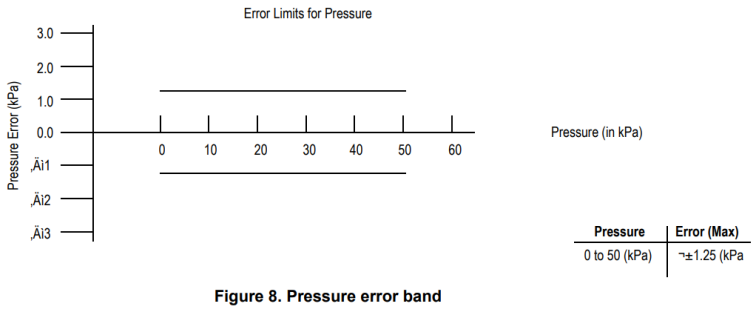


Figura 6.6: Grafico errore di pressione

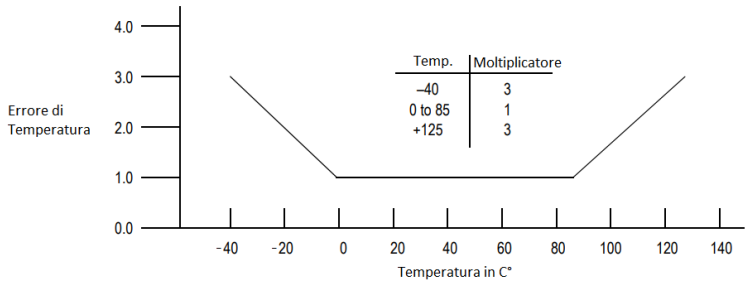


Figura 6.7: Il grafico mostra come varia l'errore di misura in funzione della temperatura di lavoro in quel momento

La figura 6.8a raffigura lo schema elettrico del sensore, mentre la figura affianco mostra i componenti raccomandati per la lettura corretta dei dati del sensore e per diminuirne l'errore. Come si può notare dalla prima figura il sensore è fatto per essere montato su un PCB, per questo progetto però esso è stato lasciato esterno al blocco protettivo contenente il chip microfluidico, e tramite dei cavi è stato portato alla board di controllo che ne ha potuto leggere i dati.

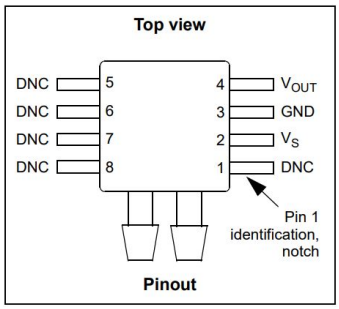
6.3.1 Bernoulli e Venturi

Come detto in precedenza tramite una derivazione del teorema di Bernoulli è possibile calcolare il flusso con l'ausilio di un sensore di pressione.
Il teorema di Bernoulli[6] è una relazione della fluidodinamica formulata dall'omonimo Dottore nel 1738. Si supponga di avere un tubo la cui sezione ed altezza varia, nel quale scorre un fluido ideale, come mostrato in figura 6.9. L'equazione di Bernoulli 6.3, sostiene che la relazione tra pressione altezza e velocità è costante al variare della sezione e del flusso nel tubo.

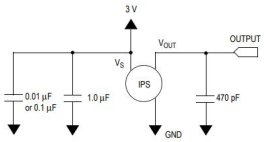
$$p_1 + \frac{1}{2} \cdot \rho v_1^2 + \rho g h_1 = p_2 + \frac{1}{2} \cdot \rho v_2^2 + \rho g h_2 \tag{6.3}$$

[6]Daniel Bernoulli(1700-1782), Matematico e fisico Svizzero, è stato autore di notevoli pubblicazioni scientifiche, tra cui la più celebre "Hydrodynamique" dove presentò il suo famoso teorema.

STUDENTSUPSI



(a) Schema elettrico sensore



(b) Interfacciamento elettrico sensore

Figura 6.8: A sinistra è mostrato lo schema elettrico del sensore, mentre a destra i componenti di cui ha bisogno per una lettura corretta

da cui:

$$p + \frac{1}{2} \cdot \rho v^2 + \rho g h = costante \tag{6.4}$$

Dove:

- p è la pressione nel tubo al punto 1 e 2
- v è la velocità del fluido
- ρ è la densità del fluido che scorre attraverso il tubo
- h è l'altezza alla quale si trova il tubo

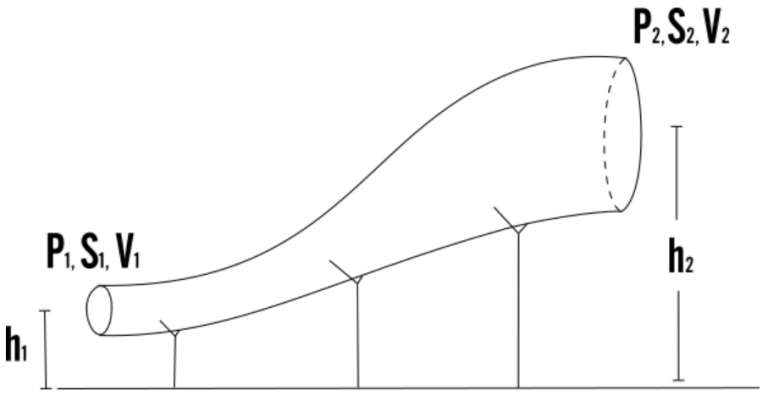


Figura 6.9: Con h si indica l'altezza, P la pressione e V la velocità

Dall'equazione di Bernoulli sono derivati diversi effetti conosciuti come: l'effetto Magnus[7], la Portanza[8] e l'effetto Venturi. Quest'ultimo, permette di calcolare il flusso di un fluido

[7]particolare effetto derivato dalla legge di Bernoulli, che descrive come varia la traiettoria di un oggetto in un fluido se esso sta ruotando.

[8]La portanza fluidodinamica (diversa dalla portanza strutturale) è la spinta ricevuta da un oggetto quando esso attraversa un fluido provocandone un cambiamento di pressione. La portanza fluidodinamica è l'effetto

STUDENTSUPSI

conoscendone pressione, densità e sezione del tubo.
Si osservi la figura 6.10 essa presenta un tubo orizzontale (quindi stessa altezza per tutto il tubo) e si supponga sia attraversato da un fluido, tale fluido subirà due diverse pressioni a seconda della sezione di tubo in cui si trova, di conseguenza varierà anche la sua velocità. Misurando la pressione del fluido nella prima e seconda sezione, conoscendo la densità e

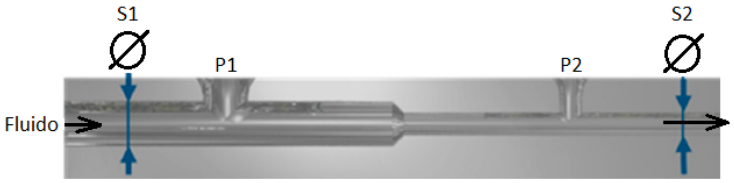


Figura 6.10: Nell'immagine è rappresentata una sezione di un tubo di Venturi

le due sezioni, si può, tramite la formula 6.5 calcolare il flusso del fluido.

$$Q = A_2 \cdot \sqrt{\frac{2(P_1 - P_2)}{\rho \cdot ((\frac{A_1}{A_2})^2 - 1)}} \tag{6.5}$$

dove:

- A_2 Seconda sezione del tubo
- A_1 Prima sezione del tubo
- P_1 Pressione misurata nella prima sezione del tubo
- P_2 Pressione misurata nella seconda sezione del tubo
- Q è

Proprio grazie all'effetto appena descritto è stato possibile misurare il flusso del liquido iniettato nella piattaforma microfluidica. Per farlo si è sfruttato il blocco in cui è stato incapsulato il chip, al suo interno è stato creato un canale con due sezioni diverse. Sopra questo canale sono stati fatti due fori utilizzati per inserire i due "beccucci" del sensore. Inoltre, il sensore utilizzato ha il vantaggio di essere differenziale, per cui la sua uscita è già la differenza di pressione tra le due sezioni, semplificando ulteriormente il calcolo del flusso.

che permette agli aerei di volare.

STUDENTSUPSI

STUDENTSUPSI

Capitolo 7

Elettronica e Controllo

7.1 Struttura

Per realizzare l'elettronica di questo progetto è stato deciso di separare in due l'intero sistema, quindi creare due differenti board.
La prima ha compito di gestire le alimentazioni relative al motore passo-passo della syringe pump ed alle micropompe, mentre la seconda, dove risiede il microcontrollore, ha il compito di acquisire dati ed inviare istruzioni a tutte le periferiche collegate.
La decisione di usare una struttura con due board separate piuttosto che una singola è data dalla possibilità di gestire anche in un secondo momento il posizionamento dell'elettronica all'interno del package scelto, infatti, in caso lo spazio risultasse inadatto, è possibile spostare una delle due board senza dover cambiare obbligatoriamente il contenitore.

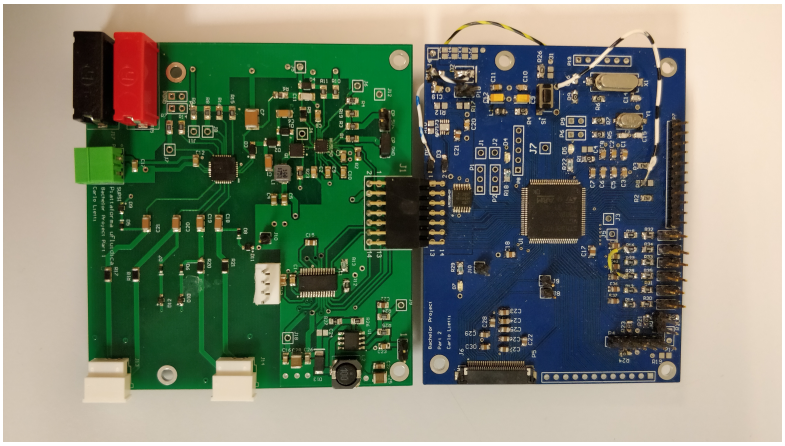


Figura 7.1: Board di Alimentazione e controllo

STUDENTSUPSI

7.2 Board di alimentazione

7.2.1 Schematici

In figura 7.2 e 7.3 sono mostrati i due schematici disegnati per la creazione di questa board, che ora verranno analizzati più nel dettaglio.

STUDENTSUPSI

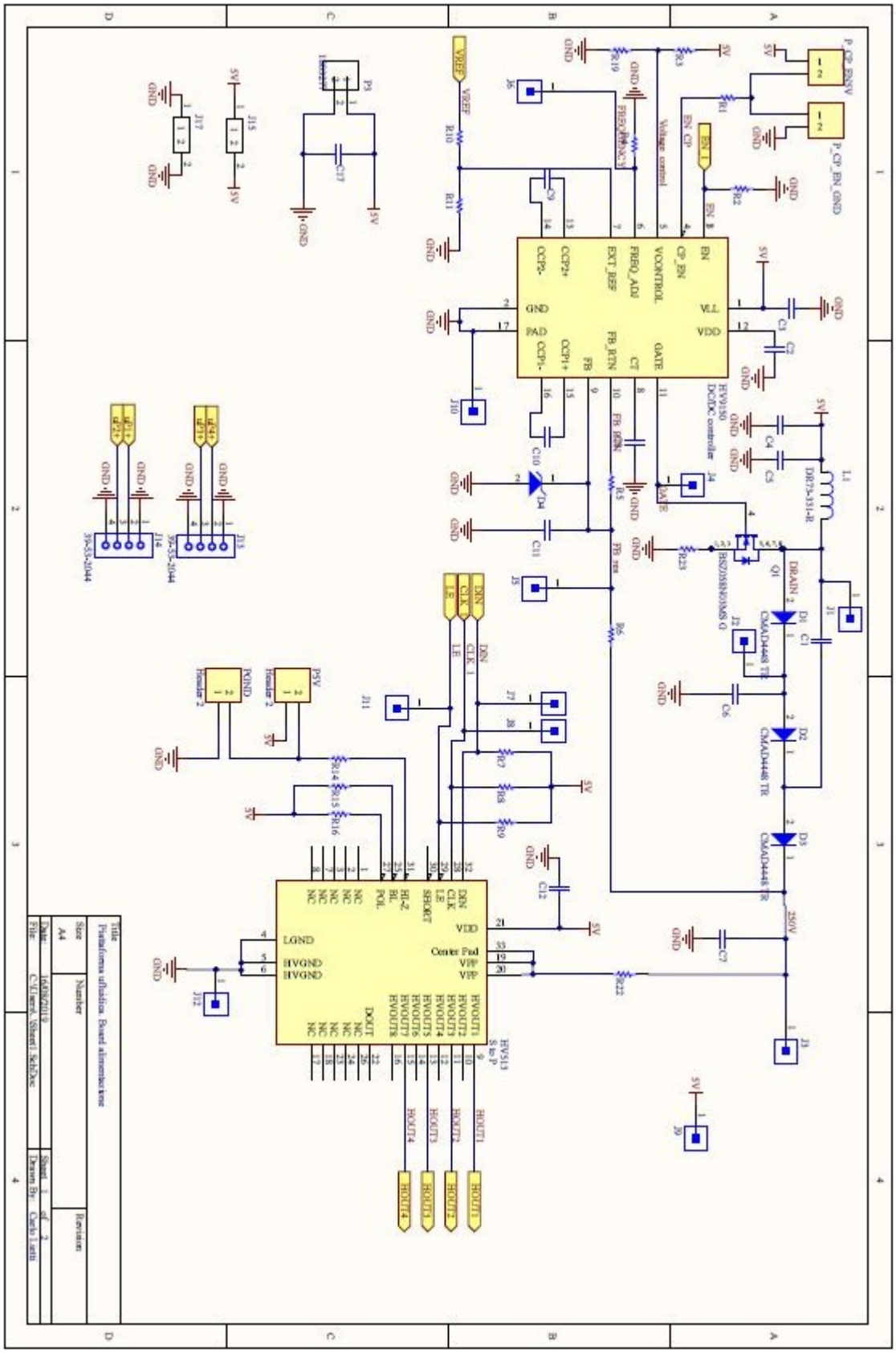


Figura 7.2: Foglio n.1 board alimentazione

STUDENTSUPSI

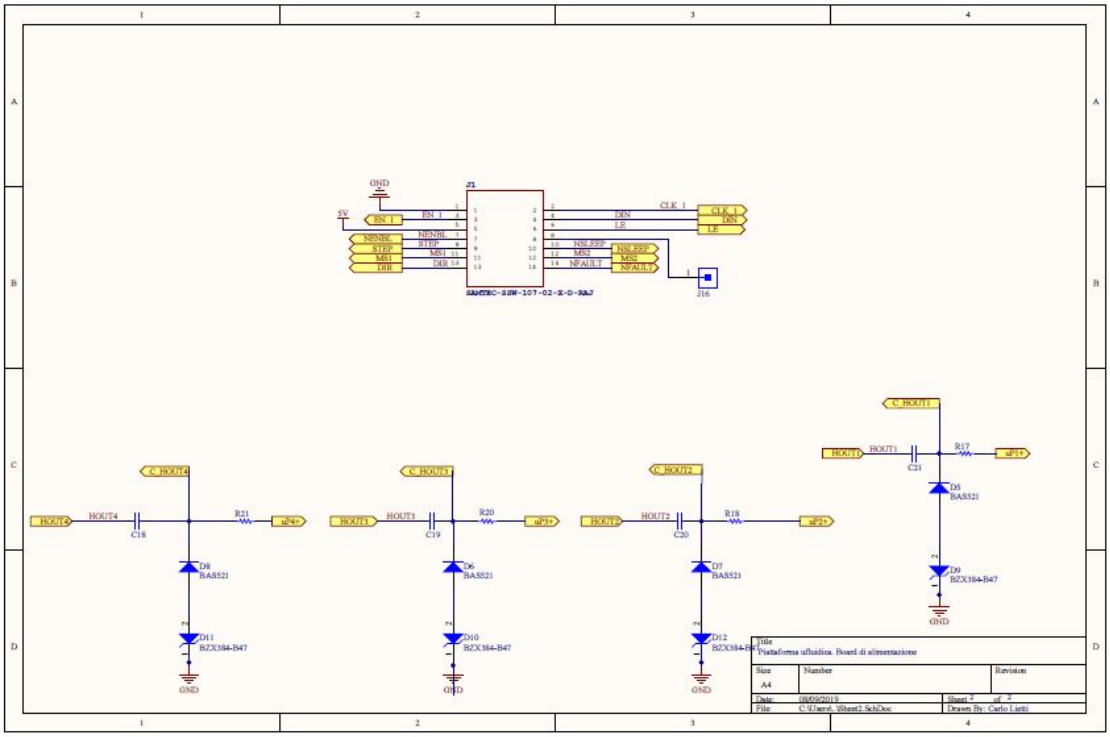


Figura 7.3: Foglio n.2 board alimentazione

STUDENTSUPSI

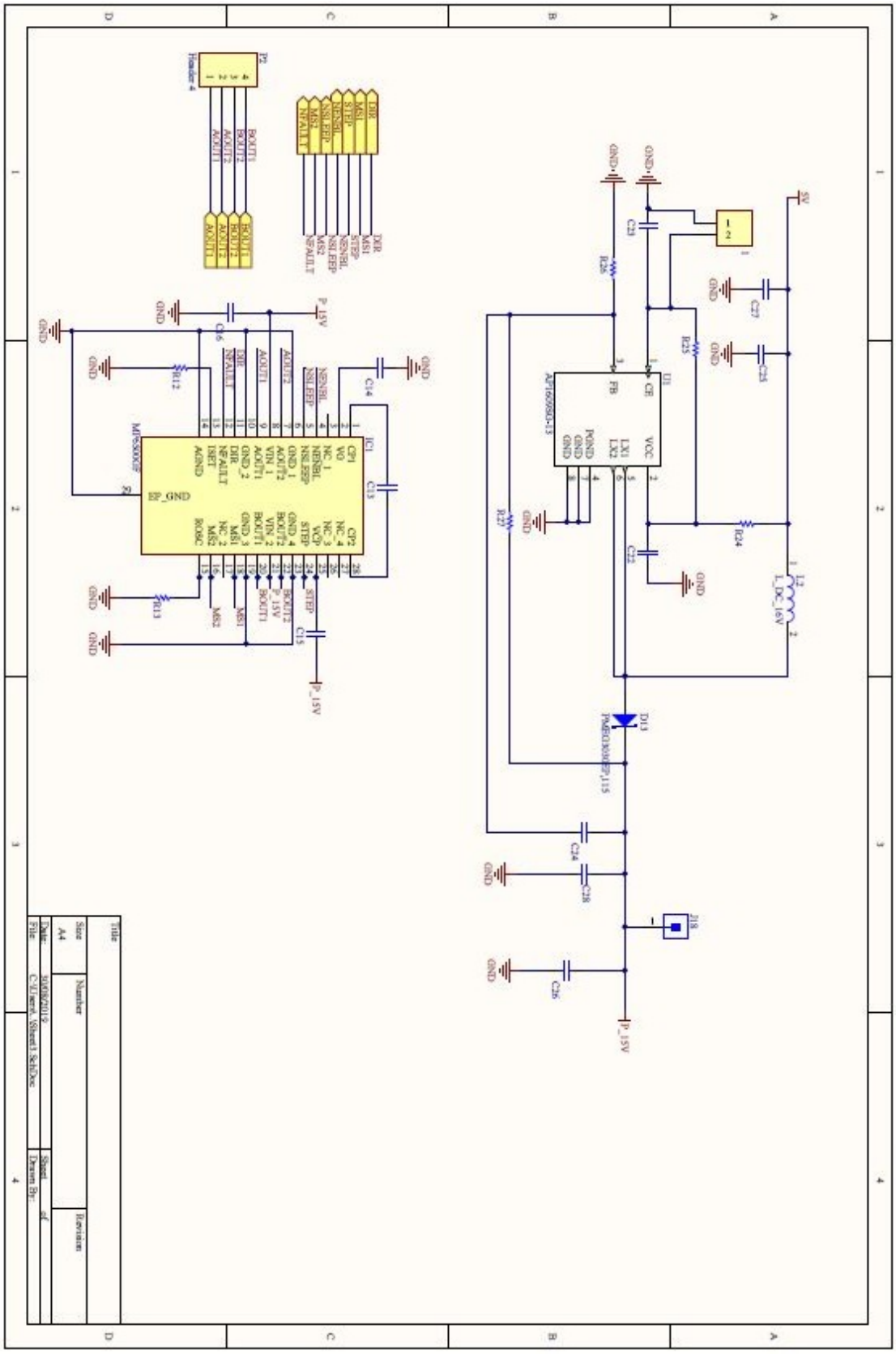


Figura 7.4: Foglio 3 Board di alimentazione

STUDENTSUPSI

7.2.1.1 Foglio 1 BdA

Nel primo foglio sono presenti le alimentazioni principali e il circuito che controlla l'avviamento delle micropompe. I due IC presenti sono l'HV9150 e l'HV513, sono rispettivamente un "controllore DC/DC" ed un "convertitore da seriale a parallelo", entrambi saranno largamente analizzati nelle sezioni successive.

