

Desenvolvimento de um *Data Logger* para acompanhamento de variáveis de processo

Cristiano Sbabo e Ivandro Cecconello

Resumo

Data loggers comerciais possuem um custo elevado comparados a modelos Open Source. O objetivo do trabalho é criar um protótipo de coletor de dados de baixo custo, que possa coletar e registrar variáveis de processo, como temperatura, umidade e sinais de máquina. O coletor foi desenvolvido com a utilização da placa de prototipagem rápida Arduino, principal componente do sistema. O *data logger* pode ser conectado a uma máquina para coleta e seus dados são transmitidos ao display remoto que pode ser ligado a um PC para registro dos mesmos. Como resultado, dados registrados podem ser analisados através de planilha eletrônica. A utilização de display remoto traz ao protótipo uma novidade, visto que os coletores comerciais não possuem essa opção. Diante dos resultados, foi possível criar um coletor de dados robusto, de baixo custo e grande flexibilidade de programação para uso na coleta de diversos sinais analógicos e digitais.

Palavras-chave

Indústria 4.0. Data Logger. Open Source. Internet das Coisas.

Data Logger development to monitor process variables

Abstract

Commercial data loggers are expensive compared to Open Source models. The objective of the work is to create a low cost data collector prototype, which can collect and record process variables, such as temperature, humidity and machine signals. The collector was developed using the Arduino rapid prototyping board, the main component of the system. The data logger can be connected to a machine for collection and its data is transmitted to the remote display that can be connected to a PC to record them. As a result, recorded data can be analyzed using an electronic spreadsheet. The use of remote display brings a novelty to the prototype, since commercial collectors do not have this option. In view of the results, it was possible to create a robust, low-cost data collector with great programming flexibility for use in the collection of various analog and digital signals.

Keywords

Industry 4.0. Data Logger. Open Source. Internet of Things.

I. INTRODUÇÃO

O surgimento da Indústria 4.0 traz o conceito de integração entre sistemas, tecnologias e coleta de informações para apoio às tomadas de decisão e respostas críticas [1]. A *Industrial Internet of Things* (IIoT) permite a integração entre muitas dessas tecnologias, como suporte da nuvem para diferentes aplicações na indústria [2]. A adoção dessas tecnologias está associada aos benefícios como a fabricação de produtos altamente customizados por fábricas inteligentes, comunicação Máquina a Máquina (M2M) e sistemas físico-cibernéticos [3].

A gestão das operações de manutenção de equipamentos na indústria torna-se complexa à medida que as companhias crescem e ampliam suas operações, seus ativos e se integram aos conceitos de produção *Lean*. Uma das sete grandes perdas nos sistemas de produção industriais é a perda de velocidade, onde o equipamento tem mau funcionamento temporário ou trabalha em marcha lenta; da mesma forma, a diferença entre o projeto do equipamento e a velocidade operacional [4,5]. Por

isso, da importância da coleta de dados em tempo real do equipamento e uma pronta resposta sobre essas perdas. Na área de manutenção, os indicadores de desempenho mais utilizados são o *Mean Time Between Failures* (MTBF) e o *Mean Time to Repair* (MTTR). No primeiro caso, mensura-se o tempo entre a ocorrência de falhas em um equipamento, muito utilizado para indicar a confiabilidade de um sistema. No segundo caso, o tempo empregado para realizar o conserto e o mesmo voltar às condições anteriores à falha [6].

O monitoramento de variáveis de um processo industrial lida com a coleta, registro e transmissão de dados medidos para um computador, para armazenamento em arquivo e análise externa do desempenho. Esses dados são amostrados em função do tempo decorrido de coleta [7].

Data Loggers comerciais requerem um investimento alto, comparados a um sistema *Open Source*, e não permitem uma flexibilidade, personalização e adaptabilidade para cada aplicação [8]. A contribuição deste trabalho diz respeito a desenvolver um sistema de baixo custo para empresas de

pequeno porte, servindo como opção diante de modelos comerciais.

Nesse trabalho, pretende-se desenvolver um *Data Logger* com linguagem livre de programação e baixo custo, como protótipo para utilização no acompanhamento de variáveis de processo, como, por exemplo, temperatura, unidade, pressão e vazão de fluídos, tempos e sinais de ciclo, através do uso de software e hardware livre [9], servindo como ferramenta para análises de falhas, acompanhamento de variáveis específicas no desenvolvimento de novos produtos nas linhas de fabricação, a fim de entender sua influência na qualidade do produto final. Em resumo, o objetivo desse artigo é desenvolver e validar um protótipo de *Data Logger* para utilização na indústria.

II. REFERENCIAL TEÓRICO

A. Indústria 4.0

A Indústria 4.0, como iniciativa alemã, visa criar fábricas inteligentes, onde as tecnologias de fabricação são atualizadas e transformadas por sistemas cyber-físicos, internet das coisas e computação em nuvem e tomar decisões inteligentes através da comunicação em tempo real e cooperação entre humanos, máquinas, sensores e sistemas em geral [10]. As principais tecnologias da Indústria 4.0 são a produção inteligente, produção habilitada para IIoT, modelo de computação em nuvem, virtualização e tecnologias orientadas a serviços [10,11]. Essas tecnologias englobam nove pilares: robôs autônomos, simulação, integração vertical e horizontal, segurança cibernética, computação em nuvem, manufatura aditiva, realidade aumentada e *big data* [12,13].

