



UTILIZAÇÃO DA PLATAFORMA ARDUINO COMO FERRAMENTA DE ENSINO PARA ENGENHARIA

Leonardo S. Mai - leonardomai@yahoo.com.br
Anderson A. Schwertner - anderson0854@hotmail.com
Mauro F. Rodrigues - mauro.rodrigues@unijui.edu.br;
Maurício de Campos - campos@unijui.edu.br
Paulo S. Sausen - sausen@unijui.edu.br
Mateus F. Schonardie - mateus.schonardie@unijui.edu.br

Universidade Regional do Noroeste do estado do Rio Grande do Sul (UNIJUÍ) Departamento de Ciências Exatas e Engenharias (DCEENG).
Rua Lulu Ingelfritz, Nº 480, Bairro São Geraldo
98700-000 – Ijuí – Rio Grande do Sul

Resumo: *A partir do surgimento do conceito de “Internet das Coisas”, existe uma forte tendência no aumento na demanda por profissionais com habilidade e formação em programação. Em vista disso, fica clara a necessidade de instruir os estudantes de engenharia para que estejam qualificados a atuar neste novo segmento do mercado. Para isso, é fundamental que primeiramente ocorra uma profunda análise das metodologias empregadas no ensino de programação, buscando torná-las mais atrativas para os jovens estudantes. Neste artigo serão discutidos os conceitos de aprendizado construtivista e Physical Programming, além de uma análise a respeito da maneira com que a plataforma Arduino, como ferramenta de ensino de programação, auxilia os estudantes de engenharia a compreender melhor o funcionamento de seus códigos.*

Palavras-chave: *Arduino, Construtivismo, Programação.*

1 INTRODUÇÃO

A próxima revolução apontada pelas comunidades científicas mundiais será a Internet das Coisas do inglês *Internet of Things* (IoT). Segundo Xia et al. (2012), a humanidade já está presenciando o início desta era, a qual será responsável por grandes transformações na forma como interagimos com o mundo à nossa volta.

O termo “*Internet das coisas*” foi utilizado pela primeira vez por Kevin Ashton em 1999, contudo, sua definição sofreu alterações conforme a tecnologia evoluía. Hoje, IoT representa a ideia da interconexão dos objetos que utilizamos no dia a dia, através do emprego de sistemas embarcados¹ na sua construção. Sistemas embarcados estão presentes em inúmeros

¹ Sistema composto por software e hardware no qual o microprocessador é totalmente dedicado ao dispositivo que controla.



equipamentos e dispositivos do nosso cotidiano, como por exemplo máquinas de lavar, fornos de micro-ondas, automóveis, equipamento hospitalares, dispositivos móveis entre outros.

O Conceito de IoT deve alterar o modo como vivemos, pois, estaremos totalmente conectados em todos os sentidos. Neste cenário, a IoT possibilitará a interação entre os dispositivos conectados a ela e seus usuários. Pode-se imaginar, por exemplo, uma geladeira que através de um conjunto de sensores monitore a quantidade de alimentos em seu interior. A partir desta informação em combinação com o perfil dos usuários de uma residência, pode-se gerar uma lista de compras e encaminhar ao sistema de um supermercado, o qual solicita a confirmação e o pagamento diretamente no smartphone do usuário.

De acordo com Gubbi et al. (2013), a IoT promoverá uma evolução da atual Internet em uma rede de objetos interconectados, os quais além de coletarem informações do ambiente por intermédio de sensores, utilizarão dos padrões já existentes de comunicação para fornecerem dados em tempo real que poderão ser utilizados por outros objetos e sistemas também ligados a essa rede.

Uma pesquisa realizada pela ABI Research (2014) estima que até o ano de 2020 haverá cerca de 40,9 bilhões de dispositivos conectados à Internet, criando um novo mercado que movimentará bilhões de dólares nos próximos anos. Em vista disso, surgem questionamentos a respeito de quem serão os agentes articuladores e qual será o papel do Brasil neste aspecto.

