

2. Blink Generation

Blink activity was simulated by first generating a single positive sinusoidal pulse, $y = \sin(x)$, $0^\circ \leq x \leq 180^\circ$. Then, a single time course of blink activations was constructed by concatenating together this sequence of positive sinusoidal pulses. Pulse duration and inter-pulse spacing were both specified by the user, and could be set to either a fixed or varying integer number of data sample points, where each sample point corresponded to a 0.004 second time interval. The height of each pulse could also be set to a fixed or varying quantity to simulate blinks of different intensities. By varying these parameters, in part or in whole, we created different sets of blink - contaminated EEG data.

This time course of blink activations, represented in MATLAB by a row vector of size $1 \times \text{NumberSamples}$, then right multiplied the blink template, a $\text{NumberChannels} \times 1$ column vector, thereby projecting the blink activations onto the detectors. The result was a matrix of size $\text{NumberChannels} \times \text{NumberSamples}$, in which each row represents the contribution of the time course of blink activations at its corresponding detector, and each column represents one instant of blink activity at all the detectors. This matrix was then added to the array of "blink-free" EEG data, essentially superimposing the totality of blink activity onto the data to construct the array of blink contaminated EEG data.

$$X_{\text{Blinky}} = X_{\text{BlinkFree}} + \text{Template}_{\text{Noise}} * \text{BlinkActivation}$$

Discussion of Results

The overriding objective in creating these data sets was to develop a suite of data with which we could test the ability of different ICA algorithms to remove blinks from EEG recordings. To this end, we first constructed the "blink-free" EEG data set, rich in cortical activity yet free of blink contamination. This was the baseline or gold-standard to which filtered EEG, after blink removal, would be subsequently compared. We then constructed seven distinct sets of blinks, which were then added to the "blink-free" EEG, resulting in seven sets of "blinky" EEG, each with a unique blink morphology. Now, in subsequent studies, some or all of the following comparisons can be made:

- Compare filtered data to the "blink-free" data;
- Compare extracted blinks to the simulated, inserted blinks;
- Compare corresponding projections of extracted blink activations to the blink template.

Since we know not only the structure of the simulated blinks but also at which sample points they were inserted, detailed future comparisons can be made employing a variety of metrics.

Synthesized Blink Data Specifications

Clean, Blink-Free, Data Parameters

Channels (Rows): 34
Samples (Columns): 59419
Minimum Data Value: - 87.43 / -113.94 (Band-Pass Filtered / Not Band-Pass Filtered)
Maximum Data Value: +75.46 / +110.90 (Band-Pass Filtered / Not Band-Pass Filtered)

Blink Template Parameters

Channels (Rows): 34
Samples (Columns): 1
Minimum Data Value: - 44.95 (Channel #: 34)
Maximum Data Value: +148.07 (Channel #: 06)

'Average' Blink Characteristics

Blink Duration @ eyeball (millisecond): 65
Blink Frequency @ eyeball (millisecond⁻¹): 0.0002
Blink Duration @ EEG detector (samples): 100
Blink Duration @ EEG detector (millisecond): 400
Inter-Blink Spacing @ EEG detector (samples): 1250
Inter-Blink Spacing @ EEG detector (millisecond): 5000

$$\text{Relative Blink Strength} = \frac{\text{Max | Blink Intensity |} - \text{Max | Clean Data Intensity |}}{\text{Max | Clean Data Intensity |}}$$

*Note: "Blinky" Data = Clean, Blink-Free, Data + Blink Template * Sinusoidal Pulses*

"Blinky" Data Set # 1:

Relative Blink Strength: 25% (MATLAB BSF: 0.93625)
Constant Blink Spacing
Constant Blink Duration

"Blinky" Data Set # 2:

Relative Blink Strength: 50% (MATLAB BSF: 1.12350)
Constant Blink Spacing
Constant Blink Duration

"Blinky" Data Set # 3:

Relative Blink Strength: 100% (MATLAB BSF: 1.49800)
Constant Blink Spacing
Constant Blink Duration

"Blinky" Data Set # 4:

Relative Blink Strength: 200% (MATLAB BSF: 2.24700)
Constant Blink Spacing
Constant Blink Duration

"Blinky" Data Set # 5:

Relative Blink Strength: 400% (MATLAB BSF: 2.75208 / 3.74500)
Constant Blink Spacing
Constant Blink Duration

"Blinky" Data Set # 6:

