

# 3軸デジタル加速度センサーを使用したフル機能の歩数計の設計

著者: Neil Zhao

## はじめに

毎日の運動の進捗を調べるモニターやモチベータとして普及している歩数計は、健康増進や減量に取り組む人々の励みとなっています。初期の設計は、錘式のメカニカル・スイッチを使って歩行動作を検出し、単純なカウンタで表示するものでした。装置を振ると、金属製ボールが前後にスライドしたり振り子が動いてストップに当たる音などがします。

今日の先進的な歩数計は、微小電気機械システム (MEMS) の慣性センサーと高度なソフトウェアを使い、高い確率で真の歩行動作を検出します。MEMS 慣性センサーにより、歩行動作の検出が正確になり、誤検知が減少します。低コスト、省スペース、省電力の条件を満たす MEMS 慣性センサーの利点を生かすことで、携帯音楽プレーヤーや携帯電話など多くの民生向けポータブル電子機器に歩数計が組み込まれています。アナログ・デバイセズの小型、薄型、低消費電力の3軸加速度センサー ADXL335、ADXL345、ADXL346 はこのようなアプリケーションに最適です。

本稿では、人間の歩行動作の特性に関する研究にもとづき、3軸加速度センサー ADXL345 を使用して歩行動作の検出とカウントを行い、距離と速度のほか、ある程度の消費カロリーの計算まで可能な多彩な機能を備えた歩数計のリファレンスデザインをご紹介します。

ADXL345 の独自の組み込み 32 レベル FIFO (First-In, First-Out) バッファは、歩数計アプリケーション用にデータの保存と演算が可能であり、ホスト・プロセッサの使用を最小限に抑え、

携帯型機器にとって最大の課題であるシステムの消費電力を低減することができます。分解能は 13 ビット (4mg/LSB) で、歩行速度が遅い場合 (1 歩の加速度変化が約 55mg 程度) でも、十分な精度で測定することができます。

## モデルの理解

ここでは、ランニングやウォーキングの解析に使用できる特性の中から、関連パラメータとして加速度を選びます。人間の動作の3つの成分 (および関連する軸) は、図1に示すように前方向 (ロール軸)、垂直方向 (ヨー軸)、横方向 (ピッチ軸) です。ADXL345 は、その3つの軸、すなわち x、y、z 軸方向の加速度を検出します。歩数計の向きは決まっていないため、測定精度が運動軸と加速度センサー測定軸の関係に決定的に依存しないようにします。

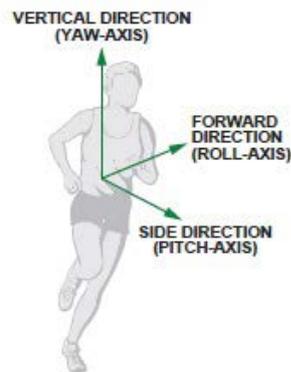


図1. 各軸の定義

人の歩き方の性質について考えてみましょう。図2は、歩行時の1歩を歩行動作の単位サイクルとして定義したもので、歩行サイクルの各段階と垂直方向および前方向の加速度の変化との関係を示しています。

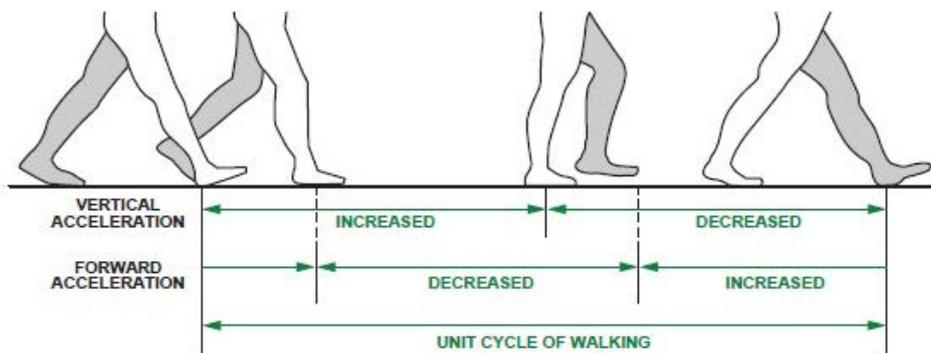


図2. 歩行の各段階と加速度パターン

図3は、走っている人の垂直、前方、横方向加速度に対応するx、y、z軸方向の測定値の標準的なパターンを示したものです。歩数計をどのように身に付けているかにかかわらず、少なくとも1つの軸に相対的に大きな加速度の周期的変化が見られます。このため、ウォーキングやランニングの単位サイクルを検出するには、3軸すべてにおける加速度のピーク値検出と動的にしきい値を決めるアルゴリズムが不可欠です。