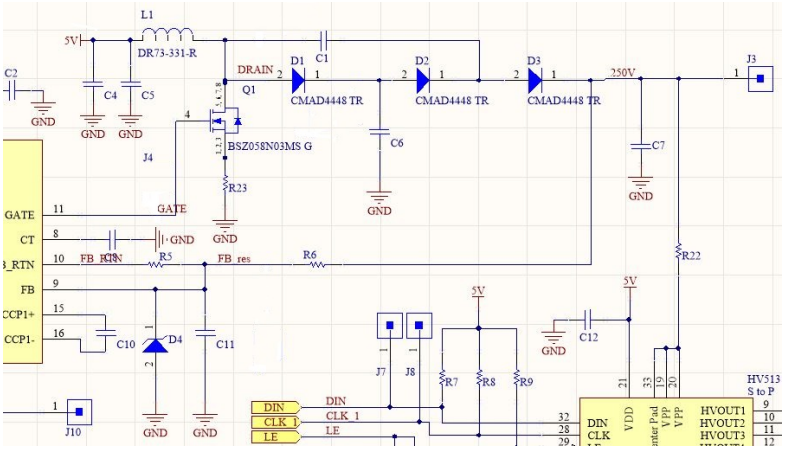
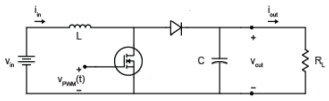
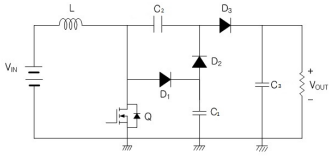


Figura 7.5: Circuito di Boost

Il punto fondamentale di questo schematico è il circuito che si occupa del boost di tensione da 5V a 250V, di cui si può vedere una versione ingrandita in figura7.5. Il funzionamento è simile ad un normale boost DC/DC, con la particolarità che per aumentare la tensione di un fattore 50 si usa un doppio stadio a diodi, piuttosto che uno singolo. La differenza tra i due circuiti è mostrata in figura 7.6, come si può notare dalle immagini nella versione usata per questo progetto, sono stati aggiunti due diodi e due condensatori. Questa configurazione è detta "Voltage doubler".



(a) DC/DC boost a uno stadio



(b) DC/DC boost con voltage doubler

Figura 7.6: Differenti tipologie di DC/DC boost

Tale modifica riduce notevolmente lo "stress di tensione" imposto al diodo, siccome viene diviso sui due stadi. Ciononostante questo circuito, richiede il medesimo apporto di energia di un boost normale, inoltre la corrente che circola è più alta della sua controparte.

7.2.1.1.1 Funzionamento Si assuma che, accensione del transistor la situazione sia:

STUDENTSUPSI

- Il condensatore C1 sia carico a 125V
- C2 sia carico a 125V
- C3 sia carico a 250V

una volta acceso il transistor, scorre una corrente attraverso l'induttore, il quale si carica. Nel frattempo, il condensatore C1 si scarica tramite il diodo D3 e va a fornire energia al condensatore C3. Il compito di C2 è mantenere la tensione su C1 ad almeno 125V, mantenendo un'offset[9] costante sul diodo.
Nel momento in cui il transistor viene spento, l'induttore fa scorrere una corrente attraverso D1, che carica C2. Successivamente, tramite D2 la tensione di C2 viene portata fino a 250V, infine per tutta la durata di questa fase l'induttore carica C3 e mantiene la tensione di uscita stabile. Per far si che il circuito funzioni al meglio sono stati scelti componenti che fossero fatti apposta per questo tipo di applicazione, in particolare:

- **Diodi** per applicazioni di questo tipo sono necessari dei diodi switching, che possano sostenere tensioni piuttosto elevate, superiori a 100V. La scelta è ricaduta sui "**RF071MM2STFTR**", prodotti dalla "Central semiconductor", questi possono sostenere uno una frequenza di switching fino a 83MHz.
- **MOSFET** I fattori più importante per il MOSFET sono la tensione massima applicabile tra i due estremi Drain-source, la massima corrente sostenuta dal transistor e la resistenza quando "ON", infatti il DC/DC controller usato raccomanda una resistenza minore possibile per il transistor al fine di evitare che il componente si scaldi troppo quando in funzione. Viste queste specifiche da rispettare, si è optato per il "**BSZ900N15NS3**", questo componente è particolarmente indicato applicazioni simili, in quanto sostiene alte tensioni, fino a 150V, consente il passaggio di una corrente fino a 13A ed ha una resistenza quando "on" di 90mΩ, inoltre il suo package consenta un'ottima dissipazione termica in quanto presente una piazzola fatta, appunto, per dissipare il calore sviluppato.

7.2.1.2 Foglio 2 BdA

Nel secondo schematico 7.3 sono presenti:

- I circuiti di clamping utilizzati per portare la tensione in ingresso alle micropompe da 0V-250V a -50V-200V, questo poiché le micropompe richiedono una tensione negativa per un corretto funzionamento. Un circuito di clamping(così è definito un circuito che può aggiungere una componente DC ad un segnale AC), si basa su 3 componenti principali, un diodo, un condensatore ed una resistenza.

[9] L'offset è un valore di tensione DC, che sposta la il riferimento di un segnale '0', verso l'alto o verso il basso, a seconda della tensione applicata

STUDENTSUPSI

7.2.1.3 Foglio 3 BdA

Il terzo schematico è dedicato al pilotaggio del motore della syringe pump, esso presenta due componenti principali:

- L'"**AP1609SG**, è un convertitore D/DC, viene utilizzato per portare la tnsione di alimentazione 5V, ad una tensione che possa pilotare correttamente il motere delle syringe pump.
- L'"**MP6500**", ovvero il driver del motore passo-passo utilizzato per muovere la syringe pump, più i componenti passivi necessari per garantirne il corretto funzionamento.

STUDENTSUPSI

7.2.2 PCB BcA

Le immagini 7.7, 7.8, 7.9 e 7.10 mostrano la scheda prodotta per la borad di alimentazione.

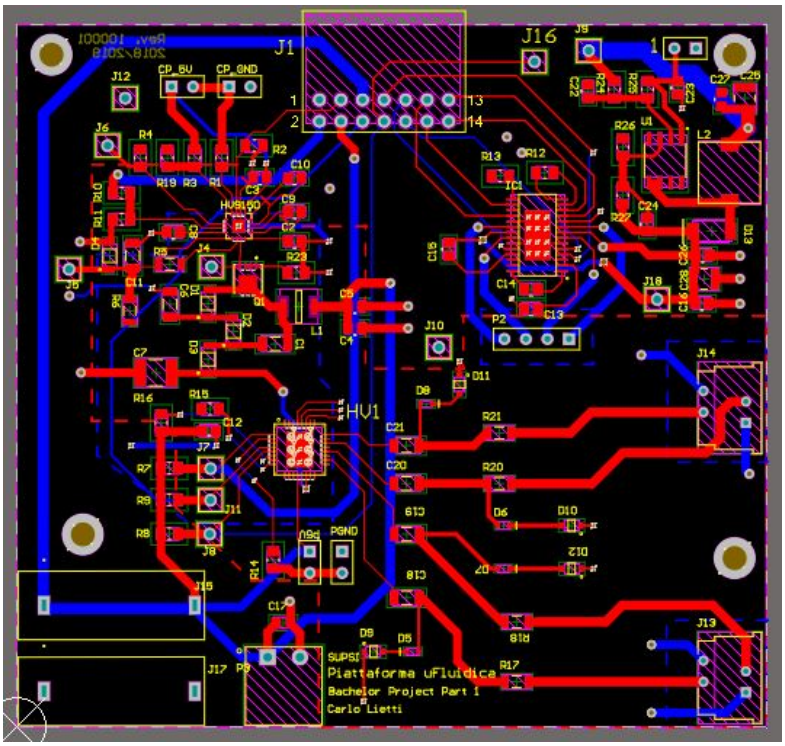


Figura 7.7: Pcb board di alimentazione, senza piani di alimentazione

STUDENTSUPSI

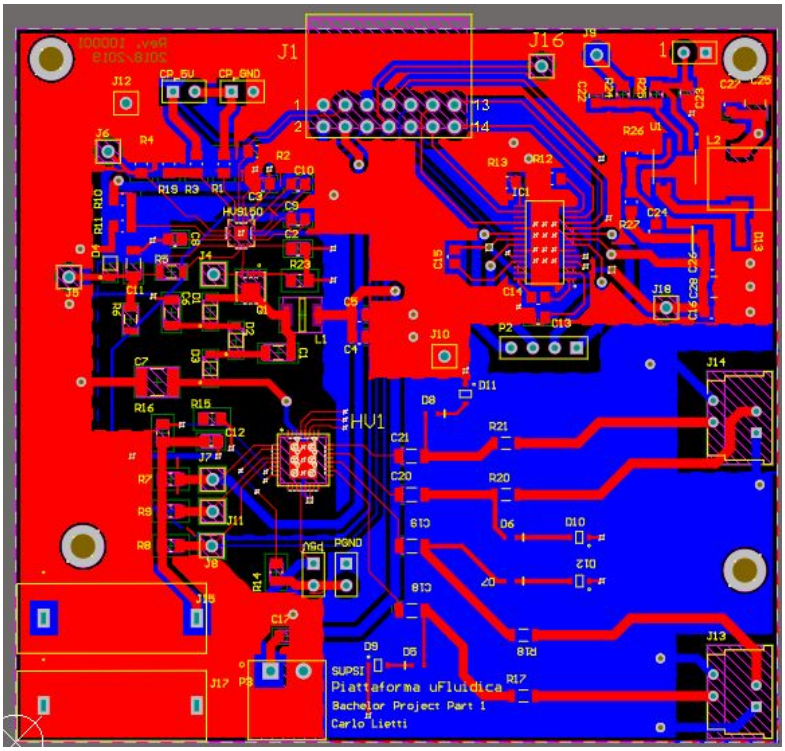


Figura 7.8: Pcb board di alimentazione, TOP layer

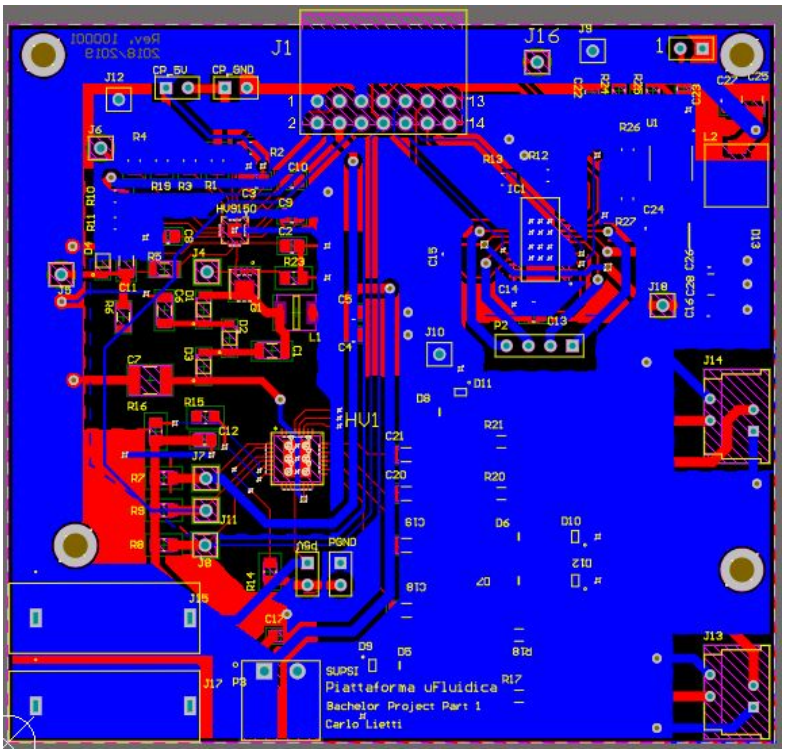


Figura 7.9: Pcb board di alimentazione, BOTTOM layer

STUDENTSUPSI

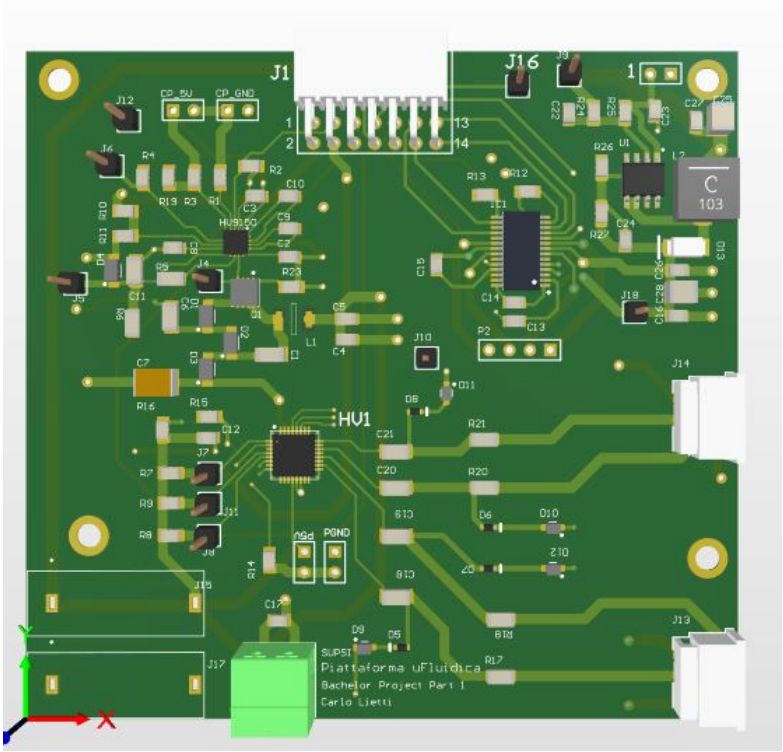


Figura 7.10: Pcb board di alimentazione, 3d

Disegnando queste board si è cercato di mantenere le piste il più larghe possibili, così da ridurre al minimo le impedenze delle piste. Sono stati posti dei piani di massa sia nel layer TOP, che in quello BOTTOM, inoltre, come si può notare dalla figura 7.9, nel piano BOTTOM, una parte di layer è stato messo a 5V. in tal modo portare le alimentazioni in tutta la board è risulto più semplice. In figura 7.11 viene mostrato un'ingrandimento della

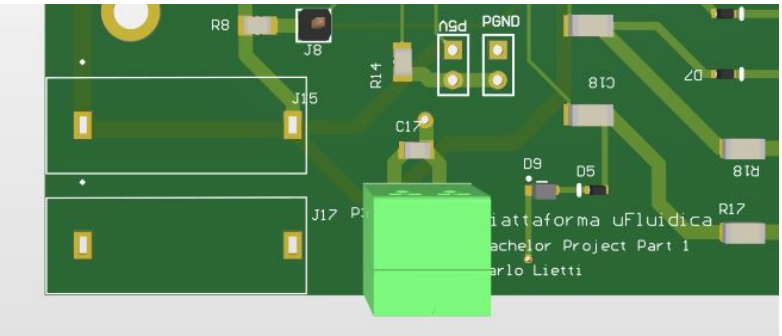
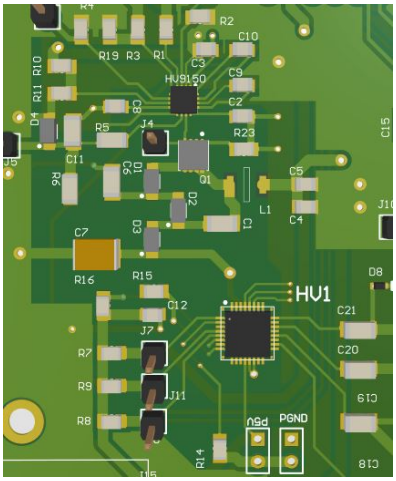


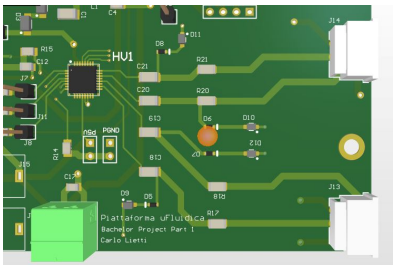
Figura 7.11: Connettori board di alimentazione

sezione di questa board, che si occupa di portare l'alimentazione a tutti i componenti. Come si può notare la board si può alimentare con due sistemi differenti o tramite delle boccole da laboratorio, oppure utilizzando un connettore phoenix. Sono stati predisposti due sistemi diversi di alimentazione, per rendere più semplice la fase di test del sistema.

STUDENTSUPSI



(a) Circuiti di alimentazione micropompe



(b) Connettori per micropompe

Figura 7.12: Circuiti di alimentazione micropompe

Le figure 7.12, mostrano un'ingrandimento dei circuiti utilizzati per alimentare le micropompe. In questa sezione del PCB, si è preferito eliminare il piano di massa, nell'immagine 7.12a, sia nel layer TOP, che in quello BOTTOM, mentre nell'immagine 7.12b, solamente nel layer TOP (tranne nei connettori, in cui è stato rimosso anche sotto). É stata fatta questa scelta, poiché dovendo alimentare le micropompe con una tensione di 250V, non si voleva rischiare che questa tensione potesse propagarsi nel resto del circuito, rischiando di danneggiarlo irreparabilmente.
La figura 7.13, mostra la sezione dedicata all'alimentazione e controllo della Syringe pump, sulla destra è posizionato il circuito di step-up, che trasla la tensione d'ingresso fino a 12V ca. Sulla sinistra, invece, è posizioneato il driver del motore, che ne controlla: step, passo, corrente e direzione.