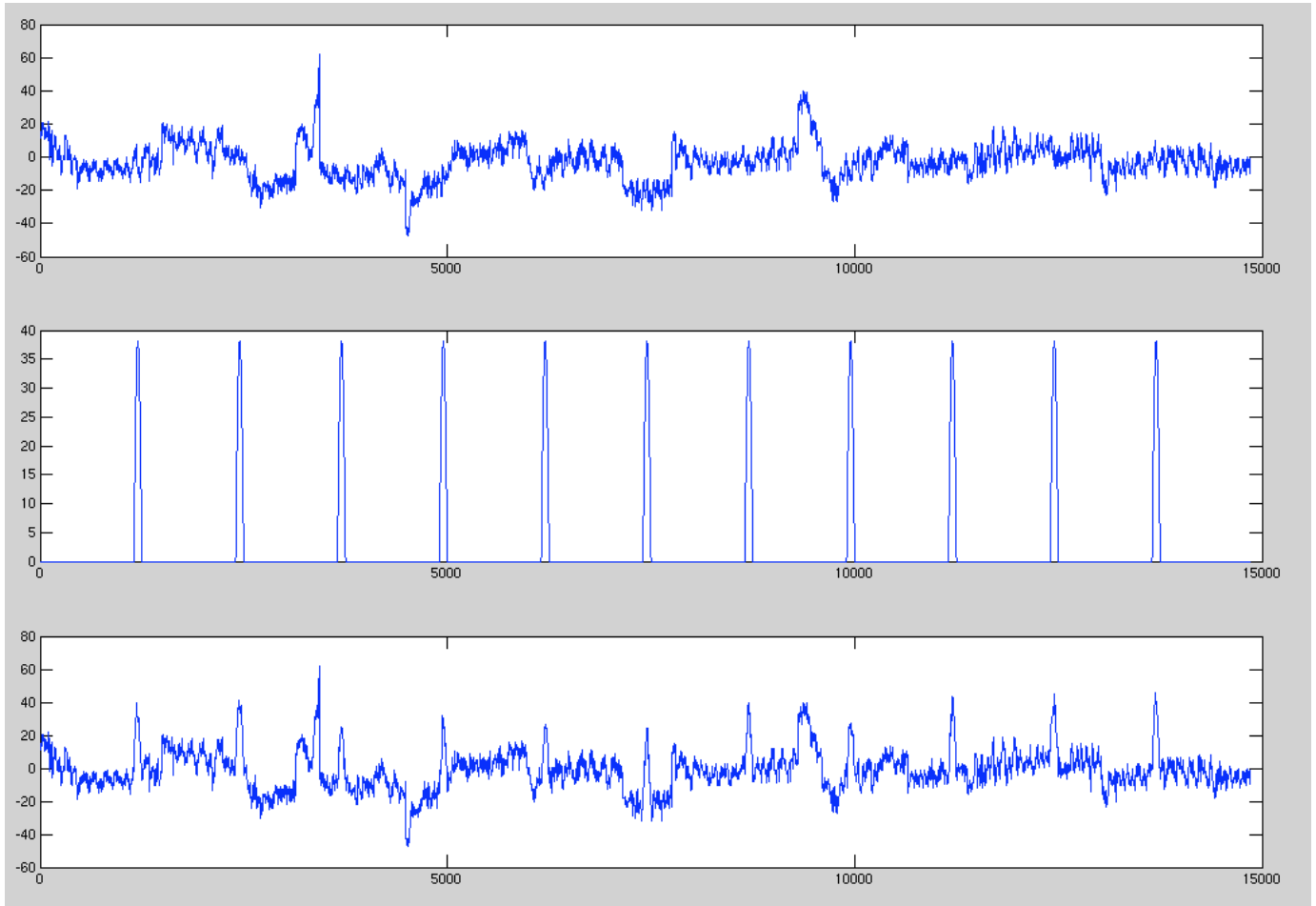
Relative Blink Strength: 50% - 200% (MATLAB BSF: 1.12350 - 2.24700)
Variable Blink Spacing (samples): 625 - 2500
Variable Blink Duration (samples): 50 - 200

"Blinky" Data Set # 7:

Relative Blink Strength: 25% - 400% (MATLAB BSF: 0.93625 - 3.74500)
Variable Blink Spacing (samples): 312 - 5000
Variable Blink Duration (samples): 25 - 400