Diante do conceito de indústria 4.0, três tipos de integração são observados: integração horizontal, vertical e de engenharia ponta a ponta. Na primeira, ocorre a integração entre empresas relacionadas, modelos de negócios. Na segunda, integração vertical, ocorre a integração entre processos e sistemas da empresa, máquinas e redes inteligentes integradas, fornecendo informações massivas à rede [3,14,15]. A integração de engenharia ponto a ponto une a cadeia produtiva de um produto, desde fornecedores, design, produção, manutenção e reciclagem [1,16].

Dentro da IIoT, hoje, apenas alguns sensores e máquinas fazem uso de computação embarcada e estão organizados em uma pirâmide de automação vertical para integração com um sistema mais abrangente de controle de processo [12]. Isso permitirá a interação, permitindo controle em tempo real e troca de dados de processo [11].

B. Real Time Data Acquisition

A aquisição de dados em tempo real é primordial na tomada decisões, no âmbito da indústria 4.0. Aplicações em diversas áreas estão em estudo, como na engenharia agrícola, através de monitoramento de temperatura, umidade, e velocidade do ar de ventilação [17]. Assim, monitorando as variáveis de processo, pode-se extrair análises de dados para melhorias de processo, ou ainda, os sistemas podem tomar decisões a partir do conjunto de dados e condições mensurados [18].

A aplicação em controle de sistemas solares [19] também propõe o desenvolvimento de coletores de baixo custo, precisos e livre de fontes externas e softwares de controle

caros em comparação com o custo total do sistema. Outro ponto relevante é o acesso rápido aos dados, em tempo real.

Na indústria, embora a automação de máquinas e processos forneça suporte à coleta de dados, muitos equipamentos não estão preparados para tal proposta. Assim, a coleta, muitas vezes, precisa ser feita por sistemas paralelos, como *Data Loggers*, de acordo com necessidades específicas, como desenvolvimento de produto, processo, ou mesmo acompanhamento de processos ou sinais para manutenção [20].

C. Tecnologias de coleta e transmissão de dados

Uma rede de sensores sem fio consiste em diversos sensores identificados por nós, interface de comunicação, fonte de alimentação e módulos de rádio frequência (RF) [21]. A variedade de tecnologias sem fio existentes pode ser usada em ambientes industriais de várias maneiras. Além disso, variedade de padrões como IEEE 802.15.4, ZigBee, OPC UA, WirelessHART, Bluetooth, 6LoWPAN, WiMax e Wi-Fi podem ser usados [22]. Os sistemas IIoT e M2M se referem à presença de dispositivos identificáveis de maneira única, implantados no espaço para coletar dados e fornecer serviços inteligentes [23].

Diversos tipos de comunicadores de baixo custo podem ser utilizados, como o Zigbee e o NRF24L01 (Nordic) [10]. O uso desse módulo foi aplicado na transmissão de dados climáticos via rádio frequência com microcontroladores da família AVR, [24]. O módulo da Nordic ainda pode fornecer a maior velocidade de transmissão de dados sob a premissa de pequeno consumo de energia [25].

Para a coleta de dados e transmissão, pode-se utilizar diversas famílias de microcontroladores, como o PIC da Microchip e o AVR, da Atmel. O microcontrolador AVR ATmega328 foi instalado em uma plataforma de prototipagem rápida, conhecida como arduino, que já contém o que é esperado para auxiliar o microcontrolador, especialmente a comunicação USB com um PC [26]. O Arduino é uma boa plataforma para prototipagem porque utiliza menos solda e é amplamente utilizado no mundo acadêmico [22].

Essa plataforma de prototipagem rápida pode ser utilizada como interface entre a máquina ou sensores e a transmissão via RF até o receptor. Sua plataforma abriga 8 pinos analógicos e 14 pinos digitais para operações de entrada e saída, suportando comunicação *Inter-Integrated Circuit (I2C)* e *Serial Peripheral Interface (SPI)* [21,24].

A linguagem de programação utilizada comumente nos arduinos é a C++, devido à plataforma de programação livre do próprio fabricante de mesmo nome. Pode ser estabelecida a comunicação entre a plataforma arduino e o Excel, por exemplo, através de *Visual Basic (VBA)* [27]. Nesse caso, o sistema pode funcionar como *Data Logger*, recebendo e enviando dados à plataforma arduino, de forma direta, ou através de módulos RF e *wi-fi* [10,21,22,24, 25].

III. MATERIAL E MÉTODOS

Para a elaboração do protótipo de *data logger* proposto nesse artigo, os requisitos desejados de funcionamento são descritos na sequência, a fim de atender as necessidades para a aplicação proposta. Também será apresentada a aplicação para validação, bem como as etapas de desenvolvimento do

protótipo e os custos envolvidos.

A. Requisitos de coleta e funcionalidades

Os principais pontos na validação da criação do coletor de dados são a flexibilidade de aplicação e programação perante os coletores comerciais, bem como o baixo custo de implementação comparado a modelos de mercado.

