

# EMBEDDED MODEL CONTROL IN LABVIEW REAL-TIME

Enrico Canuto | Fabio Musso | Danilo Quagliotti

Presso il Politecnico di Torino si voleva sviluppare e gestire un'applicazione Embedded Model Control (EMC) semplice ma efficace in LabVIEW Real-Time. Erano richieste una frequenza di campionamento di controllo di 10 kHz ed una risoluzione ADC/DAC di almeno 16 bit. Infine, il codice EMC doveva essere uguale a quello sviluppato in un ambiente di simulazione C++ proprietario.

Il codice EMC real-time è stato sviluppato come DLL (Dynamic Link Library) ed incluso in un ciclo 'While' di LabVIEW. Le parti di interfaccia utente, configurazione di controllo, acquisizione dei dati e loro trasmissione e salvataggio, sono state sviluppate in LabVIEW Real-Time. Un'applicazione di regolazione termica semplice ma raffinata ha permesso di testare e dimostrare la bontà dell'approccio.

## DEFINIZIONE DI EMC

L'Embedded Model Control (EMC) è una tecnologia centrata su un'immagine a equazioni di stato di un impianto e del suo ambiente: l'Embedded Model. Tale immagine è incorporata nell'unità di controllo e pilotata in real-time da comandi e disturbi stimati per ottenere una copia delle misure dell'impianto: l'uscita del modello. Le discrepanze fra l'impianto reale e la sua immagine sono messe in luce dalla differenza fra le misure e la loro stima - l'errore del modello - e costituiscono una sorgente sufficiente per gli algoritmi di feedback. L'obiettivo del feedback è quello di estrarre, dall'errore del modello, l'informazione in grado di predire i disturbi stimati. In questo modo, lo stato del disturbo viene costantemente monitorato e può essere regolato grazie ai comandi inviati all'im-

pianto. Se le misure includono richieste dell'operatore, queste vengono convertite in real-time nelle traiettorie desiderate che l'Embedded Model può ripetere per soddisfare l'operatore. L'architettura essenziale dell'EMC (fig. 1) include: l'Embedded Model che opera come un simulatore real-time in parallelo al gruppo impianto/ambiente; la Measurement Law, pilotata dalle misure ADC e dalle richieste dell'operatore; infine, la Command Law, che viene pilotata dalle predizioni dell'Embedded Model e invia i comandi al DAC. Poiché gli algoritmi delle Measurement e Command Law discendono direttamente dall'Embedded Model, è necessario regolare solo i loro parametri. La regolazione può iniziare da procedure computazionali e procedere quindi attraverso la simulazione interna e i test sul campo.

## PERCHÉ LABVIEW REAL-TIME?

Sono state effettuate diverse prove per trovare una piattaforma adatta per lo sviluppo e la gestione del codice EMC, indipendentemente dal fatto che si tratti di apparati industriali, schede sperimentali per applicazioni spaziali o strumenti scientifici. L'EMC real-time richiede infatti:

- Tempi di ciclo deterministici inferiori a 0,1ms (frequenza di campionamento di 10kHz), per garantire l'esecuzione sicura dei moduli EMC,
- Il trasferimento dello stato dell'Embedded Model al computer host e alla memoria di massa in corrispondenza di ogni ciclo: 100 byte nelle applicazioni semplici, pari a 10 Mb/s,
- ADC e DAC accurati, multicanale e con risoluzione superiore a 16/18 bit,
- Un sistema di sviluppo in grado di eseguire la compilazione di codice C++, la fase di linking e il debugging passo-passo del codice a bordo del target,
- Un sistema di sviluppo con interfaccia utente attraente,
- Una piattaforma compatibile con gli standard industriali,
- Prezzi compatibili con i budget universitari.

LabVIEW ha risposto alla maggior parte dei requisiti sopra esposti, con una tendenza alla totale conformità. La conformità è stata testata attraverso applicazioni EMC semplici ma raffinate, sollevando anche nuove questioni che saranno oggetto di future indagini.

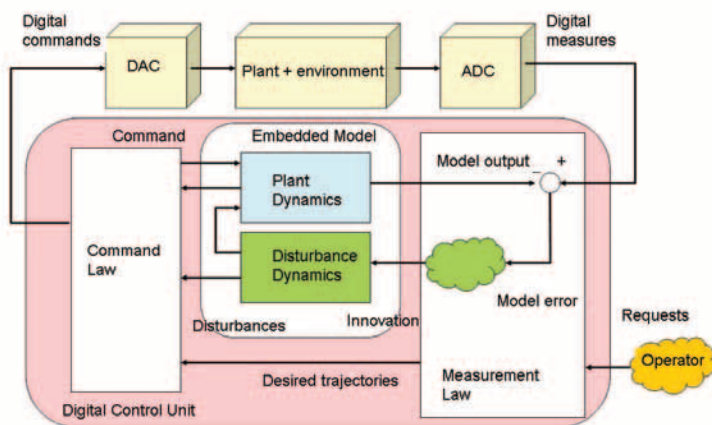


Fig. 1 - Schema a blocchi essenziale dell'EMC

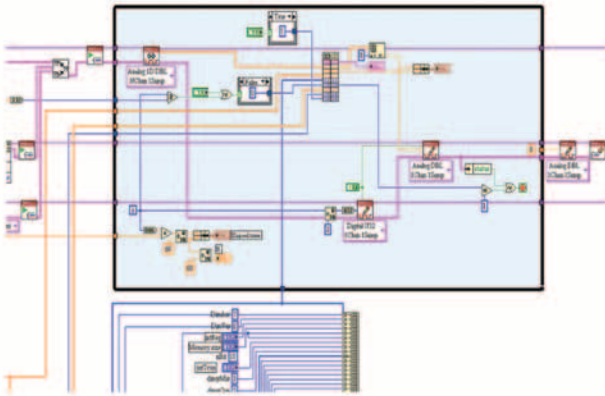


Fig. 3 - Ciclo While dell'EMC e parte della programmazione DAQ e dell'interfaccia di configurazione

**EMC E LABVIEW REAL-TIME:  
RISULTATI OTTENUTI E FUTURI OBIETTIVI**

Il processore real-time (NI PXI-8186, il target) e la relativa scheda di acquisizione (NI PXI-6289 Serie M) sono inseriti in uno chassis PXI-1042.