"Blinky" Data Set # 1

Channel # 1: First 15,000 Data Samples

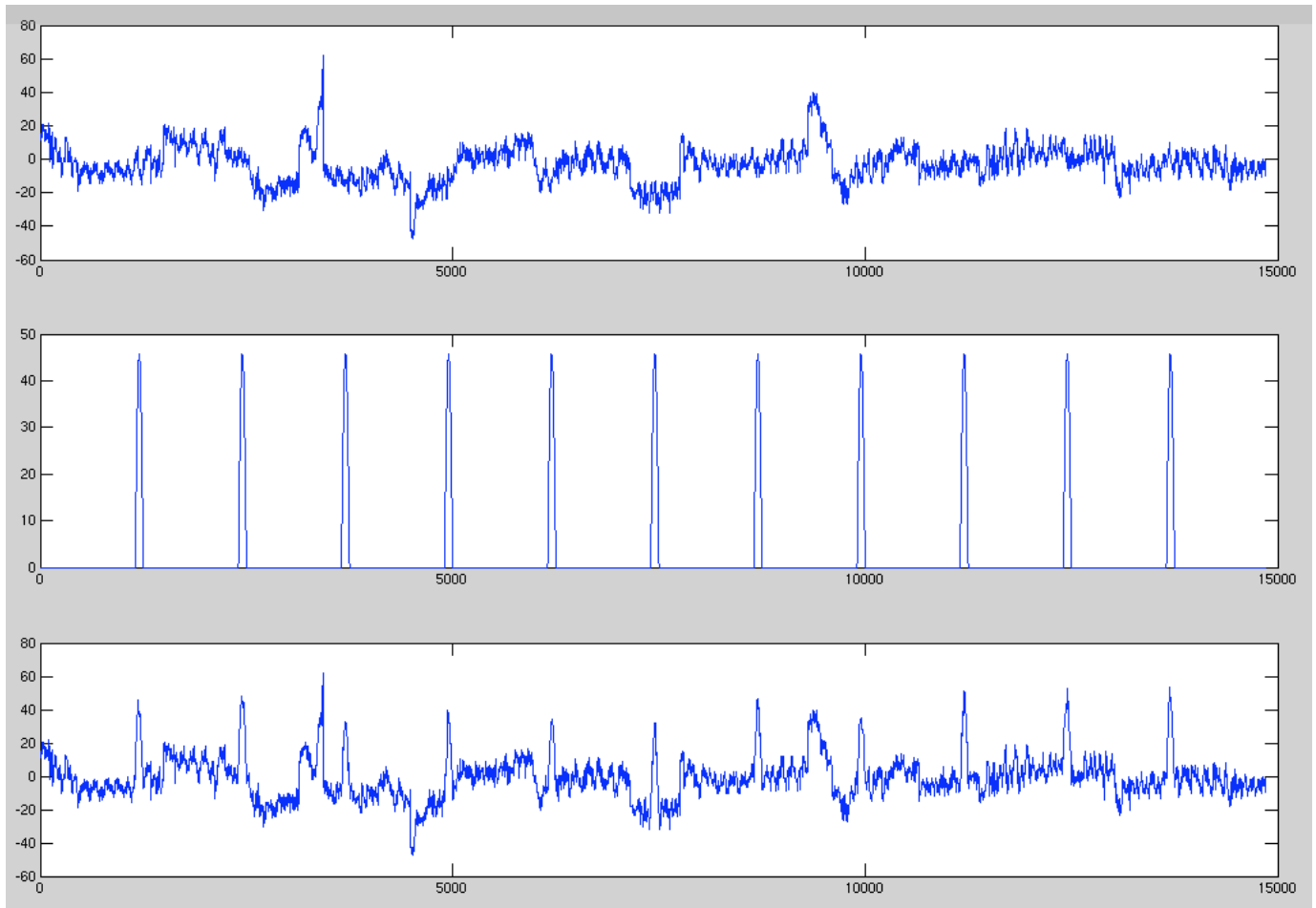


From Top to Bottom:

- Clean, Blink-Free, Data
- Blink Stream
- Clean, Blink-Free, Data + Blink Template * Blink Stream

