Biomedical Signal Processing

# ECG R-Peak Detection

梁　勝　富
成功大學 資訊工程系
sfliang@mail.ncku.edu.tw

Office: 資訊系館 12F 65C06, Tel: Ext. 62549
Lab:神經運算與腦機介面實驗室 (3F 65301)

2023

# ECG 量測範例

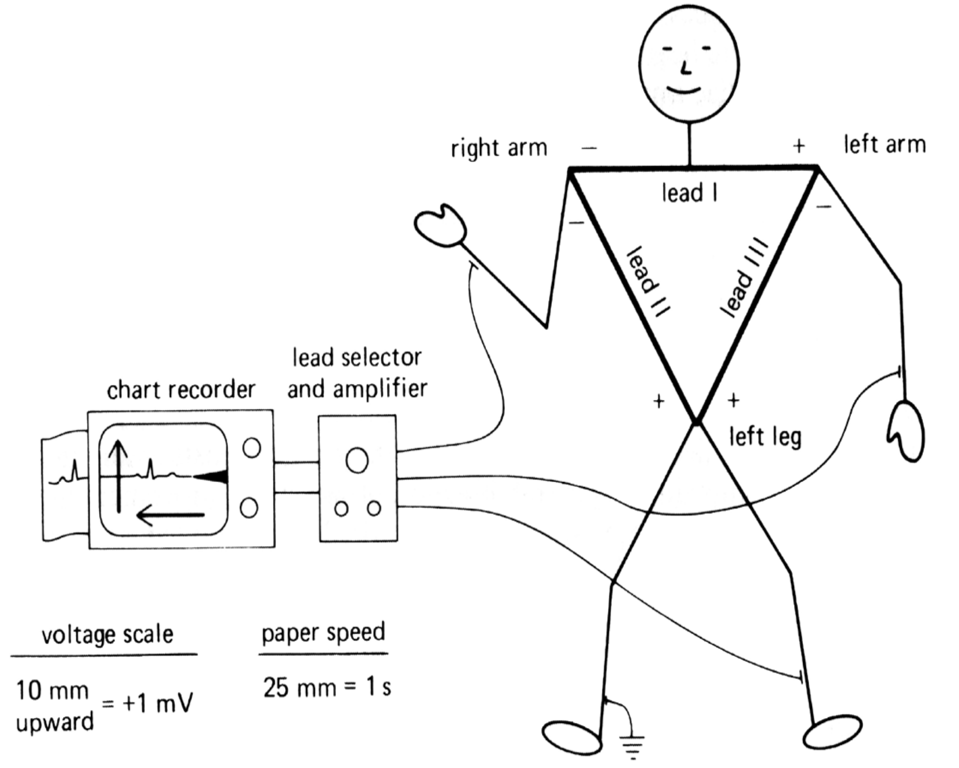


Figure 5-2 Einthoven's electrocardiographic conventions.

# ECG 量測範例

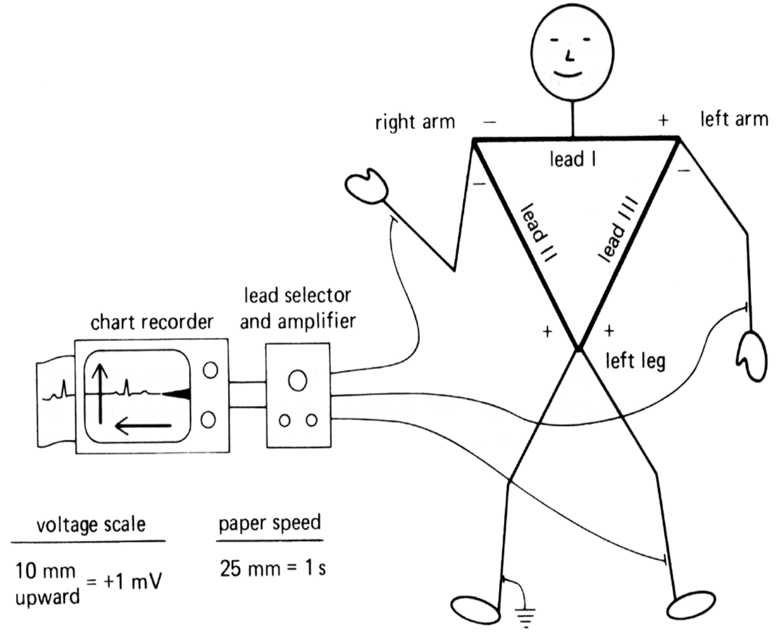
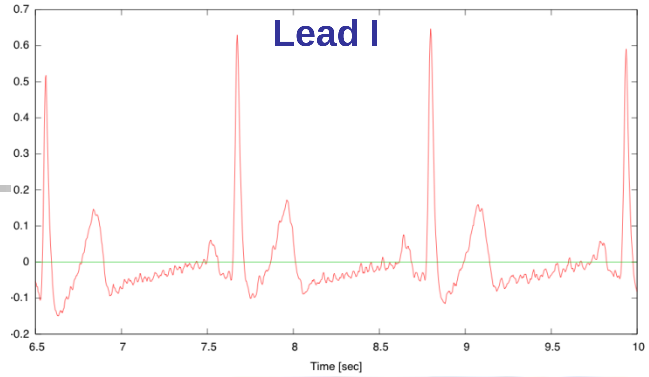
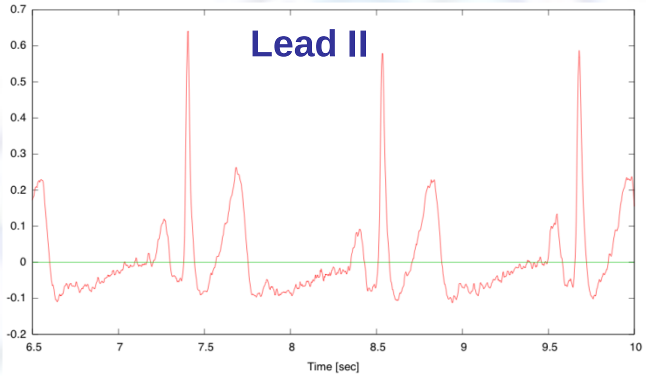
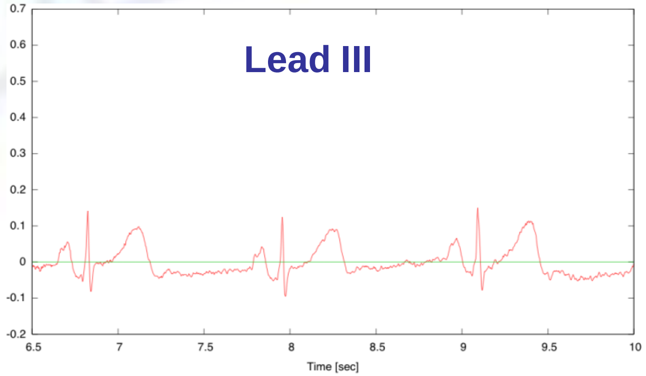


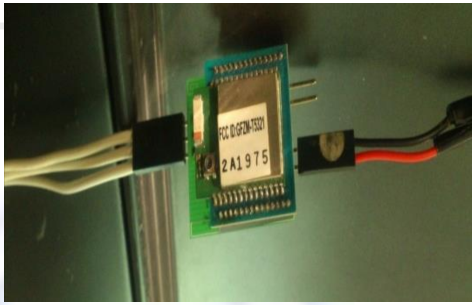
Figure 5-2 Einthoven's electrocardiographic conventions.