Nas últimas décadas, apesar de muitos avanços, o Brasil tem sido considerado apenas um mercado consumidor de tecnologia. O conhecimento nessa área, com bases sólidas em Matemática e Ciências, deve ser o impulso, uma camada inicial, para subsidiar esse novo paradigma de vida e consumo. A capacidade criativa é essencial para a inovação, mas sem o conhecimento capaz de transformar essa criatividade em produtos e realizações, dificilmente o Brasil entrará nesse mercado como desenvolvedor de sistemas e equipamentos inteligentes, sendo, novamente, apenas um mercado consumidor a ser conquistado.

Dentro do meio acadêmico, as carreiras de engenharia elétrica e ciências da computação estão diretamente relacionadas ao conceito de IoT, uma vez que o desenvolvimento de sistemas embarcados envolve conceitos de software, hardware e comunicação. Tendo em vista essa perspectiva, segundo Rubio et al. (2013), nos últimos anos os cursos de engenharia elétrica passaram a introduzir cada vez mais em seu currículo disciplinas que abordam programação, seja para a utilização de ferramentas matemáticas ou no emprego de microcontroladores no desenvolvimento de sistemas embarcados.

Com o intuito de fornecer aos alunos conhecimentos básicos para o desenvolvimento do raciocínio lógico, estas disciplinas normalmente fazem parte da grade curricular dos semestres iniciais, nos quais o aluno geralmente não é capaz de relacionar a programação com as demais atividades de sua futura profissão (MELO et al., 2014).

Além disso, segundo Ariza (2016), “disciplinas voltadas para o ensino de algoritmos² são tradicionalmente problemáticas, uma vez que exigem do aluno habilidades cognitivas como criatividade e lógica”. Somado a esses fatores, Koliver et al. (2004) afirmam que “a falta de preparo da maioria dos estudantes acarreta em altos índices de desistência e reprovação, em matérias cujo o principal objetivo é auxiliar no desenvolvimento da habilidade de projetar soluções”.

Segundo Pereira Júnior e Rapkiewicz (2004), o modo tradicional de apresentar o conteúdo das disciplinas que envolvem programação se faz de forma expositiva, concentrando-se na resolução de exercícios. Entretanto, esse método não consegue motivar os alunos a se interessar pela disciplina, pois não é claro para os mesmos a importância de certos conteúdos para sua formação. Desta forma, o ponto fundamental do ensino de lógica de

² Sequência de regras, raciocínios ou operações que soluciona um determinado problema.



programação é encontrar uma maneira de motivar o aluno, de forma a despertar nele o interesse, fazendo com que tome gosto pelo aprendizado.

É necessário que se crie um vínculo entre os estudantes e esse novo paradigma, utilizando plataformas e metodologias que incentivem a nova geração, já acostumada com um mundo interconectado, a deixar de atuar como simples usuários, instigando o conhecimento científico para que se tornem os desenvolvedores dos sistemas que irão compor a IoT.

Através dos anos, diversas metodologias que visam melhorar ensino de programação vêm sendo discutidas e experimentadas dentro do meio acadêmico. O trabalho desenvolvido por Rubio et al. (2014) sugere o desenvolvimento de módulos de ensino com a plataforma Arduino, capazes de, a partir de circuitos eletrônicos, demonstrar comportamentos interessantes, através de sinais luminosos, sonoros e pequenos motores, que refletem a lógica de programação de forma visível. Desta forma, o aluno pode aprender criando efeitos luminosos a partir do acionamento de LEDs, ou até mesmo dando instruções a um robô, superando assim o alto nível de abstração da forma tradicional de ensino.

É neste contexto, que no presente artigo é apresentada uma metodologia alternativa de ensino com base no uso da plataforma Arduino. Tal estratégia tem sido aplicada pela Universidade Regional do Noroeste do Estado do Rio Grande do Sul (UNIJUI), em caráter experimental, na forma de minicurso para os alunos dos semestres iniciais do curso de engenharia elétrica. Por conta da alta procura e aproveitamento dos alunos, esta metodologia deverá, futuramente, ser incluída nos componentes curriculares pertinentes no curso de graduação.

