

Um robô por aluno: uma realidade possível

Rafael V. Aroca¹, Renata Pitta¹, Aquiles Burlamaqui¹, Luiz M. G. Gonçalves¹

¹DCA – Universidade Federal do Rio Grande do Norte (UFRN)
Natal – RN – Brazil

{aroca,pitta,aquiles,lmarcos}@natalnet.br

Abstract. *We present here some works developed at the NatalNet laboratory that make possible projects such as “one robot per student” from the economical and technical points of view. The described solution can be applied for students of all ages, making the system suitable for pre-scholar activities, basic level teaching, high school, technical courses and graduate courses. The low cost of the didactic robot presented in this work is due to the fact that it does not have a processor, as it relies on the mobile phone of its user as the control unit (in Brazil, there is more than one mobile phone for each inhabitant). Moreover, a user-friendly programming platform is also presented.*

Resumo. *Apresentamos neste artigo alguns trabalhos desenvolvidos no laboratório NatalNet que tornam viável iniciativas do tipo “um robô por aluno” tanto do ponto de vista econômico quanto técnico. A solução descrita pode ser usada com alunos de todas as idades, possibilitando seu uso em atividades pré-escolares, no ensino básico, fundamental, médio, técnico e superior. O baixo custo do robô didático discutido neste trabalho deve-se ao fato deste não possuir processador, já que ele conta com o telefone celular do usuário como unidade de controle (no Brasil existe mais de um telefone celular para cada habitante). Além disto, uma plataforma de programação amigável também é apresentada.*

1. Introdução

O uso de robótica em ambientes educacionais vem se tornando cada vez mais presente nos mais diversos níveis de ensino, desde o ensino para pequenas crianças até na graduação e pós graduação. De fato, diversos autores já demonstraram que o uso de robôs na educação proporciona ambientes estimulantes e motivadores [Soto et al. 2006, Hamblen and Hall 2004, Alves et al. 2011, Howard and Graham 2007], além de uma experiência única de aprendizado [Weinberg and Yu 2003]. De fato, Rawat e Massiha demonstram dados confirmando que o retorno dos alunos em aulas que envolvem robótica é “supreendentemente positivo” [Rawat and Massiha 2004].

Contudo, existem dois principais problemas para a adoção mais abrangente de tecnologias de robótica na educação: os altos custos e a dificuldade de uso. Com relação aos custos, Alves et al também alertam que os altos custos de robôs podem impedir seu uso em salas de aula [Alves et al. 2011, Galvan et al. 2006]. Além do mais, Lumsden e Ortega-Sanchez afirmam que existe uma escassez de plataformas robóticas de baixo custo [Lumsden and Ortega-Sanchez 2010]. Este problema é ainda mais grave no Brasil, onde muitas escolas e alunos sofrem dificuldades econômicas, não podendo adquirir facilmente robôs ou kits de robótica.

A questão de facilidade de uso também é peça fundamental para o sucesso de projetos de robótica educacional. O uso do sistema deve ser fácil não apenas para os alunos, mas também para os professores, já que muitos educadores não possuem grande experiência com *softwares* de robótica. Neste sentido, Dipietro et al explicam que quando o *software* de um sistema computacional é intuitivo, a adoção em massa deste produto é uma consequência [Dipietro et al. 2008], citando o exemplo do *tablet* pioneiro, o *iPad* da *Apple*.

Neste artigo propomos uma solução para esses problemas através do uso de um robô de baixo custo (cerca de R\$ 50,00) que utiliza o telefone celular dos alunos como unidade de controle. Como o Brasil possui mais de 1,3 telefones celulares para cada habitante [Teleco 2012a], acreditamos que caso um aluno não possua um telefone celular, ele poderá obter um facilmente, mesmo que seja um modelo antigo que alguém não use mais. Para facilitar o uso, todo ambiente de programação do robô fica embarcado no celular, e pode ser acessado por qualquer computador via *web*, dispensando a necessidade de instalação e configuração de *softwares* específicos. O ambiente permite tanto programação gráfica em alto nível conectando blocos, quando em baixo nível usando várias linguagens de programação. Com esta solução, esperamos tornar viável uma realidade de “um robô por aluno”.

