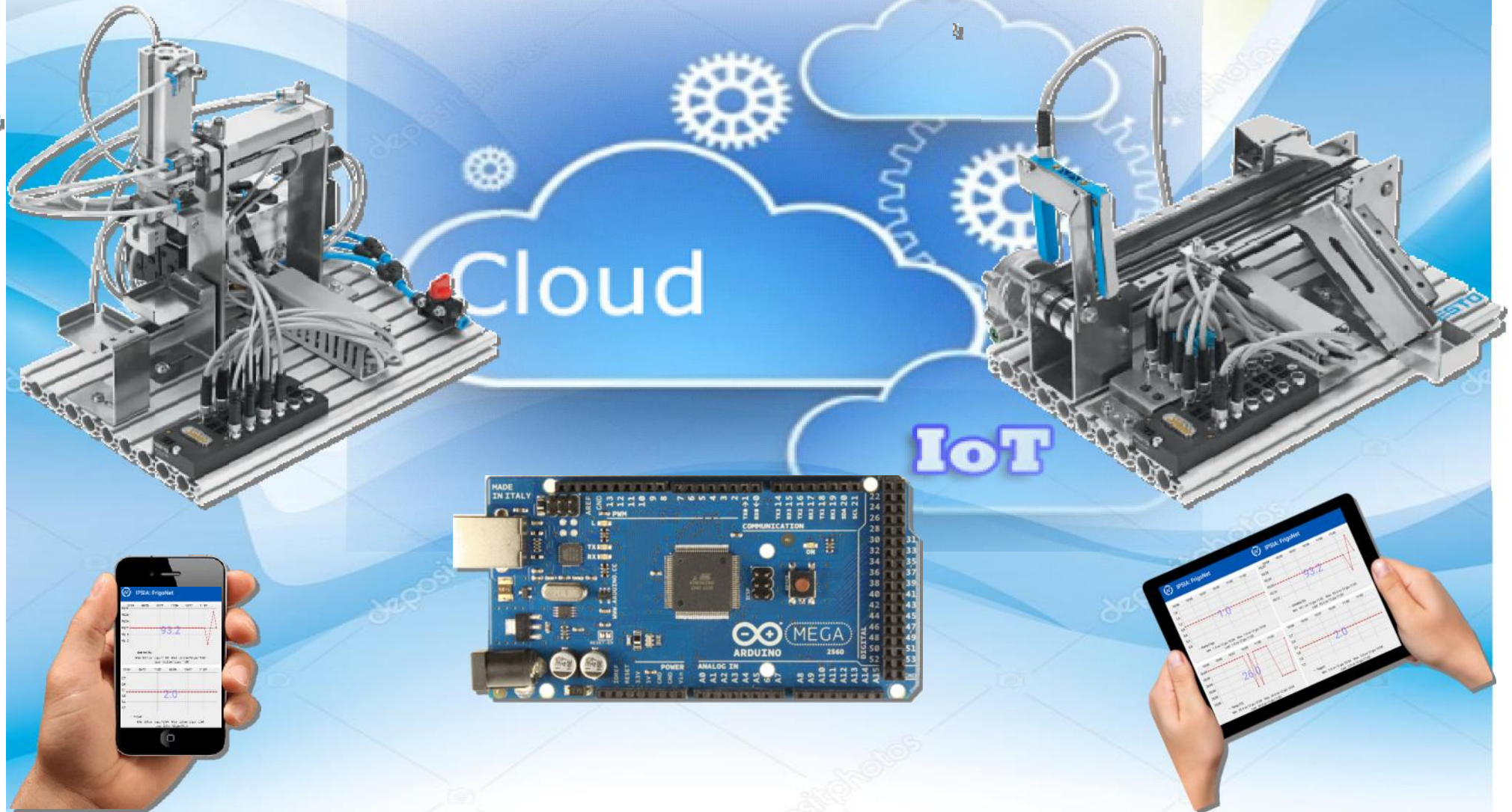


# Mec\_Lab 4.0

## Meccatronica con App IoT



A.S. 2018-2019 - Cl. 4A\_IPAI- 5A\_IPAI (Ind. Elettronico) - Ref. Progetto: *Prof. Franco Tufoni* - *Prof. Enrico Ruggieri*  
IPSIA - Antonio Guastaferrò - Viale dello Sport, 60 – 63074 San Benedetto del Tronto (AP) [www.ipsia.edu.it](http://www.ipsia.edu.it)

## Indice

Descrizione	2
Funzionamento	7
Descrizione dei singoli Moduli	
1) Arduino Mega 2560 (Sistema di sviluppo programmabile)	10
2) Shield Traslatore livelli 5v/24v - 24v/5v	11
3) Manipolatore Festo (Handling station)- Spostamento dei pezzi semilavorati	14
4) Nastro trasportatore FESTO - Conveyor station - Traslazione pezzi sulla linea di produzione	18
5) Modulo Ambiente: CCS811 (CO2, TVOC) - DHT22 (Temp, Umidità)	20
6) Switch Mode - Modulo Pulsanti	22
7) Display LCD I2C 20x4	23
8) Modulo Wi-Fi ESP 8266 - Led IoT	24
9) Canale ThingSpeak: <a href="https://thingspeak.com/channels/643451">https://thingspeak.com/channels/643451</a>	25
Software di gestione	28
Upgrade - A.S. 2019-2020	31
Sitografia	31

## Arduino Mec\_Lab 4.0 - Meccatronica con App IoT

### Descrizione

**Mec\_Lab 4.0** è un sistema **Meccatronico IoT** (Internet of Things), in grado di riprodurre in scala alcune fasi di un processo di produzione industriale. L'**App. IoT** integrata consente di inviare ad un **Cloud** i dati di **produzione** ed alcune grandezze **ambientali** (Temperatura, Umidità e qualità dell'aria (**CO2** e **TVOC**)) del luogo di lavoro.

Il monitoraggio, da remoto, dei parametri ambientali del luogo di lavoro consente di controllare e garantire un luogo sano e sicuro in base alle attuali norme di sicurezza.

Il prototipo elaborato dagli studenti del 4 e 5 elettronica come integrazione al percorso Alternanza Scuola Lavoro, con la supervisione dei docenti di indirizzo, ha una forte valenza **didattica multidisciplinare**.

**MecLab 4.0** permette l' apprendimento dei concetti base della **meccanica, pneumatica, elettronica e programmazione**, inoltre, tramite l' App **IoT** consente di introdurre il concetto di **Industria 4.0** in cui è importante ed indispensabile la disponibilità dei dati di produzione in un Cloud per la consultazione da remoto con un dispositivo mobile: Tablet, Cellulare, Smartphone, ecc.

**MecLab 4.0** è composto da:

1. **Manipolatore Festo**  
*- Spostamento dei pezzi semilavorati*
2. **Nastro trasportatore Festo**  
*- Traslazione pezzi sulla linea di produzione*
3. **Selezionatore semilavorati**  
*- Smistamento in base al tipo di materiale*
4. **Unità di controllo Elettronica programmabile**  
*- Unità programmabile per la gestione dei sensori ed attuatori della parte meccanica ed ambientale*
5. **Modulo Wi-Fi**  
*- Trasmissione dei dati di produzione ed ambientali al Cloud*

**MecLab 4.0** tramite l' unità di controllo elettronica presenta tre modalità di programmazione:

- a) Mini PLC con software Ladder
- b) Festo FluidSim con l'interfaccia EasyPort (Programmazione e simulazione)
- c) Arduino Mega con lo Shield traslatore di livelli (5v/24v - 24v/5v) (Software IDE)

Il modello proposto prevede la gestione e programmazione con **Arduino Mega 2560**.

In fig. 1 è riportato lo schema a blocchi, in fig. 2 lo schema elettrico, in fig. 3 - Foto del prototipo con il gruppo classe