"Blinky" Data Set # 2

Channel # 1: First 15,000 Data Samples

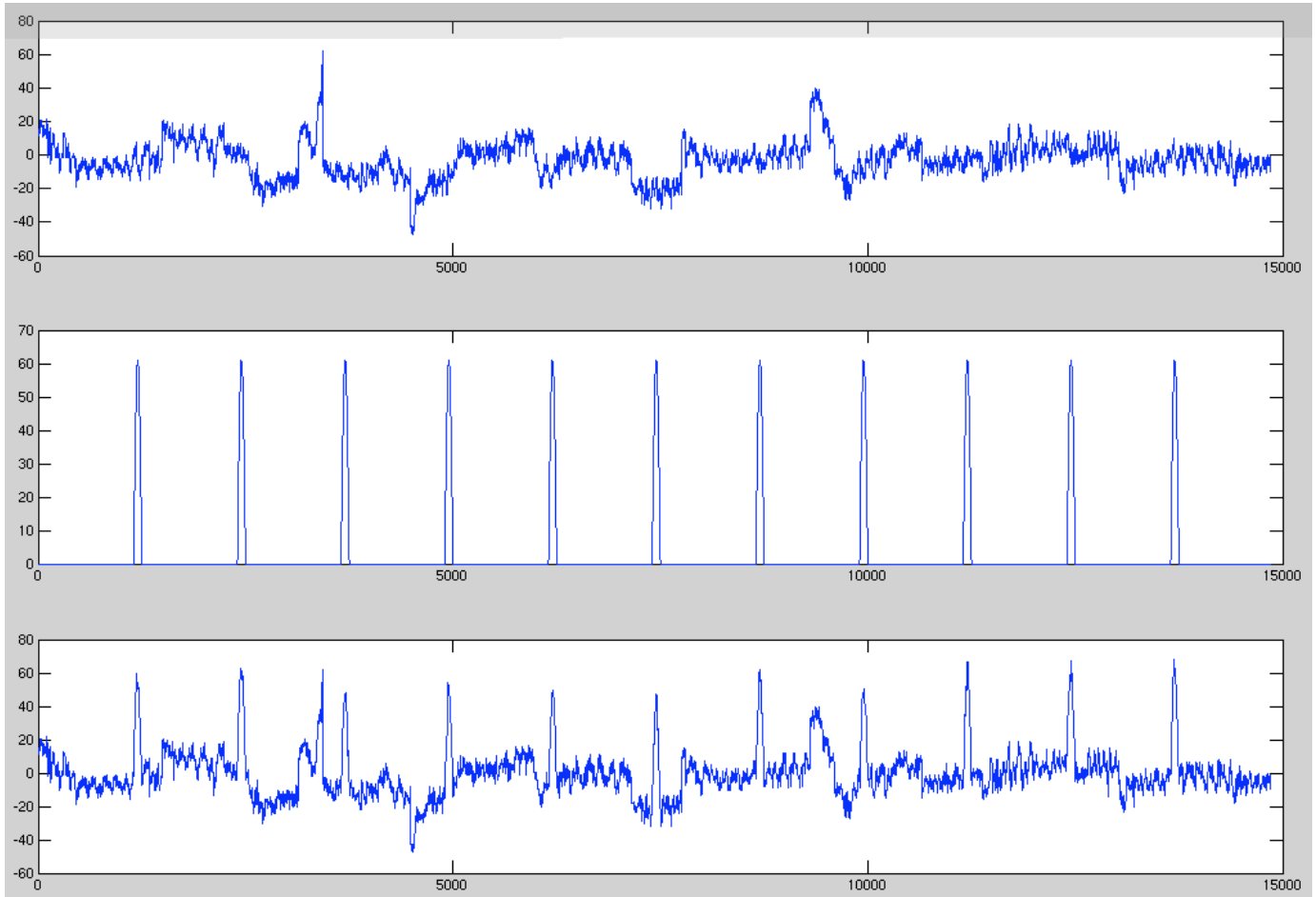


From Top to Bottom:

- Clean, Blink-Free, Data
- Blink Stream
- Clean, Blink-Free, Data + Blink Template * Blink Stream

