

# Desenvolvimento de módulos computacionais para plataforma robótica aplicada à educação técnica

*Development of computational modules for robotics platform applied to technical education*

**Antonio Valerio Netto** <sup>1</sup>  [orcid.org/0000-0001-9215-8531](https://orcid.org/0000-0001-9215-8531)

<sup>1</sup> Bolsista DT CNPq, Brasília, Brasil.

**E-mail do autor principal:** [antonio.valerio@pq.cnpq.br](mailto:antonio.valerio@pq.cnpq.br)

## Resumo

---

O artigo descreve o desenvolvimento de módulos computacionais para serem empregados na área de tecnologia educacional. O foco principal foi o ensino técnico tendo como parceiro educacional a escola SENAI de São Carlos (SP). A relevância do projeto consiste na combinação de tecnologias para o aprimoramento da plataforma robótica nacional, RoboDeck. E no desenvolvimento de um material didático que contribua com o processo de aprendizagem durante as aulas práticas dos cursos técnicos de eletrônica, mecatrônica e computação do SENAI-SP. Esse material foi criado para compor um kit educacional e auxiliar o professor a direcionar as atividades práticas usando a plataforma robótica móvel. Este material teve como foco o ensino de três disciplinas: robótica, eletrônica digital e linguagem de programação.

**Palavras-Chave:** Tecnologia educacional; robótica móvel; ensino técnico; RoboDeck.

## Abstract

---

*The article describes the development of computational modules to be used in the area of educational technology. The main focus was technical education, having as educational partner the SENAI school in São Carlos (SP). The relevance of the project is the combination of technologies for the enhancement of the national robotic platform, RoboDeck. And in the development of a didactic material that contributes to the learning process during the practical classes of SENAI-SP's electronic courses, mechatronics and computing. This material was created to compose an educational kit and help the teacher to direct the practical activities using the mobile robotic platform. This material focused on the teaching of three disciplines: robotics, digital electronics and programming language.*

**Key-words:** Educational technology; mobile robotics; technical education; RoboDeck.

## 1 Introdução

A robótica é uma grande área da engenharia e um de seus campos chama-se robótica educacional. Trata-se da aplicação da tecnologia no campo pedagógico, sendo mais um instrumento que oferece aos alunos e professores do ensino técnico, a oportunidade de vivenciar experiências semelhantes às que terão na vida real, dando a estes a chance de solucionar problemas, mais do que somente observar formas de solução.

A robótica educacional tem grande potencial como ferramenta interdisciplinar, visto que a construção de um novo mecanismo, ou a solução de um novo problema, frequentemente extrapola a sala de aula. Na tentativa natural de buscar uma solução, o aluno questiona professores de outras disciplinas que podem ajudá-lo a encontrar o caminho mais indicado para a solução do seu problema. A robótica, então, assume o papel de uma ponte que possibilita religar fronteiras anteriormente estabelecidas, agindo como um elemento de coesão dentro do currículo das escolas técnicas [1].

Em linhas gerais, a robótica educacional é servida por várias soluções de acordo com a faixa etária e do contexto pedagógico que se deseja trabalhar. Existem brinquedos pedagógicos com eletrônica de controle, kits educacionais com foco em alunos do ensino fundamental e ensino médio. Conteúdo didático e competições utilizando kits de montagem robótica e até robôs móveis inteligentes de pequeno porte para o nível técnico e de graduação, que também podem ser aplicados em pesquisas por alunos de pós-graduação.

É importante diferenciar os modelos dos kits e as aplicações para não desestimular os alunos e professores, além de transmitir informações equivocadas das possíveis possibilidades do uso da tecnologia, como ferramenta de desenvolvimento educacional ao longo do ciclo acadêmico do aluno que vai desde o ensino fundamental até a pós-graduação. Existem kits robóticos para crianças e existem kits robóticos para os futuros engenheiros. Cada um tem seu grau de complexidade e usabilidade para estimular os jovens aprendizes. Como contexto estratégico, o aprendizado técnico nas áreas de robótica e sistemas embarcados, permite no médio prazo, um diferencial competitivo para o País, pois promove a melhoria do aprendizado

de futuros profissionais para trabalhar em projetos com alto valor agregado [2].

No Brasil existem poucos fabricantes de kits robóticos, um deles desenvolveu uma plataforma chamada RoboDeck, utilizada para educação técnica e pesquisa [3]. O modelo é capaz de se adaptar a projetos tanto de estudantes do ensino médio técnico como também da graduação e pós-graduação. O robô é constituído de um *hardware* básico, formado pelos componentes principais, e um *hardware* opcional, constituído de módulos como câmara, placa de alta performance e por dispositivos que podem ser acoplados a depender da necessidade de cada projeto.

Trata-se de uma plataforma universal e aberta que permite acoplar outros módulos. Isso facilita a pesquisa porque não é necessário partir da "estaca zero" todas as vezes que se for desenvolver um novo projeto técnico. Permite ainda que professores e alunos direcionem os trabalhos diretamente aos módulos tanto de *hardware* como de *software*, em vez de se preocuparem com as funcionalidades básicas. O RoboDeck permite que cada usuário possa programá-lo de diversas formas [4]. A ideia é que a tecnologia para controlá-lo não seja específica, mas sim aberta [5] [6]. No caso dos pesquisadores e hobbistas, esta plataforma pode ser utilizada para o desenvolvimento de aplicações mais sofisticadas. Exemplos de novos módulos e funcionalidades podem ser encontrados em [7], [8] e [9].