Este texto está organizado da seguinte forma: primeiro apresentamos na seção 2 alguns trabalhos relacionados que incluem sistemas robóticos de baixo custo e ambientes de programação. Em seguida, a seção 3 apresenta nossa proposta de solução com detalhes para ser reproduzida. Finalmente, a seção 4 apresenta nossas considerações finais.

2. Trabalhos relacionados

A ideia de um “robô por aluno” (URA), que pode ser uma extensão do projeto já existente “um computador por aluno” (UCA), foi sugerida várias vezes pelo Prof. Dr. Aquiles Burlamaqui em comunicações pessoais realizadas no laboratório NatalNet-UFRN. Uma proposta semelhante feita pelo Prof. James McLurkin, de uma universidade americana, ganhou destaque da imprensa internacional no final do ano de 2011, bem como recebeu diversos prêmios pela proposta de um robô por aluno de engenharia.

McLurkin defende um curriculum acadêmico onde cada aluno deve ter seu próprio robô, permitindo assim testes e estudos de lições individuais, bem como a realização de trabalhos em equipe [Rice University News 2011]. Para tanto, McLurkin criou uma plataforma robótica de baixo custo (segundo o autor) chamada *R-One* que custa cerca de US\$200,00. O autor também abriu uma empresa e pretende comercializar este robô a partir de 2012. Ele acredita que este custo possibilite a adoção em massa destes robôs em escolas. No momento, cada estudante novato da universidade de Rice, nos EUA, recebe seu próprio robô *R-One* no primeiro dia de aula [Rice University News 2011].

Também com a intenção de possibilitar que cada aluno tenha seu próprio robô para fazer lições em casa, Lauwers desenvolveu o *Finhch*, um robô com custo de US\$99,00 já à venda no mercado americano [Inovação Tecnológica 2012]. Este robô inclui sensores de temperatura, luminosidade, acelerômetro, sensor de toque, LEDs, câmera, microfone e alto falantes. Segundo Lauwers, o foco deste robô não é para ensino de robótica, mas para facilitar e tornar mais atrativo o aprendizado de programação de computadores (na versão atual usando Java). A intenção desta iniciativa é tornar aulas de programação

mais interessantes, dispensando a necessidade dos alunos construírem seus próprios robôs [Inovação Tecnológica 2012]. Deve-se notar que este robô não é autônomo: ele não possui baterias e opera somente quando conectado a um PC via cabo USB. Todos algoritmos são executados no PC, que envia comandos para leitura de sensores e movimento das rodas [Lauwers and Nourbakhsh 2010].

Outros autores também apresentam projetos de robôs de baixo custo com valores entre US\$ 200,00 e US\$ 1.000,00 [Greenwald and Kopena 2003, Dinh and Inanc 2009, Hamblen and Hall 2004, Lumsden and Ortega-Sanchez 2010]. Nota-se assim que o termo “baixo custo” é relativo. Neste intervalo de preço também se encontra o Lego NXT Mindstorms, uma plataforma robótica amplamente usada em ambientes escolares. Uma tecnologia nacional para ambientes educacionais é o robô Curumin, comercializado pela empresa Xbot. Embora ele já seja adotado em diversas escolas, seu custo é ainda maior que os valores já mencionados. Este robô oferece uma ferramenta de programação para plataforma Windows, que permite aos alunos definirem as tarefas do robô conectando blocos graficamente [Netto et al. 2012], assim como é feito no ambiente do Lego NXT.

Um dos problemas dos sistemas mencionados é que todos eles requerem a instalação de *softwares* específicos e configuração de computadores. Embora esta tarefa não seja altamente complexa, ela pode ser difícil de ser realizada por professores de escolas de nível médio e fundamental sem experiência em informática. Além do mais, para instalação dos *softwares*, é necessário privilégios de administrador no computador, dificultando a instalação dos ambientes de programação em computadores de uso público como escolas, bibliotecas e tele-centros.

