

Mario Ursino (marioursino)

DIY REFLOW OVEN

10 December 2018

Premessa

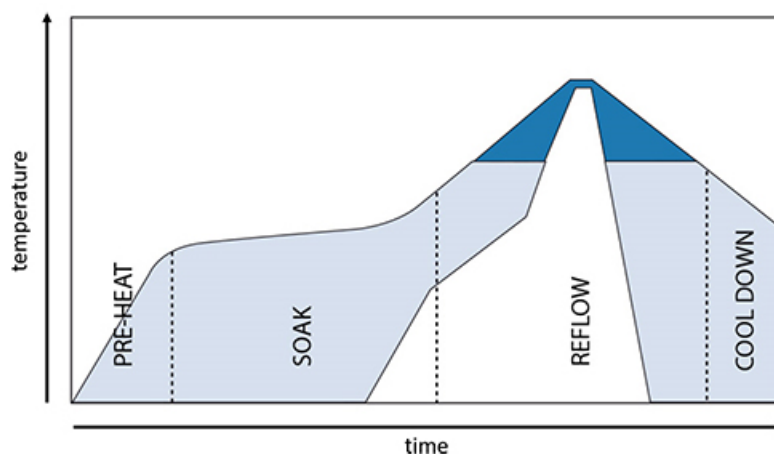
In questo articolo parlerò della trasformazione di un forno da 35 € in un forno per saldatura *reflow*, ossia quel tipo di saldatura che tramite uno *stencil* e della pasta saldante permette di ottenere PCB assemblati di qualità.

Tecnica di saldatura

Citando [Wikipedia](#):

"La saldatura per *rifusione* è un processo in cui viene utilizzata una pasta saldante (una miscela di stagno in polvere e *flussante*) per fissare temporaneamente uno o più componenti elettrici ai loro punti di contatto, dopo di che l'intero assemblaggio viene sottoposto a calore controllato, che fonde la lega e salda permanentemente i componenti. La fusione può essere completata in appositi forni, utilizzando lampade ad infrarossi oppure con saldatori ad aria calda, lavorando singolarmente ogni componente."

Sulle piazzole libere del PCB viene depositata la pasta saldate tramite uno stencil. Affinché la pasta saldi correttamente i componenti, è necessario sottoporla ad un trattamento termico con un determinato profilo temporale, suddiviso in più intervalli, come mostrato di seguito:



Profilo termico suddiviso in intervalli

Preheat

I solventi nella pasta saldante evaporano.

Soak

I solventi completano l'evaporazione, il fluxante si attiva e deossida piazzole e pin dei componenti; la board viene lentamente riscaldata affinché il gradiente termico tra le sue parti sia minimizzato.

Reflow

La parte metallica della lega raggiunge la fusione, connettendo meccanicamente ed elettricamente i componenti alle piazzole della board.

Cool down

Il raffreddamento solidifica la lega metallica.

Personalmente ho notato che, ammesso che la scheda non sia estremamente semplice, il tempo necessario per la preparazione alla saldatura (stesura della pasta tramite *stencil*) è ampiamente ripagato dalla velocità e dalla qualità del processo *reflow*; questa tecnica inoltre permette di saldare componenti [no-leads](#) con precisione e affidabilità, cosa che col saldatore non è per nulla scontata.

La scheda finita, inoltre, non deve essere pulita dal fluxante e non presenta *sbavature* di alcun tipo. Inoltre, se il volume del forno lo permette, si possono saldare interi pannelli in un unico passaggio.

La costruzione del forno è stata per me motivo di curiosità più che di necessità, dato che sono necessarie nozioni di controllo digitale, termodinamica, programmazione firmware e software.

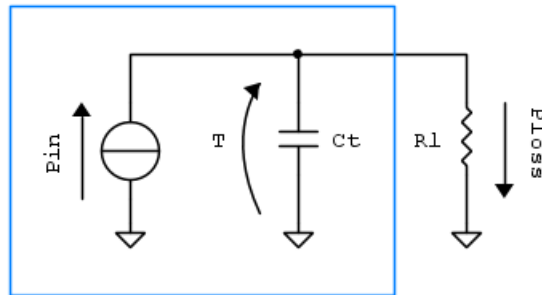
Forno elettrico e isolamento

Affinché la scheda PCB possa seguire correttamente il profilo termico, il forno deve disporre di potenza sufficiente a riscaldarla (per convezione e irraggiamento) alla velocità richiesta. È chiaro inoltre che, per seguire il profilo, è necessario un sensore di temperatura interna e un controllo automatico che agisca sulla potenza immessa nella camera, modulando la potenza erogata dalle resistenze riscaldanti.

Ho scelto quindi un fornello per riscaldare le pizzette, l'ho comprato in Austria andando a trovare un mio amico che lavora lì. Si tratta di un fornello molto economico, per la precisione si chiama "MPO 3520 Multi pizza oven", marca *Clatronic*, con due resistenze al quarzo che ho pagato circa 35 €. La potenza dichiarata è di 1000 W, ed è la maggiore che abbia trovato per un fornello così piccolo.

La potenza del forno determina la massima pendenza del profilo termico: in particolare, approssimando il sistema come un generatore di potenza termica (resistenze) chiuso in volume

(interno del forno) caratterizzato da una capacità termica fissa C_t e da una resistenza termica fissa R_l (pareti del forno verso l'ambiente), otteniamo il circuito equivalente seguente:



Intuitivamente, poca potenza e/o uno scarso isolamento (R_L piccola) degradano sia la massima pendenza $\frac{\delta T}{\delta t}$ che la massima temperatura raggiungibile. In particolare, quest'ultima si ottiene quando il transitorio termico è finito e la potenza immessa dalle resistenze corrisponde a quella dissipata verso l'ambiente. La quantificazione delle grandezze del circuito equivalente verrà trattata alla fine dell'articolo, mentre per ora mi limiterò a dire che:

1. Maggiore potenza è meglio
2. Maggiore isolamento è meglio

Questo è intuitivamente sensato in quanto sarà il controllo a regolare la potenza immessa nel volume: una maggiore potenza disponibile darà un ampio margine al controllo, mantenendo il sistema regolato lontano dalla saturazione. Una potenza delle resistenze insufficiente, d'altro canto, non sarà compensabile da nessun sistema di controllo.

Il forno appena acquistato si presentava così (senza pizza):

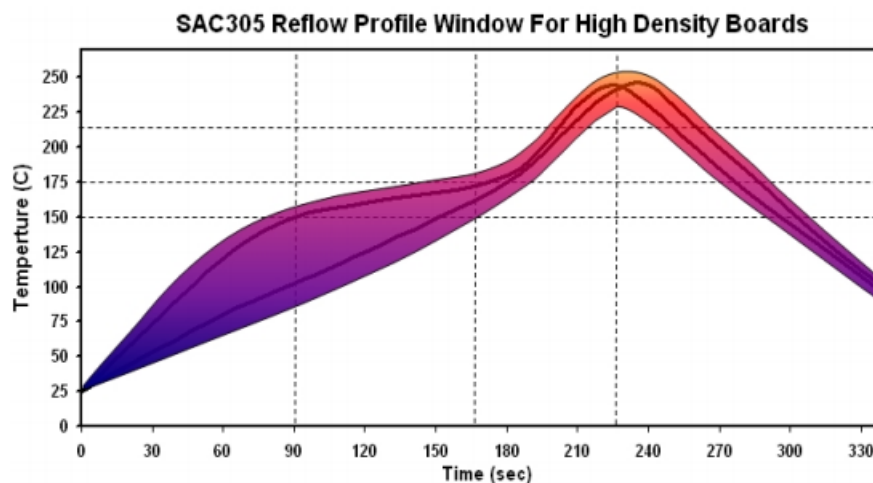


Forno "di fabbrica"

Purtroppo non ho fatto nessuna foto quando l'ho comprato.

Il fornello *stock* utilizza un sensore di temperatura a lamella come termostato: si può selezionare quali resistenze usare (sopra/sotto/entrambe) e il tempo di cottura tramite le manopole. Le resistenze vengono comandate con isteresi dal termostato, che a regime mantiene la temperatura nell'intorno della sua soglia "di scatto". A noi questo non interessa, poiché l'unica cosa che rimarrà di questo sistema sono le resistenze.

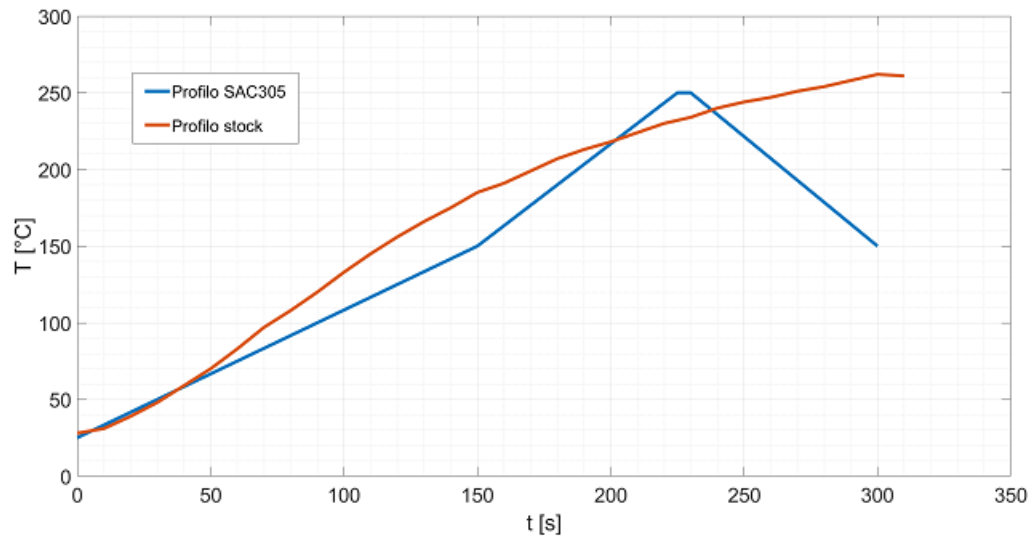
Per potere quantificare la bontà dell'isolamento del forno è interessante misurare il profilo termico che sarebbe ottenibile con il forno "di fabbrica". Come vedremo, l'isolamento di base può essere ampiamente migliorato. Per contestualizzare la misura, scegliamo un profilo *lento* per una lega ad alta temperatura senza piombo (Sn96.5 Ag3 Cu0.5, codice SAC305). È comunque possibile selezionare leghe a bassa temperatura di fusione, ma rimane una soluzione debole che limita l'utilizzo del forno.