2 MATERIAIS E MÉTODOS

2.1 Metodologias de ensino

Durante o processo de ensino-aprendizagem dos fundamentos de programação se percebe que grande parte dos alunos apresentam dificuldades em assimilar as abstrações envolvidas e que são requeridas pela disciplina. Situação essa, agravada pelas técnicas monótonas de resolução de exercícios na forma textual, visando o aprendizado por meio da repetição. E, por esta razão, é preciso despertar no aluno o interesse pelo aprendizado procurando superar suas dificuldades. Neste sentido, um ponto crucial no processo de aprendizagem da lógica de programação é como motivar o aluno, a despertar o interesse pelo aprendizado, procurando superar suas dificuldades como a falta de habilidades matemáticas (MELO et. al., 2014).

O método tradicional de ensino baseia-se na crença de que o conhecimento deve ser transferido de uma fonte conhecedora (professor) para um receptor (aluno). Contudo, desta forma, o conhecimento transmitido ao aluno está limitado ao conhecimento previamente acumulado pelo professor e sua capacidade de transmiti-lo aos estudantes, além disso, tal conhecimento está sujeito às interpretações do professor baseadas em seu ponto-de-vista.

Em contrapartida, em um modelo de ensino construtivista o próprio aluno constrói o conhecimento baseado em seu entendimento e experiências. Modelo no qual o papel do professor será de guiar o aluno adentro da jornada de construção do conhecimento, através de atividades pedagógicas práticas. Deste modo, o processo de aprendizagem será focado nos métodos de pensamento e entendimento, ao invés de apenas memorização do conteúdo (GRAVEN & BJORK, 2016). Este modelo de ensino proporciona, também, ao aluno descobrir a metodologia para a construção de novos conhecimentos independentes da presença do professor, conhecimento este, fundamental na formação de cientistas que buscam avançar as barreiras do conhecimento científico.



Para o caso do ensino de programação, há uma dificuldade particular por conta da abstração da atividade. Linhas de um programa são compiladas e transformadas em operações matemáticas, resultando em um processo digital que nem sempre se traduz em uma representação física. Caso o aluno não consiga abstrair o funcionamento de seu programa e entender o processo que está ocorrendo, o mesmo ficará frustrado pois pensará que suas ações não apresentam resultados. Esta frustração se traduzirá em desinteresse e conseqüentemente dificuldades de aprendizado. Nesta situação nem mesmo o modelo construtivista despertará o interesse do aluno, uma vez que para a formação do conhecimento é necessário, inicialmente, entendimento do processo.

Para contornar esta situação, surgiu o conceito de *Physical Programming*, sendo esta, uma metodologia de ensino de algoritmos/programação que visa a construção de sistemas físicos interativos utilizando hardware e software para responder a estímulos físicos, ou, promovê-los. Desta forma, o aluno não está mais preso à realização de operações matemáticas e tratamento de dados, podendo agora, utilizar componentes físicos como entrada e saída de seu programa. Isto proporciona ao aluno visualizar as ações de seu programa sendo executadas e o resultado destas, não sendo mais necessário um alto grau de abstração para o completo entendimento. Contudo, para a aplicação deste conceito é necessária uma plataforma que fará a interligação entre o meio digital (software) com o meio físico (hardware). Atualmente com a popularização dos microcontroladores, e conseqüentemente sua redução de preço, estes passaram a ser peças chaves na aplicação do *Physical Programming*. Este microcontrolador tem como função executar o programa escrito, para então, realizar uma ação e gerar uma resposta, agora podendo ser, também, física.