Uma solução proposta por diversos autores, é disponibilizar ambientes de programação e desenvolvimento de robótica via *Web* [Belousov et al. 2001, Garrett and Thornton 2005, Aroca et al. 2012b, Popescu et al. 2008], de forma que qualquer computador ou dispositivo com um navegador *Web* pode ser usado para programar e controlar o robô, dispensando a necessidade de instalação de *softwares*. Um projeto de destaque é o *Mindstorms Internet Control Environment (MICE)* [Garrett and Thornton 2005], que permite programar e controlar Kits Lego via *Web* usando uma variação da linguagem C. Outro projeto interessante, criado pelo Google e atualmente mantido por pesquisadores do MIT, é o *MIT App Inventor*, um sistema *Web* que permite criar aplicações para telefones celulares com sistema operacional Android de forma gráfica (conectando blocos), incluindo comandos para controle de Legos Mindstorms, e em seguida gerar um pacote de instalação para ser instalado em telefones com o sistema Android [MIT 2012].

O *MIT App Inventor* explora uma possibilidade interessante, que é o uso de um telefone celular como unidade de controle de um robô, e permite que toda programação seja feita via *web* e de forma gráfica. Entretanto, tanto este como os outros projetos mencionados têm custos relativamente altos para escolas com poucos recursos financeiros e certamente para que cada estudante adquira o seu próprio robô. Na seção 3 descrevemos nossa proposta para oferecer um sistema com alta flexibilidade, poder computacional e baixo custo.

3. Nossa solução

Nossa proposta consiste em uma técnica para construção de robôs didáticos usando um telefone celular como unidade de controle, como descrito originalmente por Aroca et al [Aroca et al. 2012a]. Os telefones celulares atuais têm capacidade de processamento entre 500 e 1000 vezes superiores às capacidades dos computadores de bordo das naves Apollo que foram para a Lua. Além do mais, os telefones contam com diversos recursos úteis para um robô, como por exemplo acelerômetros, câmeras, reconhecimento de voz, síntese de voz e acesso à Internet.

O uso de telefones celulares como base de uma plataforma robótica é ainda mais interessante no Brasil, já que em Maio de 2012, o Brasil tinha aproximadamente 255 milhões de celulares [Teleco 2012a] (1,3 celulares para cada habitante [Teleco 2012a]). Além disto, o telefone celular, é o segundo equipamento eletrônico mais presente na vida de nossa população. Como se sabe, a televisão é o aparelho eletrônico predominante nos domicílios brasileiros. Assim, uma pesquisa feita em 2010, que incluiu áreas rurais, mostrou que 95% dos lares possuem televisão, seguido de 83% que possuem celulares. Já os lares com microcomputador são 38% [Teleco 2012b]. Além disto, outra pesquisa mostra que 77% das famílias que ganham menos de 10 salários mínimos possuem celulares. Para as pessoas que ganham mais que 10 salários, este número é superior a 97% [Teleco 2011]. Outra vantagem desta abordagem é reaproveitar telefones celulares usados, já que estes aparelhos são frequentemente trocados no Brasil.

Tipicamente os robôs educativos são construídos com microcontroladores, como o PIC, o MSP430 ou o AtMega (popular por ser usado no Arduino). Estes sistemas são práticos e de baixo custo, porém limitados em capacidade computacional, além de necessitarem diversos sensores externos, e um nível de conhecimento intermediário de eletrônica e computação para seu uso. Quando se deseja construir robôs que devem executar algoritmos mais sofisticados e sistemas e visão de máquina, normalmente se usa um notebook ou netbook como unidade de controle, permitindo assim executar algoritmos mais avançados, usar a câmera, Internet e outros recursos de um PC. A solução que propomos tem a vantagem de aproveitar diversos sensores já disponíveis em um telefone celular, e atingir capacidade de processamento comparável com a de um netbook. A tabela 1 faz uma breve comparação de cada um destes sistemas. Note que tanto no robô com PC quanto com microcontrolador, diversos sensores adicionais devem ser comprados, conectados, e trabalho adicional de programação deve ser feito para poder usar estes sensores.