Profilo termico consigliato per lega SAC305

Osservando l'immagine, possiamo decidere di essere "contenti" riuscendo a replicare le rampe di riscaldamento fino al raggiungimento della temperatura di picco. Una volta raggiunto questo obiettivo, i profili per leghe a minore temperatura di fusione saranno certamente raggiungibili.

Per valutare la performance del forno *stock* ho misurato la risposta del sistema alimentato alla massima potenza (entrambe le resistenze collegate alla tensione di rete). In pratica, il sistema da controllare ha come ingresso la potenza erogata dalle resistenze e come uscita la temperatura interna. Di seguito la risposta del forno di fabbrica è confrontata con il profilo che vogliamo raggiungere.



Confronto tra i profili termici stock e SAC305

L'isolamento presente è già buono: la velocità di riscaldamento iniziale è sufficiente, mentre la temperatura di picco sarebbe raggiungibile con un po' di pazienza. Questa curva è piuttosto confortante, un controllo automatico potrebbe già permettere il corretto funzionamento del forno, tuttavia il rendimento del forno è molto scarso ed il margine è piuttosto stretto, soprattutto considerando che la scheda, per quanto piccola, aggiungerà un contributo alla capacità termica del sistema peggiorandone la performance.

Affinché la potenza immessa nella camera di riscaldamento rimanga confinata e riscaldi la scheda PCB è necessario migliorare l'isolamento già presente. L'interno del forno è dotato di una fessura per il ricircolo dell'aria posta sul fondo, inoltre le pareti sono scarsamente riflettenti. Ho scelto di isolare la camera del forno con del nastro di alluminio: si tratta in pratica della comune carta stagnola (forse un po' più spessa) che da un lato è stata resa adesiva con una colla per alta temperatura.



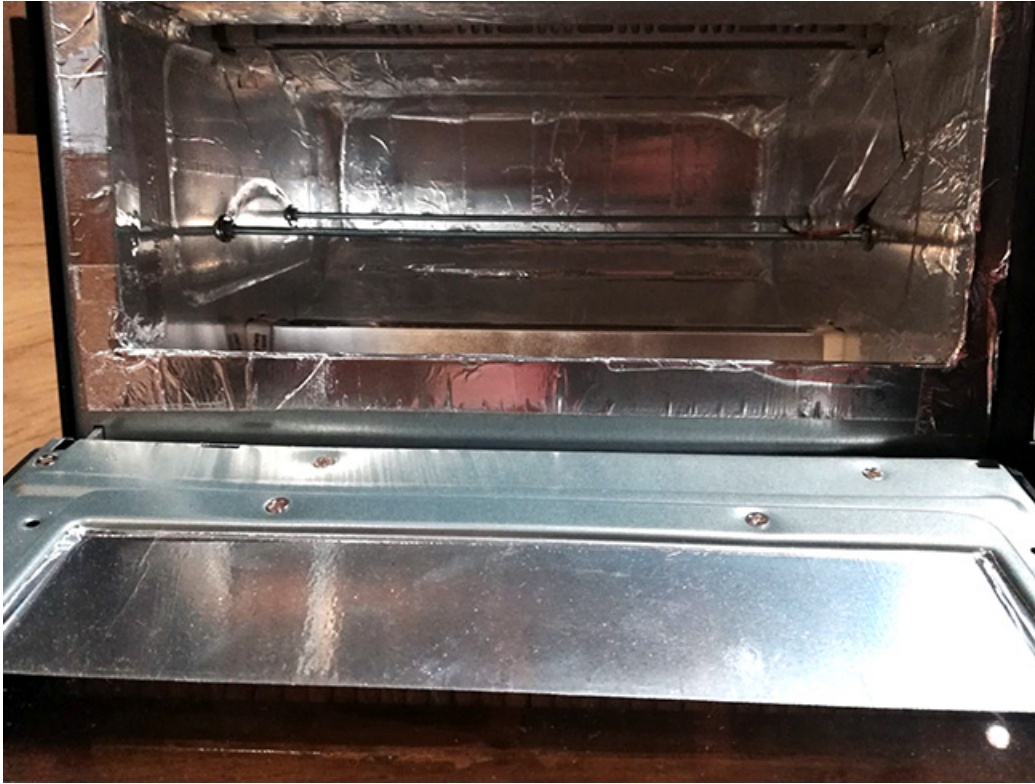
Nastro adesivo in alluminio per l'isolamento della camera interna

Ho completamente rivestito la camera interna, coprendo anche il coperchio in vetro e lasciando solo una sottile fessura per osservare l'interno. Ho inoltre scelto di orientare entrambe le resistenze

al quarzo verso l'interno (in origine tra le resistenza e la camera era interposta una griglia metallica che ne limitava il potere di irraggiamento). La termocoppia K è visibile nella foto seguente sulla destra, in futuro potrebbe essere utile sostituirla con una versione *clamp* da fissare sulla PCB in lavorazione. Ho aggiunto poi due barre filettate "in bolla" per poggiare la scheda.

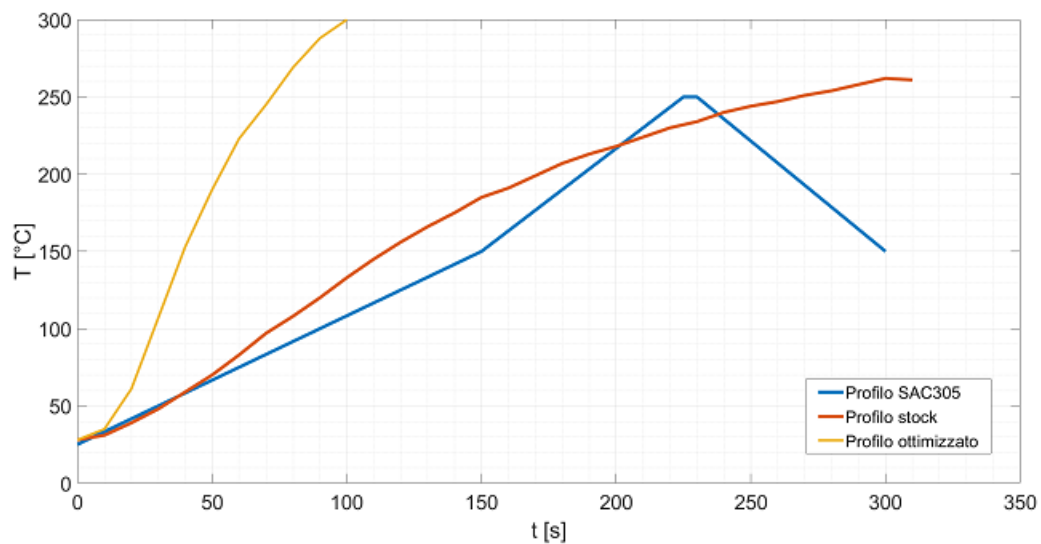


Camera isolata (termocoppia visibile a destra)



Camera isolata

In seguito alle modifiche, la risposta a piena potenza del forno è ampiamente migliorata (anche troppo), come mostra il grafico seguente.



Confronto tra i profili termici stock, SAC305 e con isolamento ottimizzato

Ora che il forno è ben isolato, possiamo essere sicuri di poter seguire robustamente tutti i profili termici che ci interessano.

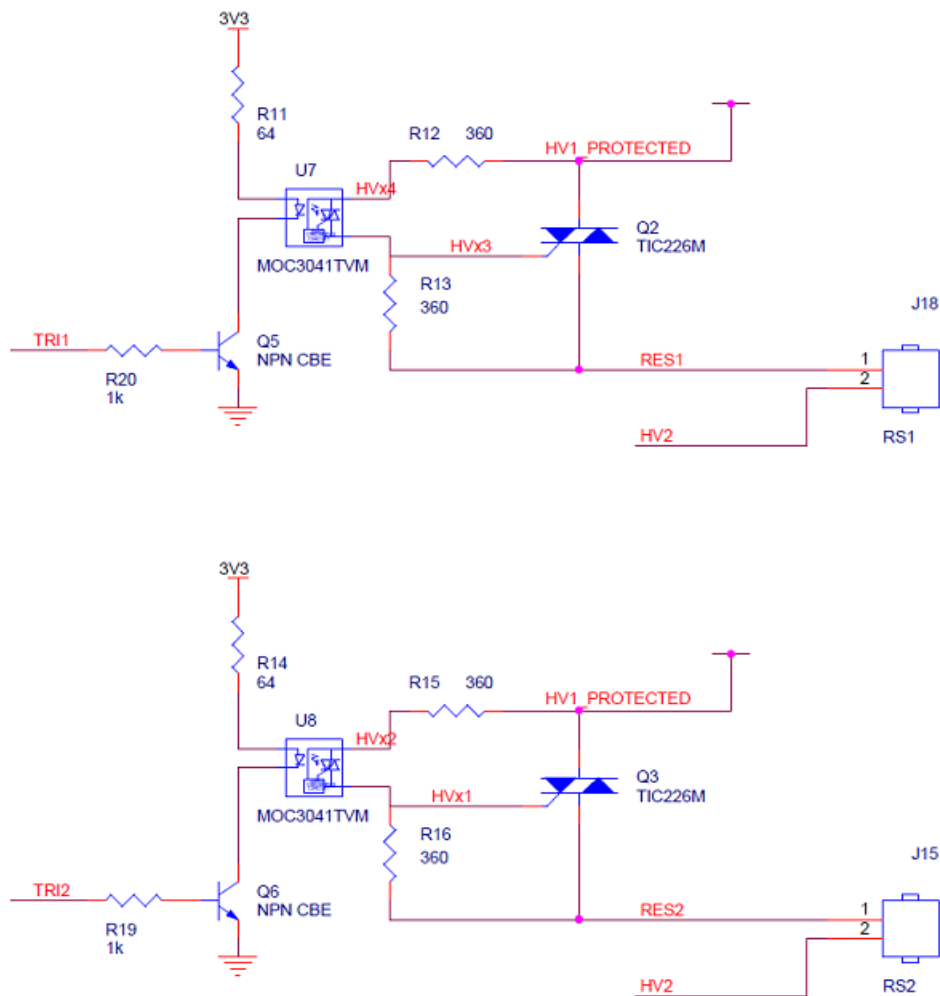
Scheda di controllo

La scheda di controllo che ho realizzato svolge le seguenti funzioni:

1. Misura della temperatura interna
2. Parzializzazione della potenza erogata dalle resistenze
3. Controllo automatico del profilo
4. Gestione interfaccia LCD per monitoraggio dello stato del processo
5. Comunicazione con software di impostazione del profilo e dei parametri del controllore

L'ultimo punto è più o meno facoltativo: volendo il profilo può essere scritto nel firmware e rimanere immutabile, oppure può essere delegato ad una interfaccia più semplice mostrata direttamente su LCD. Nel mio caso ho preferito fare le cose in modo più completo e ho scritto un software di gestione che descriverò a fine articolo.

