



## Kirkegaard Winkler (Kirkegaard)

# LO STRAPPO

25 August 2011

Le domande di un folle sono più numerose delle risposte di 1000 savi ( anonimo )

I sistemi di controllo PID come abbiamo già detto sono utilizzati in prevalenza nei processi industriali. Il tipo di controllo dove il pid viene utilizzato maggiormente è quello di temperatura. Questo tipo di sistema è però fra i più “semplici” ( le virgolette sono d’obbligo) da implementare. Questa facilità è dovuta sostanzialmente dal fatto che le derivate di un controllo in temperatura sono poco ripide e in genere in questi casi il parametro derivativo risulta quindi inutile, rendendo possibile la sua omissione e facendo diventare questi controlli da complessi PID a “semplici” PI.

Uno dei primi progetti firmware che ho sviluppato è stato il più difficile mai affrontato. In questo caso il modulo doveva controllare in velocità un motore asincrono monofase. Che detta così sembra una cosa cosmica, ma per molto tempo fra gli amici ero quello che faceva girare i cestelli delle lavatrici....

La complessità di questo controllo era dovuta del fatto che il motore asincrono ha risposte molto più veloci rispetto a quelle di una, ad esempio, camera termica. Questo rende necessario, meglio dire indispensabile, l’implementazione del terzo parametro derivativo. Questo parametro però tipicamente crea notevoli difficoltà nella fase di tuning perchè è fortemente condizionato dallo stato del sistema. In molti casi potrebbe essere necessario avere più set di parametri perchè questo permetterebbe di adattare la risposta a seconda dello stato attivo o disattivo del sistema. Tento di spiegarmi meglio entrando in un campo che è più vicino ai fisici che ai sistemisti come me.... Nel caso appunto di un motore asincrono la derivata dell’accelerazione è molto diversa rispetto alla derivata della decelerazione. Questo è dovuto a diversi fattori che includono l’inerzia del motore stesso, ma non solo. Questa differenza nel sistema è una dissimetria ed è difficile da modellizzare analiticamente dal controllo, così i parametri del pid dovranno “supportare” anche questo tipo di situazione.

Le dissimetrie sono molto fastidiose da vedere, sono poco estetiche, fanno sbuffare nella loro inevitabilità. Troppo spesso passiamo il tempo a cercar di far quadrare il cerchio e magari alla fine ci illudiamo pure di esserci riusciti. Anche i controlli di temperatura hanno questa dissimetria ma come abbiamo detto poc'anzi le derivate sono molto più lievi e le risposte del sistema molto più lente e questo agevola non

poco il controllo di processo. Ma ritorniamo al nostro motore asincrono perché le sorprese non finiscono. I motori di questo tipo hanno una velocità operativa superiore ai 10000 rpm ( rpm = rotate per minute ). Il mio attuatore era un triac e la regolazione anziché essere di tipo **burst firing** era a **taglio di fase** e questo ci obbligava a lavorare all'interno della semionda di rete. Nel nostro paese la frequenza della rete è di 50Hz e così in 10 msec dovevo riuscire a stabilizzare in velocità il nostro motore. Parlare di 10 msec è umanamente un tempo piccolo ( un battito di ciglia sono circa [100 msec...](#) ) ma non per un microcontrollore che può arrivare a qualche nanosecondo per istruzione. Ogni controllo ha il suo feedback che nel nostro caso era un segnale di tachimetrica che andava dalle poche decine di hertz a qualche decina di kilohertz, anche in questo caso uno scherzo per un tipico ingresso contatore di un micro moderno.

Ma... e c'è sempre un ma nelle cose che rende la vita così infausta e caotica tanto da renderla difficile, ostile ma mai noiosa. Dicevamo... In questo momento non parliamo di un micro moderno ma di un micro vecchio (!) di venti anni fa. Cosa è accaduto in venti anni ? Tutto e niente i microcontrollori di venti anni fa erano più essenziali e le prestazioni alla fine, dico quelle del core, non sono migliorate di molto. Sorrido sempre quando guardo sistemi embedded con Java che vengono utilizzati per accendere dei led...

Quel micro era il nuovissimo microcontrollore della Microchip ed era il [PIC16C57](#). La scrivania era sgombra e sopra avevo due fogli fotocopiati con il pinout del micro. Sergio, il direttore tecnico non ufficiale, mi aveva dato qualche sommaria indicazione per quel progetto parallelo che da lì a poco mi avrebbe occupato per i prossimi sei mesi. Non era necessario che si dilungasse troppo in descrizioni tecniche anche perché ero sempre un novizio, ma avevo già concrete precoci esperienze. Ricordo comunque che mi accorsi subito di una caratteristica di quel micro leggendo velocemente quei fogli fotocopiati. Quel microcontrollore a differenza di quelli che avevo utilizzato fino a quel momento, tipo l'8051 o il 68HC11 o anche lo z80, non aveva **interrupt**. Ebbene si non aveva e non ha interrupt.

Decidere di mettere un microcontrollore su una scheda lavatrice era una cosa innovativa perché esistevano già integrati che facevano questo autonomamente e molto bene. Sino a quel momento questo tipo di mercato era particolarmente statico e in Italia l'innovazione era solo di riuscire a diminuire il costo dell'elettrodomestico. Con l'avvento dei microcontrollori si poteva però incominciare a pensare a qualcosa di meglio e la funzionalità inventata era quella dell'antisbilanciamento. Questa funzione consisteva nel modulare la velocità di centrifuga a seconda del carico sbilanciato del cestello. Prima di avviare la centrifuga il modulo doveva fare delle verifiche sul carico per poi avviare la velocità di centrifuga più alta possibile ( considerando anche che più è alta più i panni sono asciutti... ). Questa funzione avrebbe evitato le famose passeggiate dell'elettrodomestico nei bagni dei consumatori e

anche in qualche caso i danni che l'alta velocità della centrifuga causava.

Ora però in verità la nostra storia è molto lontana nel tempo rispetto al progetto sviluppato. Questa premessa ci torneranno utili per capire il resto della storia. Il modulo era terminato e funzionante e la ditta a cui stavamo tentando di vendere il controllo motore era la San Giorgio una storica azienda produttrice di lavatrici. Un giorno i moduli che avevamo già prodotto e che erano in continuo test non superarono una prova. Il tecnico di laboratorio montò un nuovo set di cinghie sulla lavatrice. Le cinghie servono per collegare l'albero del motore elettrico alla ruota del cestello. Con la pressante globalizzazione che li avrebbe da lì a poco ingoiati, trovarono un nuovo economico fornitore e così ne comprarono uno stock in prova. L'esperto tecnico un giorno chiamò contestando al nostro modulo un fastidioso, rumorino dovuto al leggero slittamento della nuova cinghia montata sulla lavatrice. Ovviamente questo accadeva nell'elettrodomestico dove lavorava il nostro modulo e non in quello della scheda concorrente che era prodotta da quelli della [Procond](#). Questo rendeva il nostro modulo non accettabile per la produzione e la logica simmetrica inventata dalla Martini diventò inevitabile... no produzione no stipendio...

Estratto da "<http://www.electroyou.it/mediawiki/index.php?title=UsersPages:Kirkegaard:lo-strappo>"