

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE ELETRÔNICA

MIKHAIL ANATHOLY KOSLOWSKI

**KIT DIDÁTICO DE BAIXO CUSTO PARA O ENSINO DE
MICROCONTROLADORES DA FAMÍLIA MCS-51 USB**

TRABALHO DE CONCLUSÃO DE CURSO

CURITIBA

2014

MIKHAIL ANATHOLY KOSLOWSKI

**KIT DIDÁTICO DE BAIXO CUSTO PARA O ENSINO DE
MICROCONTROLADORES DA FAMÍLIA MCS-51 USB**

Trabalho de conclusão de curso apresentado ao Departamento Acadêmico de Eletrônica como requisito parcial para obtenção do grau de Engenheiro no curso de Engenharia Industrial Elétrica com Ênfase em Eletrônica e Telecomunicações da Universidade Tecnológica Federal do Paraná.

Orientador: Gustavo Benvenuti Borba

CURITIBA

2014

RESUMO

KOSLOWSKI, Mikhail. KIT DIDÁTICO DE BAIXO CUSTO PARA O ENSINO DE MICROCONTROLADORES DA FAMÍLIA MCS-51 USB. 81 f. Trabalho de conclusão de curso – Departamento Acadêmico de Eletrônica, Universidade Tecnológica Federal do Paraná. Curitiba, 2014.

Cursos de nível técnico, tecnólogo e de engenharia em eletrônica e correlatos utilizam kits didáticos para o ensino de microcontroladores. A arquitetura MCS-51 ainda é uma bastante utilizada para este fim, por apresentar características como conjunto de instruções CISC, amplo suporte, grande disponibilidade de ferramentas de desenvolvimento e de modelos de microcontroladores. Este trabalho apresenta o desenvolvimento de um novo kit didático para o ensino de microcontroladores da família MCS-51, denominado P51USB. Um dos aspectos de destaque deste projeto está no uso do microcontrolador AT89C5131A-M, que possui um controlador USB *Full Speed* integrado, além de outros periféricos. A presença do controlador USB permite a utilização do kit sem a necessidade de circuitos gravadores externos ou de portas seriais e paralelas que não são usuais em computadores modernos, além de dispensar a utilização de uma fonte de alimentação externa. O kit P51USB ainda conta com conversor analógico digital, relógio de tempo real, barramento para display LCD e porta serial para depuração. Para facilitar a utilização do kit P51USB pelos alunos, foram desenvolvidos também tutoriais, detalhando a operação do kit, e exemplos de códigos para os seus principais periféricos. O kit foi utilizado de forma piloto por uma turma de Engenharia Eletrônica da UTFPR campus Curitiba no segundo semestre do ano de 2013, obtendo uma avaliação satisfatória por parte dos professores e alunos.

Palavras-chave: Microcontroladores, Kit didático, AT89C5131, 8051

ABSTRACT

KOSLOWSKI, Mikhail. EDUCATIONAL LOW COST KIT FOR TEACHING MICROCONTROLLER OF USB MCS-51 FAMILY. 81 f. Trabalho de conclusão de curso – Departamento Acadêmico de Eletrônica, Universidade Tecnológica Federal do Paraná. Curitiba, 2014.

Technical, technologist, engineering and related courses in electronics use educational kits for teaching microcontrollers. The MCS-51 architecture is still widely used for this purpose, due to its characteristics such as CISC instruction set, extensive support, wide availability of both development tools and models for microcontrollers. This project presents the development of a new educational kit for teaching MCS-51 family microcontrollers, called P51USB. The innovation of this project lies in its use of the AT89C5131A-M microcontroller. In addition to all MCS-51 characteristics, it has a USB Full Speed controller and some other peripherals. The presence of the USB controller allows the use of the kit without the need for any external circuits or serial and parallel ports to write the microcontroller's memory and exempt the use of an external power supply. The P51USB kit also has an integrated analogic to digital converter, real time clock, LCD display bus and a serial port for debugging. This saves time on mounting those circuits and helps focusing on learning the microcontroller. To ease the use of P51USB kit by students, tutorials detailing the operation of the kit have also been developed, as well as code samples for their key peripherals. The kit was used as a pilot project for classes of an Electronic Engineering course in the second half of the year 2013 on UTFPR Curitiba's campus, obtaining a satisfactory evaluation by teachers and students.

