Informatica e Sistemi in tempo reale

Visualizzatore ECG

//Immagine

//Data

Alessia Guida 522181 [alessiaguida6@gmail.com](mailto:alessiaguida6@gmail.com)

Lorenzo Irace 501354 [irace.lorenzo@gmail.com](mailto:irace.lorenzo@gmail.com)

# Descrizione generale

L’applicazione realizzata si occupa di riprodurre un elettrocardiogramma prelevando i dati da un file e di rilevare la frequenza al fine di segnalare eventuali anomalie.

Il segnale ECG rappresenta le correnti rilevate sulla superficie del corpo umano generate dall’attività cardiaca. Ne risulta una struttura pseudoperiodica, come si può vedere in figura.

//disegno

Il calcolo della frequenza cardiaca può essere effettuato tramite l’identificazione dei complessi QRS che corrispondono ai battiti. Essi sono dati dalla somma di tre onde:

* Onda Q: è negativa e di piccole dimensioni, e corrisponde alla depolarizzazione del setto interventricolare;
* Onda R: è un picco molto alto positivo, e corrisponde alla depolarizzazione dell’apice del ventricolo sinistro;
* Onda S: è un’onda negativa anch’essa di piccole dimensioni, e corrisponde alla depolarizzazione delle regioni basale e posteriore del ventricolo sinistro.

Per riconoscere tali complessi si utilizzano algoritmi basati sul calcolo delle derivate. La nostra scelta è ricaduta sull’algoritmo di Ahlstrom-Tompkins:

1. Calcolo del modulo della derivata prima
2. Filtraggio
3. Calcolo del modulo della derivata seconda
4. Calcolo della somma del modulo della derivata prima filtrato e del modulo della derivata seconda
5. Calcolo delle soglie

Si ha un complesso QRS se un punto supera la soglia primaria e un certo numero di punti successivi supera la soglia secondaria. Nel caso di un ECG campionato a 250 Hz si considerano i 6 valori successivi, 4 nel caso di 120 Hz. //da correggere

Tramite la frequenza è possibile diagnosticare le seguenti patologie:

* Bradicardia: riduzione critica del ritmo cardiaco, mediamente sotto i 50 battiti al minuto
* Tachicardia: seria accelerazione del battito cardiaco, mediamente sopra i 120 battiti al minuto

Tali parametri possono però variare in base alle caratteristiche ed alla storia clinica del paziente, pertanto è utile che all’utente venga permesso di modificare i valori limite per la segnalazione delle irregolarità.

# Interfaccia utente

//immagine

L’interfaccia può essere divisa in 4 sezioni: il grafico dei valori, la frequenza cardiaca, i led per le segnalazioni e le soglie di decisione.

Il grafico viene fatto scorrere inserendo progressivamente i nuovi valori con una rappresentazione ciclica (i primi valori vengono sovrascritti una volta raggiunto il termine destro dell’area dedicata).

La frequenza cardiaca è mostrata in alto a destra nella schermata. Quando questa non rispetta i vincoli indicati per le patologie si “accende” il relativo led.

//immagine led

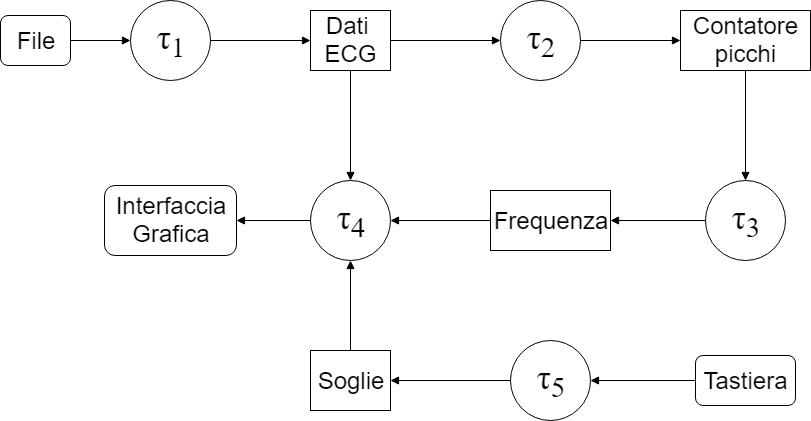
I valori di soglia considerati sono mostrati a video sopra il grafico. L’utente li può modificare da tastiera:

* Tasto ↑ : incrementa la soglia massima (tachicardia)
* Tasto ↓ : decrementa la soglia massima
* Tasto → : incrementa la soglia minima (bradicardia)
* Tasto ← : decrementa la soglia minima

# Task

L’applicazione viene realizzata tramite l’interazione di diversi task, ognuno con specifiche funzionalità. In particolare:

* Un task di lettura τ1
* Un task per il calcolo dei picchi tramite complessi QRS τ2
* Un task per il calcolo della frequenza cardiaca τ3
* Un task che si occupa della grafica a schermo τ4
* Un task per leggere i comandi da tastiera τ5



Il task di lettura preleva i singoli campioni da un file .txt e li salva in un array circolare. Dopo aver raccolto almeno 5 dati, inizia a segnalare al task di calcolo dei picchi che i valori a disposizione sono sufficienti per il suo lavoro. Questo, applicando l’algoritmo di Ahlstrom-Tompkins, calcola Y3(i-2) alla segnalazione del dato X(i). Sfruttando questo valore, verifica quindi se sono presenti le condizioni per un picco e, in caso affermativo, incrementa il relativo contatore. Tuttavia, essendo le soglie calcolate facendo riferimento al massimo valore di Y3 ottenuto con tutti i campioni, si necessita di un numero di campioni sufficiente ad approssimarle in modo adeguato, per poi procedere ad un progressivo aggiornamento. Tenendo in considerazione una frequenza di campionamento pari a 150 Hz e il fatto che in un secondo è generalmente compreso un battito o almeno una parte del complesso QRS, si sfrutta un set di 150 valori per inizializzare le soglie.