3.1. Hardware

Para que o celular possa controlar o robô, basta conectar nosso circuito à saída de fones de ouvido do celular, e através de diferentes frequências de áudio (análogas às notas musicais: cada nota tem uma frequência) geradas pelo celular, um circuito eletrônico decodifica estas frequências e controla o movimento de cada motor. Até dois motores podem ser ligados em cada canal de áudio, de forma que a saída estéreo típica permite controlar 4 motores, ou 2 motores e 4 quatro atuadores do tipo liga/desliga simultaneamente. A figura 1 mostra o diagrama elétrico completo do circuito de controle. Notamos que embora tenhamos mencionado um celular com sistema Android, qualquer aparelho

Recurso	Netbook	Telefone (*)	Microcontrolador
Velocidade da CPU / Memória	1GHz/1GB	500MHz/512MB	40MHz/8KB
Duração da bateria	8-12h	10-20h	20h (\$25)
Câmera	VGA	2x6 MegaPixel	+\$100 + trabalho
Receptor GPS	+\$200	Embutido	+\$200 + trabalho
Acelerômetro	+\$70	Embutido	+\$70 + trabalho
Wifi / Bluetooth	Embutido	Embutido	+\$250 + trabalho
Reconhecimento de voz	Médio	Ótimo	+ trabalho
Bússola	+\$70	Embutido	+\$70 + trabalho
Microfone / alto falantes	Embutido	Embutido	+\$20 + trabalho
Visão computacional	OpenCV	OpenCV	+ trabalho
Linguagens de programação	Muitas	Muitas	Assembler, C
Controle de motores	+\$100	+\$50	Embutido
Tamanho	Médio	Pequeno	Pequeno
Preço (R\$)	700 (1140)	250 (300)	100 (835)

Tabela 1. Tabela comparativa de unidades de controle para robôs. O preço final é o preço apenas da unidade de controle, e o valor entre parênteses é o preço da unidade de controle com os sensores. (*) Preço de um telefone celular com sistema operacional Android no mercado brasileiro em Jun/2012 com os recursos mencionados

que possa gerar sons pode ser usado. Assim, o sistema não é limitado a um celular, já que qualquer outro dispositivo que emita sons pode ser usado, como um PC, um *tablet*, ou até mesmo um reproduzidor de músicas (MP3 por exemplo). A tabela 2 mostra um resumo dos custos do circuito.

O circuito é baseado no trabalho de Aroca et al. [Aroca et al. 2012a], onde maiores detalhes podem ser encontrados, inclusive detalhes da extensão deste sistema para leitura de estado de sensores também usando áudio. No circuito da figura 1, um decodificador (MT8870) de *Dual Tone Multi Frequency*, DTMF, recebe o sinal sonoro do celular e o decodifica em 4 bits. Estes bits são usados para controlar uma chave analógica (4066N) que altera o valor da resistência conectada ao circuito gerador de pulsos (555). O 555, por sua vez, gera um sinal do tipo *Pulse Width Modulation*, PWM, com ciclo determinado pelo resistor escolhido via sinal de áudio. Dois servo-motores são conectados a cada um dos 555, e o sinal de PWM controla cada motor de forma independente com comandos de frente e ré. Variações no intervalo do áudio enviado para o circuito podem controlar a velocidade dos motores. Os resistores variáveis (*trimpots*) são para ajuste fino de velocidade e direção de cada motor, e podem ser substituídos por resistores convencionais após estabelecer um valor de resistência desejável para cada robô.

Os motores utilizados são servos tipicamente usados em aeromodelos. A grande vantagem destes motores é seu baixo custo, e o fato deles já incluírem caixa de redução e circuito de controle de motor, reduzindo os gastos. Eles podem ser usados diretamente para construir pequenos braços robóticos. Caso o projeto seja um robô móvel, eles devem ser modificados para rotação contínua. Recomendamos o uso de polar-rotores de antenas parabólicas, já que estes já permitem rotação contínua, sendo necessário apenas desconectar os terminais do sensor de posição e ligar dois resistores de mesmo valor no lugar

Item	Quantia	Valor (R\$)
CI MT8870	1	6,90
CI 555	2	1,70
CI 4066	1	1,30
Diversos (resistores, capacitores, etc)	1	2,00
Motores (polar rotor de parabólica)	2	28,00
Placa de circuito impresso	1	9,00
TOTAL		48,9

Tabela 2. Custos do protótipo construído com peças compradas em pequena quantia. Material comprado em Jul/2012

deste sensor para que ele funcione em rotação contínua.