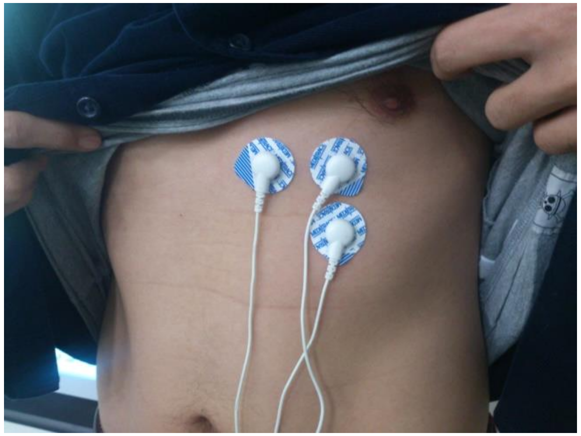


# The Recording System

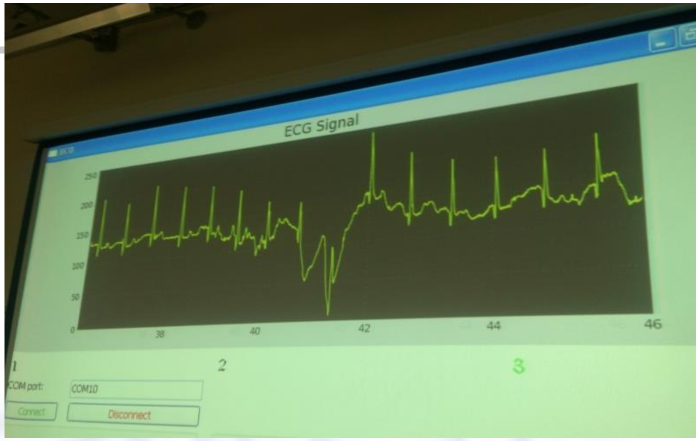


- Amplifier & Analog Filter
  - Gain: 1000 times
  - Cut-off frequency: 0.8 ~ 80 (Hz)
- ADC:
  - cc2530 ( 8051 base )
  - wireless transmission ( Zigbee )
- GUI & DSP:
  - Python implementation
  - Cross platform
  - Real-time system