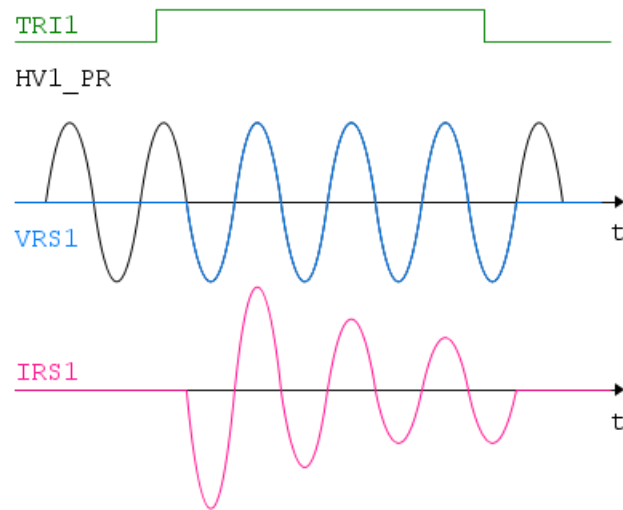
Di seguito, a pezzi, riporto lo schematico della scheda di controllo con una rapida descrizione delle varie parti (non sopporto dover aprire l'immagine a tutto schermo quando leggo gli articoli).



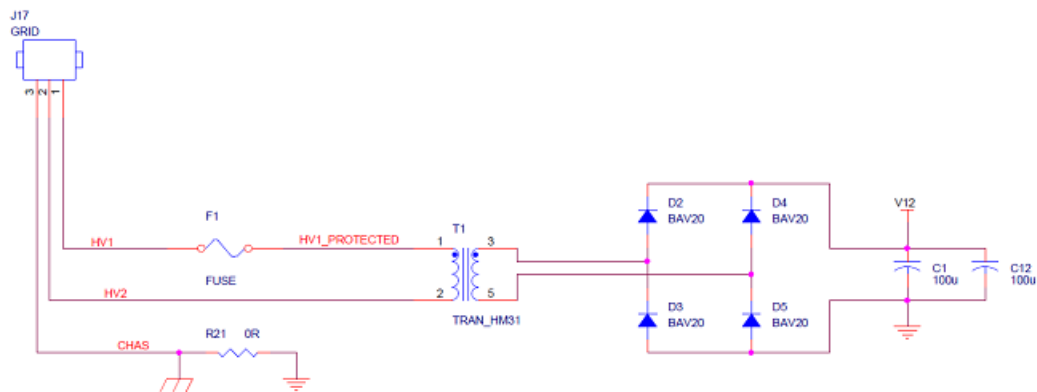
Sistema di innesco delle resistenze di potenza

Le resistenze al quarzo, una volta calde, raggiungono una resistenza di circa $120\ \Omega$. Per controllarne la potenza erogata, è sufficiente modularne il duty-cycle tramite due TRIAC. Due TIC226M pilotano le due resistenze, mentre il loro controllo e l'isolamento galvanico dalla logica è affidato ai due zero-crossing TRIAC driver MOC3041. Questi dispositivi sono composti da un optoisolatore, un mini-triac e un sistema di rilevamento dello zero-crossing della tensione di rete.

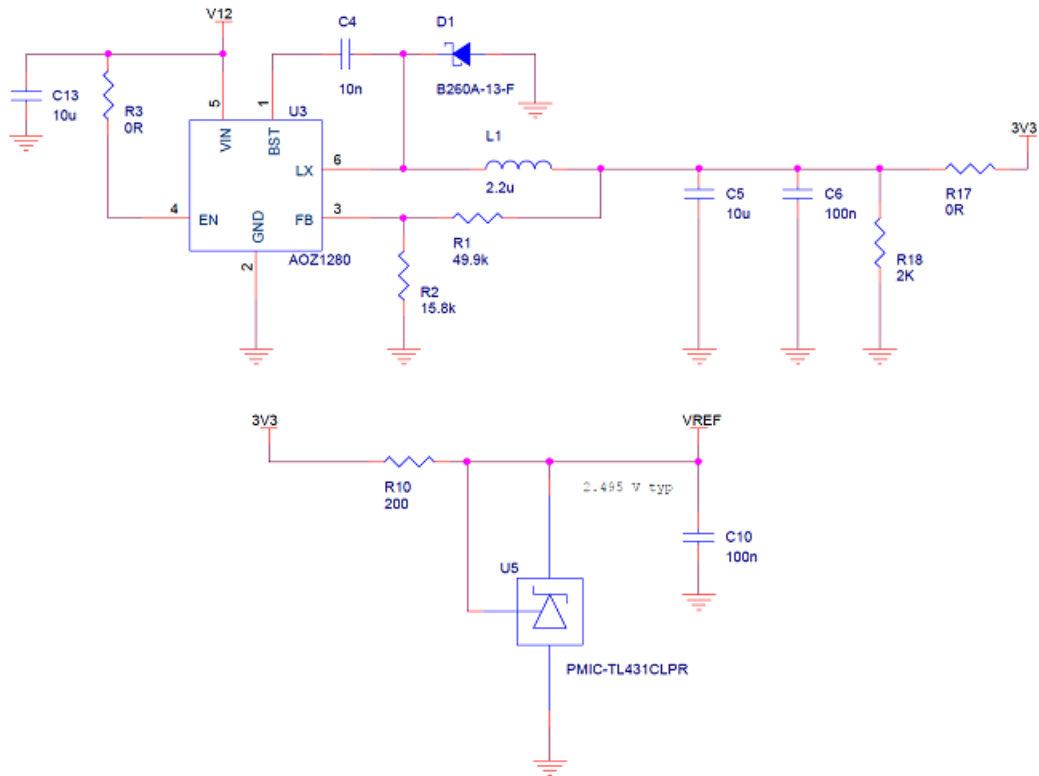
In sintesi, è sufficiente accendere il led al primario per alimentare, dal successivo semiperiodo di rete, la corrispondente resistenza di potenza. Viceversa, in assenza di segnale, l'optoisolatore rilascia il gate del TRIAC facendolo spegnere allo zero-crossing della corrente.



Nello schema, riferito alla prima resistenza, ho voluto evidenziare il fenomeno di riscaldamento che ne incrementa la resistività durante i primissimi istanti di alimentazione che ho osservato con una sonda di corrente.

