

Plataforma de Supervisão de uma Célula Robótica

Supervisory Platform of a Robotic Cell

Allan Amaral¹, Aluno do Curso de Engenharia Elétrica, André Cunha², Aluno do Curso de Engenharia Elétrica, Prof. Dr. Vicente Idalberto Becerra Sablón³, Doutor em Engenharia Elétrica, Profa^a. Dra. Annete Faesarella⁴, Doutora em Ciência e Engenharia de Materiais.

¹Universidade São Francisco, Itatiba, Brasil, allan.matheus@hotmail.com, ²Universidade São Francisco, Itatiba, , Brasil, andre.acunha@outlook.com, Universidade São Francisco, Itatiba, Brasil, vicente.sablon@usf.edu.br, ⁴Universidade São Francisco, Itatiba, Brasil, annete.faesarella@usf.edu.br

Resumo: Apresenta-se neste trabalho a modelagem, simulação e implementação de uma plataforma de supervisão de uma célula robótica, tendo como objetivo propiciar o acompanhamento e a gestão de um determinado processo industrial, esse que pode ser acompanhado localmente ou remotamente por seus gestores, tudo isso através de um dashboard que agrupa as informações coletadas na célula pela plataforma. Essas informações são provenientes de sensores controlados por Arduino conectado à rede (IoT). Tal solução tem a possibilidade de ser ampliada para diferentes tipos de plantas industriais, porém para caráter demonstrativo, será limitada apenas à célula robótica.

Abstract– The present paper shows a modeling, simulation and implementation of a platform for supervision a robotic cell. The objective of the project is providing the monitoring and management of a industrial process, which one can be monitored locally or remotely by its management, all this equipment is showed through a dashboard that groups the information collected in the cell, by the platform. All the information comes from sensors controlled by ®Arduino, which one is connected to the network (IoT). Such a solution has the possibility of being extended to different types of industrial plants, but for demonstrative purposes, it will be limited only to the robotic cell.

Keywords-- *Supervision Platform, Robotics, IoT, Industry 4.0, Advanced Manufacturing*

I. INTRODUÇÃO

Um mundo em que objetos possam trocar informação entre si diretamente e por meio de um único protocolo de comunicação: essa realidade tem se tornado cada vez mais presente no dia a dia das pessoas e não é diferente dentro da indústria. Com tanta tecnologia presente na atualidade, a indústria caminha para uma 4^o revolução, denominada de Indústria 4.0.

O conceito de indústria 4.0 está relacionado à eliminação de desperdícios em geral e melhor aproveitamento do sistema que se tem dentro do ambiente fabril. Máquinas trocando informações entre si, realizando *setup* autônomo, enviando as mais diversas informações para um servidor, onde possa ser acessado de qualquer lugar. Essas são apenas algumas características desta nova revolução.

Em um mercado financeiro extremamente competitivo, todos os pilares da indústria 4.0 são de suma importância, para uma empresa que deseja eliminar gastos desnecessários e ainda aprimorar processos. Com a automação e robótica cada vez mais presentes dentro da indústria, o poder de comunicação e tomadas de decisões sem intervenção humana direta no processo é cada vez maior. O que também possibilita que diversas informações sejam coletadas ou inserida ao processo, sem a necessidade de ter que se locomover até a célula de interesse.

A computação em nuvem em conjunto com a automação e robótica, possibilita que processos automatizados enviem os mais diversos tipos de informação sobre o processo em tempo real. Além de manter essa atualização constante, torna-se possível salvar essa informação para ser acessada por qualquer interessado, de qualquer lugar, a qualquer hora. Sem a necessidade de ocupar espaço de memória de um servidor físico na empresa.

O objetivo geral deste trabalho consiste em desenvolver os conhecimentos sobre a Indústria 4.0. Em específico, estudar, projetar, modelar e implementar a interação entre automação, robótica e computação em nuvem aplicados em uma plataforma de supervisão de uma célula robótica, de modo a permitir acesso remoto sobre informação em tempo real do processo em questão.

II. REFERENCIAL TEÓRICO

2.1 Indústria 4.0

O conceito de indústria 4.0 que surgiu em meados de 2012 na Alemanha tendo como principal objetivo modernizar e tornar a indústria local ainda mais competitiva, deu início a uma nova forma de enxergar tanto os profissionais que atuam na indústria, que agora necessitam de conhecimentos interdisciplinares, quanto a mudanças nos sistemas integrados de manufatura que passaram a se tornar descentralizados através de armazenamento de dados em nuvens [1].

Embora tenha a Alemanha como seu país de origem, a Indústria 4.0 se difundiu e hoje está presente no mundo todo. Mesmo o Brasil, que tem uma participação atualmente de menos de 10% do setor industrial no seu Produto Interno

Bruto (PIB), tem enxergado essa nova revolução como uma oportunidade para o país [2].

Diversas áreas tecnológicas estão presentes dentro desta 4ª Revolução Industrial, de forma a reunir e analisar os dados com maior eficiência, permitindo com isso a redução de desperdícios, menor tempo de produção, maior qualidade dos produtos e então o aumento da competitividade da organização. Dentre essas tecnologias envolvidas, nove podem ser citadas como pilar para a Implementação do sistema como um todo: *Big Data*, Robótica, Simulação, Integração de Sistemas, Internet Industrial das Coisas, Segurança Cibernética, Computação em Nuvem, Manufatura Aditiva e Realidade Aumentada [3].

Cada pilar possui sua importância, no entanto para este trabalho será implementado apenas cinco dessas grandes tecnologias.

2.2 Robótica

Atualmente é comum encontrar os mais diversos tipos de robôs, com aparência de insetos, animais, carros, humanos entre outros. Este artigo contemplará apenas os Robôs Industriais.

De acordo com a *International Organization for Standardization* (ISO), por meio da norma (ISO 8373:2012) um Robô Industrial é definido como um “Manipulador multipropósito reprogramável, controlado automaticamente e programável em três ou mais eixos” [4].

Os robôs industriais demonstram-se extremamente eficazes para operações que exigem velocidade, precisão, alta carga horária de trabalho e tarefas que possam colocar em risco a saúde ou mesmo a vida de um colaborador. À medida que há um aumento no uso de robôs dentro da indústria e estes por sua vez estão se tornando dia a dia mais robustos e flexíveis, gera uma relação custo versus benefício interessante e atrativo para o empresário [5].

