



AN-1419

アプリケーション・ノート

ADUX1020 によるジェスチャ認識

著者: Tyler Ray

はじめに

ADUX1020 は、高精度なジェスチャ認識を単一点検出し、近接度を検出可能な光学センサーならびにフロントエンドの集積デバイスです。この光学センサーは入射光の強度と入射角を検出することができます。さまざまなノイズが存在する環境でも、光学的および電子的な周辺光フィルタリングを組み合わせることにより、確実な動作が可能です。このデバイスは、非接触ジェスチャ認識と制御、あるいは近接度の検出を可能にするために、複数のモードに設定することができます。

ジェスチャ認識の基礎

ADUX1020 は光の入射角を計測できるので、内蔵の光学センサーと 1 個の外付け LED のみでジェスチャを検出することができます。デバイスをサンプル・モードに設定すると、その 4 本の出力チャンネルから入射光の入射角を計算できます。ジェスチャ認識機能を使用する場合は、以下の手順に従ってください。

1. ADUX1020 を動作させる前に、クロックを校正します。32 kHz クロックと 32 MHz クロックの校正方法については、ADUX1020 のデータシートを参照してください。
2. レジスタ 0x45 のビット [3:0] に 0x3 を書き込んで、ADUX1020 の動作モードをサンプル・モードに設定します。
3. デバイスによって測定されたデータを収集します。先入れ先出し (FIFO) と割り込みを使ってレジスタからデータを読み出す方法については、ADUX1020 のデータシートを参照してください。データは、データ・レジスタから直接読み出すか、またはレジスタ 0x60 の 64 バイト FIFO のビット [15:0] から読み出すことができます。
4. ADUX1020 の 4 本の出力チャンネルのデータは、光の入射角の計算に使用します。チャンネル X₁、チャンネル X₂、チャンネル Y₁、およびチャンネル Y₂ のデータを収集した後は、以下の式によって角度と強度を計算します。

$$\text{水平方向の角度: } \theta_x = \frac{X_1 - X_2}{X_1 + X_2} \text{ (ラジアン)}$$

$$\text{垂直方向の角度: } \theta_y = \frac{Y_1 - Y_2}{Y_1 + Y_2} \text{ (ラジアン)}$$

$$\text{強度: } I = X_1 + X_2 + Y_1 + Y_2 \text{ (コード)}$$

このデータの例を図 1 と図 2 に示します。

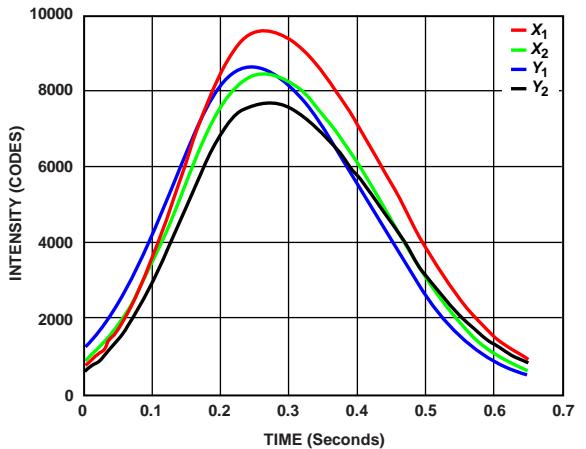


図 1. 強度と時間の関係、ADUX1020 の未加工データ出力

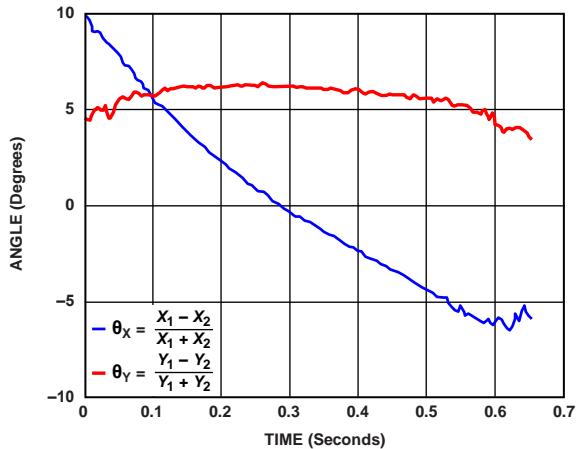


図 2. 角度と時間の関係、ADUX1020 の角度計算

アナログ・デバイセズ社は、提供する情報が正確で信頼できるものであることを期していますが、その情報の利用に関して、あるいは利用によって生じる第三者の特許やその他の権利の侵害に関して一切の責任を負いません。また、アナログ・デバイセズ社の特許または特許の権利の使用を明示的または暗示的に許諾するものではありません。仕様は、予告なく変更される場合があります。本紙記載の商標および登録商標は、それぞれの所有者の財産です。※日本語版資料は REVISION が古い場合があります。最新の内容については、英語版をご参照ください。