A figura 2 mostra o robô proposto construído usando partes de brinquedos velhos em sua estrutura.

3.2. Software

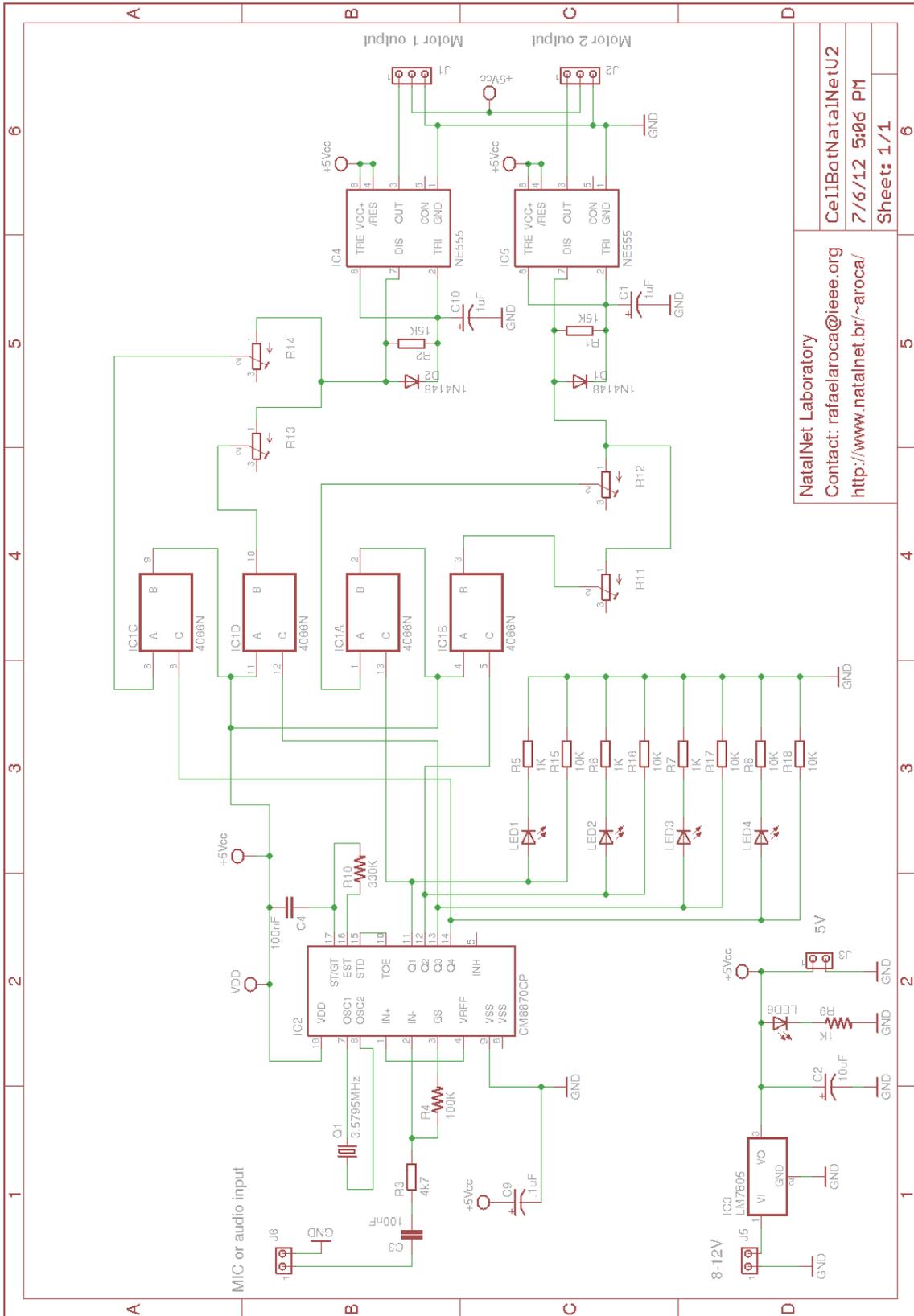
Como prova de conceito, desenvolvemos um *software* para controle e programação dos robôs para plataforma de celulares Android. Se necessário, este sistema pode ser desenvolvido para outras plataformas de telefonia celular.