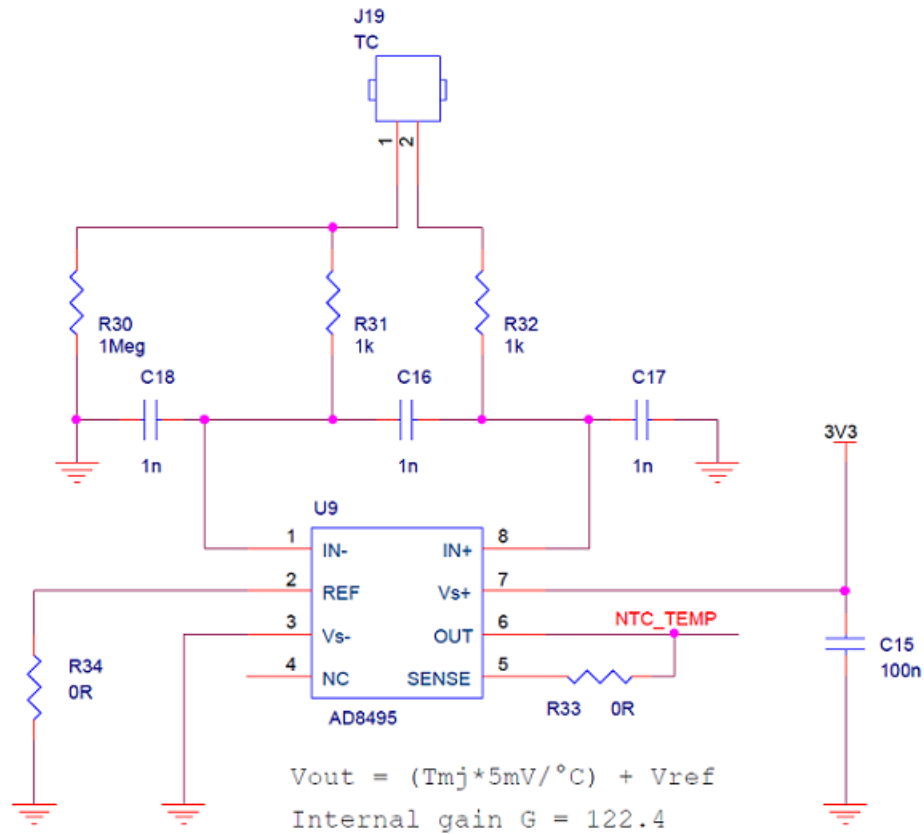


Primo stadio di alimentazione



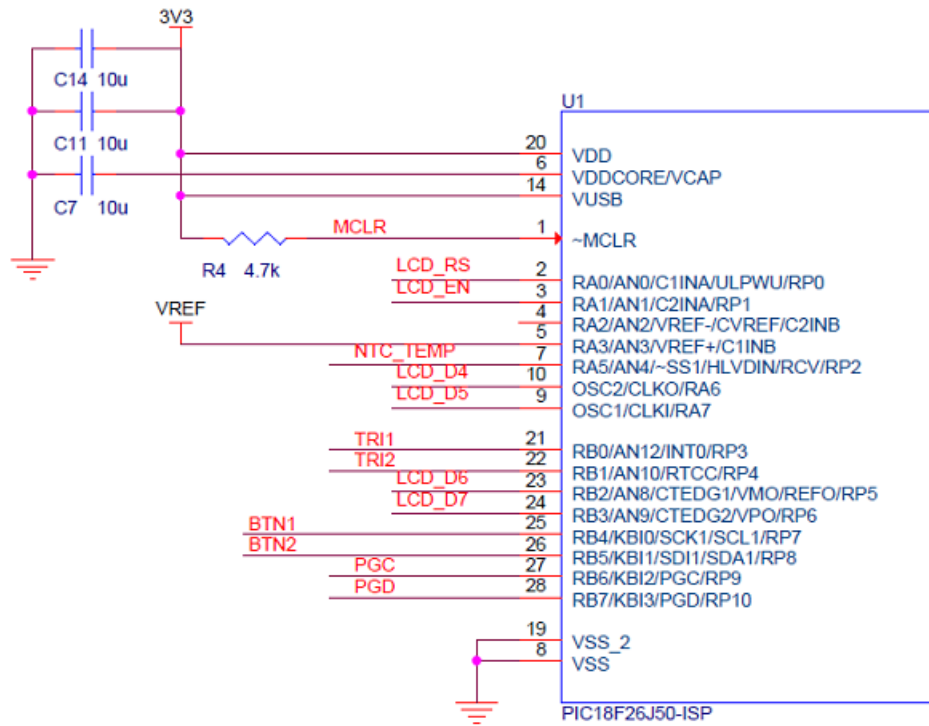
Buck integrato e tensione di riferimento

L'alimentazione per la logica è ottenuta tramite un piccolo trasformatore e un buck integrato. Nell'immagine è mostrata la generazione della tensione di riferimento che sarà usata dall'ADC del microcontrollore per la lettura della temperatura.

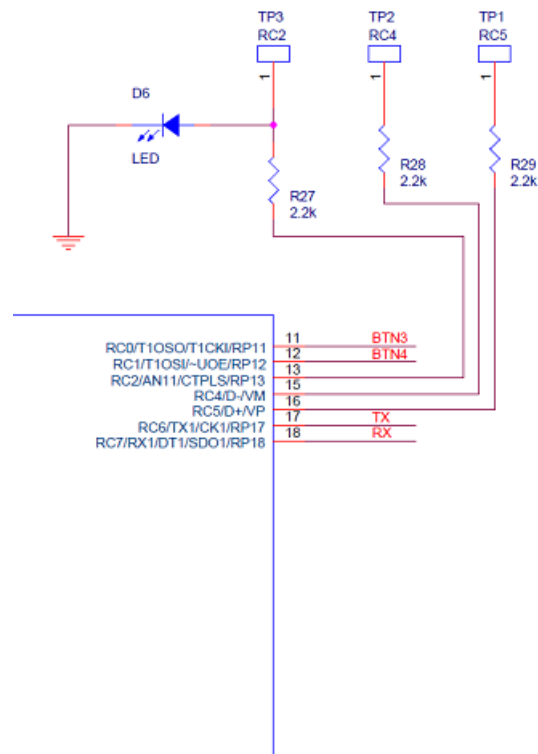


Amplificatore differenziale compensato per la misura della temperatura nella camera

Per la misura di temperatura ho utilizzato una termocoppia K, per semplificarci la vita ho usato un amplificatore completamente integrato con compensazione del giunto freddo.

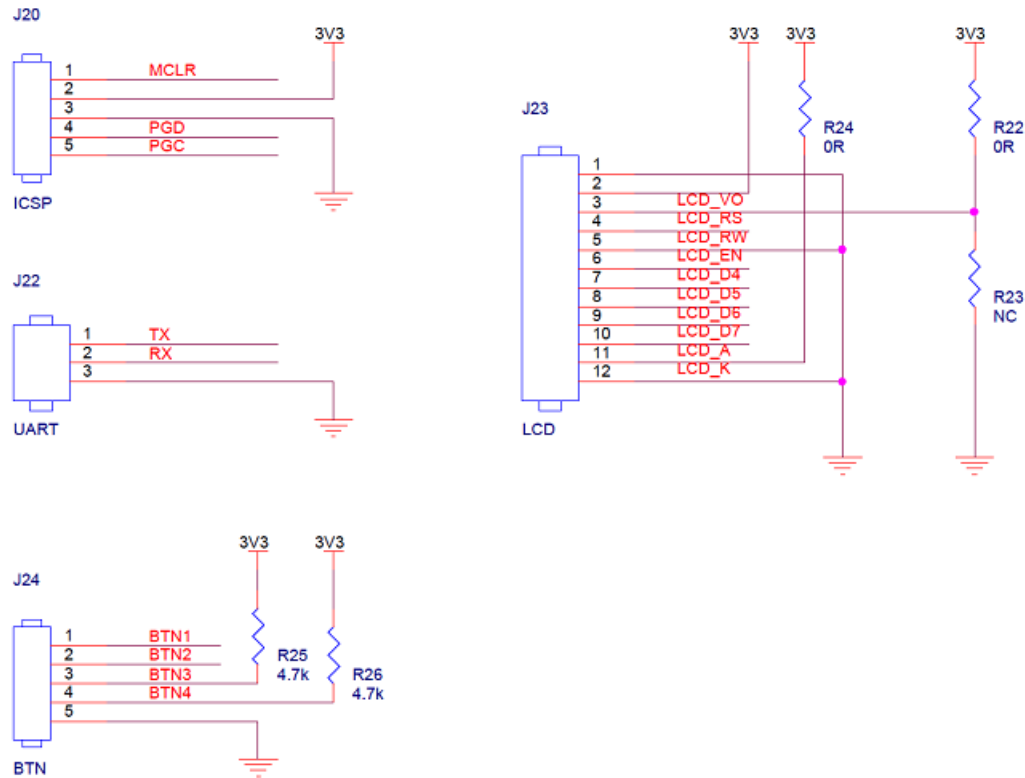


Prima metà del microcontrollore

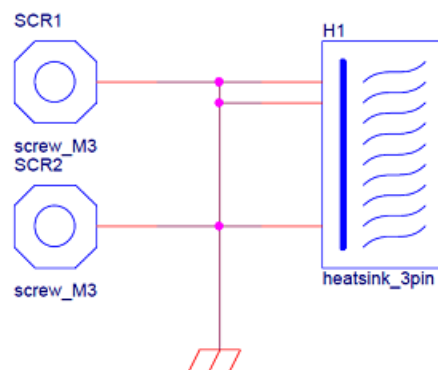


Seconda metà del microcontrollore

Il microcontrollore è un PIC18f26j50, un po' sovradimensionato per questa applicazione, ma ne avevo un po' in laboratorio.



Interfaccia e connettori

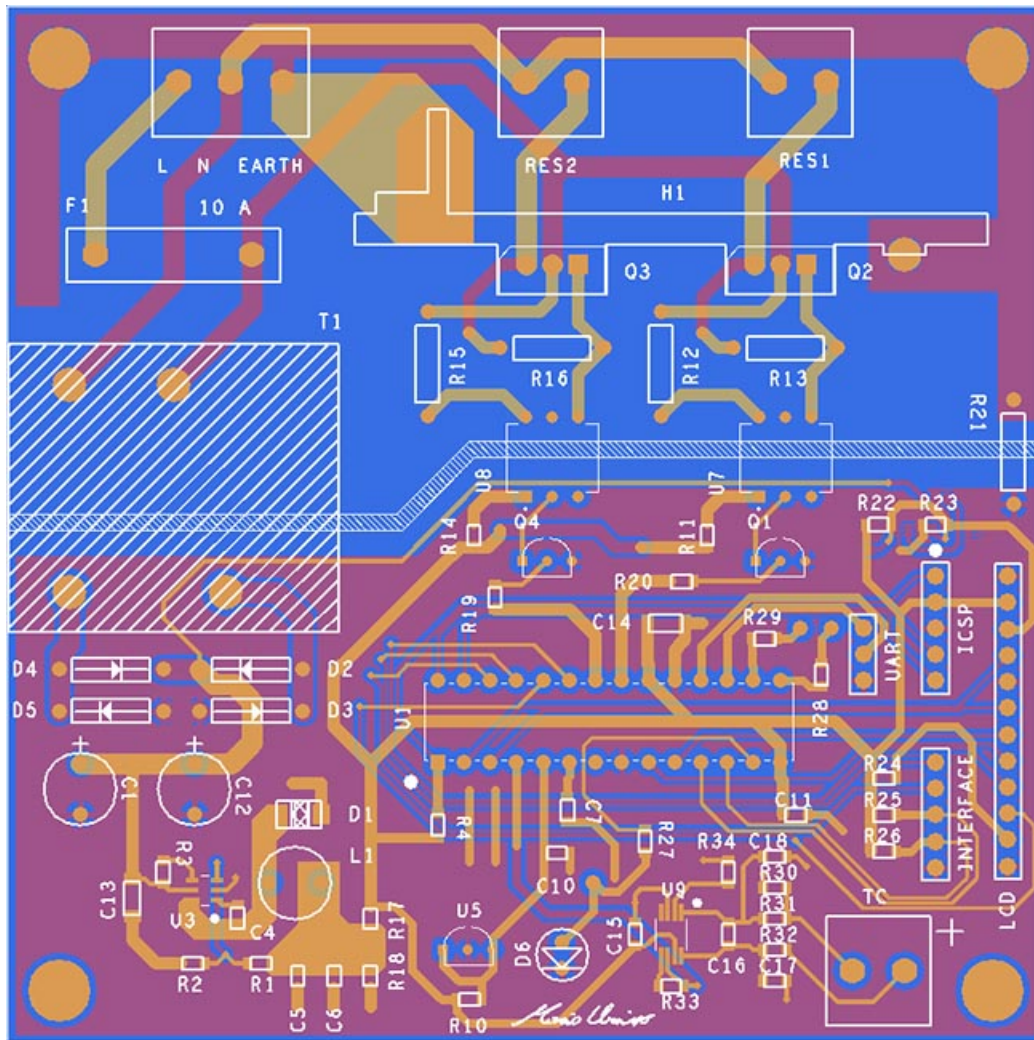


Dissipatore per i TRIAC

Come si vedrà nel seguito, la scheda è collegata ad un LCD alfanumerico e a due pulsanti per iniziare e interrompere il processo di saldatura. Il dissipatore è consigliabile vista la potenza media dissipata dai TRIAC.