# Mec\_Lab 4.0

## Meccatronica con App IoT

Schema a blocchi

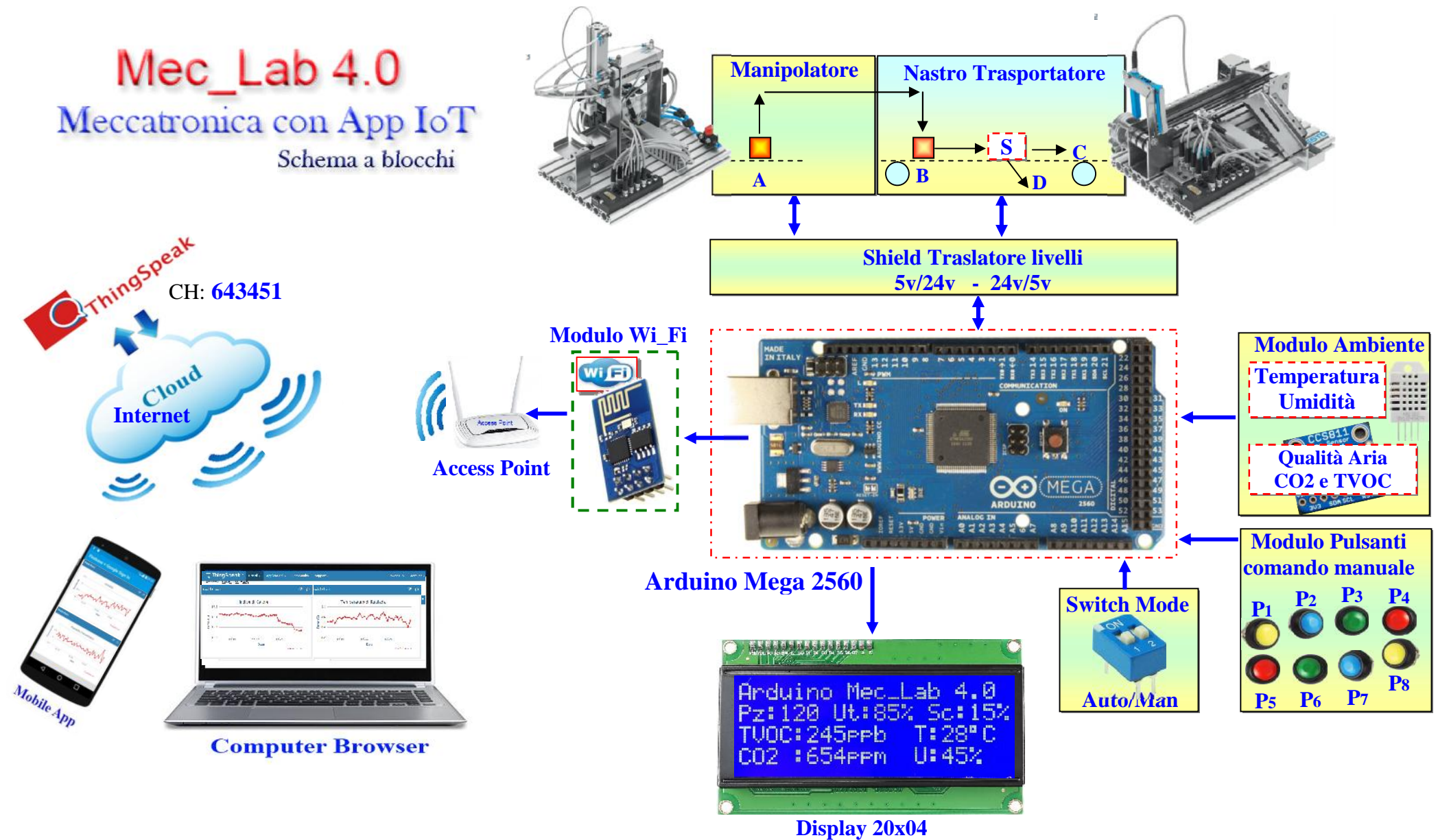


Fig. 1 Schema a blocchi

# Mec\_Lab 4.0

## Meccatronica con App IoT

Schema elettrico

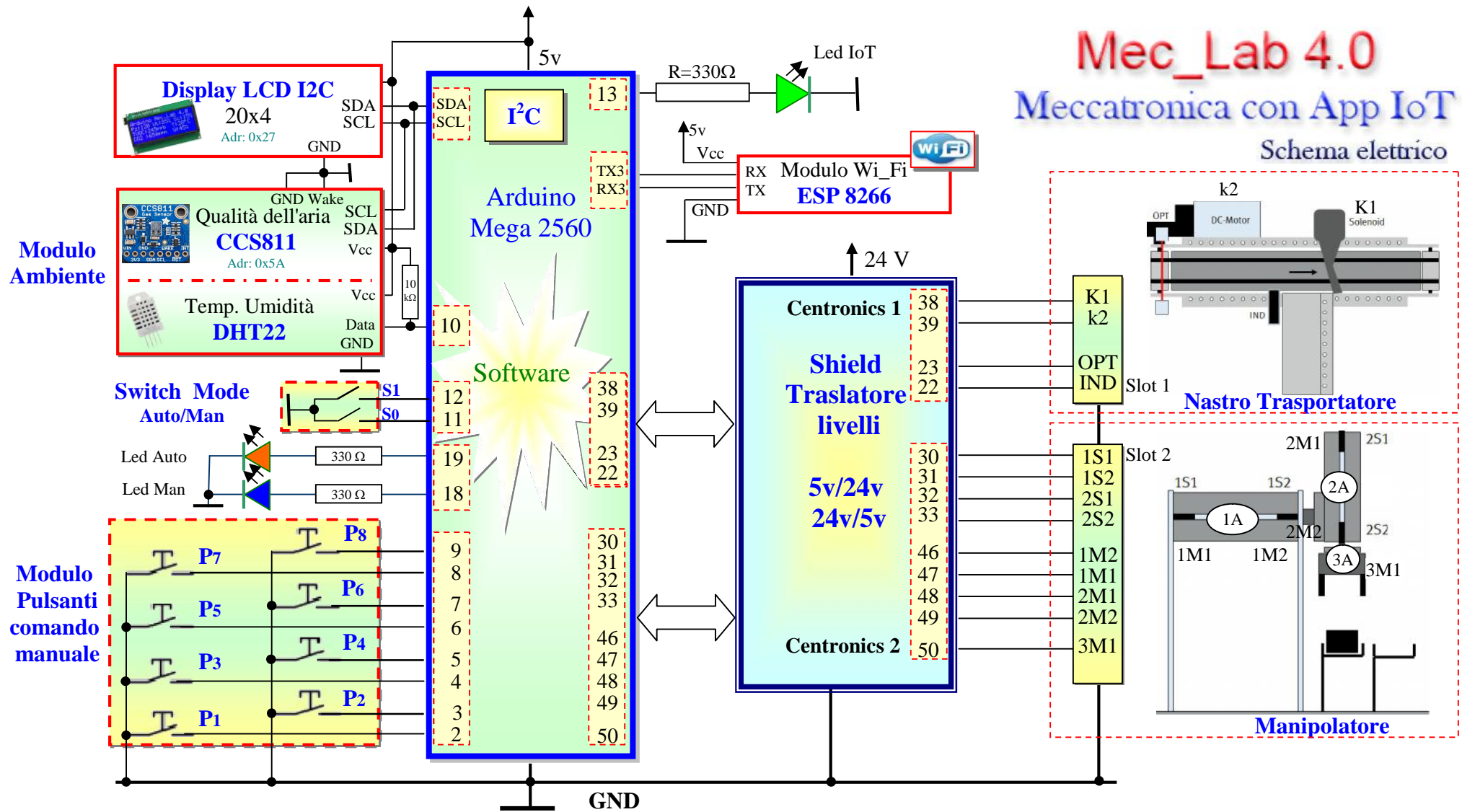


Fig. 2 - Schema elettrico



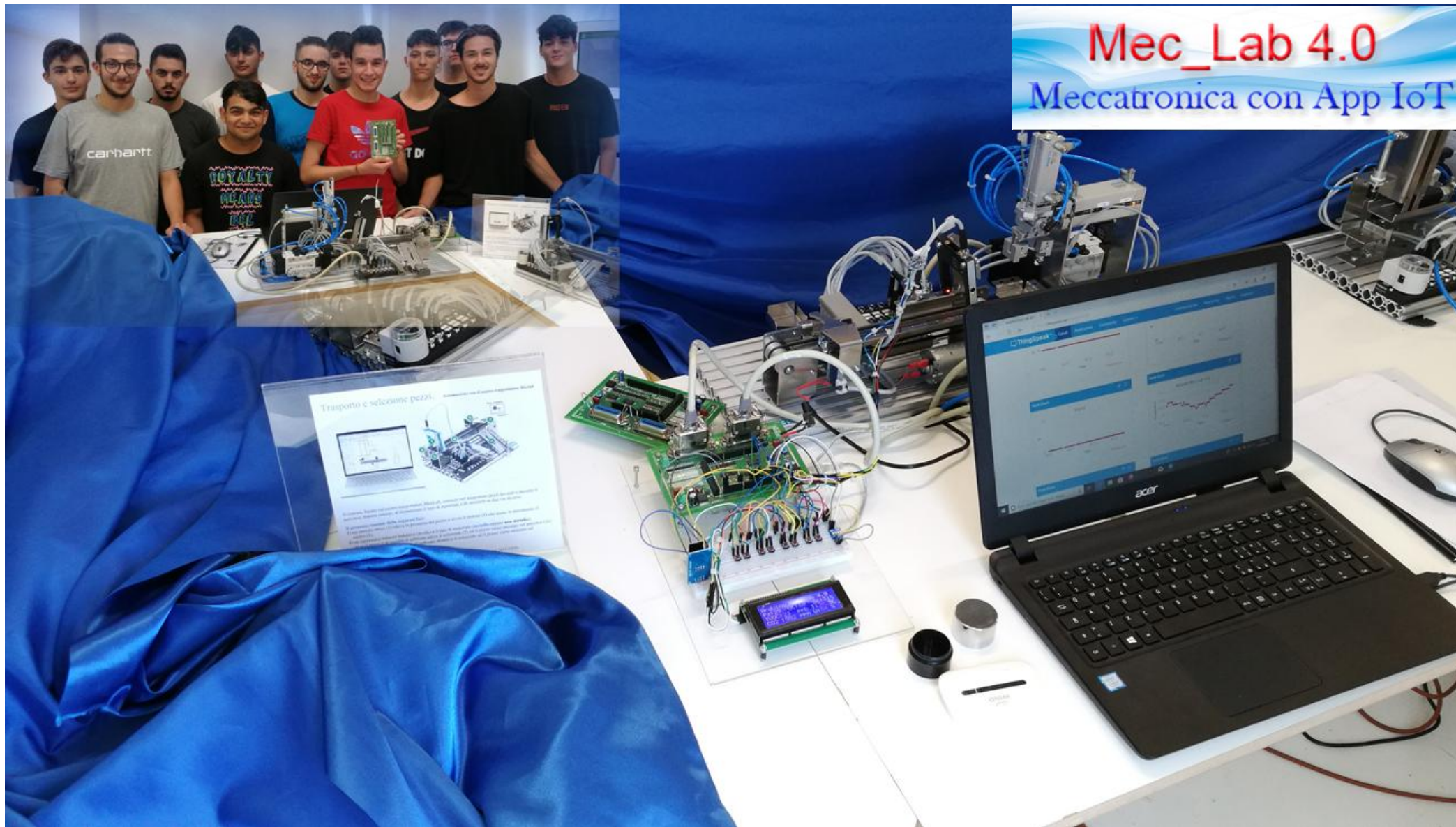


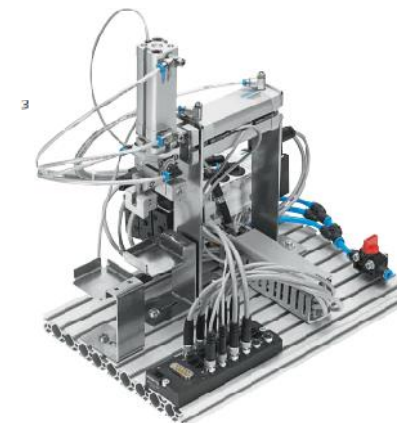
Fig. 3 - Foto del prototipo con il gruppo classe

Di seguito è riportata la tabella connessioni : **Arduino Mega - Shield MecLab - Moduli MecLab**

**Slot distributore Manipolatore - Arduino Mega con Shield MecLab (Centronics 2)**

**Tabella connessioni - Manipolatore**

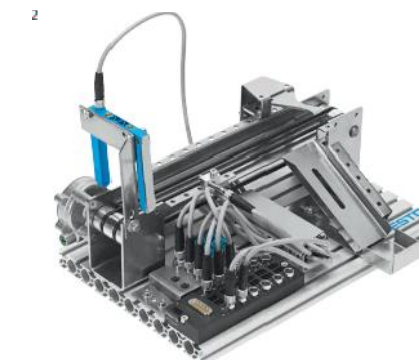
Slot 2 - MultiPin	Designation	Pin Arduino	Description
0	1S1	30 - Input	Magnetic proximity sensor at cylinder 1A, rear
2	1S2	31 - Input	Magnetic proximity sensor at cylinder 1A, front
4	2S1	32 - Input	Magnetic proximity sensor at cylinder 2A, top
6	2S2	33 - Input	Magnetic proximity sensor at cylinder 2A, bottom
1	1M1	46 - Output	Advance cylinder 1 - Ori.
3	1M2	47 - Output	Retract cylinder 1 - Ori.
5	2M1	48 - Output	Advance cylinder 2 - Vert.
7	2M2	49 - Output	Retract cylinder 2 - Vert.
9	3M1	50 - Output	Close gripper



**Slot distributore Nastro Trasportatore - Arduino Mega con Shield MecLab (Centronics 1)**

**Tabella connessioni - Nastro Trasportatore**

Slot 1 - MultiPin	Designation	Pin Arduino	Description
0	IND	22 - Input	Inductive sensor
1	K1	38 - Output	Solenoid/deflector relay output
2	OPT	23 - Input	Opto sensor
3	K2	39 - Output	DC-Motor



## Funzionamento

Sono previste due modalità, **manuale/automatica**, selezionabili tramite **Switch Mode**.

In tabella sono riportate le modalità e la funzione dei singoli pulsanti.

Tabella pulsanti: Funzionamento manuale				Tabella pulsanti: Funzionamento automatico			
Switch Mode: S0=Off, S1=Off				Switch Mode: S0=On, S1=Off			
Pulsante	Pin Arduino	Modalità Pin	Funzione	Pulsante	Pin Arduino	Modalità Pin	Funzione
P1	2	INPUT_PULLUP	Chiude pinza	P1+P8	2,9	INPUT_PULLUP	Start
P2	3	INPUT_PULLUP	Apri pinza				
P3	4	INPUT_PULLUP	Cilindro-Ori	P4	5	INPUT_PULLUP	Stop
P4	5	INPUT_PULLUP	Cilindro-Ori				
P5	6	INPUT_PULLUP	Cilindro-Vert				
P6	7	INPUT_PULLUP	Cilindro-Vert				
P7	8	INPUT_PULLUP	Start Nastro				
P8	9	INPUT_PULLUP	Stop Nastro				

### Manuale

Tramite i pulsanti P1÷ P8 è possibile attivare in modo indipendente i singoli attuatori (*Elettrovalvole, cilindri, pinza e nastro trasportatore*) ed impostare un funzionamento **Step by Step**. Questa modalità permette di far comprendere il funzionamento delle singole parti, utile da punto di vista didattico.