STUDENTSUPSI

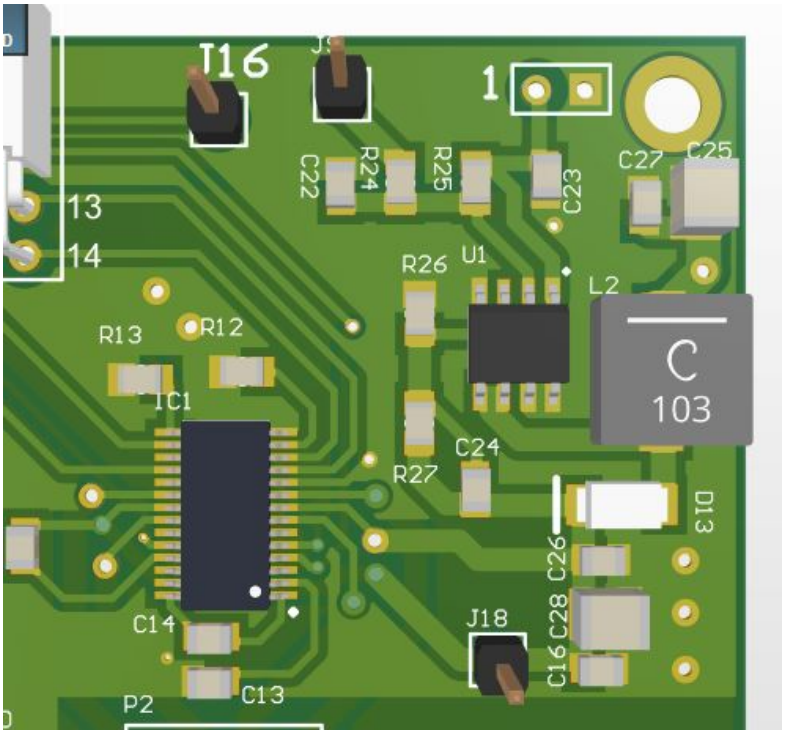


Figura 7.13: Driver motori Syringe pump

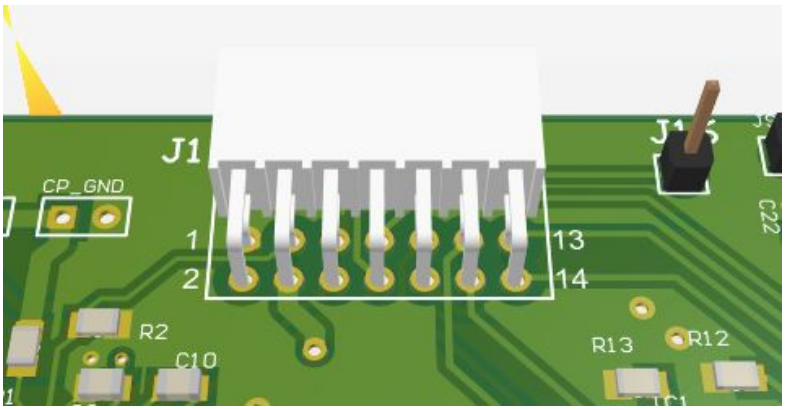


Figura 7.14: Connettore per board di controllo

Il connettore mostrato in figura 7.14 si occupa di comunicare con la board di Controllo, esso ha il compito di portare i segnali inviati dal microcontrollore ai diversi integrati della board di Alimentazione, ma viene utilizzato anche per fornire l'alimentazione all'altra board.

7.2.3 HV9150

Questo componente è un particolare DC/DC controller per la precisione è un "hysteretic high voltage DC/DC step up controller", ovvero un DC/DC controller step-up con isteresi. Esso è caratterizzato da una tensione minima di alimentazione di 2.7V, che può essere por-

STUDENTSUPSI

tata fino a 500V, inoltre questo componente è particolarmente indicato per l'alimentazione di sistemi "piezoelettrici", come le micropompe utilizzate nel progetto. In figura 7.15 è mo-

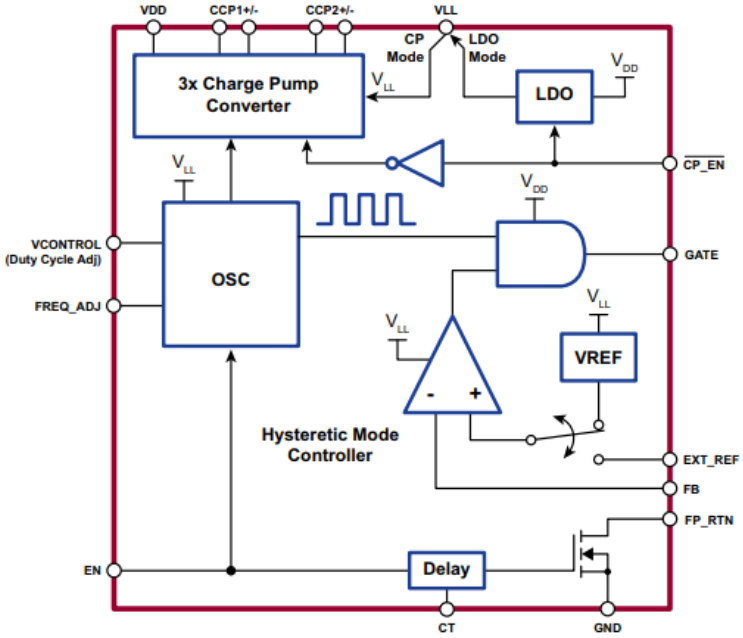


Figura 7.15: Hv9150 sch

strata la struttura interna dell'HV9150, questo componente è composto da un'oscillatore, un comparatore ed un driver, inoltre in questo componente sia la frequenza dell'oscillatore che il duty cycle d'uscita sono ad un valore fisso al fine di aumentare sensibilmente la stabilità del sistema.

Il funzionamento è molto simile ai diversi step-up di questo tipo, ovvero, il comparatore interno al componente genera una serie di impulsi i quali comandano il gate del transistor esterno; esso con la sua commutazione permette all'induttore di caricarsi e scaricarsi aumentando la tensione d'uscita.

Nel momento in cui l'uscita raggiunge il valore desiderato, il comparatore smette di inviare impulsi mantenendo il suo livello stabile. Una volta che la tensione d'uscita scende sotto la soglia richiesta il comparatore invia nuovamente impulsi. Avere un sistema con isteresi è piuttosto vantaggioso, poiché semplifica le operazioni ed aumenta la stabilità del sistema.

LDO presente all'interno del componente ha il compito, se abilitato, di portare la tensione in tutto il componente ad un livello che permette il corretto funzionamento a tutte le sue periferiche interne ovvero 3.3V, pur garantendo una tensione sul gate del transistor di almeno 5V. Per quanto riguarda la charge pump essa ha lo scopo portare una tensione tra 5V e 12V sul Gate del MOSFET esterno, questo poiché una tensione di 3.3V non è ottimale per pilotare alte tensioni, infatti, aumentando la tensione sul gate si può migliorare la cosiddetta "on-resistance" del transistor, ovvero la resistenza che esercita il transistor quando è "ON",

STUDENTSUPSI

cioè lascia scorrere corrente tra drain e source.
La charge pump interna del componente può funzionare solo con tensioni che vanno da 2.7 a 4.5V, da qui la necessità di utilizzare LDO, interno in caso di alimentazioni esterne a questo range. Lo scopo del componente nel circuito è alimentare l'HV513, esso, infatti, come descritto più avanti è in grado di "amplificare un segnale in ingresso fino alla tensione con cui viene alimentato, la quale può raggiungere i 300V.// Il componente viene fornito in un package 16-Lead QFN,qui sotto sono riportati i suoi piedini e le rispettive funzioni:

1. **VLL** É il pin di alimentazione.
2. **GND** É il pin di massa.
3. **EN** Il pin di "Enable", se portato a livello logico '1' abilita le periferiche interne al componente, viceversa le tiene spente.
4. **CP EN** Tramite questo piedino si può selezionare la modalità LDO, o charge pump.
5. **VCONTROL** Trmite la tensione fornita a questo pin si seleziona quale dei 5 possibili duty cycle utilizzare
6. **FREQ ADJ** posizionando una resistenza tra questo piedino e massa, si imposta la frequenza di lavoro dell'oscillatore interno
7. **EXT REF** Permette di impostare un riferimento per il comparatore esterno.
8. **CT** In questo piedino serve come protezione, posizionando un condensatore, si evita che il diodo interno sia soggetto ad una corrente troppo grande, la quale potrebbe danneggiarlo.
9. **FB** Questo pin fornisce un feedback sulla tensione prodotta al fine di raggiungere il vvalore desiderato
10. **FB RTN** Quando il componente non è attivo, la rete di feedback contiua a consumare energia, per questo motivo si utilizza questo piedino, che accende un MOSFET interno collegato a massa nel momento in cui viene disabilitato, la corrente passerà per una serie di resistenza che diminuiranno parzialmente i consumi.
11. **GATE** Controlla il gate del transistor
12. **VDD** Fornisce in uscita la tensione prodotta dalla charge pump.
13. **CPP1/2** Questi 4 piedini servono per il corretto funzionamento della charge pump interna.

STUDENTSUPSI

7.2.4 Funzionamento e Dimensionamento HV9150

La prima cosa da fare con questo ic è necessario decidere in che modalità si vuole operare, le possibilità sono due:

1. **LDO** "Linear Regulator" viene usata in nel caso sia presente un'alimentazione di 5V o superiore, in questo caso l'alimentazione del componente viene affidata ad un LDO interno, esso aumenta l'efficienza dell'integrato ed inoltre riesce a fornire al gate del transistor esterno una tensione sufficiente per essere pilotato.

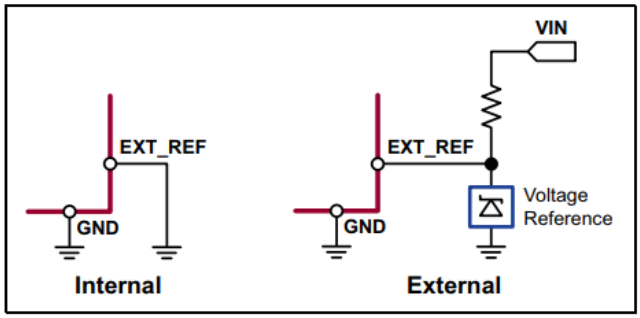
2. **CP** "Charge Pump Converter" Nel caso in cui la tensione di alimentazione fosse compresa tra 2.7V e 4V, si può utilizzare questa configurazione, essa sfrutta tre circuiti di charge pump interni per portare la tensione dell'HV513 ad un livello compreso tra 5V e 12V.

Per impostare la modalità scelta è sufficiente fornire al piedino CP_EN la tensione di alimentazione, nel caso si voglia usare la modalità **LDO**, oppure impostarlo a 0V per l'altra. In base all'applicazione può variare la precisione ed il valore che si usa come referenza per il comparatore interno, essa infatti può essere:

- $Interna$ ovvero una tensione generata internamente del valore di 1.25V $\pm 2.5\%$

- $Esterna$ una tensione compresa tra 0.5V e Vin-1.4V ottenibile con un semplice partitore resistivo.

Per impostare l'una o l'altra referenza bisogna impostare il piedino EXT_REF a 0V, nel caso in cui si scelga la referenza $interna$ ed alla tensione scelta come riferimento viceversa. La figura 7.16 mostra le due configurazioni. Inoltre è importante sottolineare che non è sufficiente dare una tensione diversa da 0V al piedino per abilitare la referenza interna, è necessario anche fornire al pin EN un fronte di salita, il quale farà in modo che l'ic utilizzi la tensione esterna come referenza.



h

Figura 7.16: Voltage reference circuit

STUDENTSUPSI

Come detto in precedenza, questo componente permette solo un duty cycle fisso, esso infatti una volta impostato sarà sempre quello qualsiasi sia la frequenza di oscillazione dell'oscillatore interno. I possibili valori di duty cycle impostabili in questo integrato sono 4 e si impostano tramite la tensione fornita al pin $Vcontrol$, la tabella 7.2.4 riposta il valore del duty cycle in base alla tensione applicata

Vcontrol	Duty Cycle
0V-0.18Vin	0%
0.22Vin-0.38Vin	50%
0.42Vin-0.58Vin	62.5%
0.62Vin-0.78Vin	75%
0.82Vin-1Vin	87.5%

Per dimensionare i componenti principali di questo convertitore si parte da dei parametri che si hanno a disposizione fin dall'inizio:

- **Vin** Tensione fornita in ingresso, ovvero 5V, ottenuti grazie al convertitore AC/DC utilizzato per alimentare la board.
- **Vout** Tensione voluta in uscita, nel nostro caso 250V,necessari per alimentare le micropompe (vedi cap.(MEMS))
- **Rout** Resistenza di carico, Che in questo caso aveva un valore di circa 32k
- **Duty cycle** Per quanto riguarda il duty cycle, invece si è scelto 87.5% questo perché, come vedremo a breve il componente consente solo 4 valori di duty cycle ed essendo questo il maggiore si è scelto per aumentarne l'efficienza.

Con questi parametri è possibile calcolare la **Fgate** ovvero la frequenza con cui il transistor viene commutato. Per farlo si trova dapprima il fattore **k**, con questa formula:

$$Vout = \frac{Vin}{2} \cdot (1 + \sqrt{1 + \frac{4 \cdot D^2}{k}}) \tag{7.1}$$

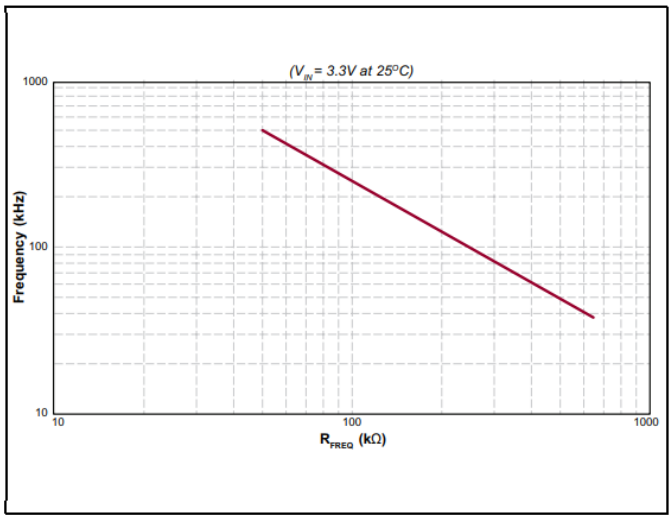
Con le dovute sostituzioni si ottiene

$$k = \frac{Vin^2 \cdot D^2}{Vout \cdot (Vout - Vin)} \tag{7.2}$$

Ora, si sceglie un valore di **L** o induttanza che possa andare bene, solitamente nell'ordine dei μH per questo tipo di applicazioni dopodiché tramite la formula 7.4 si calcola la **Fgate** necessaria per garantire l'uscita voluta con il duty cycle che scelto

$$k = \frac{2 \cdot L \cdot Fgate}{Rout} \tag{7.3}$$

STUDENTSUPSI



h

Figura 7.17: Grafico "Rfreq-Fgate"

$$Fgate = \frac{k \cdot Rout}{2 \cdot L} \tag{7.4}$$

dove:

1. **Fgate** É la frequenza di commutazione del MOSFET
2. **L** É l'induttanza
3. **k** É il fattore calcolato in precedenza
4. **Rout** É la resistenza di carico

trovata la frequenza di gate, si sceglie la **Rfreq** questa resistenza va posizionate sul pin $Freqadj$ e imposta la frequenza dell'oscillatore, la resistenza va scelta in base al grafico 7.17 il quale riporta la curva "Resistenza-Frequenza".
In base a frequenza e induttanza, il circuito varierà il suo assorbimento di corrente è possibile calcolarne il picco e la sua componente nell'induttanza. Ciò consente di adattare il design della board, poiché una corrente molto grande aumenta i consumi ed inoltre potrebbe danneggiare le piste in cui circola se esse non sono adeguate alla sua intensità. Le Formule usate per calcolare la corrente di picco **Ipeak** e la corrente media **IL** sono rispettivamente l'eq.7.5 e la 7.6

$$Ipeak = \frac{Vi \cdot D}{L \cdot Fgate} \tag{7.5}$$

$$I_L = \frac{Vi}{Ron} \cdot [1 - e^{-\frac{D}{Fgate} \cdot \frac{Ron}{L}} \tag{7.6}$$

dove:

1. **IL** E' la corrente che scorre attraverso la bobina

STUDENTSUPSI

2. **Vi** E' la tensione d'ingresso.
3. **Ron** E' la resistenza del MOSFET quando è acceso, ovvero lascia scorrere corrente tra Drain e Source.
4. **D** Duty cycle.
5. **Fgate** La frequenza di gate.
6. **L** L'induttanza scelta.

La tabella 7.2.4 riporta i valori scelti e le modalità impostate per questo progetto.

Mode	LDO
Reference	External
Vin	5 V
Vout	250 V
Duty cycle	87.5%
Fgate	333.8 kHz
L	15 μH
Rfreq	75 kΩ
Ipeak	0.89 A
IL	0.87 A

STUDENTSUPSI

7.2.5 HV513

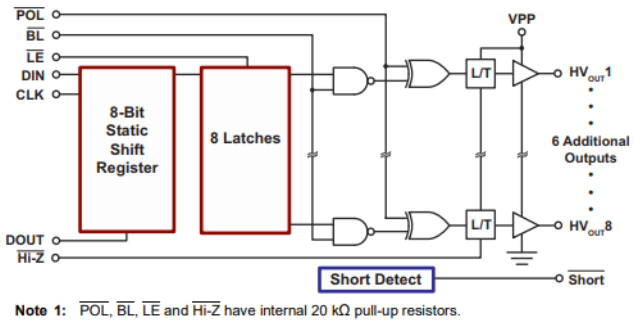


Figura 7.18: HV513 sch

l'HV513 è un "8 channel serial to parallel converter with high-voltge push-pull output", in pratica questo componente è in grado di portare su diverse uscite il segnale fornito in ingresso, aumentandone la tensione fino al valore con cui viene alimentato il componente. Lo schema interno di questo IC è mostrato in figura 7.18, come si può vedere esso è composto da 8 "shift register", i quali trasportano i dati ad ogni fronte di salita del clock, dopodiché i dati vengono spostati nei "latch". A questo punto in base all'uscita abilitata il segnale viene portato alla tensione d'alimentazione e in uscita si ottiene il segnale che si voleva amplificato.
Il componente è contenuto in un package di tipo "32-lead QFN", è composto da 32 piedini, oltre a quelli di alimentazione, sono presenti:

1. **BL** Necessario se si vogliono spegnere tutte le uscite e portarle ad un livello logico "0";
2. **HVOUT0.7** Sono le 8 possibili uscite del segnale ad alta tensione.
3. **POL** Necessario in caso si volesse invertire la logica degli input ovvero renderli "negati"
4. **CLK** Il clock che controlla il progredire dei dati negli shift register
5. **LE** Abilita i latch del componente
6. **SHORT** Questo piedino indica che uno degli output non ha raggiunto lo stato logico desiderato
7. **HI-Z** Se abilitato imposta tutte le uscita ad alta impedenza.
8. **DIN** Data input, ovvero dove vengono trasmessi i segnali che poi verranno "amplificati" e portati in uscita.

STUDENTSUPSI

9. **DOUT** Necessario se si vogliono mettere più HV513 in cascata, questo componente permette lo shift dei dati sul piedino DIN del componente successivo.

La particolarità di questo convertitore è la tensione a cui può pilotare le sue uscite, ciò lo rende ottimo per l'utilizzo di sistemi piezoelettrici, inoltre, viste le sue 8 uscite, permette di pilotarne più di uno in contemporanea, come appunto accade nella board costruita.

STUDENTSUPSI

7.2.6 MP6500GF

Lo scopo di questo IC è controllare il motore passo passo della syringe pump, il componente è un driver per stepper che contiene al suo interno un regolatore di corrente e può fornire 4 diverse modalità di passo, da 1 a 1/8. Può erogare fino a 2.5A di corrente ed ammette un range di tensione che va da 4.5V a 35V.
In figura 7.19 è mostrato il modello interno del driver. Come si può notare il comando del motore è controllato da una logica interna, la quale oltre a gestire i diversi ingressi, controlla la corrente su ciascuno dei due ponti ad H e la commutazione degli 8 transistor. La frequenza con cui lavora la logica è controllata da un timer, a sua volta impostato tramite una resistenza posta all'esterno del componente chiamata R_{osc}. La charge pump interna ha il compito di pilotare i gate dei transistor, che, essendo di potenze richiedono una tensione sufficientemente alta per essere correttamente pilotati. Il componente è composto da 28 pin, oltre a quelli di alimentazione sono presenti:

1. **DIR** Stabilisce in che direzione gira il motore, a dipendenza del livello logico fornitogli.

2. **MS1/MS2** Essi controllano la dimensione del passo del motore, seguendo la tabella

3. **nENBL** Abilità il funzionamento dei componenti interni se portato a '0', altrimenti disabilita l'integrato.

