





**Dispositivo per il
monitoraggio della
conservazione
alternativa del cibo**

6

Dispositivo per il monitoraggio della conservazione alternativa del cibo

Il dispositivo è collocato nell'ambito "Conservazione cibo alternativa tramite azione batterica" sfrutta il processo di trasformazione della materia alimentare tramite fermentazione. Raccolte le informazioni (riassunte nell'analisi dei processi qui di seguito) relative a tre tipologie di fermentazione lattica: infusione, salamoia e scolarura è stato progettato un apparato in grado di monitorare i tre processi basandosi sul controllo dei parametri sensibili che hanno in comune: temperatura interna, temperatura esterna e livello di acidità. Il dispositivo rende possibile per il Soggetto il monitoraggio di questi dati in modo che possa intervenire sul processo per portarlo a termine con successo.

Le informazioni vengono raccolte dalla scheda di controllo Arduino configurabile appositamente dal Soggetto; in seguito vengono visualizzati sul display sotto forma di un grafico la cui scala è dimensionale e controllabile in relazione alle tempistiche del processo.

La struttura del dispositivo è adattabile a contenitori generici presenti nell'ambito domestico, permette l'immobilizzazione del cibo per evitare la sua ossigenazione, trattandosi di processi anaerobici, e il suo mescolamento tramite il componente "asta-mestolo".

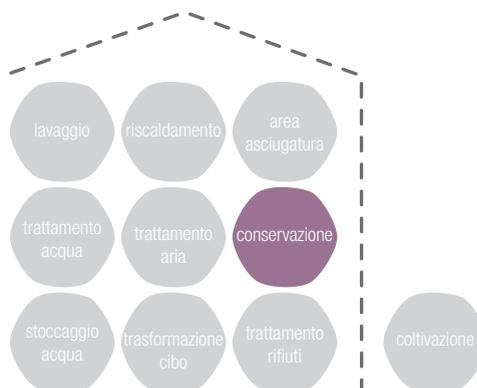
La sua conformazione "rudimentale" è facilmente riproducibile, anche attraverso pratiche DIY e riconosce la sua ragion d'essere nella caratteristica distintiva del "prodotto facilitatore".

Il monitoraggio del processo è stato inteso come "passivo", il dispositivo non produce le condizioni necessarie al suo svolgimento ma si limita al suo controllo, delegando questo compito al Soggetto che può sfruttare pratiche riconducibili alla cultura peculiare del territorio in cui si trova. Inoltre il Soggetto si relaziona con altri ambiti domestici valorizzando scarti che provengono da altri processi: come ad esempio la temperatura di altri dispositivi dispersa nell'ambiente interno.

INDICE DELLA SEZIONE

236	6.1	La conservazione alternativa del cibo.
242	6.2	Il concept.
245	6.3	Il prototipo.
247	6.4	Test prototipo. Kefir.

CLASSIFICAZIONE PROGETTO



Ambito Domestico



Dispositivo indipendente
Parte di un sistema
Dialogo con il territorio
Valorizzazione output

Relazione Sistemica



Facilitatore
Contestualizzato
Declinabile

Rapporto con il Territorio

6.1 La conservazione alternativa del cibo.

La cultura legata ai processi di trasformazione del cibo nel corso del XX secolo è passata dall'essere un sapere locale fatto di ricette e metodi che richiedono manualità e profonda esperienza, a una pratica industriale che si appropria del processo e si fa carico della trasformazione, lasciando visibile solamente il risultato finale.

Il cibo, materia rinnovabile per eccellenza, viene gestito dagli utilizzatori finali solo con pochi accorgimenti riguardanti preparazione, cottura e conservazione. In casa sono pressoché inesistenti le pratiche che stanno a metà tra la conservazione e la trasformazione, vale a dire i processi come la fermentazione e la lievitazione che da sempre hanno un ruolo centrale per ottenere materia prima per la cucina.

L'aspetto interessante di questi processi è rilevante sotto il punto di vista sistemico e sta nella capacità di rigenerare in modo autonomo prodotti di scarto. Questa valorizzazione avviene normalmente grazie all'azione di batteri o funghi, tramite un bassissimo utilizzo di energia e con tempi piuttosto dilatati.

Le opportunità sono numerose, questi processi potenzialmente coinvolgono una grande varietà di cibi e i loro risultati sono apprezzabili come metodi alternativi per la conservazione a lungo termine.

Attualmente la conservazione alimentare in ambito domestico è soddisfatta dall'elettrodomestico frigorifero. In questo dispositivo sono confluite tutte le funzioni relative all'ambito della conservazione portate a termine in maniera "intensiva": l'abbattimento della temperatura, ad un livello molte volte inferiore al necessario, allinea e banalizza tutte le peculiarità relative alla conservazione di prodotti differenti riducendole all'unica e sterile esigenza della refrigerazione.

Il fattore tempo nel lungo periodo è invece gestito dal congelatore, dispositivo ausiliario le cui ricadute a

livello ambientale sono molto pesanti.

Prodotti fermentati come il pane e la birra venivano già sicuramente prodotti dai Sumeri e dagli Egizi.

In realtà la fermentazione è una forma di deterioramento che produce come risultato un alimento che, per qualche ragione, ha proprietà più desiderabili delle materie prime o ingredienti utilizzati per produrli. Spesso l'avvio delle fermentazioni è inevitabile: basta creare condizioni anche lievemente selettive (per esempio grazie all'aggiunta di piccole quantità di sale, con la creazione di un ambiente anaerobio, per aggiunta di alcune sostanze inibitorie) per favorire lo sviluppo degli agenti delle fermentazioni al posto degli agenti di deterioramento. Spesso gli agenti delle fermentazioni, oltre a competere efficacemente con i microrganismi indesiderati, rendono l'ambiente sfavorevole per lo sviluppo e la sopravvivenza di questi ultimi e possono produrre sostanze ad azione antimicrobica.

Ancora oggi molti prodotti fermentati, anche nei paesi sviluppati, sono ottenuti utilizzando tecniche tradizionali, basate sulla presenza naturale della microflora desiderata e sull'uso di materie prime non pastorizzate. D'altra parte altri prodotti, come molti formaggi e lattici fermentati, vengono prodotti in condizioni altamente standardizzate e con l'aggiunta di colture di avviamento (starter) contenenti i microrganismi desiderati, selezionati e spesso migliorati geneticamente. Anche se si usa il termine fermentazioni non tutti i processi impiegano microrganismi anaerobi e quindi veri processi fermentativi.

Ripensare e attualizzare prodotti, per lo più di origine orientale, come il kefir, il miso, l'amasake, ma anche i più familiari yogurt, pane, birra e formaggi, può essere una opportunità per evolvere il nostro modello di consumo del cibo.

Questo orizzonte, quasi del tutto inesplorato se non dal punto di vista dell'autoproduzione casalinga, può essere ricco di spunti per il Soggetto consapevole che vuole mantenere sotto controllo la qualità del proprio cibo; inoltre l'interesse può essere trasmesso anche all'industria dell'elettrodomestico, che può osservare questo processo e spingersi verso un progetto per la conservazione alternativa, non un ennesimo elettrodomestico la cui forma e funzionamento sono standardizzati e globalizzati, ma un sistema di componenti progettato sulla base delle peculiarità culinarie di un territorio e che sfrutti processi e metodi direttamente derivanti dalla cultura materiale di quella comunità.

Le fermentazioni conosciute nel Mondo Occidentale sono relativamente poche se paragonate a quelle del resto del pianeta, esse sono dedicate soprattutto a svolgere un ruolo preminente nella sterilizzazione e nella conservazione di cibi o bevande, oppure per garantire un discreto miglioramento gustativo e di digeribilità, come avviene ad esempio nel pane.

Nel resto del mondo (Asia, America Latina, Africa) hanno invece una tradizione più forte, mantenuta ancora oggi, anche a livello industriale. In questi casi i tipi di fermentazione sfruttati sono moltissimi e vengono realizzati usando gli stessi agenti di fermentazione per sostanze diverse (come nei casi di sorgo (durrha) e mais) oppure diversificando sia gli agenti che i cibi da fermentare (quindi batteri o diversi

miceti per sostanze diverse quali legumi, vegetali, pesci, carni, ma soprattutto cereali: sesamo, miglio, riso, soia, teff). Queste fermentazioni sono usate prevalentemente per migliorare il contenuto nutrizionale delle sostanze di origine (proteine, e vitamine), producendo sostanze batteriologicamente pure (in termini di commestibilità), oltre che di sapore gradevole e durature nel tempo.