"Blinky" Data Set # 3

Channel # 1: First 15,000 Data Samples

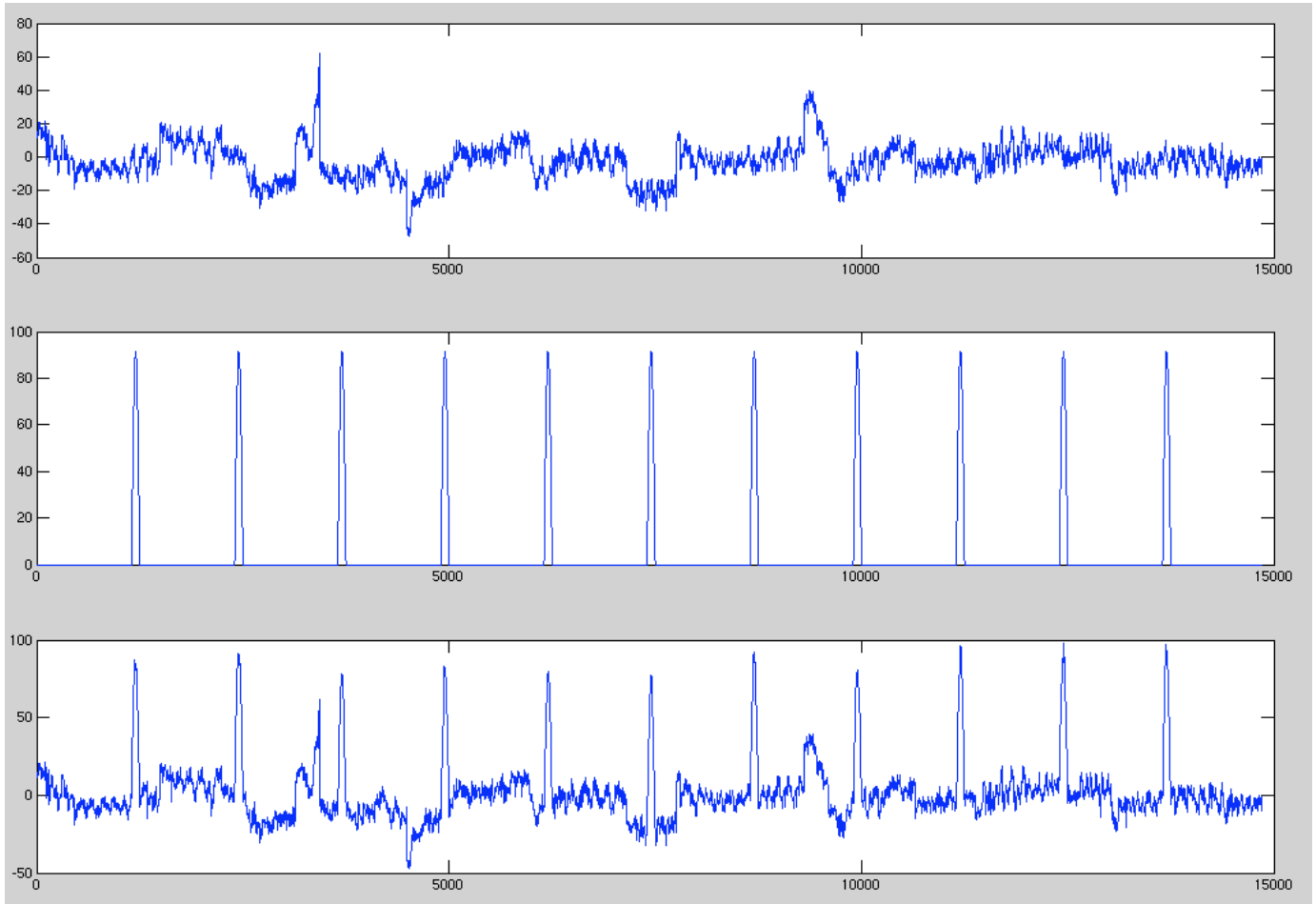


From Top to Bottom:

- Clean, Blink-Free, Data
- Blink Stream
- Clean, Blink-Free, Data + Blink Template * Blink Stream