4. **nSLEEP** Se viene riceve un livello logico '0', l'integrato entra in "power safe mode", ovvero disabilita alcune funzionalità per consumare meno.

5. **ISET** In base alla resistenza che viene posta su questo pin, l'integrato imposta la corrente da erogare al motore.

6. **Rosc** La frequenza del generatore PWM interno, viene impostata tramite la resistenza posta su questo piedino.

7. **nFAULT** É un pin di protezione e commuta a '0', se l'integrato rileva un malfunzionamento interno.

8. **AOUT1/2** Sono le uscite del primo ponte ad H interno.

9. **BOUT1/2** Sono le uscite del secondo ponte ad H interno.

10. **CP1/2** Tra questi due piedini va posizionato un condensatore, necessario per il corretto funzionamento della charge pump dell'ic.

STUDENTSUPSI

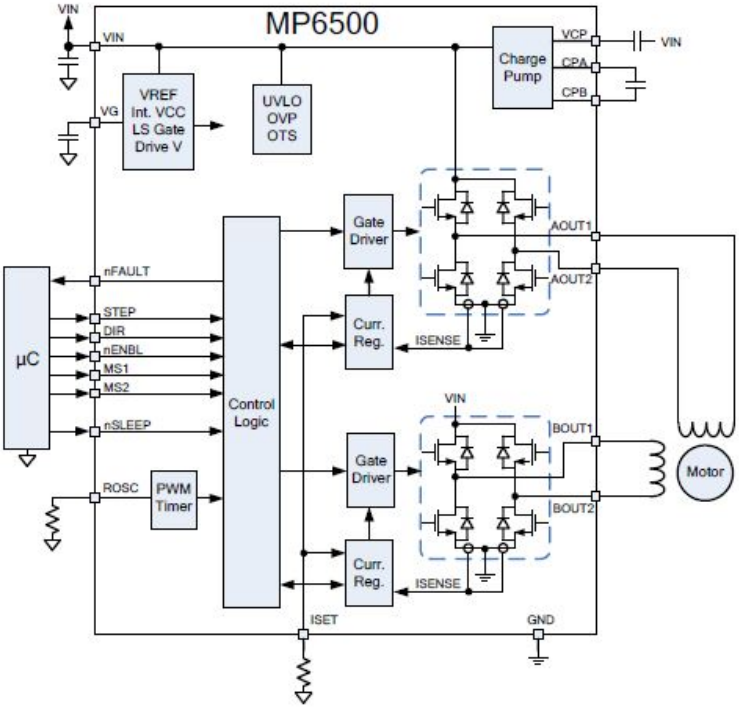


Figura 7.19: MP6500 schema interno

7.2.7 Funzionamento e dimensionamento MP6500

7.2.7.1 Controllo di passo

Per controllare il movimento del motore occorre inviare un fronte di salita al piedino **STEP** dell'integrato , l'impulso inviato deve avere una larghezza di almeno $1\mu s$. Una volta rilevato l'impulso esso viene trasferito al traslatore interno del componente. Questo traslatore facendte parte della logica interna ha il compito di controllare il DAC interno ed il flusso di corrente in ogni avvolgimento del motore.
La quantità di corrente che scorre all'interno di ogni avvolgimento dipende dai due piedini MS1 ed MS2 che impostano la grandezza del passo, maggiore sarà la grandezza, minore la corrente richiesta. La tabella 7.2.7.1 mostra le possibili combinazioni di MS1 ed MS2 e i passi risultanti. La gestione della direzione di rotazione è affidata al piedino **DIR** nel caso di un livello logico alto il motore girerà in senso orario, viceversa in senso antiorario. Prima di impostare passo e direzione la logica interna richiede un "set-up time" di 200 ns.

MS1	MS2	Passo
0	0	Passo intero
0	1	!/2 passo
1	0	1/4 passo
1	1	1/8 passo

STUDENTSUPSI

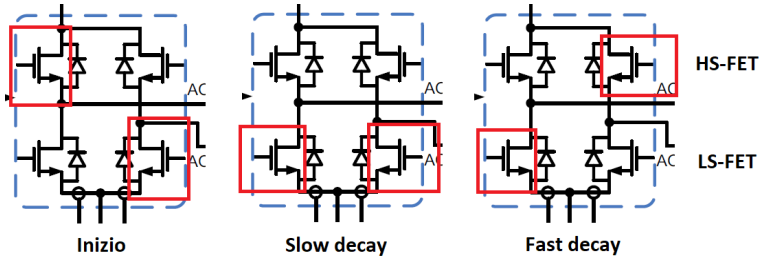


Figura 7.20: In rosso sono riquadrati i MOS-FET accesi

7.2.7.2 Controllo di corrente

La corrente è regolata da una PWM programmabile, appena azionata la PWM per un tempo detto T_{blank} il dispositivo atto a monitorarla viene ignorato questo poiché inizialmente vi sono degli spike di corrente che altrimenti verrebbero segnalati come sovracorrenti, questi fattori impongono un minimo tempo di accensione per la PWM. Una volta passato il transiente iniziale i due MOSFET visibili in figura **??**(inizio) si accendono facendo scorrere una corrente negli avvolgimenti del motore. Quando la corrente raggiunge una determinata soglia pre-impostata, il comparatore interno spegne l'HS-FET, ovvero il MOSFET connesso a Vin. A questo punto ci sono due possibilità o si accendono i due LS-FET, ovvero i due MOSFET connessi a massa, lasciando scaricare l'avvolgimento tramite quelli, oppure, per velocizzare il processo si possono accendere i due MOSFET precedentemente spenti così da lasciar scorrere più Corrente dalla bobina e scaricarla più rapidamente, questa modalità prende il nome di "Fast decay", mentre quella vista in precedenza di "Slow decay", in particolare, il tempo impiegato dal circuito per far scaricare l'avvolgimento da questa modalità è definito "Toff". Superata nuovamente la soglia verranno accesi i MOSFET iniziali e il processo si ripeterà.

La costante "Toff", viene calcolata tramite questa formula:

$$T_{off} = 115 \cdot R_{osc}(k\Omega) \tag{7.7}$$

dove:

- R_{osc} è la resistenza posta sul piedino omonimo, che imposta la frequenza della PWM interna.
- T_{off} è il tempo durante il quale la bobina viene scaricata tramite i due LS-FET.

Da questa equazione si evince che tramite la resistenza R_{osc} è possibile impostare la costante di tempo per lo "Slow decay". Per quanto riguarda la corrente che scorre nel motore, essa va impostata tramite il piedino **ISET**, il quale limita la corrente in uscita dall'integrato

STUDENTSUPSI

in base al valore del resistore che gli viene collegato. La formula 7.2.7.2 riporta il calcolo necessario per la scelta della resistenza:

$$I_{max} = 71(k\Omega) \cdot R_{ISET} \tag{7.8}$$

Inoltre è possibile calcolare anche la corrente su ogni avvolgimento a seconda della grandezza di passo selezionata, la formula7.2.7.2 mostra il calcolo.

$$I_{TRIP} = \%I_{trip} \cdot I_{MAX} \tag{7.9}$$

Dove:

- I_{TRIP} è la corrente nell'avvolgimento.
- %I_{TRIP} è la percentuale di corrente assorbita dall'avvolgimento a seconda del passo utilizzato, la tabella di tutte le percentuali è visibile sul datasheet, che è segnalato nelle referenze di questa tesi.
- I_{MAX} è la corrente massima che può scorrere nel motore, calcolata tramite l'equazione 7.2.7.2.

Questo integrato utilizza un metodo automatico per il controllo del flusso di corrente nella bobina. Subito dopo che la bobina è stata caricata, l'integrato lascia che si scarichi con la modalità lenta "Slow decay". Dopo un tempo T_{OFF}, l'ic verifica che la corrente si trovi al valore della soglia di I_{TRIP}, se ciò non accade l'ic entra in "fast decay" o decadimento veloce, così da garantire una variazione della corrente d'uscita il più rapido possibile. Una volta raggiunta la soglia, il sistema torna nuovamente in "Slow decay". In figura 7.21 è mostrato il funzionamento appena spiegato.

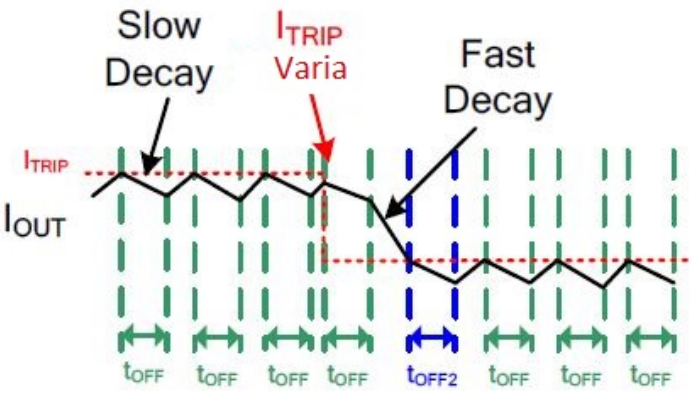


Figura 7.21: Comportamento I_{OUT}in funzione di I_{TRIP}

STUDENTSUPSI

7.2.7.3 Segnalazione di errori

All'interno dell'integrato sono presenti 4 circuiti di protezioni principali, nel caso in cui uno di essi rilevasse un'errore, o un malfunzionamento, il piedino $nFAULT$, viene posto a '0', così da segnalare che un circuito di protezione è entrato in funzione. Il piedino è collegato direttamente con il microcontrollore, il quale se riceve un fronte di salita su quel piedino, interrompe immediatamente il funzionamento dell'ic. I 4 circuiti di protezione sono:

- OCP Quando viene rilevata una sovracorrente nell'ic, il circuito di protezione OCP o "Over-current protection", limita la corrente che scorre nei MOSFET. Per farlo. dapprima disabilita il comando che controlla i gate degli stessi, se questa procedura non diminuisce la corrente entro un certo tempo, detto "deglitch time", il pin $nFAULT$, viene posto a '0' e tutti i MOSFET dei ponti ad H vengono disabilitati. Dopo 2.4 ms vengono riattivati tutti i MOSFET e l'ic torna ad operare normalmente, finché non viene rilevata una nuova sovracorrente. Una sovracorrente può derivare sia dagli avvolgimenti del motore, sia da un corto tra massa e V_{IN} dell'integrato, in entrambi i casi viene attivato il circuto di protezione OCP.

- OVP Se la tensione dovesse superare i 36V, ovvero la tensione massima accettata dall'ic, il ponte ad H viene immediatamente disabilitato ed il piedino $nFAULT$, viene portato a '0'. Non appena la tensione torna sotto la soglia, questa protezione si spegne el'ic torna a funzionare normalmente.

- UVLO Nel caso in cui la tensione V_{IN}, scendesse sotto la soglia minima, ovvero 4.5V, tutti i dispositivi interni dell'ic vengono disattivati ed il piedino $nFAULT$ portato a '0'. Il circuito torna al suo normale funzionamento, una volta che la tensione torna sopra la soglia minima.

- Thermal shudown Questa protezione si attiva quando la temperatura dell'ic supera una determinata soglia. A quel punto tutti i ponti ad H dell'integrato vengono disattivati ed nFAULT, viene portato a '0'.

7.2.8 AP1609

Il compito di questo integrato è quello di traslare la tensione in ingresso di 5V, ad una tensione che consenta di pilotare correttamente il motore della syringe pump. Il componente, è un convertitore step-up DC/DC ad alta efficienza, esso permette una corrente di uscita fino a 2.4A ed accetta una tensione d'ingresso che va da 2.5V a 6V. In figura 7.22 è mostrato lo schema interno del componente, come si può notare al suo interno sono presenti due comparatori, il primo confronta la tensione sul piedino di feedback, con la referenza interna, in modo da correggere sempre l'errore dell'uscita. Il secondo, invece, controlla il gate del MOSFET. Per farlo viene generato internamente un segnale PWM, che viene confrontato

STUDENTSUPSI

con l'uscita del comparatore di feedback, in questo modo esso può regolare l'accensione del MOSFET, in base all'andamento dell'uscita. In figura 7.23 è mostrato il schema elettrico

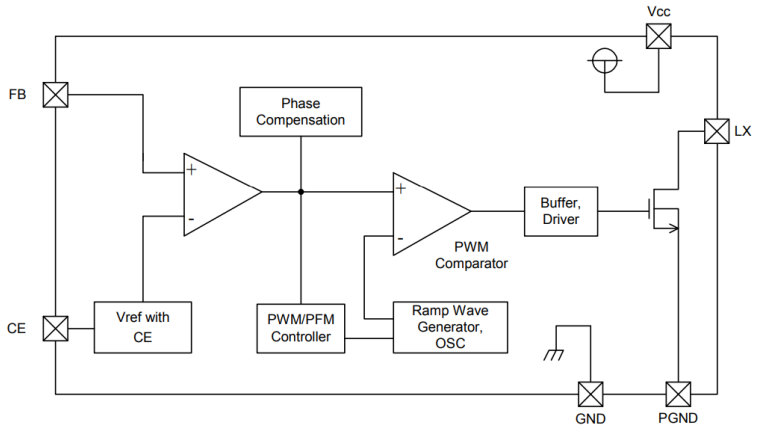


Figura 7.22: Schema elettrico interno AP1609

del componente. La funzione dei diversi piedini è:

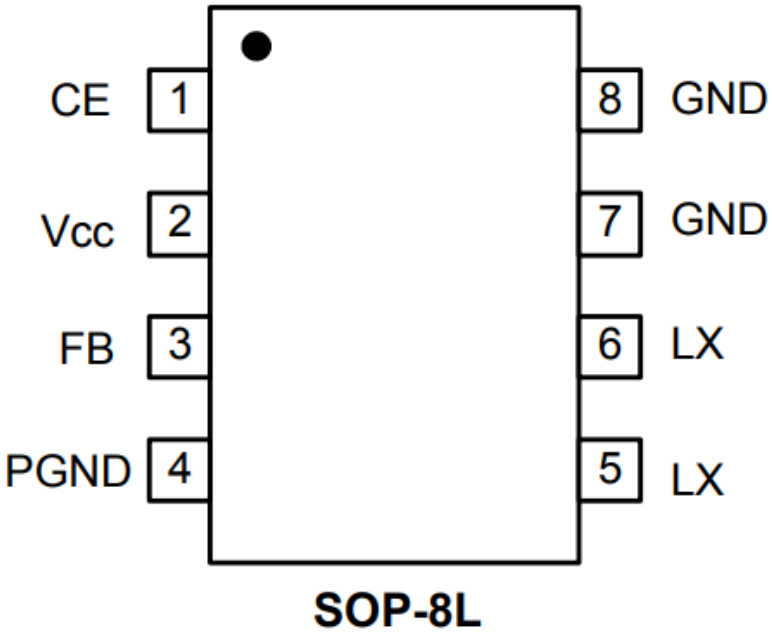


Figura 7.23: Schema elettrico AP1609

- **CE** É il chip eneble, se posto a '1' abilità i circuiti interni.
- **Vcc** Pin di alimentazione.
- **FB** É in pin di feedback, permette la regolazione della tensione d'uscita

STUDENTSUPSI

- **PGND** É il pin di massa del MOSFET interno.
- **LX** Pin di switch, l'induttore ed il diodo vanno connessi a questo pin
- **GND** Pin di massa.

7.2.9 Dimensionamento AP1609

In base all'alimentazione fornita, questo integrato consiglia due tipi di configurazioni differenti.
Nel caso in cui l'alimentazione fosse inferiore a 5.5V, viene utilizzato il circuito in figura 7.24. La configurazione è la tipica di un DC/DC boost. Il funzionamento è molto simile a quello visto per l'alimentazione delle micropompe tramite l'HV513.
Prima fase Nella prima fase, il MOSFET interno al componente è aperto e lascia scorrere una corrente al suo interno per un certo tempo xT, durante questo tempo la bobina si carica con una corrente pari a :

$$\Delta I_L = \frac{V_I}{L} \cdot xT \tag{7.10}$$

Seconda fase Dopo un tempo xT il transistor interno viene spento, la bobina che si era caricata in precedenza, si scarica attraverso il diodo, mantenendo l'uscita al valore desiderato e caricando i condensatori. La corrente che scorre nella bobina in questa fase vale:

$$\Delta I_L = \frac{V_O - V_I}{L}(T - xT) \tag{7.11}$$

Sapendo inoltre, che la corrente che scorre nella bobina deve essere uguale in entrambe le fasi appena descritte (quando a regime), risulta che:

$$\Delta I_L = \frac{V_I}{L} xT = \frac{V_O - V_I}{L}(T - xT) \tag{7.12}$$

da cui:

$$V_I \cdot xT = (V_O - V_I) \cdot (T - xT) \tag{7.13}$$

$$xT = \frac{V_O - V_I}{V_I} \cdot (T - xT) \tag{7.14}$$

$$\frac{V_O}{V_I} = \frac{1}{1 - xT} \tag{7.15}$$

l'ultima equazione scritta è la funzione di trasferimento del circuito. Per calcolare l'uscita del prodotta dall'AP1609, si usa la formula 7.16, come si può vedere l'uscita dipende dal rapporto tra le due resistenze R1 ed R2, in particolare, da datasheet è specificato che il valore di quest'ultima sia maggiore di 10kΩ

$$V_{OUT} = 1.20 \cdot (1 + \frac{R1}{R2}) \tag{7.16}$$

STUDENTSUPSI

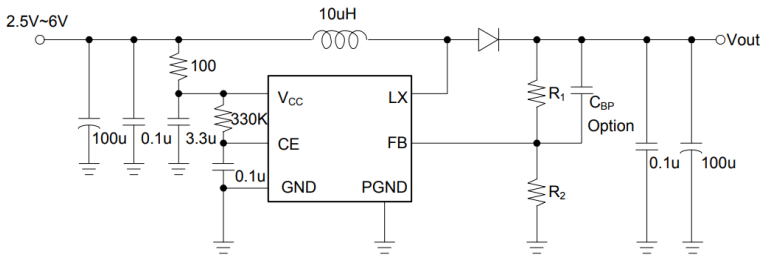


Figura 7.24: Interfacciamento AP1609

Il circuito alternativo che si può utilizzare è quello presente in figura **??**. Tale circuito viene utilizzato nel caso in cui la tensione di ingresso sia superiore a 5.5V, che è la tensione massima fornibile in ingresso all'integrato. Per questa ragione viene messo un diodo zener davanti al piedino d'ingresso dell'alimentazione dell'ic, lo scopo del componente è portare la tensione ad un livello che il

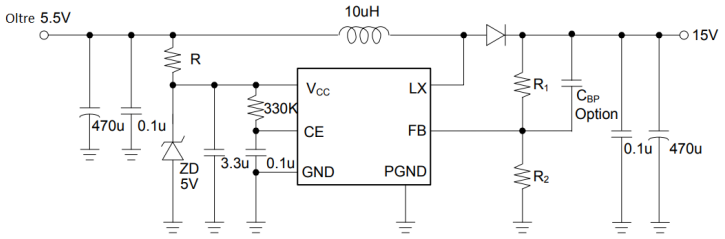


Figura 7.25: circuito AP1609 per alte tensioni

STUDENTSUPSI

7.3 Board di controllo

7.3.1 schematici

Questa scheda ha il compito di gestire tutte le periferiche facenti parte del progetto, il che comprende, tutti i diversi segnali necessari alla board di alimentazione per funzionare correttamente, i sensori usati per il controllo di flusso e pressione dei liquidi, il display e i diversi pulsanti utilizzati. Lo schematico è stato diviso in molteplici fogli al fine di risultare il più ordinato possibile. La sua alimentazione arriva da un connettore, collegato direttamente con la board di alimentazione, questo connettore ha anche il compito di trasmettere i comandi ai diversi IC presenti sulla BdA.