Figura 72.
Birra di millet. Caroon.

Figura 73.
*Bimba messicana
che beve il pozol*



Approfondimento. I cibi basati sulla fermentazione dal mondo.

Africa

Ogi. Sudafrica. Crema poco densa ottenuta da mais e sorgo.



Mawe. Africa Occidentale (costa atlantica). Pasta acida di farina di mais.



Mahewu. Sudafrica. Crema densa da mais e farina di frumento.



Injera. Africa Orientale. Focaccina da forno, spugnosa ed acida di farine di Teff e Frumento



Kishk. Egitto. Pasta acida da frumento in grani, e latte.



Kissra. Sudan. Pane basso, acido, da farina di sorgo.



Banku. Ghana. Pane acido da mais e Cassava.



Kenkey. Ghana. Panetti di sola pasta di mais fermentata.



Chicha. *Zona andina.* Bevanda a basso/medio livello alcolico, acida.



Pozol. *Messico ed America centrale.* Fermentazione dalla farina di mais.



America

Tesguino. *America centrale.* Bevanda acida, alcolica, da grani di mais.



Atole. *Messico.* Crema acida da granella umida di mais.



Chu. *Cina.* Agente fermentatore utilizzato in svariati modi.



Nuruk. *Corea.* Pastone di riso, grano, orzo fermentato.



Asia

Meju. *Corea.* Pastone di granella di soia, fermentata.



Koji. *Giappone.* Pastone tipo polenta da granella o farina di riso e grano.



Approfondimento. I cibi basati sulla fermentazione dal'Europa.

Bubod. *Filippine.* Pastone fermentato di riso da granella o farina.



Marchaa. *India.* Torta bassa di riso, fermentazione micetica.



Ragi. *Indonesia.* Panetti compatti di riso, fermentazione micetica.



skyr



pickled herring



sauerkraut



pissalat



machetto





6.2 Il concept.

Le ragioni che hanno animato questo progetto risiedono proprio nell'obiettivo di valorizzare adeguatamente una tale varietà di processi da sfruttare come metodi alternativi per la conservazione. In un Sistema Casa gli ambiti che hanno a che fare con la risorsa cibo sono molto importanti e recuperare questi tipi di processi, direttamente collegati con le

usanze culinarie della comunità o con le materie prime alimentari presenti sul territorio, offre delle alternative molto interessanti ai processi di conservazione attuali che, come sottolineato in precedenza, vengono riduttivamente svolti con l'abbattimento della temperatura nel frigorifero o nel surgelatore.

Figura 74.
Monitoraggio
fermentazione batterica

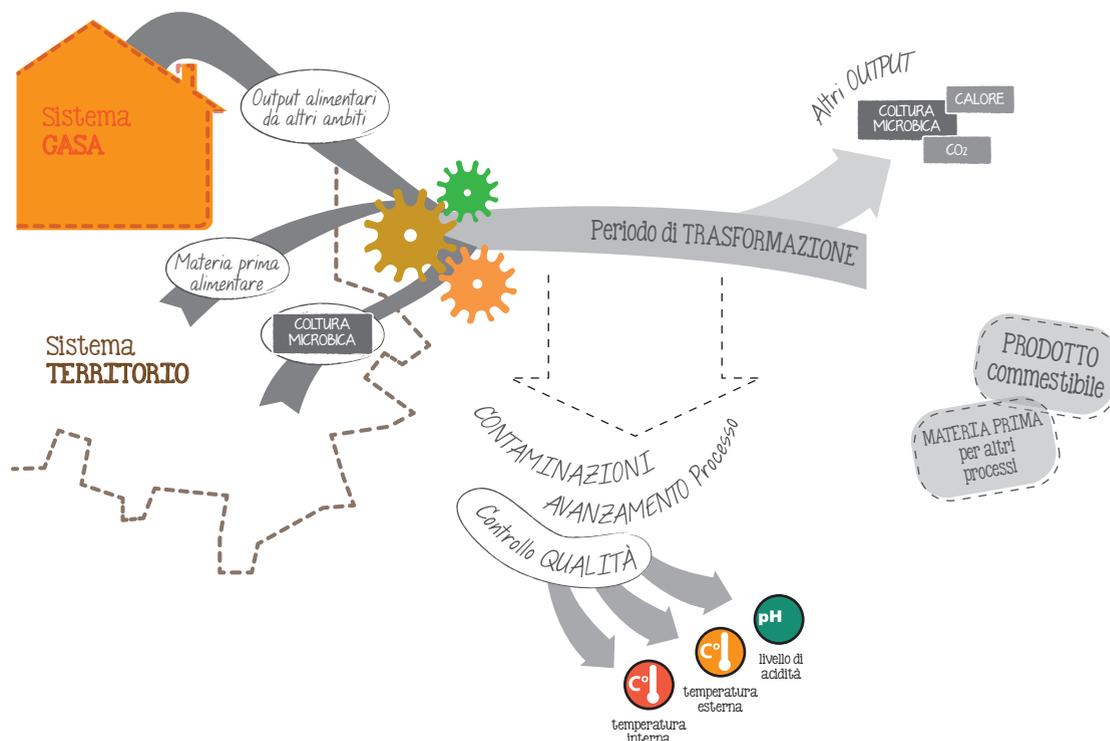


Figura 75.
I tre processi controllati dal
dispositivo messi a confronto

Il dispositivo progettato fornisce al Soggetto un semplice aiuto per poter mettere in pratica, o riscoprire, queste tecniche; consiste in un'interfaccia in grado di monitorare alcuni parametri direttamente riconducibili al processo. Questa sua funzione cerca di supplire alla perdita del patrimonio conoscitivo e culturale su cui si basano questi processi proprio per poter aiutare l'utilizzatore nella sua riscoperta fornendogli un feedback immediato a proposito dello svolgimento del procedimento. Questo dispositivo in realtà può essere visto come un sistema di componenti ridotto all'essenziale, le cui caratteristiche specifiche permettono un utilizzo aperto e differenziabile in base alle esigenze del Soggetto. Per capire meglio di cosa si tratta è necessario partire dalla ricerca che identifica i metodi controllabili, questi sono essenzialmente di tre tipi:

La fermentazione lattica ottenuta per infusione.

La fermentazione lattica ottenuta tramite la salamoia.

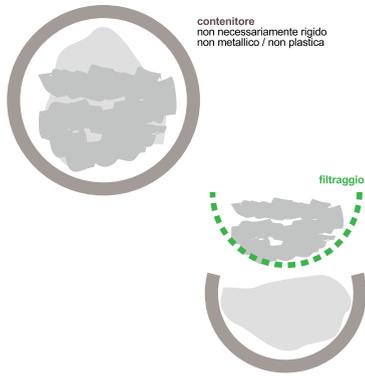
La scolatura.

Dagli schemi possiamo notare che questi procedimenti sono molto semplici e necessitano solo di alcuni accorgimenti:

- l'assenza o la presenza di ossigeno;
- la presenza di organismi starter, direttamente dipendenti dal substrato che si intende processare;
- alcune caratteristiche del contenitore;
- il controllo e il mantenimento di tre parametri fondamentali: la temperatura ambiente, la temperatura interna del prodotto fermentante e il suo livello di acidità (Ph).