O trabalho foi dividido em tarefas que estão relacionadas ao desenvolvimento dos dispositivos de *hardware* e sistemas de *software* que compõem os novos módulos do RoboDeck, além da criação e aprimoramento do material didático a ser aplicado em disciplinas dos cursos de mecatrônica e de eletrônica da rede SENAI-SP [10].

Na Seção 2 do artigo é apresentado o módulo de integração, que viabiliza a utilização dos *shields* desenvolvidos para o Arduino. Esse módulo permite que outros desenvolvedores utilizem a mesma tecnologia de desenvolvimento de *Shields* para criar seus próprios módulos que tanto podem ser utilizados no Arduino Uno quanto na plataforma RoboDeck. Além disso, é possível programar o RoboDeck via interface de desenvolvimento do Arduino. Na Seção 3 é apresentado o módulo de teleoperação que é capaz de permitir que o usuário controle o robô a distância via Internet (não

presencial) ou Wi-Fi (presencial). Além disso, foi desenvolvido um *middleware* para que o RoboDeck possa ser controlado utilizando um servidor de aplicação com acesso a Internet. Na Seção 4 é apresentado o desenvolvimento de uma bancada educacional multidisciplinar baseado na arquitetura do RoboDeck, chamada WorkDeck. Na Seção 5 é apresentado uma visão sobre o trabalho desenvolvido para geração de um material didático para transferência de conhecimento que foi realizado em parceria com a empresa fabricante do RoboDeck e com os professores do SENAI São Carlos (SP). Foram geradas apostilas com os planos de aula específicas de cada uma das três disciplinas da grade do curso técnico de Mecatrônica e Eletrônica. São elas: microcontroladores, tecnologia de comunicação e robótica. Por fim, na Seção 6 são apresentadas as considerações finais referente ao trabalho desenvolvido.

## 2 Módulo de integração com Arduino

Foi desenvolvido um módulo que permitisse executar duas ações distintas junto à plataforma do RoboDeck. A primeira possibilitou que os usuários pudessem acoplar placas eletrônicas com funcionalidades já existentes baseada na arquitetura do Arduino. E a segunda foi utilizar a mesma interface de programação em C/C++ do Arduino para programar diretamente as funcionalidades do RoboDeck. Em linhas gerais, uma placa Arduino é composta por um controlador, algumas linhas de E/S (Entrada/Saída) digital e analógica, além de uma interface serial ou USB (*Universal Serial Bus*) que é utilizado para programação em tempo real. A placa básica não possui recursos mais sofisticados, como por exemplo, conexão para rede Ethernet ou Wi-Fi, porém é comum acrescentar funcionalidades, por meio de placas adicionais chamadas de *shields*.

Inicialmente, foi estudada a arquitetura do microcontrolador Atmel usado nas placas do Arduino Uno para entender como realizar uma correta integração com a plataforma atual do RoboDeck que utiliza o microcontrolador ARM 9. Esta ação foi priorizada, pois um limitante do projeto estava relacionado aos custos que uma placa eletrônica dedicada poderia atingir. Dependendo do valor tornaria inviável algumas aplicações comerciais devido aos custos de produção. Além disso, existem

situações que é adequado que o processamento local (*master*) seja responsável por apenas enviar os comandos já processados para o RoboDeck. Estes comandos são formados por uma quantidade limitada de *bytes* na transmissão que podem ser interpretadas pelo *kernel* do sistema (microcontrolador ARM9) presente no RoboDeck.

O módulo foi desenvolvido considerando a premissa da plataforma robótica universal que norteia o desenvolvimento do RoboDeck. Essa premissa preza pelo desenvolvimento da descentralização do processamento evitando que tudo fique dependente do processador da placa mãe. Assim sendo, o módulo foi projetado para ser um acessório de acoplamento do qual o usuário pode ou não fazer uso. Caso venha utilizar o módulo, o usuário terá entre outras facilidades, uma biblioteca que permitirá o desenvolvimento de outras aplicações baseadas na tecnologia Arduino, e até mesmo, modificá-la por meio da inserção de novos comandos que venham a atender as aplicações que o usuário desejar. Em linhas gerais, o dispositivo tem a proposta de permitir que os usuários de ambas as plataformas possam desenvolver projetos integrados.

Para os usuários que estão habituados com o Arduino é possível utilizar seus conhecimentos para controlar o RoboDeck, ou mesmo, utilizar somente como uma placa para programação compatível com o Arduino Uno para desenvolver seus projetos pessoais ou de faculdade. Para os usuários da plataforma RoboDeck é possível utilizar todos os *shields* disponíveis no mercado que são compatíveis com o Arduino Uno. Na Internet, existem vários *sites* que disponibilizam placas prontas para diversas funções como: *display*, GPRS (*General Packet Radio Service*), *joystick*, etc.

Foram desenvolvidas duas versões do dispositivo, uma básica e outra com funções embarcadas. As duas funções embarcadas são: GPS (*Global Positioning System*) e Acelerômetro. As características principais são: a possibilidade de realizar a programação pela IDE (*Integrated Development Environment*) do Arduino, e a existência de uma função de auto-reset da placa. Além disso, possui *Jumpers* para habilitar e desabilitar o GPS e o acelerômetro. A placa possui uma capacidade de energizar mais *shields* que o Arduino Uno e permite um conector para entrada analógica (0 a 20V). Também possui dois conectores molex para entradas e saídas digitais e para sua

alimentação existem três opções: por meio de um regulador de tensão (6V a 12V), via USB e ou via a alimentação do próprio RoboDeck. Na Figuras 1 e 2 são apresentadas as versões disponíveis ao final do desenvolvimento.

