

**DEVRY | FANOR**

**CURSO GESTÃO DE TECNOLOGIA DA INFORMAÇÃO**

**INFORMÁTICA NA EDUCAÇÃO: EVOLUÇÃO DOS MODELOS E METODOLOGIAS EDUCACIONAIS**

Judah Holanda Correia Lima

Matr. 171013705

Abril - 2017

JUDAH HOLANDA CORREIA LIMA

**INFORMÁTICA NA EDUCAÇÃO: EVOLUÇÃO DOS MODELOS E METODOLOGIAS EDUCACIONAIS**

Trabalho da Disciplina Temas Tecnológicos em Humanidade, do Curso de Gestão de Tecnologia da Informação, da DeVry | Fanor

Orientadora: Alyne Ricarte

Fortaleza – Ceará

2017

LISTA DE FIGURAS

[Figura 35: Imagem do resultado da tradução de um código de Divisão Easembly para os Assemblies do 8051 e do z80 e sua versão feita manualmente para 8051 11](file:////Users/Judah/Articles/TEMAS%20TECNOLÓGICOS%20EM%20HUMANIDADES.docx#_Toc481003564)

[Figura 36: Imagem do resultado da tradução de um código de Divisão Easembly para os Assemblies do 8051 e do z80 e sua versão feita manualmente para 8051 13](#_Toc481003565)

# **RESUMO**

A tecnologia evolui exponencialmente, e isso resulta em mudanças em todos os setores, e a educação não é diferente. Portabilidade, acessibilidade a uma infinidade de informação devido a internet, interatividade com o conceito de gamification, interatividade, integração interdisciplinar, cooperação e uma série de facilidades que a informática possibilita ou, pelo menos, facilita. Nesse artigo será visto a evolução na educação proporcionada pela informática.

**Palavras-chave**: Montadores *Assembly online*. *Softwares.* Estudo de uma arquitetura.

SUMÁRIO

[RESUMO 4](#_Toc481003410)

[INTRODUÇÃO 6](#_Toc481003411)

[1 OBJETIVO 6](#_Toc481003412)

[2 METODOLOGIA 6](#_Toc481003413)

[3 DESENVOLVIMENTO DO TEMA 6](#_Toc481003414)

[CONSIDERAÇÕES FINAIS 6](#_Toc481003415)

[REFERÊNCIAS 8](#_Toc481003416)

[ANEXO 13](#_Toc481003417)

# **INTRODUÇÃO**

# 1 OBJETIVO

Demostrar a importância da tecnologia da informação na educação, mostrando da educação proporcionado pela tecnologia.

# 2 METODOLOGIA

A metodologia utilizada para a criação desta arquitetura será realizada a partir da pesquisa e estudo do estado da arte de projetos, sistemas e artigos que se assemelham ao tema e, assim, demonstrar a importância da tecnologia da informação na educação.

# 3 DESENVOLVIMENTO DO TEMA

# CONSIDERAÇÕES FINAIS

A partir da análise dos resultados obtidos foi possível perceber que a tecnologia permite uma revolução no sistema de ensino atual. No entanto existem alguns problemas.

Então,

O trabalho futuro

# REFERÊNCIAS

[1] CANFORA, G. ***Migrating interactive legacy systems to Web services***. *European Conference. Bari*. 22-24 *March* 2006. Disponível em: <http://ieeexplore. ieee.org/xpl/login.jsp?tp=&arnumber=1602355&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxpls%2Fabs\_all.jsp%3Farnumber%3D1602355>. Acesso em: 26 maio 2015.

[2] KIM, S.M.; ROSU, M.C. ***A Survey of Public Web Services***. *E-Commerce and Web Technologies*.*Springer/Berlin- Heidelberg*, 2004. Disponível em: <http://link.springer.com/book/10.1007%2Fb100074>. Acesso em: 26 maio 2015.

[3] CHU, K.W.S.; KENNEDY, D.M. *Using online collaborative tools for groups to co-construct knowledge*. ***Online Information Review***, v. 35, n. 4, p. 581-597, 2011.

[4] MILLER, M. ***Cloud Computing****: Web-Based Applications That Change the Way You Work and Collaborate Online. Que Publishing*, 2008.

[5] BOURNE, J.; HARRIS, D;; MAYADAS, F. *Online Engineering Education: Learning Anywhere, Anytime*. ***Journal of Engineering Education***. p. 130-146, 2005.

[6] HYDE, R. *The Art of Assembly Language*. No Starch Press, 2003.

[7] SMITH, J.R. ***Programming the PIC Microcontroller with MBasic***. EUA: *Newnes*, 2005.

