



Universidade Federal do ABC

Engenharia de Instrumentação, Automação e Robótica

Anderson Carlos dos Santos

IMPRESSÃO ROBOTIZADA COM CARTUCHOS DE TINTA INKJET

Monografia

Santo André - SP
2017

Anderson Carlos dos Santos

IMPRESSÃO ROBOTIZADA COM CARTUCHOS DE TINTA INKJET

Monografia

Monografia apresentada ao curso de graduação da Universidade Federal do ABC, como
requisito parcial para obtenção do grau de bacharel em Engenharia de Instrumentação,
Automação e Robótica.

Orientador: Dr. Luis Alberto Martinez Riascos

Santo André - SP
2017

Agradecimentos

À minha mãe Lúcia Helena e ao meu pai José pelo carinho, paciência e confiança que tiveram em mim durante toda graduação, por terem acreditado e trabalhado para que conquistasse meus objetivos. Esse trabalho é tanto mérito de vocês quanto meu.

Agradeço aos meus amigos; aos que ficaram em Minas, aos que conheci em São Paulo e aos que conheci na minha estadia no Canadá, sendo cada um ao seu tempo as pessoas com quem tive maior contato nesse período e responsáveis pelas maiores diversões.

Aos familiares que estavam presentes; meus avós, padrinhos e primos. E em especial a minha avó pelo carinho de preparar bolachas para mim sempre que retornava à Minas.

Agradeço a técnica Lívia de Castro pela grande ajuda auxiliando no laboratório para a conclusão do projeto e ao Professor Dr. Luis Riascos por ter me aceito como aluno a ser orientado, acatando o meu projeto.

Agradeço por final, a UFABC e ao CNPq por terem sido fonte de patrocínio em pesquisa ou intercâmbio durante minha graduação, sendo essas atividades fundamentais na qualidade do projeto desenvolvido.

Resumo

Este projeto trata do desenvolvimento de uma ferramenta que realiza impressões diretas em diferentes tipos de superfície utilizando uma ferramenta acoplada a um braço robótico. Foi projetado e construído um suporte e desenvolvido um controlador. O controlador é próprio para sua operação e o braço robótico é programado para que trabalhe em harmonia com a ferramenta, mantendo ambos um contato para realizar o processo de impressão. Para a reprodução de imagem foram utilizados técnicas de processamento de imagens para tornar a programação robótica mais rápida e automatizada. O desenho inicial foi pensado de acordo com as dificuldades atuais da indústria *livery* na aviação.

Palavras chaves: impressão direta, inkjet, robótica, reprodução de imagem, sistemas embarcados.

Abstract

This project is about a developing of a tool that can print directly onto different types of surfaces using a tool combined with a robotic arm. It was projected and builded a carrier and developed a micro-controller. The micro-controller and the robotic arm is programmed to work in harmony with the tool, holding both connected to realize the printing process. To reproduce the image was utilized image processing techniques to become the robotic programming faster and automatized. The initial design was thought according to the nowadays difficulties of aircraft livery companies.

Key words: direct printing, inkjet, robotics, image reproducing, embedded systems.

Lista de Figuras

1	Imagen de divulgação do centro de desenvolvimento da Airbus Hamburgo, ref.[9].	6
2	Circuito de ganho de tensão utilizado pelo Inkshield, imagem retirada da ref.[6].	10
3	Diagramação lógica usando um demultiplexador de 4 para 16 bits acionando o cartucho HP C6602, imagem retirada da ref.[6].	11
4	Simulação da operação do multiplexador recebendo 4 sinais de entrada do Arduino, imagem retirada da ref.[6].	11
5	Vistas ortogonais e em perspectiva isométrica da ferramenta.	15
6	Visualização do compartimento. Fonte: autores.	16
7	Visualização lateral da ferramenta acoplada. Fonte [7] e autores.	16
8	Visualização da ferramenta acoplada. Fonte [7] e autores.	17
9	Visualização interna do compartimento. Fonte: autores.	17
10	Componentes eletrônicos utilizados para montar o Inkshield.	22
11	InkShield conectado ao Arduino.	22
12	Teste realizados com o cartucho manualmente.	23
13	Círculo de conexão entre os dispositivos.	23
14	Visão superior da ferramenta.	24
15	Visão frontal da ferramenta.	24
16	Visão inferior da ferramenta.	24
17	Ferramenta acoplada no robô realizando testes.	25
18	Símbolo das Olimpíadas de Tóquio em 2020.	26
19	Imagen vetorizada.	27
20	Imagen em preto e branco.	27
21	Imagen binária.	27
22	Começo do processo de impressão robotizada.	28
23	Metade do processo de impressão robotizada realizado.	28
24	Resultado final do processo de impressão robotizada.	29
25	Datasheet do CI ULN2803 usado no shield.	33
26	Tabela verdade do demultiplexador CD74HC4067.	33

Lista de Tabelas

1	Componentes utilizados.	12
---	-------------------------	----

Conteúdo

1	Introdução	4
1.1	Impressões diretas	6
1.2	Efetuadores	7
2	Objetivo	8
3	Metodologia	9
3.1	Círculo Eletrônico	9
3.2	Programação da Ferramenta	13
3.3	Projeto Mecânico	15
3.4	Programação do Robô Manipulador IRB 1600	18
4	Resultados	22
5	Conclusão	31
6	Referências	32
7	Apêndice	33
7.1	Componentes Eletrônicos	33

1 Introdução

A robótica industrial teve início em 1961 com a criação do robô Unimate por George Devol nos Estados Unidos[1]. Desde então os robôs entraram no mercado industrial e até os anos de 1970 o Unimate não possuía grandes adversários. Na década de 70, empresas japonesas entraram no mercado robótico ao mesmo tempo que grandes empresas como General Motors e General Electric começaram a manifestar seu interesse em usar robôs para automatizar suas produções. Como consequência da grande demanda por robôs, surgiram também novas empresas nos Estados Unidos e na Europa. Até o final da década de 70, o ocidente dominava a tecnologia robótica, com destaque aos grandes centros acadêmicos americanos.

Os robôs articulados são o tipo mais comum presente na indústria hoje. A maioria desses robôs são seriais ou de cadeia cinematográfica aberta, armazenando uma série de posições em sua memória. O primeiro robô articulado com 6 eixos controlados eletronicamente foi fabricado pela KUKA em 1973, denominado Famulus[2].

Com a possibilidade de realizar movimentos mais complexos e um controle preciso, os robôs se tornaram uma mão de obra essencial na indústria, principalmente na automobilística, a eles são confiados os trabalhos de manuseio, soldagem, pintura, perfuração entre demais utilidades. A evolução da robótica também produziu ganhos na indústria aeroespacial, médica e militar. O manipulador multi-propósito controlado automaticamente programável e reprogramável, dito braço robótico, apresentou ser capaz de façanhas impossíveis ao homem utilizando os mesmos princípios utilizado pelos robôs industriais. Como exemplo podemos citar braços robóticos na Estação Espacial Internacional, que possibilitou o acoplamento de diversas naves espaciais e operações externas na estação[3]. Robôs cirurgiões também são uma realidade cada vez mais frequente vistos em hospitais do mundo todo, onde diversos braços robóticos em sincronia realiza uma operação.

A robótica não se limitou a categoria manipuladores e expandiu para a robótica móvel. Assim, robôs puderam se locomover e estar em lugares que o homem não consegue ainda chegar, tornando os robôs mais que a nossa força de trabalho, mas também olhos e ouvidos da humanidade.

No que se refere a robótica industrial, a indústria já encontra um nível de maturidade. As tarefas mais cotidianas e básicas apresentam robôs com controle suficiente para realizar as operações de maneira eficiente e rápida. No entanto trabalhos específicos podem requerer a necessidades de desenvolvimento de novas técnicas e ferramentas robóticas. Enquanto grandes empresas de robóticas concentram-se em aperfeiçoar o controle e a precisão de robôs, dando-lhe ao mesmo tempo maiores cenários de atuação e redução de tamanho custo e peso, a indústria ainda carece de profissionais capazes de utilizar os robôs manipuladores de maneira inteligente e inovadora para específicas operações. Enquanto em países desenvolvidos há níveis altíssimos de automatização, países em desenvolvimento carecem dessa modernização.