Para a programação da placa, foi utilizada a mesma interface de programação padrão do Arduino que pode ser encontrado no site da comunidade: [www.arduino.cc/en/Main/Software](http://www.arduino.cc/en/Main/Software)

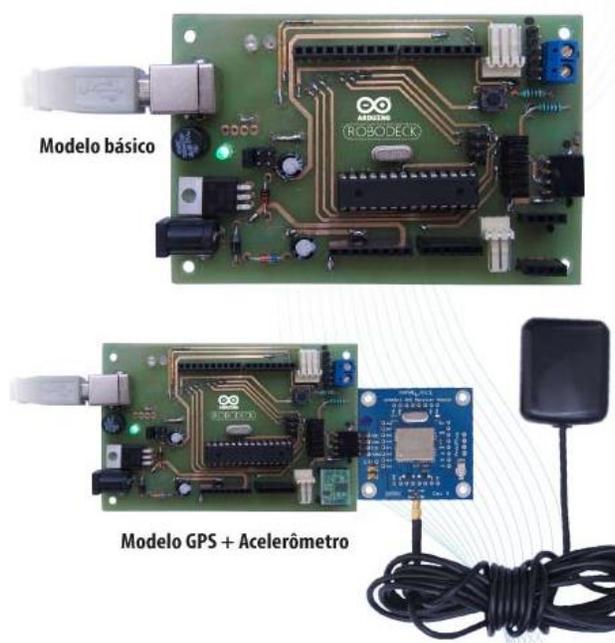


Figura 1: Duas versões referentes ao desenvolvimento de placa de integração baseada em Arduino. Fonte: própria.

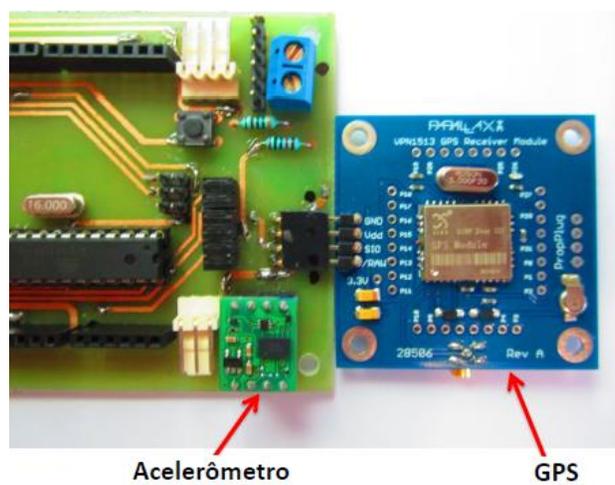


Figura 2: Apresentação dos dois dispositivos da função embarcada. Fonte: própria.

O software para programação do Arduino é uma IDE que permite a criação de *sketches* para a placa Arduino. A linguagem de programação é modelada a partir da linguagem *Wiring*. Quando pressionado o botão *upload* da IDE, o código escrito é traduzido para a linguagem C e é transmitido para o compilador "avr-gcc", que realiza a tradução dos comandos para uma linguagem que pode ser compreendida pelo microcontrolador. A IDE apresenta um alto grau de abstração, possibilitando o uso de um microcontrolador sem que o usuário conheça o mesmo. Essa IDE possui uma linguagem própria baseada na linguagem C e C++. O ciclo de programação do Arduino pode ser dividido da seguinte maneira:

- Conexão da placa a uma porta USB do computador;
- Desenvolvimento de um *sketch* com comandos para a placa;
- *Upload* do *sketch* para a placa, utilizando a comunicação USB;
- Aguardar a reinicialização, após isto ocorrerá à execução do *sketch* criado.

A partir do momento que foi realizado o *upload*, o Arduino não precisa mais do computador: o Arduino executará o *sketch* criado, desde que seja ligado a uma fonte de energia. A IDE pode ser obtida gratuitamente no site do Arduino.

### 3 Descrição do funcionamento do módulo de RPS (*Remote Presence System*)

Nesta seção é descrito o desenvolvimento de um módulo que permite aos usuários comandarem o RoboDeck de modo remoto. Por exemplo, uma escola poderia adquirir o robô e criar um laboratório de robótica onde ficaria fisicamente o robô. Os alunos, de dentro de uma sala de aula, poderiam por meio de uma interface, comandar o robô móvel via um sistema remoto com interface amigável. Este sistema envia os comandos escolhidos via rede Ethernet para um servidor de aplicação. Este servidor envia o comando para o IP do robô que deve estar conectado via Wi-Fi com a Internet. O aluno visualizará a execução da sequência de comandos enviada por

meio do vídeo capturado pela própria câmera no robô e exibido na tela do aplicativo. A interface desenvolvida buscou ser intuitiva para permitir que uma pessoa leiga rapidamente possa interagir com o robô. A interface foi desenvolvida em Linguagem C++ para Windows.

Para os usuários que desejam utilizar o robô para fins de pesquisa, é possível utilizar as bibliotecas disponibilizadas baseadas na linguagem C/C++. Além de funções já existentes na biblioteca (".dll") já compilada que realizam a comunicação dos pacotes de dados com o servidor de aplicação. O objetivo da construção das referidas bibliotecas é criar a possibilidade de manipulação, por parte dos usuários, dos módulos anteriormente desenvolvidos. O foco é permitir que os usuários programadores possam utilizar determinados ambientes de programação, bastando acrescentar as bibliotecas desenvolvidas e disponibilizadas pela empresa. As bibliotecas apresentadas foram contempladas com classes que possibilitem o desenvolvimento de ações específicas para cada dispositivo do RoboDeck.