Fermentazione Lattica: INFUSIONE

Latte, Bevande



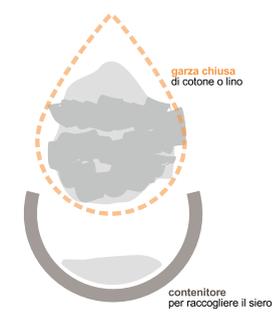
Fermentazione Lattica: SALAMOIA

Verdure, Frutta, Semi, Legumi, Pesce



SCOLATURA

Yogurt, Formaggi



TEMPISTICHE PROCESSO

24 h

UNIRE LATTE/BEVANDA ALL'ORGANISMO STARTER

48 h

MANTENERE COSTANTE LA TEMPERATURA

72 h

FILTRARE E RACCOGLIERE LO STARTER

OLTRE
3 gg

PREPARARE IL CIBO

AGGIUNGERE SALE O ACQUA + SALE

SCHIACCIARE CON FORZA STRATO DOPO STRATO RIEMPENDO IL CONTENITORE

SOVRAPPORRE UN PESO AL CONTENUTO PER EVITARE IL GALLEGGIAMENTO DEL CIBO E IMMERGERE TOTALMENTE CON CURA

OSSIGENARE SQUOTENDO QUOTIDIANAMENTE FINO A INIZIO FERMENTAZIONE

RIMUOVERE LE MUFFE FORMATE IN SUPERFICIE QUANDO NECESSARIO

ESTRARRE IL CIBO CHE SI INTENDE CONSUMARE (LA SUA CONSERVAZIONE IN LUOGO FRESCO E ASCIUTTO DURA 6 MESI)

COMPRIMERE IL CIBO RESTANTE OTTENENDO LA SUA TOTALE IMMERSIONE

TEMPI A DISCREZIONE DEL SOGGETTO E DEI SUOI CONSUMI

PREPARARE IL CIBO

RIPORRE IL CIBO ALL'INTERNO DI UNA GARZA O UN TELO GENERICO

SOVRAPPORRE IL CIBO AVVOLTO AD UN CONTENITORE PER RACCOGLIERE IL SIERO

APPENDERE E SCOLARE

FORMARE DELLE MASSE DI DIMENSIONI RIDOTTE CON L'IMPASTO OTTENUTO

RIPORRE IN UN BARATTOLO RICOPRENDO CON OLIO AROMATIZZATO

CIBO FERMENTATO

Nozioni per il progetto. Ruolo dei microrganismi starter nei prodotti fermentati.

I microrganismi cosiddetti "starter" consentono di:

- Rendere l'ambiente inospitale per i microrganismi indesiderati per abbassamento del pH (produzione di acidi organici per fermentazione), abbassamento dell'Eh, produzione di altri composti inibitori a basso peso molecolare, produzione di batteriocine.
- Produrre acidi per fermentazione dei carboidrati, con alterazione del sapore e della struttura dell'alimento.
- Produrre altri composti volatili o meno (etanolo, acetaldeide, CO₂, diacetile, etc.) per fermentazione dei carboidrati con alterazione del sapore, dell'aroma o della

struttura dell'alimento.

- Produrre sostanze sapide o aromatiche o loro precursori attraverso il metabolismo delle proteine e dei grassi.
- Influenzare il colore dell'alimento per produzione di pigmenti o favorendo la produzione di sostanze colorate stabili.
- Migliorare le caratteristiche nutrizionali di un alimento, distruggendo glucosidi tossici, oligosaccaridi, inibitori delle proteasi.
- Produrre vitamine o migliorare la digeribilità delle proteine.

Nozioni per il progetto. Fattori selettivi nelle fermentazioni alimentari.

Parametri che possono determinare la riuscita del processo fermentativo:

- bassi pH.
- alte concentrazioni di zuccheri o di sale.
- presenza di inibitori naturali.
- presenza di conservanti.
- atmosfera anaerobica.
- competizione microbica.
- temperature alte o basse.

È stato quindi scelto di intervenire sugli ultimi due punti perché traducibili dal punto di vista pratico in un dispositivo mentre i primi due vengono lasciati alla competenza del Soggetto poiché dipendenti nello specifico dal procedimento.

Per quanto riguarda i parametri, comuni ai tre processi, sono stati riconosciuti al tempo stesso come i più semplici da comprendere per il Soggetto e come i più rappresentativi per il feedback da visualizzare.

Data la loro discreta semplicità è possibile approcciarsi ad essi con strumenti altrettanto semplici, da questo nasce spontaneamente l'atteggiamento partecipativo richiesto all'utilizzatore che caratterizza poi l'intero progetto anche dal punto di vista produttivo.

È possibile infatti misurare queste temperature e il Ph con normali sensori trovabili in commercio e archiviare questi dati attraverso la scheda di controllo Arduino per poi visualizzarli temporaneamente su un mini display collegato alla scheda.

Dopo alcuni tentativi è stato chiaro che questo progetto poteva prendere vita in modo semplice ed immediato

ricorrendo a pratiche della cultura Open.

Avendo chiaro il concept su cui sviluppare il progetto è stato scelto di mettere a punto direttamente un prototipo funzionante tenendo in conto solamente questi parametri ricorrendo ad un'essenzialità quasi rozza, ma che potesse essere riproducibile attraverso dinamiche D.I.Y. in qualsiasi condizione.

Per questo motivo in un secondo momento anche il contenitore è stato accantonato volontariamente, in modo da non dare vincoli sui volumi e sulle quantità di cibo da processare. Quindi, se il contenitore è scelto dall'utilizzatore secondo le proprie esigenze, il dispositivo è munito di un componente adattabile ad esso che consente di mantenere in substrato fermentante sotto il livello del liquido in questione, funzione necessaria per la fermentazione lattica tramite salamoia, in quanto anaerobica; questo componente è la spirale che, grazie alla sua forma e ad un materiale flessibile, può aderire alle pareti interne di contenitori di dimensioni diverse impedendo il galleggiamento.

6.3 Il prototipo.

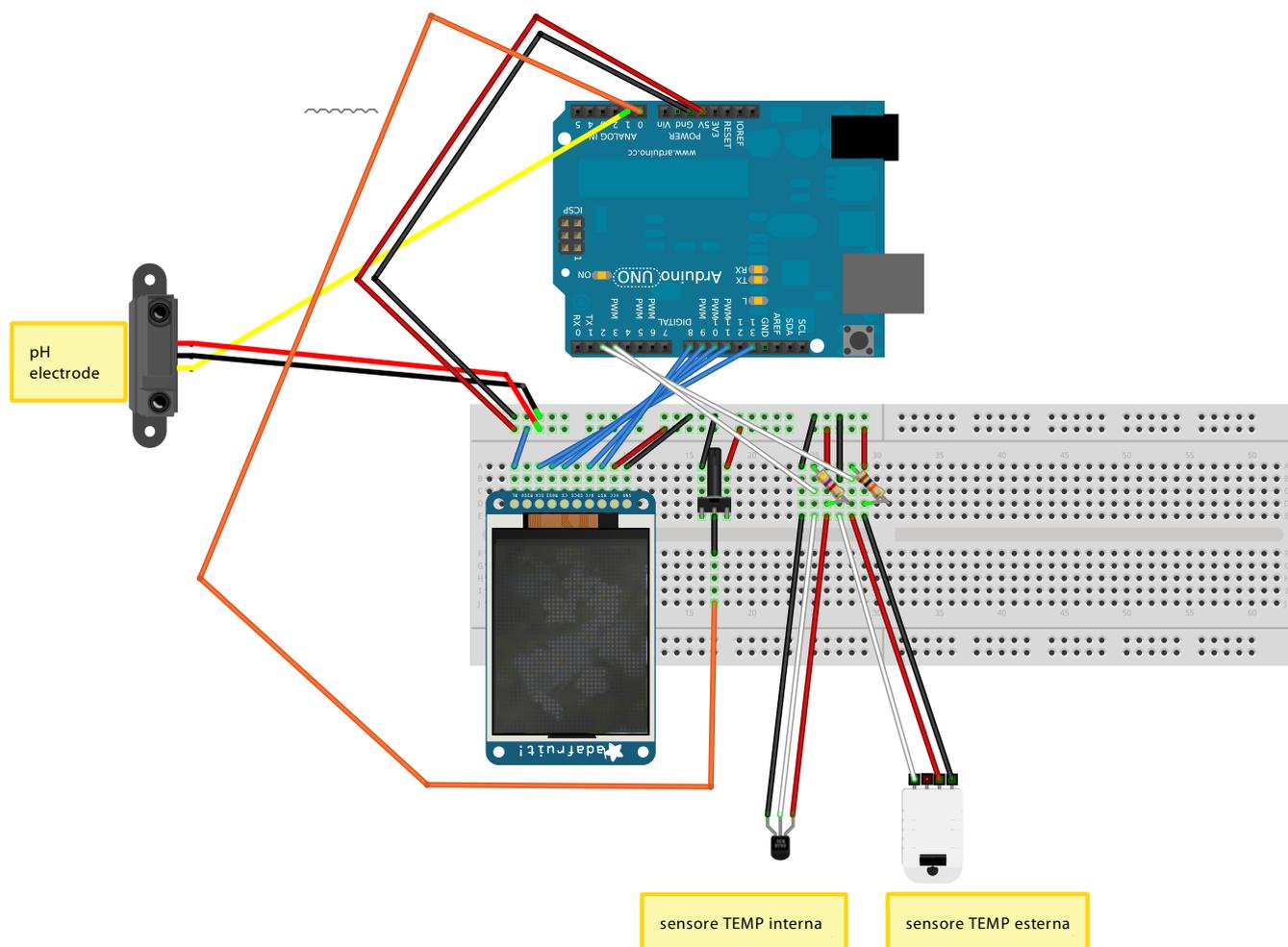
La costruzione del prototipo impegna me e il Dottor Alessandro Balbo, Assegnista di ricerca presso il Dipartimento di Architettura e Design (DAD) del Politecnico di Torino. Se nell'ideazione del concept la progettazione è stata condivisa in questo frangente il nostro contributo è stato complementare: da una parte è stata sviluppata la parte tecnica riguardante la scheda di controllo Arduino, la sua programmazione e il funzionamento dei sensori con la relativa visualizzazione dei dati sul display, mentre dall'altra è stata prodotta la "struttura" del dispositivo ricorrendo appunto a tecniche semplici, riproponibili facilmente e praticabili con strumenti alla portata di tutti: sono stati utilizzati unicamente sega, lima e trapano. Il box che contiene la scheda di controllo è stato ricavato da una scatola per impianti elettrici, adattata