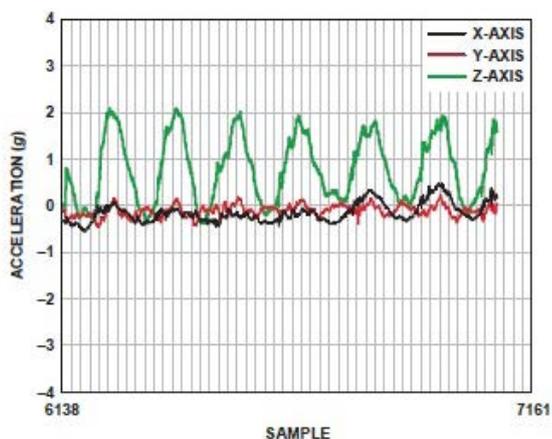


図3. 走っている人のx、y、z方向加速度の標準的なパターン

## アルゴリズム

### 歩行パラメータ

デジタル・フィルタ：まず、図3に示す信号の平滑化のためにデジタル・フィルタが必要です。図4に示すように、4個のレジスタと1個の加算ユニットを使用できます。もちろん、もっと多くのレジスタを使用して加速度データをさらに平滑にすることも可能ですが、そのようにすると応答速度が低下してしまいます。

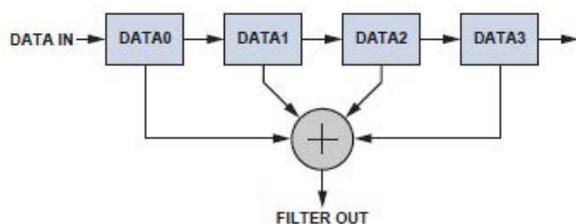


図4. デジタル・フィルタ

図5は、歩行者が装着している歩数計で、加速度変化が最も大きい軸のデータにフィルタ処理を行ったものを示しています。走っている場合、ピークtoピーク値はさらに大きくなります。

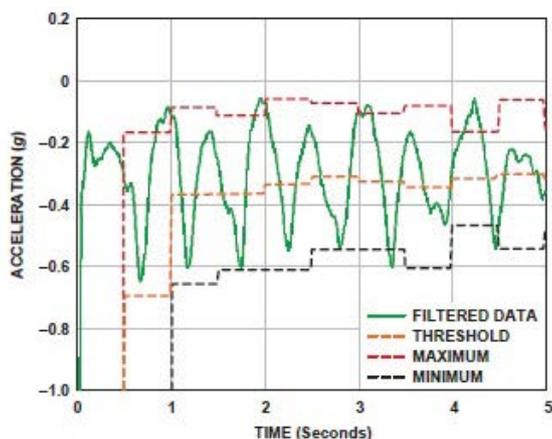


図5. 変化が最も大きい軸のフィルタ処理後のデータ

動的しきい値と動的精度：システムは、3軸の加速度の最大値と最小値を50サンプルごとに継続的に更新します。その平均値、 $(Max + Min)/2$ を動的しきい値レベルといいます。次の50サンプルについては、このしきい値レベルを使って歩行が行われて

いるかどうかを決定します。50サンプルごとに更新されるため、動的しきい値になります。これは適応型の方法であり、なおかつ十分に高速です。動的しきい値のほか、図6に示すように動的精度を使ったフィルタ処理も行われます。

有効な1歩かどうかを決定するには、線形シフト・レジスタと動的しきい値が使われます。線形シフト・レジスタには2つのレジスタ、sample\_newレジスタとsample\_oldレジスタがあります。これらのレジスタ内のデータは、それぞれsample\_newおよびsample\_oldといいます。新しいデータ・サンプルが取り込まれると、sample\_newは無条件にsample\_oldにシフトされます。しかし、sample\_resultがsample\_newレジスタにシフトされるかどうかは、条件によって異なります。加速度の変化があらかじめ設定された精度よりも大きい場合、最新のサンプル結果のsample\_resultはsample\_newレジスタにシフトしますが、それ以外の場合sample\_newレジスタは変化しません。これらのシフト・レジスタ群によって高周波ノイズを除去し、精度の高い決定が可能になります。