Na Figura 3 é possível observar a interface desenvolvida. Onde no canto esquerdo superior está a imagem transmitida em tempo real pelo RoboDeck. Logo abaixo está um espectro de RGB (*Red, Green, Blue*) que é retirado da imagem captada. No centro da tela, existem nove comandos para controlar a movimentação do robô sendo que o botão central gera o comando de "stop" do robô.



**Figura 3:** Interface do aplicativo desenvolvido em Windows utilizando C++. Fonte: própria.

Para utilizar, basta arrastar com o dedo (a tela é *touchscreen*) o botão do movimento desejado e

colocar no retângulo do lado direito inferior. O sistema vai colocar em uma pilha de execução e ao final da execução desse comando ele apaga dessa pilha. A rota realizada pelo robô pode ser visualizada na parte superior da tela. Na parte inferior da tela, à direita, encontra-se o botão iniciar, após clicá-lo o sistema passa a executar a sequência de comandos programados. A lista de comandos pode ser visualizada na tela branca logo acima do botão verde de iniciar.

Estas ações são realizadas em tempo real. Contudo, a execução de um comando pode variar entre 5–30 segundos. Essa variação de tempo encontra-se diretamente relacionada à qualidade da conexão da Internet com a qual o servidor de aplicação está conectado. Se o comando não conseguir ser executado. Aparece na tela do sistema esta impossibilidade. Outra proteção advém dos sensores do robô caso algum sensor impeça o RoboDeck de progredir com o movimento. Uma informação é transmitida para o sistema e é exibida na tela que um sensor foi acionado e o robô não pode realizar a movimentação exigida. Os testes práticos mostraram que existe muita variação na execução do tempo.

Na Figura 4 é exibida uma sequência de execução de movimento utilizando a interface remota desenvolvida.



**Figura 4:** Comando do RoboDeck sendo executado pela interface remota. Fonte: própria.

#### 4 Descrição da bancada didática

Com o objetivo de permitir que o aluno possa ter uma visão ampla da integração entre sensores e atuadores, conectados a diferentes famílias de microcontroladores, foi desenvolvida uma bancada baseada na arquitetura do RoboDeck. Nesta bancada didática, chamada de WorkDeck, existe um nano-computador com sistema operacional Debian, ligado ao microcontrolador ARM9, onde tanto o professor quanto o aluno, podem desenvolver aplicações usando o sistema Debian diretamente no microcontrolador. Dessa forma, é possível criar e gerenciar diversos tipos de comunicações, como serial, UART (*Universal Asynchronous Receiver/Transmitter*), I2C (*Inter-Integrated Circuit*), Wi-Fi e *bluetooth*.

O sistema básico é semelhante à plataforma RoboDeck. Possui uma placa-mãe com *software* embarcado na qual é acondicionada todas as demais partes: bússola, dispositivo GPS, um acelerômetro, câmera USB com microfone embutido, quatro sensores de ultrassom e luminosidade, quatro sensores de proximidade *infrared*; um *hub* USB; sensor de temperatura e umidade; um dispositivo de conexão Wi-Fi; um dispositivo de conexão *bluetooth*, entre outros.

A bancada é montada sob um gabinete retangular, confeccionado de material plástico ou madeira; um painel frontal, o qual permite visualizar informações de funcionamento e ter acesso à leitura de sinais digitais e analógicos de todos os dispositivos e módulos por meio de *borners* de saídas e *leds*; fonte de alimentação conectada a rede de energia elétrica por meio de cabo de energia; dois moto-redutores helicoidais; *display* de visualização LCD e de sete segmentos para visualização de caracteres; arranjo de *leds*; *borner* de entrada e saída; botão de liga/desliga.

Em linhas gerais, a forma de uso do WorkDeck é semelhante ao RoboDeck contudo neste caso, o foco não está na mobilidade e sim na percepção do entendimento dos sinais digitais e eletrônicos, além dos dados de entrada e saída. Onde é importante medir esses sinais e acompanhar os dados enquanto o sistema está em funcionamento usando osciloscópio, multímetro, geradores, analisadores, etc. Por isto, existe a necessidade que a plataforma robótica esteja sob uma mesa para que os usuários tenham acesso a todos os dispositivos de entradas e saídas de sinais analógicos e digitais. Além disso, a

autonomia de energia elétrica (bancada ligada na tomada elétrica) é importante para não ficar dependendo do carregamento e troca de baterias. Pode-se ligar um computador na bancada e também trabalhar de forma sincronizada, enviando e recebendo dados por meio de conexão.

A Figura 5 mostra a arquitetura do sistema. Um ponto principal é que o aluno pode trabalhar inicialmente com cada módulo separado, e depois iniciar uma integração. O Módulo de Alta Performance (MAP) foi criado para rodar em uma placa microprocessada que possua o sistema operacional Linux. Essa placa tem como objetivos principais: tornar o kit robótico autônomo e fornecer comunicação de banda larga entre o kit robótico e o meio externo. O MAP permite que sejam desenvolvidos aplicativos robóticos capazes de atuar como controladores do kit robótico, enviando comandos ao microprocessador ARM9. Os aplicativos robóticos também podem ser utilizados para expandir os comandos robóticos ao implementar comandos que podem ser acessados pelos controladores do kit.