"Blinky" Data Set # 4

Channel # 1: First 15,000 Data Samples

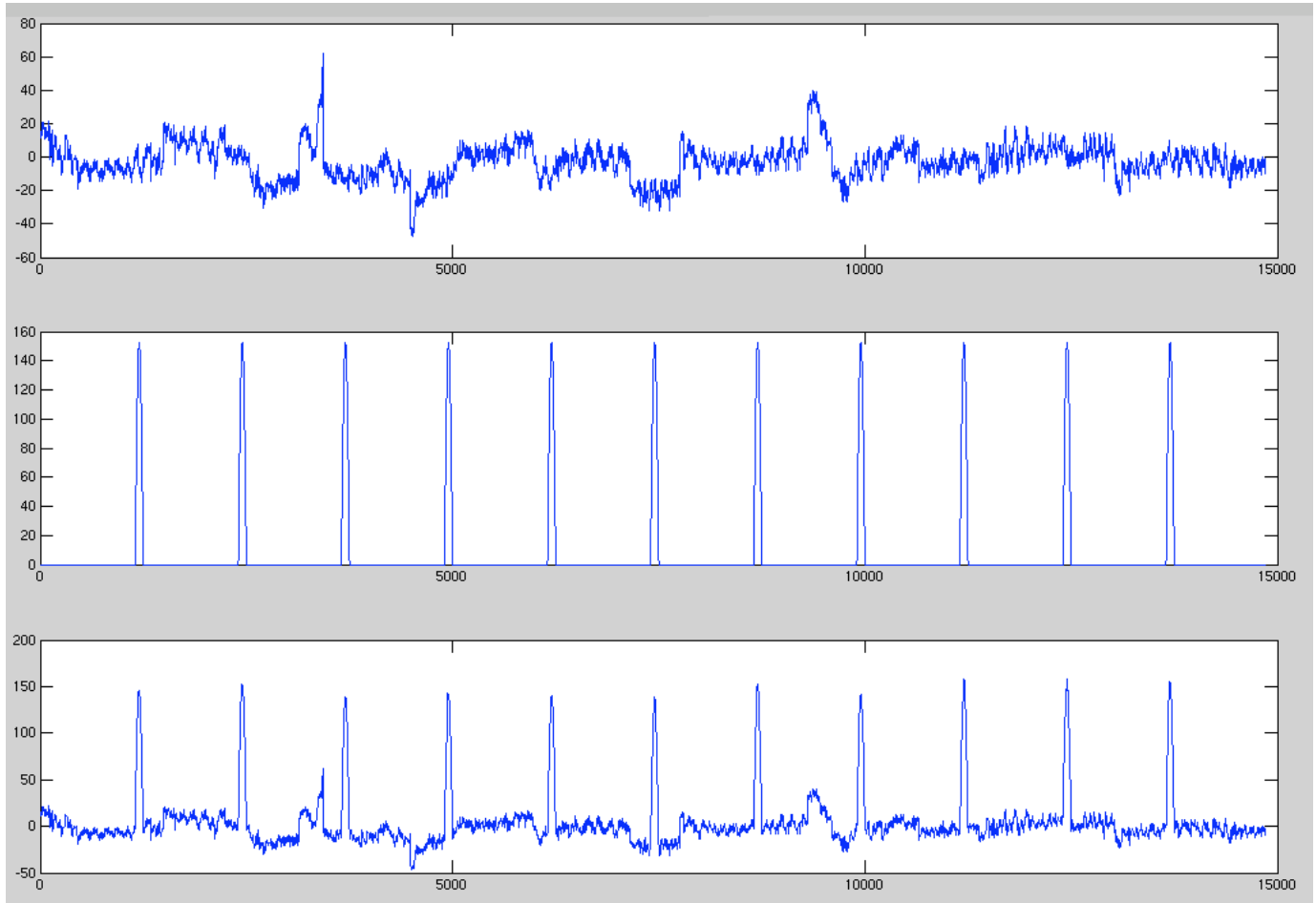


From Top to Bottom:

- Clean, Blink-Free, Data
- Blink Stream
- Clean, Blink-Free, Data + Blink Template * Blink Stream

