

IoT を活用した生産工程の可視化に関する研究（第 3 報）

生産技術部

福島章吾 田中徹 中野太郎

大坪昭文

本研究は、IoT 技術を利用して生産工程のデータを収集するシステムを構築し、各種センサノードから取得した製造工程のデータ（作業環境の温度、湿度等）と製品品質の検査結果を対比して管理する「生産工程の可視化」を目的としている。本年度は、製造工程のデータと製品品質の検査結果を関連付けるため、工程管理の情報をデジタル化する仕組みの検討を行った。その結果、製造ロット毎の工程管理票（紙媒体）に基本情報を付加したバーコードを設けて、各作業工程でそのバーコードを読み取ることで、作業内容や作業時間等を工程管理のデータとして取得できた。また、取得したデータの可視化方法を検討し WEB アプリを作成したことにより、同一ネットワーク内にある任意のクライアント（PC 等）のブラウザから整理・加工された情報を閲覧できた。

1. はじめに

県内製造業の生産工程では、製造に係る諸条件や生産管理数量、検査結果等の数多くの情報が収集・管理されている。しかしながら、これらの情報が他部門と共有されていないために、製品の不良原因等に対して具体的な対策を取れない場合がある。

一方、近年、身の回りのモノ一つ一つがインターネットに代表されるネットワークに繋がり、様々な情報（数量、環境、位置等）を収集・活用できる Internet of Things（以下 IoT）技術が注目されている¹⁾。さらに、IoT 向けのデバイス（シングルボードコンピュータ、無線モジュール等）が普及し、IoT を手軽に導入できる環境が整ってきている。

県内製造業においても、IoT 技術を活用して、各部門で収集・管理している情報とこれまで取得が難しかった情報を関連付けて一元的に管理し、製品品質を安定させたいという要望がある。

そこで、本研究では、IoT 技術を利用して生産工程のデータを収集する計測管理システムを構築し、各種センサノードから取得した製造工程のデータと製品品質の検査結果を対比して管理する「生産工程の可視化」を目的に取り組む。

前年度までの研究により、図 1 に示すように「データベースサーバ」、「ゲートウェイ」及び「複数のセンサノード」で構成される計測管理システムを製作し、モデル企業の生産現場における作業環境の温度や湿度等の製造工程のデータを、専用インターネット回線を通じて当センターのデータベースへ安定

的に取得できることを確認した。また、モデル企業の製品の品質検査では全数検査が行われており、規格の範囲内に収まっているか専任の検査員が性能試験を併用して合否判定を行っている。そのため、技術継承という課題があり、検査員のスキルに頼らずに個々の製品毎の品質を定量化できる検査方法を提案してきた²⁾。

本研究では、製造工程のデータと製品品質の検査結果を対比して管理することを目的としているが、現状では工程管理の情報がデジタル化されておらず、関連付けもされていないため、製品検査で不良となった場合に、その製品の製造工程における正確な製

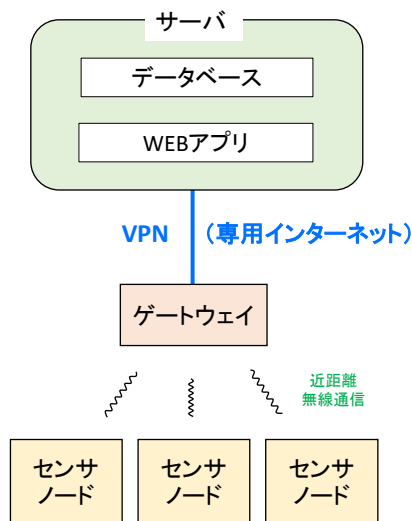


図 1 計測管理システムのイメージ

造状況が特定できない。そこで、本年度は、図 2 に示すように製造工程のデータと製品品質の検査結果との関連付けを行うため、工程管理の情報をデジタル化する方法を検討し、バーコードによるデータ収集を行った。次に、収集できるようになった製造工程のデータと工程管理のデータを利活用するため、WEB アプリによる可視化を行った。

2. 工程管理の情報のデジタル化

2.1 デジタル化方法の検討

製造工程のデータと製品品質の検査結果を関連付けるためには、作業内容と作業時間の情報をデジタル化して計測管理システムのデータベースに保存しておく必要がある。多くの企業の生産現場では製品をロット単位で工程管理票（紙媒体）を付与して管理することが行われており、各工程の作業終了後に担当者が 1 日分の作業内容を記録しているため、記入漏れ等が起きやすく正確さに欠ける場合がある。