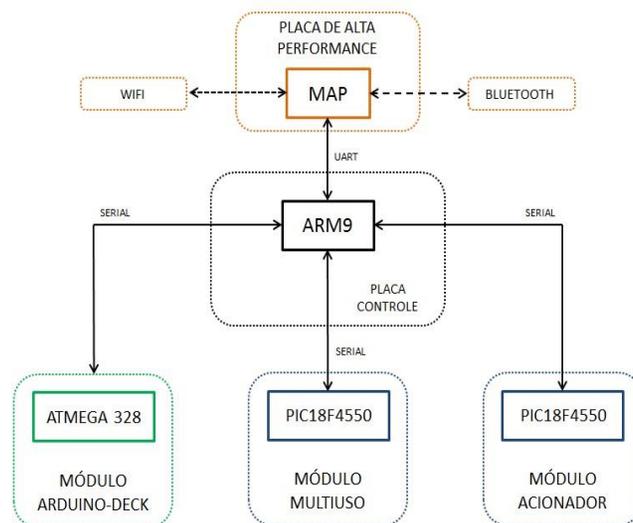


Figura 5: Visão geral da arquitetura do kit. Fonte: própria.

Para que o professor possa verificar o funcionamento correto da bancada, foi desenvolvido o aplicativo GIGA-TESTE (Figura 6). Essa aplicativo pode ser instalado em um *notebook* com Windows. É utilizada à rede Wi-Fi para se comunicar com a bancada. Para este tipo de teste, os microcontroladores presentes na bancada já

possuem um *software* embarcado gravado, pronto para receber o comando do *software* GIGA-TESTE.

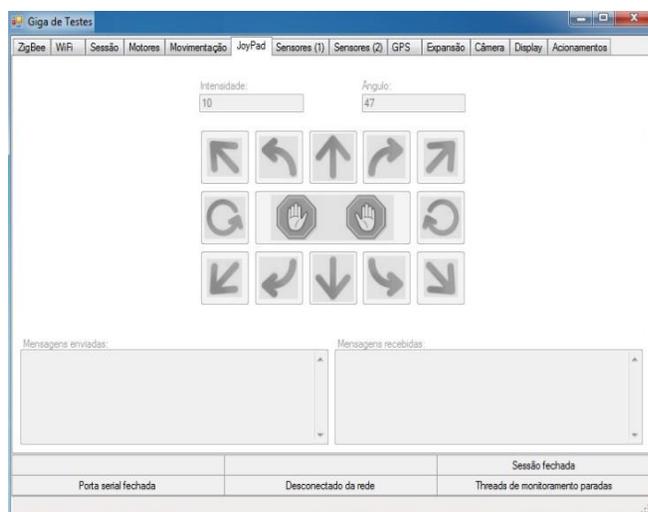


Figura 6: Software GIGA-TESTE. Fonte: própria.

É possível ter acesso a todos os arquivos fontes (".hex") originais dos *softwares* embarcados de cada microcontrolador. Com isto é possível regravar os microcontroladores quantas vezes forem necessárias para incluir e testar as aplicações desenvolvidas no laboratório (aulas práticas), e depois retornar o *software* original. Um exemplo da utilização dos aplicativos robóticos é o interfaceamento com um módulo de visão robótica. Depois de conectada uma câmera USB ao kit, um aplicativo robótico de captura de imagens pode ser criado e carregado, de forma a permitir que um controlador capture as imagens enviadas pela câmera por meio de um aplicativo que pode ser desenvolvido pelos alunos com a supervisão do professor.

Na Figura 7 é possível ver o gabinete do WorkDeck. O mesmo possui dimensões de 652 mm de comprimento, 300 mm de largura e 490 mm de altura. Possui quatro pés de apoio na base com regulagem de altura. A parte traseira contém duas portas para acesso a parte interna do kit. O gabinete possui dois *coolers* para refrigeração da parte interna e duas alças de plástico dispostas nas laterais. O painel frontal possui a nomenclatura dos componentes, indicação dos pontos de alimentação e o nome dos módulos (Figura 8).



Figura 7: Software GIGA-TESTE. Fonte: própria.



Figura 8: Visão geral do painel frontal. Fonte: própria.

Com relação à alimentação da bancada, existe uma fonte primária, com tensão de entrada nominal 85~264 Vac e com tensão de saída 24 Vcc. A fonte tem proteção de curto-circuito, de aquecimento e sobre-tensão. Além disso, possui fusível de entrada, rearme automático e resfriamento por convecção natural. No painel existe uma chave geral para acionar todo o sistema e um *led* verde para indicar que o sistema foi acionado.

A bancada é composta por um computador embarcado com as seguintes características:

<http://dx.doi.org/10.25286/rep.v4i1.930>

Processador INTEL ATOM N270 CPU - 1.60 GHz; dois canais de áudio – Entrada / Saída; Memória SO-DIMM DDR2 2GB/667MHz; Memória gráfica; 06 conectores para USB 2.0/1.1, sendo quatro dispostos na parte externa do kit e duas na parte interna; Alimentação de 12Vcc; Temperatura de trabalho de 0 a 60° C; Conector RJ-45; Conector externo DVI-I; Cartão de memória COMPACT-FLASH 8GB; Botão de pulso *on-off* e Botão de pulso reset. Possui *led* verde para mostrar que o computador embarcado está ligado ou desligado e *led* vermelho para mostrar o processamento do sistema.