O Brasil como um país emergente ainda carece da automatização na indústria, o que deixa o país em baixos *ranks* de competição industrial mundial. O Brasil importa seus robôs e não consegue desenvolver projetos de automação suficientes para manter a indústria de grande tecnologia nacional. Empresas que exportam produtos dependem extremamente de importação de serviços industrializados para sua sobrevivência e competitividade. Enquanto países como a Alemanha experimenta sua “indústria 4.0” com fábricas fortemente automatizadas o Brasil ainda depende do trabalho manual para a produção[10].

No entanto, o Brasil ainda se destaca em tecnologia em determinados assuntos. As tecnologias desenvolvidas aqui na área de petróleo, aeronáutica e mineração entre outras estão entre as mais avançadas do mundo. Tais indústrias devem apresentar um elevado nível de automatização para se manterem no topo. O uso de robôs nessas indústrias pode representar maior segurança e precisão além de serem vitais dependendo da operação realizada. A capacidade de desenvolver projetos que automatizam serviços sem depender

exclusivamente da indústria estrangeira pode tornar o Brasil mais competitivo e inovador.

Uma maneira de automatizar a produção seria a utilização dos robôs articulados para realizarem tarefas que até então são realizadas manualmente ou através de máquinas que apresentam restrições no seu uso. Para isso, pode ser desenvolvida ferramentas que quando acopladas ao braço robótico sejam capazes de realizar novas operações. Assim, não seria necessário o desenvolvimento de novos robôs operadores, mas somente de novas ferramentas que deem novas utilidades a tecnologia já usada. Essas ferramentas podem conter sofisticados circuitos de instrumentação possibilitando uma grande extensão do campo de utilidade. Ferramentas mais complexas poderiam utilizar algoritmos de visão computacional e inteligência artificial de acordo com a necessidade. O trabalho poderia se tornar mais seguro e mais minucioso utilizando tais técnicas, a produção poderia ser otimizada e aumentada e o custo reduzido.

A industria atual ainda não possui ferramentas capazes de realizar impressões em diferentes tipos de superfícies. A impressão direta é um mercado restrito que algumas pessoas começaram a se aventurar recentemente. É possível encontrar alguns projetos utilizando impressão direta com acionadores mecânico em conjunto com motores de passo, no entanto até agora não existe nenhum serviço que realize impressões utilizando robôs manipuladores. Disso, surge o nome impressão robotizada.

Uma das grandes vantagens da impressão ink-jet é a ausência de contato entre o cartucho e o material a receber impressão (geralmente o papel). Assim, como se procura a possibilidade de estender a lista de materiais que podem ser impressos, essa característica ajuda pois evitaria de danificar materiais mais sensíveis e dispensando um eventual controle de força.

Os cartuchos ink-jet são amplamente utilizado para impressões de alta resolução em impressoras residenciais e profissionais. Sua tecnologia de baseia em jatos de pequenos pontos de tintas que constituem uma imagem, assim como os pixels constituem uma imagem virtual, os pontos constituem uma imagem impressa e a resolução é dada em dpi (dots-per-inch ou pontos por polegadas em português). Cada cartucho possui diferentes especificações relacionadas à velocidade com que enviam os pontos de tintas e o número de pontos que podem emitir simultaneamente. O jato de tinta é acionado através de um pulso elétrico causada por uma tensão em um transdutor que produz uma pressão mecânica e empurra uma gotícula de tinta gerando o jato.

Serviços de desenho em fuselagem são feito manualmente através de pinturas em camadas, um processo demorado que emitem gases tóxicos nocivos ao ambiente e aos trabalhadores além de ser um dos principais fatores de peso do avião. Grandes empresas como a Airbus já investem em pesquisas para desenvolvimento de pinturas "livery" [4] automatizadas que possibilitem uma nova geração de liveries aviões comerciais, devido a grandes exigência de linhas aéreas quererem exibir sua identidade em mensagem visuais ou celebrar eventos através de pinturas em aviões. A criação de uma ferramenta que automatizasse os serviços de livery da Airbus se deu sem o uso de robôs manipuladores, utilizando um sistema de impressão de maior escala.

O Brasil possuindo a terceira maior fabricante de aviões do mundo e sendo uma das 10 maiores economias terá que encarar os desafios da automatização das suas empresas por si e se manter competitivo internacionalmente. O trabalho apresenta uma ideia inicial de como automatizar o processo de pintura, realizando impressões de baixa resolução utilizando robôs manipuladores. Além do mais, a utilização de robôs torna o uso genérico e não restringindo a apenas uma aplicação com determinadas dimensões e peculiaridades, podendo ser facilmente readaptável a outro trabalho.

1.1 Impressões diretas

A tecnologia de impressões direta baseia-se na tecnologia de impressões 3D, onde se imprime um "arquivo" com uma dispensador que pode se locomover em três direções. A impressão direta de imagens é um mercado mais restrito, e talvez por isso não tenha apresentado tanto desenvolvimento como as da impressões 3D. A diferença entre os dois tipos de impressões se dá pelo material usado e pela utilização de cores, na impressão 3D frequentemente não há uma exigência quanto a reprodução das cores focando o resultado no formato.

Não há atualmente no mercado nenhuma ferramenta que ofereça o serviço. Uma pesquisa no Google retorna resultados de impressões diretas em tecidos, porém a tecnologia utilizada é de uma impressora convencional diferente dos atuadores com movimentos tridimensionais. Alguns projetos utilizam robôs móveis para a impressão direta.

Na referência [6], é possível encontrar alguns projetos DIY utilizando impressões diretas com o dispositivo a ser usado no trabalho. Porém, nenhum dos projetos exibidos apresentam bom grau de resolução, nem uma adaptação do dispositivo a um atuador.

O projeto mais desenvolvido em impressão direta encontrado é o *ZUtA Pocket Printer* criado pela ZUtA labs Ltd. Um pequeno robô pode receber informações sem fio e percorrer uma superfície plana imprimindo as informações desejadas. O FAQ do produto informa que o robô apenas trabalha com a cor preta e que embora imprima em qualquer superfície, a empresa recomenda seu uso apenas em papel. O produto encontra-se em pré-venda com expectativa de lançado no terceiro quadrimestre do ano de 2017[8].

Como já mencionado, a Airbus possui um sistema exclusivo de impressão inkjet de êxito lançado em 2016. Por ser uma tecnologia exclusiva não há muitas informações sobre. Os resultados são uma economia de 3Kg no peso do avião e o uso de apenas 500ml de tinta para a pintura da cauda de um avião da companhia Thomas Cook. Para a Airbus, a tecnologia de impressão direta se relaciona com a visão da indústria digital de pintura, tornando impressões e pinturas possíveis sem máscara[9]. Traçando o objetivo de pintar o avião inteiro através de um sistema automatizado no futuro e não apenas a cauda.



Figura 1: Imagem de divulgação do centro de desenvolvimento da Airbus Hamburgo, ref.[9].

1.2 Efetuadores

Os fabricantes de robôs especificam nos catálogos as informações relativas às características dimensionais e de desempenho dos robôs, como acurácia, repetibilidade, carga máxima de manipulação, número de graus de liberdade, volume de trabalho e etc. Estes dados fornecem ao usuário subsídios suficientes para que se tenha uma ordem de grandeza das condições operacionais de um determinado robô[10].

As informações indicadas nos catálogos relacionam-se essencialmente à extremidade terminal do manipulador mecânico, ou seja, ao último elo. Portanto, torna-se necessária a inclusão de um componente capaz de promover a interação entre a extremidade terminal do manipulador mecânico e o objeto a ser trabalhado. Este componente é o efetuador[10].

Os efetuadores podem ser divididos em dois grandes tipos: as ferramentas especiais e as garras mecânicas. Este trabalho discute uma ferramenta especial.

As ferramentas têm como função realizar uma ação ou trabalho sobre uma peça, sendo relacionadas principalmente a operações de processamento e controle de qualidade[10].

As ferramentas são geralmente rigidamente fixas às extremidades terminais dos robôs, não possuindo movimentação relativa a estes. A função primordial do robô nestes casos é posicionar e orientar a ferramenta em relação à peça que será trabalhada[10].