歩行時に1歩を踏み出す動作は、加速度曲線が動的しきい値を下回るときに、加速度プロットのスロープが負である (sample\_new < sample\_old) 場合と定義されます。

ピーク検出：歩数カウンタは、どの軸の加速度変化が最も大きいかに応じて、x軸、y軸、またはz軸から歩数を計算します。加速度の変化が小さすぎる場合は、無視されます。

歩数カウンタはこのアルゴリズムで問題なく機能しますが、感度が高すぎることもあります。歩行やランニング以外の理由で歩数計が非常に速く、あるいは非常に遅く振動した場合も、歩数カウンタは歩数としてカウントしてしまうでしょう。一定のリズムをもった真の歩数を検出するためには、このような振動を除外する必要があります。この問題は、時間窓とカウント調整を利用して解決できます。

時間窓は、無効な振動を除外するために使用します。ここでは、人間が走るときの最大歩数を1秒あたり5歩、歩くときの最小歩数を2秒あたり1歩と仮定します。したがって、有効な2歩の間隔を[0.2 ~ 2.0s]の時間窓として定義し、2歩の間隔がこの時間窓に入らない動きはすべて除外します。

ADXL345は、出力データレートを選択できるため、時間窓の実装に便利です。T<sub>A</sub> = 25°C、V<sub>S</sub> = 2.5V、V<sub>DD I/O</sub> = 1.8Vにおいて設定可能なデータレート (および消費電流) を表1に示します。

表1. データレートと消費電流

出力データレート (Hz)	帯域幅 (Hz)	レートコード	I <sub>DD</sub> (μA)
3200	1600	1111	145
1600	800	1110	100
800	400	1101	145
400	200	1100	145
200	100	1011	145
100	50	1010	145
50	25	1001	100
25	12.5	1000	65
12.5	6.25	0111	55
6.25	3.125	0110	40

このアルゴリズムは50Hzのデータレートを使用します (20ms)。intervalレジスタは、人間が2歩進む間にデータが更新された回数を記録します。intervalの値が10 ~ 100の場合、2歩進む間の時間が有効な範囲にあることを意味します。それ以外の値は時間窓の範囲外であり、歩行動作は無効になります。カウント調整は、それぞれの歩行動作が一定のリズムのパターンの一部かどうかを決定します。歩数カウンタには、検索調整と確認調整という2つの動作状態があります。歩数カウンタが開始する

ときは、検索調整モードで動作します。たとえば、有効な4歩の連続した歩行動作によって動作の規則性が確認されたものとします。その後結果がリフレッシュされて表示され、歩数カウンタが確認調整モードになります。このモードでは、歩数カウンタが有

効な1歩ごとにリフレッシュされます。ところが、1歩でも無効な歩行動作が確認されると歩数カウンタは検索調整モードに戻り、有効な歩行動作が4回連続して確認されるのを待ちます。歩行パラメータのアルゴリズムのフローチャートを図7に示します。