Os manipuladores industriais ainda podem ser subdivididos como do tipo: Articulado, Cartesiano, SCARA, Esférico e Cilíndrico e a sua aplicação determinará qual o melhor modelo a ser empregado. Através dos conjuntos de engrenagens, é possível obter um maior torque na articulação exigindo um torque significativamente menor do atuador.

2.2.1 Composição de um Manipulador Articulado

O Manipulador Articulado ou Antropomórfico, popularizado por “Braço Robótico”, possui duas articulações “ombro”, que permitem movimento de rotação em torno do eixo vertical e de elevação do membro todo, uma articulação “cotovelo” paralelo a articulação de elevação do ombro e duas ou três articulações do punho. A figura 1 ilustra um manipulador do tipo articulado [5].

Para a movimentação das articulações são necessários atuadores que podem ser do tipo pneumático, hidráulico ou elétrico, acompanhado de um conjunto de engrenagens.

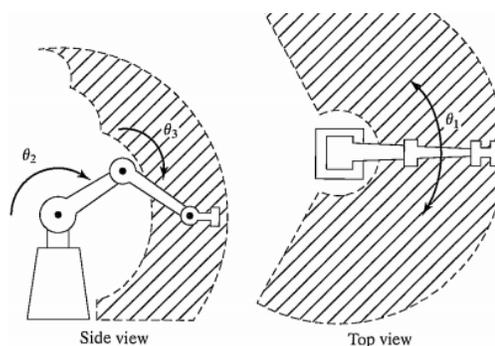


Figura 1: Ilustração da vista lateral e superior respectivamente, de um manipulador articulado e área de alcance. [5]

Porém, há uma relação inversa *torque versus velocidade*, ou seja, à medida que se ganha força se perde velocidade. Além disso, as engrenagens necessitam um cuidado especial, pois no caso de folga entre os dentes da engrenagem acarretará em posicionamento com falta de precisão, e manter as engrenagens muito justas acaba gerando perda de energia no sistema. Por isso acaba sendo utilizado, caixas de engrenagens de precisão e de tamanho reduzido. A figura 2 ilustra uma associação de um conjunto de engrenagens [6].

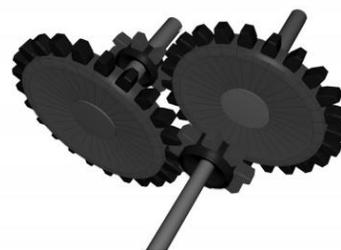


Figura 2: Associação série de dois conjuntos de engrenagens com redução 3:1 obtendo uma redução total de 9:1
Fonte: (MATARIC, 2007)

Comumente, os motores são mais utilizados do que qualquer outro tipo de atuador nessa classe de manipuladores. Embora haja diversos tipos de motores, os servomotores se destacam em aplicações em que se deseja torque, velocidade e precisão [6], [5].

Os servomotores possuem particularidades em relação a sua alimentação, o seu rotor é composto por ímãs permanentes alinhados sobre sua superfície e por um sensor que realiza a leitura da posição do eixo, além de conjuntos de engrenagens já acoplados em seu interior.

Os servomotores não permitem um controle simples como por meio de uma chave comutando diretamente o sinal de uma fonte. É necessário um dispositivo específico denominado de servoconversor, que além de enviar o sinal necessário para a movimentação do servo, também recebe um sinal de *feedback*, construindo assim uma malha fechada e eliminando a possibilidade de falhas sem detecção pelo sistema. A junção do servomotor ao servoconversor caracteriza-se por servoacionamento. A figura 3 ilustra um diagrama de malha fechada.

Informações como, velocidade e/ou posição podem ser estimadas ou medidas pelo controlador. Porém na grande maioria das aplicações são utilizados sensores para a aquisição dessas informações em tempo real. Embora existam diversos sensores, os mais utilizados para essa função são os *Resolvers* e *Encoders*, que podem ser utilizados separadamente ou em conjunto dependendo exclusivamente da necessidade da aplicação [7].

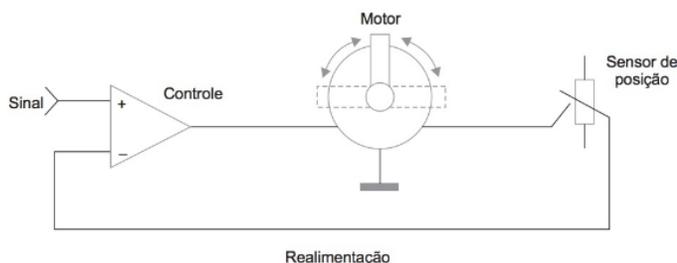


Figura 3: Diagrama de malha fechada composto por um servoconversor, servomotor de corrente contínua e sensor de *feedback* de posição do rotor.[7]

É possível dividir um servoconversor em estágio de controle e estágio de potência. O estágio de potência é responsável pelo controle do servomotor diretamente. Por meio de componentes semicondutores como Transistor Bipolar, FET (*Field Effect Transistor* – Transistor de efeito de campo) de potência, IGBT (*Insulated Gate Bipolar Transistor* – Transistor Bipolar de Porta Isolada), entre outros que atuam apenas em dois estados: em corte ou saturados, é possível efetuar o controle do motor de forma mais eficiente. No circuito de potência é de fato onde ocorre o chaveamento dos maiores valores tanto de tensão quanto de corrente elétrica. Porém esses componentes costumam ter frequência de chaveamento alta e quem é responsável por enviar os sinais necessários para que isso ocorra é o estágio de controle do servoconversor [7], [8]. O estágio de controle é responsável pela aquisição de dados, processamento e envio de sinais para o sistema de potência. Essas etapas ainda podem ser subdivididas em conversores A/D (análogo para digital), D/A (digital para analógico) e lógica de controle, que fazem uso normalmente de microcontrolador, dispositivo de memória e dispositivo de E/S (entrada e saída de dados). Algumas das funções do microcontrolador no âmbito de servoacionamento incluem, aquisição de dados a ser controlado, executar funções que foram implementadas em seu algoritmo a partir dos dados coletados e efetuar as conversões A/D dos sinais de entrada que se deseja controlar, determinação de valores de referência para modulação PWM (*Pulse Width Modulation* – Modulação por Largura de Pulso) [7].