con gli accessi necessari per il cablaggio dei sensori e una parete trasparente attraverso la quale è possibile osservare il display e accedere all'unico tasto di regolazione per l'azzeramento della raccolta dati e il refresh del relativo grafico.

Da questo nucleo centrale, cuore del dispositivo, parte un tubo che andrà immerso nella soluzione e svolge la funzione di sede per i cablaggi dei sensori e eventuale mestolo per il mescolamento del substrato fermentante. La connessione tra il box e l'asta-mestolo è reversibile ad incastro e presenta un adattatore in gomma per garantire la tenuta stagna dove necessaria.

Come accennato precedentemente il contenitore è generico e scelto dall'utilizzatore, al suo interno troverà sede il componente anti-galleggiamento a forma di spirale.

Figura 76.
Schema di simulazione per la programmazione di Arduino



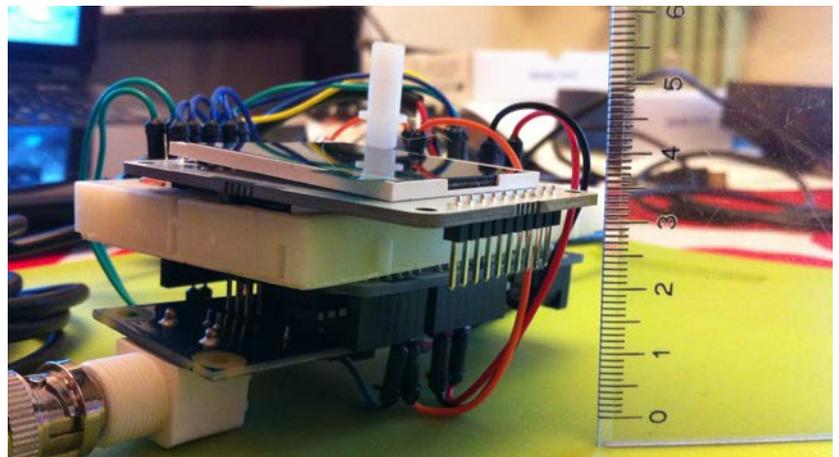
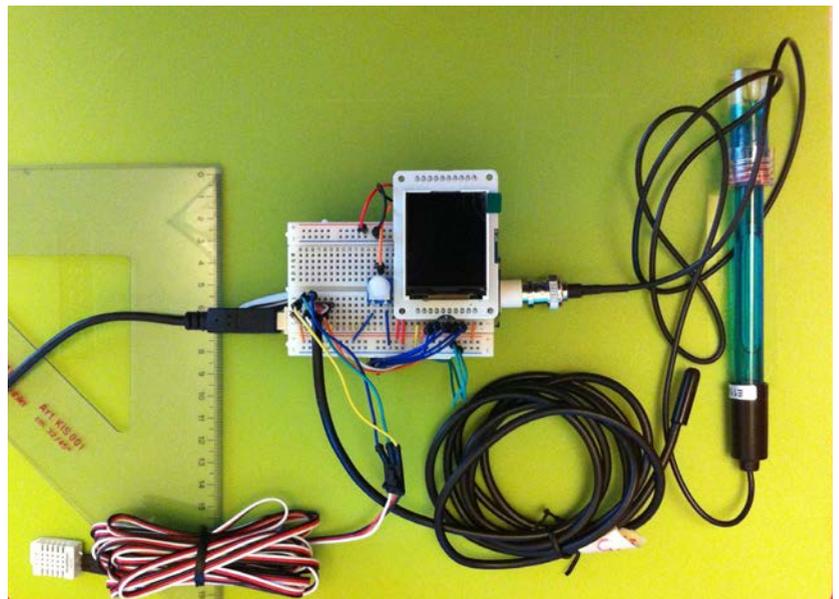
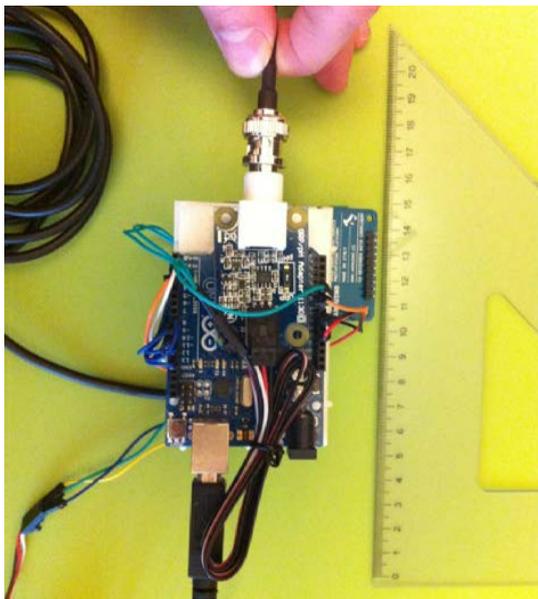


Figura 77.
Messa a punto della scheda
di controllo Arduino



6.4 Test prototipo. Kefir.

Il prototipo è stato testato con la preparazione dello yogurt di kefir a base di latte. Il monitoraggio dei parametri è avvenuto nelle 24 ore successive alla preparazione, i risultati ottenuti sono stati costanti durante il processo e dopo solo un giorno di fermentazione il kefir era pronto per essere gustato. Fasi della preparazione:

1. I grani di Kefir sono riposti in un contenitore domestico generico in vetro pulito e perfettamente asciutto.

Per ogni cucchiaino di granuli di kefir sono stati aggiunti circa 250 ml di latte (come da dose consigliata, questa può variare a seconda dei gusti personali del Soggetto)

2. Il barattolo non è stato coperto volutamente perché il processo è aerobico. Una volta unito starter e substrato fermentante il contenitore è stato riposto in un luogo semi buio, senza luce diretta.

3. Dopo 24 ore il kefir è pronto; questo lasso di tempo può arrivare fino alle 78 ore se si desidera un prodotto più acido.