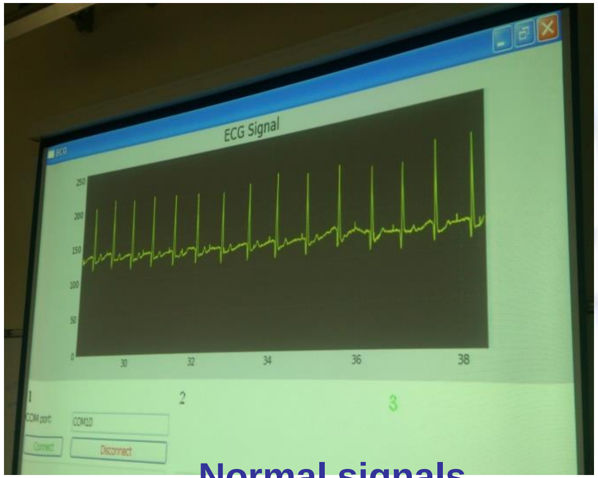
# Artifacts



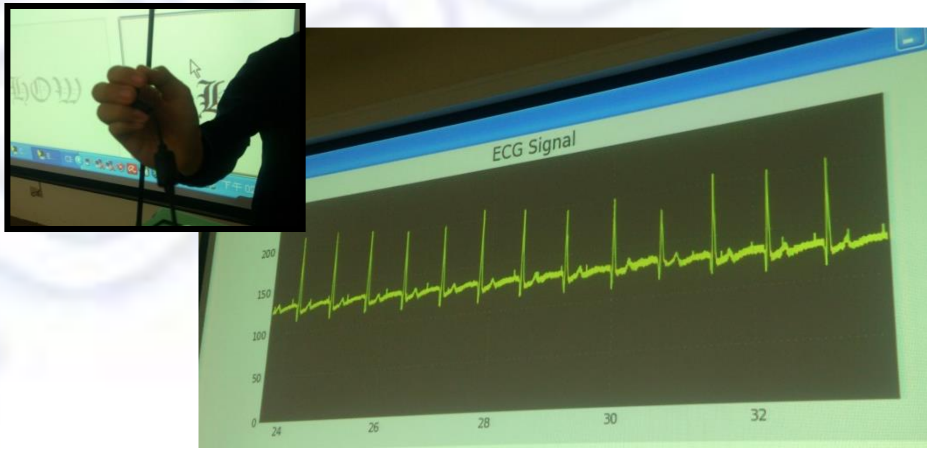
Electrode placement



Movement artifacts

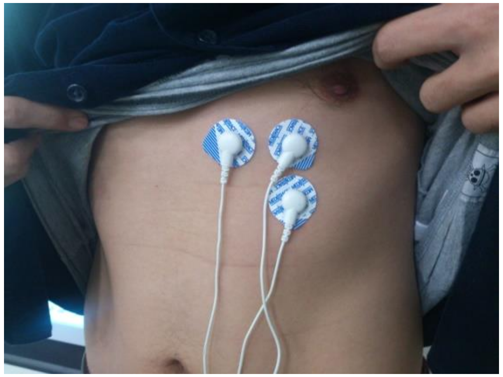


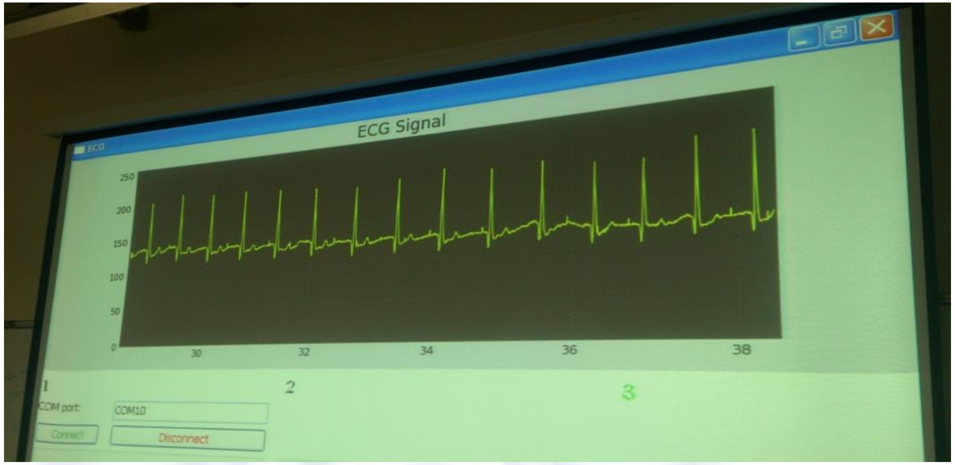
Normal signals

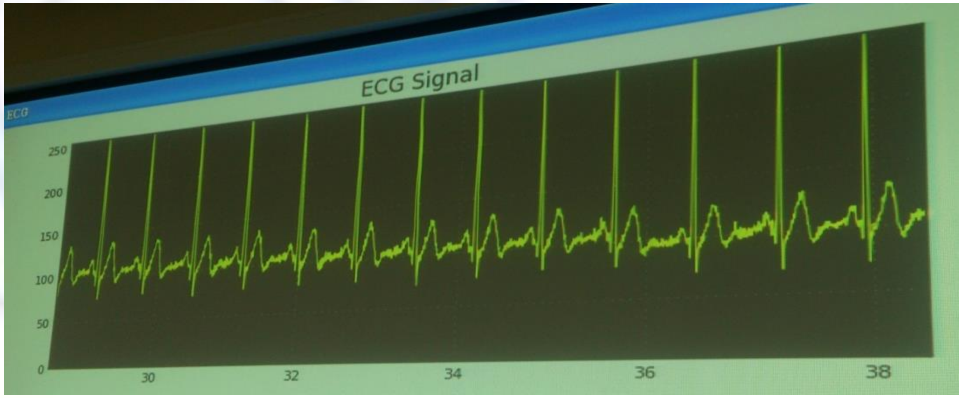


60Hz artifacts

# 電極擺放差異




ECG
ECG Signal
250
200
150
100
50
0
30
32
34
36
38
1
2
3
COM port:
COM10
Connect
Disconnect


ECG
ECG Signal
250
200
150
100
50
0
30
32
34
36
38

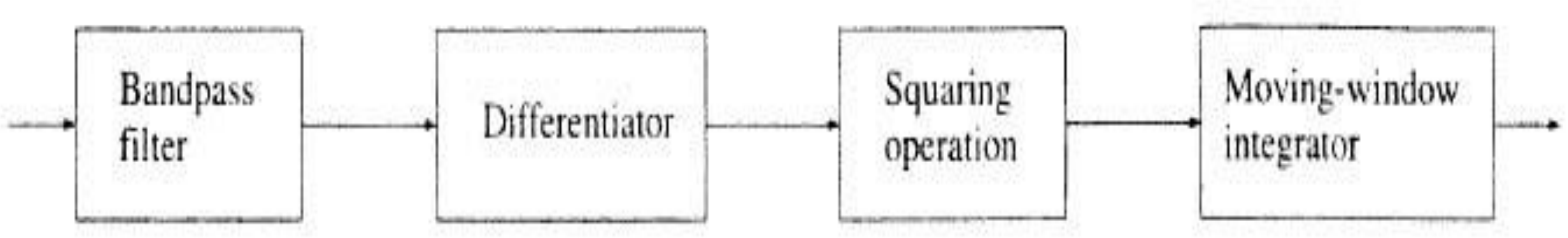
# The Pan-Tompkins Algorithm for QRS Detection

- An Algorithm to detect QRS complexes in an ongoing ECG signal.
- Ref
  - 4.3.2 of Biomedical Signal Analysis, Rangaraj M. Rangayyan, Wiley.
  - Pan, J. and Tompkins, W.J., “A real-time QRS detection algorithm,” IEEE Tans. On Biomedical Engineering, vol. 32, pp. 230-236, 1985.
  - Tompkins WJ, Biomedical Digital Signal Processing. Prentice-Hall, 1995.

# The Pan-Tompkins Algorithm for QRS Detection

- Pan-Tomplins algorithm is based on analysis of the slope, amplitude, and width of QRS complexes.
- It includes a series of
  - bandpass (lowpass+high pass)
  - derivative,
  - squaring,
  - integration,
  - adaptive thresholding & search procedures.

# The Pan-Tompkins Algorithm for QRS Detection



Bandpass filter → Differentiator → Squaring operation → Moving-window integrator

Figure 4.4 Block diagram of the Pan-Tompkins algorithm for QRS detection.

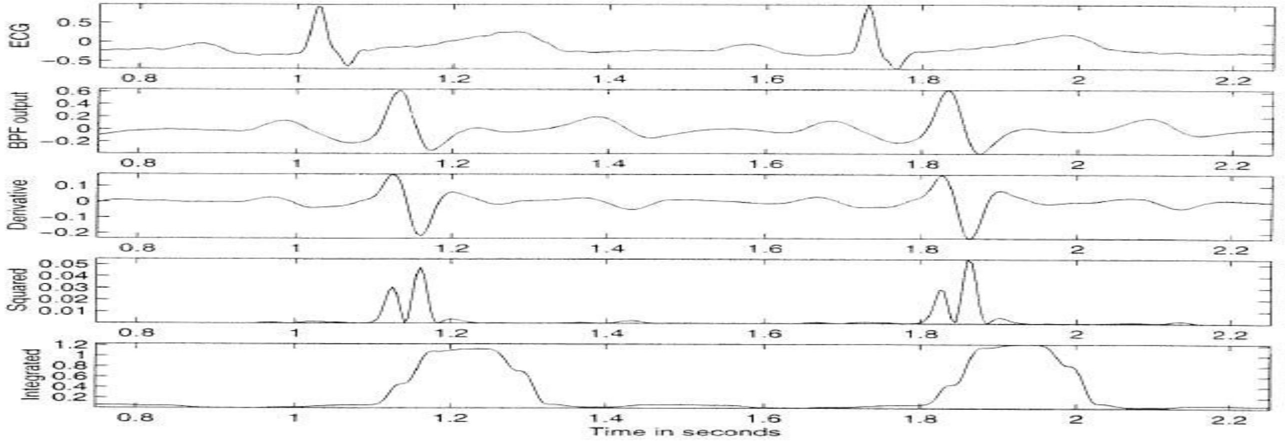


**Figure 4.6** Results of the Pan-Tompkins algorithm. From top to bottom: two cycles of a filtered version of the ECG signal shown in Figure 3.5 (the same as that in Figure 4.2): output of the bandpass filter (BPF, a combination of lowpass and highpass filters): output of the derivative-based operator; the result of squaring; and 100× the result of the final integrator.

