

# エンドユーザ開発と IoT 活用による現場作業活動実績の可視化

古川 慈之\*

産業技術総合研究所

**要旨:** 企業内の業務に関する知識・技術・技能の伝承を支援するためには、それらの存在を明確にし、多くの人が認知できる形式で可視化することが重要である。本稿では、安価な小型 PC とセンサおよび IO モジュールを用いた IoT 技術の活用によって、主に中小製造業を対象とした現場作業者の活動を可視化する取り組みについて述べる。このような可視化自体は、市販の製品やサービスとして各事業者から提供される範囲が広がりつつあるが、本研究ではこのような可視化を使用者自ら実現するエンドユーザ開発の観点で考察する。

## 1 はじめに

企業内の業務に関する知識・技術・技能の伝承支援には、それらの存在を明確にし、多くの人が認知できる形式で可視化することが重要である。近年注目される「IoT」の技術の内、センサを用いた計測と可視化を活用することで、知識・技術・技能の伝承支援に貢献することが可能である。

本稿では、IoT 技術を活用した業務システムを中小製造業の現場に適用し、活動実績の自動収集と可視化を実現する取り組みについて述べる。このような可視化自体は、市販の製品やサービスとして各事業者から提供される範囲が広がりつつあるが、本研究ではこのような可視化を使用者自らが実現するエンドユーザ開発の観点から考察する。

## 2 エンドユーザ開発と IoT 活用

### 2.1 MZ Platform を用いた IT 化

エンドユーザ開発 (End-User Development [1] [2]) とは、ソフトウェアおよびシステムの利用者が、自ら開発を実施する形態を指す。一般的にソフトウェアの作成者と使用者は分離されているが、実際にソフトウェアおよびシステムを用いる現場が開発と運用により深く関わると、エンドユーザ開発に近い形態に移行することが見込まれ、そのような状況は業務システムの開発を内製に移行する産業界からの報告事例[3] [4]に表れている。

著者が所属する産業技術総合研究所 (産総研) では、中小製造業が自社用の業務ソフトウェアやシステムを構築すること (IT 化) を支援する目的で、エンドユーザ開発を支援する独自のソフトウェア開発環境「MZ Platform」の研究開発と普及活動を実施している[5] [6]。この MZ Platform を用いた中小製造業の IT 化のシナリオは次のようになる (図 1 参照)[7]。ここでは工業用部品の金属加工を想定しており、受注した製品に対する加工工程が計画された後、その計画に基づいて作業指示書が発行され、各作業者はその指示に沿って加工作業を実施する。複数の工程を経て完成した製品は、検査を受けて出荷される。作業者は、作業開始と終了の時刻等を記録した書類を提出することで、作業実績を報告する。この作業実績報告を IT 化することで、実績報告の集計を自動化し、進捗状況をリアルタイムに社内共有することを目標としているものとする。MZ Platform は、この IT 化の部分ソフトウェア作成の面で支援する。

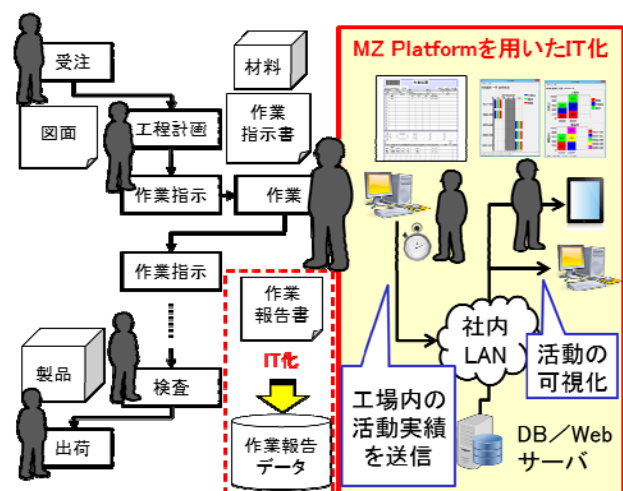


図 1 : MZ Platform を用いた IT 化[7]

\*連絡先：産業技術総合研究所  
〒305-8564 茨城県つくば市並木 1-2-1  
E-mail: y-furukawa@aist.go.jp

一方, IT 化の実現形態に目を向けると, 一般的な PC を工場内に配置してデータ入力・表示するだけでなく, 図 2 に示すように多様な形態の入出力が選択可能となっている. その中で, 現場の作業者がデータ入力を実施する代わりに, 機械から直接データ収集すること (IoT 型) が選択肢に入りつつある.