Keywords: Microcontrollers, Educational kit, AT89C5131, 8051

LISTA DE FIGURAS

FIGURA 1	– Placa didática P51N.	14
FIGURA 2	– Diagrama em blocos do microcontrolador AT89C5131A-M.	20
FIGURA 3	– Fluxo de verificações após o <i>reset</i>	23
FIGURA 4	– Fotografia do <i>kit</i> P51USB completo.	25
FIGURA 5	– Diagrama em blocos do sistema.	26
FIGURA 6	– Utilização das chaves para acessar os três modos de operação do <i>kit</i>	28
FIGURA 7	– Tela inicial do Keil uVision 4.	32
FIGURA 8	– Janela para escolha do nome e local de armazenamento do projeto do uVision.	33
FIGURA 9	– Tela para escolha do microcontrolador.	33
FIGURA 10	– Tela para escolha do microcontrolador.	34
FIGURA 11	– Menu para acessar as configurações do projeto.	35
FIGURA 12	– Configuração para criação de arquivo Intel HEX.	36
FIGURA 13	– Configuração para a não inclusão automática dos símbolos do 8051.	36
FIGURA 14	– Criação de um novo arquivo.	37
FIGURA 15	– Janela com as opções para salvar arquivo.	38
FIGURA 16	– Menu para adicionar arquivos ao projeto.	39
FIGURA 17	– Opções para assembler e linkar o projeto.	40
FIGURA 18	– Opções para montar e ligar o projeto.	41
FIGURA 19	– Saída do programa em caso de sucesso.	41
FIGURA 20	– Exemplo de dispositivo corretamente enumerado no gerenciador de dispositivos do Windows.	43
FIGURA 21	– Tela inicial do FLIP, com destaque para a seleção do microcontrolador. ..	43
FIGURA 22	– Tela para a seleção do microcontrolador.	44
FIGURA 23	– Seleção do meio de comunicação.	44
FIGURA 24	– Tela da conexão USB.	45
FIGURA 25	– Item do menu para o carregamento do arquivo Intel HEX.	46
FIGURA 26	– Informações sobre o programa carregado.	46
FIGURA 27	– Início do processo de gravação.	47
FIGURA 28	– Fim do processo de gravação.	48
FIGURA 29	– Placa executando o programa <i>hello world</i>	49
FIGURA 30	– Configuração para utilização do FlashMon51.	50
FIGURA 31	– Escolha dos parâmetros da porta serial.	50
FIGURA 32	– Detalhe da porta serial no gerenciador de dispositivos do Windows.	51
FIGURA 33	– P51USB em modo <i>debug</i>	52
FIGURA 34	– Opção do menu para iniciar sessão de download.	52
FIGURA 35	– Computador e P51USB em processo de depuração.	53

LISTA DE SIGLAS

ADC	Conversore analógico-digital
CDC	<i>Communication Device Class</i>
CI	Circuito integrado
DAC	Conversores digital-analógico
DAELN	Departamento Acadêmico de Eletrônica
DAINF	Departamento Acadêmico de Informática
EUART	<i>Enhanced universal asynchronous receiver/transmitter</i>
HID	<i>Human Interface Device</i>
I/O	Entrada e saída
IDE	Ambiente de Desenvolvimento Integrado
ISP	<i>In-System Programming</i>
JRE	<i>Java Runtime Enviroment</i>
MEC	Ministério da Educação
PCA	<i>Programmable Counter Array</i>
PWM	Modulação por largura de pulso
RTC	Relógio de tempo real
SO	Sistema operacional
UART	<i>Universal asynchronous receiver/transmitter</i>
UTFPR	Universidade Tecnológica Federal do Paraná