# Lowpass Filter

- Integer coefficients were used to reduce computational complexity.

$$H(z) = \frac{1}{32}\frac{(1-z^{-6})^2}{(1-z^{-1})^2}.$$

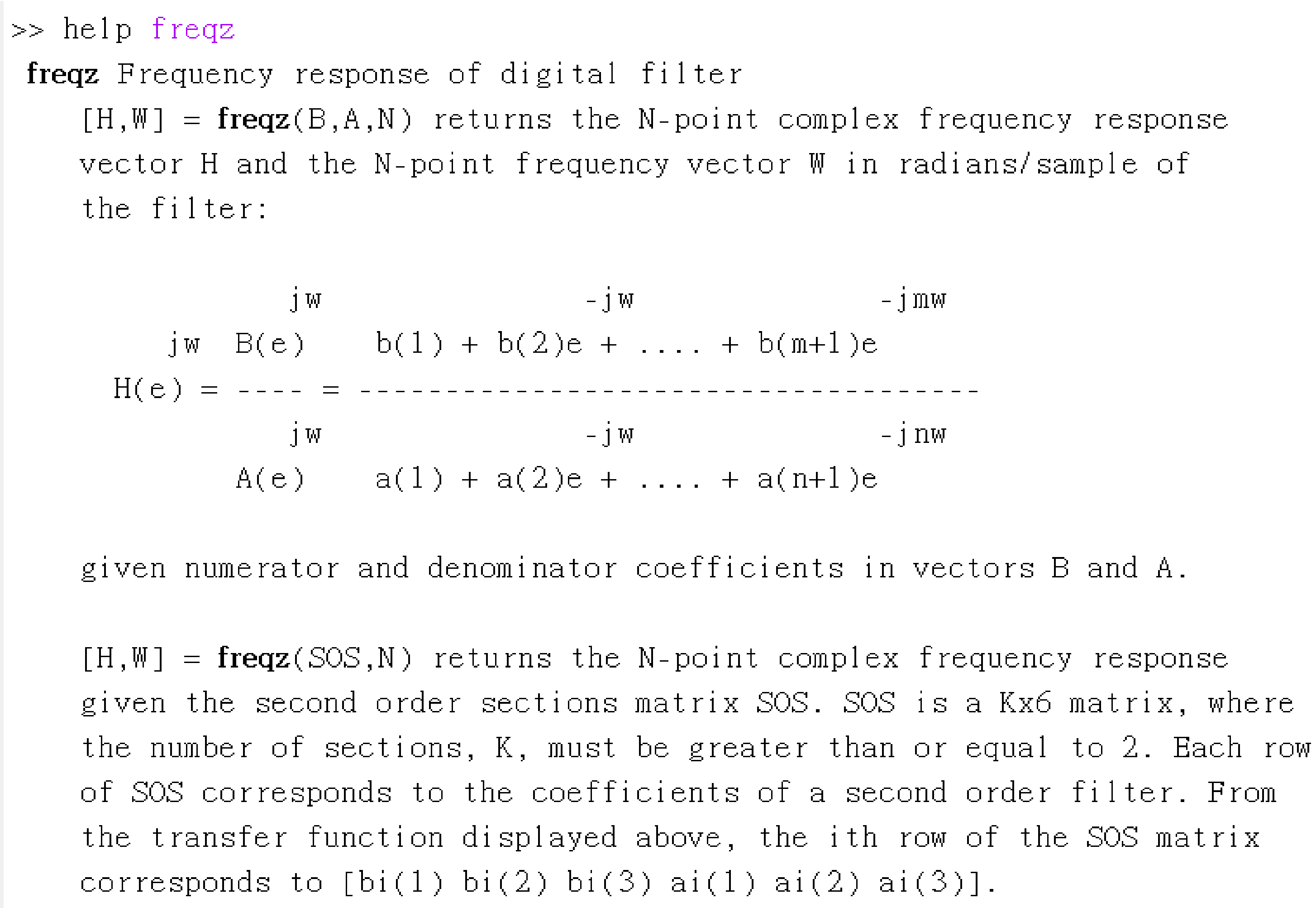
- The output y(n) related to the input x(n) is

$$y(n) = 2y(n-1) - y(n-2) + \frac{1}{32}[x(n) - 2x(n-6) + x(n-12)].$$

# Lowpass Filter

- It introduces a delay of 5 samples or 25 ms with the sampling rate being 200 Hz.
- The cutoff frequency is 11 Hz and it provides an attenuation greater than 35 dB at 60 Hz.
- It effectively suppresses power-line interference, if present.