イベント検出とジェスチャ・ウィンドウ・アルゴリズム

このセクションに示す以下のアルゴリズム記述の疑似コードは、MATLAB の構文に基づいています。ループ、case 文、および関数は MATLAB の標準フォーマットに従っており、いずれも実際の MATLAB コードから得られたものです。

ADUX1020 評価ソフトウェアは、5 種類の単位ジェスチャをサポートしています。つまり、上にスワイプ、下にスワイプ、左にスワイプ、右にスワイプ、およびクリックです。アルゴリズムは、これらのジェスチャの 1 つにデータを分類する前に、まずイベントの発生を待ちます。

さらに、イベントを検出する前に、4 本のチャンネルそれぞれのオフセットを除去する必要があります。これは、デジタル的にオフセットを差し引くか、デバイスのオフセットを変更することによって行います。オフセットのデジタル除去は最も簡単な処理方法ですが、デバイスのオフセットを変更する場合は A/D コンバータ (ADC) のダイナミック・レンジを最大限に使用するので、高いダイナミック・レンジが求められる場合に用いることを推奨します。0x3A、0x3B、0x3C、および 0x3D の各レジスタには ADC のオフセットが含まれていて、その公称設定値は 0x2000 です。これらのオフセットを変更するには、各チャンネルの 16 ビット出力を ADC コードで測定して、それを ADC オフセット・レジスタ CHx_OFFSET 内の 16 ビット数値（公称値は 0x2000）から差し引きます。次に、この減算の結果を ADC オフセット・レジスタに書き込みます。

イベントの開始は、強度データまたはその時間微分係数があらかじめ設定された閾値を超えた時として定義できます。このアプリケーション・ノートでは、強度の微分係数を使用する方法を採用していますが、両方の方法を用いることができます。

強度データは、高周波ノイズを除去するためにローパス・フィルタを通します。通常、高周波ノイズの除去は、コーナー周波数が 10 Hz の 2 次バターワース・ローパス・フィルタを使用して行います。

$$I_F = LPF(I)$$

連続するフィルタリング済み強度サンプルの差を取ることによって、フィルタリング済み強度の微分係数を計算します。

$$\frac{dI}{dt}(n) = I_F(n+1) - I_F(n)$$

イベントは、強度の微分係数 dI/dt が既定の閾値 I_{TH} を超えた時に開始されたと認識されます。閾値の単位は ADC コードです。通常、この閾値は 200 ADC コードに設定されますが、より大きいイベントを除去して望ましくないイベントがジェスチャとして分類されないようにするために、あるいは、より小さいイベントをジェスチャに含めるために、増減させることができます。ジェスチャの開始は、 dI/dt の値が閾値に達するまで、この値を連続的にスイープすることで検出できます。

For $n = 1 : \text{numel}(dI/dt)$

```
if  $dI/dt(n) > I_{TH}$ 
    gestureStart = n
    break
```

ジェスチャの終了は、 dI/dt データを逆にスイープすることにより検出します。スイープは、ジェスチャ開始時間にジェスチャの最大長さを加えた時点から開始して、フィルタ済み強度の微分係数が $-I_{TH}$ より小さくなるまで続けます。ジェスチャの開始時間と終了時間の決定例を図 3 と図 4 に示します。

For $n = (\text{gestureStart} + \text{gestureDuration}) : -1 : \text{gestureStart}$

```
if  $dI/dt(n) < -I_{TH}$ 
```

```
gestureStop = n
```

```
break
```