O PWM é amplamente empregado nos mais diversos tipos de aplicação, e consiste na ideia de manter um sinal de onda quadrada com frequência constante, porém variando seu *Dutycycle* (ciclo de trabalho) de forma a controlar o valor da tensão média na saída do conversor [8].

Em teoria, tratando os transistores como componentes ideais, os controles PWM não possuem perda uma vez que não dissipam potência alguma durante os estados de chaveamento.

Quando em corte, o transistor não permite a passagem da corrente elétrica e quando em saturação permite a passagem de corrente, mas não possui queda de tensão entre seus terminais de ligação. Porém na prática, estes dispositivos não são capazes de chaveamento em tempos infinitamente pequenos e durante esses intervalos de mudança de estado, ocorre uma pequena dissipação de potência. No entanto, ainda sim essas perdas são ínfimas se comparado a um circuito linear equivalente [7].

2.2.2 Microcontrolador

Na grande maioria dos equipamentos de acionamentos industriais é empregado o uso de um microprocessador embarcado para controle do dispositivo. Esse componente por sua vez possui um programa composto de algoritmos dedicados para a aplicação em questão. O tipo de microprocessador que mais se adequa nesses casos, é aquele que contempla na mesma pastilha CPU (*Central Processing Unit* – Unidade Central de Processamento), Circuitos de memória e periféricos de E/S dedicados como conversores A/D, D/A e saídas do tipo PWM. A esse dispositivo é dado o nome de microcontrolador [7].

Um microcontrolador pode ser programado em linguagens de baixo nível como ASSEMBLY, ou de alto nível, como BASIC ou PASCAL, no entanto a linguagem mais empregada atualmente para a programação de microcontroladores é conhecida por uma linguagem de nível médio, denominada de linguagem em C, com a qual é possível ter certo controle do *hardware*, mas também é possível que pessoas leigas da área de eletrônica possam programar [9], [10].

Além das entradas e saídas digitais, outro periférico importante encontrado nos microcontroladores é o de protocolos de comunicação serial e/ou paralela. A grande maioria dos microcontroladores possui ao menos um tipo de protocolo de comunicação, como RS232, USB, I2C, RS485 entre outras. Esse tipo de periférico possibilita a troca de informações entre componentes eletrônicos ou mesmo com computadores, HMI (*Human-Machine Interface* – Interface Homem e Máquina) entre outros componentes. Também é possível por meio de um *Driver* externo, converter o protocolo de comunicação existente no microcontrolador por outro que deseja utilizar. O que aumenta ainda mais a gama de possibilidades de comunicação entre os dispositivos [9].

2.3. Internet das Coisas (IoT)

Originado do Inglês, o termo IoT (*Internet of Things*) é a tradução de Internet das Coisas, e como o próprio nome já diz, transmite o conceito de que a internet pode conectar todas as coisas. Criada com o intuito de facilitar a vida das pessoas, surge a ideia de que todos os equipamentos utilizados pela sociedade podem estar conectados entre si, tornando possível a interação entre os mesmos e trazendo uma nova perspectiva do futuro [11].

A IoT pode ser descrita como a junção de diversas tecnologias, onde elas são complementares para viabilizar a integração do ambiente ciber-físico. Os blocos básicos para a construção da IoT, mostrados na figura 4, são: Identificação, Sensores/Atuadores, Comunicação, Computação, Serviços e Semântica [12].

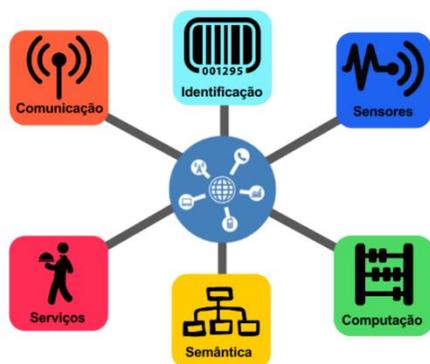


Figura 4 – Blocos Básicos da IoT. [12]

Um sistema ciber-físico (*Cyber-physical system*) consiste na combinação de um componente de *software* com partes mecânicas ou eletrônicas. O controle, o monitoramento, a transferência de dados e o intercâmbio de dados são geralmente executados via internet em tempo real.

A característica básica de comunicação utilizada na internet é fundamentada em dois principais elementos: cliente e servidor. Sempre que o cliente necessita realizar a comunicação ele acessa servidor. O servidor por sua vez, sempre deve estar disponível, esperando a comunicação do cliente. O servidor é um *software* que sempre mantém uma porta de comunicação acessível, no aguardo da comunicação do cliente e seu endereço deve ser conhecido por todos que desejam acessá-lo. Já o cliente que também é um *software*, deve iniciar a comunicação com o servidor diretamente pelo usuário ou automaticamente, devido a um evento ou ação externa. Podendo atuar como cliente, os dispositivos IoT podem acessar servidores para atualizar informações sobre seu funcionamento. Nos projetos de IoT, diversos servidores podem ser utilizados, sendo eles servidores de arquivos, web, banco de dados, entre outros [13].

Porém com uma quantidade enorme de dispositivos a serem conectados à rede, seria necessário ampliar o limite de endereços disponíveis, só tornando possível a ideia de IoT com o avanço do protocolo TCP/IP, utilizado para endereçar todos os dispositivos conectados na Internet, através do IP (*Internet Protocol*). Até pouco tempo atrás, o protocolo utilizado era o IPv4 (*Internet Protocol version 4*), que no caso permitia até $4,3 \times 10^9$ endereços de IP (4,3 bilhões). Esse limite acabou se esgotando, impactando na possibilidade novos dispositivos se conectarem à rede. Para solucionar esse problema, a *Internet Engineering Task Force* (IETF) desenvolveu o IPv6 (*Internet Protocol version 6*). Sendo a versão mais recente e utilizando endereços 128 bits, o

protocolo IPv6 permite aproximadamente $3,4 \times 10^{38}$ endereços de IP [14].

A IoT apresenta uma proposta irrecusável para o mundo, e obviamente o Brasil não podia ficar de fora. Pensando nisso, o BNDES, em parceria com o Ministério da Ciência e Tecnologia, Inovações e Comunicações (MCTIC), apoiou a realização de um estudo para o diagnóstico e a proposição de plano de ação estratégico para o país em IoT [15].