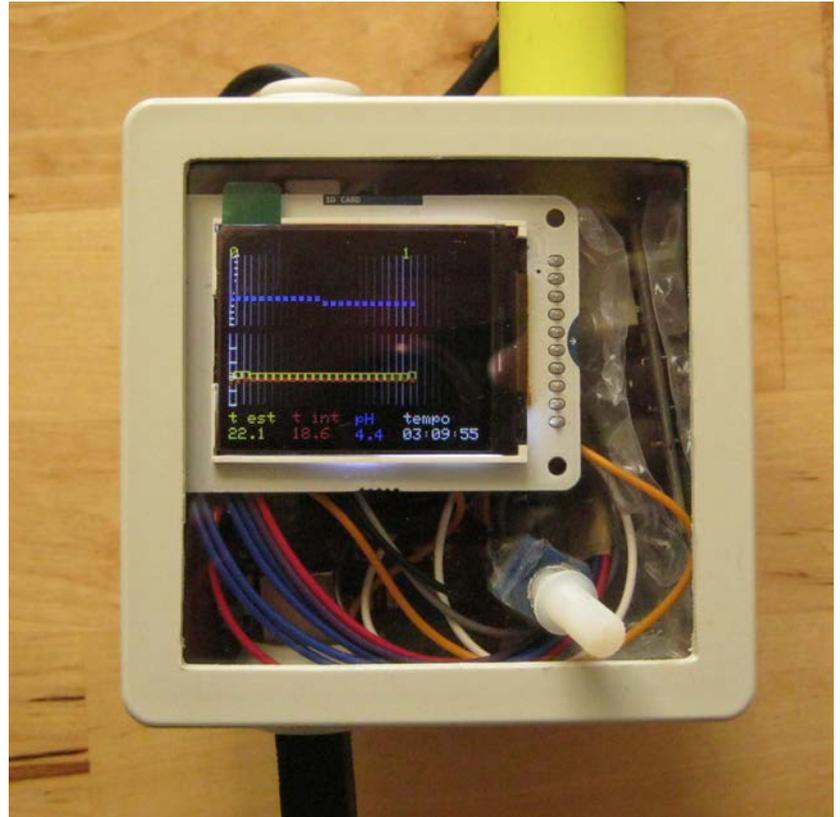
4. Il kefir è stato raccolto. In base alla tecnica utilizzata durante la raccolta si può ottenere anche in questo caso un risultato più o meno acido decidendo se mischiare oppure no il siero con il latte.

I granuli rimasti possono essere riutilizzati con la stessa procedura.

5. Il Kefir filtrato è pronto per essere consumato. Può essere mangiato puro oppure con l'aggiunta di miele, frutta fresca, frutta sciroppata, cereali, caffè, cioccolato liquido o in scaglie oppure in qualsiasi altro modo si preferisca. E' possibile tenerlo in un ambiente a bassa temperatura controllata per circa una settimana, oltre a questo tempo la sua acidità potrebbe aumentare troppo in quanto il suo processo di fermentazione continua lentamente.



Figura 78.
Procedura di preparazione del Kefir con l'utilizzo del dispositivo



			
misurazione (ogni ora)	temperatura esterna	temperatura interna	livello di acidità
1	21	16	5
2	22	17	5
3	22	18	5
4	21	18	5
5	21	18	5
6	21	18	5
7	21	18	5
8	21	18	5
9	21	18	5
10	21	18	5
11	21	18	5
12	21	18	4
13	21	18	4
14	21	18	4
15	21	18	4
16	21	18	4
17	21	18	4
18	21	18	4
19	21	18	4
20	21	18	4
21	21	18	4
22	21	18	4
23	21	18	4
24	22	18	4

Figura 79.
Il dispositivo in funzione

Figura 80.
*Il grafico rappresentante
l'andamento dei parametri*

Figura 81.
*Sintesi dei risultati ottenuti.
Elenco misurazioni ogni ora*

Fermentazione
Lattica: INFUSIONE



Legumi



Yogurt



Birra



Frutta



Aceto

Fermentazione
Lattica: SALAMOIA



Olive



Verdura



Pesce



Conserva



Verdura cotta

SCOLATURA



Tofu



Formaggio fresco



Caglio



Formaggio di yogurt



Ricotta

Figura 82.
Alcuni cibi diffusi la cui
preparazione può essere
monitorata con il dispositivo