O ambiente de programação foi batizado como *anwide: Android web based integrated development environment*, e disponibilizado na Internet como *software* livre, de forma que tanto o *software* quanto o código fonte (*open source*), podem ser baixados livremente a partir do site *web* <http://code.google.com/p/anwide/>.

Após ser instalado e executado no telefone, o *anwide* mostra o IP (endereço do dispositivo na rede) do celular onde ele está instalado. Para possuir este endereço, o celular deve estar conectado na Internet, em uma rede local ou operando no modo roteador (AndroidAP). Em seguida, o usuário deve usar um navegador *web*, em qualquer computador conectado à mesma rede, ou no próprio telefone e acessar o IP exibido na tela. Ao acessar o sistema, o usuário terá acesso ao aplicativo *Web* mostrado na figura 3. Lembramos que o *anwide* é executado completamente no telefone do usuário, dispensando conexão com Internet ou o uso de qualquer computador auxiliar.

Como pode ser visto na figura 3, a área do lado direito permite construir um programa apenas encaixando blocos. Quando o programa está pronto, o usuário clica em um botão que transforma os blocos em um programa na linguagem Python, e o executa no celular. Embora nossos testes tenham sido feitos em Python, outras linguagens são suportadas. Se o usuário desejar, ele pode editar, também via *Web*, diretamente o código fonte Python e executá-lo, permitindo alto grau de personalização e a possibilidade de programação avançada.

Embora ainda não tenhamos implementado todas funções do sistema operacional Android, já é possível usar a síntese de voz deste sistema. No momento estamos trabalhando em blocos para controle de robôs Lego Mindstorms e blocos para leitura de



NataINet Laboratory
 Contact: rafaelaroca@ieee.org
 http://www.natainet.br/~aroca/

CellBotNataINetU2
 7/6/12 5:06 PM
 Sheet: 1/1

Figura 1. Diagrama elétrico do circuito para controle de dois motores a partir de tons de áudio provenientes de um telefone celular

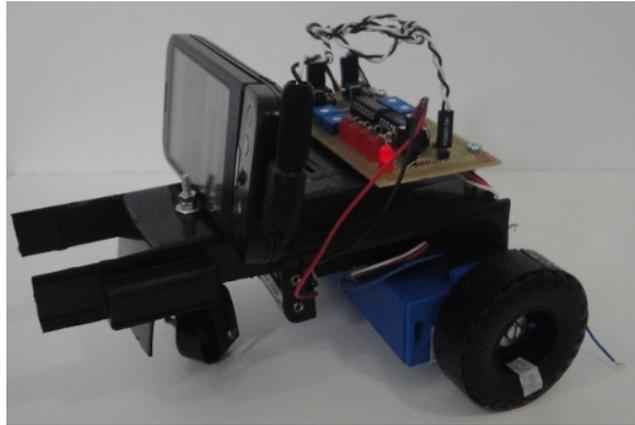


Figura 2. Robô de baixo custo construído usando um telefone celular como computador de controle

sensores do telefone celular. Estes recursos já estão funcionais usando diretamente a linguagem Python. Mais detalhes sobre o uso da linguagem Python para leitura de sensores do celular com sistema Android e controle de robôs podem ser encontrados no trabalho de Aroca et al [Aroca et al. 2012b].