O uso de ferramentas está associado diretamente às tarefas a serem realizadas. Dentre as ferramentas mais tradicionais utilizadas em operações de processamento estão: o porta-eletrodo, a pistola de aspersão (para pó, jateamento de superfícies e etc.), a pistola de pintura, as tochas para soldagem TIG e MIG/MAG, o dispositivo para soldagem/corte à plasma, o conjunto de pinças para soldagem por pontos, o dispositivo para soldagem/corte à laser, o porta-esmeriladora, o maçarico para corte oxiacetilênico, a pistola para limpeza por jato d'água, a pistola para corte por jato d'água e etc[10].

2 Objetivo

O objetivo final do projeto é projetar e construir uma ferramente que possa ser acoplada a um braço robótico articulado que seja capaz de realizar impressões de baixa resolução em superfícies. Para isso será usado um cartucho de tinta ink-jet de 72 dpi de apenas uma cor. A ferramenta deve possuir um microcontrolador eletrônico devidamente programado para realizar os desenhos. O robô se comunicará com a ferramenta que possuirá uma programação embarcada própria, sendo a programação do robô ajustada, e a programação da ferramenta ajustada apenas para tarefas de maiores complexidades. A ferramenta deve apresentar grau de resolução válido para imagem maiores, ser inofensiva ao meio ambiente e utilizar um pequeno volume de tinta.

3 Metodologia

Por se tratar de uma ferramenta ainda em desenvolvimento na indústria foi escolhido o mesmo caminho que a Airbus tem usado para desenvolver mencionado na referência [4], a impressão ink-jet. Embora cartuchos ink-jet não possuam grande variedade de bibliografia que expliquem sua operação, é possível encontrar fontes que detalham o comportamento básico.

De acordo com a referência [5] os cartuchos de tinta das impressoras HP necessitam de uma tensão de 20V para produzir a pressão mecânica necessária para o jato de tinta.

3.1 Circuito Eletrônico

O Arduino Uno é um microcontrolador vastamente usado em estudos acadêmicos embora não seja usado em produção industrial. Porém é uma ferramente útil para o estudo do desenvolvimento de um produto devido a facilidade de manuseio.

O primeiro empecilho apresentado ao usar o Arduino para ser o controlador de uma peça que realize impressão é que o Arduino trabalha em uma tensão de 12V, assim como grande parte dos componentes eletrônicos. Para se contornar o problema foi usado o Inkshield [6].

O Inkshield é um shield desenvolvido em um trabalho de graduação da University of Washington. Para desenvolver impressoras DIY (do-it-yourself) alunos da universidade usaram um shield capaz de aumentar a tensão de saída do microcontrolador para que se possa ser utilizado o Arduino no controle de cartuchos ink-jet. Através de capacitores e indutores, o shield é capaz de produzir uma tensão de 20V. O Inkshield pode ser facilmente acoplado no Arduino e utilizado desde que o Arduino esteja conectado a uma fonte de 12V, sendo uma tensão menor insuficiente para o seu funcionamento. O Inkshield é um hardware open source com todo seu funcionamento detalhado na referência [6].

O circuito é representado na figura 2, sua operação que produz o ganho de tensão de 8V utilizando 3 capacitores, 1 indutor, 4 resistores, 1 LED tipo zenner e 1 amplificador operacional.

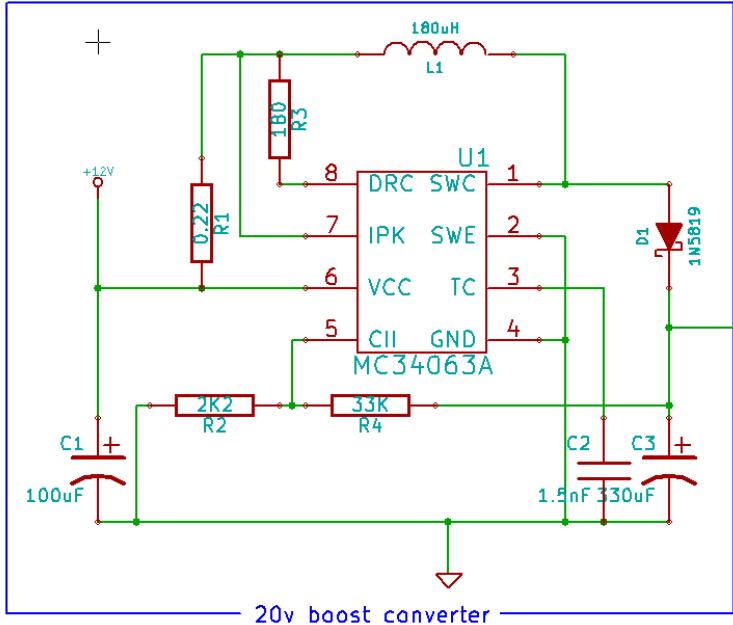


Figura 2: Circuito de ganho de tensão utilizado pelo Inkshield, imagem retirada da ref.[6].

O circuito da Figura 2 produz a tensão de 20V necessária. Além do problema analógico, o Inkshield também resolve um problema digital referente aos pulsos eletrônicos que o cartucho necessita.

O cartucho utilizado no projeto será o HPC6602 que possui 12 pinos de emissão de jato de tinta. Isso implica a necessidade de no mínimo 12 saídas de sinais para o completo acionamento. No entanto, o Inkshield utiliza demultiplexadores para diminuir o número de saídas necessária do microcontrolador. Embora o Arduino apresente 18 saídas digitais, utilizar 12 saídas restringiria o número de recursos a serem adicionados à placa, e a placa necessitaria de mais saídas mesmo para operar apenas o cartucho.

Foi então utilizado um demultiplexador de 4 bits que pode produzir até 16 saídas. Um pequeno *delay* é ocasionado no acionamento dos 12 pinos, no entanto sua duração tem caráter desprezível. Assim, por uma especificação do projeto, as entradas analógicas do arduino foram convertidas em saídas digitais, representadas pelas letras A, B, C e D. O ”clock” do pulso é ajustado pelo controlador sendo emitido por um dos 13 pinos de saída digitais. O Inkshield utiliza um CI CD4067 e dois ULN2803 para enviar o sinal ao cartucho. Como é possível acionar até 16 saídas, foi criada uma saída auxiliar para acionamento de outros cartuchos que não será utilizada no projeto e uma saída para o controle de um cartucho de 12 pinos. Cada CI ULN2803 tem capacidade de envio de 8 pulsos digitais, sendo necessário dois para a utilização, enquanto o CI CD4067 possui 16 saídas. O datasheet do CI ULN2803 e a tabela verdade do CI CD4067 são exibidos na seção de apêndice.

A referência [6] mostra como é feita a ligação do multiplexador nos dois CIs queenviam o sinal para a placa. As duas imagens seguintes revelam o diagrama digital e a resposta esperada para o sistema.

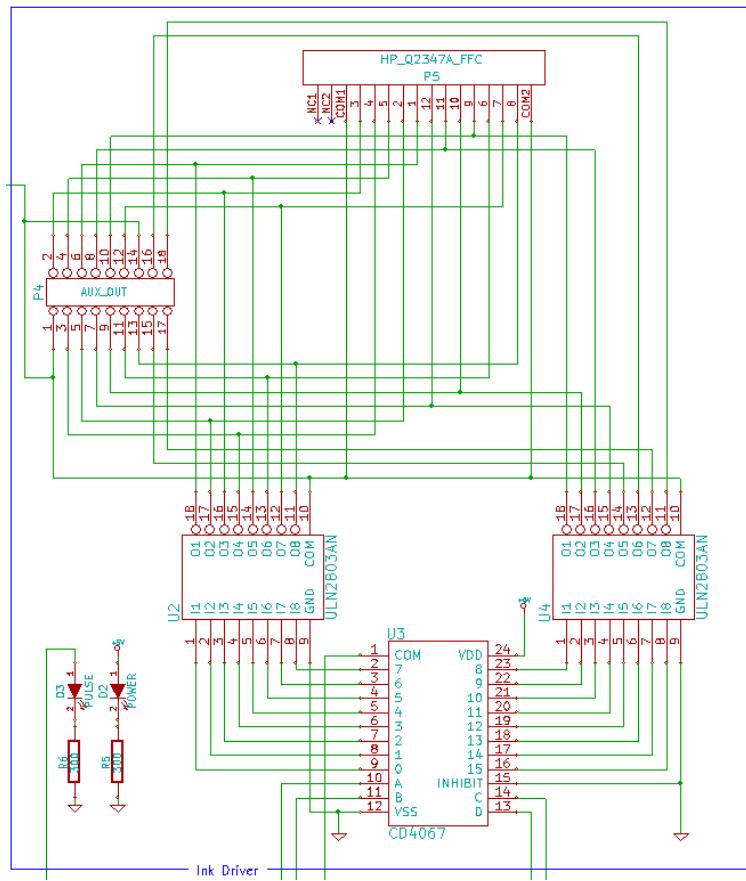


Figura 3: Diagramação lógica usando um demultiplexador de 4 para 16 bits acionando o cartucho HP C6602, imagem retirada da ref.[6].