## Task di lettura

Il task di lettura preleva i singoli campioni da un file .txt e li salva in un array circolare. Dopo aver raccolto almeno 5 dati, inizia a segnalare al task di calcolo dei picchi che i valori a disposizione sono sufficienti per il suo lavoro.

Il task è periodico con periodo dipendente dalla frequenza dell’ECG. Considerando tale valore pari a 120 Hz, è stato scelto un periodo di 8000 microsecondi. La deadline coincide con il periodo in quanto, per compiere la lettura, si ha a disposizione tutto il tempo fino al passaggio ad un nuovo dato.

La priorità del task è massima in quanto tutti gli altri task dipendono dal successo della lettura.

Le risorse condivise a cui accede sono l’array che contiene i dati attualmente mostrati a video con il relativo indice per l’inserimento e la variabile condition per segnalare al task di calcolo dei picchi la presenza di un nuovo dato. È stata preferita una variabile condition ad un semplice semaforo di sincronizzazione per evitare molteplici letture degli stessi dati da parte del task di calcolo, cosa che si sarebbe altrimenti potuta verificare in seguito a più letture consecutive senza riuscire ad eseguire il calcolo, e quindi a più signal sul semaforo successive. Ad esempio, supponiamo che t1 abbia eseguito due letture e che, prima della successiva, t2 riesca ad effettuare due volte il calcolo. Il task di calcolo legge ogni volta a partire dall’indice per l’inserimento, quindi:

* Con il semaforo di sincronizzazione, il task di lettura ha incrementato due volte il semaforo e pertanto le due wait del task di calcolo vanno a buon fine ma, prendendo i dati a partire dallo stesso indice, si calcolano due valori di y3 uguali, uno dei quali risulta scorretto.
* Con la variabile condition, il task di calcolo prende i dati la prima volta e resetta la variabile. Di conseguenza, la seconda volta trova la variabile a zero e si blocca. In questo modo viene prodotto un solo y3 (“saltando” il valore del periodo in cui non è stato eseguito il calcolo) ma non vengono generati dati scorretti.

## Task di calcolo dei picchi

Quando i valori a disposizione sono sufficienti per il suo lavoro, il task applica l’algoritmo di Ahlstrom-Tompkins e calcola Y3(i-2) alla segnalazione del dato X(i). Sfruttando questo valore, verifica quindi se sono presenti le condizioni per un picco e, in caso affermativo, incrementa il relativo contatore. Tuttavia, essendo le soglie calcolate facendo riferimento al massimo valore di Y3 ottenuto con tutti i campioni, si necessita di un numero di campioni sufficiente ad approssimarle in modo adeguato, per poi procedere ad un progressivo aggiornamento. Tenendo in considerazione una frequenza di campionamento pari a 120 Hz e il fatto che in due secondi è generalmente compreso almeno un battito, si sfrutta un set di 250 valori per inizializzare le soglie. Una volta ottenute queste, esso si occupa di creare il task per il calcolo della frequenza cardiaca che non avrebbe avuto motivo di iniziare a lavorare prima.

Nonostante la struttura pseudoperiodica, il task è aperiodico. Questa scelta è dovuta al fatto che esso non ha una vera e propria deadline non essendoci vincoli temporali sul calcolo di y3.

Come visto nell’esempio riportato per il task di lettura, sono tollerate perdite nel calcolo di Y3, data l’elevata frequenza di campionamento.

Per quanto riguarda le risorse condivise, ritroviamo l’array dei dati mostrati a video a cui accede in lettura usando come riferimento l’indice per l’inserimento e prelevando i cinque valori precedenti, quantità necessaria all’algoritmo. La scelta di condividere questo indice, è dovuta alla preferenza dell’utilizzo dei valori più recenti rispetto a quello dei valori immediatamente sucessivi agli ultimi considerati. In questo modo si evita che qualche ritardo porti a ritrovarsi con dati non aggiornati, al solo costo di qualche sporadica perdita tollerabile.

Come già detto in precedenza, per sincronizzare questo task con quello di lettura si ricorre poi ad una variabile condition apposita.

Infine, si introduce una variabile contatore per i complessi QRS riscontrati condivisa con il task di calcolo della frequenza.

## Task di calcolo della frequenza cardiaca

Il task legge ogni due secondi il valore del contatore dei picchi e lo azzera. Una volta ottenute cinque letture consecutive, somma e moltiplica per sei le singole misurazioni al fine di effettuare una stima della frequenza cardiaca in battiti al minuto, aggiornandola successivamente ad ogni nuova lettura.