## 5 Material didático

Após um estudo de viabilidade educacional, tendo como referência o projeto de curso, os planos de ensino e o perfil profissional de conclusão do alunos do Curso Técnico em Mecatrônica, onde se tem como ponto central o foco industrial. Verificou-se a possibilidade de utilização do WorkDeck e do RoboDeck em momentos distintos das unidades curriculares. Foram elencadas três disciplinas: Microcontrolador, Tecnologia da Comunicação e Robótica. Baseada nas tecnologias presentes no WorkDeck e no RoboDeck, foi visualizada duas situações de aprendizagem. A primeira, com os alunos trabalhando com as tecnologias individualmente utilizando o WorkDeck; e a segunda, fazendo a integração dessas tecnologias utilizando o RoboDeck. Para que haja a perfeita utilização e desenvolvimento dos conhecimentos foi necessário um planejamento integrado para traçar corretamente a linha de progressão dos assuntos que deveriam ser abordados. Houve também a necessidade da inclusão de alguns conteúdos curriculares para a aplicação e utilização das ferramentas desenvolvidas. Porém, não foi excluído nenhum conteúdo descrito no plano de curso original. Foi necessário trabalhar o conceito e também as habilidades dos alunos na unidade curricular “Linguagem de Programação”. Para isto, foi necessário acrescentar os seguintes conteúdos na ementa:

- Introdução à informática;
- Sistemas Operacionais;
- Sistemas Livres;
- Programação Estruturada;
- Linguagem de Programação;
- Compiladores;

- Controle e Aquisição de dados;
- Programação Orientada a Objetos.

Após o aprendizado sobre a manipulação de arquivos e compilação de sistemas, onde o aluno já possui os conceitos básicos e o domínio suficiente em programação, pode-se introduzir o conteúdo da programação embarcada. Esse assunto pode ser desenvolvido na disciplina de “Microcontroladores” com a utilização do WorkDeck como ferramenta didática. Essa disciplina é ministrada no terceiro semestre do curso com carga horária de 100 horas/aulas, onde devem ser trabalhados os seguintes conteúdos: arquitetura interna do microcontrolador, programação do microcontrolador e interfaceamento de periféricos. O microcontrolador utilizado para o desenvolvimento do conteúdo foi o PIC 16F628A. É trabalhado junto ao aluno, a forma de programação e o controle de periféricos como: *leds*, motores de passo, *displays* de sete seguimentos, *displays* de cristal líquido e alguns sensores. Tendo em vista o tempo disponível para se trabalhar os conhecimentos nesta unidade curricular, pode-se verificar que para um melhor desempenho e tempo de assimilação dos alunos, todo o conteúdo deverá ser abordado utilizando o Microcontrolador ARM9. Sendo assim, foram necessárias a adição de quatro itens no desenvolvimento do WorkDeck: um *byte* de *leds*, um *display* de sete seguimentos, um *display* de LCD e ao menos uma chave.

Concomitante a disciplina ou unidade curricular (como é chamado internamente no SENAI) Microcontroladores, ocorre à disciplina Tecnologia da Comunicação, onde são abordados conhecimentos relativos a sistema de comunicação de dados tendo como conteúdo: transmissão digital de dados, padrões de interface serial, padrões de interface paralela, protocolo de comunicação orientado a *bit* e *byte*, introdução à rede de computadores, arquitetura de redes, sistemas operacionais de rede e redes industriais. Essa unidade curricular é dividida em dois semestres com 30 horas/aulas no terceiro semestre, e 20 horas/aula no quarto semestre. Com as tecnologias presentes no WorkDeck é possível trabalhar os assuntos relacionados à comunicação, onde se pode utilizar a “giga” para fazer a parte prática aplicando as tecnologias de Wi-Fi e *bluetooth*. Esse assunto pode ser abordado no segundo semestre da unidade curricular, onde os alunos já aprenderam o funcionamento de protocolos, suas

características bem como a programação de sistemas embarcados, visto na unidade curricular Microcontroladores.

Em seguida foi possível a utilização do RoboDeck junto a unidade curricular Robótica, onde são desenvolvidos os conteúdos: introdução à robótica, estrutura de funcionamento, segurança com robôs industriais, operação e programação, aplicação de sensores, visão robótica, aplicações de robôs em sistemas automatizados, manutenção preventiva e normas de segurança. Além disso, foi acrescentado o conteúdo relacionado à robótica móvel nessa unidade curricular.

No Quadro 1 é apresentado um detalhamento sobre as disciplinas discutidas e suas relações com o RoboDeck e o WorkDeck. Foi desenvolvido um método para a construção dos Planos de Aula (PA) para cada uma das disciplinas.

**Quadro 1:** Detalhamento das disciplinas onde o RoboDeck e o WorkDeck podem atuar.

Semestre	Disciplina	Utilização	Ferramenta
3º	Microcontroladores/ Eletrônica Digital	Programação do Microcontrolador ARM9 elaborando o controle das tecnologias presentes no WorkDeck.	WorkDeck
4º	Tecnologia da Comunicação/ Linguagem de programação	Utilização do WorkDeck para trabalhar a prática dos conteúdos de comunicação de dados utilizando as tecnologias: Wi-Fi e Bluetooth.	WorkDeck
4º	Robótica	Fazer a integração das tecnologias trabalhadas com a utilização do RoboDeck.	Robodeck

Foram desenvolvidos três apostilas contendo uma quantidade variável de PAs conforme a disciplina. A apostila de linguagem de programação possui 10 plano de aulas (Quadro 2), com diversas atividades que foram baseadas no plano de ensino da disciplina. Para a disciplina de Robótica existem seis planos de aula (Quadro 3) e para Eletrônica Digital I, existem sete planos de aulas (Quadro 4).