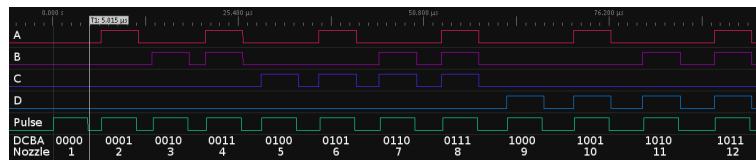


Figura 4: Simulação da operação do multiplexador recebendo 4 sinais de entrada do Arduino, imagem retirada da ref.[6].

Como mostrados nas figuras 3 e 4 o Inkshield receberá 5 sinais de entrada. Os pulsos A, B, C e D serão responsáveis pela seleção dos pinos que emitirão jatos de tinta mais uma entrada COM que funcionará como o ”clock” de acionamento.

No entanto, para esse projeto o Arduino receberá mais uma entrada referente a posição do robô manipulador. A princípio foi pensando em adotar mais duas entradas, uma referente ao robô e outra referente ao sensor de posição porém o sensor foi descartado devido a necessidade de uma precisão além dos sensores ultrassônicos de arduinos atuais. O arduino receberá uma entrada digital do robô manipulador informando o momento de iniciar a impressão, essa entrada determinará o funcionamento da saída COM do Inkshield. A nova entrada não altera o InkShield necessariamente, pois poderia ser inserida diretamente no Arduino e sua implementação é mais expressiva na parte de programação.

Assim no projeto foi utilizado as saídas digitais de A2-A5 do arduino para acionamentos dos pinos e saída digital 2 para a saída COM. A entrada digital 8 foi usada para receber a resposta do robô manipulador.

Para a construção do InkShield foram usados os seguintes dispositivos:

Tabela 1: Componentes utilizados.

1 LED Verde
1 LED Vermelho
1 Diodo 1N5819
1 Capacitor de $1.5\eta\text{C}$
1 Capacitor de $330\mu\text{C}$
1 Capacitor de $100\mu\text{C}$
1 Resistor de 0.22Ω
1 Resistor de 180Ω
1 Resistor de $2\text{k}2\Omega$
1 Resistor de $3\text{k}3\Omega$
1 Indutor de $180\mu\text{H}$
1 CI CD74HC4067
2 CIs ULN2803
1 Amp-op MC34063A
1 push button
Placa de montagem
Pinos conectores
1 conector HP Q2347A FFC

3.2 Programação da Ferramenta

O Inkshield possui uma biblioteca disponível no site de referência para o fácil acionamento dos pinos, dispensando a necessidade de ajuste pinagem e acionamento de componentes eletrônicos. Assim é possível focar apenas na programação para a necessidade específica do projeto. (A programação que define os pinos e os acionamentos estão escrita em C e C++ e são exibidas na referência [6]).

O cartucho permite o acionamento de 12 bocais da maneira desejada, sendo cada acionamento independente. A programação define quais bocais serão acionados e o intervalo entre os acionamentos, podendo ser contínuo. Assim de acordo com os testes e devido a precisão de movimentação do robô, foi verificado que o acionamento de 5 bocais equivale a um espaçamento de 1mm tornando a impressão completa sem linhas falhas no processo.

Foi verificado que a resolução da impressão não é afetada consideravelmente para emissão de jatos de distância de tintas com distâncias menores de 1 cm, logo a peça não necessitou de uma programação capaz de corrigir a posição do robô, que consegue determinar a precisão necessária. No entanto, a variação da distância entre superfície e ferramenta causa dispersamento dos jatos de tinta.

A ferramenta necessita de um tempo de atraso para que o robô atinja a próxima posição, o tempo determinado na programação é de 10ms para avanços lineares na direção xy, ou seja, movimentos que não requisitam uma movimentação no eixo z do robô quando utilizado o referencial cartesiano. A ferramenta foi então ajustada para se locomover em apenas dois eixos, sendo ativada e desativada por um sinal de entrada conectado na entrada 8 do equipamento.

Assim, a porta 8 será utilizada como a condição de acionamento da peça. A programação da peça apenas começará a operar quando o robô informar que está na posição desejada para imprimir. Assim o robô deve ter controle sobre quando realizar a impressão e quando deve fazer movimentos de deslocamento sem dispensar tinta.

A programação da ferramenta pode ser curtamente expressa pelo algoritmo:

```
//Impressao Robotizada

#include <InkShield.h> //biblioteca instalada

InkShieldA2A5 MyInkShield(2); //configuracao dos pinos a2a3a5a5
                                //sinal de saida para o cartucho no 2

const int controle = 8; //entrada do sinal I/O

void setup()
{
    pinMode(controle, INPUT); //tipo de entrada
}

void loop()
{
    boolean val = digitalRead(controle); //le posicao do I/O
    if(val == LOW)
```

```

{
    MyInkShield.spray_ink(0b0000000000000011); //bin
    delay(10); //testado
}

//fim do loop

}

```

A programação exibida realiza todos os comandos necessários para a operação da peça. Ao definirmos InkShieldA2A5 MyInkShield(2) dizemos que o hardware da peça está configurado para receber as saídas digitais 2, 3, 4 e 5 (convertidas das entradas analógicas para saídas digitais) do arduino como definição dos pinos de acionamentos e a saída 2 como o pulso COM. O controle é definido como uma entrada na porta digital 8. A função “*MyInkShield.spray_{ink}*” define quantos pinos serão acionados, nesse caso apenas as saídas 0 e 1 receberão sinal lógico alto, a representação binária foi adotada. O delay determina o tempo de atrasado definido experimentalmente para o robô avançar linearmente em micro-segundos. A condição de acionamento fica em loop infinito uma vez que a peça foi ligada.

Para essa programação foi utilizado o nível lógico alto como o sinal de entrada para desligar a ferramenta enquanto o nível baixo aciona o equipamento.

3.3 Projeto Mecânico

Foi idealizado um projeto de uma peça para armazenar o as placas, o cartucho e conectar os fios necessários para sua operação.

O projeto respeita as condições:

- Possuir dimensão compatível com o robo ABB IRB 1600.
- Permitir fácil troca de cartucho.
- Evitar movimento dos componentes internos com a movimentação do robô.
- Entradas de cabos de sinais e fonte de tensão.
- A ferramenta não deve tocar a superfície a ser realizada a impressão.

Os seguintes *sketches* mostram o projeto da ferramenta.

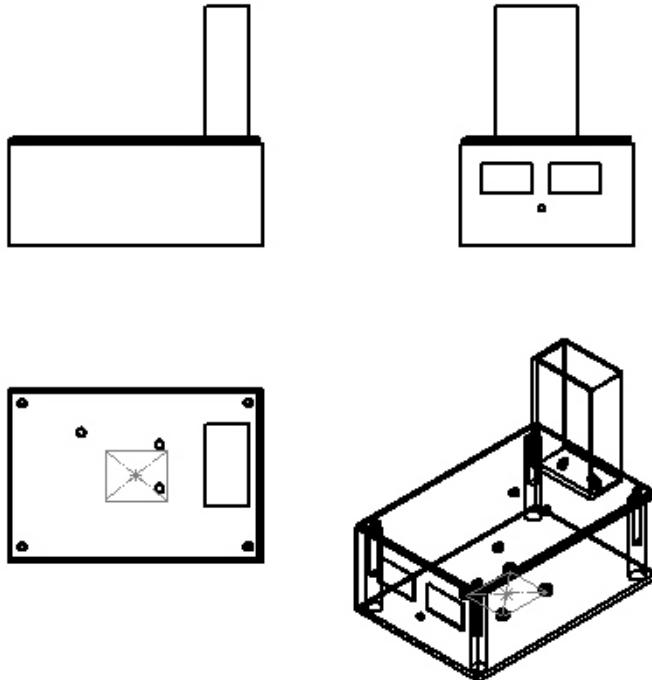


Figura 5: Vistas ortogonais e em perspectiva isométrica da ferramenta.