2.2 Plataforma Arduino

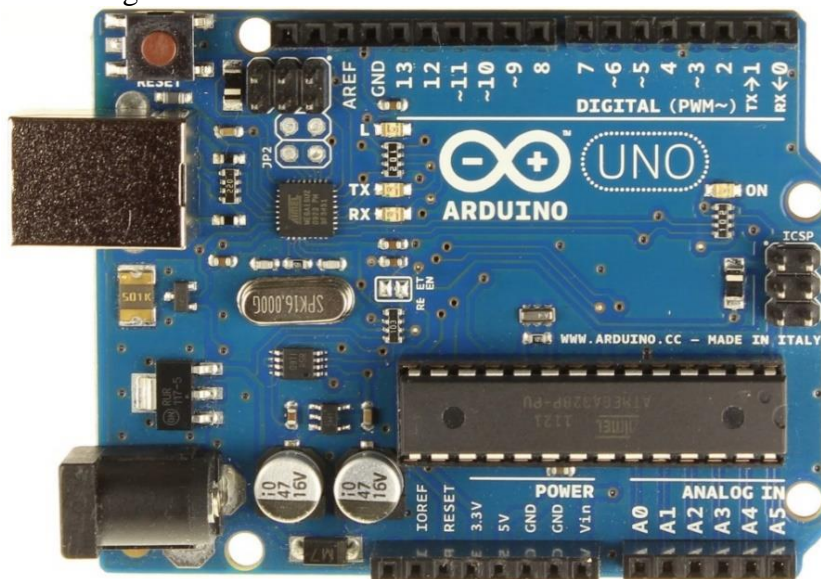
O Arduino é uma plataforma *opensource*³ de desenvolvimento que engloba Hardware e Software com o objetivo de fornecer uma base para criação de projetos de forma facilitada e econômica. Foi criado inicialmente em 2005 para auxiliar estudantes sem conhecimentos prévios de eletrônica e programação como uma plataforma de prototipagem, permitindo a conexão e interação ente o mundo físico e o mundo digital. A plataforma do Arduino é formada por:

- Um microcontrolador AVR Atmega fabricado pela Atmel.
- Software bootloader integrado ao microcontrolador que permite gravar o programa de forma facilitada além de contar com configurações pré-programadas.
- Placa de desenvolvimento formada pela placa de circuito impresso, microcontrolador, conectores e componentes secundários necessários para o funcionamento, apresentada na Figura 1.
- Software de programação IDE, software que será utilizada para programar, compilar e gravar o programa criado, mostrada na Figura 2.

³ É um modelo de desenvolvimento que promove um licenciamento livre para o design ou esquematização de um produto, e a redistribuição universal deste mesmo.



Figura 1- Placa de desenvolvimento Arduino Uno.



Fonte: Arduino.org⁴.

Figura 2 – IDE de programação Arduino.

```
Fade | Arduino 1.8.2
Arquivo Editar Sketch Ferramentas Ajuda

Fade

int led = 9;           // the PWM pin the LED is attached to
int brightness = 0;   // how bright the LED is
int fadeAmount = 5;   // how many points to fade the LED by

// the setup routine runs once when you press reset:
void setup() {
  // declare pin 9 to be an output:
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  // set the brightness of pin 9:
  analogWrite(led, brightness);

  // change the brightness for next time through the loop:
  brightness = brightness + fadeAmount;

  // reverse the direction of the fading at the ends of the fade:
  if (brightness <= 0 || brightness >= 255) {
    fadeAmount = -fadeAmount;
  }
  // wait for 30 milliseconds to see the dimming effect
  delay(30);
}
```

Arduino/Genuino Uno em COM18

Fonte: *print screen* da aplicação no sistema operacional Windows 10.

⁴ Disponível em: <<http://www.arduino.org/products/boards/arduino-uno>> Acesso em: 17 de mai. de 2017.



A principal vantagem do Arduino, frente as demais plataformas concorrentes, refere-se ao uso de bibliotecas, que consistem em partes de códigos pré-programados que permitem a realização de funções avançadas de forma simplificada. Estudos sobre a reusabilidade de código fonte demonstram que entre 40% e 60% de um código pode ser adaptado em mais de uma aplicação, 75% das funções são comuns entre diferentes códigos fontes e apenas 15% do código escrito é único para cada determinada aplicação (EZLAN et al., 2002). Isso se dá pelo fato de grande parte das aplicações utilizarem funções genéricas de entrada, saída e tratamento de dados.