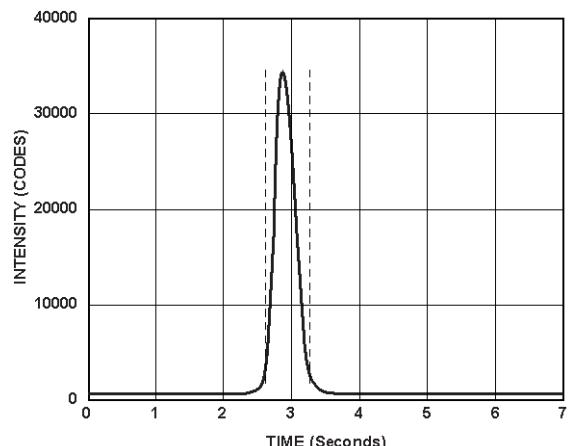


図 3. 強度と時間の関係、およびジェスチャの開始時間と終了時間

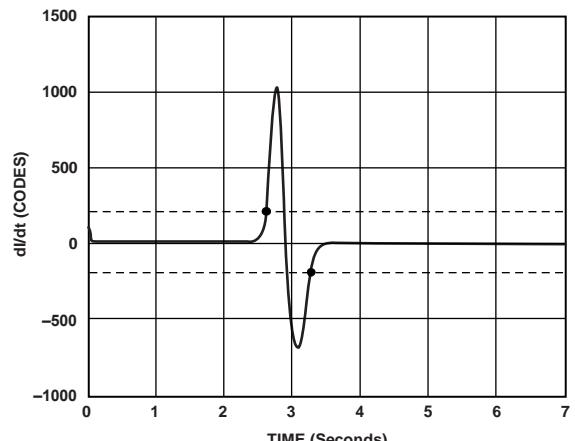


図 4. dI/dt と時間の関係、およびジェスチャの開始時間と終了時間

開始時間と終了時間によってジェスチャを定めたら、ジェスチャに含まれる θ_X と θ_Y の最大値と最小値を計算することができます。 $\max(\theta_X) - \min(\theta_X)$ と $\max(\theta_Y) - \min(\theta_Y)$ の値が両方とも閾値の θ_{TH} より小さい場合、そのイベントはクリックとして分類されます。イベントがクリックでない場合は、ジェスチャの方向を決定するために、さらに処理が必要です。 θ_X と θ_Y の間の線形最小二乗適合線は、次式を使って、そのジェスチャが上下に分類されるものか左右に分類されるものかを決定します。

$$\theta_Y = m\theta_X + b$$

ここで、 m は勾配です。

この線の勾配の絶対値 m が 1 未満の場合、そのジェスチャは左右方向のジェスチャです。この線の勾配の絶対値 m が 1 より大きい場合、そのジェスチャは上下方向のジェスチャです。計算を簡単にするために、基本的な強度閾値によってもジェスチャの開始時間と終了時間を決定できることに注意してください。

If $\max(\theta_X) - \min(\theta_X) < \theta_{TH}$ and $\max(\theta_Y) - \min(\theta_Y) < \theta_{TH}$
 gesture = click

else

```
(m, b) = leastSquaresFit (θX, θY)
if m ≤ 1
  if θX (gestureStart) > θX (gestureStop)
    gesture = left
  else
    gesture = right
else if m > 1
  if θY (gestureStart) > θY (gestureStop)
    gesture = up
  else
    gesture = down
```

最小二乗適合線の例を図 5 に示します。このジェスチャの θ_X 値にはかなり広がりがあるので、クリックではありません。最小二乗適合線の勾配は 1 未満であり、 θ_X (gestureStart) > θ_X (gestureStop) なので、これは左スワイプです。

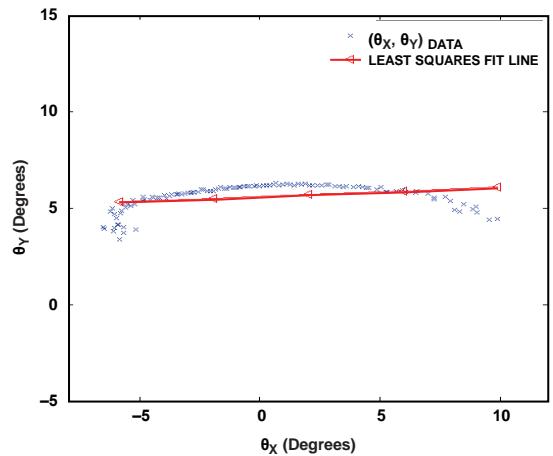


図 5. θ_Y と θ_X の関係、および最小二乗適合線

この方法のパラメータ ($-I_{TH}$, θ_{TH} と $gestureDuration$) を調整して、あらゆるユースケースに使用できるようにアルゴリズムを改良すると、これは、ジェスチャ認識の唯一可能な方法になります。ADUX1020 を使って実現できるジェスチャ認識の方法は他にもあります。計算負荷が問題となる場合は、ジェスチャの開始点と終了点だけを考えて、その間を無視することができます。この単純な方法は非常に信頼性が高く、効果的ですが、開始点と終了点の S/N 比が適切な値となるように、強度または強度の微分係数の閾値 I_{TH} を十分に高く設定する必要があります。

光学ジェスチャ評価ツールに表示される強度の dI/dt と、スワイプ・ジェスチャのジェスチャ・プロットを図 6 に示します。図 7 は、右スワイプとして認識された単位ジェスチャです。