# Frequency Response of Digital Filter

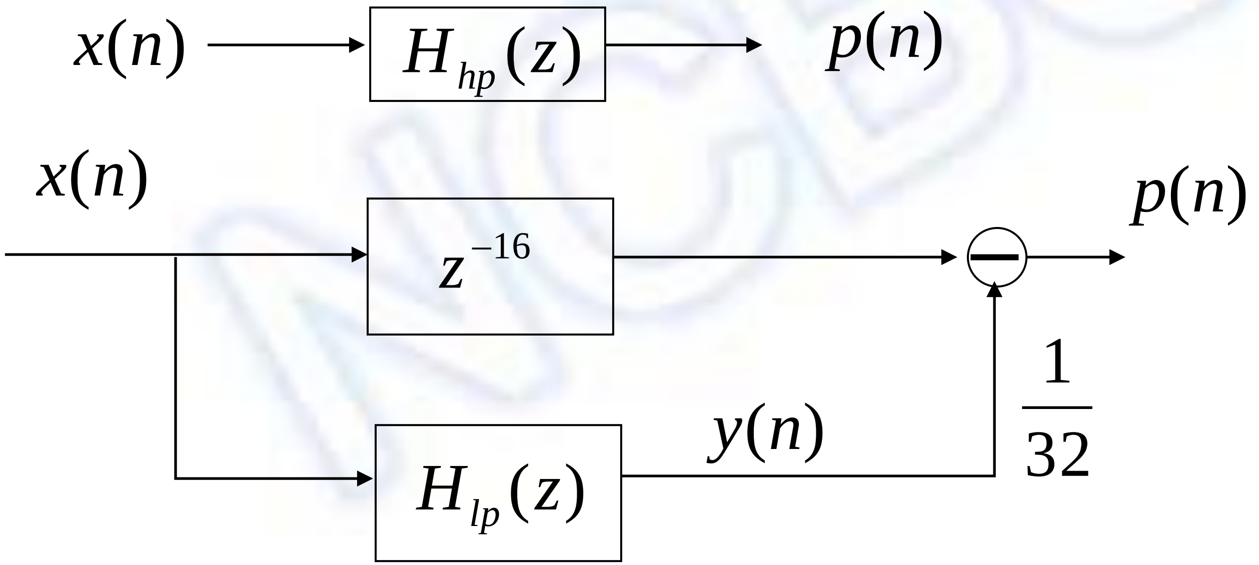


```
>> help freqz
 freqz Frequency response of digital filter
    [H,W] = freqz(B,A,N) returns the N-point complex frequency response
    vector H and the N-point frequency vector W in radians/sample of
    the filter:

                jw               -jw              -jmw
         jw  B(e)    b(1) + b(2)e + .... + b(m+1)e
      H(e) = ---- = ------------------------------------
                jw               -jw              -jnw
             A(e)    a(1) + a(2)e + .... + a(n+1)e

    given numerator and denominator coefficients in vectors B and A.

    [H,W] = freqz(SOS,N) returns the N-point complex frequency response
    given the second order sections matrix SOS. SOS is a Kx6 matrix, where
    the number of sections, K, must be greater than or equal to 2. Each row
    of SOS corresponds to the coefficients of a second order filter. From
    the transfer function displayed above, the ith row of the SOS matrix
    corresponds to [bi(1) bi(2) bi(3) ai(1) ai(2) ai(3)].
```

# Highpass Filter

- The highpass is implemented as an allpass filter minus a lowpass filter.

$$H_{hp}(z) = z^{-16} - \frac{1}{32} H_{lp}(z)$$



# Highpass Filter

- The lowpass $H_{lp}(z)$ is

$$H_{lp}(z) = \frac{(1 - z^{-32})}{(1 - z^{-1})};$$

- The input-output relationship of $H_{lp}(z)$ is

$$y(n) = y(n-1) + x(n) - x(n-32).$$

# Highpass Filter

$$\frac{P(z)}{X(z)} = H_{hp}(z) = z^{-16} - \frac{1}{32}H_{lp}(z) = z^{-16} - \frac{1}{32}\frac{Y(z)}{X(z)}$$

$$P(z) = X(z)z^{-16} - \frac{1}{32}Y(z)$$

- The output p(n) of the highpass is

$$p(n) = x(n-16) - \frac{1}{32}\left[y(n-1) + x(n) - x(n-32)\right],$$

- The highpass filter has a cutoff frequency of 5 Hz and introduces a delay of 80 ms.

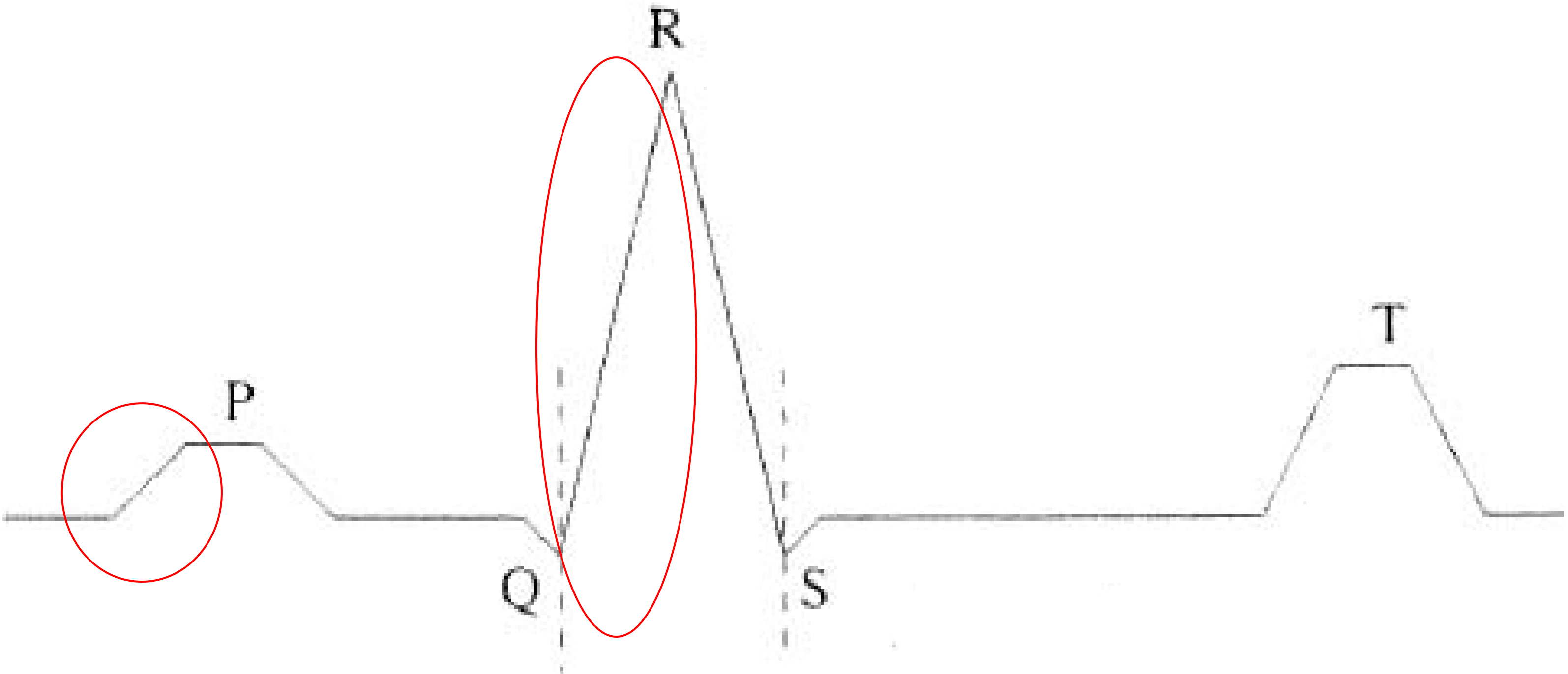
# Derivative Operator

- The derivative operation is

$$y(n) = \frac{1}{8}\left[2x(n) + x(n-1) - x(n-3) - 2x(n-4)\right],$$

- It approximates the ideal d/dt up to 30Hz.
- The derivative procedure suppresses the low-frequency components of P and T.
- It provides a large gain to the high-frequency components arising from the high slopes of the QRS.