As características principais estão elencadas na Tabela 1 e comparadas a outro modelo de coletor comercial.

TABELA 1: Comparação das características

Características	Coletor Comercial	Coletor proposto
Entradas analógicas	8	4
Entradas/ saídas digitais	8	6
Display configurável	✓	✓
Botões de interação	✗	✓
Modbus RS485	✓	✗
Envio de e-mail	✓	✗
Expansão de memória por SD card	✓	✓
Conexão USB	✓	✓
Display local e remoto	✗	✓
Comunicação via RF	✗	✓
Software configurador	✓	✓
Aquisição dos dados em planilha eletrônica	✗	✓
Teclado	✓	✗
Display remoto com funções de mestre	✗	✓

Devido ao coletor de dados ser criado utilizando código aberto, suas características de flexibilidade de aplicação são maiores em relação aos modelos de coletor de dados comerciais; seu display pode ser programado para exibir diferentes informações de dados coletados e, para cada aplicação, pode-se configurar ou criar um programa específico de coleta de dados. Muito útil na coleta e posterior análise de variáveis de processo tais como temperatura, umidade, vazão, vibração e tempos de ciclo. Essa característica de flexibilidade torna-se um ponto importante.

O coletor pode obter sinais de máquina pelas suas entradas digitais, para sinais *on/off*, ou, pelas suas entradas analógicas, pode coletar dados de sensores com nível de 0-5V de forma direta ou via uso de transdutores com essa faixa de sinal ou ainda com uso de conversores de sinal de 0-10V (faixa comumente utilizada em transdutores comerciais), para sinais 0-5V diretamente nas entradas analógicas do coletor. Outros conversores ainda podem converter os sinais de 4-20mA de transdutores para 0-5V que podem ser lidos nas entradas analógicas. Ainda, os botões do coletor podem ser configurados para ações do display, como, por exemplo, mudanças de tela e acionamentos das saídas digitais.

Quanto ao armazenamento de dados, é possível a utilização de um cartão SD para expandir o número de dados salvos. No caso e uso do SD, pode-se armazenar diretamente no coletor da máquina os dados coletados e a transmissão dos dados em tempo real para o display remoto.

Com o uso de display remoto, o mesmo pode ser conectado a um comutador e a coleta de dados pode ser feita por meio de uma planilha eletrônica, onde os dados ficam registrados. A partir da planilha os dados podem ser manipulados de acordo com as necessidades.

A fim de otimizar a ocupação da memória, a coleta de dados pode ser realizada e configurada em intervalos definidos pelo usuário na programação do coletor. Dessa maneira, define-se a frequência de registro dos mesmos.

Como o coletor possui 6 entradas/saídas digitais configuráveis no momento da programação, pode-se definir, através da leitura dos dados coletados, diferentes alarmes, acionamentos, caso, por exemplo, o valor registrado exceda um limite pré-definido. Esse alarme pode acender um sinalizador na sala de controle, por exemplo, através do display remoto. Portanto, dessa forma, o coletor pode reagir em tempo real a leitura dos dados.

B. Aplicação para validação do protótipo

O uso de coletores de dados na indústria é amplo, como já mencionado anteriormente. Para a validação do protótipo proposto, foi realizada a coleta de dados de uma máquina afiadora. Os dados coletados foram a temperatura e umidade do quadro elétrico, a fim de registro e controle, como integração da manutenção preditiva do equipamento, devido ao mesmo apresentar histórico de queima de placas eletrônicas. Portanto, com os dados torna-se possível determinar se essa causa tem origem da umidade ou temperatura excessiva. Também foram coletados os sinais de máquina operando ou parada e máquina com defeito. Esses sinais foram obtidos através das entradas digitais do coletor e com base nesses dados, pode-se calcular o MTBF, MTTR e a disponibilidade da máquina (D).

Os sinais digitais para coleta do estado da máquina provêm da comunicação com o Controlador Lógico Programável (CLP) da máquina. O coletor possui opto-acopladores que isolam os sinais externos e convertem o nível lógico de 24 V para 5V, comumente utilizado em microcontroladores. Para os dados de temperatura e umidade, utiliza-se um sensor DHT 22, próprio para essa aplicação, que transmite em uma das entradas digitais o valor lido dessas variáveis. Sua capacidade de leitura está entre -40 e 80 °C e umidade de 0 a 100%.

No display do coletor pode-se observar em tempo real os valores de temperatura e umidade lidos, bem como os indicadores de manutenção a disponibilidade. Os mesmos dados são transmitidos para o display remoto, o qual mostra os mesmos valores. A partir desse display, registra-se esses dados na planilha eletrônica com frequência programada de 10 minutos, porém, a leitura do display possui apenas o atraso da comunicação RF, que, nesse caso, pode ser considerado na faixa de milissegundos.

C. Desenvolvimento

O desenvolvimento do protótipo baseia-se no microcontrolador AVR ATmega328P. A partir dele foram

acrescentados periféricos com funções específicas. Esse microcontrolador contém o programa em linguagem C++. Ele executa o processamento dos sinais de entrada, comunicação entre os periféricos e a atualização das saídas. A Figura 1 representa o fluxo de informações entre os principais componentes do coletor, que são o sensor de temperatura e umidade, o módulo de comunicação, o display LCD e o próprio arduino, com o processador ATmega 328P.