2.4. Computação em Nuvem

A Computação em Nuvem, tradução de *Cloud Computing*, nada mais é do que o novo modelo de computação, do qual permite que os clientes possam acessar diversas aplicações e serviços de qualquer lugar independentemente da plataforma, a partir de um terminal conectado à “nuvem” [16].

Estima-se que 85% da capacidade da computação do mundo esteja ociosa segundo a IBM, portanto a computação em nuvem é parte da evolução natural da TI (Tecnologia da Informação) e um meio de aprimorar o uso da capacidade computacional do mundo. A *Cloud Computing* também é flexível, fazendo com que as organizações possam contratar apenas uma parte dos serviços em nuvem [17].

Um dos principais motivos para a construção de serviços fundamentados em Computação em Nuvem é que provedores de serviços que utilizam serviços de terceiros, pagam de acordo com o seu uso. Isso evita investimentos iniciais em TI e o custo é previsível de acordo com o aumento da demanda. Além disso, é possível ainda que provedores de serviços até repassem o custo de computação, rede e armazenagem para usuários finais, devido ao uso ser contabilizado [18].

Segundo o NIST (*National Institute Of Standards and Technology*), um modelo de computação em nuvem deve conter características essenciais, conforme as descritas a seguir e mostradas na figura 5:

Autoatendimento sob demanda: funcionalidades computacionais providas de forma automática, sem interação humana.

Ampla Acesso a serviços de rede: recursos disponíveis através da internet, acessados de forma padronizada, permitindo assim o acesso de diversas plataformas.

Pool de recursos: recursos do provedor são utilizados por vários usuários, sendo realocados conforme demanda.

Elasticidade rápida: funcionalidades rápidas e elasticamente providas, assim como rapidamente liberadas.

Serviços mensuráveis: os sistemas de gerenciamento da computação em nuvem monitoram de forma automática os recursos de cada serviço (largura de banda, processamento e armazenamento).

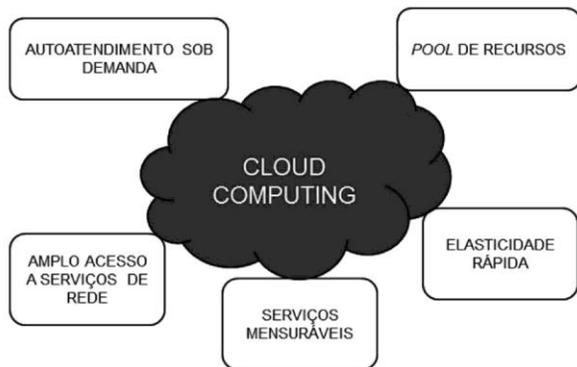


Figura 5: Características essenciais da nuvem definidas pelo NIST [17].

2.5 Integração de Sistemas

Atualmente não se pode desconsiderar o uso da TI (Tecnologia da Informação) e os sistemas de informação se quiser atender as necessidades informacionais de uma organização e sua consequente geração e transmissão [19], [20]. Mesmo assim, a maioria dos sistemas de TI nos dias atuais não estão totalmente integrados.

Empresas, fornecedores e clientes quase sempre possuem sistemas e processos distintos. Os departamentos como engenharia, produção ou serviço também entram nesta lista. Até mesmo os processos da empresa junto ao nível do chão de fábrica não estão totalmente integrados. Mas com a chegada da indústria 4.0, empresas e departamentos se tornarão muito mais coesos, criando redes universais de integração de dados, permitindo processos verdadeiramente automatizados [21].

A integração de sistemas pode ser classificada por dois tipos, sendo integração horizontal e vertical. Entende-se como horizontal a integração dos vários sistemas de TI para suporte na implementação de processos de agregação de valor, como vendas, fabricação, engenharia, logística e serviço, seja ela dentro ou fora da empresa. Já a vertical, entende-se como a integração dos sistemas em níveis hierárquicos diferentes de um processo produtivo, como por exemplo nos níveis de sensores, fabricação, execução e controle, além do nível de recursos corporativos [22].

Partindo da ideia de que a TI flexibiliza sistemas existentes e auxilia na circulação de informações de usuários para executar suas tarefas, e que esses sistemas formam a fonte de dados da organização, analisam-se a seguir, as ferramentas existentes para geração, tratamento e comunicação da informação, a *Business Intelligence*.

Inteligência de negócios (ou *Business Intelligence*, em inglês) é um processo que possui técnicas, tecnologias e *softwares* para encontrar causas ou explicações para eventos e resultados a partir uma imensa análise de dados. Resumidamente o BI deve procurar causas dos problemas e as melhores práticas a serem aplicadas. Além de encontrar o problema mais comuns entre as máquinas de uma indústria, é necessário conhecer o porquê, para que assim possa-se eliminar as falhas e amenizar os prejuízos. O processo de BI

pode utilizar sistemas de gerenciamento, ferramentas de *Data Mining* com dados de sistemas tradicionais. Com isso, pode-se dizer que o BI está muito próximo de quem toma as decisões [23].

III. MÉTODO UTILIZADO

Fez-se uso de ferramentas como *softwares* de simulação, sempre que o uso destes for possível uma vez que permitem criar, desenvolver e simular o sistema antes da concepção física do projeto. Este recurso permite prever possíveis obstáculos até então não considerados e determinar a solução, poupando tempo e gastos desnecessários.

Depois de obtido o resultado desejado nas simulações via *softwares*, foram executadas simulações físicas com o intuito de verificar os mesmos resultados na prática.

Somente então, após todas as etapas de simulações virtual e físicas concluídas, a célula foi implementada.

3.1. A CÉLULA

Para pontuar cada etapa do projeto é necessário pensar no sistema como um todo. O protótipo será constituído por um micro robô antropomórfico com três graus de liberdade, um estágio para entrada aleatória de peças, outro para leitura do tipo de peça, que será realizada por um sensor e o terceiro estágio onde as peças são alocadas de acordo com seu tipo. A planta envia dados como quantidade de peças separadas até o momento, se está operando ou parada entre outros dados para um PC localizado na célula. Estas informações são salvas em um banco de dados instalado no PC e acessado por um *Business Intelligence*, que mostra através de um *dashboard* os dados de forma gráfica e numérica para o usuário. Essas informações podem ser acessadas mesmo remotamente, pois foi aplicada uma arquitetura de computação em nuvem do tipo privada.