# Derivative Operator



# Squaring

- The squaring operation

$$y(n) = x^2(n)$$

will

  - make the result positive
  - emphasize large differences from QRS complexes
  - suppress small difference arising from P and T.
  - The high-frequency components related to the QRS are further enhanced.

# Integration

- A derivative based operation will exhibit multiple peaks in the duration of a single QRS complex.
- A moving-window integration is performed to smooth the output of the preceding operations in Pan-Tompkins algorithm.

# Integration

- The moving-window integration filter

$$y(n) = \frac{1}{N}\left[x(n-(N-1)) + x(n-(N-2)) + \cdots + x(n)\right].$$

is used to smooth multiple peaks in a single QRS complex.

  - large window will result large values since QRS and T are merged.
  - small window  till yield several peaks for a single QRS.
  - N=30 is suitable for *fs*=200 Hz.

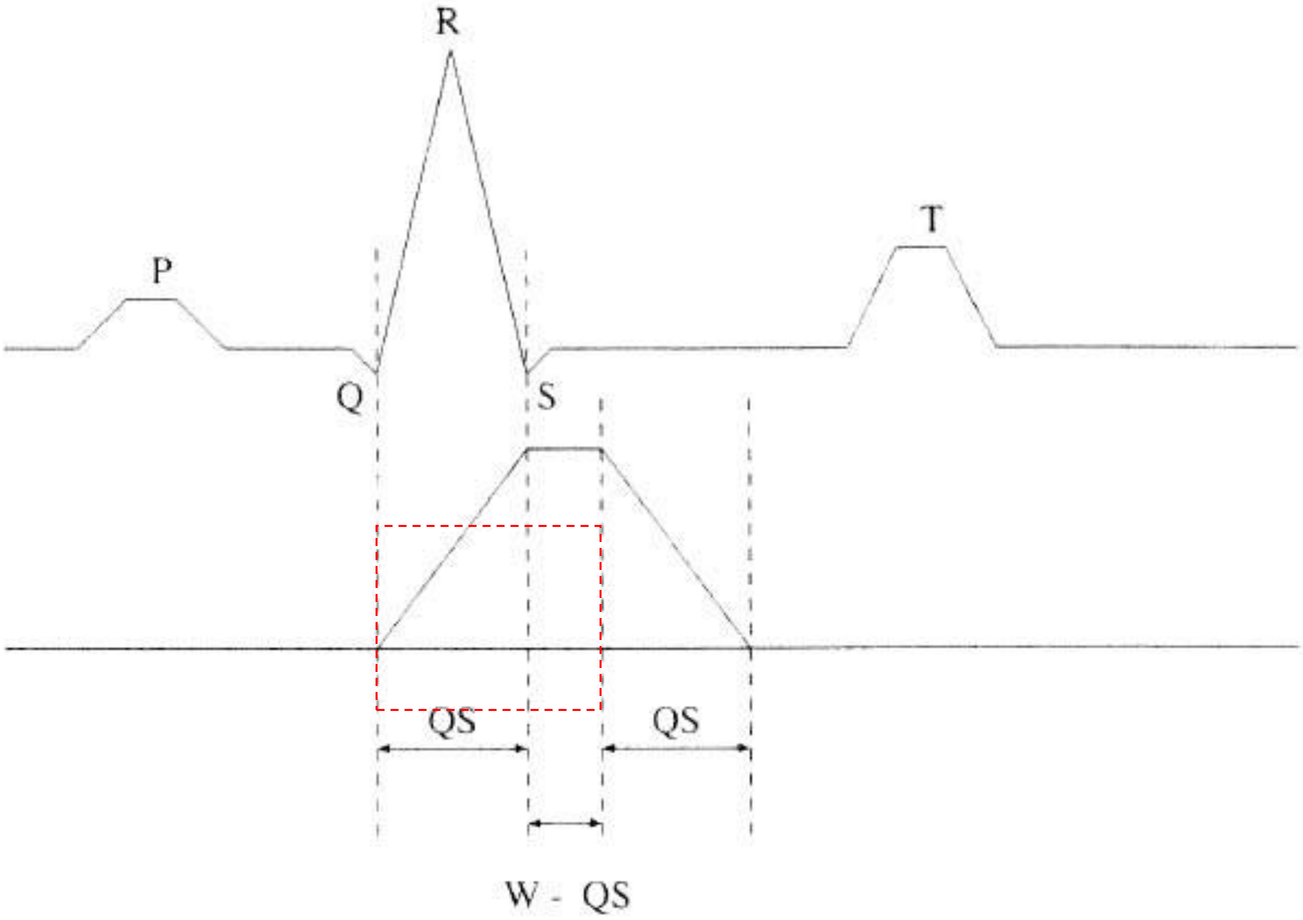


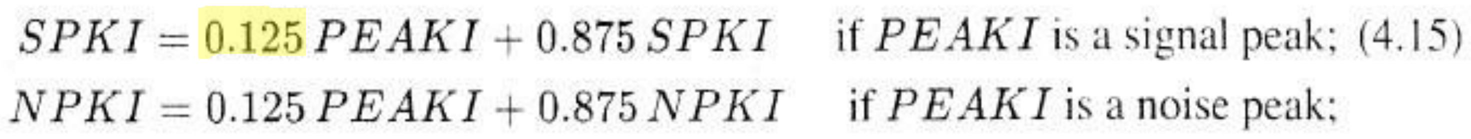
**Figure 4.5** The relationship of a QRS complex to the moving-window integrator output. Upper plot: Schematic ECG signal. Lower plot: Output of the moving-window integrator. QS: QRS complex width. W: width of the integrator window, given as $N/f_s$ $s$. Adapted from Tompkins [27].

# Adaptive Thresholding

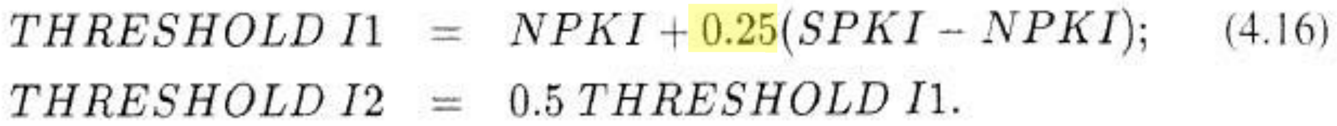
- If a peak exceeds a threshold called THRESHOLD I1, it is classified as a QRS peak.
- THRESHOLD I1 should be updated according the QRS peaks and noise peaks.
- How to do it?