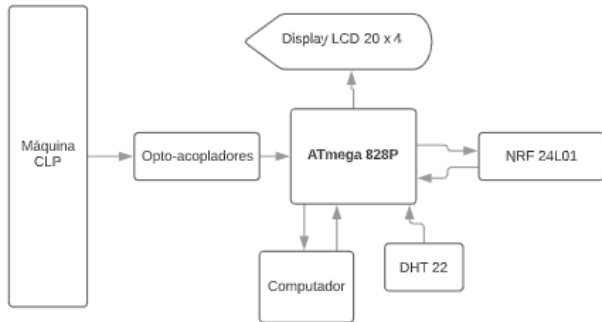


Figura 1: Diagrama de blocos do fluxo de informações

A alimentação do coletor, bem como do ATmega328P é realizada através da fonte da própria máquina, fornecendo 24VDC. O coletor possui um regulador de tensão com saída de 5VDC, para ajustar o valor de entrada da alimentação.

O display LCD utilizado no protótipo possui 4 linhas e 20 colunas para escrita de símbolos alfanuméricos. Sua alimentação provém do microprocessador e é de 5VDC. A comunicação é através da I2C, utilizando apenas 2 vias para dados, utilizando apenas 2 portas do microcontrolador. O LCD possui um conversor integrado para essa comunicação.

O sensor de temperatura e umidade DHT 22 recebe alimentação de 5VDC do microcontrolador e possui um pino de dados. Possui um sensor capacitivo de umidade e um termistor para medir a temperatura do ar. Ambos os sensores então ligados a um controlador de 8 bits, por isso da comunicação digital serial do pino de dados com o ATmega328P. Possui precisão de $\pm 0,1$ °C para leitura de temperatura e 0,1% para umidade.

A Figura 2, na sequência, representa os componentes principais do coletor.

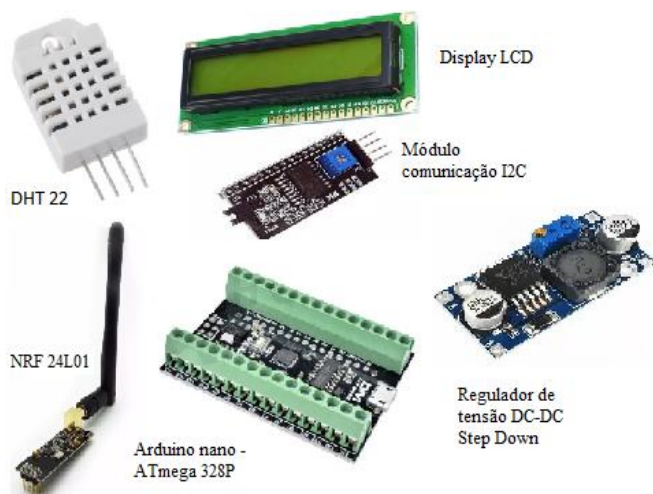


Figura 2: Componentes principais do coletor

Para a comunicação entre o coletor instalado na máquina e o display remoto, utiliza-se o módulo NRF24L01 *Wireless Transceiver*, fabricado pela Nordic. O módulo opera na frequência de 2,4 GHz, com velocidade de operação de 2Mbps. Para melhorar o alcance, o módulo trabalha com uma antena omnidirecional de 12 Dbi, de mesma frequência do módulo transmissor. Esse módulo utiliza comunicação SPI com o microcontrolador. O alcance do módulo com antena embutida pode chegar a 100 m. Com o uso dessa antena mais potente, o alcance pode chegar a 1000 m em ambientes abertos.

Para isolamento dos sinais de entrada do CLP da máquina para o coletor, foram utilizados optoacopladores nas entradas digitais do microcontrolador, o que permite o mesmo receber sinais de até 30 VDC.

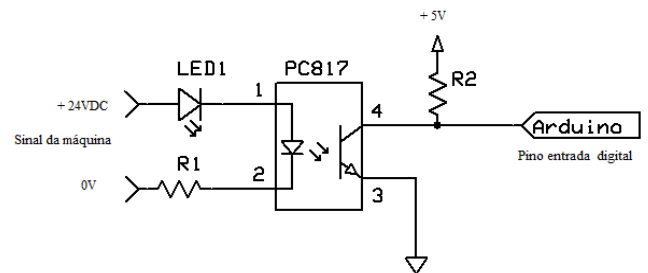


Figura 3: Esquema da isolamento das entradas digitais

Como a saída é optoisolada, o mesmo funciona como um filtro contra interferências externas, sendo que a fonte utilizada no lado do microcontrolador pode ser totalmente isolada. O pino de entrada é ligado em nível alto através do resistor R2 (resistor de *pull-up*) para evitar que o sinal fique oscilando.