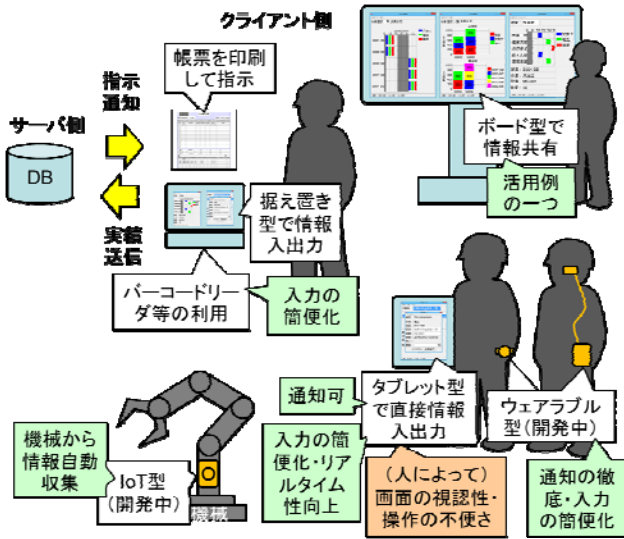


図 2 : IT 化の実現形態の選択肢[7]

## 2.2 IoT 活用とスマート製造ツールキット

これまでの MZ Platform を用いた IT 化事例では, 主に人の活動実績を人が入力し, その情報を可視化して人が見ることが中心となっていた. 今後は, 前述の IoT 型のように機械から自動で情報を収集し, その結果は人が見るだけでなくシステムが自動で状況に応じて処理を実行するシステムを構築できる必要がある. この考えから, 産総研では図 3 に示す「スマート製造環境」を想定し, その機能をエンドユーザ開発で実現するための「スマート製造ツールキット」の研究開発を実施している. 図の中央に示すように, MZ Platform を用いた IT 化はこれまで通り重要な部分を占めるが, その周辺に機械から情報を自動収集する機能を追加し, また得られた情報からシステム側で状況に応じて通知を発行する機能を追加する. なお, 機械から情報を自動収集する機能を提供する製品が既に多数存在することから, 各種規格に対応することで可能な限り多様な機器と連携できるようにする予定である. さらに, 最も重要な点として, これら追加された機能が MZ Platform と一体で配布され, ユーザが独自に試せる環境 (ツールキット) を提供することを目標としている.