3.2. O MANIPULADOR

O manipulador possui diversas variáveis a serem consideradas, e em grande maioria variáveis da área de mecânica. Como o objetivo deste trabalho é a aplicação da indústria 4.0 a uma célula robótica e não exatamente o projeto de um robô, optou-se pela aquisição de um dos modelos existentes atualmente no mercado.

Existem atualmente diversos tipos de manipuladores, desde aqueles mais voltados para aplicações industriais até os de uso *makers*. Então, para escolha do manipulador foi levado em consideração à necessidade de precisão, graus de liberdade, alcance, material do qual foi produzido e preço. O manipulador escolhido é apresentado na figura 7.

3.3. A SMULAÇÃO DO MANIPULADOR

Para a modelagem e simulação virtual do robô e do controlador, será utilizado o *software matlab robotics toolbox*.

extensão desenvolvida por peter corke para o estudo de robótica [5].



Figura 7: Manipulador com 3 graus de liberdade a ser utilizado no projeto [24].

O *Matlab Robotics Toolbox* possui modelos prontos de robôs controladores do meio industrial para simular. No entanto, também permite a criação de um novo modelo com as características desejáveis.

Para que a simulação se aproxime o máximo possível do real, é necessário que o sistema a ser testado possua o maior detalhamento possível. portanto, a necessidade de criar um novo modelo no *software* com parâmetros próximos dos que o manipulador escolhido possui, é indispensável. a figura 8 ilustra a criação de um manipulador.

```

% CRIANDO UM MANIPULADOR ROBÓTICO RRR
(3 GRAUS DE LIBERDADE)
% 1º CRIAR OS ELOS

Elo(1) = Link([0 0.30 0 pi/2]);
Elo(2) = Link([0 0 0.30 0]);
Elo(3) = Link([0 0 0.30 pi/2]);

% 2º CONEXÃO DOS ELOS POR MEIO DE
JUNTAS
R3 = SerialLink(Elo, 'name', 'RRR');

% 3º VISUALIZAR MANIPULADOR NO
ESPAÇO
Q = [pi/2 pi/3 pi/4];
R3.plot(Q)
    
```

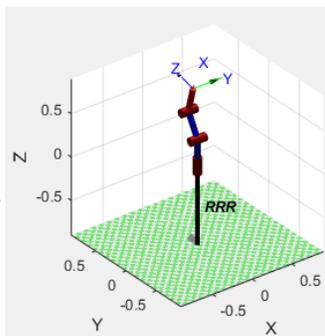


Figura 6: Manipulador com 3 graus de liberdade, criado no *matlab robotics toolbox*.

Fonte: Próprio autor.

3.4. O CONTROLADOR

O controlador também pode ser simulado por meio de parâmetros matemáticos e diagramas de blocos no *Matlab Simulink*. Porém, existem *softwares* dedicados a circuitos eletrônicos que possibilitam uma maior praticidade em alguns momentos.

Para o controle do robô foi utilizado um microcontrolador e alguns *softwares* possibilitam efetuar a programação de um microcontrolador e simular circuitos a ele conectados.

Um dos benefícios deste tipo de *software* além da simulação é a possibilidade de aproveitar o algoritmo já criado na simulação para gravar fisicamente no microcontrolador.

O microcontrolador tem a responsabilidade de coletar e interpretar os sinais de entrada como o sensor que fará a leitura das peças e determinar onde o manipulador deverá posicionar a peça. Além de também ser responsável por alimentar com informações o banco de dados do PC. A figura 7 ilustra a plataforma Arduino *open source* que será utilizado.



Figura 7: Plataforma Arduino Uno R3 [13].

A plataforma Arduino Uno R3 é composta por um microcontrolador Atmega328 que possui 14 E/S digitais que podem ser configuradas conforme aplicação, 6 entradas analógicas, um cristal oscilador de 16MHz, conexão USB para programação. Essa placa ainda conta com os protocolos de comunicação USART (*Universal Synchronous Asynchronous Receiver Transmitter*), SPI (*Serial Peripheral Interface*) e IC (*Inter-Integrated Circuit*).

3.5. O SERVO MOTOR

Para a conexão e movimentação dos elos do manipulador, foi utilizado micro servomotores. Cada junta possui um atuador desse tipo e um atuador do mesmo modelo para controle de abertura e fechamento da garra. Esse tipo de motor permite um controle preciso e de baixo custo, uma vez que possui 3 fios de ligação, sendo um para alimentação 5 V+ outro GND e por fim o de controle, que recebe um comando em PWM. É enviado um sinal com frequência Fixa em torno de 50Hz, e alterando o *duty cycle* de 1ms até 2ms que corresponde respectivamente a um movimento do eixo de 0° a 180°. A figura 8 ilustra o modelo de micro servomotor que será utilizado no projeto.



Figura 8: Micro servomotor que foi utilizado no projeto [25].

3.6. O SENSOR DE IDENTIFICAÇÃO DE CORES

No estágio de identificação de peças, foi utilizado um sensor capaz de detectar as cores vermelha, azul e verde.

O módulo TCS230 opera com faixa de tensão entre 3 e 5V, possui 2 terminais para seleção de frequência de saída (S0, S1) que podem variar de 10KHz a 600KHz. Também conta com mais 2 terminais para seleção do tipo de filtro (S2, S3) a ser utilizado em cada leitura e um terminal de *enable* (OE) para habilitar a leitura e saída de sinal do módulo.

O sensor é composto por 64 fotodiodos que captam a intensidade de luz, sendo que 48 desses componentes possuem filtros RGB (16 para a cor vermelha, 16 para a cor verde e 16 para a cor azul) e os outros 16 não possuem filtro algum. Os fotodiodos geram uma corrente de saída conforme a intensidade de luz refletida e o filtro selecionado, essa corrente é convertida em um sinal de onda quadrada com 50% de *duty cycle* e frequência proporcional à intensidade de luz refletida. A figura 9 ilustra a estrutura do componente TCS230.

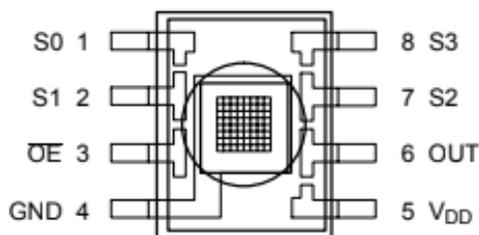


Figura 9: Composição do sensor de identificação de cores tcs230 [26].