SUMÁRIO

Projeto Técnico	10
1 INTRODUÇÃO	11
1.1 JUSTIFICATIVA	12
1.2 OBJETIVOS	15
1.2.1 Objetivo Geral	15
1.2.2 Objetivos Específicos	15
1.3 METODOLOGIA	16
1.3.1 Etapa 1	16
1.3.2 Etapa 2	17
1.3.3 Etapa 3	17
1.4 APRESENTAÇÃO DO DOCUMENTO	18
2 MICROCONTROLADOR AT89C5131A-M	19
2.1 CONTROLADOR USB 2.0 <i>FULL SPEED</i>	20
2.2 MEMÓRIA FLASH E O <i>BOOTLOADER</i>	21
2.3 <i>RESET</i>	22
2.4 <i>I/O</i>	22
2.5 <i>PROGRAMMABLE COUNTER ARRAY</i>	23
2.6 DRIVER DE LEDS	23
2.7 SPI E I2C	24
2.8 EUART/RS232	24
3 KIT P51USB	25
3.1 ADC0832	26
3.2 DS1307	26
3.3 BARRAMENTO PARA DISPLAY LCD 2X16	27
3.4 FLASHMON51	28
3.5 BIBLIOTECA DE ROTINAS <i>ASSEMBLY</i>	28
3.5.1 <i>Header</i> padrão	29
3.5.2 LCD	29
3.5.3 ADC via SPI	29
3.5.4 RTC via I2C	29
3.5.5 Outros exemplos	30
4 OPERAÇÃO DO KIT	31
4.1 GRAVAÇÃO	31
4.1.1 <i>Hello world</i>	31
4.1.2 Requisitos	31
4.1.3 Criando um programa com o uVision	32
4.1.3.1 Criando um projeto no uVision	32
4.1.3.2 Configurando o projeto	34
4.1.3.3 Adicionando o arquivo com o código fonte	37

4.1.3.4 Montando e ligando	40
4.1.4 Gravando o programa com o FLIP	42
4.1.4.1 Colocando a placa em modo <i>bootloader</i>	42
4.1.4.2 Conectando com o FLIP	43
4.1.4.3 Carregando o arquivo Intel HEX	45
4.1.4.4 Gravando	47
4.1.5 Testando o programa	48
4.2 DEPURAÇÃO	49
4.2.1 Gravação do FlashMon51	49
4.2.2 Configuração do uVision	49
4.2.3 Colocar a placa em modo <i>Debug</i>	51
4.2.4 Carga do programa via FlashMon51	52
5 CONSIDERAÇÕES FINAIS	54
6 PUBLICAÇÕES	55

Plano de Negócio 56

7 PLANO DE NEGÓCIO	57
7.1 SUMÁRIO EXECUTIVO	57
7.2 DEFINIÇÃO DO NEGÓCIO	58
7.2.1 Visão	58
7.2.2 Missão	58
7.2.3 Valores	58
7.2.4 Descrição do Negócio	58
7.3 OBJETIVOS	59
7.3.1 Objetivo Principal	59
7.3.2 Objetivos Intermediários	59
7.4 PRODUTOS E SERVIÇOS	59
7.4.1 Descrição do Produto e dos Serviços	59
7.4.2 Análise Comparativa	59
7.4.3 Tecnologia	60
7.4.4 Produtos e Serviços Futuros	60
7.5 ANÁLISE DE MERCADO RESUMIDA	61
7.5.1 Segmentação de Mercado	61
7.5.2 Segmento Alvo	61
7.5.2.1 Necessidades do Mercado	62
7.5.2.2 Tendências do Mercado	62
7.5.2.3 Crescimento do Mercado	62
7.5.3 Análise da Indústria	63
7.5.3.1 Players	63
7.5.3.2 Modelo de Distribuição	63
7.5.3.3 Modelo de Competitividade	63
7.5.3.4 Principais Players	64
7.6 PLANO FINANCEIRO	64
7.6.1 Considerações	64
7.6.2 Investimento Inicial	64
7.6.3 Projeção de Resultados e de Fluxo de Caixa	64

