Force Reconstruction Algorithm (Version 1.2)

```
Input:
    • $N_W = 30$
    • $Thr\_samples = 1500$
    • $n_\sigma = 10$
    • $press\_confirm = 5$
    • $reset\_confirm = 20$
    • $reset\_band\_scale = 1.0$
    • $avg\_multiplier = 1.0$
    • $signal2noise\_ratio = 10.0$
    • $n_{offset} = 50$
    • $hold\_duration = 2000$
    • $\alpha = 0.05$
Initialization
$pre\_trigger\_len \leftarrow press\_confirm + 5$
$pre\_trigger\_buffer[pre\_trigger\_len] \leftarrow 0$
$offset\_buffer[n_{offset}] \leftarrow 0$
$adaptive\_offset \leftarrow 0$
$integral \leftarrow 0$, $counter \leftarrow 0$
$last\_integral \leftarrow 0$
$confirm\_count \leftarrow 0$
$previous\_polarity \leftarrow 0$
$idx\_max \leftarrow 0$
$max\_post\_trigger \leftarrow 0$
$noise\_level \leftarrow 0$
$\epsilon \leftarrow 10^{-6}$
$sign \leftarrow +1$, $first\_cross \leftarrow False$
$first\_window\_done \leftarrow False$
$states \leftarrow [””,””,””]$, $idx\_states \leftarrow 0$
$in\_release\_mode \leftarrow False$
$reset\_integral \leftarrow False$
$hold\_counter \leftarrow 0$
$peak\_detected \leftarrow False$
$valid\_event \leftarrow True$
1. Threshold Estimation
$\sigma_0 \leftarrow \mathrm{std}(X_{raw}[0 : Thr\_samples])$
$\mathrm{thr}_{press} \leftarrow n_\sigma \cdot \sigma_0$
$max\_noise \leftarrow \max(X_{raw}[0 : Thr\_samples])$
Initialize adaptive offset using last $n_{offset}$ samples
2. Online Processing Loop
for each new sample $x_{raw}$ do
    $x \leftarrow x_{raw} - adaptive\_offset$
    $noise\_level \leftarrow |max\_noise - adaptive\_offset|$
    if $|x| < \mathrm{thr}_{press}$ then
        Update $adaptive\_offset$ via rolling mean
        $previous\_polarity \leftarrow 0$
    if $reset\_integral$ then
```

```
        Reset all internal variables
        $idx\_max \leftarrow 0$
        $max\_post\_trigger \leftarrow 0$
        **continue**
    **2.1 Confirmed Threshold Crossing**
    **if not** $first\_cross$ **then**
        **if** $x > \mathrm{thr}_{press}$ and $previous\_polarity \neq +1$ **then**
            $confirm\_count \leftarrow confirm\_count + 1$
            **if** $confirm\_count \geq press\_confirm$ **then**
                $first\_cross \leftarrow True$
                $sign \leftarrow +1$
                $quiet\_band \leftarrow +reset\_band\_scale \cdot \mathrm{thr}_{press}$
                $integral \leftarrow \sum pre\_trigger\_buffer$
                $confirm\_count \leftarrow 0$
        **else if** $x < -\mathrm{thr}_{press}$ and $previous\_polarity \neq -1$ **then**
            $confirm\_count \leftarrow confirm\_count + 1$
            **if** $confirm\_count \geq press\_confirm$ **then**
                $first\_cross \leftarrow True$
                $sign \leftarrow -1$
                $quiet\_band \leftarrow -reset\_band\_scale \cdot \mathrm{thr}_{press}$
                $integral \leftarrow -\sum pre\_trigger\_buffer$
                $confirm\_count \leftarrow 0$
        **else**
            $confirm\_count \leftarrow 0$
            **continue**
    **if** $|x| > max\_post\_trigger$ **then**
        $max\_post\_trigger \leftarrow |x|$
        $idx\_max \leftarrow idx\_max + 1$
    **2.2 Integration**
    **if** $valid\_event$ **then**
        $integral \leftarrow integral + sign \cdot x$
        **if** $integral < 0$ **then**
            $integral \leftarrow 0$
    **else**
        $integral \leftarrow 0$
    **2.3 HOLD / Peak Detection**
    **if not** $peak\_detected$ **then**
        **if** $integral \leq last\_integral + \epsilon$ **then**
            $hold\_counter \leftarrow hold\_duration$
            $held\_integral \leftarrow integral$
            $peak\_detected \leftarrow True$
    **if** $hold\_counter > 0$ **then**
        Output $held\_integral$
        $hold\_counter \leftarrow hold\_counter - 1$
        **continue**
    $last\_integral \leftarrow integral$
    **2.4 Release Reset Logic**
    **if** $in\_release\_mode$ **then**
        **if** $(x - quiet\_band) \cdot sign < 0$ **then**
            $quiet\_count \leftarrow quiet\_count + 1$
        **else**
            $quiet\_count \leftarrow 0$
        **if** $quiet\_count \geq reset\_confirm$ or $integral \leq 0$ **then**
            $reset\_integral \leftarrow True$
            **continue**
    2.5 Window Processing
    if window full then
        $idx \leftarrow \min(idx\_max + pre\_trigger\_len,\ N_W - 1)$
        $avg \leftarrow \frac{-buffer[0] + buffer[idx]}{idx + 1}$
        if not $first\_window\_done$ then
            if $max\_post\_trigger < signal2noise\_ratio \cdot noise\_level$ then
                Discard event
                continue
            $average_{touch} \leftarrow \mathrm{round}(avg)$
            $average_{plateau} \leftarrow average_{touch}/10$
            $first\_window\_done \leftarrow True$
            $states[0] \leftarrow set\_label(avg)$
        else
            $st \leftarrow set\_label(avg)$
            Update rolling 3-state buffer
            if $st ==$ “release” then
                $in\_release\_mode \leftarrow True$

function SET_LABEL(avg)
    if $avg \geq avg\_multiplier \cdot average_{touch}$ then
        return “press”
    else if $avg \leq -avg\_multiplier \cdot average_{touch}$ then
        return “release”
    else if $|avg| \leq average_{plateau}$ then
        return “plateau”
    else
        return “transition”
```