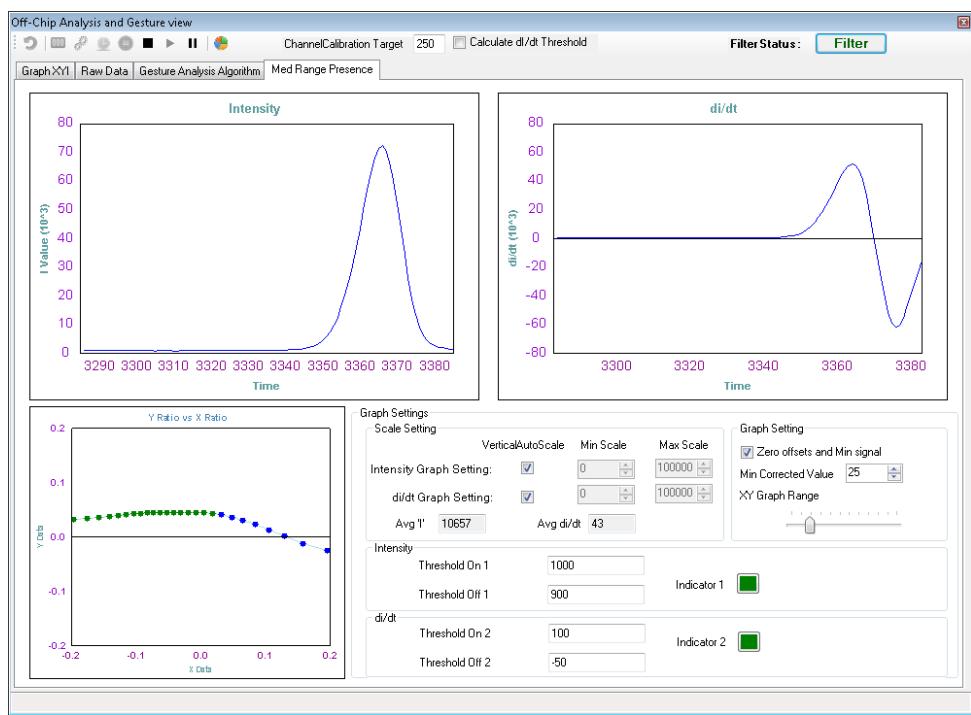
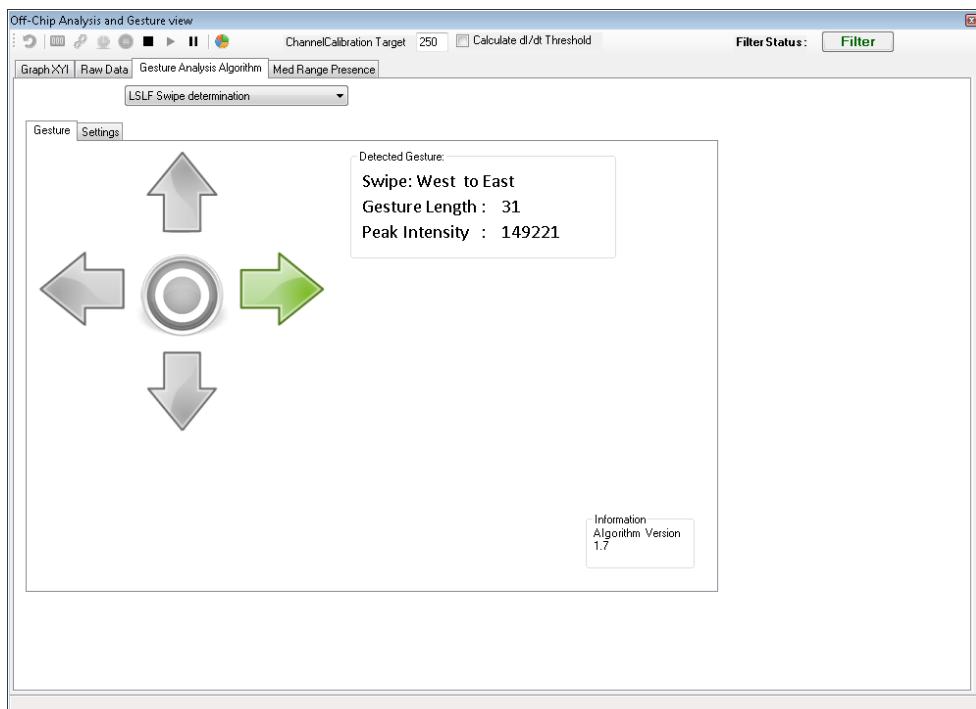
図 6. 光学ジェスチャ評価ツールによるジェスチャ認識データ（強度 dI/dt およびスワイプ・ジェスチャのジェスチャ・プロット）

図 7. 光学ジェスチャ評価ツールによるジェスチャのスワイプ判定（右スワイプとして認識）