PCB e cablaggio

Ho realizzato la scheda di controllo su 2 layers, le dimensioni finali sono di 100 mm x 100 mm.

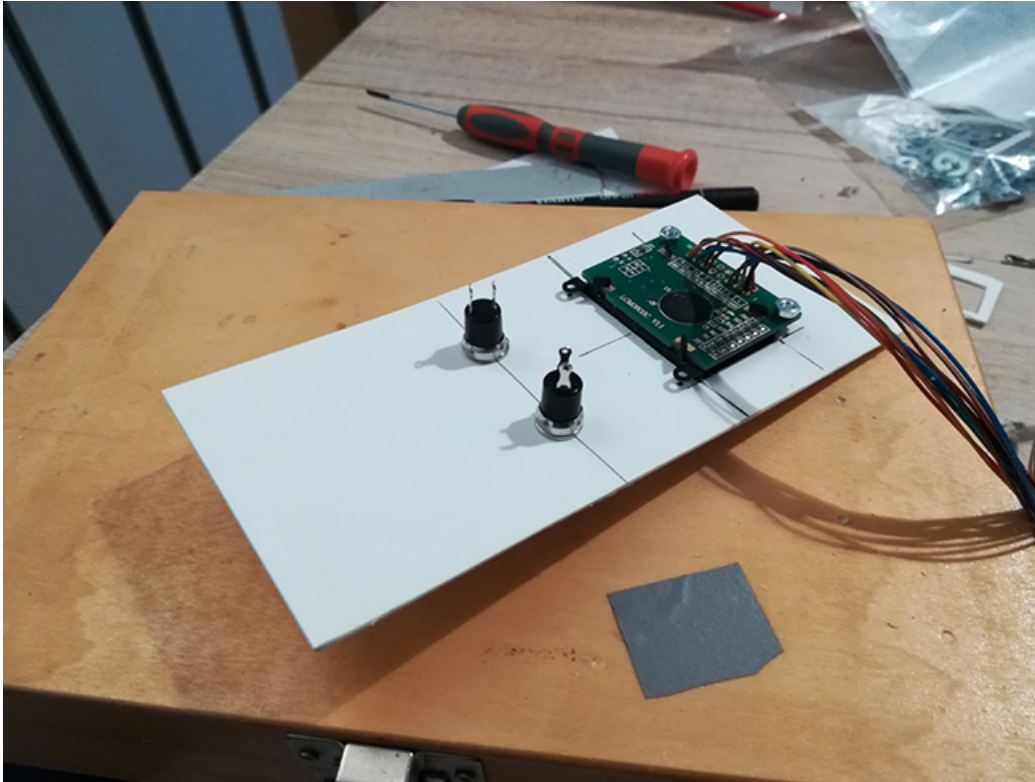


Layout della scheda di controllo

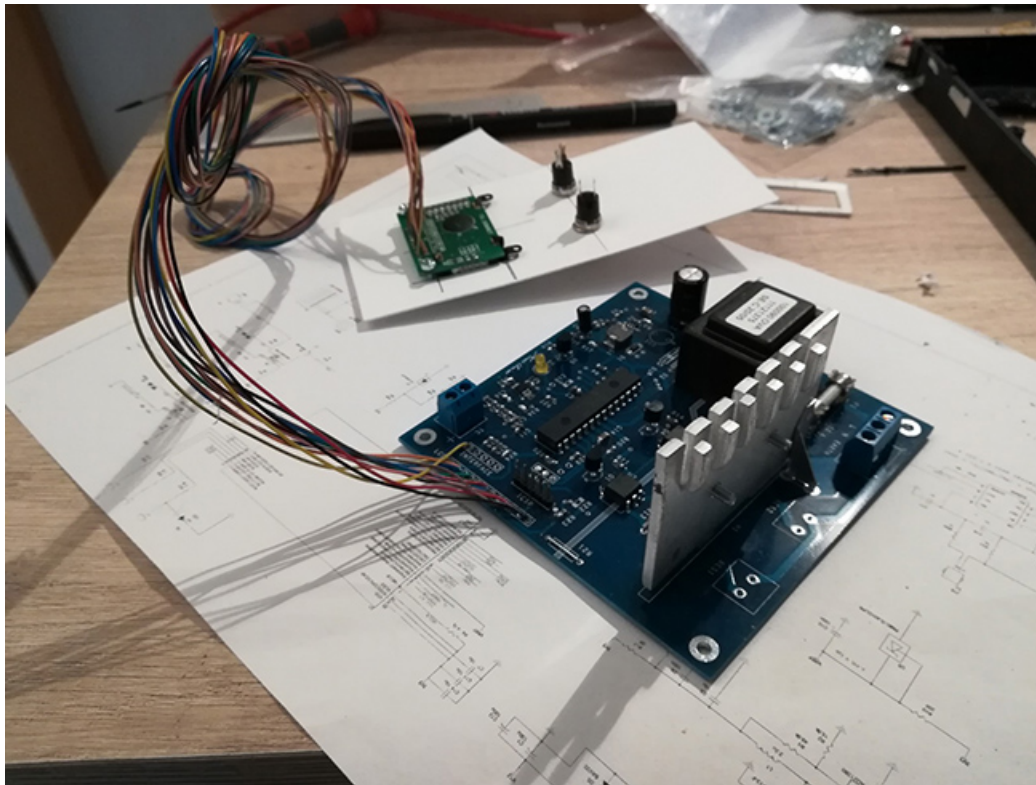
Osservando l'immagine è facile riconoscere la linea orizzontale di isolamento elettrico, che divide la sezione di controllo (in basso) da quella di potenza (in alto). In centro, U7 e U8 sono i gate-driver optoisolati, mentre a sinistra T1 è il trasformatore di alimentazione. A destra, R21 (è uno *short*) collega a terra il riferimento della logica. In alto sono presenti i TRIAC Q2 e Q3, collegati al dissipatore con un foglio di mica, e il fusibile di protezione F1.

Nella sezione inferiore, il microcontrollore gestisce l'intera scheda, rilevando la temperatura attraverso la termocoppia (collegata al connettore TC) e pilotando i driver U7 e U8. A sinistra è visibile il buck integrato con l'induttanza L1. Tutte le connessioni verso LCD, seriale e pulsanti sono a destra, sui relativi connettori.

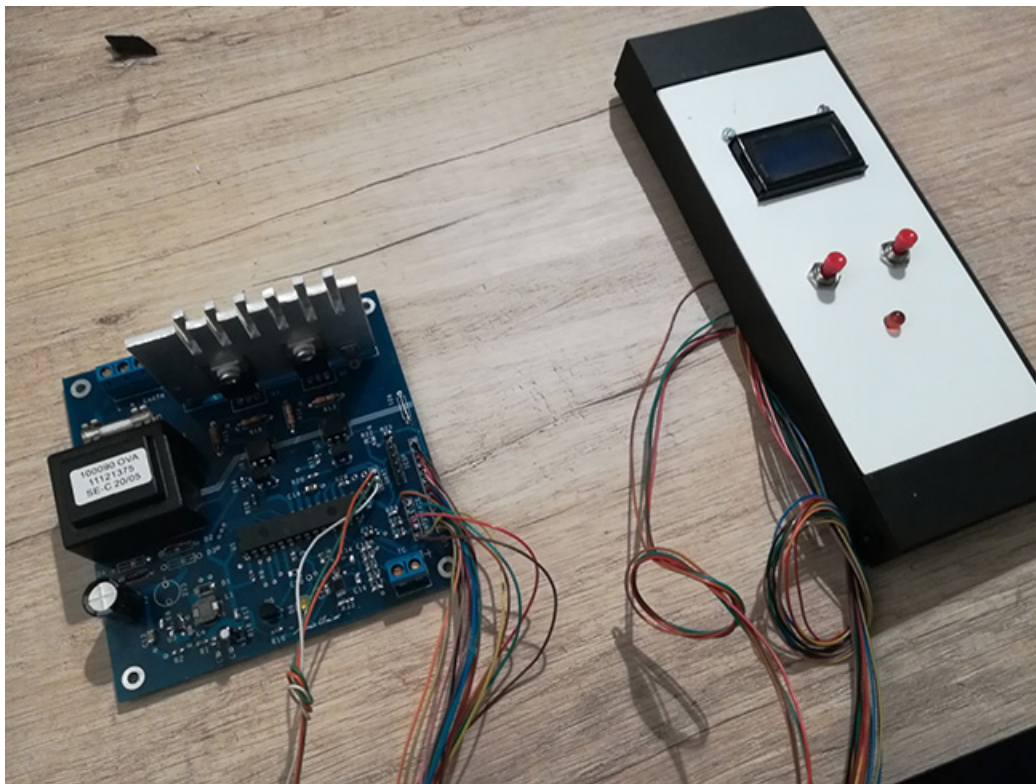
Ho costruito l'interfaccia LCD, con i due pulsanti start-and-stop, utilizzando un pannello in PCV bianco e rimuovendo brutalmente la facciata in plastica che conteneva le manopole originali. Ho inserito un led che comunichi visivamente l'azione di riscaldamento, lampeggiando con lo stesso duty-cycle utilizzato per il pilotaggio delle resistenze.