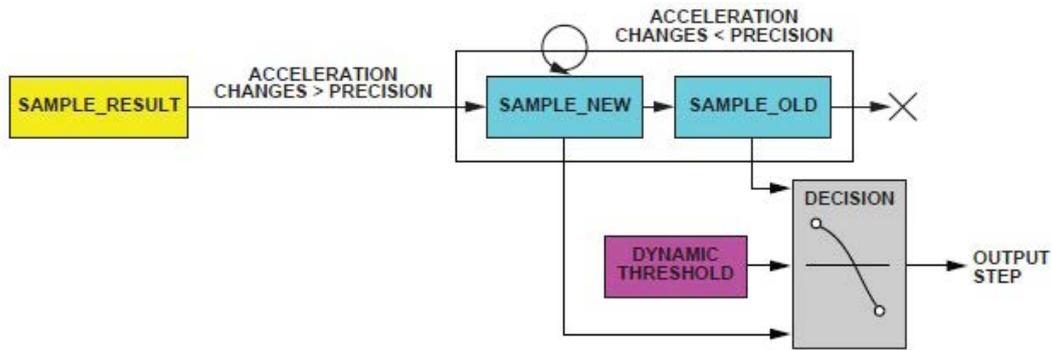


図6. 動的しきい値と動的精度

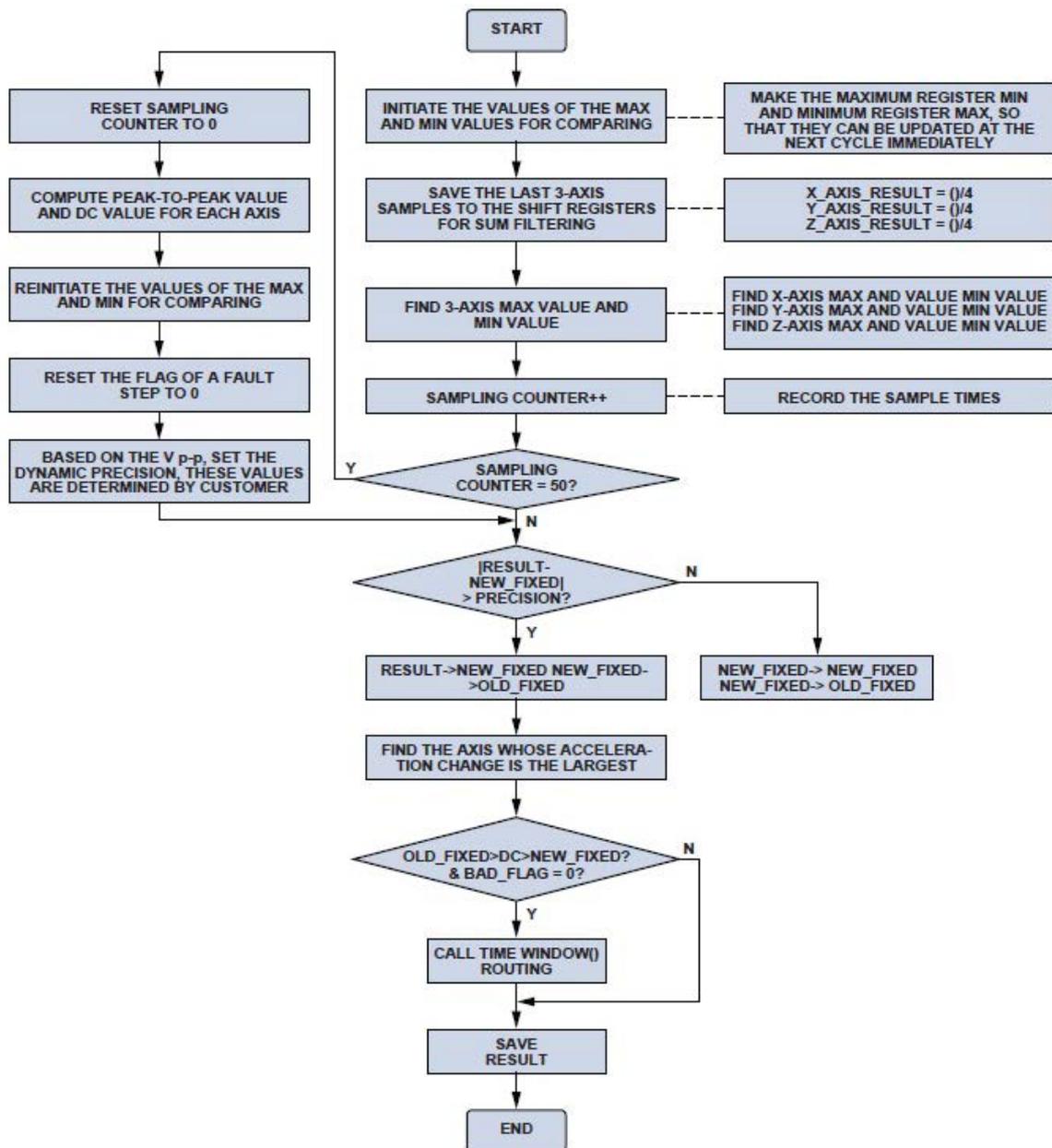


図7. 歩行パラメータのアルゴリズムのフローチャート

## 距離パラメータ

上述のアルゴリズムに従って歩行パラメータを計算した後は、式1によって距離パラメータを求めることができます。

$$\text{距離} = \text{歩数} \times 1\text{歩あたりの距離} \quad (1)$$

1歩あたりの距離は、速度とユーザーの身長によって異なります。ユーザーの身長が高い場合や走っている場合は、1歩あたりの距離が長くなります。参考設計では、距離、速度、カロリーの各パラメータを2秒ごとに更新します。したがって、2秒ごとにカウントされた歩数により現在の歩幅を判定します。現在の歩幅を判定するための実験データを表2に示します。