STUDENTSUPSI

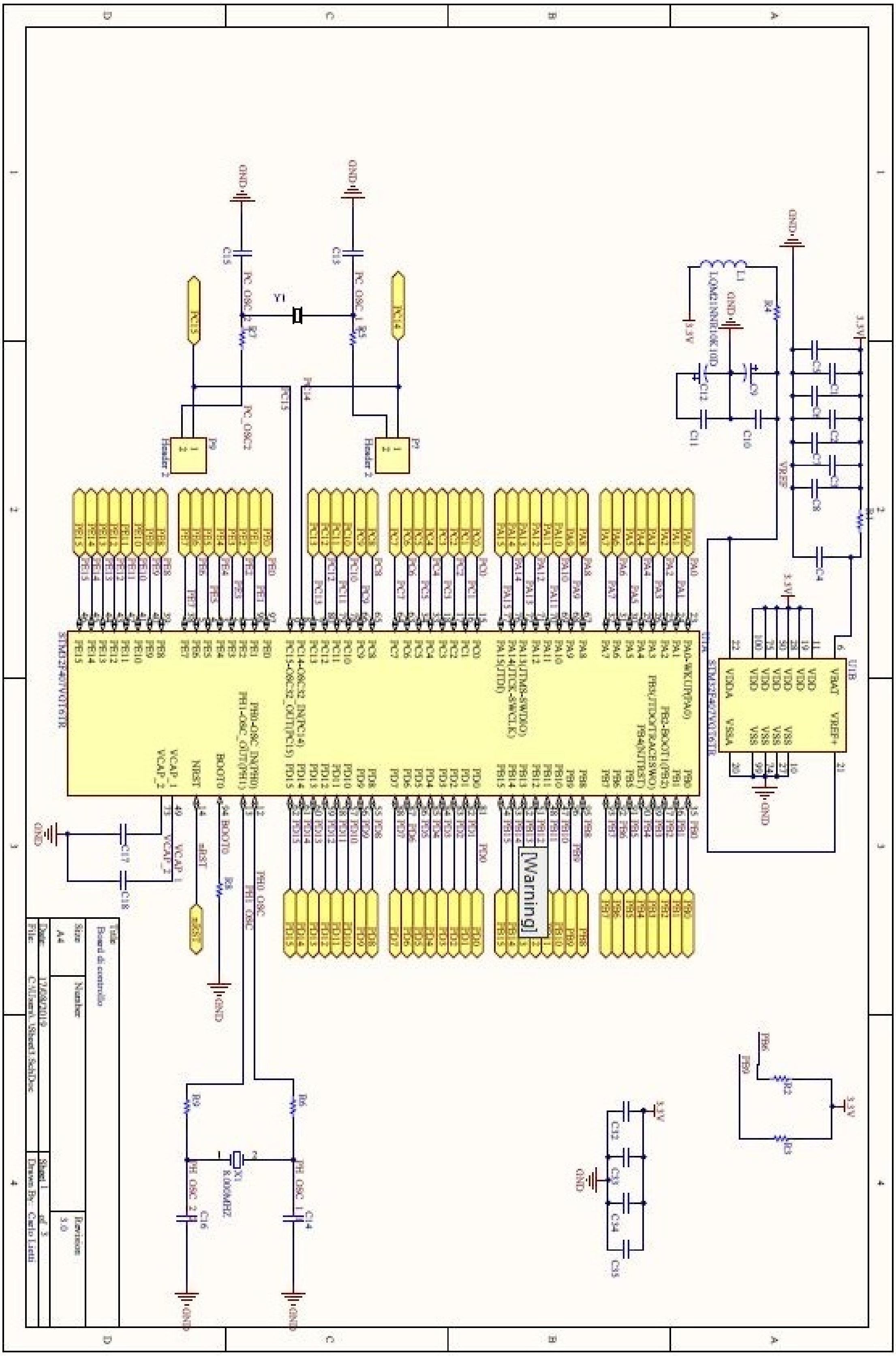


Figura 7.26: Foglio n.1 board controllo

STUDENTSUPSI

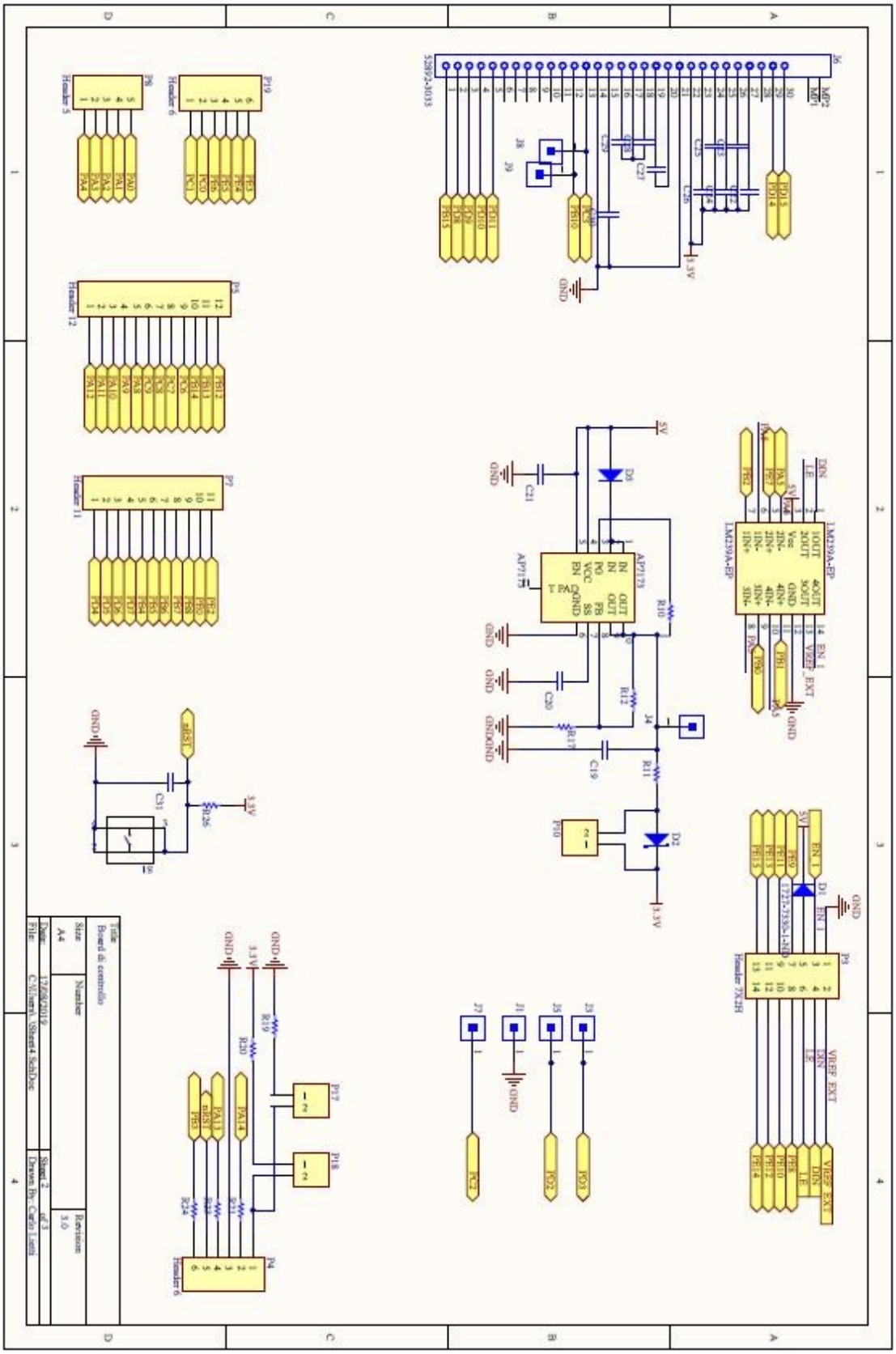


Figura 7.27: Foglio n.2 board controllo

STUDENTSUPSI

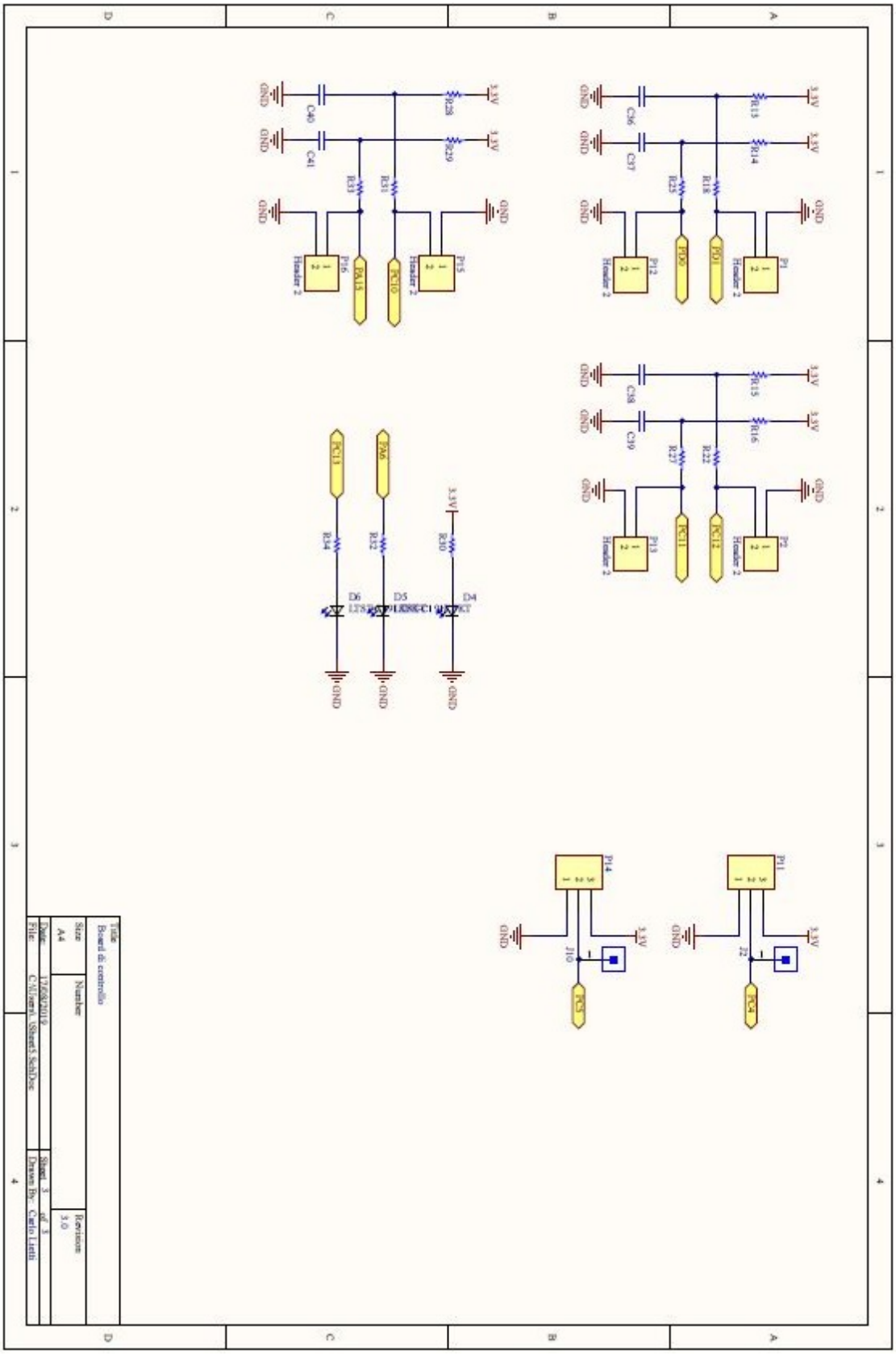


Figura 7.28: Foglio n.3 board controllo

STUDENTSUPSI

7.3.1.1 Foglio 1 BdC

Il primo foglio 7.26 mostra il microcontrollore ed i componenti necessari per il suo corretto funzionamento. Il microcontrollore scelto è il **STM32F407VG**, prodotto da "ST Microelectronics" è stato scelto per l'ottima quantità di I/O disponibili e di periferiche interne, esso verrà analizzato più nel dettaglio successivamente. Nello schema sono presenti diversi filtri per l'alimentazione,sia generale che del micro, essi sono molto utili in quanto riducono molto il rumore che si crea, sopratutto se l'alimentazione arriva da un'altra board. L'induttore L1 viene usato anch'esso come filtro, il suo obiettivo è quello di limitare l'emissione di onde EMF.

7.3.1.2 Foglio 2 BdC

Lo schematico 7.27 ha diversi scopi;

- Trasformare la tensione di alimentazione da 5V a 3.3V, poiché il microcontrollore può lavorare solamente tra 1.7V-3.6V, rendendo, di fatto, impossibile l'utilizzo di 5V come alimentazione diretta. Per fare ciò il circuito si avvale di un convertitore lineare o LDO, precisamente l'"AP7173" prodotto dalla "Diodes inc.".
- Portare ad una tensione sufficiente l'uscita del micro, affinché i componenti **HV513** ed **HV9150** siano in grado di interpretare i fronti di salita e quelli di discesa. Normalmente per compiere questo tipo di operazioni viene usato un componente definito "level shifter", tuttavia non ne sono stati trovati di adatti per i livelli di tensione applicati nel circuito. Perciò, è stato deciso di usare un comparatore, che, alimentato tramite la tensione d'ingresso nella scheda(5V), è in grado, di aumentare il livello di uscita dal micro. Il componente utilizzato per questa operazione è **LM239A-EP**
- Rendere possibile il collegamento di un display, tramite il connettore "J6", il quale è un particolare connettore "Molex".
- Rendere disponibili diverse uscite del microcontrollore all'utente tramite degli header, infatti, come si può vedere nella parte bassa, molti piedini del micro, sono stati collegati a degli header di varie dimensioni, cosicché nel caso in cui si volessero espandere le capacità della board, aggiungendo periferiche, sarebbe sufficiente collegarsi ai connettori scelti.

7.3.1.3 Foglio 3 BdC

L'ultimo schematico presenta:

- I circuiti utilizzati per interfacciare 6 pulsanti con la scheda, in particolare essi sono strutturati per essere anti-rimbalzo, ovvero, evitano che nel momento in cui viene premuto un pulsante il microcontrollore, rilevi non un solo cambio di tensione, ma multipli ,

STUDENTSUPSI

come accade in 7.29a. In tal modo se programmato per eseguire una certa routine potrebbe ripeterla decine di volte, rischiando di compromettere il corretto funzionamento del sistema.

(a) Segnale senza anti-rimbalzo (b) Segnale con anti-rimbalzo

Figura 7.29: Differenza tra due segnali prodotti da un pulsante, con o senza anti-rimbalzo

- I connettori per il collegamento dei sensori di pressione e flusso (P14 e P11). I sensori usati sono i **MP3V5010DP**, prodotti dalla "NXP", questi sensori sono in grado di misurare umidità e pressione, ciononostante, come vedremo più avanti possono essere usati anche per la misura di flusso di un liquido.

- Dei led di stato usati per verificare il corretto funzionamento dell'alimentazione(D5) e per segnalare determinati errori via software(D4, D3).

7.3.2 PCB BdC

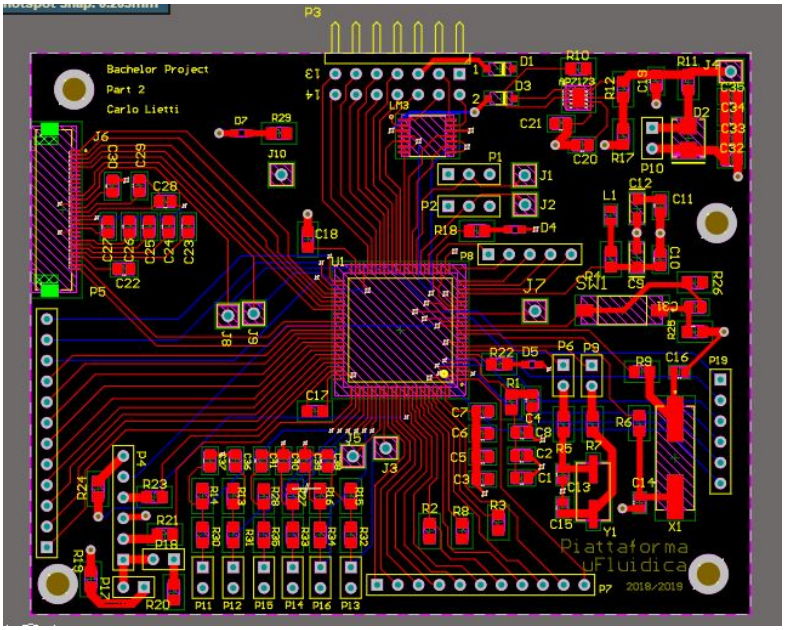


Figura 7.30: Pcb Board di controllo

La sezione superiore di questa board, di cui si può vedere una versione ingrandita in figura 7.34 si occupa di gestire l'alimentazione. Per farlo trasforma la tensione ricevuta in ingresso

STUDENTSUPSI

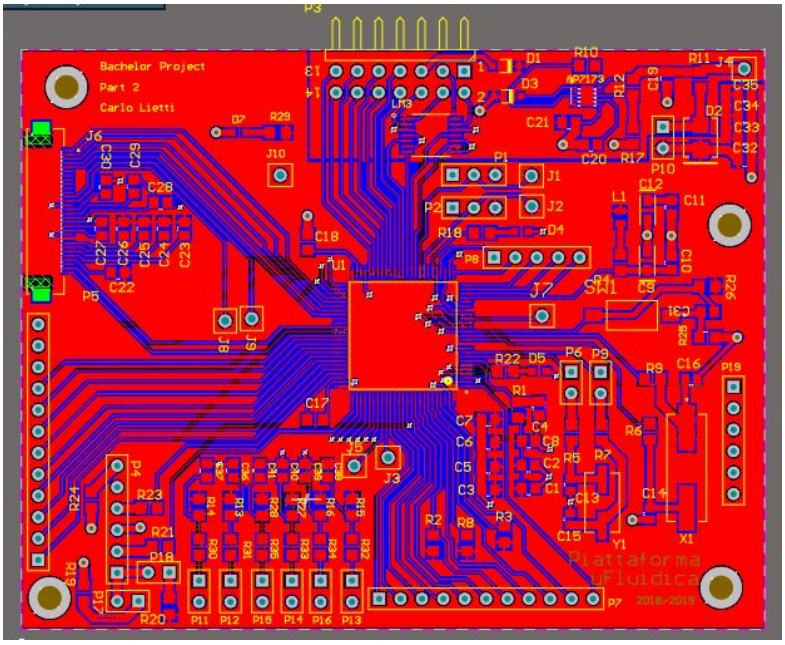


Figura 7.31: Pcb Board di controllo, TOP layer

dall'header (connesso con la board di alimentazione) di 5V, in 3.3V, tramite un LDO **AP7173**.