**Quadro 2:** Planos de estudo para disciplina de Linguagem de Programação I. Fonte: própria.

1. Introdução e conceitos básicos – organização do computador, programas, interface com os recursos didáticos utilizando e seus aplicativos.
2. Lógica de programação – conceitos, lógica, algoritmo natural, introdução aos recursos didáticos.
3. Lógica de programação – primeiros programas em blocos (instruções condicionais if, switch/case e

Leitura/Conversão.

4. Lógica de programação – instrução de laço.
5. Lógica de programação – situações problemas.
6. Programação estruturada C/C++.
7. Programas em C/C++ - primeiros programas, com instrução if, switch/case e leitura/conversão.
8. Programas em C/C++ - instruções de laço.
9. Programas em C/C++ - situações problemas.
10. Desafio de programação.

**Quadro 3:** Planos de estudo em Robótica. Fonte: própria.

1. Introdução e conceitos básicos – Apresentação do RoboDeck, componentes e *software*.
2. Compilando e gerando códigos – utilização da giga de testes e visual Studio 2008.
3. Robôs e classificações – principais propriedades e de suas funcionalidades.
4. Sensores – definição, classificação, propriedades e principais fontes erros.
5. Atuadores e movimento – classificação, propriedades e principais fontes de erros.
6. Práticas com o robô – direcionadas à navegação de robôs móveis.

**Quadro 4:** Plano de estudo em Eletrônica Digital I. Fonte: própria.

1. Introdução e conceitos básicos.
2. Identidade/negação/ E/E Negado.
3. Ou/Ou negação/ Ou exclusivo/Ou exclusive neg.
4. Famílias lógicas.
5. Programação C e circuitos combinatórios.
6. Codificadores.
7. Decodificadores.

## 6 Discussões

Foram realizadas atividades referentes à difusão e transferência do conhecimento gerados neste projeto por meio de cursos para os professores da rede SENAI do estado de São Paulo (Figura 9). Além disso, aguarda-se a melhoria da saúde financeira das

<http://dx.doi.org/10.25286/rep.v4i1.930>

escolas técnicas estaduais e federais para iniciar um trabalho de popularização dessas soluções de tecnologia educacional. Entende-se que os alunos que podem ter acesso ao conteúdo criado pelo projeto baseado em robótica estarão melhores preparados para serem absorvidos pela indústria de forma positiva, pois serão capazes de trabalhar o conceito da indústria 4.0 e realizar o aprimoramento dos produtos de IoT (Internet das Coisas) para tornarem essas soluções comercialmente mais competitivas.



**Figura 9:** Cursos para os professores da rede SENAI do estado de São Paulo. Fonte: própria.

As atividades realizadas tiveram como foco o desenvolvimento de módulos para complementar uma ferramenta para educação e pesquisa de uso multidisciplinar. O projeto teve o seu êxito na construção de uma plataforma educacional que não se limitou somente a criação de dispositivos de *hardware* e *software*, mas sim de um planejamento pedagógico atrelada a aplicação da plataforma dentro de sala de aula. Esse alicerce pode favorecer a implantação junto às escolas técnicas. Em muitos casos essas escolas recebiam um equipamento didático onde os professores de cada disciplina tinham que formular os próprios planos de aula (PAs). Neste caso, esses PAs eram dispersos e sem uma integração multidisciplinar. Com este trabalho foi possível elaborar diversos planos de aula (PA) para diversas disciplinas diferentes. Isto, além de ser positivo do ponto de vista financeiro permitiu constituir uma interconexão natural com as disciplinas abordadas pelo projeto.

O projeto foi construído com o objetivo de apoiar as atividades de aprimoramento e nivelamento do ensino para área de sistemas embarcados junto às grades curriculares dos cursos técnicos.

Com relação aos resultados de desempenho de cada um dos módulos descritos neste artigo. Foram validados por testes práticos em laboratório (validação interna e externa seguindo o processo de desenvolvimento de produto baseado na metodologia *Agile*) e também pelo uso dos equipamentos durante a execução dos cursos realizados junto aos professores da rede SENAI-SP. Foram observados aspectos relacionados à comunicação, exatidão de movimentos, quantidade de falhas de comandos enviados, etc.

Entre os testes realizados, existiram os que verificaram as condições de comunicação no caso do RPS. Por exemplo, seguindo uma sequência específica de comandos e repetindo-a por várias vezes. A média de tempo para a execução de um comando foi de 1,12 segundos. Lembrando que o sistema envia o comando pela Internet por meio do computador desktop. Este comando chega ao robô por meio da conexão *Wi-Fi*. Nos primeiros testes, o percentual de comandos não executados ficou em torno de 2%, isto é, a cada conjunto de 100 comandos era detectado que se perdia 2 comandos. Foi revisado o código do protocolo e colocado um *buffer* maior. Isto aumentou o tempo de execução de cada comando para uma média de 1,78 segundos, contudo cada conjunto de 100 comandos não foi mais detectada perda de dados.

## 7 Considerações finais e trabalhos futuros

O projeto promove a educação tecnológica e tem em seu “cerne”, a formação de recursos humanos. Tanto no caso da capacitação continuada dos professores quanto na formação dos próprios alunos dos cursos técnicos. Entende-se que este tipo de ferramenta educacional é prioritário na questão da melhoria dos profissionais que vão trabalhar nas indústrias principalmente relacionadas à criação de novos produtos e serviços. De forma geral, as instituições de ensino ao adotarem a tecnologia desenvolvida, serão capazes de otimizar o tempo e melhorar a curva de aprendizado do aluno. Também

é possível construir um contínuo desenvolvimento de planos de aula (PA) que impacte no longo prazo no aprimoramento do sistema educacional técnico por competência e não por conteúdo. É fundamental realizar as atividades práticas para poder entender como realmente funciona.