表2. 速度 (2秒あたりの歩数) と身長の間数としての歩幅

2秒あたりの歩数	歩幅 (m/s)
0~2	身長/5
2~3	身長/4
3~4	身長/3
4~5	身長/2
5~6	身長/1.2
6~8	身長
>=8	1.2 × 身長

データレートを参照し、プロセッサは対応するコマンドを100サンプルごとにPCに送ることができます。プロセッサは、それぞれの2秒間隔の開始時点でm\_nLastPedometerという変数を使って歩数を記録し、終了時点ではm\_nPedometerValueという変数を使って歩数を記録します。次に、m\_nPedometerValueからm\_nLastPedometerを引くことによって、2秒あたりの歩数を計算します。

データレートは50Hzですが、ADXL345には組込みFIFOがあるためプロセッサが20msごとにデータを読み込む必要はなく、ホスト・プロセッサの負荷は最小限に抑えられます。バッファには、バイパス、FIFO、ストリーム、トリガーの4モードがあります。FIFOモードでは、x、y、z軸の測定値から得られたデータがFIFOに保存されます。FIFO内のサンプル数が、FIFO\_CTLレジスタのサンプル・ビットで指定されたレベルに等しくなると、ウォーターマーク割込みがセットされます。先に述べたように、人の走る速さは1秒あたり5歩くらいまでなので、リアルタイムの結果を出すには、0.2秒ごとにリフレッシュする必要があります。プロセッサは0.2秒ごとにADXL345からデータをフェッチするだけでよいので、ウォーターマーク割込みによる作動が可能です。FIFOのその他の機能も非常に便利です。トリガー・モードを使用すると、FIFOが割込み前の状態を知らせてくれます。ご紹介したソリューションでは他のFIFO機能を使用していないため、これ以上は説明しません。

## 速度パラメータ

速度 = 距離 / 時間なので、上述のアルゴリズムに従って2秒あたりの歩数と歩幅をすべて計算してあるので、式2を使用して速度パラメータを求めることができます。

$$\text{速度} = 2\text{秒あたりの歩数} \times \text{歩幅} / 2\text{秒} \quad (2)$$

## カロリー・パラメータ

カロリーの消費率については正確に計算する方法はありません。カロリー消費の値を決定する要素としては、体重、運動の強度、コンディショニング・レベル、代謝作用などがあり、一般的な近似法を使用して推定することができます。カロリー消費とランニング速度の代表的な関係を表3に示します。

表3. ランニング速度 対 カロリー消費

ランニング速度 (km/h)	カロリー消費 (C/kg/h)
8	10
12	15
16	20
20	25

表3から、(3)の関係式が得られます。

$$\text{カロリー (C/kg/h)} = 1.25 \times \text{ランニング速度 (km/h)} \quad (3)$$

前に使用した速度パラメータの単位はm/sですから、km/hをm/sに変換し、式4が得られます。

$$\begin{aligned} \text{カロリー (C/kg/h)} &= 1.25 \times \text{速度 (m/s)} \times 3600/1000 \quad (4) \\ &= 4.5 \times \text{速度 (m/s)} \end{aligned}$$

カロリー・パラメータは、距離パラメータおよび速度パラメータとともに2秒ごとに更新されます。そこで、ユーザーの体重を計算に入れるために、式4を式5のように変換します。体重 (kg) はユーザーが入力します。また、1時間は1800個の2秒間隔になります。

$$\begin{aligned} \text{カロリー (C/2s)} &= 4.5 \times \text{速度} \times \text{体重} / 1800 \quad (5) \\ &= \text{速度} \times \text{体重} / 400 \end{aligned}$$