Como exibido no desenho a ferramenta é constituída por uma caixa que armazena os componentes eletrônicos e que armazena o cartucho. As aberturas laterais são para conexão do cabo de programação, da fonte de alimentação e do cabo I/O do controlador do robô. Os furos representam os lugares onde o microcontrolador será fixado e onde a ferramenta será parafusada. A caixa que armazena o cartucho deve ser parafusada na parte inferior da caixa que armazena os componentes eletrônicos, a parte inferior parafusada

na parte superior. A parte superior deve conter furos para ser parafusada ao braço robótico. Um corte entre a caixa que armazena o cartucho e o a parte inferior é feito para a passagem do cabo *flat*.

A ferramenta final deve ser semelhante as figuras 6, 7, 8 e 9:

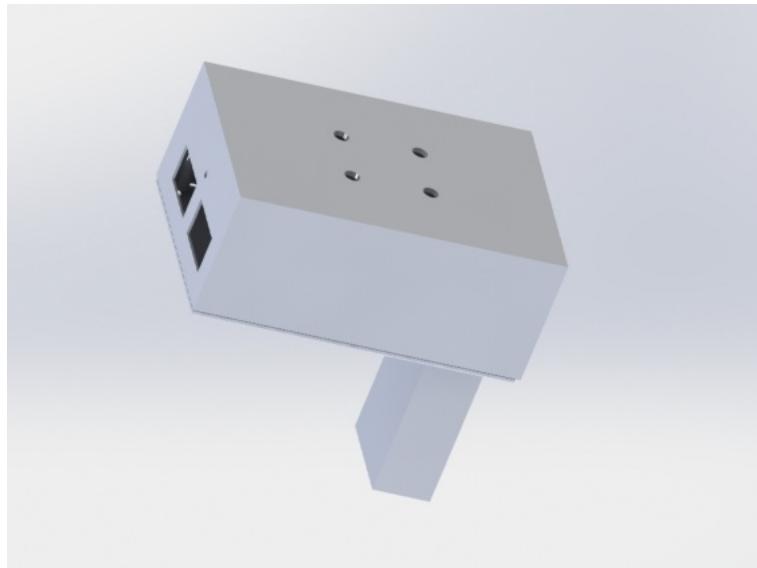


Figura 6: Visualização do compartimento. Fonte: autores.

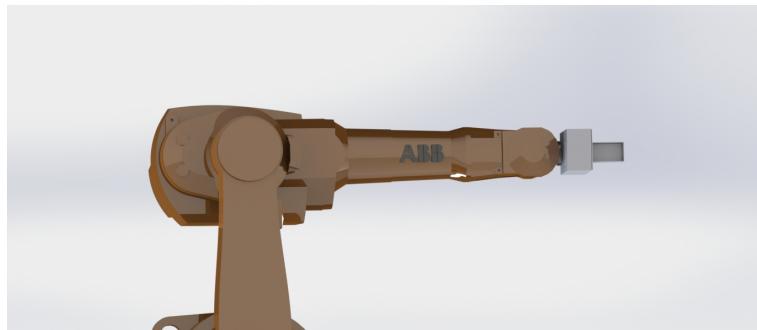


Figura 7: Visualização lateral da ferramenta acoplada. Fonte [7] e autores.

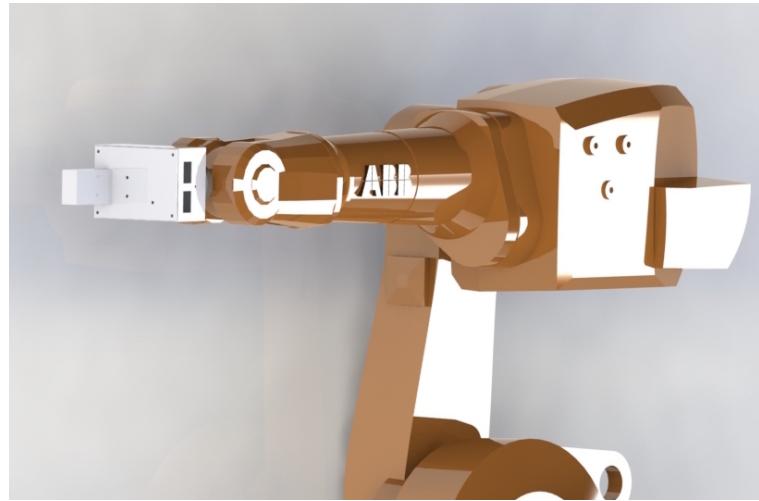


Figura 8: Visualização da ferramenta acoplada. Fonte [7] e autores.

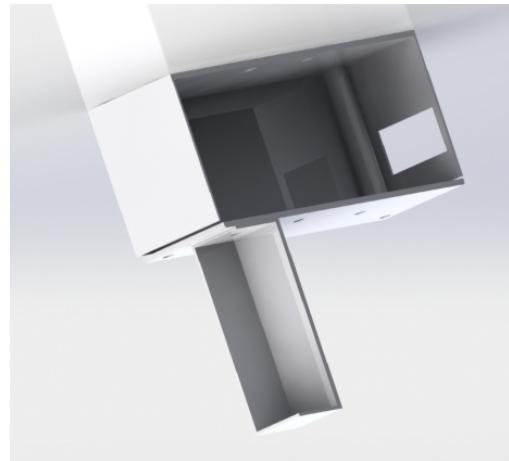


Figura 9: Visualização interna do compartimento. Fonte: autores.

Os arquivos CAD do robô ABB IRB 1600 foram retirados da referência [7].

3.4 Programação do Robô Manipulador IRB 1600

O robô deve ser capaz de se movimentar com acurácia para imprimir os pontos no local necessário. Mesmo uma imagem de baixa resolução requisitará um grande número de pontos para atingir um padrão satisfatório. É interessante pensar em desenhos de difíceis reprodução pela pintura em camadas para que o robô possam realizar com maior velocidade.

Para o desenvolvimento da ferramenta, utilizou-se as entradas I/O do controlador, assim foi configurado na programação do robô o envio de um sinal para que a ferramenta seja acionada no momento certo. Dessa forma, a ferramenta permanece desativada enquanto o robô realiza movimentos de troca de posição inicial. O sinal utilizado pelo controlador IRC5 da ABB foi o "DO10_01" que representa o sinal de saída 1 no modo I/O. O comando usado é o SetDO, 1 para acionar e o SetDO, 0 para desativar a saída digital. Lembrando que o acionamento do aparelho se dá pela lógica inversa de entrada.

Para detectar os pontos, foi utilizado o OPENCV para fazer uma leitura de pixels da imagens e informar os pontos de maneira cartesiana. De posse dos pontos, é possível programar em RAPID a programação do robô articulado e o acionamento da ferramenta.