Nozioni per il progetto. Il codice per la programmazione della scheda di controllo Arduino.

```
// Example testing sketch for various DHT humidity/temperature sensors
// DHT
// Connect pin 1 (on the left) of the sensor to +5V
// Connect pin 2 of the sensor to whatever your DHTPIN is
// Connect pin 4 (on the right) of the sensor to GROUND
// Connect a 10K resistor from pin 2 (data) to pin 1 (power) of the sensor

// waterproof DS18B20
// Connect the red to +5V
// Connect the black to GND
// Connect the white to the digital pin. Then, put a 4.7kohm resistor between the white wire
and the +5V.

// Analog pH 5V
// Connect the pin analog 1

//libreria per scrivere sulla memoria
#include <EEPROM.h>

//libreria e roba per schermo lcd
#include <SPI.h>
#include <TFT.h>
#define cs 10
#define dc 9
#define rst 8
TFT TFTscreen = TFT(cs, dc, rst);

//libreria e roba per il sensore DHT
#include "DHT.h"
#define DHTPIN 2 // what pin we're connected to
#define DHTTYPE DHT22 // DHT 22 (AM2302)
DHT dht(DHTPIN, DHTTYPE);
float h1;
float t1;
int xPosT1 = 0;
int yPosT1 = 0;

//libreria e roba per il sensore DS18B20
#include <OneWire.h>
OneWire ds(3); // on pin 3 (a 4.7K resistor is necessary)
float t2;
int xPosT2 = 0;
int yPosT2 = 0;

//libreria e roba per elettrodo pH
#include <Wire.h>
#include <DallasTemperature.h>
#define ONE_WIRE_BUS 3
OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);
int sensorPinPH = A1;
float sensorPH = 0;
int xPosPH = 0;
int yPosPH = 0;

// variabili for the potentiometer
int sensorPotenzPin = A0;
int sensorPotenz = 0;
int sensorPotenzPre = sensorPotenz;

float tacca;

// coordinate Y per l'area del grafico PH
int altezzaGraphP = 10;
int baseGraphP = 55;

// coordinate Y per l'area del grafico TEMPERATURA
int altezzaGraphT = 60;
int baseGraphT = 105;

//primo indirizzo della memoria (quanti ce ne sono?) forse 1024!
int addr = 0;

long previousMillisEEPROM = 0; // will store last time LED was updated
// the follow variables is a long because the time, measured in milliseconds,
// will quickly become a bigger number than can be stored in an int.
// 1000 = 1 sec
// 60000 = 1 min
// 3600000 = 1 ora
long intervalEEPROM = 3600000;

long days=0;
long hours=0;
long mins=0;
long secs=0;

// variabile per delay
int pausa = 2000;

void setup() {
  Serial.begin(9600);
  Serial.println("DHTxx test!");

  TFTscreen.begin();
  TFTscreen.background(0, 0, 0);
  TFTscreen.fill(50, 50, 50);

  //area grafico TEMPERATURA
  TFTscreen.rect(0, altezzaGraphT, TFTscreen.width(), baseGraphT - altezzaGraphT);
  //area grafico PH
  TFTscreen.rect(0, altezzaGraphP, TFTscreen.width(), baseGraphP - altezzaGraphP);

  TFTscreen.setTextSize(1);

  dht.begin();

  //inizializza per pH
  sensors.begin();
}

void loop() {

  //il delay se non è mancata la corrente

  int pausaScreen = 2000;
  int pausaSensori = 2000;
  int pausaGrafico = 2000;
  int pausaEEPROM = 2000;

  unsigned long currentMillis = millis();

  //----- lettura dht (temp esterna) -----
  //Reading temperature or humidity takes about 250 milliseconds!
  //Sensor readings may also be up to 2 seconds 'old' (its a very slow sensor)
  h1 = dht.readHumidity();
  t1 = dht.readTemperature();

  //----- lettura pH-----

  // occhio, ha bisogno della temperatura per calcolare il pH, quella che legge qui è la temp
  ESTERNA!
  sensors.requestTemperatures();
  sensorPH = 7.0 - (2.5 - analogRead(sensorPinPH) / 195.0) / (0.257179 + 0.000941468
  * sensors.getTempCByIndex(0));
  //Serial.println("pH: ");
  //Serial.println(sensorPH);
  //Serial.println(" ");

  //----- lettura ds waterproof-----
  byte i;
  byte present = 0;
  byte type_s;
  byte data[12];
  byte address[8];

  if (!ds.search(address)) {
    // Serial.println("No more addresses.");
    // Serial.println();
    ds.reset_search();
    delay(250);
    return;
  }

  // Serial.println("ROM =");
  // for (i = 0; i < 8; i++) {
  //   Serial.write(" ");
  //   Serial.print(address[i], HEX);
  // }

  if (OneWire::crc8(address, 7) != address[7]) {
    Serial.println("CRC is not valid!");
    return;
  }

  // the first ROM byte indicates which chip
  switch (address[0]) {
    case 0x10:
      //Serial.println(" Chip = DS18S20"); // or old DS1820
      type_s = 1;
      break;
    case 0x28:
      //Serial.println(" Chip = DS18B20");
      type_s = 0;
      break;
    case 0x22:
      //Serial.println(" Chip = DS1822");
      type_s = 0;
      break;
    default:
      Serial.println("Device is not a DS18x20 family device.");
      return;
  }

  ds.reset();
  ds.select(address);
  ds.write(0x44, 1); // start conversion, with parasitic power on at the end

  delay(1000); // maybe 750ms is enough, maybe not
  // we might do a ds.depower() here, but the reset will take care of it.

  present = ds.reset();
  ds.select(address);
  ds.write(0xBE); // Read Scratchpad
```

```

// Serial.print(" Data = ");
// Serial.print(present, HEX);
// Serial.print(" ");
for (i = 0; i < 9; i++) { // we need 9 bytes
  data[i] = ds.read();
  // Serial.print(data[i], HEX);
  // Serial.print(" ");
}
// Serial.print(" CRC=");
// Serial.print(OneWire::crc8(data, 8), HEX);
// Serial.println();

// Convert the data to actual temperature
// because the result is a 16 bit signed integer, it should
// be stored to an "int16_t" type, which is always 16 bits
// even when compiled on a 32 bit processor.
int16_t raw = (data[1] << 8) | data[0];
if (type_s) {
  raw = raw << 3; // 9 bit resolution default
  if (data[7] == 0x10) {
    // "count remain" gives full 12 bit resolution
    raw = (raw & 0xffff) + 12 - data[6];
  }
} else {
  byte cfg = (data[4] & 0x60);
  // at lower res, the low bits are undefined, so let's zero them
  if (cfg == 0x00) raw = raw & ~7; // 9 bit resolution, 93.75 ms
  else if (cfg == 0x20) raw = raw & ~3; // 10 bit res, 187.5 ms
  else if (cfg == 0x40) raw = raw & ~1; // 11 bit res, 375 ms
  // default is 12 bit resolution, 750 ms conversion time
}
t2 = (float)raw / 16.0;
//----- FINE lettura ds3-----

//t2 = 40;

// check if returns are valid, if they are NaN (not a number) then something went wrong!
if (!isnan(t1) || !isnan(h1)) {
  Serial.println("Failed to read from DHT");
} else {
  Serial.print("Humidity: ");
  Serial.println(t1);
  Serial.print(" %\n");
  Serial.print("Temperature: ");
  Serial.println(t1);
  Serial.println(h1 + "C");
}

sensorPotenz = analogRead(sensorPotenzPin);
Serial.println(sensorPotenz);

// TFTscreen.width() = 160 px
int sensorPotenzMap = map(sensorPotenz, 0, 1024, 0, TFTscreen.width());
// tacca = TFTscreen.width() / sensorPotenzMap;
// numero delle tacche
tacca = sensorPotenzMap;
Serial.println(TFTscreen.width());
Serial.println(tacca);

// metti 3600000 per l'intervallo di un'ora
// metti 60000 per l'intervallo di un minuto
// metti 1000 per l'intervallo di un secondo

// intervallo tra ogni tacca
intervalEEPROM = 3600000;
Serial.println(intervalEEPROM);
// intervallo tra ogni rilevazione (5 px del cursore) OCCHIO
intervalEEPROM /= (TFTscreen.width() / tacca) / 5;
Serial.println(intervalEEPROM);

if (abs(sensorPotenz - sensorPotenzPre) > 10) {
  // schermo nero
  TFTscreen.fill(0,0,0);
  TFTscreen.noStroke();
  TFTscreen.rect(0,0,TFTscreen.width(),TFTscreen.height());

  int sensorPotenzMap = map(sensorPotenz, 0, 1024, 0, TFTscreen.width());
  tacca = sensorPotenzMap;

  for (int i = 0; i < tacca; i++) {
    float posX = i * (TFTscreen.width() / tacca);

    // sfumatura delle tacche
    float sfuma = 255 - (255 * abs(sin(radians(*7.5)))));

    // tacche Temp
    TFTscreen.stroke(40 + sfuma/2, 40 + sfuma/2, sfuma/1.5);
    TFTscreen.line(posX, baseGraph1, posX, altezzaGraph1);

    // tacche pH
    TFTscreen.stroke(40 + sfuma/2, 40 + sfuma/2, sfuma/1.5);
    TFTscreen.line(posX, baseGraphP, posX, altezzaGraphP);

    char timePrintout[8];
    dtostrf(i, 1, 0, timePrintout);
    TFTscreen.stroke(255,255,0);
    TFTscreen.text(timePrintout, posX * 24, 5);
  }

  sensorPotenzPre = sensorPotenz;
  //Serial.println(sensorPotenzMap);
}

TFTscreen.noFill();
// tacche orizzontali Temp
TFTscreen.stroke(255, 255, 255);
for (int i = 0; i < (baseGraphT - altezzaGraphT); i += (baseGraphT - altezzaGraphT)/5) {
  TFTscreen.line(0, baseGraphT - i, 5, baseGraphT - i);
}
// tacche orizzontali pH
TFTscreen.stroke(255, 255, 255);
for (int i = 0; i < 12; i++) {
  float y = (baseGraphP - altezzaGraphP)/11;
  TFTscreen.line(3, altezzaGraphP + y*i, 3 + cos(radians((180 / 12)*i))*3, altezzaGraphP + y*i);
}

// mostra sul grafico -----
if (currentMillis - previousMillisEEPROM > intervalEEPROM) {
  previousMillisEEPROM = currentMillis;

  // scrivi Temp 1 (esterna)
  yPosT1 = map(t1, 0, 50, 0, baseGraphT - altezzaGraphT);
  TFTscreen.stroke(255,255,0);
  TFTscreen.rect(xPosT1 - 2, baseGraphT - yPosT1 - 2, 5, 5);
  xPosT1 += 5;

  // scrivi Temp 2 (interna)
  yPosT2 = map(t2, 0, 50, 0, baseGraphT - altezzaGraphT);
  TFTscreen.fill(255,0,0);
  TFTscreen.stroke(255,0,0);
  TFTscreen.rect(xPosT2 - 1, baseGraphT - yPosT2 - 1, 3, 3);
  xPosT2 += 5;

  // scrivi pH
  yPosPH = map(sensorPH, 0, 12, 0, baseGraphP - altezzaGraphP);
  TFTscreen.fill(0,75,255);
  TFTscreen.stroke(0,75,255);
  TFTscreen.rect(xPosPH - 1, baseGraphP - yPosPH - 1, 3, 3);
  xPosPH += 5;

  // registra -----
  EEPROM.write(addr, t1);
  addr = addr + 1;
  EEPROM.write(addr, t2);
  addr = addr + 1;
  EEPROM.write(addr, sensorPH);
  addr = addr + 1;
}

char dhtPrintout[8];
dtostrf(t1, 1, 1, dhtPrintout);
//rettangolo nero per l'aggiornamento
TFTscreen.fill(0,0,0);
TFTscreen.noStroke();
TFTscreen.rect(0,120,TFTscreen.width(),10);
TFTscreen.stroke(255,255,0);
TFTscreen.text("t est", 0, 110);
TFTscreen.text(dhtPrintout, 0, 120);

char dsPrintout[8];
dtostrf(t2, 1, 1, dsPrintout);
//rettangolo nero per l'aggiornamento
TFTscreen.fill(0,0,0);
TFTscreen.noStroke();
TFTscreen.rect(40,120,TFTscreen.width(),10);
TFTscreen.stroke(255,0,0);
TFTscreen.text("t int", 40, 110);
TFTscreen.text(dsPrintout, 40, 120);

char PHPrintout[8];
dtostrf(sensorPH, 1, 1, PHPrintout);
//rettangolo nero per l'aggiornamento
TFTscreen.fill(0,0,0);
TFTscreen.noStroke();
TFTscreen.rect(80,120,TFTscreen.width(),10);
TFTscreen.stroke(0,75,255);
TFTscreen.text("pH", 80, 110);
TFTscreen.text(PHPrintout, 80, 120);

secs = currentMillis/1000; //convert milliseconds to seconds
mins=secs/60; //convert seconds to minutes
hours=mins/60; //convert minutes to hours
days=hours/24; //convert hours to days
secs=secs-(mins*60); //subtract the covered seconds to minutes in order to display 59 secs
max
mins=mins-(hours*60); //subtract the covered minutes to hours in order to display 59 minutes
max
hours=hours-(days*24); //subtract the covered hours to days in order to display 23 hours max

char clockPrintout[4];
// comando c++(?) per trasformare numeri in characters
sprintf(clockPrintout, "%02d:%02d:%02d", int(hours), int(mins), int(secs));
//rettangolo nero per l'aggiornamento
TFTscreen.fill(0,0,0);
TFTscreen.noStroke();
TFTscreen.rect(110,120,TFTscreen.width(),10);
TFTscreen.stroke(255,255,255);
TFTscreen.text("tempo", 110, 110);
TFTscreen.text(clockPrintout, 110, 120);

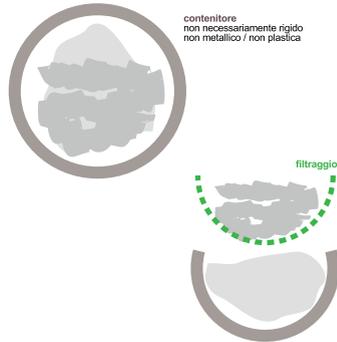
//delay(50);
delay(pausa);
}