Para o caso do Arduino, a plataforma conta com uma comunidade de mais de 300 mil membros, participando do fórum de ajuda e desenvolvimento do *Arduino Playground*, uma biblioteca virtual de exemplos e tutoriais. A utilização destas bibliotecas proporciona ao aluno a capacidade de realização de trabalhos mais complexos, uma vez que não será necessário um conhecimento aprofundado sobre o funcionamento dos sensores, dispositivos e seus protocolos de comunicação. Visto que essa consolidação de conhecimentos demanda tempo, no caso de utilização de bibliotecas o aluno apenas necessita aprender o funcionamento das mesmas e suas funções, restando tempo hábil para focar na implementação de novas funções ao seu projeto (HERTZOG & SWART, 2016).

3 RESULTADOS E ANÁLISE

Diversas pesquisas têm mostrado a eficiência da utilização do Arduino como uma plataforma de ensino, tanto dentro das Universidades quanto em oficinas que visam introduzir alunos do Ensino Médio aos cursos tecnológicos.

No estudo realizado por Rubio et al. (2013, 2014) a plataforma Arduino foi utilizada no desenvolvimento de diversos módulos que buscam servir de base ao ensino de programação em áreas como engenharia, matemática, tecnologia e ciências. Os dados estatísticos foram obtidos analisando duas turmas de uma disciplina de introdução a programação do curso de Biologia.

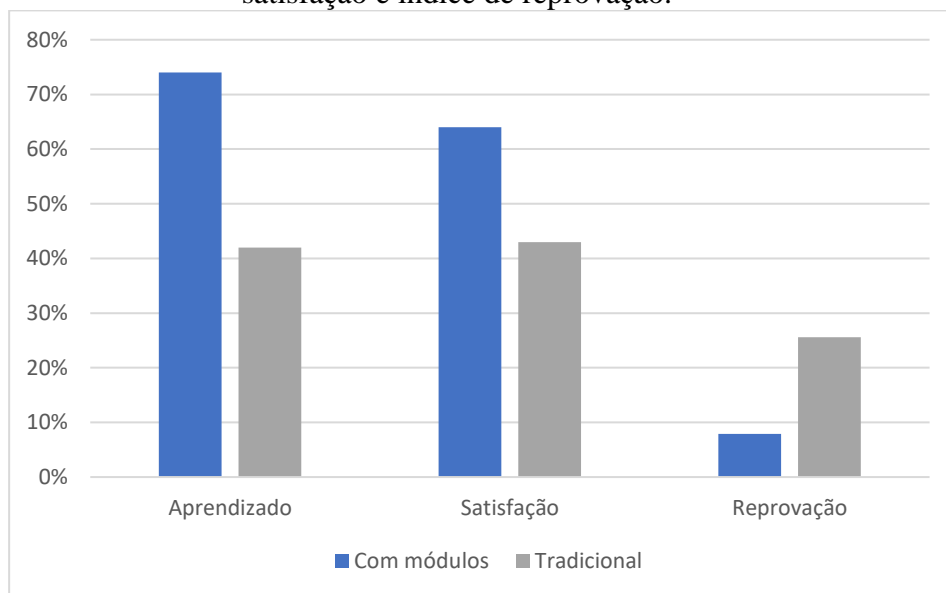
Na turma tradicional, foram utilizados slides e multimídia para apresentar conceitos teóricos introdutórios. Na segunda turma, as aulas foram aperfeiçoadas empregando experimentos com os módulos desenvolvidos. Ao comparar o desempenho dos estudantes monitorados no estudo, a turma que utilizou os módulos apresentou um aumento de 32% na quantidade de alunos que alcançaram um nível de programação considerado como bom pelos autores.

Além disso, a plataforma Arduino foi muito bem recebida pelos estudantes, uma vez que 95% dos mesmos consideraram as aulas interessantes e 85% gostaram das demonstrações realizadas pelo instrutor. Ao comparar com a metodologia tradicional utilizada na universidade, a Figura 3 destaca, também, o aumento de 21% no percentual de alunos que se sentiram confiantes e satisfeitos com seu nível de programação após as aulas.