7.6.4 Indicadores Financeiros	64
7.6.5 Análise de Break-even	65
7.6.6 Viabilidade do Empreendimento	65
REFERÊNCIAS	68
Apêndice A – ESQUEMÁTICO	71
Apêndice B – ARQUIVOS GERBER	76

PROJETO TÉCNICO

1 INTRODUÇÃO

Os microcontroladores estão presentes em praticamente todos os dispositivos eletrônicos que são utilizados atualmente, nas mais diversas aplicações, como por exemplo equipamentos de telecomunicações, automação e controle, entretenimento, medicina e transportes. Portanto, é extremamente importante que estudantes de Engenharia Eletrônica e de Computação possuam uma base sólida no projeto de sistemas microcontrolados, uma vez que estes também formam a base a base para a análise e síntese de sistemas embarcados mais complexos.

Os cursos de Engenharia Eletrônica e de Computação comumente apresentam os microcontroladores em um disciplina específica ou integrados à disciplina de arquitetura e organização de computadores (REESE; JONES, 2010; BOLANAKIS et al., 2011). Em geral, a disciplina desenvolve-se a partir de um modelo de microcontrolador específico pertencente a uma das famílias mais comuns disponíveis no mercado, como a Intel MSC51 (HSU; CHAO, 2009; ZHANG, 2009; SELVAKUMAR et al., 2011), Motorola MC68 (BOLANAKIS et al., 2011), Atmel AVR (HOVEMEYER; LINK, 2011; MA et al., 2010; ARDUINO, 2014), Microchip PIC (KEANE et al., 2012; REESE; JONES, 2010), Texas Instruments MSP430 (TI, 2014; REESE; JONES, 2010) ou ARM (ST, 2014). Existem, ainda, propostas de arquiteturas dedicadas ao uso didático (GAL et al., 2011; J et al., 2006). Os tópicos considerados fundamentais em um curso convencional de microcontroladores são a arquitetura do modelo de microcontrolador adotado, a programação em linguagem *assembly* e a comunicação com dispositivos de entrada e saída (I/O).

A metodologia mais comum adotada para o ensino de microcontroladores é a que emprega aulas teóricas e práticas, durante as quais utilizam-se como principais ferramentas um programa do tipo Ambiente de Desenvolvimento Integrado (IDE) e um hardware com o microcontrolador adotado, denominado *kit* de desenvolvimento ou *kit* didático. O IDE pode ser obtido gratuitamente em versões de avaliação sem restrição de tempo de uso, com limite para a geração de programas de 4 kbytes (KEIL, 2014; RAISONANCE, 2014; IAR, 2014), o que é suficiente para o contexto de uma disciplina. Já o *kit* didático precisa ser adquirido de

terceiros ou desenvolvido pela instituição de ensino. Ainda, os *kits* podem ser disponibilizados pela instituição para o aluno apenas para uso durante as aulas ou adquirido pelo próprio aluno.

Assim, este trabalho compreende o desenvolvimento de um novo *kit* didático, chamado P51USB, para a disciplina Microcontroladores das atuais matrizes curriculares dos cursos de Engenharia Eletrônica e de Computação da Universidade Tecnológica Federal do Paraná (UTFPR). O desenvolvimento se deu em torno do microcontrolador AT89C5131A-M da ATMEL pois este abrange todas as características desejadas.

1.1 JUSTIFICATIVA

Segundo dados obtidos nos sites do Departamento Acadêmico de Eletrônica (DAELN) e do Departamento Acadêmico de Informática (DAINF) da UTFPR, assim como a partir de conversas com ex-alunos e professores dos cursos de Engenharia Eletrônica e de Computação, foi possível apurar o histórico das disciplinas que envolvem microcontroladores, bem como a família de microcontrolador utilizada por cada uma delas. Este histórico pode ser observado no quadro 1.