Da mesma forma que o coletor de dados na máquina, o display remoto também é controlado pelo mesmo microcontrolador. A diferença está no display utilizado e na forma de alimentação, que pode ser por bateria ou pelo próprio cabo de comunicação USB ligado ao computador, sendo selecionada por uma micro chave de 2 posições. O display remoto não possui opto acopladores porque dispõe apenas de 2 entradas digitais para o uso dos 2 botões, não recebendo sinais externos. A utilização de um display gráfico OLED 0,96" dá-se ao fato de que o mesmo possui um baixo consumo de bateria e é compacto para aplicações móveis. Esse display também utiliza comunicação I2C com o microcontrolador.

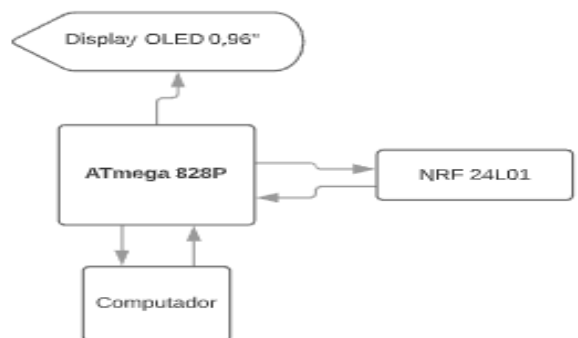


Figura 4: Diagrama de blocos do display remoto

A programação do *data logger* é realizada através da IDE do arduino, que é uma plataforma livre. Ele utiliza um código distinto do display remoto porque ambos possuem funções diferentes. O coletor calcula os indicadores de manutenção, a disponibilidade e recebe o valor de temperatura e umidade do sensor. Ele atualiza o display com os dados e registra o último valor totalizador em sua memória. Após o cálculo, ele atualiza os dados do rádio, para que o display remoto receba as informações e as exiba, bem como, se conectado a um computador, registre em uma planilha eletrônica, conforme o tempo de registro programado, como explanado anteriormente.

D. Custos envolvidos

O custo de fabricação de um protótipo torna-se superior ao do produto pronto e aprovado, porque, em geral, ajustes precisam ser feitos até o resultado esperado. Para esse protótipo, foram utilizados módulos como periféricos e os mesmos foram adquiridos separadamente, o contribui na oeração se comparado a apenas uma placa de circuito impresso integrando o sistema por completo, onde todos os componentes são comprados e é realizada a montagem. Outro ponto importante deve-se ao fato de que, quanto maior o número de unidades produzidas, mais dissolvido se torna o custo de desenvolvimento. Portanto, para esse trabalho, considera-se apenas o protótipo; o tempo de desenvolvimento integral foi considerado no custo.

Um *data logger* comercial, como o expressado anteriormente na comparação das características, custa em média R\$ 4.600,00. O equipamento proposto nesse artigo tem orçamento de R\$ 692,70, ou seja, aproximadamente 15% do custo do modelo comercial.

TABELA II: Custos do projeto

Descrição	Unid.	Custo unitário (R\$)	Custo total (R\$)
Arduino uno 328P	2	36,90	73,80
Regulador tensão DC-DC Step Down	1	12,90	12,90
Display LCD 20x 4	1	13,90	13,90
Display OLED 0,96"	1	33,90	33,90
Módulo serial I2C para LCD	1	7,90	7,90
Sensor DHT 22	1	26,90	26,90
Módulo NRF 24L01	2	32,90	65,80
Optoacopladores PC817	6	1,30	7,80
Resistores	12	0,05	0,60
Botões	4	4,80	19,20
Caixas	2	15,00	30,00
Mão de obra, montagem e desenvolvimento	20	20,00	400,00
TOTAL			692,70

IV. RESULTADOS

Com o display remoto, foi possível acompanhar a temperatura e umidade do quadro elétrico da máquina, a qual, o protótipo foi instalado. Devido ao mesmo portar bateria, foi possível transportá-lo junto ao técnico para acompanhar as grandezas medidas em tempo real, transmitidas via rádio frequência do coletor instalado junto à máquina. O sistema perdeu comunicação quando o display móvel foi levado na área externa da empresa, devido ao número de obstáculos. O uso na área interna foi satisfatório.

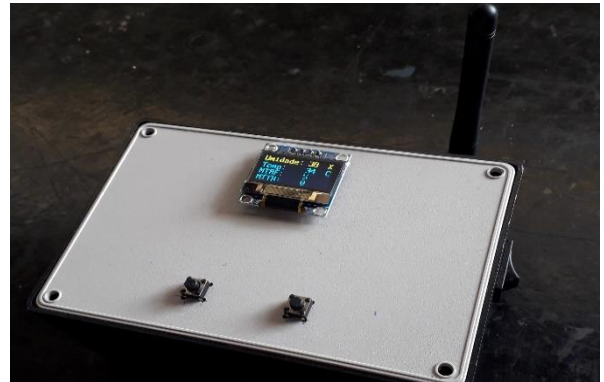


Figura 5: Display remoto do *data logger*

Os dados de temperatura e umidade foram acompanhados durante alguns dias; o consumo do módulo de comunicação, microcontrolador e display é baixo. O display mostra-se muito útil em aplicações onde é necessário acompanhamento de uma variável de processo, por exemplo, pressão, não sendo obrigatória a estadia do técnico em frente à máquina; o coletor pode ser instalado e o dado acompanhado à distância.