Interfaccia su PVC

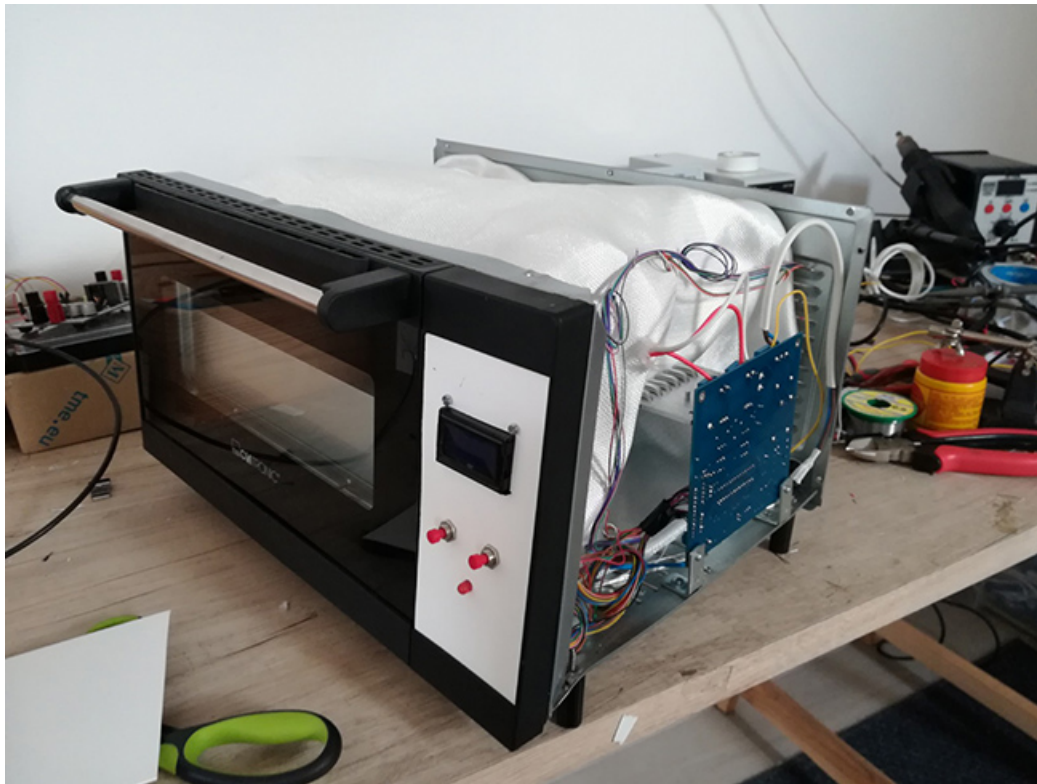


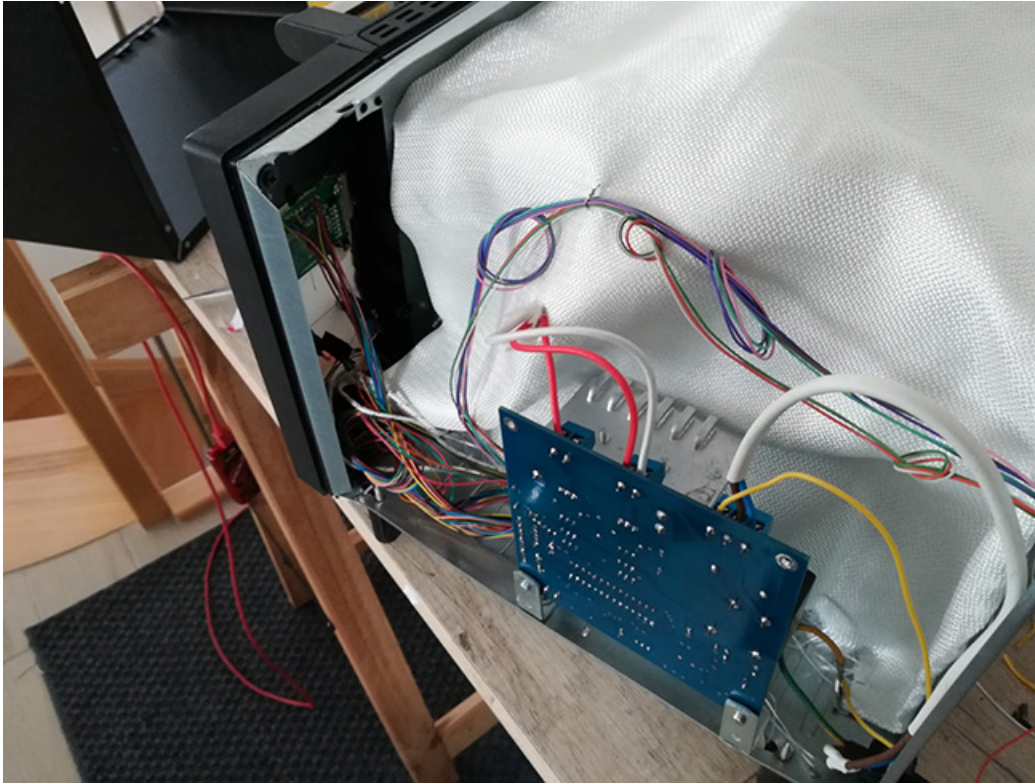
Scheda completa e interfaccia



Scheda completa e interfaccia fissata al case

Per fissare la scheda all'interno del case del forno ho rivettato alla lamiera due sostegni ad "L". Ho inoltre acquistato una coperta termica ignifuga per migliorare l'isolamento della camera dalle pareti del case. Per il collegamento delle resistenze alla scheda di controllo ho utilizzato i cavi originali per alta tensione/alta temperatura, corredati di connettori di sicurezza che ne impediscono lo sgancio accidentale. Tutte le strutture metalliche e il ground-plane della scheda sono collegati a terra. La spina è di tipo tedesco, anche in questo caso ho lasciato il cavo originale. La scheda di controllo è mantenuta a distanza ed isolata dalla parete del case di chiusura con un pannello in PCV aggiuntivo.

*Interno completo del forno*



Interno completo del forno

Firmware e controllore

Il firmware è organizzato come segue:

Ciclo main	TMR2 interrupt (5 ms)	TMR1 interrupt (10 ms)	UART interrupt
• Gestione memoria flash • UART TX • Gestione LCD	• Gestione pulsanti con debounce • Avvio processo • Annullamento processo	• Lettura termocoppia • Incremento stato • PID • Pilotaggio TRIAC • Fine processo	• Ricezione configurazione

Descrizione a blocchi del firmware della scheda di controllo

Hi riportato i blocchi nell'ordine in cui si trovano nel codice (riportato successivamente), così da facilitarne il riconoscimento.

Il ciclo *main* svolge funzioni di bassa priorità ed è ciclicamente interrotto da tre fonti di interrupt. L'unico caso in cui il ciclo a bassa priorità influenza il processo di saldatura è quando la seriale riceve un messaggio di configurazione completo, altrimenti esso è trasparente e le funzionalità principali restano delegate alle funzioni di interrupt. Il ciclo *main*, in questo caso particolare, disabilita gli interrupt e annulla il processo di riscaldamento per scrivere nella memoria programma la configurazione ricevuta tramite software (che sarà descritta meglio nel seguito). Durante il normale funzionamento, il ciclo invia periodici messaggi di stato tramite seriale e aggiorna l'interfaccia LCD.

TMR2 genera un interrupt ciclico che monitora i pulsati, implementa un *debounce* software e determina quando avviare o annullare il processo di riscaldamento in base alla selezione dell'utente.

TMR1 genera un interrupt ciclico esattamente ogni 10 ms. L'accesso all'interrupt incrementa una variabile che di fatto è il timer del controllo. Durante l'esecuzione dell'interrupt viene campionata la temperatura compensata della sonda K, viene deciso l'intervallo del profilo in cui ci si trova e viene interpolata la temperatura di riferimento istantanea. Dal setpoint così calcolato e dal campione dell'ADC si ricava l'errore di temperatura. Aggiornato lo stato del PID, viene decisa la potenza da fornire alle resistenze nei prossimi 200 ms. Di fatto, il controllo regola il duty-cycle (uguale per entrambe le resistenze) di un'onda rettangolare di periodo 200 ms. La frequenza del controllo è quindi pari a 5 Hz.