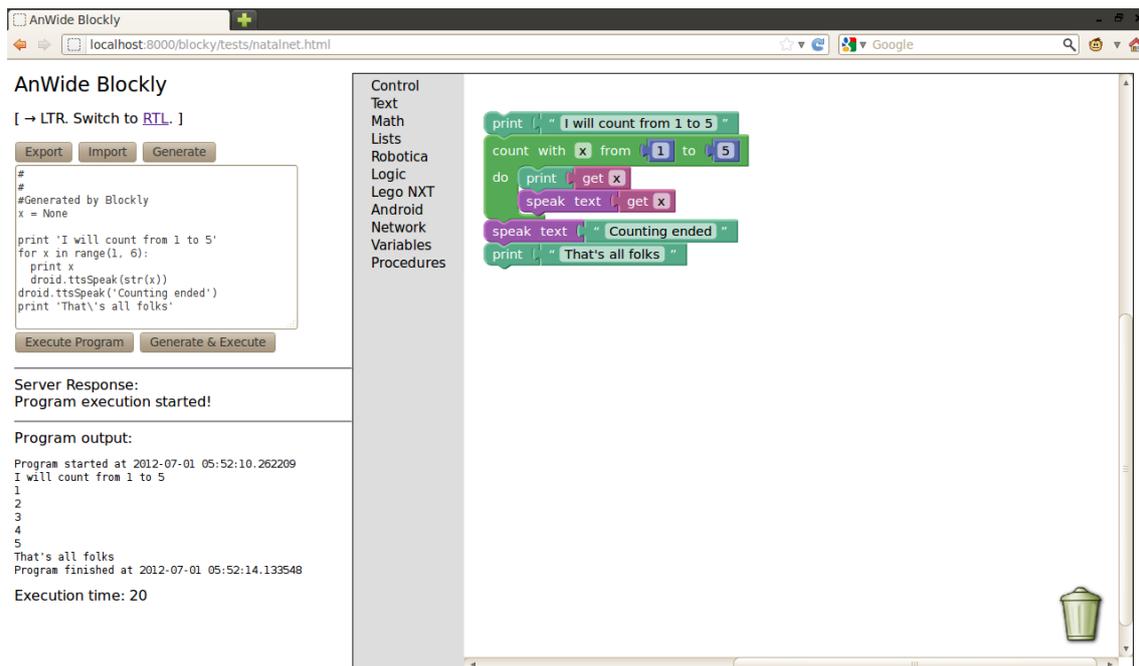


Figura 3. Captura da tela do sistema Web embarcado no celular, o *anwide*. No lado direito o programa pode ser feito conectando os blocos. No lado esquerdo, o programa é traduzido para código fonte Python, e na parte esquerda inferior o resultado da execução do programa é mostrada

4. Conclusão

Apresentamos neste trabalho uma alternativa de baixo custo para construir robôs educativos com alto grau de flexibilidade e poder de processamento. O sistema pode ser

construído por cerca de R\$50,00 (sem contar o valor do telefone) quando as peças são compradas em pequena quantia em lojas especializadas. Se compradas em maior quantia, para vários alunos, os preços podem ser reduzidos. Para 100 unidades, por exemplo, o valor reduz cerca de 30%.

Devido ao seu baixo custo, torna-se mais fácil oferecer este sistema para cada aluno, facilitando a implementação de programas do tipo “um robô por aluno”. Com isto, os alunos podem levar seus robôs para casa, bibliotecas, telecentros, e outros ambientes para experimentar e fazer lições. Como o sistema de programação é baseado na *Web*, qualquer computador com navegador *web* pode ser usado.

Lembramos que nosso sistema é patenteado, o que garante que qualquer pessoa pode construir e replicar o sistema livremente e sem custo para fins de pesquisa e uso pessoal, mas caso uma empresa venha a vender um sistema baseado em nossa técnica, devem ser pagos *royalties* à UFRN.

5. Agradecimentos

Agradecemos o apoio e o financiamento do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