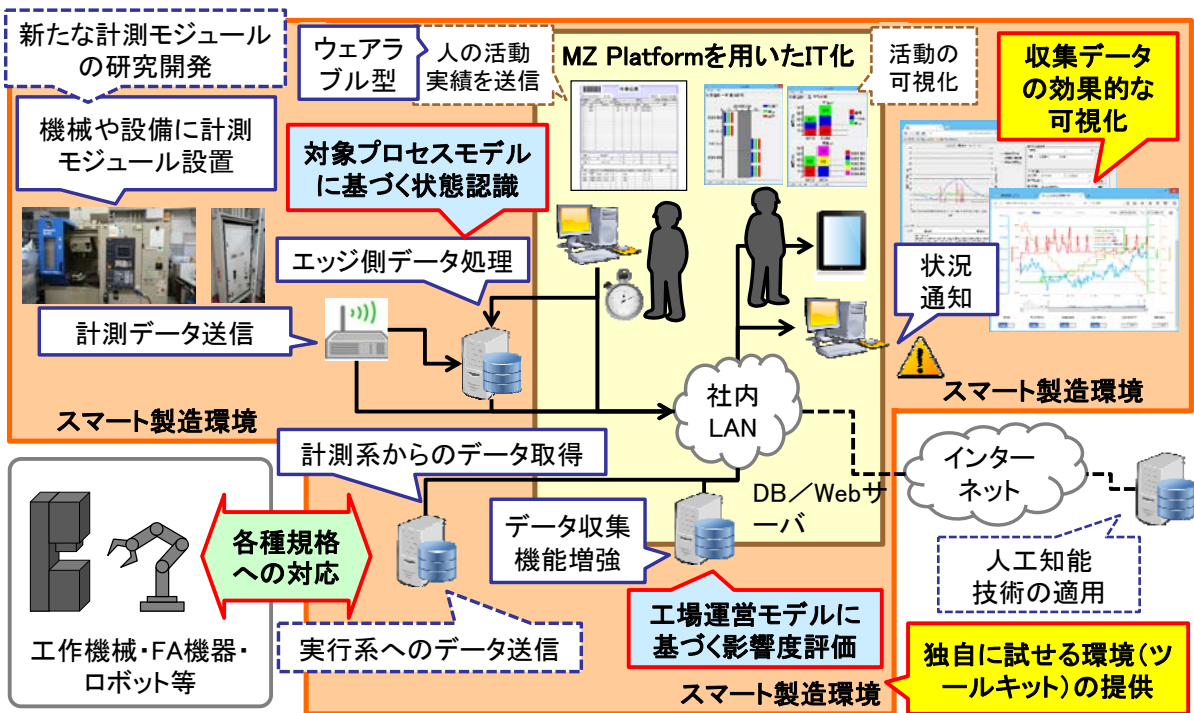


図 3 : 産総研におけるスマート製造ツールキットの研究開発

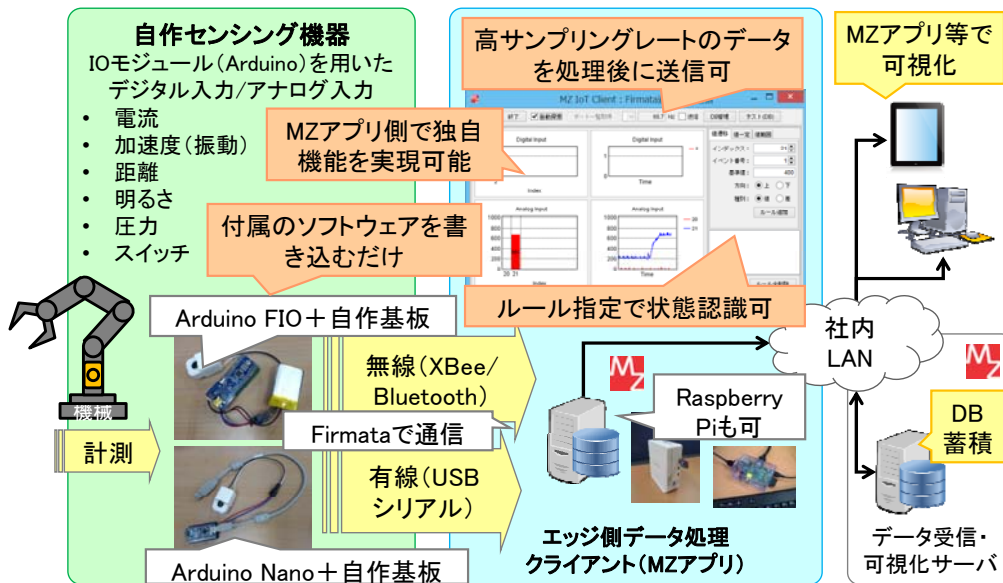


図4: センシング機器の自作による計測システム構築

### 3 自作機器を用いた計測システム

現在開発中のスマート製造ツールキットの要素である計測システム構築の機能を図4に示す。MZ Platformは、従来からシリアル通信機能を有しており、Arduino等を用いてMCU側の組込みソフトウェアを自作して各種センサを組み合わせ、シリアル通信でPC側とデータ交換（IOモジュール）を実現すれば、センシング機器を自作して計測システムを構築することが可能である[8]。しかし、MZ Platformのすべてのユーザが、MCU側の組込みソフトウェアを自作することは現実的ではない。そこで、現在はIOモジュールとしてArduinoの利用を前提に、より多くのユーザがセンシング機器を自作できるようにすることを目指している。Arduinoは独自のソフトウェア開発環境が無償で配布されており、その環境に付属するサンプルのソフトウェア（StandardFirmata）を書き込むと、PC側と特定のプロトコル（Firmata）で通信可能な状態となる。そのプロトコルで通信する機能をMZ Platform側でコンポーネントとして提供することで、組込みソフトウェアを自作することなく、PC側のMZ Platformアプリケーションを修正するだけで独自の計測システムを構築することが可能となる。ただし、現状この方法でIOモジュールが扱える入力は単純なデジタル入力とアナログ入力のみであり、I2CやSPI等は対象外である。なお、Arduinoはさまざまなバリエーションがあるが、PCとの接続を有線とする場合はArduino Nano、無線とする場合はXBeeと組み合わせたArduino FIOで動作確認をしている。