モデル企業においても、ロット単位で製品を生産しており、各ロットに工程管理票が付与されている。工程管理票には、品番（種類、規格、色）、製作数等、そのロットに関する基本情報と作業内容が記されており、各工程の作業終了後に担当者が結果を手書きで記録している。

そこで、ロット単位の作業内容、作業時間、担当者等の工程管理の情報を手軽にデジタル化する手段として、バーコードを利用してデータベースへ入力する方法を採用する。

2.2 デジタル化の仕組みの構築

前述した工程管理の情報をデジタル化するため、ロットに関する基本情報にロット番号を付加したバーコードを作成し、図 3 に示すように各作業工程でそれぞれの作業を開始又は終了する際にバーコード

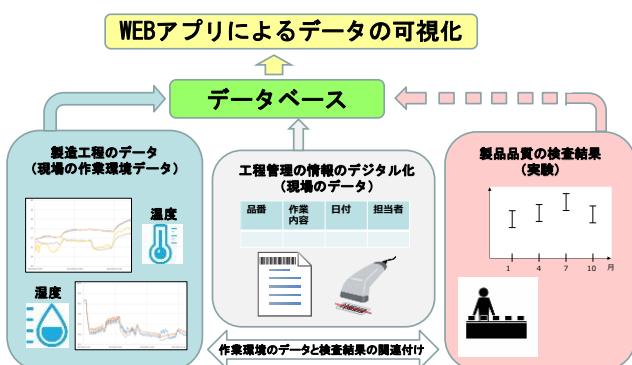


図 2 研究内容のイメージ

リーダーで読み取ることで、基本情報、作業内容、作業時間、担当者等の情報を取得する。取得した情報は、工程管理のデータとしてデータベースに蓄積することで利活用する。

(1) 工程管理の情報のデジタル化方法

生産現場の各工程においてバーコードで読み取った情報をデータベースに蓄積するため、計測管理システムにおける製造工程のデータ取得と同様に、ZigBee 規格の無線通信を使用する。図 1 に示した計測管理システムの、センサノードの一つとして、工程情報取得用バーコードリーダー（以下バーコードリーダー）を用い、ゲートウェイには、バーコードリーダーから送信されるデータを専用に取り取るシングルボードコンピュータ Raspberry Pi を用いる³⁾。ゲートウェイは、バーコードリーダーから送られてきた工程管理の情報を逐次処理して、専用インターネット回線（VPN）を経由してデータベースサーバに送信する。データベースサーバは、工程管理用のデータベースにそのデータを蓄積する。

(2) バーコードリーダーの設計・試作

バーコードリーダーに必要な要件は、生産現場に複数台設置可能であり、以下の仕様を満たすことである。

- ・電源を入れ即座に使用できる
- ・プログラムでバーコードスキャナを利用できる
- ・ZigBee 無線によりデータを送信できる
- ・送信前に読み取った内容を確認できる

上記の仕様を満たすため、USB バーコードスキャナと ZigBee 規格無線モジュール、液晶表示器を組み合わせ、これらをプログラムで制御するための汎用マイコンボード（Arduino）をメインに回路を設計する。