Il codice EMC e le relative interfacce sono stati sviluppati o incorporati in LabVIEW, installato su PC con Microsoft XP (l'host). Più specificamente:

- L'interfaccia utente, eseguita sull'host, è stata sviluppata in LabVIEW: essa permette l'acquisizione e la tracciatura delle variabili del modello, l'invio asincrono delle richieste dell'operatore all'EMC e funzionalità di configurazione.
- Tutti i moduli EMC sono stati sviluppati sotto MS Visual C++ come DLL (Dynamic Link Library) che possono essere richiamate ed eseguite in qualsiasi istante in un ciclo While di LabVIEW.
- La fase tempi è determinata dalla disponibilità del dato ADC a livello della funzione di lettura DAQmx Read. La ripetibilità è assicurata dal 'Sample Clock' della scheda di acquisizione NI PXI-6289.
- Il codice EMC, sviluppato sull'host, viene poi scaricato ed eseguito direttamente sul target.

Il codice target prevede la seguente semplice architettura sequenziale, in parte illustrata nella figura 2:

- DLL di configurazione: viene eseguita una volta sola per inizializzare i parametri dell'EMC da passare alla DDL del modello come valori di un vettore a doppia precisione oppure intero a 32 bit.
- Il ciclo While esegue inizialmente una DAQmx Read, che attende l'acquisizione del dato ADC definendo l'inizio del tempo di ciclo e a seguire una DAQmx Write, che invia al DAC i comandi precedentemente calcolati.
- Il ciclo While prosegue eseguendo la DLL del modello.
- Il ciclo While termina trasmettendo lo stato dell'EMC, ossia un numero finito di variabili a singola precisione, verso la memoria di massa del target o verso l'host.

**ASPETTO CRITICO**

L'elemento critico dell'architettura sopra illustrata è la necessità di linkare i moduli EMC a LabVIEW sotto forma di DLL. In alternativa, si potrebbe suggerire di sviluppare direttamente i moduli EMC in LabVIEW, assemblando funzioni primitive opportune. Questo potrebbe rappresentare lo sviluppo futuro, ma poiché l'EMC deve essere completamente testato e quindi estratto da un ambiente di simulazione C++ proprietario, sarebbe necessario includere o sviluppare l'intero pacchetto di simulazione in LabVIEW.

Una soluzione di compromesso potrebbe essere offerta da strumenti software di disassemblaggio delle DLL, che però attualmente mancano in LabVIEW. Indipendentemente dai futuri sviluppi, il debugging della DLL sul target appare imperativo, perché le DLL al momento sono inevitabili e il passaggio dei parametri alla DLL può essere debuggato soltanto in LabVIEW.

**UN'APPLICAZIONE SEMPLICE:  
CONTROLLO TERMICO ACCURATO**

Il sistema è una lastra di alluminio da stabilizzare termicamente in aria fra 10°C e 40°C. La temperatura viene misurata tramite un singolo termistore. La lastra può essere riscaldata o raffreddata da tre Refrigeratori Termoelettrici (TEC) montati fra la lastra e una colonna di alluminio che opera come dissipatore termico. La lastra porta un corpo che può essere influenzato da carichi termici variabili. Sono stati adottati due periodi di controllo: ogni 0,1ms l'uscita del termistore viene misurata e mediata per eliminare il rumore di misura dal loop di controllo.

La misura mediata viene quindi decimata ogni 0,1s ed elaborata dai moduli EMC fino all'inizio dei comandi ai TEC (fig. 3). Variando il periodo più lungo è possibile testare i limiti di frequenza di LabVIEW. Il periodo più breve è stato imposto dalla dinamica dell'Embedded Model. La figura 3 illustra il profilo effettivo di temperatura mediato ogni 0,1s e la sua deviazione rispetto alla traiettoria desiderata (<2 mK).

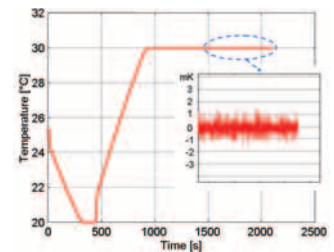


Fig. 2 - Temperatura misurata (mediata ogni 0,1s) e ingrandimento

**CONCLUSIONI**

Lo sviluppo dell'EMC sotto LabVIEW è stato apprezzato: l'interfaccia utente può essere facilmente sviluppata, la programmazione della scheda di acquisizione è immediata ed efficiente quanto la programmazione a registri, l'inclusione della DLL è efficace, a parte i problemi di debugging citati sopra, e frequenze di campionamento fino a 20 kHz sembrano raggiungibili con moduli EMC standard. Il prossimo passo sarà lo sviluppo diretto dei moduli EMC in LabVIEW.

**Note sugli autori**

Enrico Canuto, docente presso il Politecnico di Torino  
Fabio Musso e Danilo Quagliotti studenti presso il Politecnico di Torino

readerservice.it n.308