Desde a disciplina mais antiga citada no quadro 1, todas utilizaram um *kit* didático como a principal ferramenta de ensino. Pelo fato do mercado de microcontroladores mostrar-se bastante dinâmico, estes *kits* didáticos necessitam de constantes atualizações. As principais razões para isso são a obsolescência dos componentes utilizados, a opção pela adoção de outro modelo de microcontrolador, ou até mesmo de uma família diferente e mais atual de microcontrolador.

No momento, os professores que atuam na disciplina Microcontroladores dos cursos de Engenharia Eletrônica e de Computação da UTFPR relatam a necessidade da atualização do *kit* didático. O *kit* utilizado atualmente, que utiliza a família de microcontroladores MCS51, exibido na figura 1, é denominado P51N.

Atualmente, os principais problemas relacionados ao *kit* P51 são:

- Ausência de suporte nativo à interface de comunicação USB;
- Utilização de memória de programa externa, o que requer o uso de ports de uso geral como barramentos de endereços e dados, tornado-os portanto indisponíveis para o interfaceamento assíncrono com dispositivos de I/O;
- Utilização de uma CI de memória RAM modelo 62256, cuja disponibilidade no mercado tende a ser cada vez menor;

Curso	Matriz	Em vigor até	Disciplina	Período	Família
Engenharia Eletrônica	5	Atualmente	Microcontroladores	6	MCS51
			Arquitetura e organização de computadores	6	x86(PC)
			Sistemas Embarcados	8	ARM Cortex-M3
	4	2006	Digital II	5	MCS51
			Digital III	6	x86(PC)
			Microcontroladores	Optativa	
	3	2000	Digital II	5	MCS51
			Digital III	6	
			Microcontroladores	Optativa	
	2	1995	Sistemas Digitais II	9	Z80
Microcontroladores			Optativa		
Engenharia de Computação	721	Atualmente	Arquitetura e organização de computadores	5	MIPS
			Sistemas Microcontrolados	6	MCS51
			Sistemas Embarcados	8	ARM Cortex-M3

Quadro 1: Histórico das disciplinas que utilizam microcontroladores nos cursos de Engenharia da UTFPR.

Fonte: (DAELN, 2014) e (DAINF, 2014)

- Ausência de suporte nativo a interfaces de comunicação entre CIs, especificamente I2C e SPI;
- Impossibilidade de gravar o programa do usuário na flash do microcontrolador sem a utilização de um gravador externo (o programa do usuário é carregado na RAM volátil externa);
- Necessidade de uma fonte de alimentação DC externa.

Estes problemas do atual *kit* didático implicam em uma maior dificuldade para os alunos no desenvolvimento de projetos mais sofisticados, como:

- Impossibilidade de utilização do *kit* em aplicações stand-alone (exceto quando há a disponibilidade de um gravador externo);
- Tempo de adaptação elevado ao uso do *kit* comparado a soluções mais recentes;

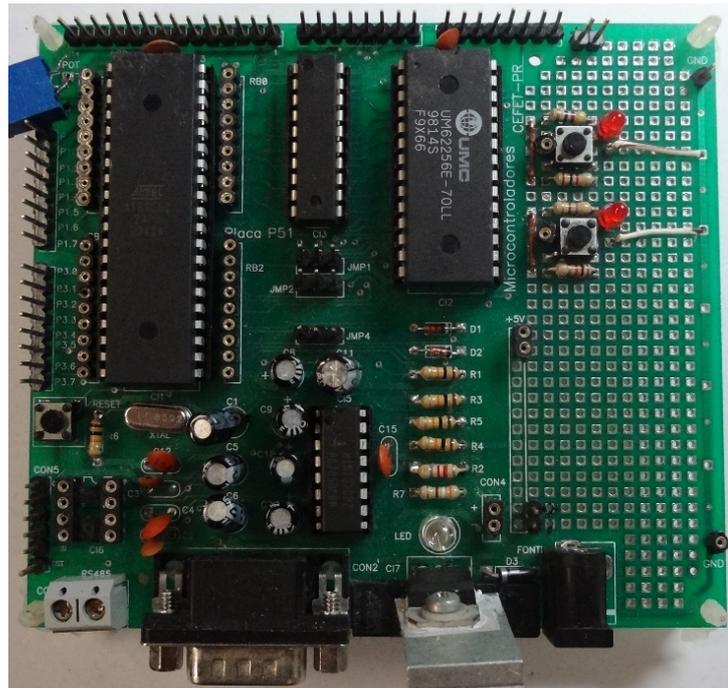


Figura 1: Placa didática P51N.