Figura 6: Protótipo do coletor de dados instalado na máquina

Nos primeiros testes realizados, ao desenergizar o coletor, os dados coletados haviam sido perdidos. Foi realizada uma atualização de software para correção desse problema, afim de atualizar o registro de dados a cada alteração dos mesmos. O display do coletor da máquina apresenta os dados coletados e calculados como previsto. É possível, também realizar uma média de, por exemplo, 20 coletas, para que o resultado se torne mais estável em alguns casos, para determinados sensores. Para a aplicação testada, não foi necessário esse cálculo pela média. A precisão de exibição foi determinada com 2 casas decimais, o que atende satisfatoriamente para fins de análise dos dados. O display da *data logger* pode ser visualizado na Figura 7.



Figura 7: Display do coletor de dados da máquina

Como resultado de registros de dados coletados a cada intervalo pré-determinado, pode-se manipular os dados e, por exemplo, gerar gráficos a fim de visualizar melhor as informações.

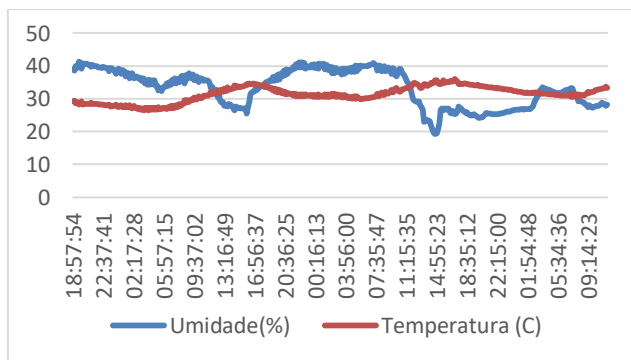


Figura 8: Dados coletados de temperatura e umidade

Os dados observados na Figura 8 foram coletados em um período contínuo de 65 horas. É possível observar como as grandezas de temperatura e umidade são inversamente

	A	B	C	D	E	F
1	Hora	MTBF (h)	MTTR (h)	Disponibilidade (%)	Umidade(%)	Temperatura (C)
2	18:57:54	41,47	0	100	38,9	29,1
3	19:07:53	41,60	0	100	38,5	29,3
4	19:17:53	41,60	0	100	39,9	28,6
5	19:27:52	41,70	0	100	39,4	29
6	19:37:52	41,70	0	100	41,2	28,3
7	19:47:51	41,70	0	100	41	28,5
8	19:57:50	42,00	0	100	39,2	29,1
9	20:07:50	42,00	0	100	40,7	28,3
10	20:17:49	42,22	0	100	40,7	28,4
11	20:27:49	42,30	0	100	40,6	28,4
12	20:37:48	42,42	0	100	40,7	28,5
13	20:47:47	42,57	0	100	40,3	28,4
14	20:57:47	42,70	0	100	40,3	28,5
15	21:07:46	42,87	0	100	39,7	28,7
16	21:17:46	43,00	0	100	40,3	28,5
17	21:27:45	43,10	0	100	40,2	28,5
18	21:37:44	43,18	0	100	40	28,5
19	21:47:44	43,42	0	100	39,9	28,4
20	21:57:43	43,52	0	100	39,6	28,4
21	22:07:43	43,68	0	100	39,5	28,3

Figura 9: Dados coletados em planilha eletrônica

proporcionais e o tempo no qual a umidade foi maior, o que poderia, em tais casos, ultrapassar os limites recomendados e afetar na vida útil do equipamento ou em placas eletrônicas do quadro de comando.

Um exemplo de coleta dos dados em planilha eletrônica pode ser observado na Figura 9, onde o coletor calcula e registra os dados de MTBF e MTTR. No período registrado não ocorreram falhas na máquina, portanto, o MTTR foi nulo e a disponibilidade 100%.

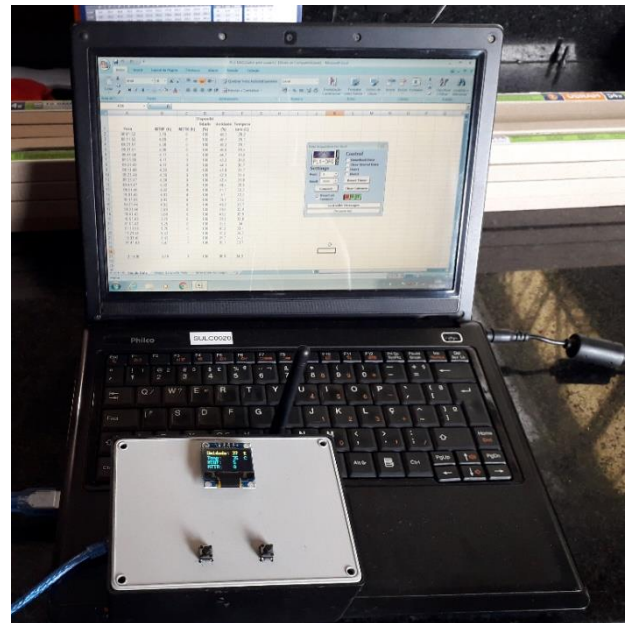


Figura 10: Display remoto registrando dados em planilha eletrônica

V. CONCLUSÕES

O presente trabalho buscou desenvolver um data logger de baixo custo para coleta de variáveis de processo. Através dos requisitos desejáveis do produto, baseado em um coletor de dados comercial, foi realizado um estudo de hardware para comportar as funções básicas de coleta, servindo como protótipo para os testes. Utilizou-se software livre para programação do mesmo, visando baixo custo e fácil implementação. Os componentes eletrônicos foram dispostos em uma caixa para ser instalada em ambiente fabril, afim de realizar os testes de coleta do protótipo. A instalação foi em uma máquina afiadora, recebendo os sinais do CLP da mesma e de sensores ligados diretamente ao coletor, gerando os dados listados na seção de resultados.