### Automatico

Un operatore posiziona il pezzo nel **punto A** (Fig. F1), la pressione simultanea di **P1+P8** avvia il processo. Il cilindro verticale del manipolatore porta la pinza sul pezzo, viene agganciato e spostato sul **punto B** del nastro trasportatore.

Il nastro trasporta il pezzo sulla linea di produzione. L'unità di **selezione S** effettua il controllo e lo smistamento (**linea C** oppure **linea D**).

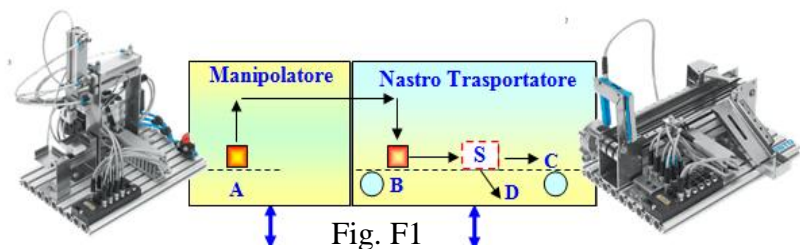
Il pulsante **Stop (P4)** ferma il processo.

In tutte e due le modalità sono attive l'unità di **selezione S** ed il **modulo ambiente**.

L'unità di **selezione S**, situata sulla linea, composta da due sensori (*ottico, induttivo*) ed un attuttore (*solenioide*), conta i pezzi e li smista in due percorsi (**linea C** oppure **linea D**) in base al tipo di materiale.

**Metallo:** considerato pezzo **utile (linea D)**      **Non metallo:** considerato **scarto (linea C)**

Il **modulo ambiente**, posizionato nelle vicinanze del gruppo Manipolatore/Nastro trasportatore, rileva la **Temperatura** (°C), l'**Umidità** (%), l'**Anidride Carbonica equivalente (eCO2 (ppm))** e le **sostanze volatili (TVOC (ppb), alcoli, aldeidi, chetoni, acidi organici, ammine, idrocarburi alifatici e aromatici)** del luogo di lavoro.





I dati elaborati tramite software memorizzati nella memoria Flash di Arduino Mega vengono visualizzati sul display locale (Fig. F2) e tramite il modulo Wi-Fi vengono inviati, con un intervallo di 15 secondi, ad un access point locale e tramite la rete internet al **Cloud ThinkSpeak** (Fig. F3) dove è stato creato un account pubblico per una successiva rielaborazione (*grafici, statistiche, diffusione online*).



Fig. F2

Tramite il seguente link è possibile consultare i dati: <https://thingspeak.com/channels/643451>

Per la consultazione tramite cellulare oppure tablet si deve scaricare un App ThingSpeak Viewer, ad esempio **ThingView** e configurarla con il canale **643451**.

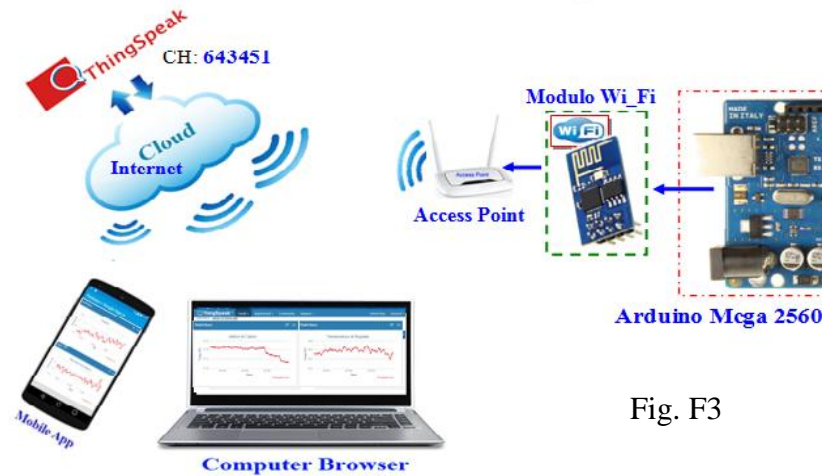
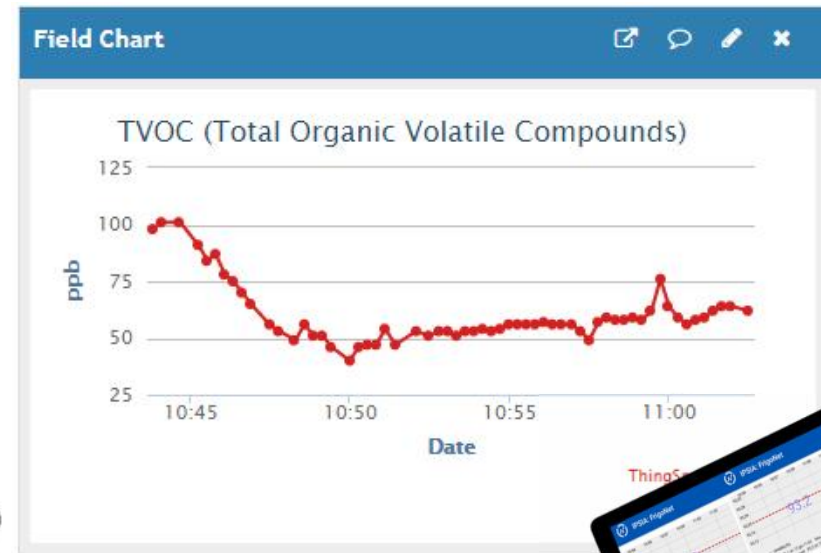
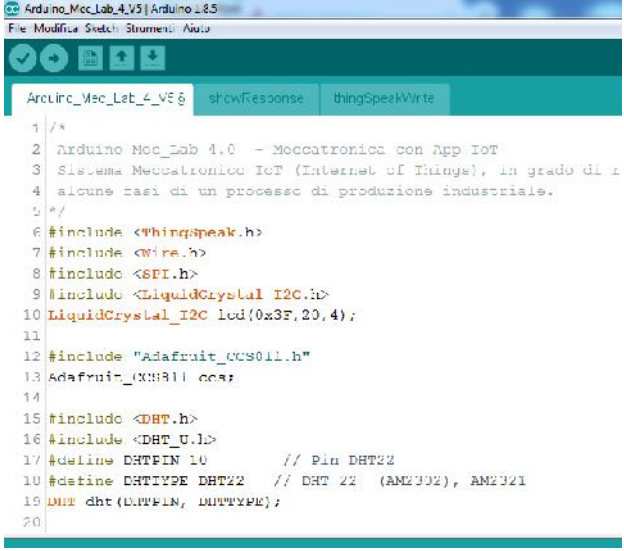


Fig. F3



Il progetto è suddiviso in una parte **Hardware** ed in una parte **Software**.

Hardware	Software
1) <b>Arduino Mega 2560 (Sistema di sviluppo programmabile)</b>	<b>Programma/Sketch</b>
2) <b>Shield Traslatore livelli 5v/24v - 24v/5v</b>	 <pre data-bbox="1339 268 1960 821"> 1 /* 2  Arduino_Mec_Lab_4_V5   Arduino_1.8.5 3  Sistema Meccatronico IoT (Internet of Things), in grado di 4  alcune fasi di un processo di produzione industriale. 5  */ 6  #include &lt;Thingspeak.h&gt; 7  #include &lt;Wire.h&gt; 8  #include &lt;SPI.h&gt; 9  #include &lt;LiquidCrystal_I2C.h&gt; 10 LiquidCrystal_I2C lcd(0x3F,20,4); 11 12 #include "Adafruit_CCS811.h" 13 Adafruit_CCS811 ccs; 14 15 #include &lt;DHT.h&gt; 16 #include &lt;DHT_U.h&gt; 17 #define DHTPIN 10 // Pin DHT22 18 #define DHTTYPE DHT22 // DHT 22 (AM2302), AM2321 19 DHT dht(DHTPIN, DHTTYPE); 20 </pre>
3) <b>Manipolatore FESTO (<i>Handling station</i>)- <i>Spostamento dei pezzi semilavorati</i></b>	
4) <b>Nastro trasportatore FESTO (<i>Conveyor station</i>)- <i>Traslazione pezzi</i></b>	
5) <b>Modulo Ambiente: CCS811 (CO2, TVOC) - DHT22 (Temp, Umidità)</b>	
6) <b>Switch Mode - Modulo Pulsanti</b>	
7) <b>Display LCD I2C 20x4</b>	
8) <b>Modulo Wi_Fi <i>ESP 8266</i> - Led IoT</b>	
9) <b>Canale ThingSpeak: <a href="https://thingspeak.com/channels/643451">https://thingspeak.com/channels/643451</a></b>	





## 2) Shield Traslatore livelli 5v/24v - 24v/5v

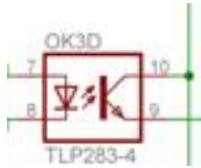
Lo Shield abbinato ad Arduino Mega permette il collegamento e la programmazione dei moduli MecLab tramite il sistema Arduino.

Gli elementi presenti nei moduli MecLab (sensori e attuatori) funzionano a 24V mentre Arduino gestisce segnali a 5V. Lo Shield effettua la traslazione dei livelli, in questo modo i segnali di Arduino sono compatibili con i segnali del MecLab.

In Fig.2.1 è riportato lo schema elettrico ed il relativo PCB, principalmente è composto

- 32 fotoaccoppiatori, 16 di Input e 16 di Output.
- 2 connettori Centronics per il collegamento dei moduli (è possibile collegare due moduli)

I fotoaccoppiatori si usano principalmente per trasferire un segnale, sia esso digitale o analogico, da un apparato ad un altro, tenendoli elettricamente isolati l'uno dall'altro.



Normalmente, un fotoaccoppiatore si presenta come un integrato plastico. Nell'interno di questo contenitore sono racchiusi

- un diodo emettitore all'infrarosso
- un fototransistor ricevente, anch'esso all'infrarosso

I due componenti, l'uno emettente e l'altro ricevente, sono separati tra loro tramite un dielettrico trasparente e questo fa sì che l'accoppiamento tra di essi sia esclusivamente ottico. Il fototransistor funziona come interruttore, la BASE viene pilotata dalla luce emessa dal diodo emettitore

- Diodo emette luce - Transistor in saturazione (Interruttore ON)
- Diodo non emette luce - Transistor Interdetto (Interruttore OFF)