Fonte: Autoria própria.

- Necessidade do uso de um conversor USB-serial, já que a interface RS232 é ausente na grande maioria dos computadores modernos.

É importante observar que o *kit* didático P51N foi desenvolvido no final da década de 90 e atualizado em 2004 (FARIA; LOPES, 2014). Durante este período de cerca de 15 anos, considerando o cenário tecnológico em questão, o *kit* atendeu com sucesso aos requisitos técnicos e pedagógicos exigidos pela disciplina. No entanto, devido às limitações mencionadas anteriormente, existe a necessidade do desenvolvimento de um novo *kit*.

Deve-se considerar que existem, ainda, pelo menos duas outras abordagens para a obtenção de um *kit* didático mais atualizado. A primeira seria a aquisição, por parte dos alunos, de um *kit* didático desenvolvido e manufaturado por terceiros (CERNE, 2014; KIT8051, 2014; MICROGENIOS, 2014). Porém, *kits* para a família de microcontroladores MCS51 costumam apresentar um custo relativamente alto. A segunda abordagem seria a aquisição de um *kit* de terceiros pela instituição, e não pelos alunos. Uma das desvantagens desta abordagem está no alto investimento necessário para a aquisição de uma quantidade de *kits* suficientes para a demanda estabelecida por várias turmas. Outra desvantagem está na dificuldade da aquisição dos componentes necessários para a manutenção dos *kits*, pois os procedimentos de compras em instituições públicas costumam ser complicados e demorados.

Ainda, é importante mencionar que, uma vez que os professores da disciplina Microcontroladores fazem uso extensivo do *kit* didático, o cenário mais adequado é realmente aquele no qual o aluno é o proprietário do *kit*, podendo assim desenvolver as atividades também em casa, sem estar restrito aos horários da aula e aos recursos disponíveis na instituição. Para isso, é importante que, além de atender aos requisitos pedagógicos, o *kit* apresente um custo acessível.

Também, é importante destacar que o *kit* não está limitado ao uso na disciplina Microcontroladores, pois pode ser uma plataforma para o desenvolvimento de trabalhos e outras disciplinas, como as de Controle, Processamento Digital de Sinais, Metrologia, Trabalho de Conclusão de Curso, entre outras.

1.2 OBJETIVOS

1.2.1 OBJETIVO GERAL

Desenvolver um *kit* didático para o ensino de microcontroladores da família MCS51 e sua documentação técnica para ser utilizado na disciplina Microcontroladores dos cursos de Engenharia Eletrônica e Computação da UTFPR.

1.2.2 OBJETIVOS ESPECÍFICOS

1. Desenvolver um hardware, denominado *kit* didático, que possua um microcontrolador da família MCS51 adequado às necessidades atuais da disciplina Microcontroladores e que possa ser adquirido e montado pelos próprios alunos;
2. Desenvolver a documentação técnica do *kit* (manual) para ser utilizada no processo de ensino e aprendizagem;
3. Desenvolver *firmwares* de exemplo contemplando a utilização de todos os periféricos disponíveis no *kit* em linguagem *assembly* e/ou C (preferencialmente em *assembly*);
4. Elaborar roteiros de experimentos que possam servir de base para aulas práticas realizadas em laboratório;
5. Desenvolver um site para a disponibilização de todo o material referente ao *kit*, nas modalidades *open-source* hardware e *open-source* software;
6. Implantar, em conjunto com os professores da disciplina Microcontroladores, a utilização da plataforma;

7. Avaliar a possibilidade de utilização da interface USB no processo de depuração em substituição à interface RS-232.

1.3 METODOLOGIA

As especificações deste projeto foram discutidas com os professores das disciplinas envolvidas em Microcontroladores, as quais irão se beneficiar do *kit* desenvolvido. Um dos pontos principais deste projeto, conforme mencionado anteriormente, é a obtenção de um produto final de baixo custo e fácil operação pelos alunos. Assim, os recursos envolvidos no seu desenvolvimento também são, na sua maioria, financeiramente acessíveis e de fácil obtenção.