```

Analisi processo

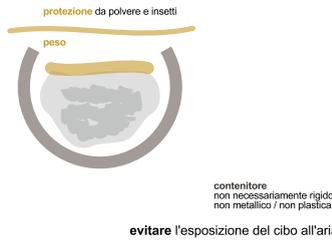
Fermentazione Lattica: INFUSIONE

Latte, Bevande



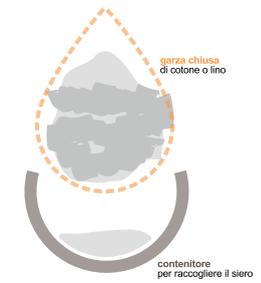
Fermentazione Lattica: SALAMOIA

Verdure, Frutta, Semi, Legumi, Pesce



SCOLATURA

Yogurt, Formaggi



TEMPISTICHE PROCESSO

24 ORE

UNIRE LATTE/BEVANDA ALL'ORGANISMO STARTER

48 ORE

MANTENERE COSTANTE LA TEMPERATURA

72 ORE

FILTRARE E RACCOLGERE LO STARTER

OLTRE 3 GIORNI

PREPARARE IL CIBO

AGGIUNGERE SALE O ACQUA + SALE

SCHIACCIARE CON FORZA STRATO DOPO STRATO RIEMPIENDO IL CONTENITORE

SOVRAPPORRE UN PESO AL CONTENUTO PER EVITARE IL GALLEGGIAMENTO DEL CIBO E IMMERGERE TOTALMENTE CON CURA

OSSIGENARE SQUOTENDO QUOTIDIANAMENTE FINO A INIZIO FERMENTAZIONE

RIMUOVERE LE MUFFE FORMATE IN SUPERFICIE QUANDO NECESSARIO

TEMPI DI DISCREZIONE DEL SOGGETTO E DEI SUOI CONSUMI

ESTRARRE IL CIBO CHE SI INTENDE CONSUMARE (LA SUA CONSERVAZIONE IN LUOGO FRESCO E ASCIUTTO DURA 6 MESI)

COMPRIERE IL CIBO RESTANTE OTTENENDO LA SUA TOTALE IMMERSIONE

PREPARARE IL CIBO

RIPORRE IL CIBO ALL'INTERNO DI UNA GARZA O UN TELO GENERICO

SOVRAPPORRE IL CIBO AVVOLTO AD UN CONTENITORE PER RACCOLGERE IL SIERO

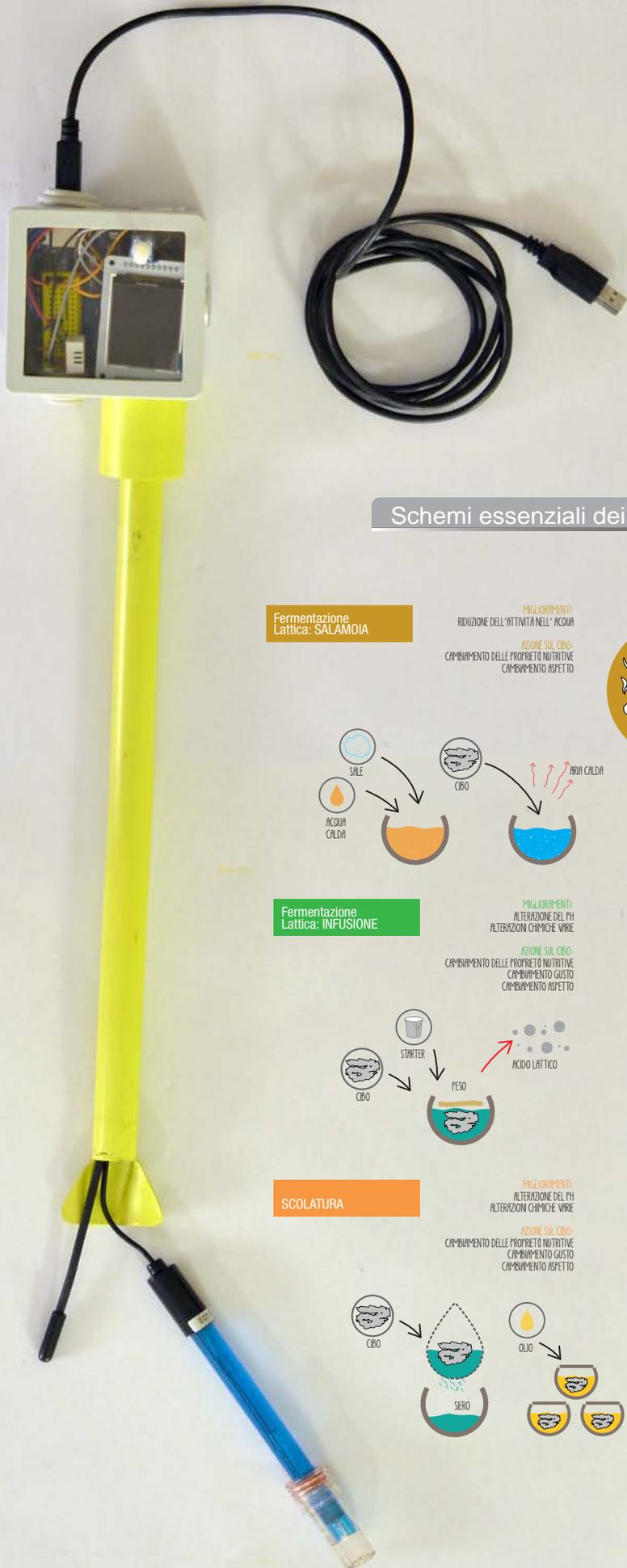
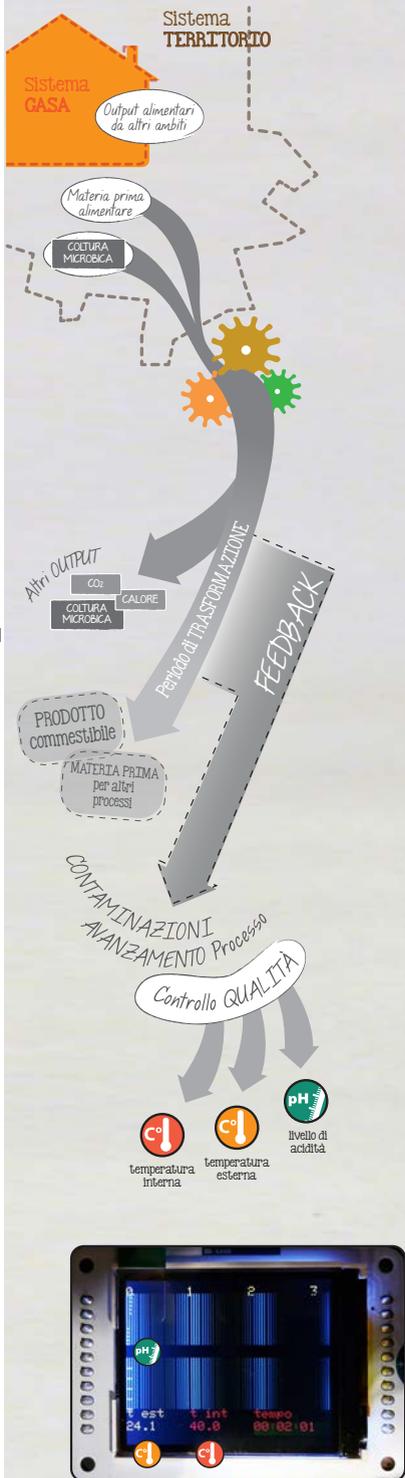
APPENDERE E SCOLARE