A criação do material didático teve por finalidade a proposição da utilização de dispositivos baseados em robótica como ferramentas didáticas no auxílio às atividades práticas nos laboratórios das escolas técnicas. Permitindo com isto, que o professor consiga fazer com que os alunos melhorem sua curva de aprendizagem e se nível de engajamento por meio de exercícios e resoluções de problemas práticos. Os dispositivos são utilizados para promover o desenvolvimento educacional e aprendizado das disciplinas: linguagem de programação, eletrônica digital, microcontroladores, tecnologia de comunicação e robótica, além de conceitos técnicos nas áreas de lógica digital, controle, para alunos do ensino técnico. Utilizando ainda como plataforma de pesquisa e desenvolvimento nas áreas de computação, eletrônica e mecatrônica para alunos de graduação e pós-graduação.

Como sugestão de trabalho futuro, poderia se realizar a integração da plataforma RoboDeck com um *framework* que facilitasse a programação de todo o sistema, como por exemplo, o ROS (*Robot Operating System*). A maior contribuição é permitir que uma API possa realizar a interação com a plataforma. Isso faz com que não seja preciso usar apenas o ambiente fornecido pela empresa fabricante (Giga de Testes e SDK em C++) para se comunicar com o robô. Além disso, é possível construir programas clientes em qualquer linguagem suportada pelo ROS, tornando assim os aplicativos independentes de linguagem e de plataforma. Outra contribuição é realizar uma implementação do SDK (*Software Development Kit*) para linguagem Java. No entanto, perdem-se as vantagens da utilização dos pacotes e bibliotecas do ROS. Uma possível melhoria para o *middleware* seria projetar uma forma de utilizar as leituras de sensores e GPS como tópicos do ROS. Ou seja, receber em intervalos regulares a leitura dos valores dos sensores por meio de um tópico. Outra possível melhoria seria modificar a forma com que o MCS (Módulo de Controle de Sessão) realiza a autenticação do controlador. Atualmente, ele permite que seja aberto um *stream*

apenas na porta 2000. Isso limita os envios de comandos ao robô.

## Agradecimentos

Ao apoio do CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico) por meio do seu programa de Desenvolvimento Tecnológico e Extensão Inovadora (DT) e do programa de Formação de Recursos Humanos em Áreas Estratégicas (RHAE). Também, agradecer ao SENAI São Carlos (SP), em especial ao diretor da unidade, Prof. Marcio Marinho.

## Referências

- [1] BEZERRA, J. **Arquitetura para integração de módulos de reconhecimento de fala em plataforma robótica móvel**. Dissertação (Mestrado em Ciência da Computação) - Universidade Metodista de Piracicaba. 2012.
- [2] VALERIO NETTO, A. ; Miranda, F. A. . **A block programming interface for educational mobile robots**. In: Robocontrol 2012, 2012, Bauru - SP. 5º Workshop in applied robotics and automation, 2012.
- [3] XBOT, **RoboDeck**. Disponível em: <http://www.xbot.com.br/educacional/robodeck/> [Visitado em Janeiro de 2018].
- [4] ORLANDINI, G. **Desenvolvimento de aplicativos baseados em técnicas de visão computacional para robô móvel autônomo**, 2012. Dissertação (Mestrado em Ciência da Computação), Universidade Metodista de Piracicaba.
- [5] MENEZES, M. C. et al., **Mapeamento e Localização para o kit Robótico RoboDeck**, 2016. Disponível em: <http://sistemas.deinf.ufma.br/anaisjim/artigos/2016/201604.pdf> [Visitado em maio de 2017]
- [6] MOTTA, B. C. **Aprendizagem por demonstração baseada em redes neurais artificiais aplicada à robótica móvel**, 2016. Disponível em: <http://www.bdm.unb.br/handle/10483/13282> [Visitado em fevereiro de 2017]
- [7] PISSARDINI, R. S. **Veículos autônomos de transporte terrestre: proposta de arquitetura de tomada de decisão para navegação autônoma**. Dissertação de mestrado. Universidade de São Paulo, 2014.

Disponível em: <http://www.teses.usp.br/teses/disponiveis/3/3138/tde-26082015-161805/en.php> [Visitado em janeiro de 2016]

**[8]** WEI, D. C. M. **Método de desvio de obstáculos aplicado em veículo autônomo**, Dissertação de mestrado. Universidade de São Paulo, 2015 <http://www.teses.usp.br/teses/disponiveis/3/3138/tde-17062016-142254/en.php> [Visitado em janeiro de 2016]

**[9]** ZANOLA, L. et al. **Implementação com Validação Real de um Controle Proporcional, Integral e Derivativo na Plataforma Robótica RoboDeck**, 2017. [http://www.xbot.com.br/wp-content/uploads/2012/10/artigo\\_Implementacao\\_ValidacaoReal\\_leandro.pdf](http://www.xbot.com.br/wp-content/uploads/2012/10/artigo_Implementacao_ValidacaoReal_leandro.pdf) [Visitado em janeiro de 2017]

**[10]** SENAI-SP. **Plano de Curso Técnico em Mecatrônica**, São Paulo: SENAI, 2008.