Assim foi criada uma programação em C++ que realizasse a programação em RAPID automaticamente. A programação é exibida abaixo:

```
//  
//  main.cpp  
//  tokyo  
//  
//  Created by Anderson Carlos dos Santos on 2017-02-10.  
//  Copyright 2017 Anderson Carlos dos Santos. All rights reserved.  
  
#include <iostream>  
  
#include <opencv2/core/core.hpp>  
#include <opencv2/highgui/highgui.hpp>  
#include <opencv2/imgproc/imgproc.hpp>  
  
using namespace std;  
using namespace cv;  
  
int main(int argc, char** argv) {  
  
    int idx;  
    int idy;  
  
    int tempy = 0;  
    int tempx = 0;  
  
    Mat img = imread("/Users/Anderson/Desktop/tokyo9.jpg");
```

```

Mat gray_img;
Mat bin_img;

cvtColor(img, gray_img, CV_BGR2GRAY);
threshold(gray_img, bin_img, 213, 200, CV_THRESH_BINARY_INV);

namedWindow("img", WINDOW_NORMAL);

imshow("img", img);
//resizeWindow("img", 700, 700);
waitKey(0);

imshow("img", gray_img);
//resizeWindow("img", 700, 700);
waitKey(0);

imshow("img", bin_img);
//resizeWindow("img", 700, 700);
waitKey(0);

Mat nonZeroCoordinates;
findNonZero(bin_img, nonZeroCoordinates);
for (int i = 0; i < nonZeroCoordinates.total(); i++) {

    idy = nonZeroCoordinates.at<Point>(i).y;
    idx = nonZeroCoordinates.at<Point>(i).x;

    if (idy!=tempy && tempy==0)
    {

        cout << "MoveL_Offs(p10," <<
        nonZeroCoordinates.at<Point>(i).x << ","
        << nonZeroCoordinates.at<Point>(i).y <<
        ",0),v30,fine,tool0;" << endl;

        tempy = idy;
        tempx = idx;

        cout << "SetDO_DO10_1,-0;" << endl;
    }
}

```

```

    }

    if( idx!=tempx && tempy!=0)
    {

        cout << "MoveL_Offs( p10 , " <<
        nonZeroCoordinates .at<Point>(i -1).x <<
        " , " << nonZeroCoordinates .at<Point>(i -1).y <<
        " ,0 ) ,v10 , fine , tool0 ;"
        << endl;

        tempx = idx ;
        tempy = 0;

        cout << "SetDO_DO10_1 , -1;" << endl;

    }

    tempx++;

}

return 0;
}

```

A programação é capaz de realizar o processamento de imagem requerido e detectar os pixels pretos da imagem convertida. O programa visualiza os pontos pixels negros da imagem binária e os armazenam em uma matriz, após a detecção a programação remove os pixels contínuos separando o código RAPID em dois processos, movimentos de acionamento da ferramenta e movimentos de deslocamento de posição. A programação do robô articulado resulta em um programa de mais de 9000 linhas e não será exibida na monografia. No entanto, a programação é semelhante ao trecho abaixo, retirado da programação final usada para a figura exibida nos resultados:

```

SetDO DO10_1 , 1;
MoveL Offs( p10 ,163 ,45 ,0 ) ,v30 , fine , tool0 ;
SetDO DO10_1 , 0;
MoveL Offs( p10 ,193 ,45 ,0 ) ,v10 , fine , tool0 ;
SetDO DO10_1 , 1;
MoveL Offs( p10 ,162 ,46 ,0 ) ,v30 , fine , tool0 ;
SetDO DO10_1 , 0;
MoveL Offs( p10 ,194 ,46 ,0 ) ,v10 , fine , tool0 ;
SetDO DO10_1 , 1;

```

De acordo com o algoritmo, a ferramenta é acionada após o comando *SetDO DO10_1, 0* realizando o trabalho de impressão e desativada após o comando *SetDO DO10_0, 1* para o robô realizar um movimento de

deslocamento.

As velocidades usadas ”v10” e ”v30” foram de acordo com as limitações apresentadas pelo robô no laboratório afim de evitar vibrações, é válido lembrar que o robô não se encontra nas instalações ideais reduzindo a velocidade usada. A precisão ”fine” é necessário uma vez que foi usado a detecção de pixels e uma precisão milimétrica.

4 Resultados

Os componentes necessários para a construção de um InkShield foram adquiridos. A lista de materiais necessários são exibidos na tabela 1.

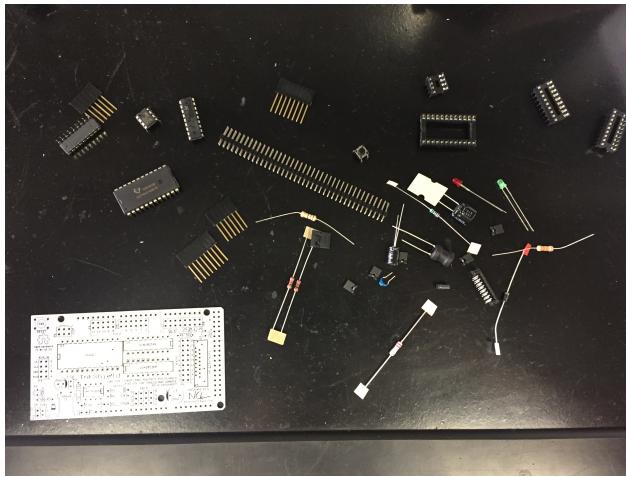


Figura 10: Componentes eletrônicos utilizados para montar o Inkshield.

Com o shield montado, podemos conectá-lo ao Arduino e ao cartucho HP C6602. A peça final conectada ao Arduino é exibido na imagem abaixo.

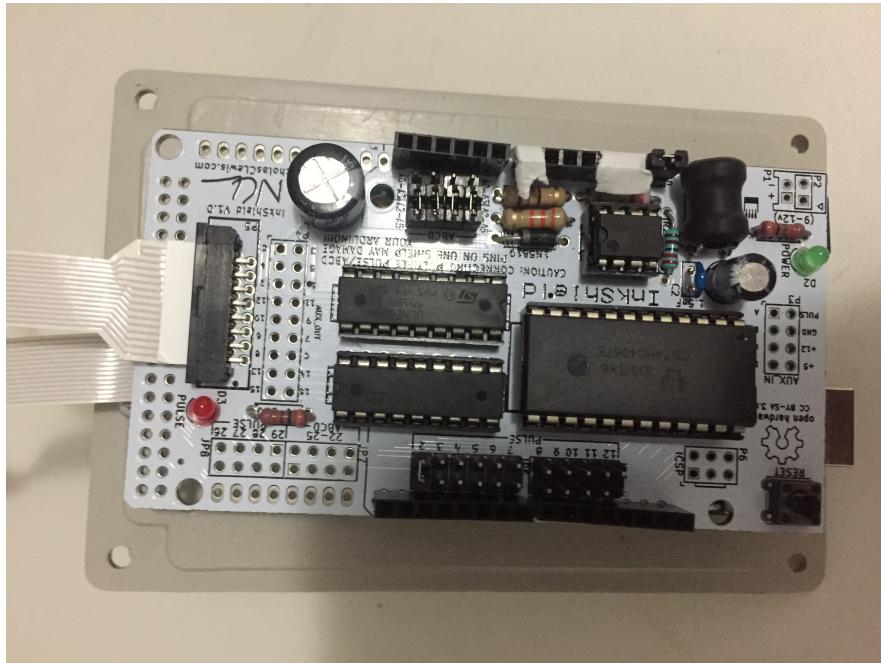
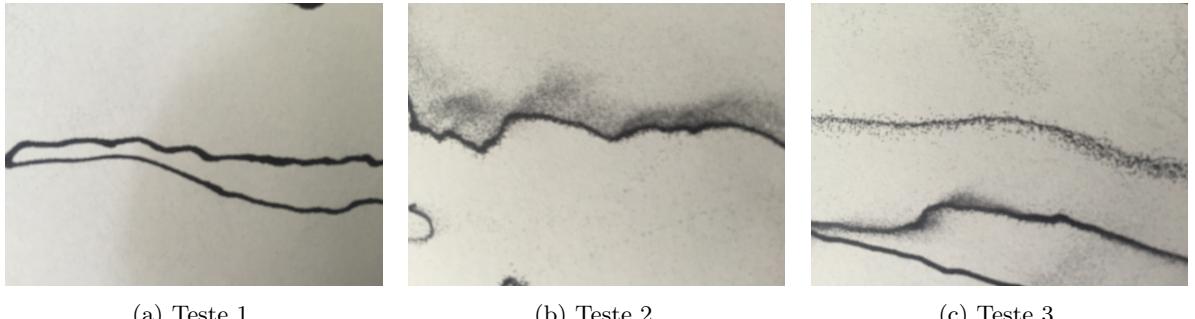


Figura 11: InkShield conectado ao Arduino.

A programação ajustada será igual a informada na seção 3.2. Dependendo do desenho a ser impresso o software da peça poderá ser alterado. Para as especificações previamente adotadas temos o seguintes traços impressos:



(a) Teste 1.

(b) Teste 2.

(c) Teste 3.

Figura 12: Teste realizados com o cartucho manualmente.