A coleta dos dados atende ao objetivo proposto, visto que os dados podem ser manipulados posteriormente de uma forma fácil via planilha eletrônica e podem ser acompanhados em tempo real no display do *data logger* ou no display remoto. O custo do coletor ficou baixo, comparado a coletores de dados comerciais e foi possível, através da programação livre, obter os mesmos dados que seriam obtidos através do modelo comercial.

O monitoramento de variáveis de processo compreende em monitorar, registrar e transmitir os dados coletados [9]. Através do uso de RF, pode-se transmitir para um computador e registrar os dados, como o uso na engenharia agrícola, para

coleta de temperatura e umidade do solo [17]. Monitorando as variáveis do processo [18], é possível realizar análises de dados para melhorias de processos. O uso de hardware livre e sistema *Open Source* na área da energia solar resultou no mesmo resultado, com controle em tempo real, registro das variáveis e possibilidade de ação perante os dados coletados [19], como no coletor proposto, embora as aplicações sejam diferentes, os mesmos resultados podem ser obtidos, visando custo, flexibilidade e facilidade de implementação.

O módulo NRF24L01 foi aplicado à coleta e a transmissão dos dados climáticos [24], onde foi obtida uma boa velocidade de transmissão de dados, com baixo consumo. Como os intervalos de coleta podem ser programados, o transmissor atendeu bem, visto o atraso da informação na casa de milissegundos. O arduino é citado em como uma boa plataforma de prototipagem porque possui engates rápidos para ligação dos fios de conexão [22]. O que se observou é que isso facilita a prototipagem, interligação dos módulos e alteração na fase de ajustes do projeto. Ainda, quanto à comunicação com planilha eletrônica, citado em [27], é possível estabelecer a comunicação utilizando a programação em *Visual Basic* com o microcontrolador, de forma a realizar a coleta dos dados em tempo real ou a cada ciclo de tempo, como programado no protótipo criado.

Na indústria, diversas aplicações do coletor foram citadas e vêm auxiliar no levantamento de dados de processos, no acompanhamento de falhas de equipamentos, sensoriamento e posterior tratamento de dados. Para indústrias alimentícias, abatedouros, alimentos refrigerados, e outros segmentos, a coleta de dados é uma exigência dos órgãos sanitários e de saúde vigentes, portanto, coletores de menor custo e fácil implementação podem tornar mais acessível a criação de relatórios de dados para envio a esses órgãos. Farmácias de manipulação, laboratórios de análises clínicas e outros estabelecimentos do segmento, da mesma forma, necessitam de uma coleta de dados rotineira; da mesma forma, o *data logger* proposto pode vir a auxiliar nessas coletas.

Durante o desenvolvimento do protótipo, algumas limitações foram observadas, como a capacidade de memória de registro de dados, caso o coletor não esteja diretamente ligado a uma planilha de coleta. A distância entre o *data logger* e o display remoto também limita o uso na coleta devido ao número de obstáculos entre eles causar a perda da comunicação. Para futuros trabalhos, a utilização de outra forma de comunicação pode ser avaliada. Outra limitação é a interface com o usuário; como sugestão de pesquisas futuras, a implementação de algumas telas de configuração rápida, para utilização de sensores padrão de mercado, o uso interno de conversores 4-20 mA ou 0-10V para 0-5V nas entradas analógicas, nível lógico utilizado pelo microcontrolador.