ウォーキングやランニングの後に休憩を取る場合は歩幅と距離に変化はなく、速度はゼロになるはずですが、したがって、休憩中のカロリー消費は約1C/kg/hourなので、式6を使ってカロリー消費量を計算することができます。

$$\text{カロリー (C/2s)} = 1 \times \text{体重} / 1800 \quad (6)$$

最後に、すべての2秒間隔のカロリーを合計すれば、合計消費カロリーを求めることができます。

## ハードウェアの接続

ADXL345は、I2C®やSPIデジタル通信プロトコルによって、どのプロセッサにも容易に接続することができます。図8は、3Vバッテリーを電源とするデモ装置の簡略接続図です。I2Cモードを選択するには、ADXL345の/CSピンをボードのVSに接続します。低コストの高精度アナログ・マイクロコントローラADuC7024によってADXL345のデータを読み出し、アルゴリズムを実装し、UART経由でPCに結果を送信します。I2Cバス用のデータSDAとクロックSCLは、ADXL345からADuC7024の対応するピンに接続します。ADXL345の2本の割込みピンは、ADuC7024のIRQ入力に接続し、さまざまな割込み信号を生成してプロセッサを作動させます。

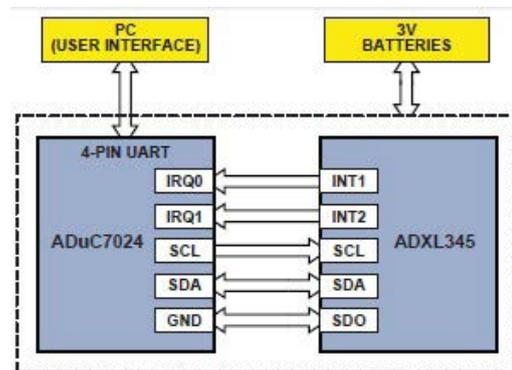


図8. ハードウェア・システムの簡略接続図

## ユーザーインターフェース

ユーザーインターフェースはテストデータを表示し、オペレータのコマンドに応答します。ユーザーインターフェースの実行後にシリアル・ポートがオープンし、通信リンクがスタートする必要があります。その後は、連続してデモを行うことができます。ユーザーが歩数計を身に付けてウォーキングやランニングをしているときのテスト・デモを図9に示します。ユーザーは、自分の身長と体重のデータを入力できます。距離、速度、カロリーの各パラメータは、これらのデータをもとに計算されます。



図9. ユーザーが歩数計を身に付けてウォーキングやランニングをしているときのテスト・デモ

## 結論

ADXL345は歩数計アプリケーション用の優れた加速度センサーです。小型で薄い3mm × 5mm × 0.95mmのプラスチック・パッケージの特長を生かすことで、このデバイスを使用した歩数計は医用機器だけでなく、コンシューマ向けの楽しい電子機器にも利用できます。測定モードで40 $\mu$ A、スタンバイ・モードで0.1 $\mu$ Aという低消費電力であるため、ADXL345はバッテリー電源の製品に理想的なデバイスです。組込みFIFOは、ホスト・プロセッサの負荷を最小限に抑えるため大きな省電力効果を持ちます。また、出力データレートを選択できるため、プロセッサのタイマーを使わずに済みます。13ビットの分解能はピークtoピーク値のわずかな変化でも検出でき、高精度の歩数計が実現します。最後に、3軸出力機能と上述のアルゴリズムを組み合わせることで、歩数計をどのような場所や向きでも装着することができます。

そのほかに次のような設計も可能です。コストの制約が非常に厳しいアプリケーションの場合や、アナログ出力加速度センサーが必要な場合は、シグナル・コンディショニング済みの電圧を出力する小型、薄型、低消費電力の完全3軸加速度センサーADXL335をお勧めします。また、基板サイズが重要な場合は、ADXL346を推奨します。ADXL345よりも多くの機能が組み込まれたこの低消費電力デバイスは、小型で薄型の3mm × 3mm × 0.95mmのプラスチック・パッケージを採用しています。電源電圧範囲は1.7 ~ 2.75Vです。

## 謝辞

Charles Lee氏とHarvey Weinberg氏の豊富な技術的知見に助けをいただきました。ありがとうございました。

## 参考文献

- 1 すべてのアナログ・デバイス製品に関するデータシートと詳細な情報については、[www.analog.com/jp](http://www.analog.com/jp)をご覧ください。
- 2 [www.analog.com/en/mems/low-g-accelerometers/products/index.html](http://www.analog.com/en/mems/low-g-accelerometers/products/index.html)