# Adaptive Thresholding

- Definition
  - PEAKI: the detected new peak
  - SPKI: peak level of QRS peaks.
  - NPKI: peak level of non-QRS peaks.



$$SPKI = 0.125\,PEAKI + 0.875\,SPKI \quad \text{if } PEAKI \text{ is a signal peak;} \quad (4.15)$$
$$NPKI = 0.125\,PEAKI + 0.875\,NPKI \quad \text{if } PEAKI \text{ is a noise peak;}$$



$$THRESHOLD\ I1 = NPKI + 0.25(SPKI - NPKI); \quad (4.16)$$
$$THRESHOLD\ I2 = 0.5\,THRESHOLD\ I1.$$

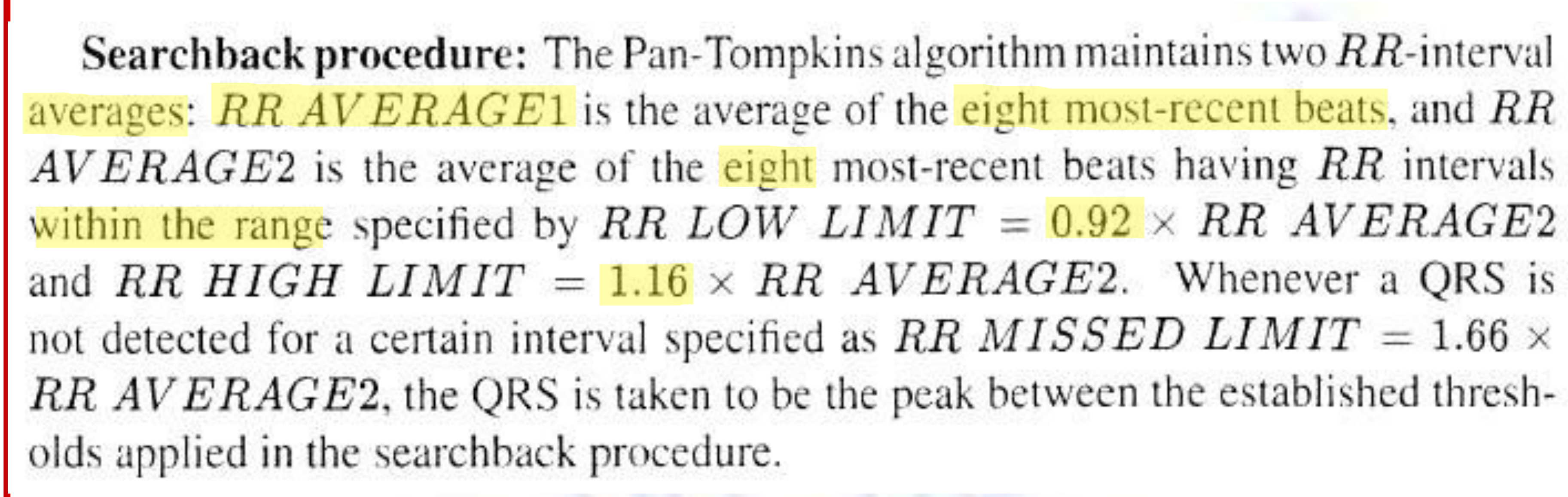
# Searchback Procedure

- Some QRS may be missed by the hard-thresholding forward searching.
- Ex. THRESHOLD I1 = 0.5, a peak is 0.48, is it a noise peak or a QRS?
- A searchback procedure is used to reconfirm the peaks.
- The averaged RR-duration is used to detect if it misses a QRS for a long period (ex. 1.66 RR-duration)

# Searchback Procedure

- If no QRS is detected for a predefined period, (ex. 1.66 averaged RR-duration), we may miss a QRS.
- Searchback is used to detect the peak by a lower threshold called THRESHOLD I2 that is 0.5 THRESHOLD I1.

# Searchback Procedure



**Searchback procedure:** The Pan-Tompkins algorithm maintains two $RR$-interval averages: $RR\ AVERAGE1$ is the average of the eight most-recent beats, and $RR\ AVERAGE2$ is the average of the eight most-recent beats having $RR$ intervals within the range specified by $RR\ LOW\ LIMIT = 0.92 \times RR\ AVERAGE2$ and $RR\ HIGH\ LIMIT = 1.16 \times RR\ AVERAGE2$. Whenever a QRS is not detected for a certain interval specified as $RR\ MISSED\ LIMIT = 1.66 \times RR\ AVERAGE2$, the QRS is taken to be the peak between the established thresholds applied in the searchback procedure.