スマート製造ツールキットでは、IOモジュールからデータを取得して処理を行うMZ Platformアプリケーションのサンプルを提供する。このアプリケーションでは、デジタル入力とアナログ入力から得られる電圧値をサンプリング間隔に応じた時系列データとして取得して表示する。取得した値は意味のあるデータに変換してからサーバ側に送信して例えばデータベースに蓄積することや、データに間引きなどの処理を加えてから送信することも可能である。また、取得した時系列データの変化に対して閾値をルールとして指定することで、特定の状態認識結果を送信することも可能である。ここでは、これらの処理をまとめてエッジ側データ処理と呼ぶ。このアプリケーションの利用方法としては、非専門家のユーザが例えばノートPCとIOモジュールおよびセンサを現場に持ち込んで、計測対象とセンサ出力の関係をPC上で確認しながら、システム構成を検討することを想定している。計測対象とセンサ出力の関係が把握できたら、どのようなデータを活用するか検討して、それに応じたエッジ側データ処理をカスタマイズすることになる。

このようにして自作した計測システムを現場に導入する場合、通常のPCを設置して運用することもできるが、より安価で小型のPCを設置したい場合はRaspberry Piを使用することも可能である。これまでに、MZ Platformの実行環境はWindows版とLinux(x86系)版に加えて、Raspberry Pi版を作成して配布している。この実行環境を導入したRaspberry Piに同じアプリケーションを配備すれば、通常のPCと同様に稼働させることができる。

## 4 中小製造業の現場への適用事例

### 4.1 適用先の概要

スマート製造ツールキットの要素である自作機器を用いた計測システム構築の適用先として、鳥取市に金属プレス工場を有する株式会社田中製作所を対象に、機器の試作と実証実験を実施した。図5に示すように既存の金属プレス加工機が10台一列に配置されており、これに対して自作機器を後付けすることでプレス機の稼働実績を自動収集する。なお、本実験は株式会社レクサー・リサーチと産総研による共同研究の一環として行われた。



※製品イメージ写真はホームページ (<http://www.heart-tanaka.co.jp/>) より転載

図5：適用先工場内の様子と製品イメージ

### 4.2 小型 PC を用いた試作機器

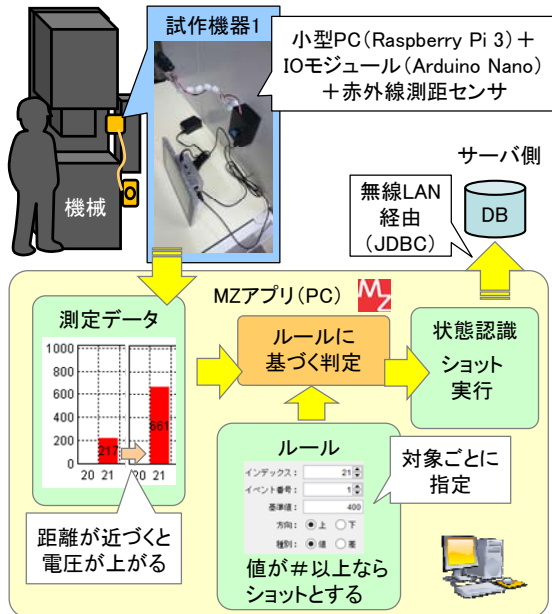


図6：小型 PC を用いた試作機器 1

事前実施した予備実験の結果に基づき、適用先の金属プレス加工機を対象とする実績収集用の試作機器1を作成した。その構成を図6に示す。基本部分は図4に示す構成であり、小型 PC の Raspberry Pi