3.7.A COMUNICAÇÃO ETHERNET

A plataforma embarcada Arduino Uno R3 é responsável por gerir todo o sistema da planta, mas não terá a função de reter certos tipos de informações que são desnecessárias para a célula. Porém de extrema importância para análises de produtividade para gestores industriais, departamento de manutenção entre outros.

Como mencionado anteriormente, o microcontrolador a ser usado não contempla o protocolo de comunicação ETHERNET TCP/IP. Usou-se então, um *shield* para que se torne possível a comunicação entre o Controlador e o PC. Há atualmente no mercado, diversos dispositivos para essa função, contudo foi escolhido para esse projeto o *shield* W5100 que é fundamentado no *chip* W5100 da Wiznet. Este *shield* foi desenvolvido exatamente para ser acoplado a uma placa Arduino, o que permite praticidade, estética e diminui possibilidades de interferência na comunicação entre as placas.

Este embarcado opera com tensão de 5V, possui além do *chip* W5100 um *buffer* de 16k, velocidade de conexão de 10/100Mb e se conecta com o Arduino através da porta SPI. A figura 10 ilustra o *shield* W5100.

3.8.O BANCO DE DADOS

O computador que é conectado ao controlador, deve ter um sistema de gerenciamento de banco de dados para armazenar as informações que não serão retidas no microcontrolador.

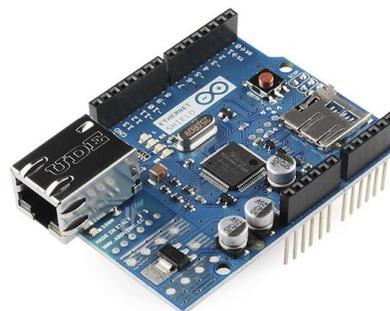


Figura 10: *Shield* W5100 para comunicação Arduino/PC via ETHERNET [13]

Para esta finalidade é instalado no computador um sistema de gerenciamento de banco de dados relacional que utiliza linguagem de consulta estruturada SQL (*Structured Query Language*) denominado MySQL [27].

O MySQL é atualmente um dos gerenciadores de banco de dados relacional mais utilizados no mundo por grandes empresas devido ao seu servidor confiável, rápido e de fácil utilização. Considerando inclusive aplicações com ênfase em *internet* [27].

Para início de testes com o MySQL é necessário estar em posse do microcontrolador a ser utilizado. A princípio é inserido um programa no embarcado mais simples com o intuito de focar apenas no envio e aquisição de informações entre os sistemas em um primeiro momento. Após desenvolvido essa primeira etapa é de fato inserido o algoritmo que é utilizado no projeto e o envio e tratamento desses dados pelo MySQL.

3.9.A DASHBOARD

Para a criação de *dashboard* é utilizada uma ferramenta da Microsoft voltada para *Business Intelligence*. O *Power BI* possui uma coleção de serviço de *software* para tornar os mais diversos tipos de informações alocadas em um servidor local, ou nuvem de forma clara e eficiente para ser visualizado e interpretado de forma rápida, permitindo melhores tomadas de decisões por gestores.

O *Power BI* possui um *Gateway* para aquisição de dados do *MySQL*, o que facilita o processo de integração entre ambos. As informações salvas no banco de dados são mostradas pelo *Business Intelligence* através de um *dashboard* interativo, e serão publicadas por meio do serviço *Web*, o que possibilita ser acessado remotamente por outras pessoas interessadas e autorizadas. A Figura 11 ilustra um *dashboard* criado no *Power BI*.

IV. RESULTADOS ESPERADOS

A plataforma de supervisão da célula robótica, nome dado ao protótipo do trabalho, tem como proposta demonstrar através de estudos os efeitos e vantagens da Indústria 4.0, aplicando conceitos dos principais pilares dessa tecnologia, tais como Computação em Nuvem, Robôs Autônomos,

Simulação e Interação de sistemas. A partir desse contexto, os principais resultados esperados, divididos por etapas são:



Figura 11: Dashboard
Fonte: Próprio autor

Etapa 01 - Realizar modelagem da célula robótica com o objetivo de documentar, entender e analisar o processo, permitindo a transformação e a automatização. Através da diagramação de processos será possível obter uma visão lógica das atividades e apresentar de forma intuitiva, como será realizado.

Etapa 02 - Espera-se desenvolver uma programação em linguagem C, a ser embarcada no microcontrolador Atmega328, criando variáveis, funções, rotinas e estruturas de controles lógicos para funcionamento demonstrativo do processo industrial aplicado à célula robótica.

Etapa 03 - Interagir o microcontrolador com dispositivo de manipulação, responsável pelo manuseio das amostras. Com o servo motor, fazendo a movimentação do elo do eixo do manipulador. E com os sensores, que identificam as cores das amostras. Juntos esses componentes executarão o processo produtivo do protótipo.

Etapa 04 - Montar uma estrutura para a microcélula robótica a fim de melhor posicionar os componentes, utilizando materiais de baixo custo como plástico, papelão e metal. O objetivo desse arranjo físico é organizar o espaço do “chão de fábrica” para se obter a forma mais eficiente do trabalho, garantindo uma produtividade eficiente.

Etapa 05 - Gerenciar o banco de dados coletado pelo microcontrolador, armazenando-o em computador via protocolo TCP/IP, utilizando linguagem de consulta estruturada SQL.

Etapa 06 - Criar uma *dashboard* utilizando o *software* Microsoft *Power BI* que realizará a leitura e das informações

Digital Object Identifier: (only for full papers, inserted by LACCEI).
ISSN, ISBN: (to be inserted by LACCEI).
DO NOT REMOVE

salvas no servidor local, exibindo as mesmas de forma clara e eficiente, propiciando a melhor tomada de decisão dos gestores.

IV. RESULTADOS OBTIDOS

Etapa 01 - Antes da aquisição dos componentes foram efetuadas diversas simulações, a fim de validar o uso de todos componentes considerados no início do projeto, e assim, evitar o desperdício de tempo e dinheiro.

As simulações foram realizadas via *software Autodesk Online*, a fim de interligar os microserves com a placa Arduino R3, onde foi dado início no desenvolvimento do *software*. Nesse momento um dos obstáculos encontrados foi a forma de controlar aceleração e desaceleração dos microserves, uma vez que, quando o manipulador estivesse em movimento, seria necessário evitar movimentações bruscas, por se tratar de manipulação e posicionamento de peças.