マイコンボード（Arduino）には、接続して利用で

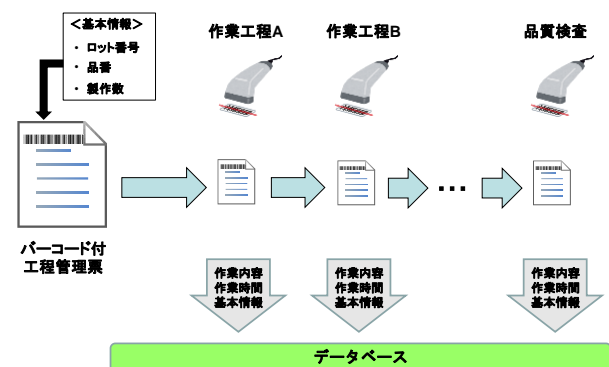


図 3 バーコードによる工程管理の情報の取得

きる周辺機器と制御するためのプログラムモジュールが多数提供されている。ここでは、USB バーコードスキャナ (PChero 製 X000D9R98J)、ZigBee 規格無線モジュール (Digi International 製 DIGI- XB24-CZ7WIT-004)、液晶表示器 (Sunlike Display Tech. Corp. 製 SC2004CSWB-XA-GB-G) を、これらを接続するために、USB ホスト機能付き Arduino (Arduino Holding 製 Arduino M0 Pro) と無線モジュール接続用シールド (Arduino Holding 製 ARDUINO WIRELESS PROTO SHIELD) を選定した。また、制御用プログラムの作成には、公開されているサンプルプログラムを参考にした⁴⁾。試作したバーコードリーダーを図 4 に示す。電源を入れてバーコードを読み取ると、順次液晶表示器にバーコードの情報が表示される。内容を確認して問題がなければ送信する。

(3) 工程情報取得の実証試験

前述したデジタル化の仕組みをモデル企業の生産現場で試用した。モデル企業では、工程管理票を Microsoft Excel を用いて作成している。そこで、工程管理票の作成時にロットの基本情報 (品番、製作



図 4 試作したバーコードリーダー

送信		
工程コード		
工程A		
工程B		
	:	
担当者コード		
担当A		
担当B		
	:	

図 5 設定用バーコード

数等) を入力すると、Excel 関数によりそのバーコー

ドが自動的に作成され、CODE39 のバーコードフォント⁵⁾として工程管理票の Excel シートに追加されるようにした。

また、図 5 に示すように作業工程名、担当者名、送信用等の共通する項目をそのバーコードとともに別途印刷しておき、作業者は決められた順番でバーコードを読み込むだけとする。共通項目はマイコンボード (Arduino) のプログラムテーブルに記述しておくことで、入力されたバーコードを元に参照し、その内容が引き出されて表示される。

今回、設計・試作したバーコードリーダーとゲートウェイをモデル企業の生産現場に設置し、特定の製品を対象として実証試験を行った。その結果、工程管理のデータを取得し、デジタルデータとしてデータベースに保存できることを確認した。

3. 収集したデータの可視化

3.1 可視化方法の検討

生産現場における工程管理のデータをデジタルデータとして収集できるようになったことから、製造工程のデータ (作業環境の温度、湿度等) と関連付けて、製造過程における作業環境と製造状況を把握するため、取得したデータの可視化方法を検討した。可視化方法として、誰でも閲覧できるようにするため、クライアント側の既存のブラウザとサーバを利用した WEB アプリを採用し、可視化画面を作成することとした。WEB アプリを用いることで、図 6 に示すように同一ネットワーク内にある任意のクライアントのブラウザから特定の URL にアクセスすることで、整理・加工された情報が閲覧可能となる⁶⁾。

WEB アプリの開発環境として、Visual Studio と

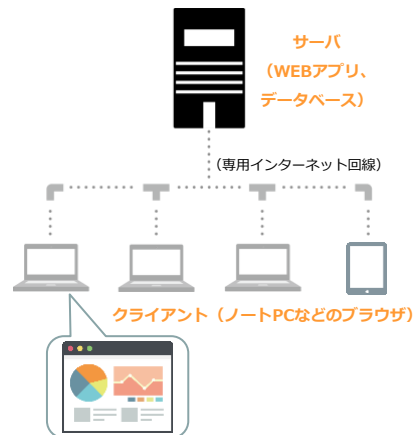


図 6 WEB アプリへのアクセスのイメージ

Node-red の 2 種類を検討した。Visual Studio は、用意されている数十種類の部品の中からグラフやボタン等目的の機能に適した部品を選択して配置し、各部品の詳細設定をすることで可視化画面を作成する。細かい設定を反映させた可視化画面を作成することができるが、設定箇所が見つげにくいいため、頻繁に使用する簡単なグラフ表示や検索機能を実装するだけでも時間を要する。一方、Node-red は、各種機能を持ったノードを接続し、各ノードの設定をすることで可視化画面を作成できる。簡易的なグラフ表示や検索機能等、収集したデータの可視化が Visual Studio と比較して容易である。したがって、県内企業への普及のしやすさを考慮し、WEB アプリの開発環境として Node-red を採用した。

3.2 可視化画面の作成

製造工程のデータと工程管理のデータを用いて図 7 に示す可視化画面を作成した。この画面は、これまで工程管理票（紙媒体）で管理されている工程管理の情報をデジタルデータとして取得できるようにすることで実現した。本画面で任意の工程管理のデータを選択すると、その作業日時における環境温度や湿度の情報が表示されるようになっている。全ての生産工程から工程管理の情報を取得することがで