STUDENTSUPSI

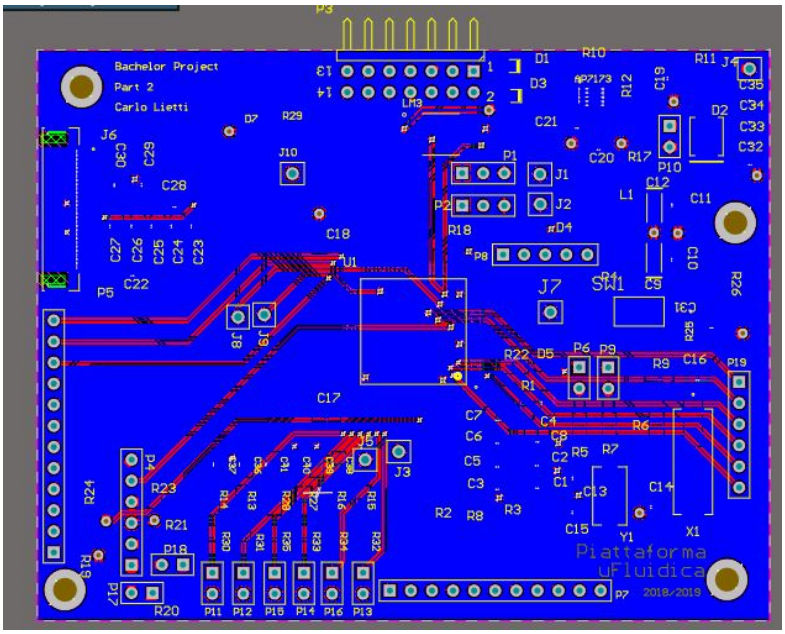


Figura 7.32: Pcb Board di controllo, BOTTOM layer

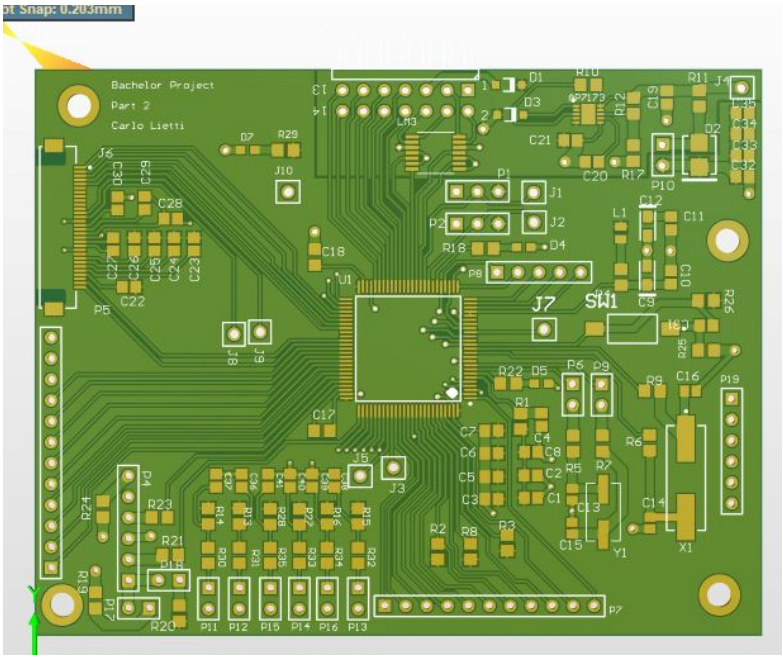


Figura 7.33: Pcb Board di controllo, vista 3d

STUDENTSUPSI

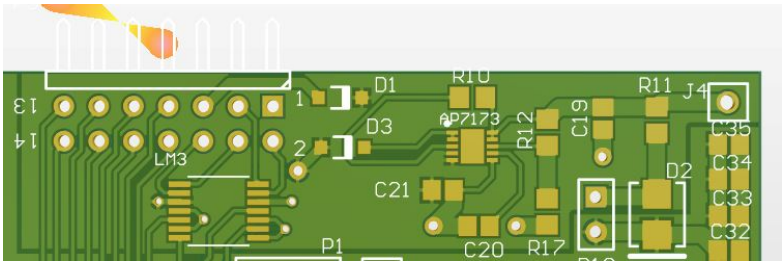
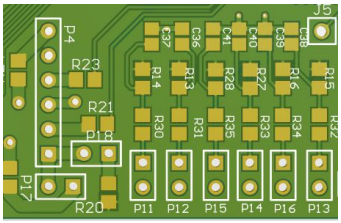


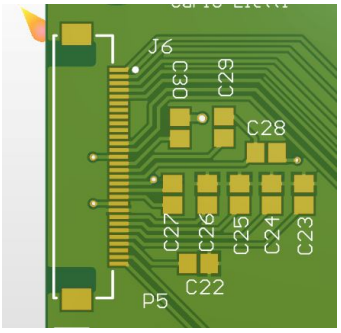
Figura 7.34: Alimentazione board di controllo

STUDENTSUPSI

In mezzo alla board è posizionato il microcontrollore, così da semplificare i collegamenti con il resto della board. Nella sezione in visibile in figura 7.35a è posizionato il connettore utilizzato per la programmazione del microcontrollore, mentre di fianco sono stati previsti gli header per i pulsanti. Nell'immagine 7.35b viene mostrato il per lcd, con le piazzuole[10] per i condensatori necessari per il suo funzionamento.



(a) Connettore per la programmazione e pulsanti



(b) Connettore per la connessione del display

Figura 7.35: Connettori per programmazione e pulsanti a sinistra, connettore per il display a destra

7.3.3 STM32F407VG

Il controllo di tutto il sistema, è stato affidato, come comunemente si fa in questi casi, ad un microcontrollore. La ragione per la quale è stato usato questo microcontrollore piuttosto che altri, è la grande quantità di I/O disponibili e le numerose periferiche interne, inoltre, la disponibilità di un evaluation board proprio per questo micro, ha permesso di semplificare molto la fase di programmazione. Le caratteristiche principali di questa MCU sono:

- Basato su un'architettura "ARM Cortex-M4", questa architettura garantisce un'ottima efficienza combinata con un numero piuttosto cospicuo di istruzioni disponibili ed una discreta capacità di effettuare "signal processing".
- Una memoria flash di 1 Mbyte, più 196 Kbyte di RAM statica, il cui scopo è salvare tutti i dati volatili mantenendo il loro valore anche nel momento in cui venisse spento il clock del dispositivo.
- Possibilità di trasmette con molteplici protocolli, tra i quali: SPI, I^2C, CANBUS, USART ed UART.
- 3 moduli ADC multicanale, separati.
- Un DAC interno a due canali.

[10] Aree predisposte su un circuito stampato per saldare i terminali di un componente

STUDENTSUPSI

- Possibilità di avere fino a 14 timer, alcuni dei quali con 4 canali, attivi in contemporanea.

7.3.3.1 Programmazione

Per programmare questo micro viene utilizzato il protocollo **ST-LINK V2** è un protocollo proprietario della "STmicroelectronics", questo protocollo consente tramite debugger[11], Di testare il proprio programma in "run-time", ovvero mentre esso sta girando sul microcontrollore usato. La connessione fisica utilizzata per la programmazione del microcontrollore è mostrata in figura 7.36, per funzionare ha bisogno di 6 pin, nell'ordine:

1. Vdd dal micro che si vuole programmare
2. SWCLK (PA14 nel microcontrollore)
3. GND
4. SWDIO (PA13 nel microcontrollore)
5. NRST (Reset del microcontrollore)
6. SWO (PB3 nel microcontrollore)

7.3.4 AP7173

Come detto in precedenza, questo componente abbassa la tensione in ingresso alla board, ovvero 5V, ad una tensione che possa alimentare il micro, in questo caso 3.3V. Il componente scelto è particolarmente indicato per alimentazione di MPU ed FPGA, inoltre è possibile programmare un "soft start" che diminuisce la corrente di accensione e quindi l'impatto che può avere sui device collegati. Per garantire una tensione stabile in uscita LDO si avvale di un sistema con retroazione, come si può vedere in figura 7.37, che mostra il modello interno del componente. Il MOSFET interno viene pilotato da un driver, il quale controlla, regolando la sua tensione il gate del transistor. Tramite questo controllo viene regolata la corrente in uscita. Contemporaneamente, il comparatore collegato al piedino FB verifica che la tensione prodotta in uscita non vado sotto la soglia di riferimento interna, in caso questo succedesse il driver del MOSFET aumenterebbe la corrente che scorre nel transistor, riportando l'uscita ad un valore corretto. Il secondo comparatore ha una funzione diversa, serve per pilotare l'uscita PG che può essere usata per segnalare che il dispositivo non sta funzionando come dovrebbe. l'AP7173 possiede 8 piedini:

[11] Un debugger in informatica è un programma/software specificatamente progettato per l'analisi e l'eliminazione dei bug (debugging), ovvero errori di programmazione interni al codice di altri programmi. Assieme al compilatore è fra i più importanti strumenti di sviluppo a disposizione di un programmatore, spesso compreso all'interno di un ambiente integrato di sviluppo (IDE), in quanto in grado di aiutare il programmatore ad individuare errori di semantica all'interno del codice sorgente del programma, altrimenti di difficile individuazione in fase di runtime.

STUDENTSUPSI

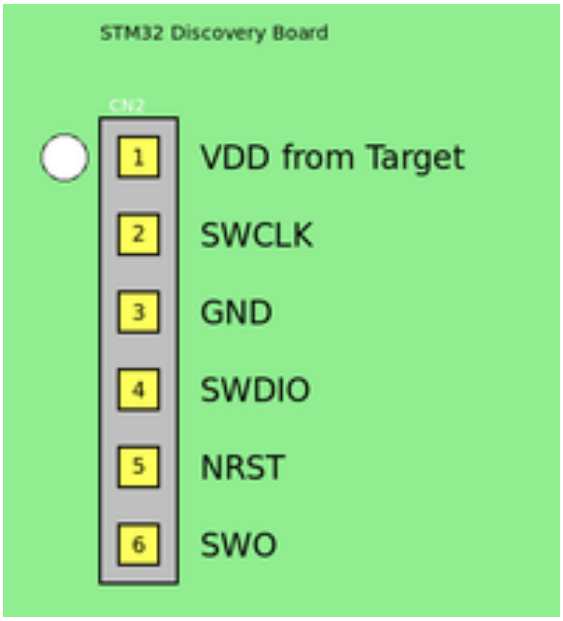


Figura 7.36: Connessione SWD

1. **IN** Tensione d'ingresso
2. **PG** Già citato in precedenza, segnala anomalie nell'uscita.
3. **VCC** Alimentazione.
4. **EN** Abilita il regolatore interno.
5. **GND** Ground.
6. **SS** Piedino di "Soft-Start", in base al condensatore connesso ad esso varia la pendenza del "Soft-start"
7. **FB** É il piedino di feedback, il partitore resitivo posto su questo pin, imposta la tensione di uscita
8. **OUT** Piedino d'uscita

7.3.5 Dimensionamento AP7173

7.3.5.1 Tensione d'uscita

In figura 7.38 si può vedere il tipico circuito di funzionamento per questo LDO, per impostare l'uscita ad un valore voluto si usa la formula 7.17

$$V_{OUT} = V_{REF} \cdot (1 + \frac{R1}{R2}) \tag{7.17}$$

STUDENTSUPSI

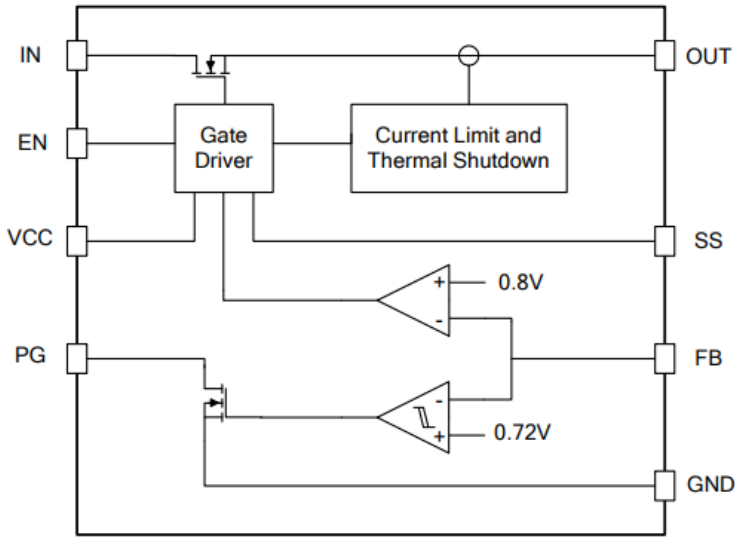


Figura 7.37: Schema interno LDO

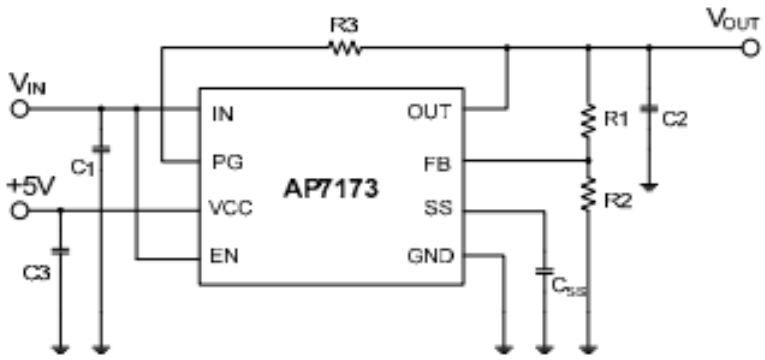


Figura 7.38: Applicazione tipica AP7173

da cui deriva

$$\frac{R1}{R2} = \frac{V_{OUT}}{V_{REF}} - 1 \tag{7.18}$$

dove:

- V_{OUT} è la tensione di uscita.
- V_{REF} è la tensione di riferimento del comparatore interno, ovvero 0.8V.
- R1,R2 sono le due resistenze che compongono il partitore resistivo

Le resistenze devono essere scelte con un valore $\leq$ a 5KΩ per garantire una corrente sufficiente nell'ingresso di feedback.

STUDENTSUPSI

7.3.5.2 Condensatori di input/output

Per un dispositivo come quello utilizzato, è molto importante che la tensione fornita in ingresso sia il più possibile libera a disturbi di ogni genere. Per questa ragione un condensatore di almeno 1μF va posizionato tra il pin di VCC e GND. Per quanto riguarda l'uscita, è un po' diversa la scelta, infatti, se l'alimentazione e l'ingresso IN sono forniti da due generatori diversi è possibile porre in uscita un semplice condensatore da 1μF, mentre invece se gli ingressi sono potati dalla stessa sorgente, è necessario aumentare questa capacità fino ad almeno 4.7μF. Nella board costruita è stata usata la seconda opzione, poiché la tensione fornita in ingresso era unica.

7.3.5.3 Soft-start

La figura 7.39, mostra come varia il transiente della tensione d'uscita, a seconda del valore del condensatore C_{SS}, che si sceglie. Come si può notare, all'aumentare della capacità del condensatore, aumenta anche il tempo che l'uscita impiega a raggiungere il valore desiderato. Per calcolare il tempo impiegato dall'integrato per raggiungere le tensione d'uscita impostata, si usa la formula 7.19

$$t_{SS} = \frac{(V_{REF} \cdot C_{SS})}{I_{SS}} \tag{7.19}$$

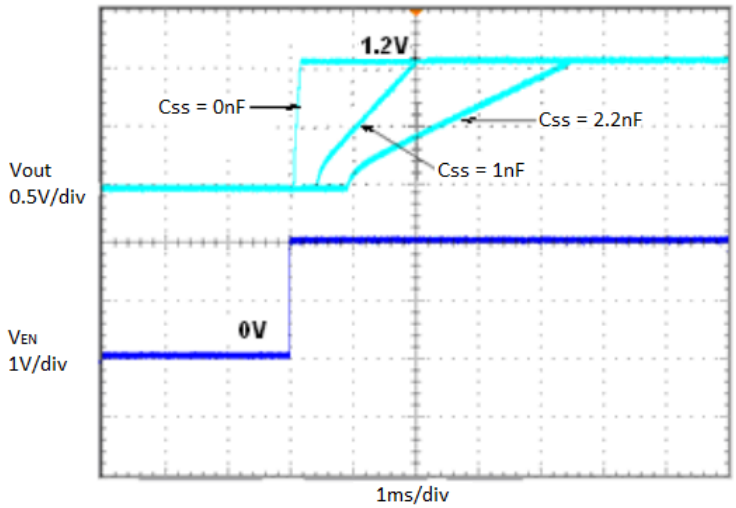


Figura 7.39: Il grafico mostra come varia la tensione di uscita V_{OUT}, a seconda del condensatore posto sul piedino SS

STUDENTSUPSI

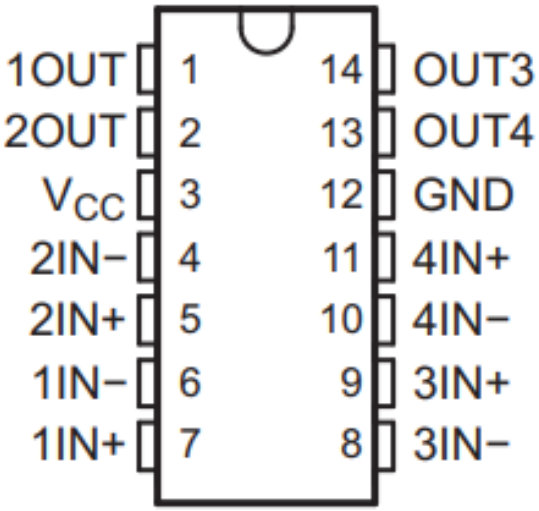


Figura 7.40: Comparatore utilizzato

7.3.6 LM239A-EP

Questo dispositivo è composto da 4 comparatori indipendenti che sono costruiti per funzionare con una singola alimentazione ed un range di tensione molto ampio. Il suo scopo nel circuito è quello di portare i livelli di tensione in uscita da alcuni piedini I/O del micro ad un valore leggibile dagli integrati presenti sulla BdA. Per farlo viene sfruttato il DAC interno del microcontrollore, esso viene impostato ad un valore di circa 2V e viene posizionato sull'ingresso '-' di ogni comparatore. Agli ingressi '+', invece, vengono collegate 4 uscita I/O del micro. In questo modo l'uscita di ogni comparatore sarà sempre al valore 0V, nel momento in cui un piedino del micro viene acceso, il comparatore verificherà quale delle due soglie è maggiore e se è quella positiva, l'uscita cambierà ed andrà al valore con il quale è alimentato l'ic, ovvero 5V.

STUDENTSUPSI

Capitolo 8

Interfaccia utente

Il compito di questa sezione è di fornire un feedback all'operatore e permettergli di interagire con il sistema. Per mostrare i dati e le informazione viene utilizzato un display LCD, mentre per comandare la struttura dei pulsanti

8.1 Il display

Per mostrare il livello di avanzamento, o i dati del sensore all'operatore viene usato un display LCD prodotto da "New Haven", esso è visibile in figura 8.1. I motivi per cui si è scelto questo display a dispetto di altri sono principalmente 3: la tensione di lavoro, il tipo di interfaccia ed il prezzo.

8.1.1 Tensione di lavoro

Il display scelto, opera con una tensione di 3.3V, la ragione che rende così importante questa scelta è da ricercare nel microcontrollore che lo gestisce. Esso, infatti, lavora con una tensione che va da 1.7V a 3.5V, perciò la massima uscita fornibile da un suo piedino di GPIO, è di 3.5V. In integrati alimentati a 5V (ovvero la seconda tensione disponibile nel circuito) la soglia che stabilisce un '1' logico è solitamente di 3.5V, siccome si è scelto di alimentare il micro a 3.3V, distinguere un '1' logico per un display a 5V non sarebbe possibile. Per questa ragione è stato usato questo display. Un'altra possibile soluzione era utilizzare un "level shifter", cioè un componente in grado di traslare dei livelli logici così da farli leggere ad integrati alimentati da tensioni più alte. Non si è scelta questa alternativa per semplificare la circuiteria.