3と有線接続の Arduino Nano を用いて、赤外線測距センサでプレス機上部の動作を検出する。図7に示す位置に設置することで、計測対象部の上下動によって赤外線測距センサから側面までの距離が変化し、アナログ入力 of 電圧値が変化する。試作機器1では、PC側で動作する MZ Platform アプリケーションで設置状況に応じた閾値を決め、電圧値が閾値を越えたらプレス機のショット実行がなされたと判定するルールを設定した。さらに、ショット実行が認識された際にデータを送信し、サーバ側のデータベースに認識結果が格納されるようにした。



図7：計測対象とセンサ位置の関係

### 4.3 PC なしに簡略化した試作機器

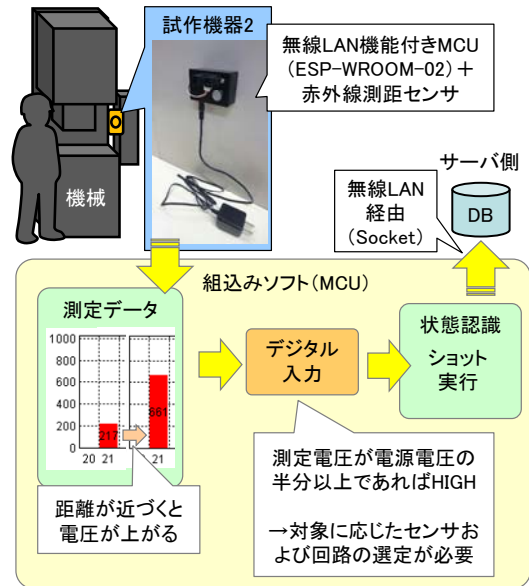


図8：PC なしに簡略化した試作機器 2

試作機器1で実証実験は可能だったが、実験の詳細を調整する段階で、1ライン分の10台全部に設置することになり、より実運用に近い形で使用可能な状態まで簡略化した試作機器2を作成した。その構成を図8に示す。こちらはPCなしで単独動作するMCU (ESP-WROOM-02) を採用し、単体で状態認識と無線LAN経由のデータ送信を実現している。セ

ンサと計測対象は試作機器 1 と同様であるが, センサの出力が MCU のデジタル入力に接続されていて, 電圧値が電源電圧の半分以上であれば HIGH と判定されることを利用している. この方法ではセンサの検出範囲と計測対象の設置環境との関係に依存して, 組み込みソフトウェアの自作や回路の作成が必要であるが, それを実現できれば PC なしの単体で機能を実現し, 閾値設定やルール判定を省ける. ショット実行が認識された際のサーバへの送信は試作機器 1 と同様である.

図 9 に, 試作機器 1 と試作機器 2 の運用時の違いを示す. 試作機器 1 は小型 PC を用いるため, OS 起動後に MZ Platform アプリケーションの操作が必要であり, 終了する場合にはアプリケーションの終了や OS シャットダウン操作が必要である. 一方, 試作機器 2 は MCU 単体の組み込みソフトウェアで動作しているため, 電源の接続と切断のみとなる.