Di seguito riporto la sezione di codice che svolge la funzione sopra descritta. La variabile *tick* è incrementata di una unità ad ogni ingresso all'interrupt, quindi ogni 10 ms, ed è resettata una volta raggiunti i 200 ms. Al reset (ogni 200 ms) viene quindi calcolato l'errore e deciso se accendere o spegnere i TRIAC. La variabile *sysPower* rappresenta la potenza media erogata dalle resistenze durante il ciclo di 200 ms ed è utilizzata insieme a *tick* per imporre il duty-cycle corrispondente.

```
// PID update
if(tick == 0)
{
    t_error = sysState.Tsetpoint - sysState.Tmeas;

    if(first_lap)
    {
        error_prev = t_error;
        first_lap = 0;
    }

    integrator += t_error*(pidCoeffs.KI/Tctrl);

    // Saturation
    if(integrator > 5)
        integrator = 5;
```

```

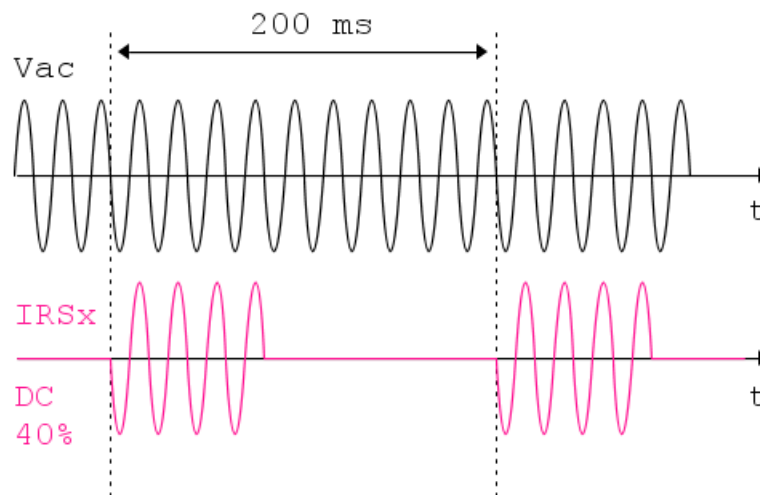
    if(integrator < -5)
        integrator = -5;

    sysState.power = pidCoeffs.KP*t_error + (pidCoeffs.KD*Tctrl)*(t_error - error_prev)
    sysState.power += integrator;
    error_prev = t_error;
}

if(sysState.power > 20)
    sysState.power = 20;
else if(sysState.power < 0)
    sysState.power = 0;
if(tick < sysState.power || sysState.power == 20)
{
    TRIAC1 = 1;
    TRIAC2 = 1;
}
else
{
    TRIAC1 = 0;
    TRIAC2 = 0;
}

```

Imponendo un DC pari al 40 %, quindi, si otterrà il comportamento seguente:



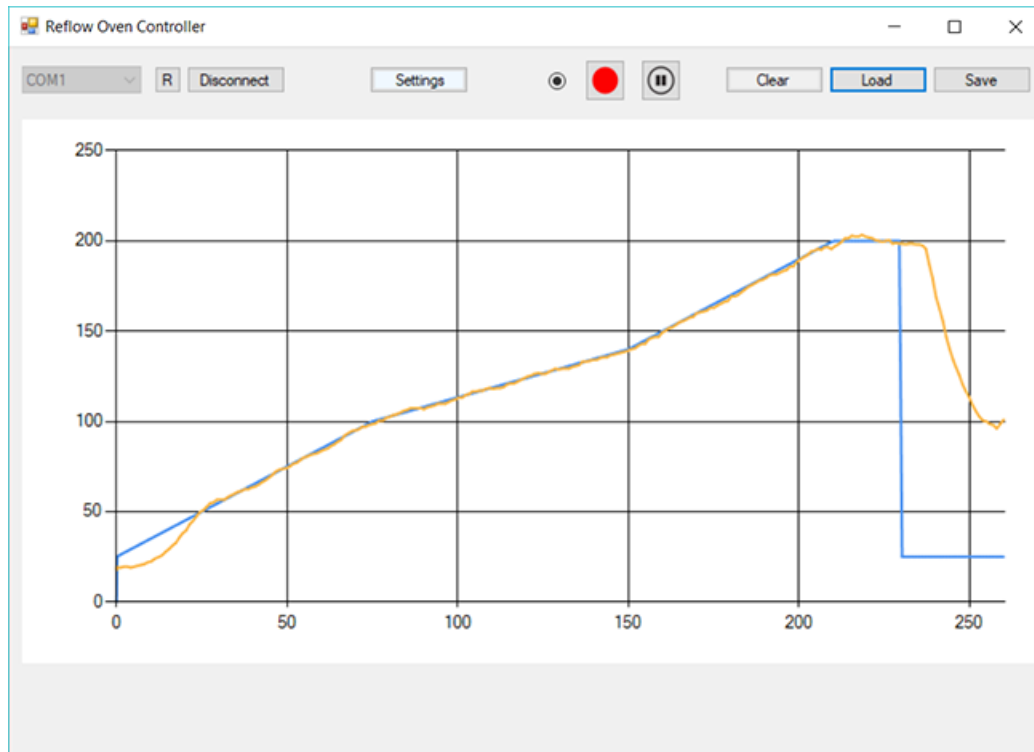
UART, che è la periferica che implementa la porta seriale, genera un interrupt alla ricezione di ogni carattere inviato dall'eventuale software su PC. Alla ricezione di una stringa di configurazione completa, comunica al loop a bassa priorità che è necessario aggiornare la memoria programma con le nuove impostazioni. È possibile configurare il profilo termico oppure i parametri del PID.

Il firmware è stato scritto per il compilatore XC8 Microchip. L'intero progetto MPLabX (contenente i sorgenti .c, per chi volesse visionarli), è disponibile [qui](#).

Software di gestione C#

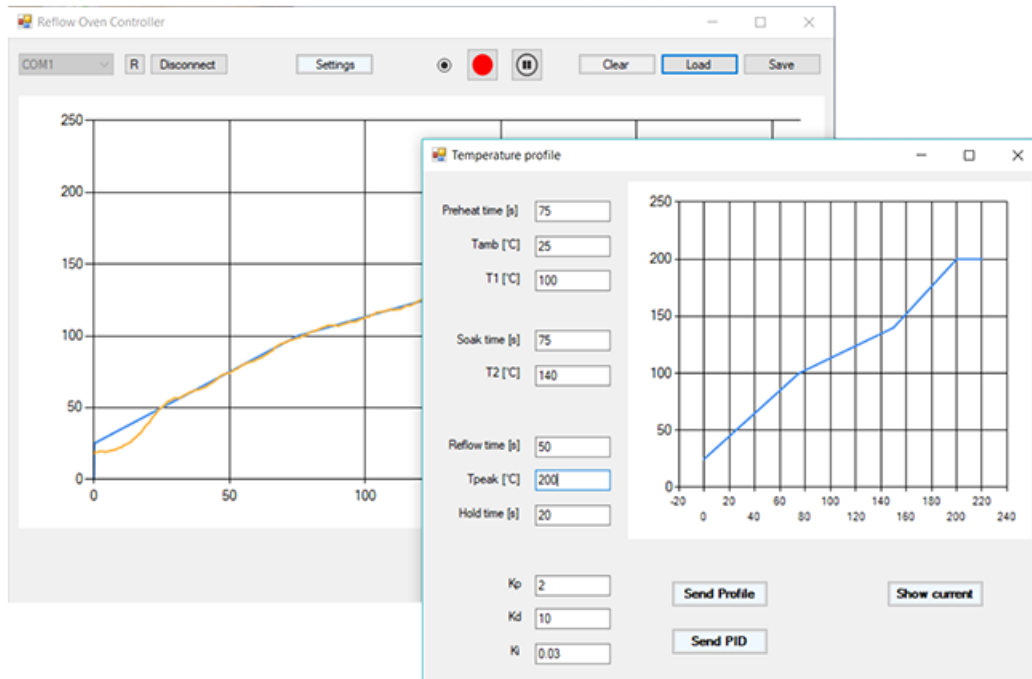
Per la visualizzazione e gestione del profilo di temperatura e, già che c'ero, per tarare i parametri del PID, ho realizzato un software in C# che comunica con il forno tramite porta seriale.

Il software, inoltre, permette di visualizzare in tempo reale la curva di temperatura e il setpoint durante il processo di saldatura. Le curve possono essere salvate e caricate in .csv. Nell'immagine seguente è riportato il profilo richiesto (in azzurro) e la temperatura reale misurata (arancione). Alla fine della fase di *reflow* (picco di temperatura) l'LCD notifica all'utente di aprire la porta per assicurare un raffreddamento rapido della scheda in lavorazione.



Schermata di visualizzazione del profilo termico impostato ed eseguito

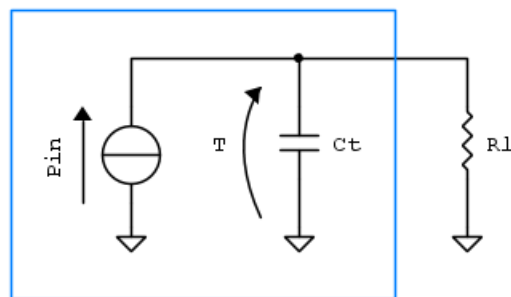
All'accensione del forno il software riceve i parametri attualmente memorizzati. Nella schermata di impostazione è possibile visualizzare il profilo appena ricevuto oppure impostarne uno nuovo. Nella stessa schermata si possono modificare i parametri del controllore automatico.



Schermata di impostazione del profilo e del PID

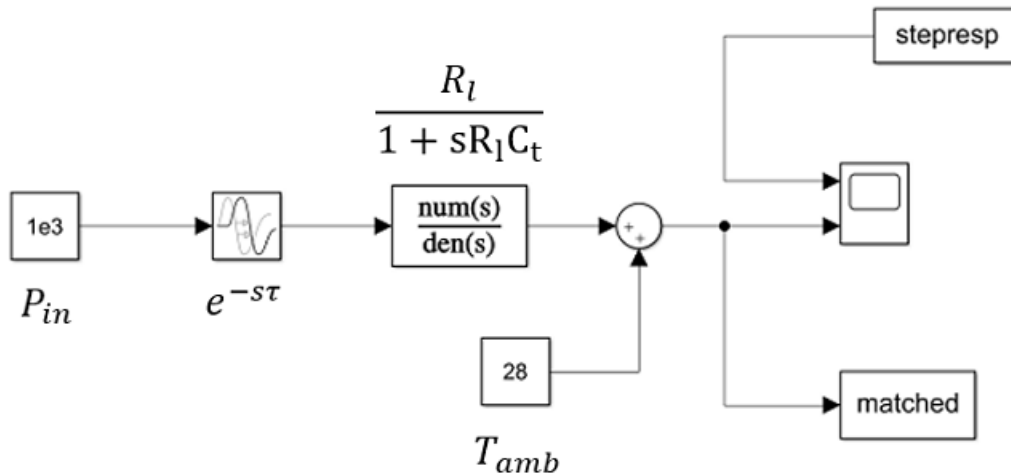
Sistema dinamico controllato equivalente

Può essere interessante tornare al circuito semplificato introdotto all'inizio dell'articolo e approfondire la questione, analizzando il sistema dinamico che descrive l'intero sistema. Questa analisi può essere d'aiuto per selezionare i parametri del controllore automatico.