[8] KUAN-CHENG, L. ***An On-line Instruction/Learning Environment for Supporting Individualized Learning in Java Programming***. *Taiwan*: *National Chung Hsing University*, 2007.

[9] MORE, A. ***Web Based Programming Assistance Tool for Novices****. International Conference: Chennai, Tamil Nadu, 14-16 July 2011*. Disponível em: <http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=6004399&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxpls%2Fabs\_all.jsp%3Farnumber%3D6004399>. Acesso em: 26 maio 2015.

[10] DATTA, A. ***Online compiler as a cloud service***. *International Conference: Ramanathapuram, 8-10 May 2014*. Disponível em: <<http://ieeexplore.ieee.org/xpl/> login.jsp?tp=&arnumber=7019416&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxpls%2Fabs\_all.jsp%3Farnumber%3D7019416>. Acesso em: 26 maio 2015.

[11] MOHAMMED, S. ***WIDE an interactive Web integrated development environment to practice C programming in distance education***. *International Conference:* Porto*, Oct. 31 2013 - Nov. 1 2013*. Disponível em: <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6701964>. Acesso em: 26 maio 2015.

[12] Minzhe, G. ***Back to Results Design of Online Runtime and Testing Environment for Instant Java Programming Assessment***. *International Conference:* Washington DC*, Abr*. 12 2010. Disponível em: < http://dl.acm.org/citation.cfm?id=1848870>. Acesso em: 26 maio 2015.

[13] ANSARI, A.N. ***Online C/C++ compiler using cloud computing***. *International Conference: Hangzhou,* 26-28 July 2011. Disponível em: <http://ieeexplore.ieee.org/ xpl/login.jsp?tp=&arnumber=6002124&url=http%3A%2F%2Fieeexplore.ieee.org%2Fiel5%2F5981419%2F6001647%2F06002124.pdf%3Farnumber%3D6002124>. Acesso em: 26 maio 2015.

[14] SALIN MD, S.I. ***Back to Results One-pass Assembler Design for a Low-end Reconfigurable RISC processor***. *International Conference:* Washington DC*, Jun. 20* 2014. Disponível em: <http://www.jatit.org/volumes/Vol64No2/20Vol64No2.pdf>. Acesso em: 26 maio 2015.

[15] ARBONE, C. **M*odel-Driven Inline Assembler Generator for Retargetable Compilers***. *International Conference:* Bucharest*,* 29-31 May 2013. Disponível em: <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6569246>. Acesso em: 26 maio 2015.

[16] Source Code Editor. PCMag. Disponível em:< <http://www.pcmag.com/encyclopedia/term/51762/source-code-editor>>. Acesso em: 26 maio 2015.

[17] Using the Metadata API to Retrive Picklist Values. Selesforce Developers. Disponível em:<<https://developer.salesforce.com/blogs/developer-relations/2008/12/using-the-metadata-api-to-retrieve-picklist-values.html>>. Acesso em: 26 maio 2015.

[18] It’s All About Intelligent Code Completion. Selesforce Developers. Disponível em:< http://code-recommenders.blogspot.com.br/2010/05/its-all-about-intelligent-code.html >. Acesso em: 26 maio 2015.

[19] Code::Blocks Wiki FAQ. Code::Blocks Wiki. Disponível em:< <http://wiki.codeblocks.org/index.php?title=FAQ> >. Acesso em: 26 maio 2015.

[20] Qt Creator Completing Code. Qt Documentation. Disponível em:< <http://doc.qt.io/qtcreator/creator-completing-code.html> >. Acesso em: 26 maio 2015.

[21] Using Intellisense. Microsoft Documentation. Disponível em:< https://msdn.microsoft.com/en-us/library/hcw1s69b.aspx >. Acesso em: 26 maio 2015.

[22] FARIAS, G; MEDEIROS, E. S. *Introdução a Computação*. Disponível em:< http://producao.virtual.ufpb.br/books/camyle/introducao-a-computacao-livro/livro/livro.pdf >. Acesso em: 26 maio 2015. 88p.

[23] BARRETO, J.M. S. *Introdução a Computação*. Disponível em:< http://www.inf.ufsc.br/~barreto/cca/arquitet/arq4.htm >. Acesso em: 26 maio 2015.

[24] MANSSOUR, I.H. S. *Linguagem de Programação C*. Disponível em:< http://www.inf.pucrs.br/manssour/LinguagemC/ >. Acesso em: 26 maio 2015.

[25] GOSLING, James; JOY, Bill; STEELE, Guy; BRACHA, GILAD; Buckley, Alex. The Java® Language Specification. Java SE. 8 ed. Redwood City – California, 2014.