VI. BIBLIOGRAFIA

- [1] S. Wang *et al.*, “Implementing smart factory of industrie 4.0: an outlook”, *International Journal of Distributed Sensor Networks*, vol. 12, no. 1, pp. 3159805, 2016.
- [2] M. Aazam, S. Zeadally and K. A. Harras, “Deploying Fog Computing in Industrial Internet of Things and Industry 4.0”, *IEEE Transactions on Industrial Informatics*, vol.14, no.10, pp. 4674-4682, 2018.
- [3] L. Dalenogare *et al.*, “The expected contribution of Industry 4.0 technologies for industrial performance”, *International Journal of Production Economics*, vol. 204, pp.383-394, 2018.
- [4] P. Jonsson, M. Lesshammar, “Evaluation and improvement of manufacturing performance measurement systems - the role of OEE”, *International Journal of Operations & Production Management*, vol.19, no.1, pp. 55-78, 1999.
- [5] B. S. de Ugarte, A. Artiba and R. Pellerin, “Manufacturing execution system—a literature review”, *Production planning and control*, vol. 20, no. 6, pp. 525-539, 2009.
- [6] M. Engelhardt and L.J. Bain, “On the Mean Time between Failures for Repairable Systems”, *IEEE Transactions on Reliability*, vol. 35, no.4, pp. 419-422, 1986.
- [7] R. Mukaro and X. Carelse, “A microcontroller-based data acquisition system for solar radiation and environmental monitoring”. *IEEE Transactions on Instrumentation and Measurement*, vol.48, no. 6, pp. 1232-1238, 1999.
- [8] L. Pocero *et al.*, “Open Source IoT Meter Devices for Smart and Energy Efficient School Buildings”, *HardwareX*, vol.1, pp. 54-67, 2017.
- [9] R. A. Delgado *et al.*, “Design and Implementation of Manufacturing Execution System with open hardware”, *Direccion y Organizacion*, vol. 48, pp. 41-45, 2012.
- [10] S. Wang *et al.*, “Towards smart factory for Industry 4.0: A self-organized multi-agent system with big data based feedback and coordination”, *Computer Networks*, vol.101, pp.158-168, 2016.
- [11] R. Y. Zhong *et al.*, “Intelligent Manufacturing in the Context of Industry 4.0: A Review”, *Engineering*, vol. 3, no. 5, pp. 616-630, 2017.
- [12] M. Rüßmann *et al.*, “Industry 4.0: The Future of Productivity and Growth in Manufacturing Industries”. *Boston Consulting Group*, vol. 9, 2015. [online]. Available: https://www.bcg.com/ptbr/publications/2015/engineered_products_project_business_industry_4_future_productivity_growth_manufacturing_industries.aspx. [Accessed:10-Jan-2020].
- [13] S. Erol *et al.*, “Tangible Industry 4.0: A Scenario-Based Approach to Learning for the Future of Production,” *Procedia CIRP*, vol. 54, pp. 13-18, 2016.
- [14] D. Christmann *et al.*, “Vertical Integration and Adaptive Services in Networked Production Environments”, *Innovations In Enterprise Information Systems Management And Engineering*, pp. 147-162, 2016.
- [15] R. Besutti, V. Machado and I. Cecconello, “Development of an open source-based manufacturing execution system (MES): industry 4.0 enabling technology for small and medium-sized enterprises”, *Scientia Cum Industria*, vol. 7, no.2, pp. 1-11, 2019.
- [16] M. P. Lara *et al.*, “Vertical and horizontal integration systems in Industry 4.0”. *Wireless Networks*, 2018.
- [17] T. Camargo *et al.*, “Thermal comfort monitoring in aviaries by a real-time data acquisition system”. *Revista Brasileira de Engenharia Agrícola e Ambiental*, vol.23, no. 9, pp. 694-701, 2019.
- [18] F. Segura, V. Bartolucci and J. M. Andújar, “Hardware/Software Data Acquisition System for Real Time Cell Temperature Monitoring in Air-Cooled Polymer Electrolyte Fuel Cells”, *Sensors*, vol. 17, no.7, pp. 1600, 2017.
- [19] A. L. Vargas, M. Fuentes and M. Vivar, “IoT application for real-time monitoring of Solar Home Systems based on Arduino™ with 3G connectivity”, *IEEE Sensors Journal*, vol.1, no.1, pp.1-13, 2018.
- [20] Catálogo FieldLogger, “Fieldlogger, Registro e Aquisição de Dados”, 2020.[online].Available: <https://www.novus.com.br/downloads/Arquivos/cat%C3%A1logo%20fieldlogger.pdf>. [Accessed:14-Feb-2020].
- [21] M. Rodriguez *et al.*, “Wireless sensor network for data-center

- environmental monitoring”, *Fifth International Conference on Sensing Technology*, pp.533-537, 2011.
- [22] D. Dobrilovic *et al.*, “Testing Zigbee RF module applicability for usage in temperature monitoring systems”, *22nd Telecommunications Forum Telfor (TELFOR)*, pp. 415-418, 2014.
- [23] P. Su *et al.*, “Decentralized fault tolerance mechanism for intelligent IoT/M2M middleware”, *IEEE World Forum on Internet of Things (WF-IoT)*, pp.45-50, 2014.
- [24] R. Sidqi *et al.*, “Arduino Based Weather Monitoring Telemetry System Using NRF24L01+”, *IOP Conference Series: Materials Science and Engineering*, vol. 336, pp.12024, 2018.
- [25] Y. Wang and Z. Chi, “System of Wireless Temperature and Humidity Monitoring Based on Arduino Uno Platform”, *Sixth International Conference on Instrumentation & Measurement, Computer, Communication and Control (IMCCC)*, pp. 770-773, 2016.
- [26] V. T. Jamdar, S. B. Deosarkar and S. V. Khobragade, “An Effective Arduino Based Communication Module for Railway Transportation System”, *Second International Conference on Intelligent Computing and Control Systems (ICICCS)*, pp. 749-752, 2018.
- [27] R. Escobar and C. A. Herrera, “Low-cost USB interface for operant research using Arduino and Visual Basic”, *Journal of the Experimental Analysis of Behavior*, vol.103, no. 2, pp. 427–435, 2015.