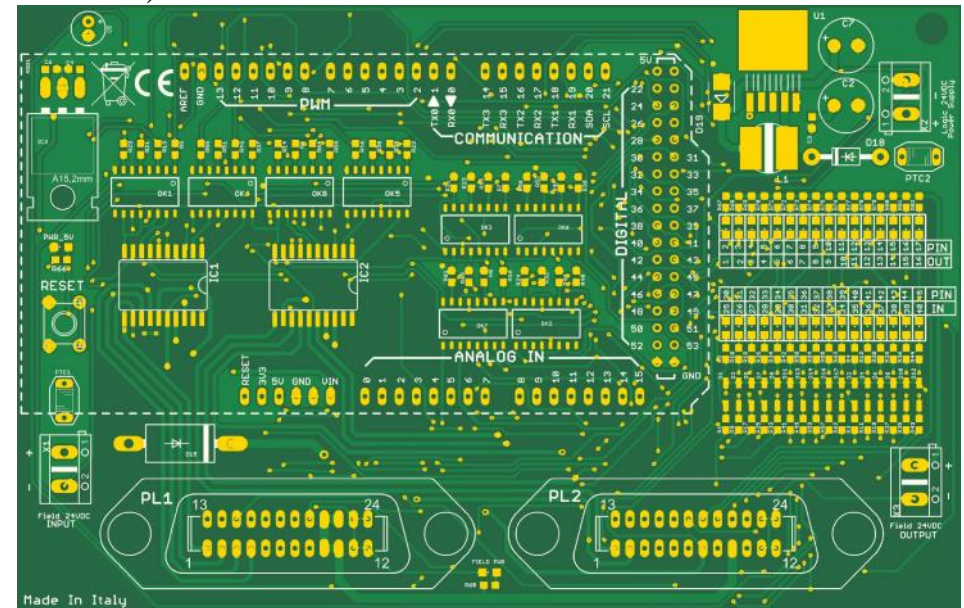
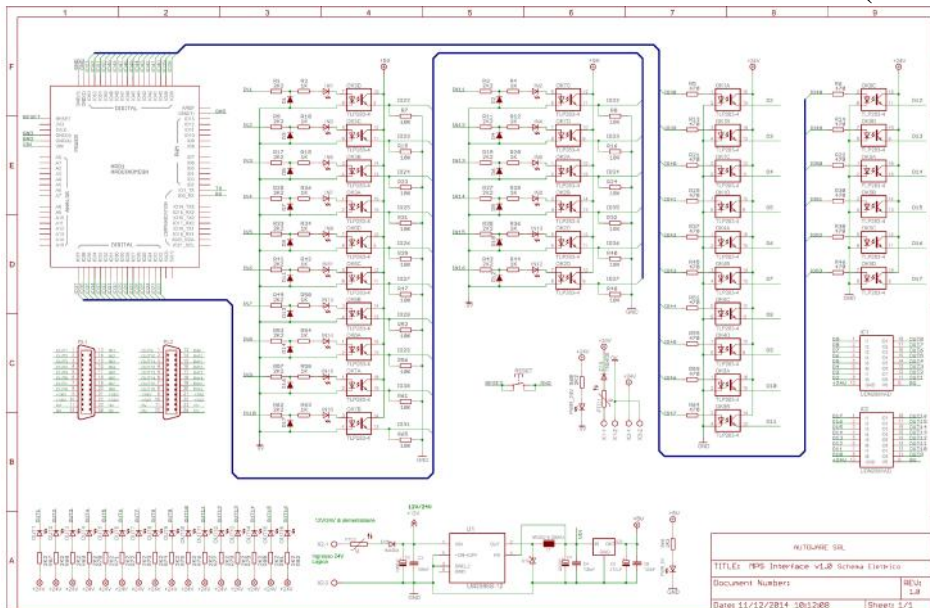
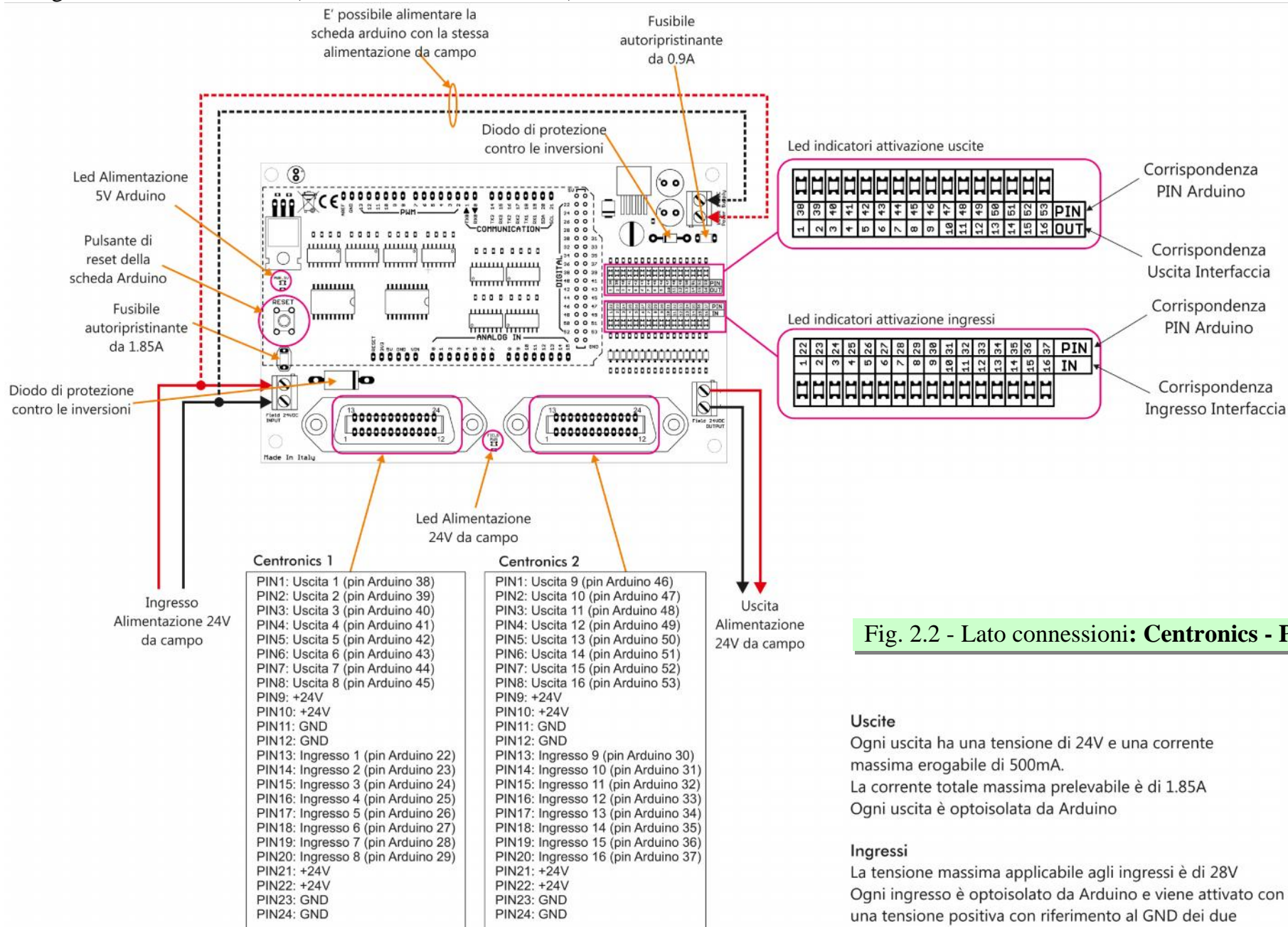


Fig. 2.1 Schema elettrico e PCB

In Fig.2.2 il lato connessioni (**Centronics - Pin Arduino**).



**Fig. 2.2 - Lato connessioni: Centronics - Pin Arduino**

#### Uscite

Ogni uscita ha una tensione di 24V e una corrente massima erogabile di 500mA.

La corrente totale massima prelevabile è di 1.85A

Ogni uscita è optoisolata da Arduino

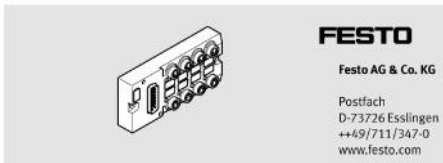
#### Ingressi

La tensione massima applicabile agli ingressi è di 28V

Ogni ingresso è optoisolato da Arduino e viene attivato con una tensione positiva con riferimento al GND dei due connettori Centronics

Il collegamento viene effettuato tramite un cavo (Centronics lato Shield Arduino, connettore Sub-D lato modulo MecLab).  
 In Fig.2.3 è riportato il distributore multipolare presente sui moduli MecLab.

**Distributore multipolare per  
 connettore M8 con 8/12 attacchi**



(it) Descrizione breve

691 696  
 0812a

Connessione M8 Posto	Connettore femmina	Occupazione dei contatti	PIN Sub-D a 15 poli	Posizione di pin e connettori femmina
0	4	Linea di segnale	1	<b>Connettore Sub-D</b> 
1	4	Linea di segnale	2	
2	4	Linea di segnale	3	
3	4	Linea di segnale	4	
4	4	Linea di segnale	5	
5	4	Linea di segnale	6	
6	4	Linea di segnale	7	
7	4	Linea di segnale	8	
8	4	Linea di segnale	9	
9	4	Linea di segnale	10	
10	4	Linea di segnale	11	
11	4	Linea di segnale	12	
0...7 o 0...11	1	24 VCC	13	<b>Connessione M8</b> 
0...7 o 0...11	3	0V	14 e 15	

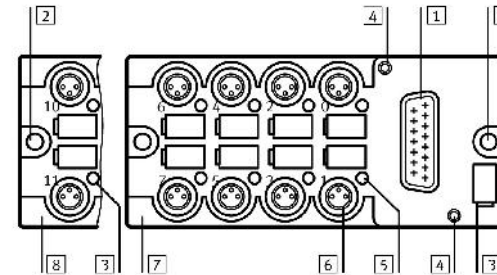


Fig. 2.3



### 3) Manipolatore Festo (*Handling station*)- - *Spostamento dei pezzi semilavorati*

Il ciclo di manipolazione è sempre presente in ogni processo, sia che si tratti di semplici operazioni di deposito, sia che riguardi complesse sequenze di assemblaggio.

I dispositivi che svolgono tali operazioni comprendono macchine diverse, dai semplici manipolatori cartesiani a 2 assi, sino ai complessi robot industriali a 6 assi.

Il manipolatore in MecLab® è realizzato con 2 cilindri pneumatici ed ha quindi 2 gradi di libertà.

Il pezzo viene bloccato da pinza ad azionamento ugualmente pneumatico.

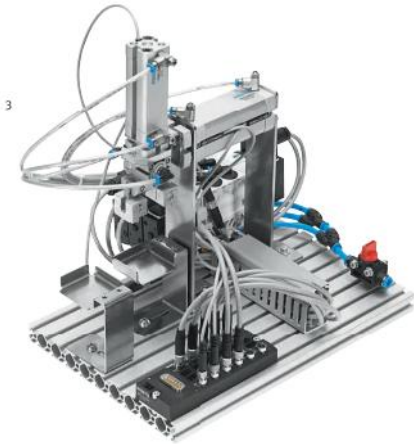
.Il manipolatore viene impiegato per trasportare i pezzi da una stazione ad un'altra; o anche per assemblare due parti dello stesso pezzo.