Etapa 02 - Após a validação da simulação, a interação entre o Arduino Uno R3 e os microserves considerados para movimentação das articulações do manipulador, verificou-se uma possível solução para simular o sensor de verificação de cores. No entanto, não foi encontrado nenhum *software* gratuito que possibilitasse a simulação do Arduino com esse sensor. Portanto, nesse caso foi analisado o *datasheet* do sensor para verificar que tipo de sinal era necessário enviar e receber as características de alimentação e consultado artigos científicos e trabalhos de graduandos em que se utiliza esse tipo sensor, a fim de obter uma maior segurança para aquisição do mesmo.

Etapa 03 - Após a validação da interligação de todos componentes anteriormente citados, foi realizada uma consulta com maior enfoque no *shield ethernet* para verificar a compatibilidade com o Arduino Uno R3, onde também não foi possível simular via *software*, apenas consulta ao *datasheet* e artigos relacionados ao assunto. No entanto, nessa consulta percebeu-se que mais pinos do Arduino ficariam inativos do que considerados em um primeiro momento. Nesse momento notou-se que a quantidade de portas digitais disponíveis no Arduino Uno para interligação dos componentes não eram suficientes. Como alternativa considerou-se multiplexar e demultiplexar alguns pinos, no entanto, para fazer o controle desses componentes seria necessário a utilização de pinos I/O, o que necessitaria de ainda mais componentes.

Uma outra alternativa seria substituir o Arduino Uno pelo Arduino Mega 2560. Considerando as duas alternativas, a segunda mostrou-se mais eficaz, uma vez que possuía mais recursos e o preço para investimento de ambas as soluções seria praticamente o mesmo. Portanto, por esse motivo foi efetuada a substituição do Arduino Uno pelo Mega 2560.

Etapa 04 - Após simulações foi dado início na compra dos componentes e assim que se obteve os componentes em mãos,

(Arduino, *shield ethernet*, manipulador robótico com microserves e sensor de cores) foi testado o algoritmo via simulador no controle dos microserves do manipulador. Após validado, foi desenvolvido um algoritmo para testar a identificação de cores pelo sensor.

Etapa 05 - Após testes isolados com ambos componentes, deu-se início no processo de posicionamento definitivo do manipulador para capturar a peça em posições pré-determinadas, em seguida movimentar a peça até o sensor, verificar a leitura e a distribuição de correta das peças. Nessa etapa, notou-se a influência da luz do ambiente e a tonalidade das cores interferiam na leitura e conseqüentemente no armazenamento correto das peças. Esse obstáculo foi possível contornar via algoritmo por meio de comparações entre *ranges* de leitura de cada cor (RGB), comparando leituras realizadas pelo sensor e resultados obtidos.

Etapa 06 - Finalizada essa etapa, foi a vez de iniciar os testes de comunicação entre o Arduino, o *shield ethernet* e computador para enviar as informações para o banco de dados do MySQL. Nessa etapa, houve dificuldades em manter o computador conectado à internet por uma rede *wireless* recebendo informações do Arduino via rede cabeada. Com isso, foi necessário alterar configurações do roteador como mudança de IP e *gateway*, entre algumas outras. Após alterações necessárias, o problema de uma rede influenciar na outra foi resolvido e deu-se início a adquirir os dados com o *Power BI* do banco de dados do MySQL.

Etapa 07 - Para o desenvolvimento do *dashboard* através do *Power BI Desktop* não houveram problemas para consultar os dados no MySQL. Porém, notou-se um problema em relação a atualização automática dos dados. Desejava-se que quando o banco de dados do MySQL fosse atualizado, o *dashboard* também fosse atualizado de forma simultânea. No entanto para essa função seria necessário adquirir uma licença paga do Power BI ou possivelmente migrar do banco de Dados do MySQL para o SQL server da própria Microsoft por exemplo. Com a licença gratuita, é possível publicar o *dashboard* na web, no entanto ela permite até 8 atualizações automáticas por dia. No entanto a atualização manual por parte de quem for visualizar o dashboard é liberada sem restrições. Por ser um dashboard voltado para BI, optou-se por manter o projeto original e manter o uso banco de dados do MySQL em conjunto com o Power BI sem as atualizações automáticas minuto a minuto.

VI. CONCLUSÕES

É bastante notório que a Indústria 4.0 irá revolucionar todo ambiente fabril, trazendo uma nova perspectiva para o futuro das corporações. Porém a maior dificuldade de acompanhar esse processo revolucionário está na resistência de certos padrões organizacionais, que muitas vezes criam uma barreira para novas tecnologias, seja por medo de investir

e não obter o resultado desejado ou mesmo pela falta de conhecimento sobre o que ela pode propiciar. Fundamentado nisso, o presente trabalho buscou apresentar as vantagens dessa tecnologia em uma escala demonstrativa, mas com o propósito de possibilitar sua ampliação em uma escala real no futuro, para que assim, possa ser implementada em pequenas linhas industriais.

O principal intuito da plataforma de supervisão da célula foi integrar a robótica com redes de comunicação, processos físicos, demonstrando os resultados de forma clara e eficiente através de um *dashboard*.

A plataforma de supervisão da célula robótica demonstrou que os resultados respondem a prova do conceito para a solução proposta e ainda, que com um baixo custo, é possível alcançar ótimos resultados na gestão de processos industriais, especialmente na confiabilidade das informações, visto que, não depende de operadores de máquinas para a coleta de dados, além do *feedback* instantâneo de qualquer lugar que se tenha conexão com a *internet*.

Atualmente no Brasil, mais do que nunca é necessário aceitar a essas novas tendências de braços abertos, pois isso trará mais competitividade com o mercado externo, adaptando-se às exigências internacionais de qualidade e livrando-se das estruturas atualmente engessadas.

Pensando em trabalhos futuros, tem-se a intenção da implementação da plataforma de supervisão da célula robótica na empresa que patrocinou o protótipo (empresa dos autores).