Referências

- Alves, S., Filho, H., Pegoraro, R., Caldeira, M., Rosario, J., and Yonezawa, W. (2011). Proposal of educational environments with mobile robots. In *Robotics, Automation and Mechatronics (RAM), 2011 IEEE Conference on*, pages 264–269.
- Aroca, R. V., Burlamaqui, A. F., and Gonçalves, L. M. G. (2012a). Method for reading sensors and controlling actuators using audio interfaces of mobile devices. *Sensors*, 12(2):1572–1593.
- Aroca, R. V., Guardiman, R. Q., and Gonçalves, L. M. G. (2012b). Web-based robot programming environment and control architecture (aceito para publicação). In *Latin American Robotics Symposium / Simpósio Brasileiro de Robótica*.
- Belousov, I. R., Chellali, R., and Clapworthy, G. (2001). Virtual reality tools for internet robotics. In *ICRA*, pages 1878–1883. IEEE.
- Dinh, H. and Inanc, T. (2009). Low cost mobile robotics experiment with camera and sonar sensors. In *American Control Conference, 2009. ACC '09.*, pages 3793–3798.
- Dipietro, L., Sabatini, A., and Dario, P. (2008). A survey of glove-based systems and their applications. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 38(4):461–482.
- Galvan, S., Botturi, D., Castellani, A., and Fiorini, P. (2006). Innovative robotics teaching using lego sets. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 721–726.
- Garrett, A. and Thornton, D. (2005). A web-based programming environment for lego mindstorms robots. In *Proceedings of the 43rd annual Southeast regional conference - Volume 2, ACM-SE 43*, pages 349–350, New York, NY, USA. ACM.
- Greenwald, L. and Kopena, J. (2003). Mobile robot labs. *Robotics Automation Magazine, IEEE*, 10(2):25–32.

- Hamblen, J. and Hall, T. (2004). Engaging undergraduate students with robotic design projects. In *Electronic Design, Test and Applications, 2004. DELTA 2004. Second IEEE International Workshop on*, pages 140 – 145.
- Howard, A. and Graham, E. (2007). To encourage and excite the next generation of engineers through human-robot interaction projects for space exploration. In *American Society for Engineering Education Annual Conference, Hawaii, June 2007, Hawaii, June*.
- Inovação Tecnológica (2012). Cientistas propõem um robô por aluno. Redação do Site Inovação Tecnológica. On-Line. Disponível em: <http://www.inovacaotecnologica.com.br/noticias/noticia.php?artigo=um-robo-poraluno>. Acesso em Jul/2012.
- Lauwers, T. and Nourbakhsh, I. (2010). Designing the finch: Creating a robot aligned to computer science concepts. In *AAAI Symposium on Educational Advances in Artificial Intelligence*.
- Lumsden, J. and Ortega-Sanchez, C. (2010). Modular autonomous robotics platform for educational use. In *TENCON 2010 - 2010 IEEE Region 10 Conference*, pages 1577 –1582.
- MIT (2012). Explore mit app inventor. On-Line. Disponível em: <http://explore.appinventor.mit.edu/>. Acesso em Jul/2012.
- Netto, A. V., Miranda, F., Silva, W., and Freitas, Y. (2012). A block programming interface for educational mobile robots. In *Robocontrol: Proceedings of 5th workshop in applied robotics and automation*.
- Popescu, D., Selisteanu, D., Dinulescu, I., and Popescu, L. (2008). Web based telematics application for robotics. In *Computing in the Global Information Technology, 2008. ICCGI '08. The Third International Multi-Conference on*, pages 19 –24.
- Rawat, K. and Massiha, G. (2004). A hands-on laboratory based approach to undergraduate robotics education. In *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on*, volume 2, pages 1370 – 1374 Vol.2.
- Rice University News (2011). All for one, 'r-one' for all. On-Line. Disponível em: <http://engineering.rice.edu/NewsContent.aspx?id=3491>. Acesso em Jul/2012.
- Soto, A., Espinace, P., and Mitnik, R. (2006). A mobile robotics course for undergraduate students in computer science. In *Robotics Symposium, 2006. LARS '06. IEEE 3rd Latin American*, pages 187 –192.
- Teleco (2011). Estatísticas de domicílios brasileiros (ibge - pnad). On-Line. Disponível em: <http://www.teleco.com.br/pnad.asp>. Acesso em Jul/2012.
- Teleco (2012a). Estatísticas de celulares no brasil. On-Line. Disponível em: <http://www.teleco.com.br/ncel.asp>. Acesso em Jul/2012.
- Teleco (2012b). Estatísticas do brasil - geral. On-Line. Disponível em: <http://www.teleco.com.br/estatis.asp>. Acesso em Jul/2012.
- Weinberg, J. and Yu, X. (2003). Robotics in education: Low-cost platforms for teaching integrated systems. *Robotics Automation Magazine, IEEE*, 10(2):4 –6.