Programmabile tramite:

- a) Mini PLC con software Ladder
- b) FluidSim con l'interfaccia EasyPort (Programmazione e simulazione)
- c) Arduino Mega con lo Shield Arduino-MecLab (Software IDE)

**L'unità comprende:**

- Modulo manipolatore
- 3 elettrovalvole, 4 finecorsa magnetici, 2 cilindri pneumatici a guida piana
- 1 pinza pneumatica,
- connettore multi-pin
- Piastra in alluminio profilato,
- Set attrezzi, pezzi semilavorati Valigetta contenitore, vaschette dei componenti
- CD con sw FluidSIM e documentazione



In Fig. 6 è riportata la struttura del manipolatore in cui sono indicati gli elementi fondamentali

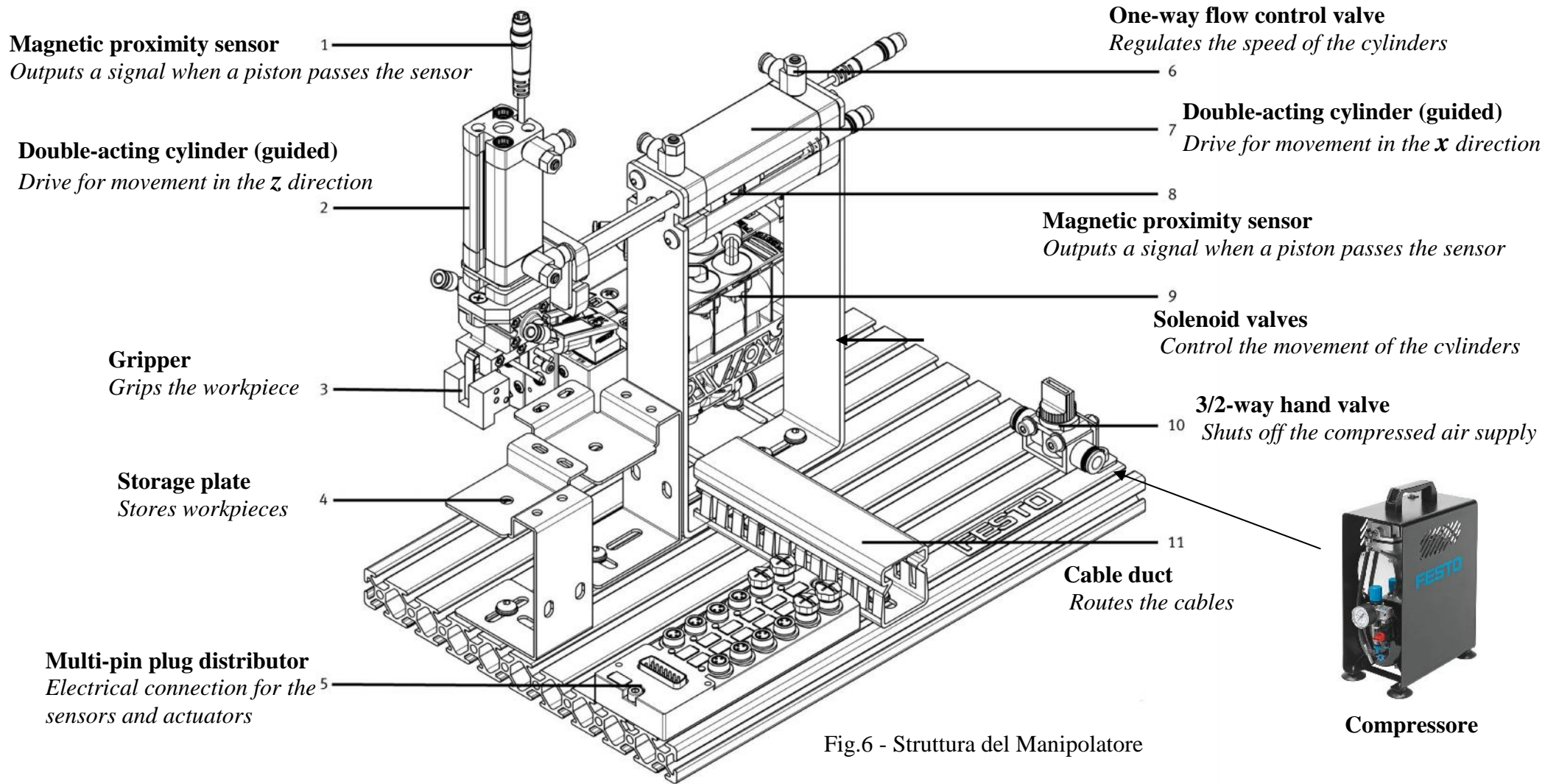


Fig.6 - Struttura del Manipolatore

Per il funzionamento occorre un compressore che va collegato al punto 10

In Fig. 7 sono riportati:

- Distributore multipolare
- Disegno della struttura dove sono evidenziate le posizioni possibili con i rispettivi codici
- Tabella di verità dei sensori (1S1, 1S2, 2S1, 2S2)

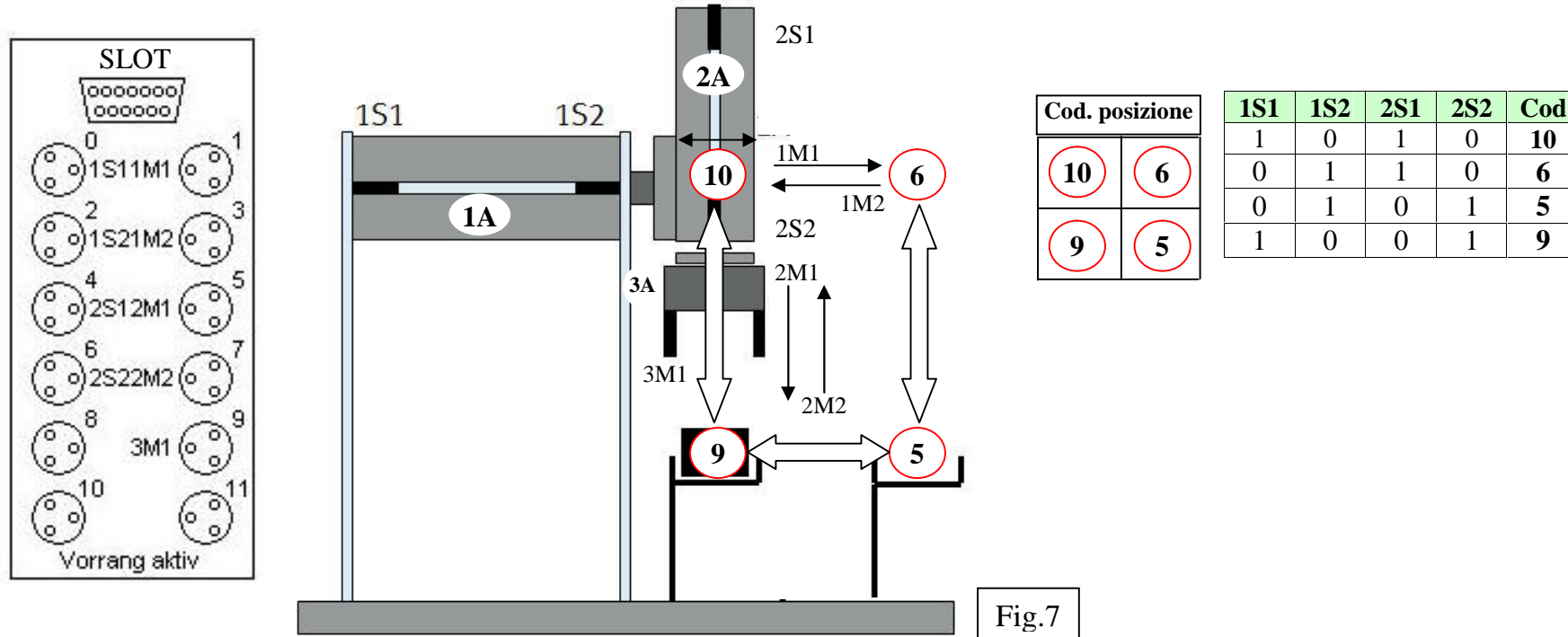



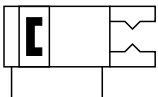

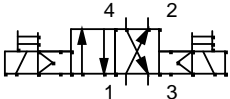

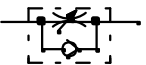
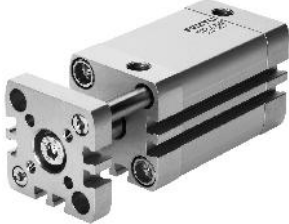
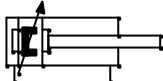



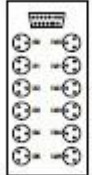

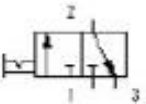

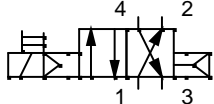

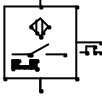
Fig.7

Il Manipolare dispone di:

- 4 **Sensori** induttivi di prossimità (1S1, 1S2, 2S1, 2S2) i sensori forniscono un livello alto ("1") oppure un livello basso ("0").  
La lettura dello stato viene effettuata con il comando *Var=digitalRead(pinsensore)*
- 2 **attuatori** (elettrovalvole) per il comando dei due cilindri a doppio effetto (1M1, 1M2, 2M1, 2M2).  
Il comando viene effettuato con il comando *digitalWrite (pinattuatore, HIGH/LOW)*
- 1 **attuatore** (elettrovalvola) per il comando della pinza (3M1)  
Il comando viene effettuato con il comando *digitalWrite (pinpinza,HIGH/LOW).*

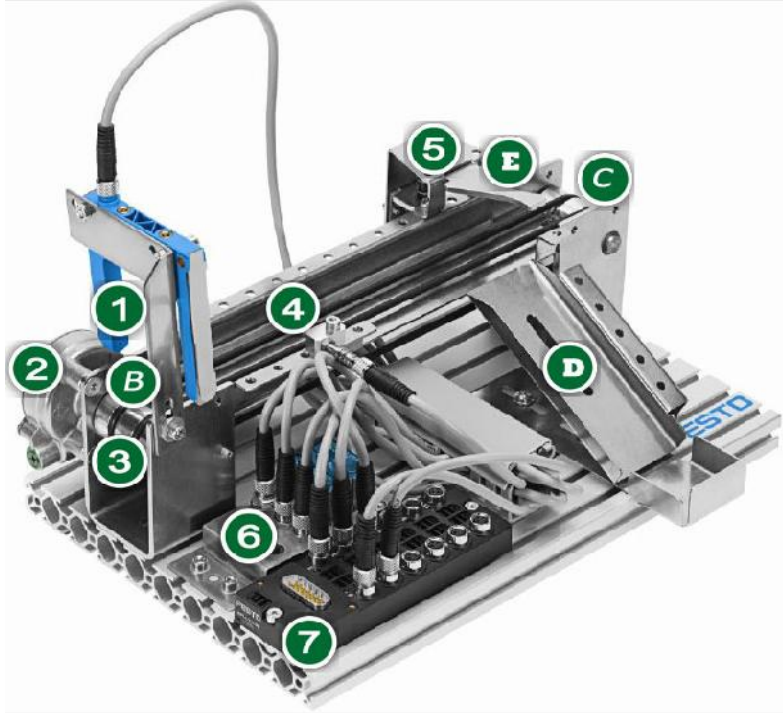


In tabella sono riportati i componenti del manipolatore con i rispettivi simboli.

N	Componente	Simbolo	Descrizione	N	Componente	Simbolo	Descrizione
1			Gripper	7			4/2-way double solenoid valve
2			One-way flow control valve	8			Double-acting cylinder
3			T-distributor for distributing the compressed air.	9			Multi-pin plug distributor, interface for connecting all actuators and sensors of the conveyor station to the control PC.
4			3/2-way stop cock for shutting off the compressed air and exhausting.				
5			4/2-way single solenoid valve				
6			Inductive proximity sensor				

#### 4) Nastro trasportatore FESTO - *Conveyor station - Traslazione pezzi sulla linea di produzione*

Il trasporto dei componenti tra due unità della linea è chiaramente funzione necessaria. Nel mondo reale vengono realizzati attraverso dispositivi non intelligenti, soprattutto i nastri trasportatori.



Questa unità di MecLab® fornisce quindi una simulazione realistica di un sistema di trasporto industriale. Il motore di azionamento del nastro è bidirezionale; in questo modo i pezzi vengono rilevati e selezionati da sensori secondo le caratteristiche. Pezzi non conformi alle specifiche vengono espulsi attraverso la slitta. Un finecorsa magnetico si può introdurre per tenere traccia della posizione del cilindro.