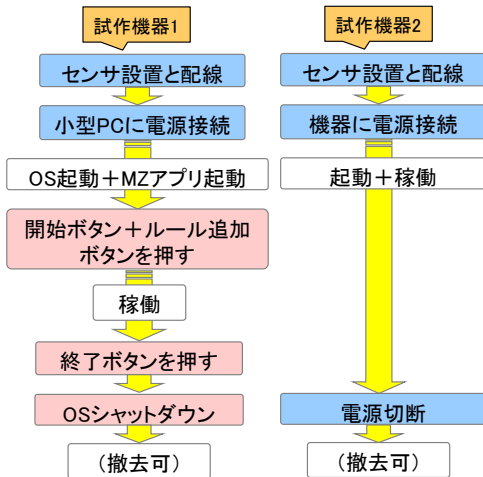


図 9 : 機器の簡略化による運用への効果

#### 4.4 実証実験

実証実験では試作機器 2 を使用し, 1 ライン分 10 台のプレス機を対象とする稼働実績収集を実施した. 実証実験固有部分の機器構成を図 10 に示す. サーバ PC は Raspberry Pi 3 を用いた. 試作機器 2 の MCU は無線 LAN 接続の設定とサーバ PC の IP アドレスを組み込みソフトウェア内で記述する必要があるため, 実験専用の無線 LAN ルータ (Buffalo WMR-433W) を準備して固定の SSID とキーとし, サーバ PC は常に固定の IP アドレスとなるように設定した. 10 台の試作機器 2 とサーバ PC は, この無線 LAN ルータを介してプライベートネットワークを構成する. なお, 今回採用した無線 LAN ルータは別の無線 LAN ルータやアクセスポイントへの接続設定を登録する

ことで, 社内 LAN やインターネットにも接続することが可能である.

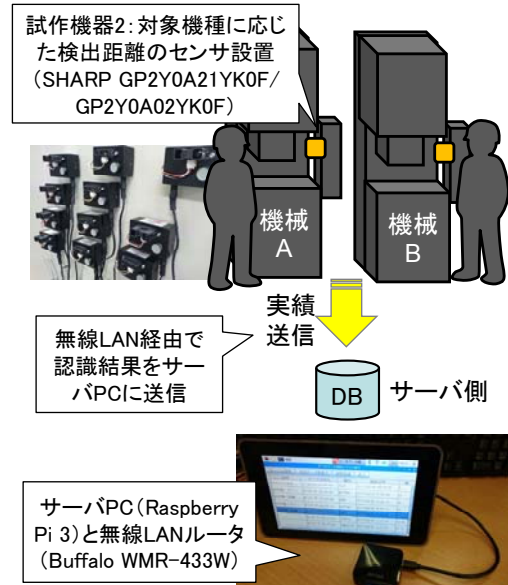


図 10 : 実証実験用の機器構成

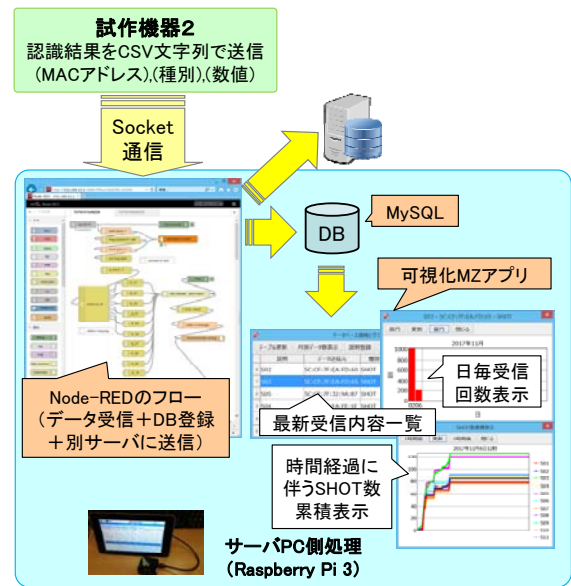


図 11 : 実証実験用のサーバ PC 側ソフトウェア

図 11 に実証実験用に準備したサーバ PC 側のソフトウェア構成を示す. データを蓄積するデータベース (MySQL) と, Node-RED のフロー, および可視化用の MZ Platform アプリケーションを配備した. 試作機器 2 からのデータ送信はソケット通信となるため, 直接データベースにデータを登録することができない. 今回はサーバ上で JavaScript のプログラムが動作する Node.js をベースに, 主に図形で直観的に処理が記述できる Node-RED でデータ受信からデ

データベース登録までの処理を定義した。

図 12 にデータ収集結果の可視化例を示す。この可視化では、ある 1 時間での各プレス機のショット数の累積回数を、時間経過に対する折れ線グラフとして示している。折れ線の傾きが急であれば短時間で多くのショット数が計測されていて、一作業のサイクルタイムが短いことを示す。逆に、折れ線の傾きが小さければサイクルタイムが長いことを示す。完全に水平になっている期間はショット数が増えない非稼働状態を示す。稼働・非稼働の境界が明確になっていて、稼働中は一本の直線になっていけば作業は特に問題ないと考えられるが、傾きの途中で水平ラインが時折見られる場合や、傾きが一定でなく直線になっていない場合は、工程や作業者の技能に改善の余地がある可能性を示していると考えられる。