8.1.2 Tipo di interfaccia

Grazie al suo driver interno **ST7565R**, il display permette di utilizzare due modalità di controllo diverse: parallelo e seriale. La figura 8.2a mostra la prima possibilità di collegamento, in

STUDENTSUPSI

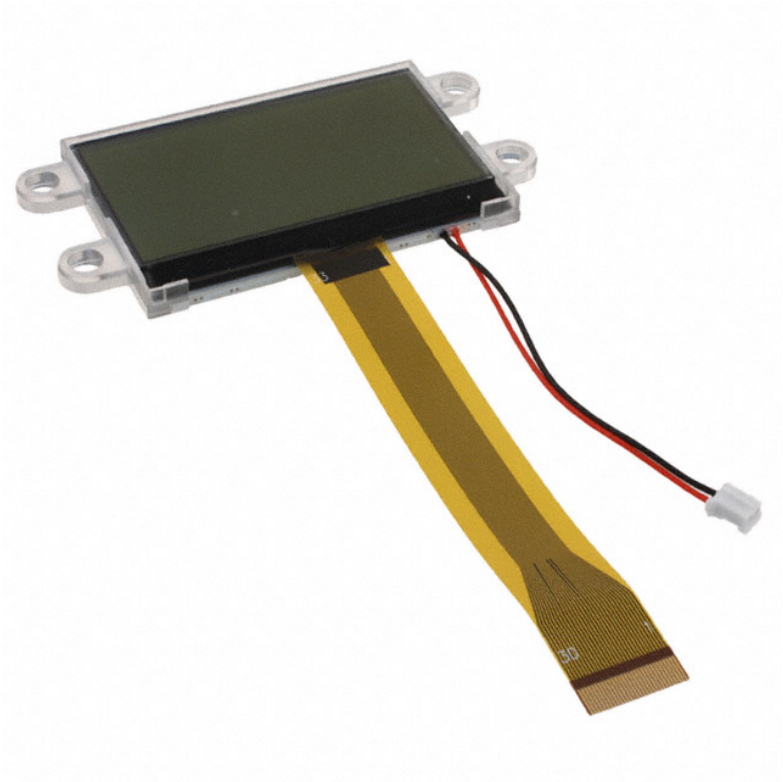


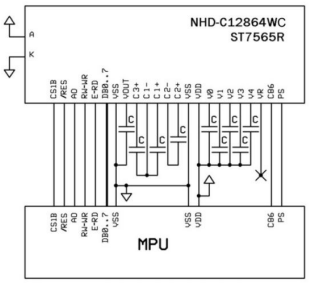
Figura 8.1: Display LCD usato

questo caso 8 pin del display vengono connessi ed ognuno di questi trasporta un bit di controllo, tali bit uniti compongono il dato o il comando. Il sistema di comunicazione in parallelo offre, inoltre, la possibilità di ricevere dei dati dal controllore interno, tali dati permettono di comprendere meglio il funzionamento del display ed aumentarne le prestazioni. Il maggiore svantaggio portato da questa configurazione è il numero di porte I/O richieste, infatti, oltre agli 8 pin di invio dati, ci sono anche altri 6 pin di controllo del display. Per questa ragione, si è scelto di utilizzare la configurazione seriale, visibile in figura 8.2b, al contrario dell'altra, utilizza solamente due piedini per gestire inviare comandi e dati, riducendo notevolmente il numero di connessioni necessarie.

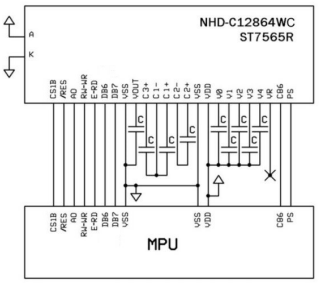
8.1.3 Il prezzo

Su questo aspetto non c'è molto da dire, siccome il budget per il progetto era limitato, si è cercato un display che permettesse di mostrare in maniera chiara i dati e che non costasse eccessivamente.

STUDENTSUPSI



(a) Schema di collegamento display con interfaccia parallela



(b) schema di collegamento display con interfaccia seriale

Figura 8.2: Le due immagini mostrano i due possibili sistemi di interfacciamento del display

8.1.4 ST7565R

Il driver di questo display viene prodotto dalla ditta "Stronix", in figura 8.3 è mostrato il diagramma a blocchi della sua struttura interna. Esso è composto da un chip principale che contiene 65x132 bit di display RAM, ciò consente di controllare singolarmente tramite due driver interni ogni pixel del display, fornendo un grado di libertà molto ampio. Sulla sinistra delle figura si trova il blocco di gestione delle alimentazioni, al suo interno, oltre ai moduli che alimentano i chip principali si trova anche una "pompa di carica" o "charge pump" che viene utilizzata per fornire l'alimentazione necessaria a luce del display per la retro-illuminazione. Sulla destra, invece, sono presenti dei blocchi atti al controllo del clock interno del driver. Infine nella parte inferiore del disegno è presente il blocco che gestisce la comunicazione con il microcontrollore esterno. É importante sottolineare, che questo disegno è relativo ad un driver "ST7565R" generico, nel caso del display utilizzato, molte delle porte mostrate non sono presenti, poiché già impostate dal costruttore. Per questo motivo verrà analizzato solo il comportamento dei piedini utilizzati nel display e non di tutti quelli presenti sul driver.

STUDENTSUPSI

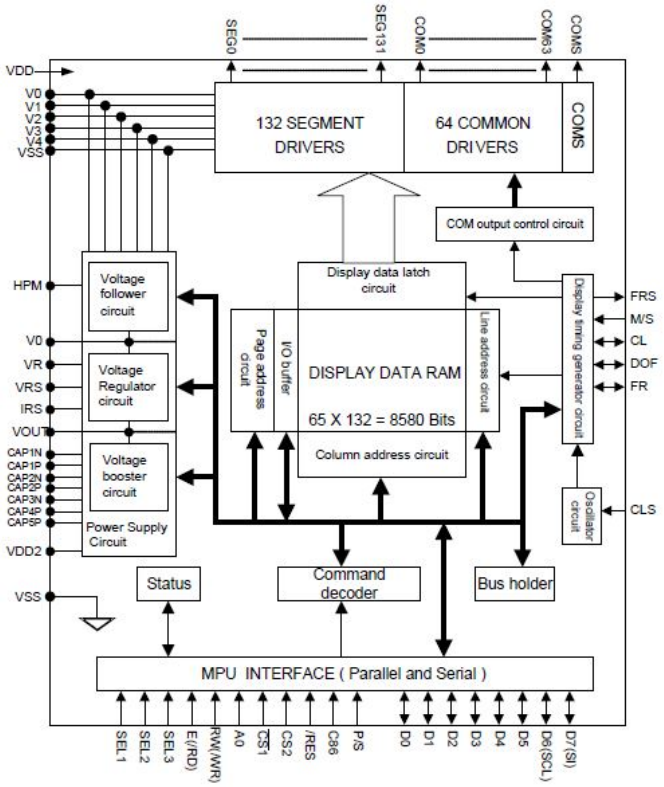


Figura 8.3: Diagramma interno ST7565R

8.1.4.1 Connettività

Come detto in precedenza il driver del display permette di essere comandato tramite due diversi tipi di comunicazione, parallelo e seriale.
A seconda della modalità scelta, sono necessarie determinate connessioni piuttosto che altre, ma non solo, infatti, cambia anche la funzione dei pin e la codifica dei comandi forniti al driver.
I piedini del display e le loro relative funzioni sono riportati nella tabella8.1.4.1

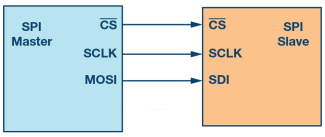
STUDENTSUPSI

CS1B	É il chip select se portato a ’0’ viene abilitata la ricezione di comandi dati
/RES	Segnale di reset, attivo basso
A0	Seleziona il regsistro interno in cui inviare i dati. ’0’ = Comando, ’1’ = Dato
R/W /WR	Se viene utilizzata la modalità 6800: ’0’ Scrittura, ’1’ Lettura. Se viene utilizzata la 8080: Segnale di scrittura, attivo basso
E /RD	É In modalità 6800, se portato ad ’1’ abilità ingressi al data bus. In modalità 8080, portarlo a ’0’ abilita la lettura dello stato interno.
DB0..DB7	In interfaccia parallela, vengono collegati direttamente al data bus e inviano comandi e dati. In interfaccia seriale vengono utilizzati solo gli ultimi 2 DB6 ’Clock e DB7 ’Data’
VR	’E un pin utilizzato per il test dell’LCD
C86	Seleziona il tipo di modalità ’0’ = 8080, ’1’ = 6800
PS	Seleziona il tipo di interfaccia, ’0’ = seriale, ’1’ = parallela

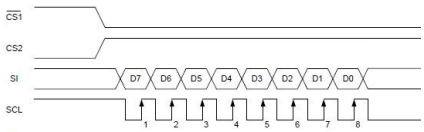
Inoltre sono presenti vari piedini relativi all’alimentazione del display.

8.1.4.2 Seriale

Quando il piedino PS = ’0’, viene abilitata l’interfaccia di comunicazione seriale. In questo caso il display comunica al micro tramite il protocollo SPI. In questo caso viene configurato come "half-duplex" figura 8.4a, ovvero la comunicazione avviene in un solo senso, dal microcontrollore verso il display. Questo tipo di protocollo si basa sull’invio sincronizzato di dati, tra un dispositivo "Master" ed un’altro "Slave". Il "Master", in questo caso il microcontrollore, avvisa il display che è pronto per inviargli dati, portando il valore del "chip select" a ’0’. A questo punto il master attiva il clock, contemporaneamente invia i dati, essi vengono interpretati ad ogni fronte di alita, perciò per trasmettere un ’1’, il microcontrollore, dovrà portare ad ’1’ il pin dei dati o "MOSI", prima che si presenti l’impulso di clock corrispondente alla posizione dove l’1’ vuole essere trasmesso, la figura 8.4b, mostra il concetto appena spiegato.



(a) Connessione SPI half-duplex



(b) schema di collegamento display con interfaccia seriale

Figura 8.4: a sinistra è mostrato il collegamento per il protocollo SPI half-duplex, mentre a destra è mostrato il diagramma dei tempi per la comunicazione tra MPU e diplay

STUDENTSUPSI

8.1.4.3 Parallelo

Se il piedino PS viene posto ad '1', si utilizza l'interfaccia parallela. Questa interfaccia da la possibilità di utilizzare due modalità di tipo diverso

- Modalità 6800, il piedino 'R/W', indica se si vuole scrivere o leggere in memoria, portandolo rispettivamente a '0' o '1', mentre, il piedino 'E' abilità gli ingressi del LCD.
- Modalità 8080, viene utilizzato un piedino '/WR', che portato a '0' indica che si vuole scrivere in memoria, mentre un piedino 'RD', se portato a '0' indica che si vuole leggere in memoria. I due piedini non possono essere posti a '0' nello stesso tempo.

Contrariamente all'altra, questa modalità non si avvale di un clock per sincronizzare la comunicazione, bensì dati e comandi vengono trasferiti quando il piedino 'E' è portato ad '1', nel caso della modalità 6800, oppure nel caso della modalità 8080, quando il pin /WR o /RD (a seconda dell'operazione) viene portato ad '0'. Rispetto all'altra, questa interfaccia permette di leggere dal display, oltre che inviare dati. Questa possibilità permette di non salvare all'interno della RAM del MPU un buffer che tenga in memoria la configurazione dei pixel del display, vengono risparmiate risorse del micro e diventa possibile pilotare il display anche tramite integrati con specifiche basse.

8.1.5 Comandi

In base alla combinazione di bit inviati in ingresso al display, si può inviare un comando, piuttosto che un'altro. Oltre alla giusta combinazione di bit da inviare, è necessario impostare correttamente anche dei bit, i quali indicano al display se l'operazione che si sta facendo è una lettura, una scrittura e se si sta inviando un dato o un comando. A seconda dell'interfaccia usata cambiano anche i piedini, da impostare per indicare al display cosa si vuole fare. In figura 8.5 è riportata una tabella che mostra quali piedini vanno impostati e come, a seconda del comando che si vuole dare. Una volta impostati i piedini, si procede ad inviare

Shared	6800 Series	8080 Series		Function
A0	R/W	/RD	/WR	
1	1	0	1	Reads the display data
1	0	1	0	Writes the display data
0	1	0	1	Status read
0	0	1	0	Write control data (command)

Figura 8.5: La tabella mostra i piedini da impostare per indicare l'operazione scelta

il comando effettivo. I comandi possibili sono 22, quelli utilizzati saranno analizzati nella sezione software

STUDENTSUPSI

8.1.6 Input esterni

Figura 8.6: Bottoni utilizzati

Per gestire le operazioni delle diverse periferiche, si utilizzano dei pulsanti 8.6 collegati tramite un circuito anti-rimbalzo al microcontrollore. Un'altra possibilità considerata, era di utilizzare un diplay touch sia per controllare le periferiche, che per mostrare i dati, ma è stata scartata per via del prezzo.

STUDENTSUPSI

STUDENTSUPSI

Capitolo 9

Software

9.1 Diagramma a blocchi

La figura 9.1, riporta il diagramma a blocchi del programma scritto per comandare i sistemi d'accesso, gestire gli input e fornire un feedback all'operatore.

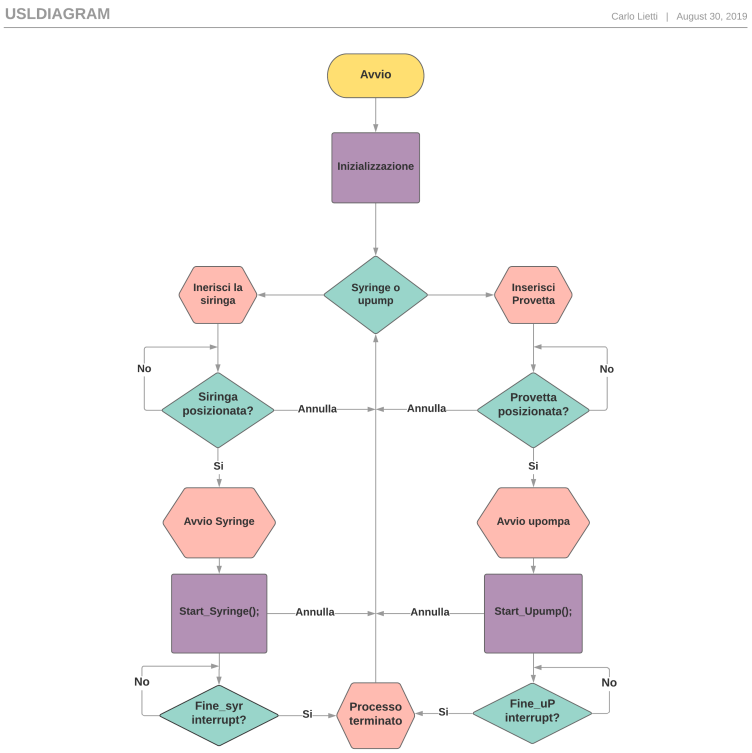


Figura 9.1: Flow chart porgramma

STUDENTSUPSI

9.2 CubeMX

Per impostare le diverse periferiche interne del microcontrollore, è stato usato un software creato appositamente da "STmicroelectronics", **CubeMX**. Questo software semplifica tutti i passaggi relativi alla scelta ed all'abilitazione dei diversi moduli interni, inoltre grazie alla sua interfaccia grafica la selezione dei diversi pin e dei clock interni risulta molto semplificata. Tramite questo software è inoltre possibile abilitare o disabilitare gli interrupt relativi ad ogni periferica(ADC, SPI, TIMER, ecc...). Questa funzione è molto utile quando non si ha familiarità, con l'ambiente di sviluppo del software. In figura 9.2 viene mostrato come sono stati impostati i pin del microcontrollore. I pin verdi indicano dei pin impostati, con una precisa funzione, i pin gialli, sono stati selezionati, ma hanno una funzione non definita o multipla, mentre i pin grigi non sono utilizzati, molti di loro, però, sono disponibili per l'utilizzo all'esterno, tramite un header[12]. Oltre a permettere di selezionare la funzione di ogni piedino

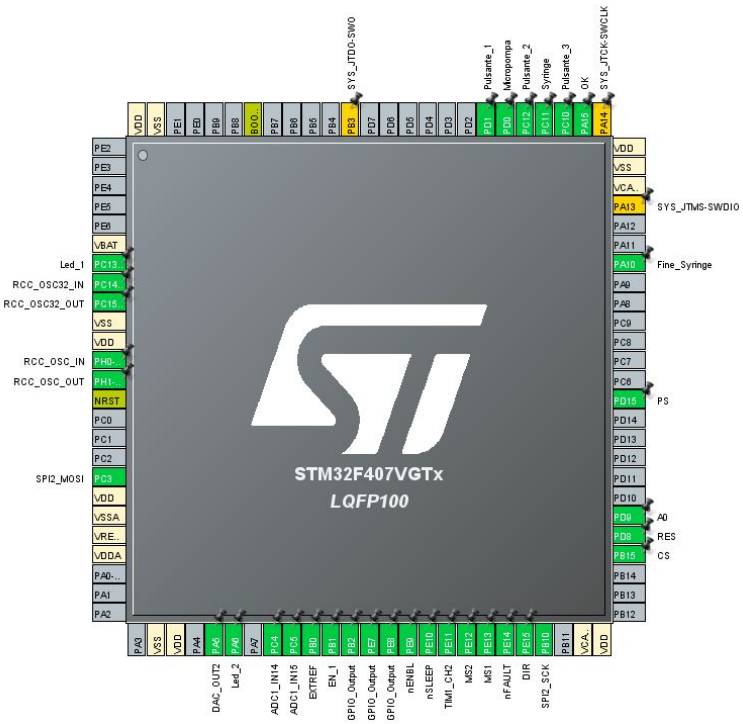


Figura 9.2: Impostazione pinout tramite CubeMX

del microcontrollore e quali moduli abilita, il programma consente anche di gestire, tramite un'interfaccia dedicata, i diversi clock presenti nel circuito. Quest'interfaccia è mostrata in figura 9.3,

[12]Tipo di connettore, utilizzato molto in elettronica.