A principal diferença percebida nessa análise está no percentual de alunos que não obtiveram desempenho satisfatório no exame final. Enquanto 25,6% dos integrantes que participaram das aulas tradicionais não alcançaram a nota adequada, somente 7,9% dos participantes das aulas experimentais ficaram com desempenho insuficiente.



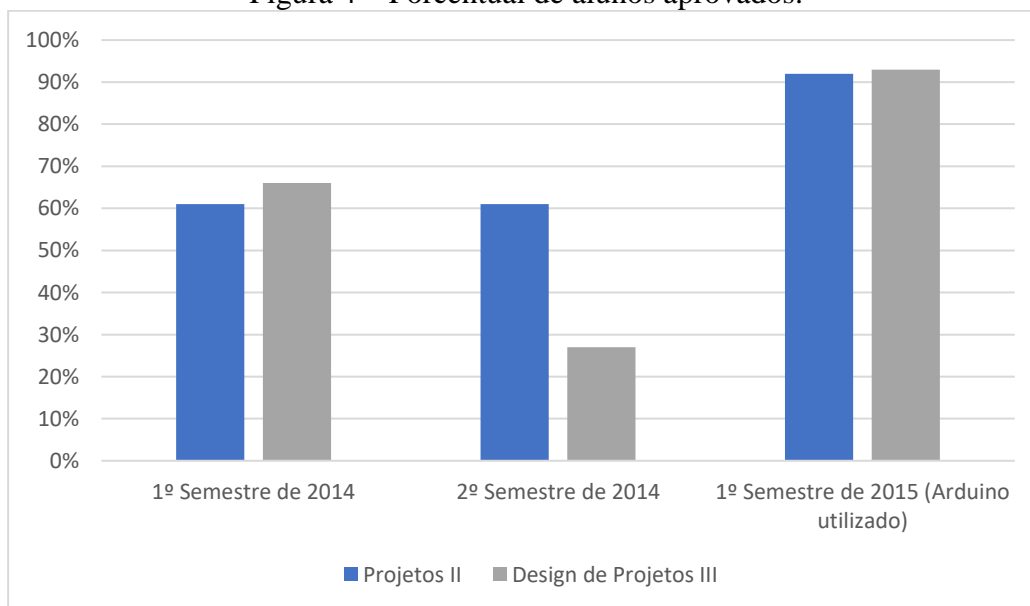
Figura 3 – Porcentagem de alunos que atingiram bom nível de programação, nível de satisfação e índice de reprovação.



Adaptado de RUBIO et al. (2013, 2014).

Na análise realizada por Hertzog e Swart (2016) a plataforma Arduino foi utilizada como base para desenvolvimento de projetos em duas disciplinas diferentes em semestres intermediários do curso de engenharia elétrica. Dos alunos monitorados, 70% afirmaram que os módulos ajudaram para o melhor entendimento da teoria. Melhorias na lógica de programação foram apresentadas por 36% dos estudantes e 26% afirmaram ter melhorado suas habilidades na resolução de problemas.

Figura 4 – Porcentual de alunos aprovados.



Adaptado de HERTZOG & SWART (2016).



A Figura 4 exibe a taxa de aprovação das duas matérias em três semestres diferentes, sendo que em 2014 o uso da plataforma Arduino era opcional. A taxa de aprovação representa a quantidade de alunos que completaram a disciplinas com aproveitamento final maior ou igual a 50%. Nota-se o surpreendente aumento desse índice no módulo de Projetos II, que subiu de 61% em 2014 (nem todos os alunos utilizavam a plataforma Arduino) para 92% em 2015 (a plataforma Arduino foi utilizada por todos os estudantes). Esse mesmo comportamento pode ser observado na disciplina de Design de Projetos III, que teve aumento de 66% no primeiro semestre de 2014 para 93% em 2015.