[26] *Write Once Run Anywhere. Computer Weekly*. Disponível em:< http://www.computerweekly.com/feature/Write-once-run-anywhere >. Acesso em: 26 maio 2015.

[27] *The Java Language Environment*. Oracle. Disponível em:< http://www.oracle.com/technetwork/java/intro-141325.html >. Acesso em: 26 maio 2015.

[28] *Java No Longer A Favorite*. Wired. Disponível em:< <http://www.wired.com/2013/01/java-no-longer-a-favorite/> >. Acesso em: 26 maio 2015.

[29] The RedMonk Programming *Language Rankings*. RedMonk. Disponível em:< <http://redmonk.com/sogrady/2015/01/14/language-rankings-1-15/> >. Acesso em: 26 maio 2015.

[30] *The History Of Java Technology*. Oracle. Disponível em:< <http://www.oracle.com/technetwork/java/javase/overview/javahistory-index-198355.html> >. Acesso em: 26 maio 2015.

[31] BUYYA, Rajkumar; SELVI, S. Thamarai; CHU, Xingchen. Object-oriented Programming with Java: essentials and Applications. New Delhi: Tata McGraw-Hill Education, 2009.

[32] VENNERS, B. Inside The Java Virtual Machine. Artima Developer. . Disponível em:< <http://www.artima.com/insidejvm/ed2/index.html>

>. Acesso em: 26 maio 2015.

[33] LINDHOLM, T; YELLIN, F; BRACHA, G; Buckley A. The Java Virtual Machine Specification. Java SE. 7 ed. Redwood City – California, 2013.

[34] *Bytecode Basics: A First Look At The Bytecodes Of The Java Virtual Machine*. Java World. Disponível em:< <http://www.javaworld.com/article/2077233/core-java/bytecode-basics.html> >. Acesso em: 26 maio 2015.

[35] Java *Bytecode: Understanding Bytecode Makes You A Better Programmer*. IBM developerWorks. Disponível em:< http://www.ibm.com/developerworks/library/it-haggar\_bytecode/ >. Acesso em: 26 maio 2015.

[36] Overview Of The .Net Framework. MSDN Microsoft. Disponível em:< https://msdn.microsoft.com/en-us/library/zw4w595w.aspx >. Acesso em: 26 maio 2015.

[37] Common Language *Runtime (CLR)*. MSDN Microsoft. Disponível em:< https://msdn.microsoft.com/en-us/library/8bs2ecf4.aspx >. Acesso em: 26 maio 2015.

[38] Object Oriented Programming Languages Pattern. Disponível em:< https://github.com/Judahh/Pattern/blob/master/General/Object%20Oriented%20Programming%20Languages%20Pattern.mediawiki >. Acesso em: 26 maio 2015.

[39] COOK, A; JARVIS, J; LEE, J. Evolving The Google Identity. Google Design. Disponível em:< https://design.google.com/articles/evolving-the-google-identity/ >. Acesso em: 26 maio 2015.

[40] MCCATHIENEVILE, C; KOIVUNEN, M. Accessibility Features of SVG. W3C. Disponível em:< http://www.w3.org/TR/SVG-access/ >. Acesso em: 26 maio 2015.

[41] HATFIELD, B.O; ZHANG, M.; ZHEN-LAN, J. ***A General-Purpose Custom-Design Edassembler*** *In C. Frontiers in Education*: IEEE, v. 2, p. F3C-1-2 5-8 Nov. 2003.

[42] TAVERNIER, K.R.; NOTREDAME, P.H. *Macro-Based Cross Assemblers.* ***Software Engineering, IEEE Transactions on***: IEEE, v. 6, p. 334 – 340, July 1980.

[43] NAKANO, K; [ITO, Y.](http://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Ito,%20Y..QT.&newsearch=true) *Processor, Assembler, and Compiler Design Education using an FPGA. Parallel and Distributed Systems: IEEE.* p. 723-728. *International Conference on Melbourne*, VIC, 8-10 Dec. 2008.