Programmabile tramite:

- Mini PLC con software Ladder
- FluidSim con l'interfaccia EasyPort (Programmazione e simulazione)
- Arduino Mega con lo Shield Arduino-MecLab (Software IDE)

L'unità comprende:

- Modulo nastro trasportatore a trascinamento con motore DC
- 1 elettrobobina (stop), Connettore elettrico multi-pin
- 1 Sensore induttivo, 1 barriera a sensore ottico
- Piastra in alluminio profilato
- Set attrezzi, pezzi semilavorati, valigetta contenitore Systainer
- CD con sw FluidSIM® e documentazione

Il nastro trasportatore MecLab è strutturato in uno chassis di alluminio dove sono alloggiati i sensori, gli attuatori e l'interfaccia di comando per il collegamento alla porta USB del Computer.

**1) Sensore ottico**

**2) Motoriduttore**

**3) Nastro**

**4) Sensore induttivo**

**5) Solenoide**

**6) Rele'**

**7) Multi\_Pin distributore**

**B) Inizio Nastro**

**C) Percorso C**


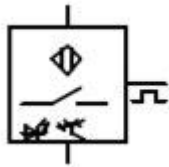





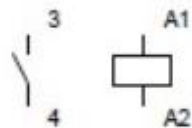

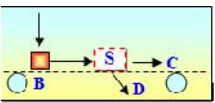

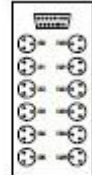

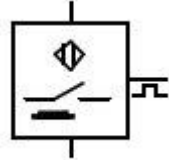
**D) Percorso D**

**E) Cannelletto**

Il Nastro Trasportatore dispone di:

- 2 **Sensori** (*induttivo, ottico*) i sensori forniscono un livello alto ("1") oppure un livello basso ("0").  
La lettura dello stato viene effettuata con il comando *Var=digitalRead(pinsensore)*
- 2 **attuatori** (*motore:nastro, solenoide:cancelletto*). L'attivazione viene effettuata con il comando *digitalWrite (pinattuatore, HIGH/LOW)*

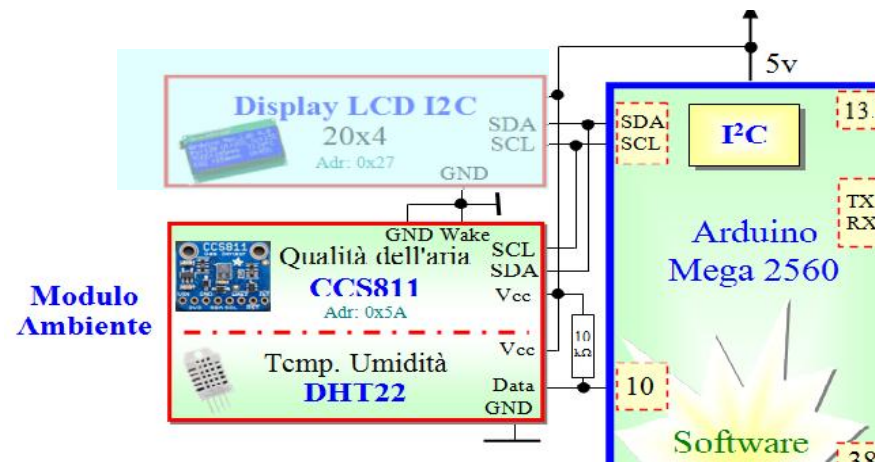
In tabella sono riportati i componenti del manipolatore con i rispettivi simboli.

N	Componente	Simbolo	Descrizione	N	Componente	Simbolo	Descrizione
1			<b>Sensore ottico</b> Opto sensor, detects any workpieces blocking the through-beam sensor's beam of light.	5			<b>Solenoide</b> Solenoid (stopper/deflector), can act as a deflector or stopper.
2			<b>Motoriduttore</b> DC geared motor, moves the belts of the conveyor and the workpieces lying on them. Can run forwards and backwards.	6			<b>Rele'</b> Relay, activates the motor, used to reverse the polarity of the motor.
3	 		<b>Nastro Trasportatore</b> Il nastro è composto di due rulli collegati con due cinghie di cui uno collegato all'albero del motoriduttore. E' possibile il movimento Avanti/Indietro.	7			<b>Multi_Pin distributore</b> Multi-pin plug distributor, interface for connecting all actuators and sensors of the conveyor station to the control PC.
4			<b>Sensore induttivo</b> Inductive sensor, can sense metal or metal-coated workpieces.				

## 5) Modulo Ambiente: CCS811 (CO2, TVOC) - DHT22 (Temp, Umidità)

Il modulo, suddiviso in due unità, permette di rilevare alcune grandezze **ambientali** (Temperatura, Umidità e qualità dell'aria (**CO2** e **TVOC**)) del luogo di lavoro.

In figura è riportato lo schema di collegamento.



**CCS811:** **eCO2** , range: 400 ÷ 8192 ppm - **TVOC** range: 0 ÷ 1187 ppb

*Anidride Carbonica Equivalente*      *alcooli, aldeidi, chetoni, acidi organici, ammine, idrocarburi alifatici e aromatici.*

**DHT22:** **Temperatura**, range: -40°C ÷ 80°C - **Umidità**. range: 0% ÷ 100%

### CCS811

Il Sensore di Qualità dell'Aria con CCS811 è un sensore di gas che può rilevare un'ampia gamma di composti organici volatili (VOC) ed è destinato al monitoraggio della qualità dell'aria interna.



Dopo essere stato collegato al microcontrollore tramite una interfaccia I2C restituirà una lettura totale dei composti organici volatili (TVOC) e una lettura dell' anidride carbonica equivalente (eCO2).

All'interno del Il CCS811 sono presenti il sensore e un piccolo microcontrollore che controlla l'alimentazione, si occupa della conversione analogico-digitale e fornisce l'interfaccia I2C.

Il sensore misura la concentrazione eCO2 entro una gamma compresa tra 400 e 8192 parti per milione (ppm) e la concentrazione TVOC (Totale del composto organico volatile) entro una gamma compresa tra 0 e 1187 parti per miliardo (ppb).

Secondo il foglio informativo fornito dal produttore il sensore può rilevare *alcooli, aldeidi, chetoni, acidi organici, ammine, idrocarburi alifatici e aromatici.*

Come ogni sensore di gas è necessaria la calibrazione per ottenere misure precise, inoltre il costruttore raccomanda di tenere acceso il sensore nuovo per 48 ore prima di utilizzarlo, durante l'utilizzo normale occorrerà attendere 20 minuti per la stabilizzazione del sensore prima di effettuare una misura.



Il Sensore di Qualità dell'Aria con CCS811 dispone di un pin di interrupt configurabile che può essere attivato quando è pronta una conversione oppure quando una lettura supera la soglia impostata dall'utente.

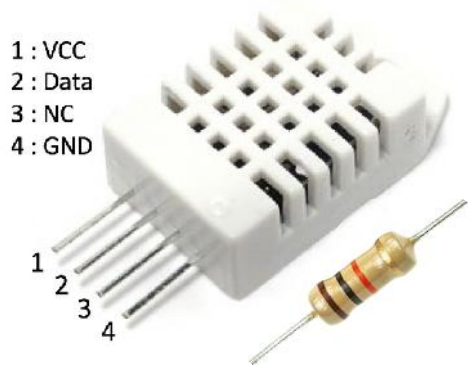
Il sensore CCS811 può effettuare una misura ogni 250 millisecondi, ogni 1 secondo, ogni 10 secondi e ogni 60 secondi.

### DHT22:

Il Sensore di Temperatura e Umidità DHT22 (conosciuto anche come RHT03) utilizza una interfaccia a singolo filo, è molto preciso e calibrato dal costruttore. Il sensore DHT22 è composto da un elemento capacitivo per la misura dell'umidità e da una NTC per la misura della temperatura, presenta un bassissimo consumo, una precisione del 2% per le misure dell'umidità e di 0,3 °C per le misure di temperatura. Per il corretto funzionamento inserire un resistore di 10KΩ di pullup tra Vcc ed il terminale data.

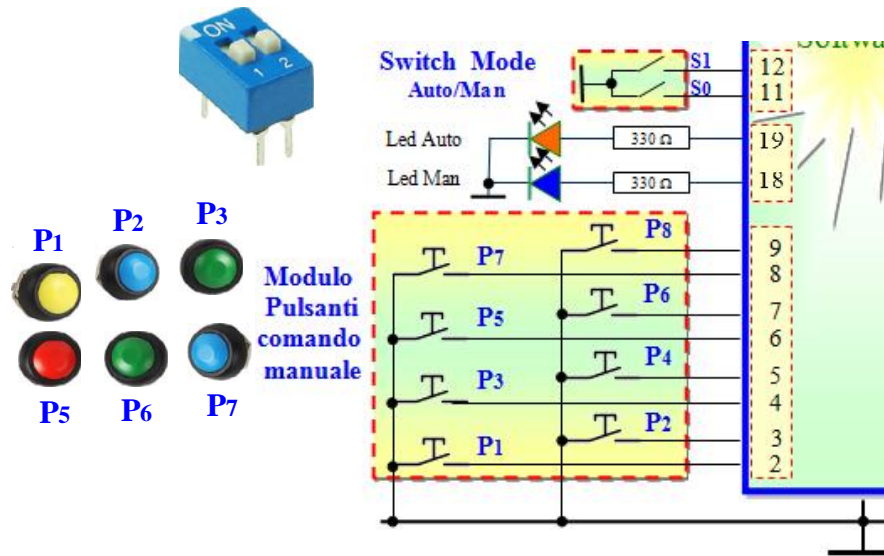
#### Caratteristiche:

- Alimentazione da 3,3Vcc a 6Vcc
- Assorbimento di 1,15mA durante la misura
- Assorbimento di 50uA a riposo
- Gamma di misura umidità: da 0 a 100% RH
- Gamma di misura temperatura: da -40°C a 80°C
- Precisione per la misura dell'umidità: ± 2% RH
- Precisione per la misura della temperatura: ± 0.5°C



## 6) Switch Mode - Modulo Pulsanti

Tramite Switch Mode si imposta la modalità di funzionamento (**Manuale/Automat**ico).



In figura è riportato lo schema di collegamento, mentre nelle tabelle le due modalità, la funzione dei singoli pulsanti e l'impostazione dei pin di Arduino Mega.

**Tabella pulsanti: Funzionamento manuale**

Switch Mode: S0=Off, S1=Off  
Led Man=ON Led Auto=OFF

Pulsante	Pin Arduino	Modalità Pin	Funzione
P1	2	INPUT_PULLUP	Chiude pinza
P2	3	INPUT_PULLUP	Apri pinza
P3	4	INPUT_PULLUP	Cilindro-Ori-Esterno
P4	5	INPUT_PULLUP	Cilindro-Ori-Interno
P5	6	INPUT_PULLUP	Cilindro-Vert-Basso
P6	7	INPUT_PULLUP	Cilindro-Vert-Alto
P7	8	INPUT_PULLUP	Start Nastro
P8	9	INPUT_PULLUP	Stop Nastro

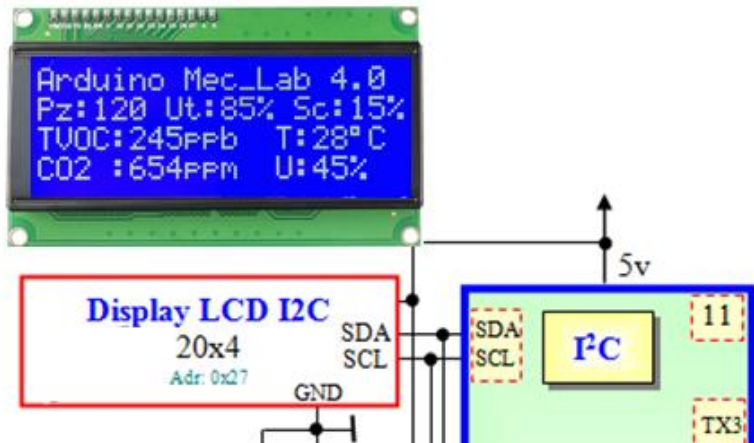
**Tabella pulsanti: Funzionamento automatico**

Switch Mode: S0=On, S1=Off  
Led Man=OFF Led Auto=ON

Pulsante	Pin Arduino	Modalità Pin	Funzione
P1+P8	2,9	INPUT_PULLUP	Start
P4	5	INPUT_PULLUP	Stop

## 7) Display LCD I2C 20x4

Il display, come evidenziato nella figura, permette la visualizzazione dei seguenti dati:



- 1) **Pz:** Pezzi semilavorati
- 2) **Ut:** Percentuale pezzi utili
- 3) **Sc:** Percentuale scarto
- 4) **TVOC:** Total Organic Volatile compounds (ppb)
- 5) **CO2:** Anidride Carbonica equivalente (ppm)
- 6) **T:** Temperatura ambiente (°C)
- 7) **U:** Umidità ambiente (%)

Il Modulo LCD dispone di 4 righe e 20 colonne è basato sul controller HD44780; dotato di retroilluminazione blu, caratteri bianchi, regolazione del contrasto e di un'interfaccia di comunicazione I2C.