きれば、工程管理票のデジタル化が可能となる。

可視化画面を Node-red を用いて作成したフローチャートの一例を図 8 に示す。Node-red には、必要な機能がない場合に、独自機能を自作するためのノード（function ノード）が用意されており、JavaScript 言語を用いて、目的の機能を記述することができる。本可視化画面の作成においては、①入力（ロット番号、工程名）があった場合にデータベースから作業日時の製造工程データを引き出す SQL クエリを生成する、②引き出したデータをグラフ表示用のデータに変換する等の処理を function ノードで作成した。

図 8 のフローチャートは以下の処理を示している。ロット番号または工程名が選択されると、時刻情報がデータベースから取り出される。その時刻情報を元に作業環境データをデータベースから取り出し、グラフに表示させる。

モデル企業の生産現場において、可視化画面を作成した結果、企業内のモバイルルータに繋がるネットワークに接続した端末のブラウザから閲覧できることを確認した。



図 7 工程管理のデータの可視化画面

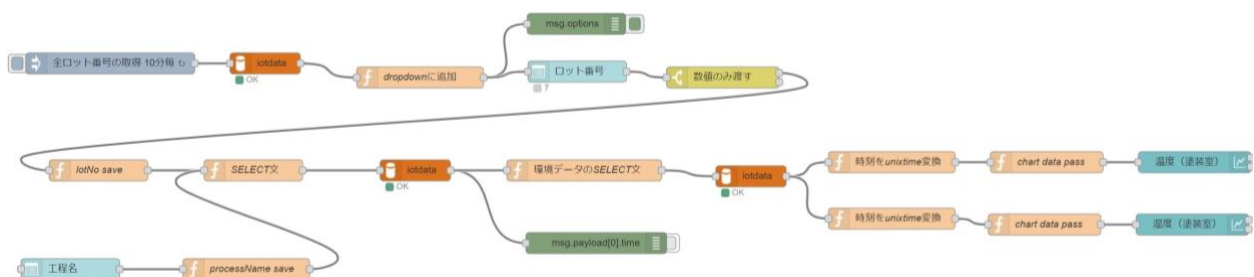


図 8 Node-red のフローチャート

4. おわりに

製造工程のデータと製品品質の検査結果をデジタルデータとして関連付けるために、製造ロット毎の工程管理票にバーコードを設けて、各作業工程でそのバーコードを読み取ることで、作業内容や作業時間等を工程管理のデータとして取得できた。また、取得したデータの可視化方法の検討を行い、Node-red を使用して整理・加工（グラフ化、表形式、検索等）した情報を可視化する WEB アプリを作成し、企業内のモバイルルータに繋がるネットワークにある任意のクライアント（PC 等）のブラウザから可視化画面を閲覧できることを確認した。

生産工程における作業環境や作業時間等の様々な情報をデジタル化できたことで、製品品質の安定化や生産工程の効率化、生産コストの削減等を検討する上で、有益な情報が蓄積できたと考えられる。しかし、本研究を進めるにあたり計測管理システムの構築は、情報セキュリティの観点から企業の社内 LAN を使用できなかつたため、専用インターネット回線（VPN）を利用して行った。今後は、企業の社内 LAN に導入できるように情報セキュリティを考

慮してサーバを構築すること、また、製品品質等の検査結果を効率的にデジタルデータとして取得する方法を含めて計測管理システムの実用化を検討する必要がある。

参考文献

- 1) 河村雅人, 大塚敏史, 小林佑輔, 小山武士, 宮崎智也, 石黒佑樹, 小島康平 : IoT/センサの仕組みと活用, 翔泳社 (2015)
- 2) 福島章吾, 田中徹, 中野太郎, 大坪昭文 : IoT を活用した生産工程の可視化に関する研究 (第 2 報), 平成 29 年度佐賀県工業技術センター研究報告書, No.26, pp.59-63, (2017)
- 3) <https://www.raspberrypi.org/>
- 4) https://github.com/felis/USB_Host_Shield_2.0
- 5) <https://www.technical.jp/barcode/font/>
- 6) 福島章吾, 田中徹, 中野太郎 : IoT を活用した生産工程の可視化に関する研究 (第 1 報), 平成 28 年度佐賀県工業技術センター研究報告書, No.25, pp.45-48, (2016)