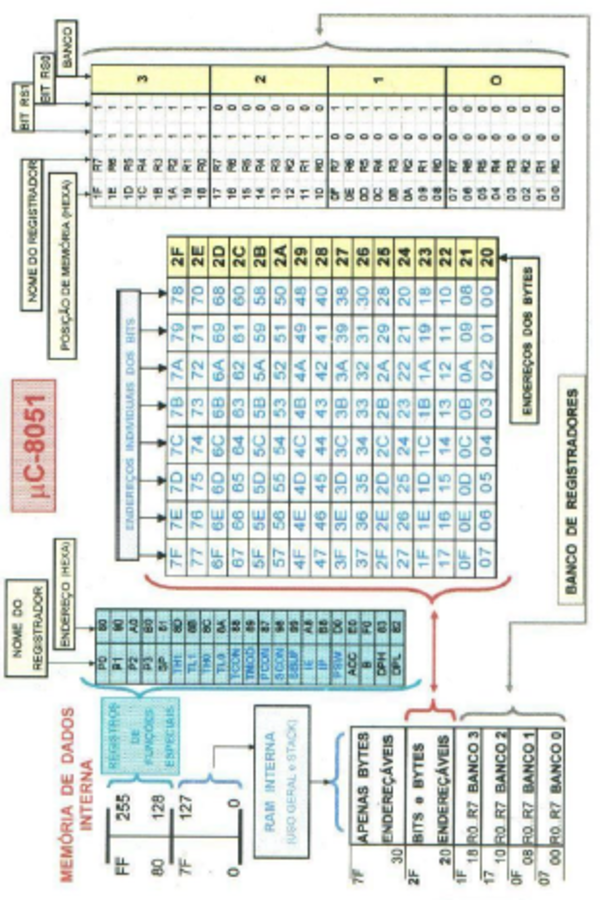
[44] MATHISKE, B. SIMON, D,; UNGAR, D. *An assembler and disassembler framework for JavaTM programmers*. ***Science of Computer Programming***, v. 70, p. 127–148, 2008.

[45] DAVIS, I.J.; GODFREY, M.W. *From Whence It Came: Detecting Source Code Clones by Analyzing Assembler. Working* ***Conference on Beverly****, MA. Reverse Engineering*: IEEE, p. 242–246,13-16 Oct. 2010.

[46] SALIN, S.I.; SULAIMAN, H.A.; JAMALUDDIN, R.; SALAHUDDIN, L.; ZAINUDIN, M.N.S.; SALIM, A.J. *Two-pass Assembler Design for a Reconfigurable RISC Processor.* ***Conference on Kuching****. Open Systems*: IEEE, p. 77-82, 2-4 Dec. 2013.

# ANEXO

**Processadores**

Foram selecionados dois processadores com os quais o sistema se preocupará, são eles: o 8051 e o Z80.

No 8051, tem-se a seguinte configuração:

Figura 35: Imagem do resultado da tradução de um código de Divisão Easembly para os Assemblies do 8051 e do z80 e sua versão feita manualmente para 8051

**Arquitetura**

***Assembly***

Todos os membros da família 8051 executam o mesmo conjunto de instruções, composto por orientações otimizadas para aplicações de controle, facilitando as operações de dados, por meio de vários modos de endereçamento. Capacita, ainda, a operação de variáveis de um *bit*, permitindo operação em sistemas que demandam processamento booleano. Adiante, descreve-se brevemente o modo de operação de várias instruções contidas neste conjunto. Maiores informações podem ser encontradas nos manuais dos fabricantes, os quais descrevem detalhadamente o conjunto de instruções.

Já no Z80 tem-se o seguinte:

**Arquitetura**

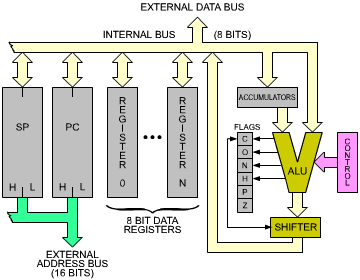


Figura 36: Imagem do resultado da tradução de um código de Divisão Easembly para os Assemblies do 8051 e do z80 e sua versão feita manualmente para 8051

***Assembly***

O 8080 (base do Z80) usa instruções que podem variar seu tamanho de 1 até 3 *bytes*. No entanto, o Z80 é equipado com instruções indexadas adicionais, as quais podem requerer um *byte* a mais. No caso do Z80, os *opcodes* são, em geral, um *byte*, exceto para instruções especiais, que requerem dois *bytes* de *opcode*.

Algumas instruções requerem que um *byte* de dados siga o *opcode*. Neste caso, esta instrução será uma de 2 *bytes*, em que o segundo *byte* consta de dados (exceto para indexação, o qual adiciona um *byte* extra). Para o restante das instruções, é possível que seja requerida a especificação de um endereço.

Um endereço requer 16 *bits* (2 *bytes*). Sabendo disto, uma instrução que use endereço pode variar, em tamanho, de 3 a 4 *bytes*.

Para cada *byte* de instrução, a unidade de controle deverá realizar um *fetch* de memória, o qual requer 4 ciclos de *clock*. Então, isso significa que quanto menor a instrução, mais rápida será a execução (assim como a maioria dos processadores).