A figura 12 exibe o comportamento do jato de tinta variando a distância entre o papel e o cartucho. Ao aumentarmos essa distância aumentamos também a dispersão dos pontos, ao mesmo tempo tornamos a impressão mais suscetível a borrar.

Apesar de o teste manual ter apresentado variações, a programação do robô não deve apresentar movimentos bruscos durante a impressão nem um erro superior a 1 cm de distância, assim o uso de um sensor ultrassônico foi desconsiderado.

Como a ferramenta não deve tocar a superfície e os componentes usados interiormente são leves, não foi necessário o uso de materiais com maiores resistências, sendo o plástico suficiente. Assim, seguindo o modelo proposto na parte 3.3, foi confeccionado a ferramenta.

A ferramenta projetada deve a adição de um divisor de tensão na lateral devido a saída do controlador I/O produzir um sinal de 24V, e o equipamento eletrônico da ferramenta permite sinais de entrada de no máximo 5V. Assim o seguinte divisor de tensão foi instalado na ferramenta, completando o circuito final dos equipamentos.

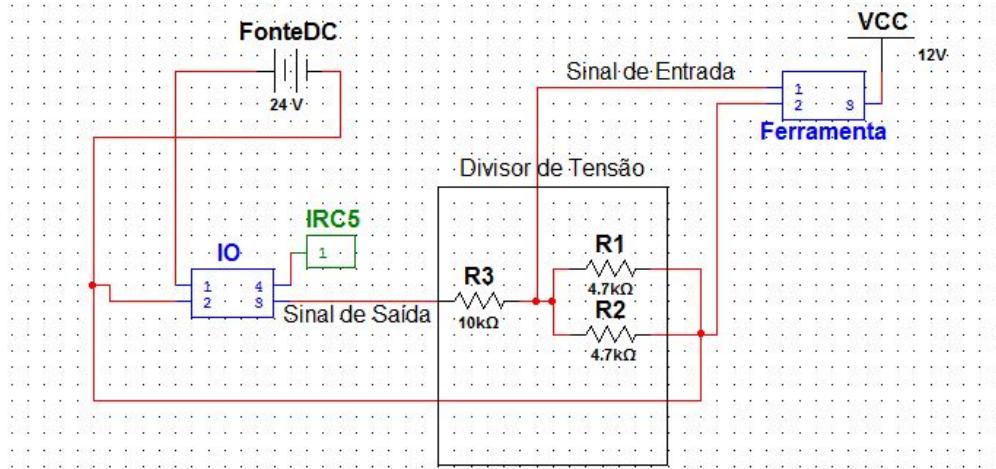


Figura 13: Circuito de conexão entre os dispositivos.

A versão final da ferramenta é exibida nas figuras 14, 15 e 16:



Figura 14: Visão superior da ferramenta.



Figura 15: Visão frontal da ferramenta.

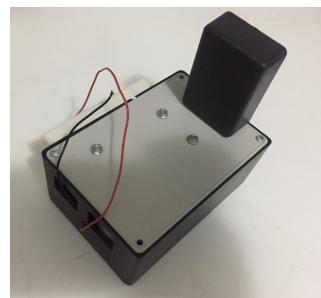


Figura 16: Visão inferior da ferramenta.

Assim, após a ferramenta ser programada como descrito na seção 3.2, a foi feita a conexão dos dispositivos, sendo a ferramenta acoplada no robô.

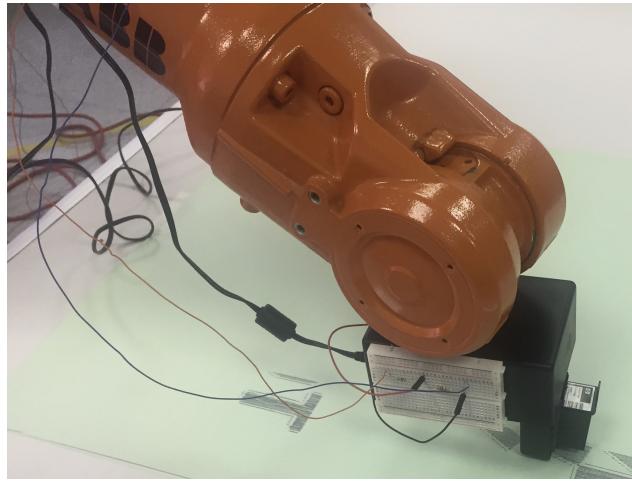


Figura 17: Ferramenta acoplada no robô realizando testes.

Após conectar os equipamentos como na figura 13, o processo de impressão pode ser iniciado.

Como mencionado no seção 3.4, a programação do robô foi automatizada através do algoritmo demonstrado. A imagem escolhida foi o símbolo das Olimpíadas de Tóquio em 2020. Como o projeto foi pensado principalmente nos problemas da indústria *livery*, foi então escolhido um símbolo que provavelmente deve estampar aviões e outros meios de divulgação e celebração nos próximos anos devido a grande importância do evento. O algoritmo utilizado no processamento de imagem é genérico podendo fazer a automatização da programação de qualquer imagem simulando um movimento de impressão. O símbolo escolhido é exibido adiante.



Figura 18: Símbolo das Olimpíada de Tóquio em 2020.

O símbolo é composto principalmente por vários retângulos que se encaixam e formam um globo com específicos ângulos entre eles. Embora a parte principal possua apenas uma cor, os anéis olímpicos são coloridos, e mesmo assim a cor não é o preto. Para ter a melhor detecção dos pixels foi verificado que a imagem deveria passar por três processos; vetorização, conversão preto e branco, conversão binária. A imagem binária é a que fornece melhor resolução no detectamento de pixel. A imagem deve estar ajustada para o tamanho que se deseja realizar a impressão direta, a redução da imagem pode causar perda de resolução devido a precisão de deslocamento do robô. Assim, para esse caso, quanto maior o desenho, maior resolução ele terá. A seguir é representado o processo que a imagem passa no algoritmo.



Figura 19: Imagem vetorizada.



Figura 20: Imagem em preto e branco.

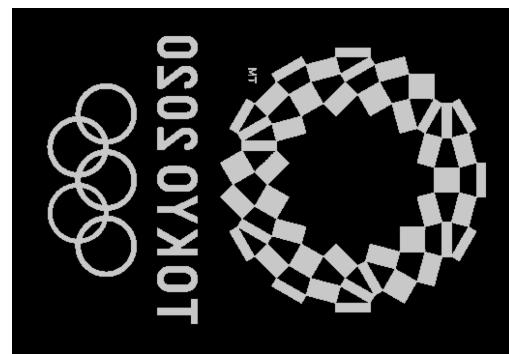


Figura 21: Imagem binária.

As imagens devem estar espelhadas e preferencialmente na vertical para que a detecção dos pontos ocorra como desejado. Essa característica provavelmente se dá por conta dos algoritmos presentes da biblioteca do Opencv. A última imagem é mostrada com as cores invertidas em relação a anterior, esse processo ocorre porque para o opencv pixels cheios são representados pela cor branca enquanto pixels vazios são representados pela cor preta.

Uma vez que todos os parâmetros estão iguais aos exibidos nessa monografia, que foram identificados através de diversos testes, o equipamento deve ser capaz de realizar a impressão robotizada direta. As próximas figuras demonstram o processo e o resultado final, que requisita um total de aproximadamente três horas para ficar completo.

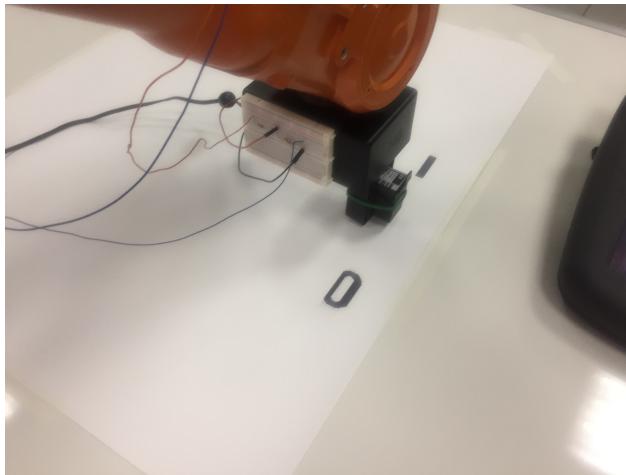


Figura 22: Começo do processo de impressão robotizada.