Lo schermo a cristalli liquidi (LCD) è un display a schermo piatto, una visualizzazione elettronica o un video che utilizza le proprietà modulanti della luce dei cristalli liquidi.

I cristalli liquidi non emettono direttamente la luce. Come mostrato nell'immagine, un'importante caratteristica di questo modulo LCD è l'interfaccia di comunicazione I2C integrata posta sul retro che ne rende estremamente semplice l'utilizzo con Arduino. Il modulo è dotato di 4 pin (Alimentazione: Vcc, Gnd) e (Dati: SDA, SCL)

Questo modulo permette di comunicare con un display LCD mediante il protocollo I2C che impegna quindi due sole porte. (**Dati: SDA, SCL**) oltre ai due terminali di alimentazione (**VCC, GND**).

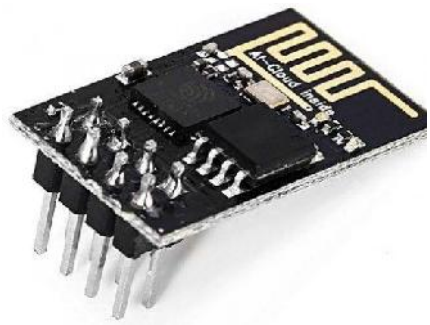
Sul modulo è presente un potenziometro per la regolazione del contrasto ed un jumper rimovibile per l'attivazione della retroilluminazione. Se il jumper viene rimosso e tra i due pin viene inserita una resistenza, si può modificare l'intensità di retroilluminazione (ad esempio con una resistenza da 470 ohm 1/4w l'intensità si dimezza).

### Settaggio dell'indirizzo del modulo

Sono inoltre presenti tre connessioni denominate A1, A2 e A3 per il settaggio (a 3-bit) dell'indirizzo I2C tra 0x20 e 0x27. L'indirizzo di default è 0x27 (A0, A1 e A2 lasciati aperti). La chiusura di una coppia si effettua collegando tra loro le due rispettive piazzole. E' ovvio che tale indirizzo hardware deve coincidere con l'indirizzo I2C nel software/libreria di gestione del Modulo. Il display viene gestito tramite la libreria **LiquidCrystal\_I2C.h**.

## 8) Modulo Wi-Fi ESP 8266 - LED IoT

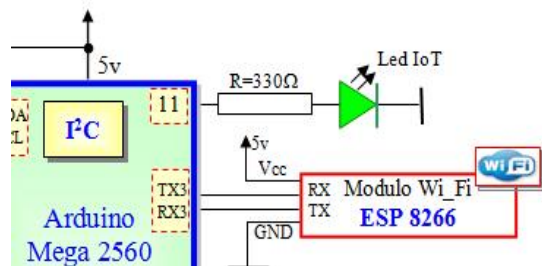
ESP8266 e' un chip dotato di un microprocessore con: 160 kbyte di RAM (64 dei quali riservati alle istruzioni), una memoria flash da 512 kbyte (e fino a 4 mega byte, in funzione del modello), funzioni aggiuntive, tra le quali 16 pin GPIO e gestione di svariati protocolli di comunicazione un firmware in grado di gestire la connessione WIFI e i comandi AT. **In questo progetto viene utilizzato per la connessione Wi-Fi**, viene gestito con i comandi AT ed è collegato sulla **Serial3** (RX3, TX3) di Arduino Mega.



### LED IoT

Il **LED verde**, segnala la trasmissione dei dati al Cloud ThingSpeak.

**LED ON :** inizio trasmissione  
**LED OFF:** fine trasmissione





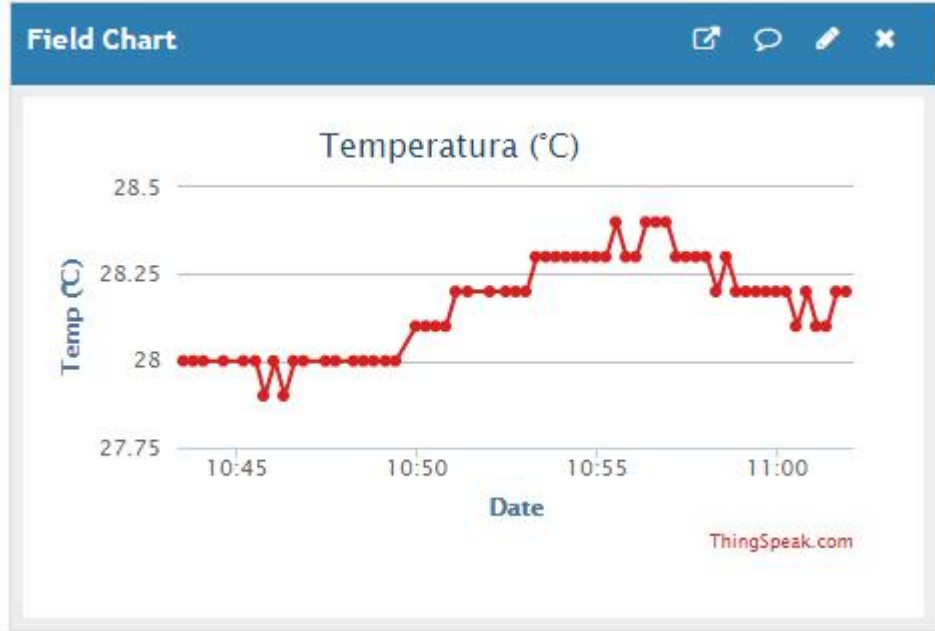
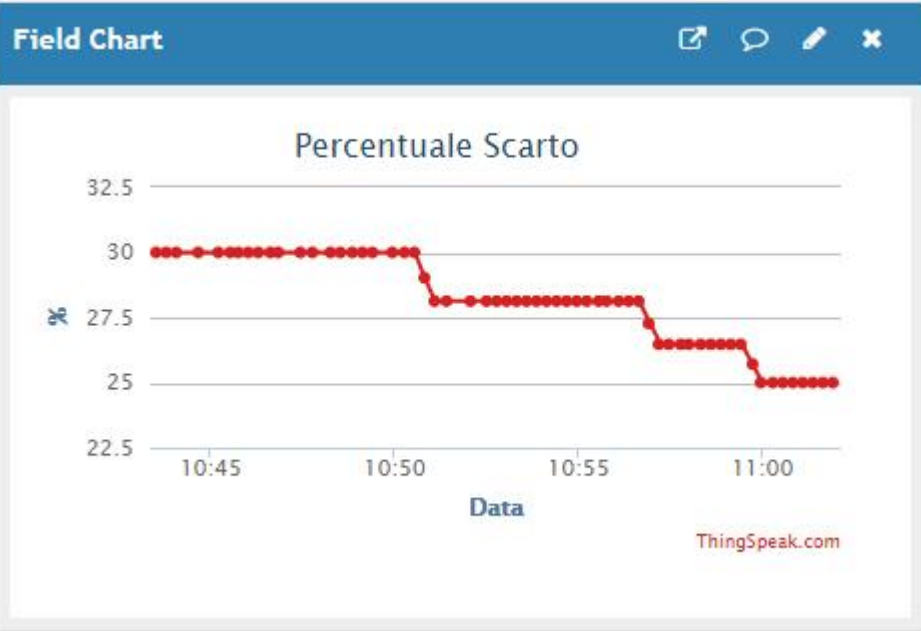
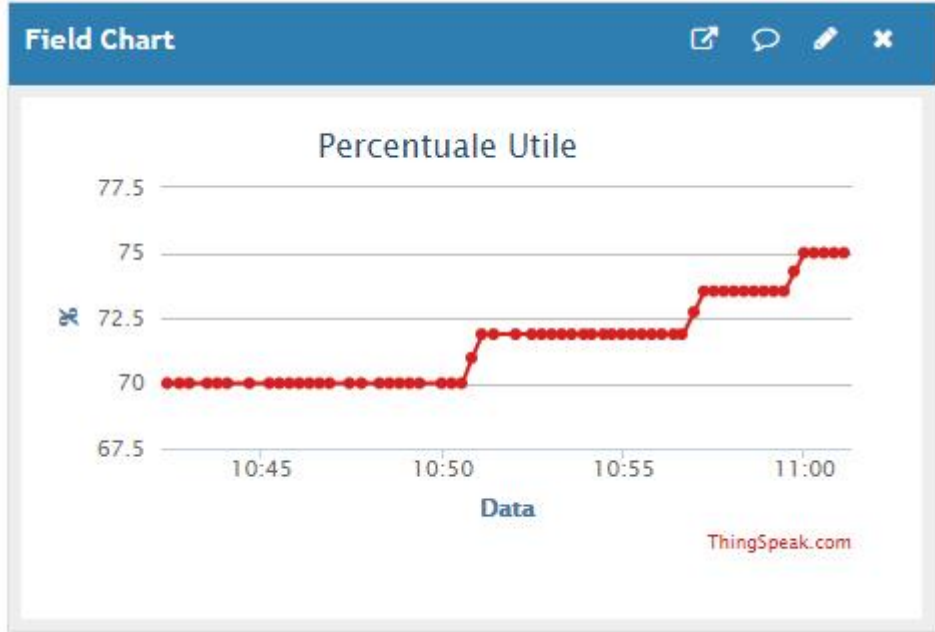
9) **Canale ThingSpeak:** <https://thingspeak.com/channels/643451>