STUDENTSUPSI

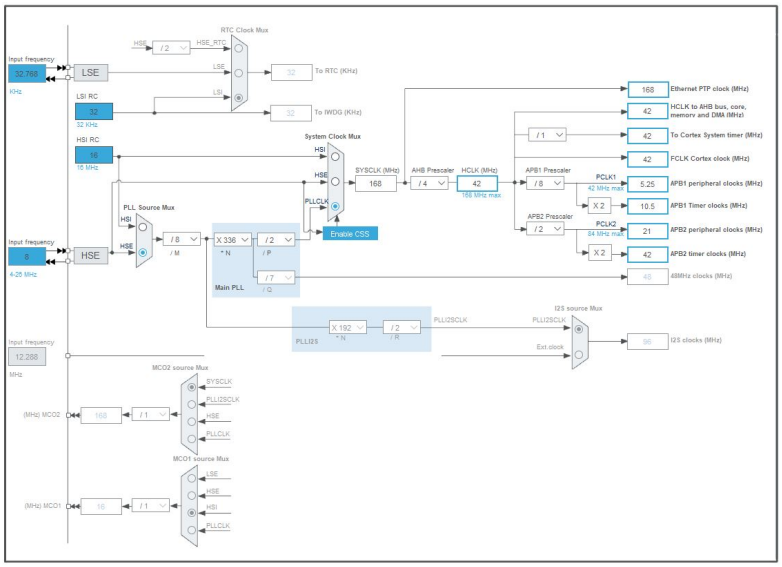


Figura 9.3: Interfaccia controllo dei clock CubeMX

9.3 Il programma

Per scrivere il programma è stato usato il software keilμVision Esso offre la possibilità di compilare e debuggare il programma caricato sul microcontrollore, anche in "run-time". In figura 9.4, sono mostrate le variabili create per il funzionamento del programma. Tutti gli interi sono utilizzati all'interno del ciclo principale, per gestire le fasi e i comandi da inviare. Per quanto riguarda la variabile *Tensione*, essa viene utilizzata insieme a $DACbyte$ per comandare il DAC interno, questo dispositivo viene poi utilizzato per fornire la soglia di comparazione al comparatore esterno. Nell'ultima variabile, vengono salvati i dati di conversione dell'ADC.

```
/* USER CODE BEGIN PV */
  int i=              1;        //inizializzo delle variabili
  int uf =            0;        //flag per micropump
  int syr =           0;        //flag per syringe pump
  int Flag =          0;        //FlagStatus generico
  int ciclo_uP =      0;        //tilizzato per la routine della micropump
  int ciclo =         0;
  float Tensione =  2.1;

  uint16_t adc_val_1 = 0;
  uint8_t DAC_byte;
/* USER CODE END PV */
```

Figura 9.4: Variabili create ed inizializzate

La figura 9.5, mostra la fase di inizializzazione del programma. Lo scopo di questa fase è portare il sistema ad uno stato neutro, infatti, nel caso di errore oppure di uno spegnimento improvviso il sistema potrebbe bloccarsi e se non venisse inizializzato, partirebbe da una posizione sconosciuta e potenzialmente poco agevole per l'utilizzatore. Tramite questa fase, invece, viene permesso all'operatore di agire senza problemi sul sistema. In figura 9.6a,

STUDENTSUPSI

sono mostrate la funzione di inizializzazione per la syringe pump e quella per la micropump. La prima, imposta i diversi piedini del driver MP6500 e porta indietro la siringa, dopodiché la blocca, così da lasciare spazio per l'inserimento di una siringa qualsiasi all'interno della syringe pump stessa. La seconda funzione, invece, imposta il valore del DAC[13] e lo abilita, così da fornire il riferimento al comparatore esterno. In figura 9.6b viene mostrata la funzione che resetta tutti i flag, usati nel ciclo principale, che viene chiamata in caso in memoria fosse rimasto salvato un valore diverso per quelle variabili, nonostante siano state inizializzate precedentemente.

```
/* USER CODE BEGIN 2 */
DWT_Init();
DAC_byte = (uint8_t)((Tensione/3)*255);          //avvio DAC Per traslazione dei livelli
HAL_ADC_Start(&hadc1);                            //Avvio ADC
LCD_Init();                                       //Inizzializzo LCD
LCD_Command(GLCD_Inizio);                         //Scrive sull LCD
Stepper_init();                                   //inizializza  la Siringa
Micropump_init();                                 //inizializza le micropompe
HAL_GPIO_WritePin(GPIOC, Led_1_Pin, GPIO_PIN_SET);//accende un led
Flag = 0;                                         //variabile usata per segnalare lo stato del sistema
/* USER CODE END 2 */
```

Figura 9.5: Inizializzazione sistema

```
void Stepper_init(void)
{
  HAL_GPIO_WritePin(GPIOE, nENBL_Pin, GPIO_PIN_SET);
  HAL_GPIO_WritePin(GPIOE, nSLEEP_Pin, GPIO_PIN_SET);
  HAL_GPIO_WritePin(GPIOE, DIR_Pin, GPIO_PIN_SET);
  HAL_GPIO_WritePin(GPIOE, MS1_Pin, GPIO_PIN_RESET);
  HAL_GPIO_WritePin(GPIOE, MS2_Pin, GPIO_PIN_RESET);
  HAL_TIM_PWM_Start(&htim1, TIM_CHANNEL_2);
  HAL_GPIO_WritePin(GPIOE, nENBL_Pin, GPIO_PIN_RESET);
  HAL_Delay(5000);
  HAL_GPIO_WritePin(GPIOE, nENBL_Pin, GPIO_PIN_SET);
  HAL_GPIO_WritePin(GPIOE, nSLEEP_Pin, GPIO_PIN_RESET);
  HAL_GPIO_WritePin(GPIOD, GPIO_PIN_12, GPIO_PIN_RESET);
  HAL_TIM_PWM_Stop(&htim1, TIM_CHANNEL_2);
}

void Micropump_init(void)
{
  //accendere il DAC
  HAL_DAC_Start(&hdac, DAC_CHANNEL_2);
  HAL_DAC_SetValue(&hdac, DAC_CHANNEL_2, DAC_ALIGN_8B_R, DAC_byte);
  HAL_GPIO_WritePin(GPIOB, GPIO_PIN_0, GPIO_PIN_SET);
  HAL_GPIO_WritePin(GPIOB, GPIO_PIN_1, GPIO_PIN_SET);
}
```

(a) Funzioni di inizializzazione

```
void reset_flag(void){
  uf =        0;
  syr =       0;
  ciclo =     0;
  Flag =      0;
  i =         10;
}
```

(b) Funzione che resetta i flag

Figura 9.6: funzioni di inizializzazione programma

A questo punto inizia il corpo principale del programma, la figura 9.7, ne mostra la prima parte, dopo aver scritto sul display di selezionare che tipo di input si vuole usare, il programma entra in un loop, da cui uscirà solo nel momento in cui verrà premuto uno dei due pulsanti di selezione, ovvero "syringe pump" o "micropump". Una volta fatta la scelta, il display scriverà di posizionare la siringa o la provetta, a seconda di ciò che ci è scelto ed attenderà ancora la pressione di un pulsante.

Se viene scelto di utilizzare la syringe pump si arriva al codice presente in figura 9.8, il programma entra in un loop, in cui controlla i flag ad ogni ciclo. All'interno del loop, il microcontrollore esegue una conversione tramite l'ADC, e scrive sul display il risultato di tale conversione. Una volta che la siringa arriva a fine corsa o l'operatore preme il tasto annulla, tutti i flag vengono resettati e il motore viene portato indietro. Durante queste operazioni vengono utilizzate le funzioni visibili in figura 9.9

[13] Convertitore da digitale ad analogica, consente di traslare un segnale digitale (composto da '1' e '0'), in uno analogico

STUDENTSUPSI

```c
while (1)
{

  LCD_Data(GLCD_Scelta);                        //mostra scelta iniezione
  while (Flag == 0){                            //aspetta la scelta
  }
  if (Flag == 1 && uf ==1){                     //scelta micropomp
    LCD_Data(GLCD_Posiziona_prov);              //mostra posiziona provetta, premi ok per continuare
    while(Flag == 1 && uf ==1){                 //aspetta il tasto ok
    }
  }
  else if(Flag == 1 && syr == 1){               //scelta la syringepump
    LCD_Data(GLCD_Posiziona_syr);               //mostra posiziona la siringa e premi ok
    while(Flag == 1 && syr == 1){                 //spetta la scelta
    }
  }
  else
    __NOP;
```

Figura 9.7: Prima fase del programma

```c
  if (Flag == 2 && syr == 2){
    LCD_Data(GLCD_Avvio_syringe);               //avvio syringe pump
    Start_syringe();                            //avvia sirynge pump
    while(ciclo > 0)                                  //loop che va finché la sringe pump non finisce
    {
      HAL_ADC_PollForConversion(&hadc1, 10);        //verifica che l'ADC non sia convertendo
      adc_val_1 = HAL_ADC_GetValue(&hadc1);         // inizia la conversione ADC
      LCD_adc_dato(adc_val_1);
    }
  }
```

Figura 9.8: Funzionamento syringe pump

```c
void Start_syringe(void)
{
  HAL_GPIO_WritePin(GPIOE, nSLEEP_Pin, GPIO_PIN_SET);     //Porta a '1' nSLEEP così il driver esce dalla modalità sleep
  HAL_GPIO_WritePin(GPIOE, DIR_Pin, GPIO_PIN_SET);        //imposta la direzione
  HAL_GPIO_WritePin(GPIOE, MS1_Pin, GPIO_PIN_RESET);      //imposta il passo
  HAL_GPIO_WritePin(GPIOE, MS2_Pin, GPIO_PIN_RESET);      //imposta il passo
  HAL_GPIO_WritePin(GPIOE, nENBL_Pin, GPIO_PIN_RESET);    //abilita i ponti ad H interni al driver
  HAL_TIM_PWM_Start(&htim1, TIM_CHANNEL_2);               //Attiva la PWM che invia gli impulsi al driver
}

void Stop_syringe(void)
{
  HAL_GPIO_WritePin(GPIOE, nENBL_Pin, GPIO_PIN_SET);      //Disabilità i ponti ad H
  HAL_GPIO_WritePin(GPIOE, nSLEEP_Pin, GPIO_PIN_RESET);   // abilita la modalità SLEEP
  HAL_TIM_PWM_Stop(&htim1, TIM_CHANNEL_2);                //Spegna la PWM
  Stepper_init();
}
```

Figura 9.9: Funzioni di start e stop per la syringe pump

STUDENTSUPSI

Nel caso in venisse scelta la micropompa come sistema d'accesso, invece, si usata la parte di programma mostrata in figura 9.10, contrariamente all'altro sistema in questo viene creato un segnale di clock alzando ed abbassando un piedino del microcontrollore ogni 416 micro secondi. In questo modo si avranno 4 impulsi ogni 3.33ms, che corrispondono ad una frequenza per serie d'impulsi di 300Hz. Il motivo per cui si utilizza questo metodo deriva dal componente HV513, esso, infatti, per impostare l'uscita da abilitare riceve quattro impulsi di clock sul piedino CLK_1, ad ogni fronte di salita del clock, verifica che valore è presente sul pin DIN, se è '1' salva '1', mentre se è '0' salva '0', dopo 4 impulsi, converte gli '1' e gli '0' trovati in decimale e così seleziona l'uscita. In figura 9.11 è mostrato il concetto appena spiegato. Nella routine che controlla la micropump, vengono azionate le uscite 1 e 3 a ripetizione, poiché sono connesse alla micropompa tramite un connettore "molex FFC".

```
for(i = 0; i < 8; i++)
{
  if(i == 0 || i == 4 || i == 5)
  {
    DWT_Delay(416);
    HAL_GPIO_WritePin(GPIOB, DIN_Pin, GPIO_PIN_SET);     //invio
    HAL_GPIO_WritePin(GPIOB, CLK_1_Pin, GPIO_PIN_SET);   //impulso di clock per HV513
    DWT_Delay(416);
    HAL_ADC_PollForConversion(&hadc1, 10);               //verifica che l'ADC non sia convertendo
    adc_val_1 = HAL_ADC_GetValue(&hadc1);                // inizia la conversione ADC
    LCD_adc_dato(adc_val_1);
    HAL_GPIO_WritePin(GPIOB, DIN_Pin, GPIO_PIN_RESET);
    HAL_GPIO_WritePin(GPIOB, CLK_1_Pin, GPIO_PIN_RESET);
  }
  if(i== 1 || i == 2 || i == 3 || i == 6 || i ==7)
  {
    DWT_Delay(416);
    HAL_GPIO_WritePin(GPIOB, CLK_1_Pin, GPIO_PIN_SET);   //impulso di clock per HV513
    DWT_Delay(416);                                      //Delay di 416us
    HAL_ADC_PollForConversion(&hadc1, 10);               //verifica che l'ADC non sia convertendo
    adc_val_1 = HAL_ADC_GetValue(&hadc1);                // inizia la conversione ADC
    LCD_adc_dato(adc_val_1);                             //Delay di 416us
    HAL_GPIO_WritePin(GPIOB, CLK_1_Pin, GPIO_PIN_RESET);//impulso di clock per HV513
  }
 }
}
```

Figura 9.10: Controllo micropompa

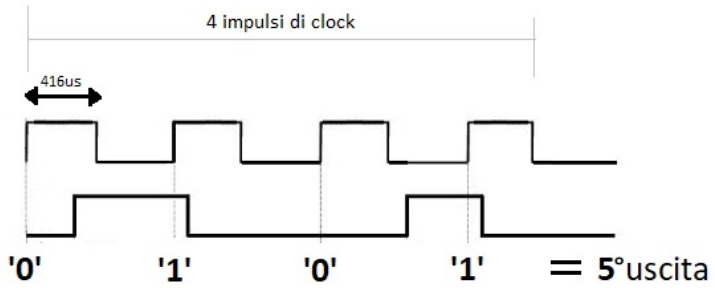


Figura 9.11: Funzionamento comando micropompa

Per consentire all'operatore di svolgere le funzioni desiderate, vengono utilizzati dei pulsanti, impostati come interrupt esterni tramite cubeMX, ad ogni pressione i pulsanti aggiornano i flag, che a loro volta impostano quale periferica attivare e quale no. In figura 9.12 viene mostrata la routine di interrupt relativa ai pulsanti.

STUDENTSUPSI

```
if(GPIO_Pin == Syringe_Pin){
  if(Flag == 0)
    {
      Flag = 1;
      syr = 1;
    }
  else
    __NOP;
//  HAL_GPIO_TogglePin(GPIOD, GPIO_PIN_13);
}
  if(GPIO_Pin == OK_Pin){
    if(Flag == 1 & uf == 1){
      Flag = 2;
      uf = 2;
    }
    if(Flag == 1 & syr == 1){
      Flag =2;
      syr =2;
    }
  }
  //HAL_GPIO_TogglePin(GPIOD, GPIO_PIN_13);
```

Figura 9.12: Comportamento pulsanti

Le funzioni relative all'utilizzo del display sono molte, per questa ragione ne verranno analizzate solamente due. Le altre sono visibili nel datasheet del display. In figura 9.13 sono mostrate le due funzioni principali del display. Entrambe servono per controllare il display, la differenza è che $LCDCommand$, invia dei comandi al display che ne variano i parametri, come ad esempio: Contrasto, colore, orientamento ecc. L'altra, invece, serve per inviare dire al display cosa scrivere a schermo. Nel caso dell'invio di un comando il piedino A0 deve essere posto a '0', mentre nel caso contrario A0 deve essere posto a '1'.

```
void LCD_Data(uint8_t data) {
    uint8_t dati[1];
    dati[0] = data;
    HAL_GPIO_WritePin(GPIOD, A0_Pin, GPIO_PIN_SET);       //Alza A0
    HAL_GPIO_WritePin(GPIOD, CS_Pin, GPIO_PIN_RESET);     //abbassa il Chip selct
    HAL_SPI_Transmit(&hspi2, dati, 1, 10);                //Trasmettte tremite SPI
    HAL_GPIO_WritePin(GPIOD, CS_Pin, GPIO_PIN_SET);       //deseleziona il chip select
    HAL_GPIO_WritePin(GPIOD, A0_Pin, GPIO_PIN_RESET);     //Abbassa A0

}

void LCD_Command(uint8_t command) {
    uint8_t commandazzi[1];
    commandazzi[0] = command;
    HAL_GPIO_WritePin(GPIOD, A0_Pin, GPIO_PIN_RESET);     //Abbassa A0
    HAL_GPIO_WritePin(GPIOD, CS_Pin, GPIO_PIN_RESET);     //Abbassa il Chip select
    HAL_SPI_Transmit(&hspi2, commandazzi, 1, 10);         //Trasmette tramite SPI
    HAL_GPIO_WritePin(GPIOD, CS_Pin, GPIO_PIN_SET);       //Deseleziona il chip select
    HAL_GPIO_WritePin(GPIOD, A0_Pin, GPIO_PIN_SET);       //ALZA A0

}
```

Figura 9.13: Funzioni per il passaggio di dati o comandi al display

Il programma scritto è piuttosto inefficiente, in quanto blocca spesso la CPU del microcontrollore in loop, da cui esce solo nel momento in cui riceve un interrupt esterno. in tal modo vengono utilizzate molte più risorse del necessario aumentando i consumi e diminuendo le performace, fortunatamente il sistema di alimentazione è in grado di erogare tutta la corrente necessaria, per via del fatto che è connesso alla tensione di linea. Ciononostante, in futuro,

STUDENTSUPSI

si potrebbe aggiornare il programma, in modo da diminuire l'uso della CPU, aumentando le performance diminuendo il rischio di un blocco del sistema.

STUDENTSUPSI

Capitolo 10

Spesa economica

Per realizzare la questo progetto sono stati predisposti 600 Chf, la tabella 10 mostra le spese effettuate

Struttura Syringe pump con motore	104.33
Micropump	198.65
PCB	10.00
Elettronica	128.83
Chip microfluidico	91.20
Struttura	25.00

Contando anche le tasse si è usato tutto il budget senza, però, sforarlo. La spesa più importante sono state le micropompe, infatti, non è stato possibile acquistarne 1 singolarmente, poiché la ditta venditrice le vende a pacchetti di 5. Acquistandone anche solo 3 si sarebbero risparmiati circa 80 Chf.

STUDENTSUPSI

STUDENTSUPSI

Capitolo 11

Miglioramenti futuri

Questa tesi lasci un ampio margine di implementazioni possibili, non a caso sono stati previsti già predisposti dei collegamenti per ulteriori dispositivi esterni. I miglioramenti possibili per questo progetto sono svariati, tra i più importanti troviamo:

- Predisporre un sistema di misura dell'impedenza di una membrana formata da peptidi auto-autoassemblanti in modo da consentire la confutazione della teoria descritta inizialmente.
- Predisporre il sistema per essere in grado di compiere una misura fluorimetrica su una membrana microfluidica, lo scopo è di capire se sia possibile rilevare che tipo di virus viene trasportato.
- Correggere gli errori delle board create.
- Aggiungere un maggior numero di sensori, che abbiano una maggiore precisione e siano costruiti appositamente per l'ambito microfluidico. Così da permettere di controllare in modo più accurato dosi e velocità dei fluidi utilizzati.
- Creare una struttura solida, che garantisca un buon grado di sterilità contro i fattori esterni ed adatta al posizionamento delle diverse periferiche utilizzate.
- Migliorare la sezione riguardante i sistemi d'accesso, in modo da garantire a scienziati e ricercatori, maggiore possibilità di scelta per l'ingresso nel sistema.

STUDENTSUPSI

STUDENTSUPSI

Capitolo 12

Conclusioni

É stato parzialmente realizzato un sistema che permette, in modo immediato, di pompare un liquido o un gas all'interno di un chip microfluidico. Il sistema può funzionare correttamente se si correggono i difetti dovuti al programma ed all'alimentazione degli attuatori. Per controllare ogni apparecchiatura interfacciata al sistema è stata realizzata da un'elettronica ad hoc, che consente di pilotare agevolmente tutte le sezioni di esso. Durante la realizzazione sono stati incontrati diversi problemi, che non hanno permesso di progredire troppo nel perfezionamento del sistema. Primo tra tutti il dimensionamento dei vari integrati e l'alimentazione di essi. In particolare, mettere in funzione correttamente la syringe pump ha richiesto molto tempo per via delle tensioni e dell'alta corrente richiesta. Per quanto riguarda la micropump, invece, l'elettronica ha funzionato correttamente. Un'altro problema riscontrato è relativo al display, per metterlo in funzione, infatti, ci sono volute diverse prove, poiché le librerie per comandarlo sono state riscritte dall'inizio.

STUDENTSUPSI

STUDENTSUPSI

Capitolo 13

Bibliografia

Gianluca Montù (*Driver per motori piezoelettrici, design, realizzazione e controllo*)

Niccolò Rondelli (*Studio e sviluppo di tecnologie microfluidiche per analisi di acidi su Lab-on-Chio*)

Krzysztof Adamski,Bartosz Kawa , Rafał Walczak (*3D printed flowmeter based onVenturi effect with integrated Sensors*)

Siti:

- www.wikipedia.org
- https://edeca.net/pages/the-st7565-display-controller
- https://www.elveflow.com/microfluidic-tutorials/microfluidic-reviews-and-tutorials/materials-for-microfluidic-chips-fabrication-a-review-2017/
- www.mouser.com
- www.digikey.com
- www.snapeda.com
- www.stmicroelectronics.com
- www.stackoverflow.com
- www.youtube.com

https//.webusers.fis.uniroma3.it

STUDENTSUPSI

Capitolo 14

Ringraziamenti

Per la realizzazione di questa tesi si ringrazia:

- La tutta la mia famiglia, per il supporto e la pazienza dimostrata nei miei confronti durante tutto il lavoro di tesi.
- La mia ragazza Martina che non ha mai smesso di supportarmi, sia che fosse vicina, sia che fosse lontana.
- La mia classe universitaria la E3A, i cui componenti sono sempre stati pronti a darmi una mano, In particolare Mattia Ceretti che mi ha aiutato molto durante per la saldatura e per la scelta dei componenti.
- Il Signor Loredano Tonietti per la disponibilità nell'aiutare.
- I miei amici che hanno sempre cercato di aiutarmi nonostante non sapessero bene cosa stessi facendo.
- Simone Zaffaroni e Gabriele Cenedese, per essermi stati vicini ed avermi costretto a metterli in questi ringraziamenti.
- Jacopo Gangi che mi ha scritto un giorno si e un no per sapere come proceva il lavoro, senza mai smettere di esserne positivo al riguardo.
- Il Professor Gianluca Montù, per la sua gentilezza.

STUDENTSUPSI