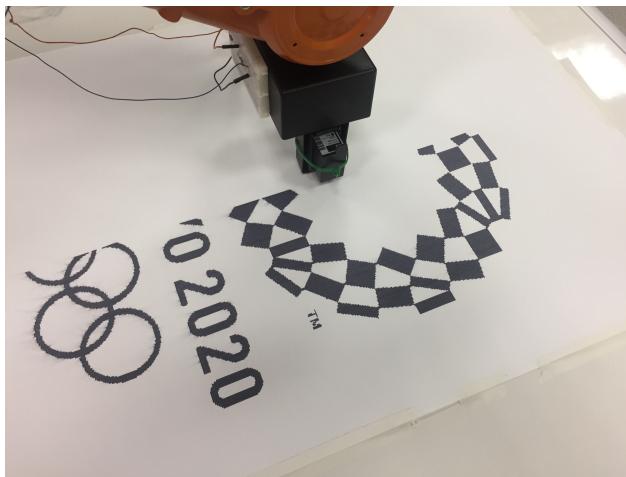


Figura 23: Metade do processo de impressão robotizada realizado.



Figura 24: Resultado final do processo de impressão robotizada.

A figura 24 também exibe alguns problemas, como os borrados e uma pequena perda de resolução. O borrado acontece devido a mesa utilizada no laboratório não ser linear, possuindo diferenças entre seus extremos. Para que a ferramenta não tocasse o objeto, foi configurado a posição do robô a partir do ponto de maior altitude da mesa, esse ponto localizava-se na parte superior do desenho, onde a figura apresenta menos borrados, a parte inferior o cartucho ficava mais distante ultrapassando 1 cm de distância da superfície causando maior dispersão do jato de tinta. A perda de resolução acontece pela diminuição da imagem e devido a precisão de deslocamento do robô como já comentado.

A resolução da reprodução poderia ser aumentada de duas formas, aumentando o tamanho da reprodução ou possuindo robôs com maiores precisão de deslocamento. O processo ideal seria o acionamento de um bocal por ponto capturado, representando 1 pixel por deslocamento. O deslocamento do robô IRB 1600 é milimétrico, na imagem reproduzida cada ponto capturado representa 1 mm, assim se tem 20% da resolução possível, necessitando então de um robô com deslocamento de precisão de 0.2mm para uma perfeita reprodução em uma superfície de 50x66cm ou ampliar em 5 vezes a área da figura. É possível que uma programação mais sofisticada da ferramenta, permitindo que a ferramenta decida quais bocais acionar de acordo com o ponto capturado da imagem, permita uma resolução bastante superior. Essa é a tecnologia usada em impressoras, no entanto sua implementação exigiria um sistema mais sofisticado de comunicação entre controlador e ferramenta.

O tempo de três horas pode ser otimizado. No atual projeto, a ferramenta realiza impressões encontrando os pontos a partir de um eixo da posição vertical, assim a ferramenta volta sempre à primeira linha de determinada coluna, ou ao primeiro ponto no eixo x de um respectivo valor em y. A programação do robô poderia ser incrementada para imprimir “indo-e-voltando” ao invés de apenas “indo”. As velocidades do robô também poderiam ser aumentadas substituindo as condições que se encontra no laboratório. Porém, uma precisão maior poderia exigir um *delay* maior na programação da ferramenta, porém esse *delay* poderia ser compensado com o ajuste da velocidade do braço robótico. Assim, o tempo é uma variável dependente de cada projeto a ser executado mas com grande margem a ser reduzido.

5 Conclusão

O processo foi capaz de realizar impressões com um nível satisfatório em superfícies. A superfície no caso era uma folha de cartolina de 50x66cm, no entanto qualquer superfície plana que permitisse a adesão da tinta utilizada poderia ter sido utilizado.

O trabalho mostra o princípio do desenvolvimento completo de uma ferramenta que possa realizar os trabalhos de *liveries* através de sistemas automatizados. Um circuito mais sofisticado poderia utilizar mais cartuchos, um novo algoritmo poderia detectar pixels de acordo com o padrão RGB e adaptação de sensores e uma nova programação poderia até permitir o trabalho em superfícies curvas.

Assim, realizando impressões diretas de uma cor em superfícies planas o processo de impressão robotizada resulta em uma automação eficiente desde que seja adotadas tecnologias de cartucho inkjet combinadas ou não com robôs industriais. O trabalho propõe o uso dos robôs industriais pela ausência da necessidade de se construir atuadores mecânicos próprios para o serviço, no entanto como referenciado, a Airbus obteve sucesso construindo seus atuadores. O trabalho não considerou as dificuldades que se encontraria com a química da tinta, sendo que diferentes tipos de tinta podem exigir cartuchos diferentes, nem outros problemas técnicos como o de acúmulo de energia estática na pintura. Desde que seja aperfeiçoado o circuito, tornando-o capaz de reproduzir diferentes cores e aperfeiçoada a programação do robô para desenho em superfícies curvas, o sistema deverá ser capaz de reproduzir com êxito qualquer imagens de grande dimensão em diferentes tipos de superfícies.

6 Referências

- [1] Cruz, G. "A História da Robótica até os dias de hoje", Ciência e Tecnologia. 2013. Acessado em 20/08/2016, disponível em:
< https://cienciaetecnologias.com/robotica-historia/ >
- [2]Riascos, L. M. "Fundamentos de Robótica: manipuladores e robôs móveis", São Paulo, 2010. Ed. Pleiade
- [3]"Canada Arm", Canadian Space Agency (CSA). 2015. Acessado em 20/08/2016, disponível em:
< http://www.asc-csa.gc.ca/eng/canadarm/ >
- [4]"Think ink: Airbus revolutionises how an airline's livery is applied to jetliners", Airbus, 2014. Acessado em 20/08/2016, disponível em:
< http://www.airbus.com/newsevents/news-events-single/detail/think-ink-airbus-revolutionises-how-an-airlines-livery-is-applied-to-jetliners/ >
- [5]Gilliland, M. "Inkjet Applications", 2005. Ed. Woodglen Press
- [6]Lewis, N. C. "InkShield", Nicholas C Lewis. 2012. Acessado em 20/08/2016, disponível em:
< http://nicholaslewis.com/projects/inkshield/theory/ >
- [7]"IRB CAD Models", ABB. 2017. Acessado em 03/03/2017, disponível em:
< http://new.abb.com/products/robotics/industrial-robots/irb-1600/irb-1600-cad >
- [8]ZUTA Labs. "The first mini robotic printer". Disponível em: *< http://www.zutalabs.com/ >*. Acessado em 03/03/2017.
- [9]Maurer, R, "Airbus Innovation Days: Direct Printing Technology", Airbus. Maio de 2016.
- [10]Romano, V. F. "Robótica Industrial: Aplicação na indústria de manufatura e de processos", 2002. Ed. Edgar Blucher.

7 Apêndice

7.1 Componentes Eletrônicos

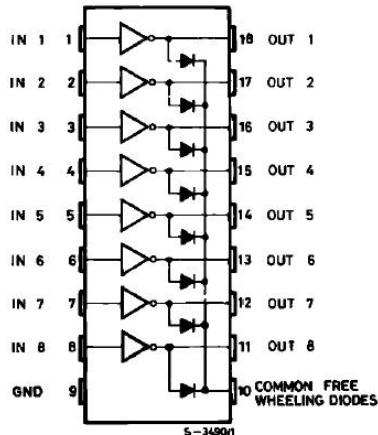


Figura 25: Datasheet do CI ULN2803 usado no shield.

TRUTH TABLE					
S0	S1	S2	S3	E	SELECTED CHANNEL
X	X	X	X	1	None
0	0	0	0	0	0
1	0	0	0	0	1
0	1	0	0	0	2
1	1	0	0	0	3
0	0	1	0	0	4
1	0	1	0	0	5
0	1	1	0	0	6
1	1	1	0	0	7
0	0	0	1	0	8
1	0	0	1	0	9
0	1	0	1	0	10
1	1	0	1	0	11
0	0	1	1	0	12
1	0	1	1	0	13
0	1	1	1	0	14
1	1	1	1	0	15

H= High Level
L= Low Level
X= Don't Care

Figura 26: Tabela verdade do demultiplexador CD74HC4067.