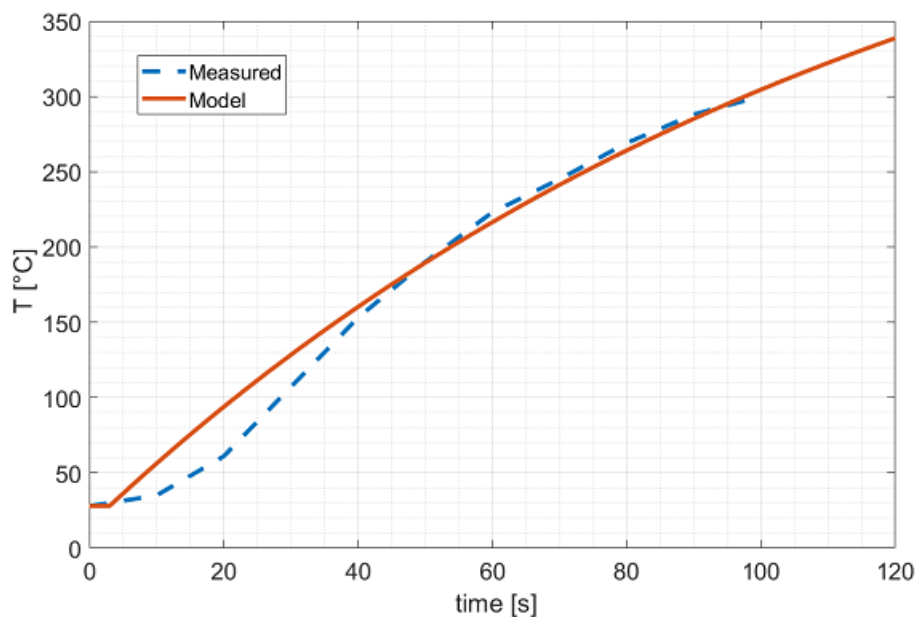
Consideriamo la potenza erogata dalle resistenze come ingresso del sistema e la temperatura della camera come uscita. Implementato il sistema dinamico che descrive il circuito termico, possiamo calibrarne i parametri utilizzando la risposta al gradino precedentemente misurata. Il sistema seguente è utilizzato per tarare la capacità termica e la resistenza di perdita paragonando la risposta misurata *stepresp* alla temperatura simulata *matched*. Al sistema dinamico è stato aggiunto un ritardo che approssimi la propagazione del calore dalle resistenze al centro della

camera, dove è posizionata la termocoppia. L'approssimazione migliore, probabilmente, sarebbe una combinazione pesata di singoli poli che descrivano l'inerzia termica delle resistenza, l'irraggiamento e la convezione. Ho valutato che non ne sarebbe valsa la pena.



Sistema dinamico per la calibrazione dei parametri termici

Con $C_t = 250$ J/K, $R_l = 0.5$ K/W e $\tau = 3$ s si ottiene un'approssimazione sufficiente della curva misurata, come si osserva dalla figura seguente:

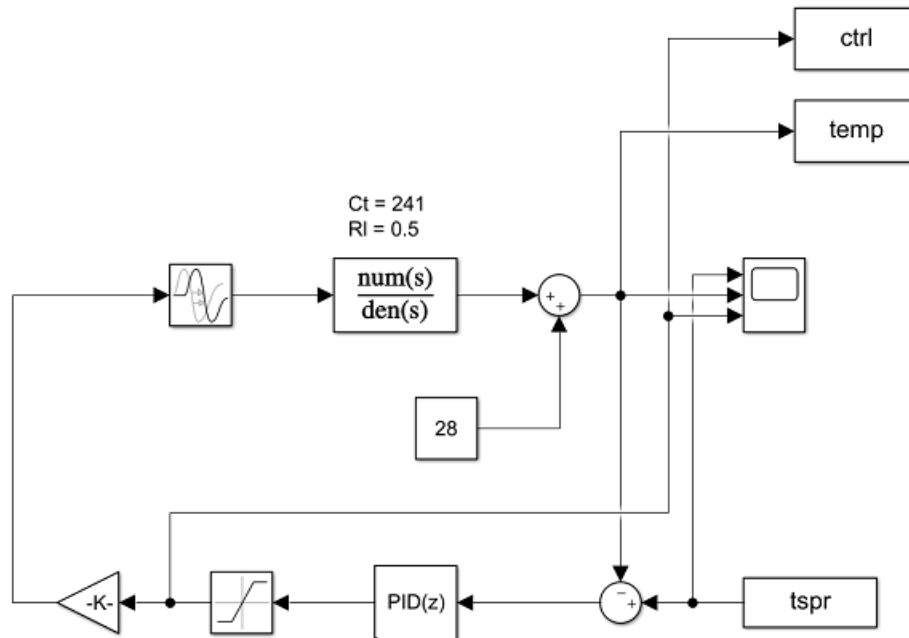


Risposta misurata e simulata con i parametri calibrati

A questo punto, con un sistema dinamico che approssima sufficientemente bene quello reale, possiamo scegliere i parametri del controllo automatico. Questa procedura, ovviamente, può anche essere fatta sperimentalmente.

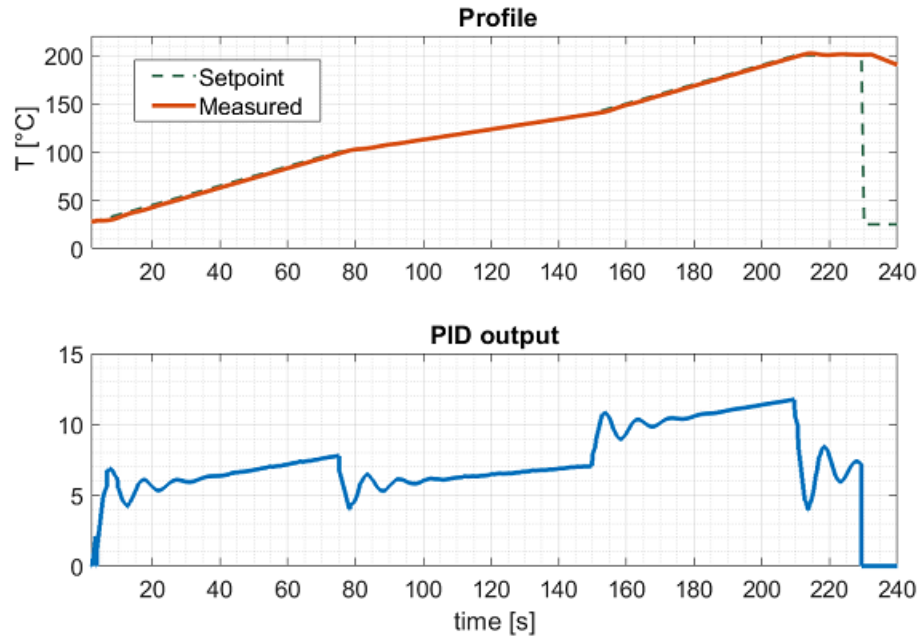
Il sistema dinamico controllato è rappresentato di seguito, dove il controllore automatico è stato implementato nella versione campionata PID(z) con il corrispondente periodo di 200 ms. Nello schema è inoltre riportata la saturazione e il guadagno equivalente *visto* dal firmware (si veda il codice riportato sopra). Al controllore è fornito il profilo da seguire *tspr*. *temp* è la temperatura simulata, mentre *ctrl* è l'output del PID, corrispondente alla *sysPower* del firmware.

Nota: una seconda saturazione, nel firmware, è applicata direttamente all'integratore. Nel sistema seguente questa non è presente. La differenza in questo caso particolare non ha importanza in quando l'integratore non satura mai.



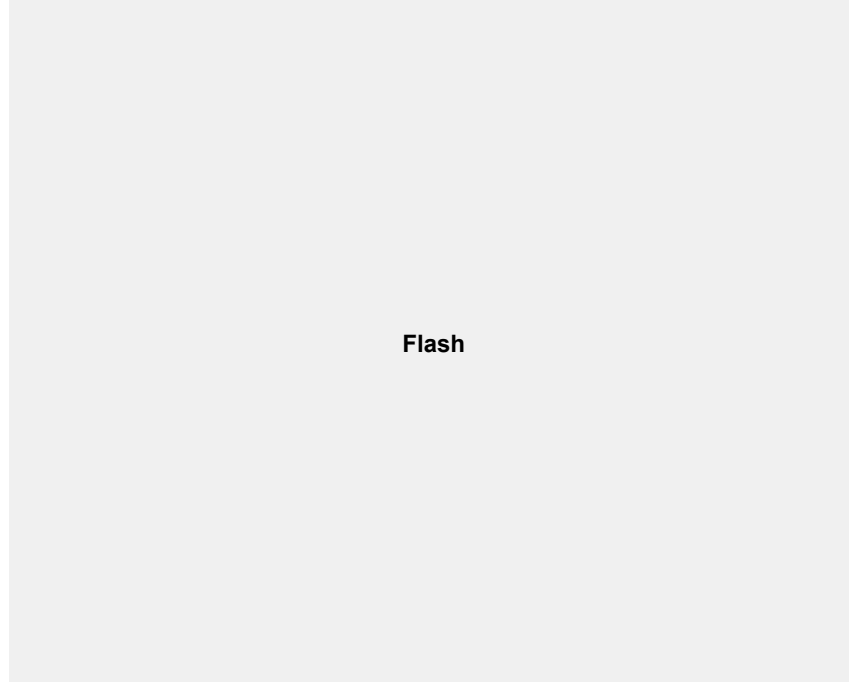
Sistema dinamico controllato

Possiamo simulare l'intero processo di saldatura, osservando il comportamento del controllore automatico.



Profilo simulato e output del controllore PID(z)

Per finire, il video



Flash

Estratto da "<https://www.electroyou.it/mediawiki/index.php?title=UsersPages:Marioursino:diy-reflow-oven>"