FORMARE DELLE MASSE DI DIMENSIONI RIDOTTE CON L'IMPASTO OTTENUTO

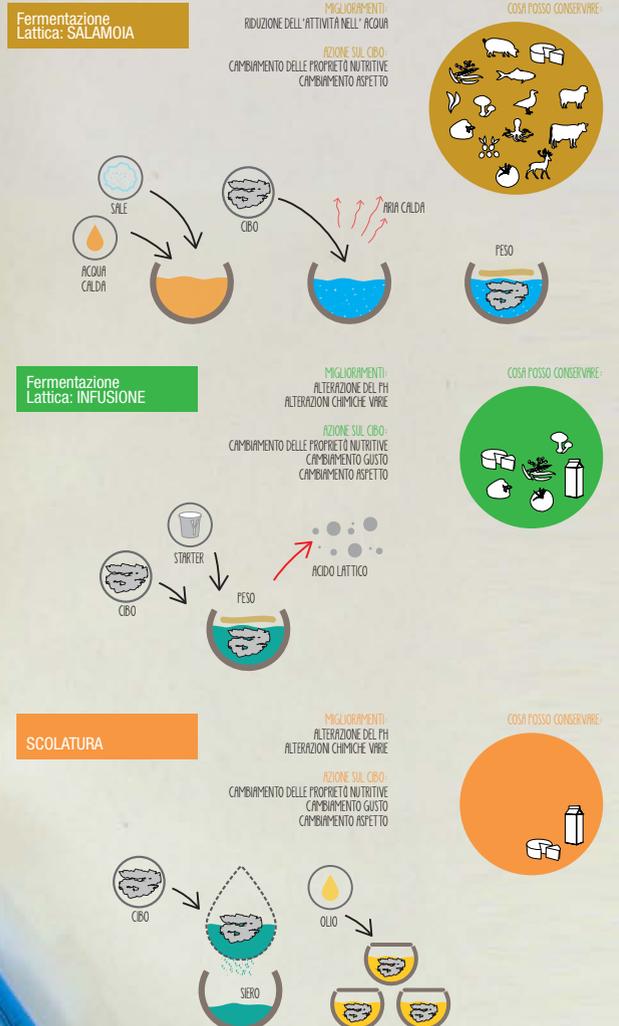
RIPORRE IN UN BARATTOLO RICOPRENDO CON OLIO AROMATIZZATO

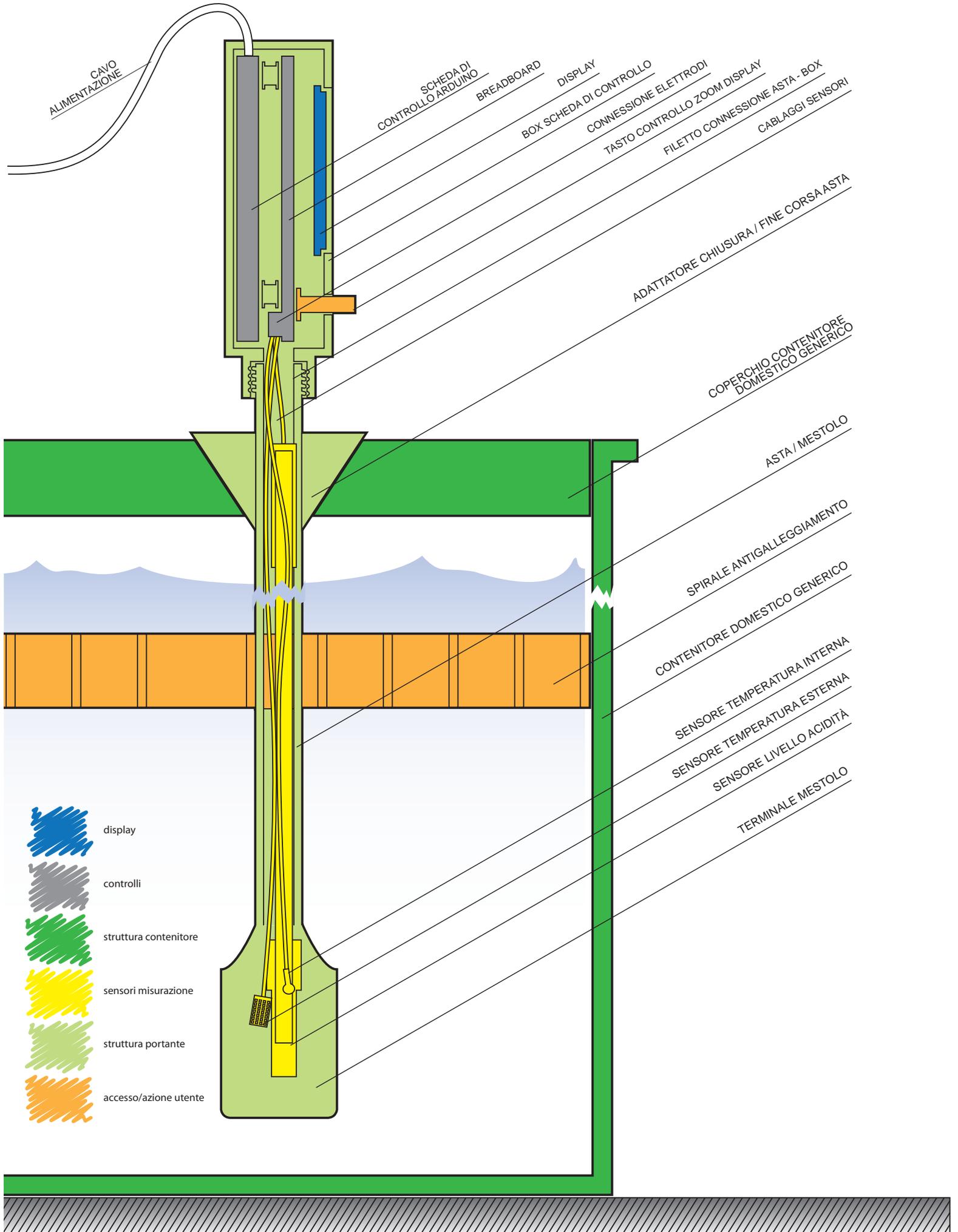
CIBO FERMENTATO

Monitoraggio trasformazione batterica



Schemi essenziali dei processi





CONNESSIONE
USB ALIMENTAZIONE

CAVO
ALIMENTAZIONE

CABLAGGIO
SENSORE
TEMPERATURA
INTERNA

SENSORE
TEMPERATURA
INTERNA

CABLAGGIO
SENSORE
ACIDITA

SENSORE
ACIDITA

CONNESSIONE
ELETTRODI

ASTA
MESTOLO

FILETTO
CONNESSIONE
ASTA - BOX

SPIRALE
ANTIGALLEGGIAMENTO

ADATTATORE CHIUSURA
FINE CORSA ASTA

TERMINALE
MESTOLO

DISPLAY

SCHEDA DI
CONTROLLO
ARDUINO

BREADBOARD

CONNESSIONE
ALIMENTAZIONE

TASTO
CONTROLLO

SENSORE
TEMPERATURA
ESTERNA

LUNOTTO

CHIUSURA
BOX

BOX

GUARNIZIONE
CONNESSIONE
ASTA

GUARNIZIONE
CABLAGGIO
ALIMENTAZIONE