"Blinky" Data Set # 5

Channel # 1: First 15,000 Data Samples

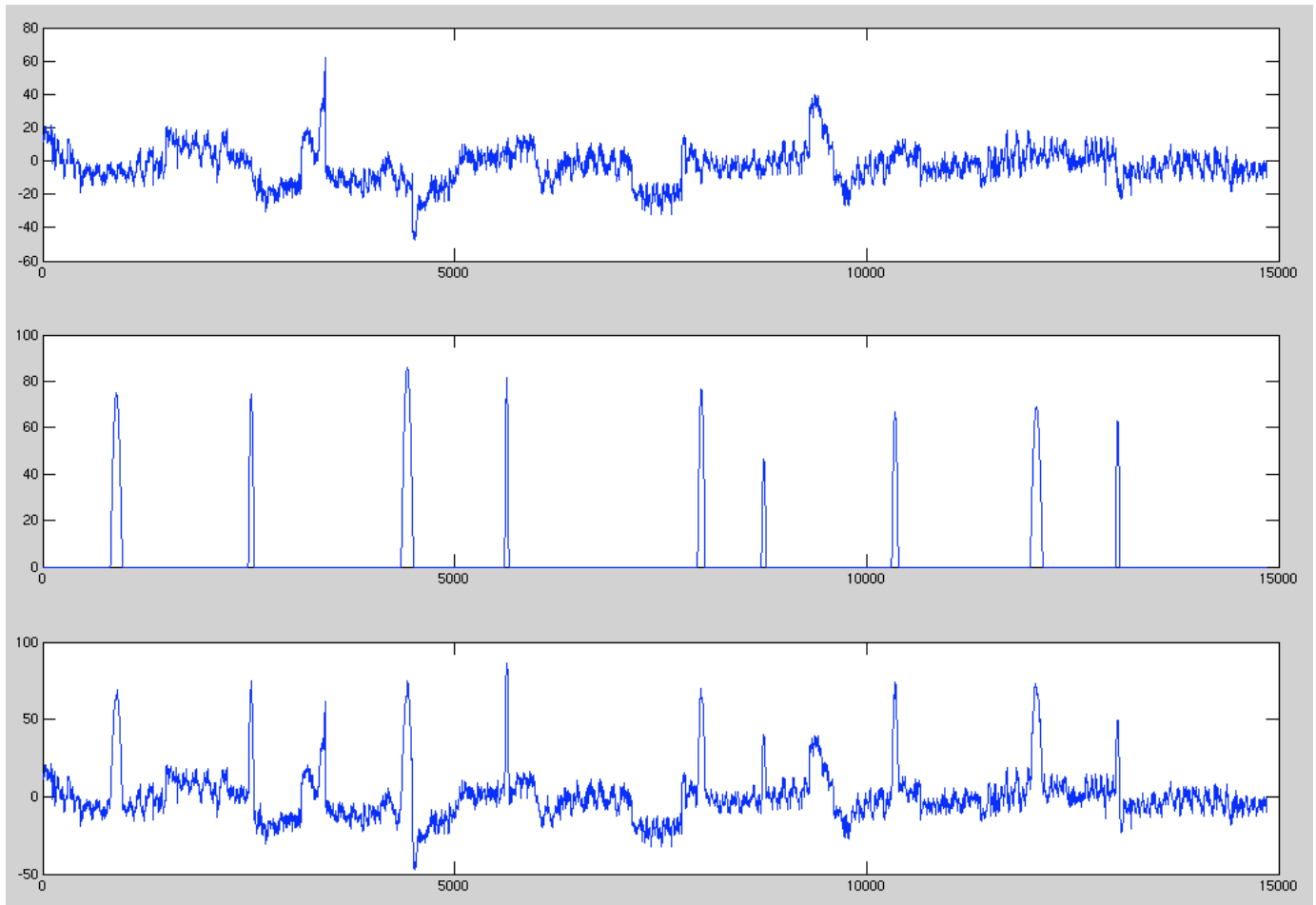


From Top to Bottom:

- Clean, Blink-Free, Data
- Blink Stream
- Clean, Blink-Free, Data + Blink Template * Blink Stream

"Blinky" Data Set # 6

Channel # 1: First 15,000 Data Samples

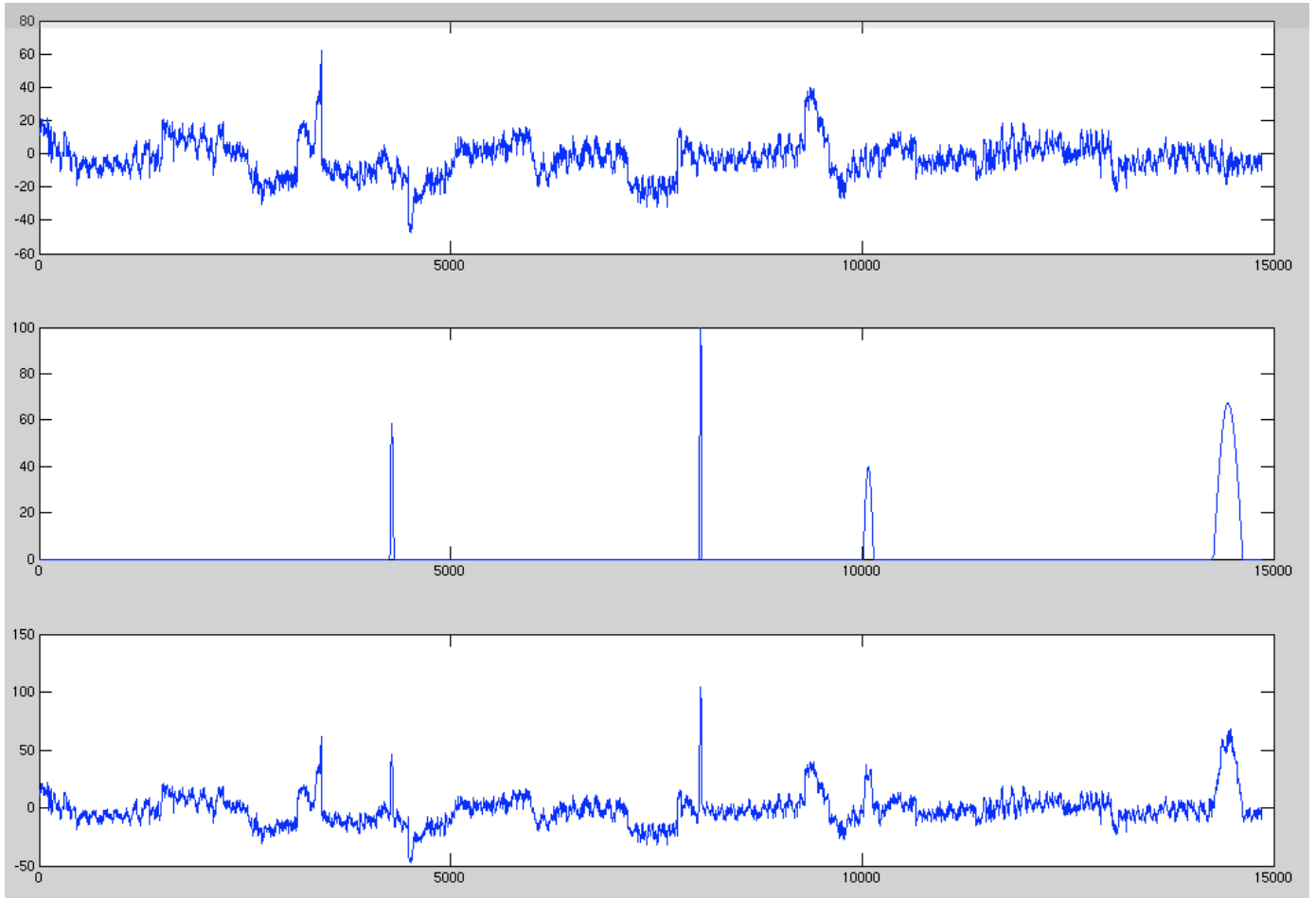


From Top to Bottom:

- Clean, Blink-Free, Data
- Blink Stream
- Clean, Blink-Free, Data + Blink Template * Blink Stream

"Blinky" Data Set # 7

Channel # 1: First 15,000 Data Samples



From Top to Bottom:

- Clean, Blink-Free, Data
- Blink Stream
- Clean, Blink-Free, Data + Blink Template * Blink Stream

```

% MakeBlinkSimData (Note: BSF = Blink Scale Factor)

clear all; close all; clc; load BlinkFreeEegData.mat; load blinkTemplate.txt -ascii;

origDataWidth = size(blinkFreeEeg, 2); remDataWidth = origDataWidth; blinkStream = []; i = 1;

usrMsg = sprintf('Number of data samples: %d', origDataWidth); disp(usrMsg);
usrMsg = sprintf('Maximun EEG data value (rounded): %d', round(max(max(blinkFreeEeg)))); disp(usrMsg);
usrMsg = sprintf('Minimun EEG data value (rounded): %d', round(min(min(blinkFreeEeg)))); disp(usrMsg);
usrMsg = sprintf('Maximum blink template value (rounded): %d', round(max(blinkTemplate))); disp(usrMsg);
usrMsg = sprintf('Minimum blink template value (rounded): %d', round(min(blinkTemplate))); disp(usrMsg);
disp('-----');
minBlinkWidth = input('Minimum blink width (in # of data samples): ');
maxBlinkWidth = input('Maximum blink width (in # of data samples): ');
minBlinkSep = input('Minimum blink separation (in # of data samples): ');
maxBlinkSep = input('Maximum blink separation (in # of data samples): ');
minBlinkHght = input('Minimum blink scale factor: ');
maxBlinkHght = input('Maximum blink scale factor: ');

notRandomBlinks = (minBlinkWidth == maxBlinkWidth) && (minBlinkSep == maxBlinkSep) && (minBlinkHght == maxBlinkHght);
randomBlinks = ~notRandomBlinks;

if notRandomBlinks
    usrMsg = sprintf('\nCreating fixed, non-random, blink stream...'); disp(usrMsg);
    blinkWidth = minBlinkWidth;
    blinkSep = minBlinkSep - blinkWidth;
    blinkHght = minBlinkHght;
    t = linspace(0, pi, blinkWidth); y = blinkHght * sin(t); blink = [zeros(1, blinkSep) y];
else
    usrMsg = sprintf('\nCreating random blink stream...'); disp(usrMsg);
    blinkWidth(i) = floor(minBlinkWidth + (maxBlinkWidth - minBlinkWidth) * rand);
    blinkSep(i) = floor(minBlinkSep + (maxBlinkSep - minBlinkSep) * rand - blinkWidth(i));
    blinkHght(i) = minBlinkHght + (maxBlinkHght - minBlinkHght) * rand;
    t = linspace(0, pi, blinkWidth(i)); y = blinkHght(i) * sin(t); blink = [zeros(1, blinkSep(i)) y];
end

while (remDataWidth >= length(blink))

    blinkStream = [blinkStream blink]; remDataWidth = remDataWidth - length(blink);

    if randomBlinks
        i = i + 1;
        blinkWidth(i) = floor(minBlinkWidth + (maxBlinkWidth - minBlinkWidth) * rand);
        blinkSep(i) = floor(minBlinkSep + (maxBlinkSep - minBlinkSep) * rand - 0.5 * (blinkWidth(i-1) + blinkWidth(i)));
        blinkHght(i) = minBlinkHght + (maxBlinkHght - minBlinkHght) * rand;
        t = linspace(0, pi, blinkWidth(i)); y = blinkHght(i) * sin(t); blink = [zeros(1, blinkSep(i)) y];
    end

end

if (remDataWidth ~= 0)
    pad = zeros(1,remDataWidth);
    blinkStream = [blinkStream pad];
end

blinkSegPntr = find(blinkStream ~= 0); blinkFreeSegPntr = find(blinkStream == 0);
blinkStream = blinkTemplate * blinkStream; blinkyEeg = blinkFreeEeg + blinkStream;

save('SimBlinkData', 'blinkStream', 'blinkyEeg', 'blinkSegPntr', 'blinkFreeSegPntr', 'blinkHght', 'blinkSep', 'blinkWidth');

figure;
subplot(3,1,1); plot(blinkFreeEeg(1, 1:round(origDataWidth/4)));
subplot(3,1,2); plot(blinkStream(1, 1:round(origDataWidth/4)));
subplot(3,1,3); plot(blinkyEeg(1, 1:round(origDataWidth/4)));

```

```
set(gcf, 'Position', [100 50 1124 818]);
```