VII. REFERÊNCIAS

- [1] ALMEIDA, P. S. (2019). *Indústria 4.0: princípios básicos, aplicabilidade e implantação na área industrial*. São Paulo: Érica.
- [2] MINISTÉRIO DA INDÚSTRIA, C. E. (2019). *Grupo de trabalho da indústria 4.0 criado pelo Ministério da indústria, Comércio Exterior e Serviços*. Acesso em 25 de maio de 2020, disponível em Site do Grupo de trabalho da indústria 4.0: <http://www.industria40.gov.br>.
- [3] KUPPER, D. (2018). *Boston Consulting Group*. Acesso em 25 de maio de 2020, disponível em Site da Boston Consulting Group: <https://www.bcg.com/pt-br/capabilities/operations/embracing-Industry-4.0-rediscovering-growth.aspx>.
- [4] INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. (s.d.). ISO 8373:2012. *Robots and robotic devices — Vocabulary*.
- [5] CRAIG, J. J. (2005). *Introduction to Robotics - Mechanics and Controls*. Prentice Hall.
- [6] MATARIC, M.J., *The Robotics Primer*, 2006, disponível em: https://www.researchgate.net/publication/220690176_The_Robotics_Primer.
- [7] PENEDO, S. R. (2014). *Servoacionamento: Arquitetura e Aplicações*. São Paulo: Érica.
- [8] MOHAN, N., UNDELAND, T. M., & ROBBINS, W. P. (1995). *POWER ELECTRONICS - Converters, Applications and Designs*. John Wiley & sons, Inc.
- [9] GIMENEZ, S. P. (2015). *MICROCONTROLADORES 8051: Conceitos, Operação, Fluxogramas e Programação*. São Paulo: Érica.

- [10] ALMEIDA, R. M., MORAES, C. H., & SERAPHIM, T. D. (2016). *PROGRAMAÇÃO DE SISTEMAS EMBARCADOS: Desenvolvendo Software para Microcontroladores em linguagem C*. Rio de Janeiro: Elsevier.
- [11] ALECRIM, E. O que é Internet das Coisas (Internet of Things)? [S.l.]: Info Wester, 2017. Disponível em: <<https://www.infowester.com/iot.php>>. Acesso em: 4 março de 2018.
- [12] SANTOS, SILVA, CELES, NETO, PERES, VIEIRA, GOUSSEVSKAIA E LOUREIRO, 2016, disponível em: <https://homepages.dcc.ufmg.br/~mmvieira/cc/papers/internet-das-coisas.pdf>.
- [13] OLIVEIRA, S. d. (2017). *INTERNET DAS COISAS COM ESP8266 ARDUINO E RASPBERRY PI*. São Paulo: Noatec.
- [14] FACCIONI FILHO, Mauro. Internet das Coisas: livro didático. Palhoça: UnisulVirtual, 2016. p. 11 - 28. Disponível em: <https://www.researchgate.net/profile/Mauro_Fazion_Filho/publication/319881659_Internet_das_Coisas_Internet_of_Things/links/59c038d5458515e9cfd54ff9/Internet-das-Coisas-Internet-of-Things.pdf> Acesso em: 28 maio de 2020.
- [15] BNDS. Internet das Coisas: um plano de ação para o Brasil. Chamada pública, 2017. Disponível em: <<https://www.bndes.gov.br/wps/portal/site/home/conhecimento/pesquisaedados/estudos/estudo-internet-das-coisas-iot/estudo-internet-das-coisas-um-plano-de-acao-para-o-brasil>> Acesso em: 31 de maio 2020.
- [16] SILVA, F. H. R. Um estudo sobre os benefícios e os riscos de segurança na utilização de Cloud Computing; 2010. 15f. Artigo científico de conclusão de curso apresentado no Centro Universitário Augusto Motta, UNISUAM-RJ.
- [17] VERAS, Manoel. Cloud Computing Nova Arquitetura da TI. Rio de Janeiro/RJ. Brasport livros e Multimídia Ltda., 2012.
- [18] SOUZA, Flávio R. C., MOREIRA, Leonardo O. M. e MACHADO, Javam C. Computação em Nuvem: Conceitos, Tecnologias, Aplicações e Desafios. 2009. Artigo científico de conclusão de curso apresentado na Universidade Federal do Ceará, UFC.
- [19] , Rezende, Abreu, 2003, disponível em: <https://www.seer.ufg.br/read/article/view/42705>.
- [20] REGINATO, Luciane e NASCIMENTO, Auster Moreira. Um estudo de caso envolvendo Business Intelligence como instrumento de apoio à controladoria. Rev. contab. finanç. [online]. 2007, vol.18, pp.69-83. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1519-70772007000300007&lng=en&nrm=iso> Acesso em 30 de maio de 2020.
- [21] RÜßMANN, M. et al. Industry 4.0: The Future of Productivity and Growth in Manufacturing Industries. Boston Consulting Group. 2015. Disponível em: <http://www.inovasyon.org/pdf/bcg.perspectives_Industry.4.0_2015.pdf > Acesso em 31 maio 2020.
- [22] PLATTFORM INDUSTRIE 4.0. Umsetzungsstrategie Industrie 4.0 - Ergebnisbericht der Plattform Industrie 4.0. BITKOM. 2015. Disponível em < <https://www.bitkom.org/noindex/Publikationen/2015/Leitfaden/Umsetzungsstrategie-Industrie-40/150410-Umsetzungsstrategie-0.pdf> > Acesso em 31 de maio 2020.
- [23] LOH, Stanley. BI na era do big data para cientistas de dados - indo além de cubos e dashboards na busca pelos porquês, explicações e padrões. Porto Alegre/RS, 2014.
- [24] RoboCore, 2020, disponível em: https://www.robocore.net/?gclid=CjwKCAjwu5CDBhB9EiwA0w6sLXuEfKr0VlpSFsLTvrSsWM-juHwKJcIrO1ujvglvy9Cd8lZ3-vl-TxoC9K8QAvD_BwE.
- [25] Tower Pro, 2020, disponível em: <https://www.usinainfo.com.br/servo-motores/servo-motor-sg5010-tower-pro-180-11kgfcm-de-alto-torque-6074.html>.
- [26] SOLUTIONS, T. A. (2004). *TCS230 PROGRAMABLE COLOR LIGHT-TO-FREQUENCY CONVERTER*. *The Robotics Primer 2007* Cambridge Massachusetts Institute of Technology
- [27] MANZANO, J. A. (2011). *MySQL 5.5 - Interativo: Guia Essencial de Orientação e Desenvolvimento*. São Paulo: Érica.