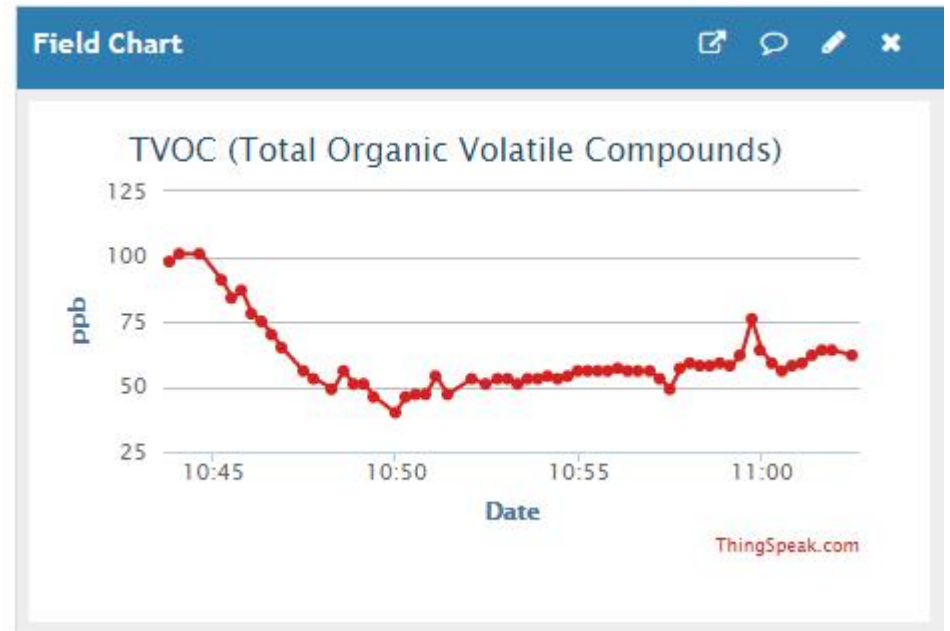
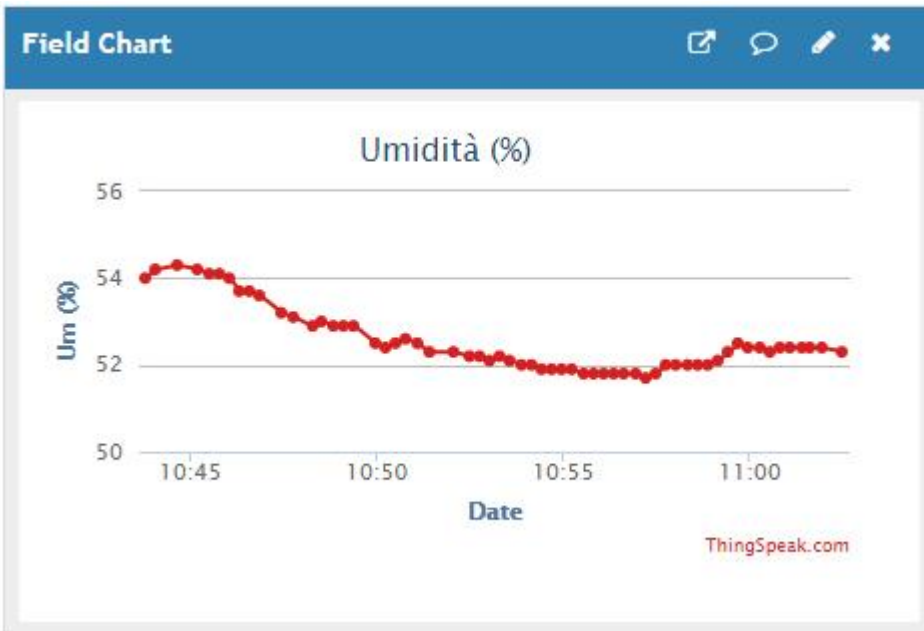
Tramite il seguente link è possibile consultare i dati: <https://thingspeak.com/channels/643451>

Per la consultazione tramite cellulare oppure tablet si deve scaricare un App ThingSpeak Viewer, ad esempio ThingView e configurarla con il canale 643451.

In figura è riportato lo **screenshot** del test effettuato il **20-06-2019**

The screenshot shows the ThingSpeak website interface for a channel named 'Arduino Mec\_Lab 4.0'. The browser address bar displays 'https://thingspeak.com/channels/643451'. The navigation bar includes the ThingSpeak logo and menu items: Canali, Applicazioni, Community, Support, Commercial Use, How to Buy, Account, and Logout. Below the navigation bar, the channel title 'Arduino Mec\_Lab 4.0' is displayed, along with social media sharing buttons for Watch, Tweet, Like (0), and Condividi. The channel details section shows the ID '643451', author 'arduinoipsia', and access level 'Public'. A description 'Meccatronica con App IoT' and tags 'ipsia, arduino, esp8266' are also visible. A menu bar below the details includes 'Private View', 'Public View', 'Channel Settings', 'Sharing', 'Chiavi API', and 'Data Import / Export'. At the bottom, there are buttons for 'Add Visualizations', 'Add Widgets', 'Export recent data', 'MATLAB Analysis', and 'MATLAB Visualization'.





## Software di gestione (Estratto)

Di seguito è riportato un estratto del software (Sketch) di gestione

```
/*  
  Arduino Mec_Lab 4.0 - Meccatronica con App IoT  
  Sistema Meccatronico IoT (Internet of Things), in grado di riprodurre in scala  
  alcune fasi di un processo di produzione industriale.  
*/  
#include <ThingSpeak.h>  
#include <Wire.h>  
#include <SPI.h>  
#include <LiquidCrystal_I2C.h>  
LiquidCrystal_I2C lcd(0x3F,20,4);  
#include "Adafruit_CCS811.h"  
Adafruit_CCS811 ccs;  
#include <DHT.h>  
#include <DHT_U.h>  
#define DHTPIN 10 // Pin DHT22  
#define DHTTYPE DHT22 // DHT 22 (AM2302), AM2321  
DHT dht(DHTPIN, DHTTYPE);  
String apiKey = "0EJJS59YG5HLQ0F6"; // replace with your channel's thingspeak WRITE API key  
String ssid="WIND-WiFi"; // Wifi network SSID  
String password ="j7e1xxx"; // Wifi network password  
boolean DEBUG=true;  
boolean OS1,OS2; //variabili fine corsa cilindro orizzontale  
boolean VS1,VS2; //variabili fine corsa cilindro verticale  
boolean opto, ind, optoA; //variabile sensore ottico, sensore induttivo del nastro  
long tempo,tempo1,intervallo=15000.0,intervallo1=123;  
// Assegnazione Pin di Arduino  
byte pin1M1=46, pin1M2=47,pin1S1=30,pin1S2=31; // Orizzontale (Sensori e cilindri)  
byte pin2M1=48, pin2M2=49,pin2S1=32,pin2S2=33; // Verticale (Sensori e cilindri)  
byte pinza=50; // Gripper (Pinza)  
byte PO1=4,PO2=5,PV1=6,PV2=7;// pin pulsanti Cilindri  
byte Pg1=2,Pg2=3; // pin pulsanti Pg=Pinza  
byte PNA=8,PNS=9; // Pin pulsanti nastro  
byte posizione=0; // Variabile cod. posizione cilindri  
int ritardo=1500,CO2,TVOC;
```



```

float TempMecLab, UmMecLab ; // Variabili Temperatura-Umidità MecLab
float pezzi=0,pm=0,pnm=0,metallo=0,nonmetallo=0;
byte pinopto=23, pinind=22; //sensore ottico conta pezzi e induttivo

void setup()
{
  DEBUG=true; // enable debug serial
  Serial.begin(9600);
  Serial3.begin(115200); // Your esp8266 module's speed is probably at 115200.
  showResponse(1000);
  //Inizializzazione CCS811
  if(!ccs.begin()){
    Serial.println("Failed to start sensor! Please check your wiring.");
    while(1);
  }
  //calibrate CCS811 temperature sensor
  while(!ccs.available());
  float temp = ccs.calculateTemperature();
  ccs.setTempOffset(temp - 25.0);
  //Fine calibrate CCs811 temperature sensor
  dht.begin();///Inizializzazione DHT22
  lcd.init(); lcd.clear();lcd.backlight();
  lcd.setCursor(0,0); lcd.print("Arduino Mec_Lab 4.0");
  lcd.setCursor(0,1); lcd.print("Pz:  Ut:  % Sc:  %");
  lcd.setCursor(0,2);lcd.print("TVOC:  ppb T:  ");lcd.print(char(223)); lcd.print("C");
  lcd.setCursor(0,3);lcd.print("CO2 :  ppm U:  %");

  // ===== Connessione Access Point Wi-Fi =====

  Serial3.println("AT+CWJAP"); //
  showResponse(1000);
  Serial3.println("AT+CWMODE=1"); // set esp8266 as client
  showResponse(1000);
  Serial3.println("AT+CWJAP=\\\"+ssid+\\\",\\\"+password+\\\""); // set your home router SSID and password
  showResponse(5000);
  if (DEBUG)

```

```

{
  lcd.setCursor(18,1); lcd.print("OK");
}
// =====
pinMode (pin1M1,OUTPUT);pinMode (pin1M2,OUTPUT);pinMode (pin2M1,OUTPUT); pinMode (pin2M2,OUTPUT);pinMode(pinza,OUTPUT);
pinMode(PO1,INPUT_PULLUP);pinMode(PO2,INPUT_PULLUP); pinMode(PV1,INPUT_PULLUP);pinMode(PV2,INPUT_PULLUP);
pinMode(Pg1,INPUT_PULLUP);pinMode(Pg2,INPUT_PULLUP); // Pulsanti comando Pinza
pinMode(PNA,INPUT_PULLUP);pinMode(PNS,INPUT_PULLUP); // Pulsanti comando Nastro
pinMode(pinopto,INPUT);
pinMode (22,INPUT);
pinMode (38,OUTPUT);pinMode (39,OUTPUT);
OS1=digitalRead(pin1S1); OS2=digitalRead(pin1S2);
VS1=digitalRead(pin2S1); VS2=digitalRead(pin2S2);
posizione=8*OS1+4*OS2+2*VS1+1*VS2;

// Imposta i cilindri nella posizione iniziale (Cod.10)
digitalWrite(pin2M1,LOW); delay(500);digitalWrite(pin2M2,HIGH); delay(500);
digitalWrite(pin1M1,LOW); delay(500);digitalWrite(pin1M2,HIGH); delay(500);
digitalWrite(pin2M2,LOW); digitalWrite(pin1M2,LOW); digitalWrite(pinza,LOW);
delay(ritardo); tempo=millis();
}

void loop()
{
// Inizio Lettura CCS811
if(ccs.available()){
  float temp = ccs.calculateTemperature();
  if(!ccs.readData())
  {
    CO2=ccs.getCO2(); TVOC=ccs.getTVOC();
    lcd.setCursor(5,2);lcd.print("  "); lcd.setCursor(5,2);lcd.print(TVOC,DEC);
    lcd.setCursor(5,3);lcd.print("  ");lcd.setCursor(5,3);lcd.print(CO2,DEC);
  }
  else{
    Serial.println("ERROR!");
    //while(1);
  }
}

```

```

}
//delay(500);
// Fine Lettura CCS811
// Lettura Temperatura - Umidità con DHT 22
TempMecLab = dht.readTemperature();// Lettura Temperatura
UmMecLab = dht.readHumidity(); // Lettura Umidità
lcd.setCursor(15,2);lcd.print(" "); lcd.setCursor(15,2);lcd.print(TempMecLab,1);
lcd.setCursor(15,3);lcd.print(" ");lcd.setCursor(15,3);lcd.print(UmMecLab,0);
//-----
.....
.....
.....

if (millis()-tempo>=intervallo)
{

// TX dati al canale thingspeak
thingSpeakWrite(pezzi,pm,pnm,TempMecLab,UmMecLab,TVOC,CO2); tempo=millis();
//lcd.setCursor(18,1); lcd.print(" ");
}

}

```

---

## Upgrade - A.S. 2019-2020

- 1) Estensione della linea di produzione con altri moduli FESTO
- 2) Espansione del modulo ambiente con sensori specifici (*Sensore di fiamma, sensori di gas, sensori di sostanze volatili*)
- 3) Gestione allarmi con attuatori acustici e luminosi
- 4) Sostituzione del modulo ESP8266 con **Arduino MKR4000** (*elaborazione dati di prod., dati ambientali, TX- Cloud, visualizzazione su monitor 19"*)
- 5) Sostituzione del display LCD 20x04 con Monitor da 19"

## Sitografia

- a) [www.arduino.cc](http://www.arduino.cc)      b) [www.festo-didactic.com](http://www.festo-didactic.com)      c) [www.adafruit.com](http://www.adafruit.com)      d) [www.sparkfun.com](http://www.sparkfun.com)