4 CONCLUSÕES

Novas formas de cativar a atenção dos alunos em sala de aula tem sido um dos principais temas em discussão pela comunidade acadêmica. Em um mundo conectado como o que vivemos, métodos tradicionais de ensino se tornaram monótonos e desinteressantes ao olhar dos alunos, resultando em baixos índices de aproveitamento e interesse por parte dos mesmos. Matérias que envolvem programação apresentam um desafio ainda maior, uma vez que requerem um alto grau de abstração e lógica, duas áreas que, usualmente, os estudantes apresentam dificuldades. A implementação de um método de ensino construtivista, aliado ao conceito de *physical programming*, faz uso de microcontroladores e componentes eletrônicos para traduzir de forma física os resultados de um código que até então ficava preso a tela do computador.

Com base nos dados apresentados neste artigo, é evidenciado o aumento do desempenho e da satisfação dos alunos quando utilizada a plataforma Arduino. Desta forma, o uso do Arduino pode ser uma maneira de despertar o interesse e atrair jovens estudantes para a área de sistemas embarcados, visando formar profissionais qualificados a atuar nesse emergente mercado.

Agradecimentos

Os autores agradecem a FAPERGS pelas bolsas de iniciação tecnológica e iniciação científica e a UNIJUÍ.

REFERÊNCIAS BIBLIOGRÁFICAS

ABI RESEARCH. **The Internet of Things will drive wireless connected devices to 40,9 Billion in 2020**. Disponível em: <<https://www.abiresearch.com/press/the-internet-of-things-will-drive-wireless-connect/>> Acesso em: 17 maio 2017.

ARIZA, Jonathan Álvarez. A proposal for teaching programming languages through open hardware tools. IEEE 8th International Conference on Engineering Education, p. 202-207, 2016.

EZRAN, Michel; MORISIO, Maurizio; TULLY, Colin. Practical Software Reuse, Londres: Springer, 2002. 216 p.

GRAVEN, Olaf Hallan; BJØRK, Joakim. The use of an Arduino pocket lab to increase motivation in Electrical Engineering students for Programming. IEEE International Conference on Teaching, Assessment and Learning for Engineering, p. 239-243, 2016.

Organização



Promoção





GUBBI, Jayavardhana et al. Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems*, v. 29, p. 1645-1660, 2013.

HERTZOG, Pierre Eduard; SWART, Arthur James. Arduino – Enabling engineering students to obtain academic success in a design-based module. *IEEE Global Engineering Education Conference*, p. 66-73, 2017.

KOLIVER, Cristian; DORNELES, Ricardo Vargas; CASA, Marcos Eduardo. Das (muitas) dúvidas e (poucas) certezas do ensino de Algoritmos. *XII Workshop sobre Educação em Informática*, p. 949-960, 2004.

DE MELO, Rosangela Maria et al. Using Arduino as pedagogical strategy in learning logic programming. *9th Iberian conference on Information Systems and Technologies*, p. 1-5, 2014.

PEREIRA JÚNIOR, José Carlos; RAPKIEWICZ, Clevi Elena. O Processo de Ensino-Aprendizagem de Fundamentos de Programação: Uma Visão Crítica da Pesquisa no Brasil. *Workshop de Educação em Computação – Congresso da Sociedade Brasileira de Computação*, 2004.

RUBIO, Miguel Angel et al. Enhancing an introductory programming course with physical computing modules. *IEEE Frontiers in Education Conference*, 2014.

RUBIO, Miguel Angel et al. Using Arduino to enhance computer programming courses in science and engineering. *EDULEARN13 Proceedings Conference*, p. 5127-5133, 2013.

XIA, Feng et al. Internet of Things. *International Journal of Communications Systems*, v. 25, p. 1101-1102, 2012.

USING THE ARDUINO PLATFORM AS A TEACHING TOOL IN ENGINEERING

Abstract: *Due to the emergence of the “Internet of Things”, there has been a strong increase in the demand for professionals able to program. It is clear that engineers have to be instructed so that they are qualified to work on this new market. For such, it is fundamental that a deep analysis of the methodologies used in programming teaching takes place, seeking to make them more attractive to young students. In this paper, the concepts of constructivism learning and physical programming will be discussed, as well as an analysis of how the Arduino platform, as a programming teaching tool, helps engineering students to better understand how their codes works.*

Key-words: *Arduino, Constructivism, Programming.*

Organização



Promoção