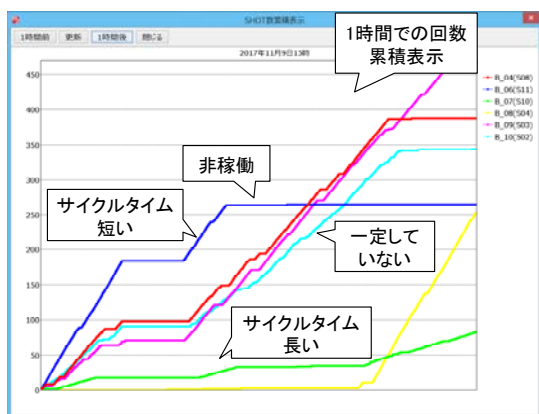


図 12 : データ収集結果の可視化例

#### 4.4 考察

今回の実証実験では、PC なしの試作機器 2 を用いて、現場での実運用に適した稼働実績収集が実現できた。一方で、MCU 側で事前に組み込みソフトウェアを作成しているため、現場での状況の変化や環境の違いに対応した柔軟な変更は難しい。さらに、計測対象とセンサ出力の関係が把握できていない状況では、このような機器を作成することは不可能であるため、試作機器 1 のように PC を用いた機器の作成や予備実験は必須である。PC と MZ Platform アプリケーションであれば、現場での状況変化や環境の違いにソフトウェア側で柔軟に対応することが可能である。つまり、本稿で示したようなシステム構築をエンドユーザ開発で実施する場合には、まず PC ベースで計測対象とセンサ出力の関係を把握し、期待する動作が実現できるかどうかの確認をすることが必要である。現場での実運用上の問題がなければそのまま使用し、実運用上必要である場合はさらに PC

なしの機器作成に進むと良い。また、データ可視化に関しては、対象に応じた可視化方法を個別に検討する必要があるため、こちらもスマート製造ツールキットの提供範囲の要素ではあるが、カスタマイズ可能なソフトウェア資産の配布と自習用のコンテンツ提供が重要であろう。

## 5 むすび

本稿では、IoT 技術を活用して中小製造業における現場活動実績の自動収集と可視化を実現する取り組みについて報告し、エンドユーザ開発の観点から考察した。MZ Platform を拡張したスマート製造ツールキット開発は途上であるが、このような技術とツールの提供によって、現場の活性化に多少なりとも貢献できれば幸いである。

## 謝辞

本研究の一部は、株式会社レクサー・リサーチと産総研の共同研究の一環で実施され、その実証先として株式会社田中製作所にご協力をいただいた。また、当該共同研究は NEDO 事業「ロボット活用型市場化適用技術開発」の助成を受けている。この場を借りて、関係各位に感謝の意を表する。

## 参考文献

- [1] Sutcliffe, A., Mehandjiev, N.: End-User Development, CACM, 47(9), pp.31-32, 2004.
- [2] Lieberman, H., et al.(ed.): End-User Development, Springer, pp.31-32, 2004.
- [3] 日経コンピュータ編: 開発・改良の切り札 システム内製を極める, 日経 BP, 2011.
- [4] ICT 経営パートナーズ協会: 超高速開発が企業システムに革命を起こす, 日経 BP, 2014.
- [5] 古川ら: 業務システムを対象としたエンドユーザ開発支援の課題と展望, 2014 年度人工知能学会全国大会(第 28 回)論文集, 1L4-NFC-05a-3in, 2014.
- [6] 澤田ら: 高度な専門知識不要の IT システム開発ツール: MZ Platform—製造業におけるエンドユーザー開発の実現—, Synthesiology, 8(3), pp.158-168, 2015.
- [7] 古川: 現場作業者を支援する活動実績可視化基盤の構築, 2017 年度人工知能学会全国大会(第 31 回)論文集, 2F4-NFC-03b-1, 2017.
- [8] 古川ら: 業務システムのエンドユーザ開発とフィジカルコンピューティング, 2015 年度人工知能学会全国大会(第 29 回)論文集, 14-NFC-05a-1, 2015.