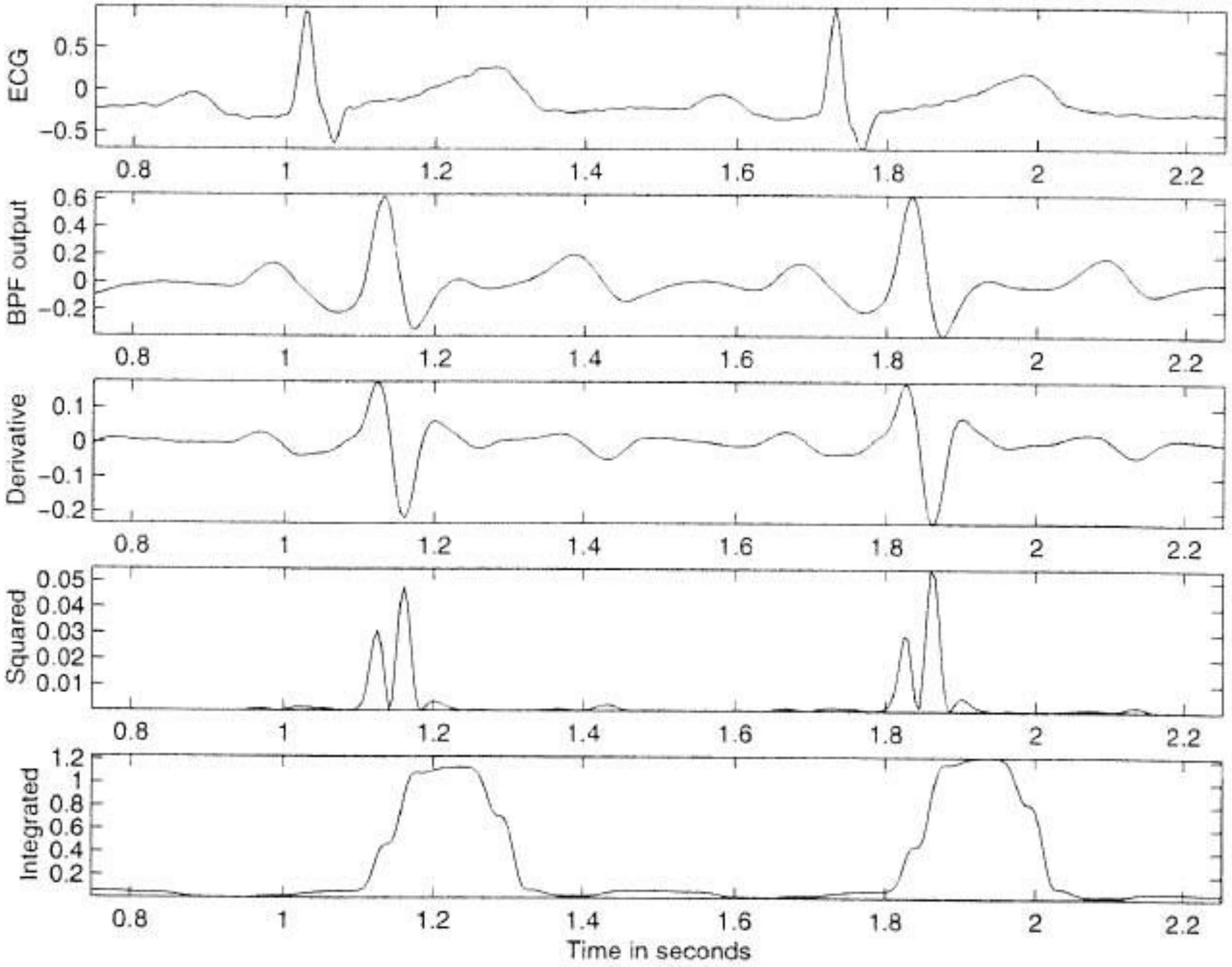


**Figure 4.6** Results of the Pan-Tompkins algorithm. From top to bottom: two cycles of a filtered version of the ECG signal shown in Figure 3.5 (the same as that in Figure 4.2); output of the bandpass filter (BPF, a combination of lowpass and highpass filters); output of the derivative-based operator; the result of squaring; and 100× the result of the final integrator.

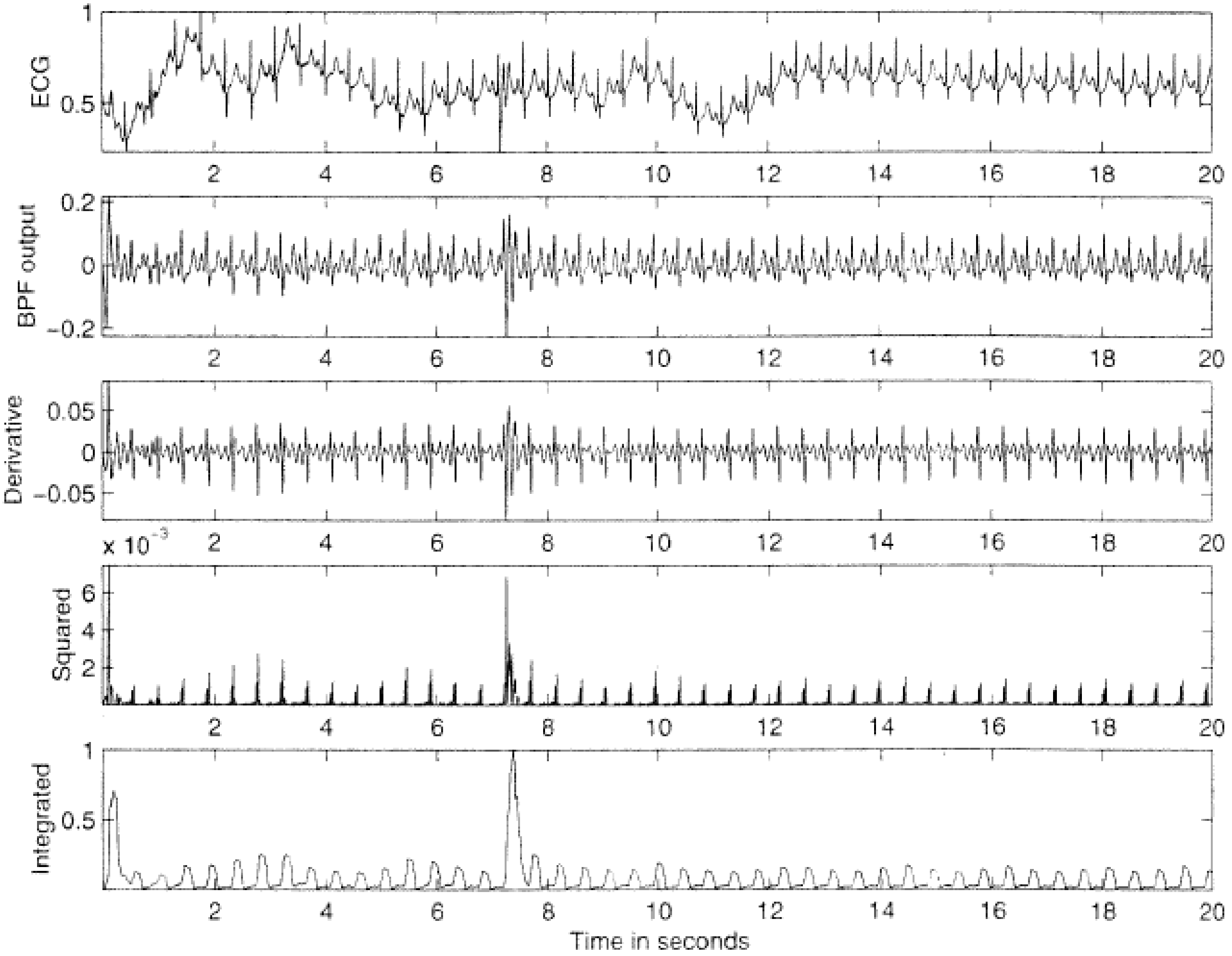


**Figure 4.26** Results of the Pan-Tompkins algorithm with a noisy ECG signal. From top to bottom: ECG signal sampled at 200 $Hz$; output of the bandpass filter (BPF); output of the derivative-based operator; the result of squaring; and normalized result of the final integrator.

Further study: Ch 4.9 of Biomedical Signal Analysis, (pp. 222-225)