Com base nestas discussões e visando atingir todos os objetivos, o projeto foi dividido em 3 etapas listadas a seguir.

1.3.1 ETAPA 1

1. Especificação de todos os componentes que irão compor o *kit*: Selecionar o microcontrolador, os principais periféricos e qualquer outro componente crítico que seja parte do *kit*;
2. Projeto do esquemático: Com base nas especificações dos componentes selecionados na etapa anterior, compor o esquemático elétrico;
3. Projeto da placa de circuito impresso (PCI): Com base no esquemático elétrico desenvolvido na etapa anterior e nas especificações dos componentes selecionados, criar o layout da placa de circuito impresso;
4. Fabricação do primeiro protótipo: Escolher um ou mais fabricantes de PCI para produção de uma ou mais placas para serem testadas e iniciar o processo de compra dos componentes para a montagem do protótipo;
5. Testes do protótipo: Montar e testar uma ou mais PCIs fabricadas na etapa anterior. Nesta etapa será possível determinar qual fabricante de PCI, dentre aqueles escolhidos na etapa anterior, é o mais adequado para o projeto;
6. Desenvolvimento da documentação técnica: Ao longo de toda a etapa 1 será desenvolvida parte da documentação técnica do *kit*.

A passagem da etapa 1 para a etapa 2, determinada Gate 1, será no momento em que o protótipo estiver completamente testado.

1.3.2 ETAPA 2

1. Correção de eventuais problemas encontrados durante os testes: De acordo com os testes efetuados na etapa 1, os erros serão corrigidos para que não ocorram na versão final do *kit*;
2. Fabricação da versão final da PCI: A versão final da PCI será fabricada no fabricante selecionado na etapa 1, e servirá como base para a documentação final do *kit* e os roteiros de experimentos;
3. Avaliação da possibilidade da utilização da interface USB no processo de depuração: Será avaliada a possibilidade de substituição da interface RS-232 no processo de depuração. Independente do resultado desta etapa, a versão final do *kit* não sofrerá alterações e, portanto, esta tarefa poderá ser postergada ou até mesmo descartada em prol das demais;
4. Desenvolvimento dos roteiros de experimentos: Como parte da documentação do *kit* serão desenvolvidos alguns roteiros de experimentos que façam o uso extensivo do *kit* e que poderão ser utilizados pelos professores da disciplina Microcontroladores como base em aulas de laboratório;
5. Desenvolvimento de *firmwares* de exemplo: Como parte da documentação do *kit* serão desenvolvidos programas de exemplo de todas as funcionalidades do *kit*, que poderão servir como base para o desenvolvimento de projetos mais complexos;
6. Conclusão da documentação técnica: Alterações eventuais decorrentes da correção de algum erro encontrado nos testes;
7. Desenvolvimento do site e disponibilização da documentação: O Site será desenvolvido para divulgação e disponibilização da documentação técnica do *kit* para o público e será inspirado naquele desenvolvido por Simon Schubert para o *kit* MC HCK (SCHUBERT, 2014).

A passagem da etapa 2 para a etapa 3, determinada Gate 2, será no momento em que a versão final da placa estiver fabricada, testada e completamente documentada.

1.3.3 ETAPA 3

1. Conclusão de relatório do trabalho: Concluir o relatório que será entregue como parte dos requisitos para aprovação na disciplina Trabalho de conclusão de curso 2;

2. Apresentação do trabalho à banca avaliadora: Apresentação pública do trabalho como parte dos requisitos para aprovação na disciplina Trabalho de conclusão de curso 2.

1.4 APRESENTAÇÃO DO DOCUMENTO

Este documento descreve o processo de desenvolvimento do Trabalho de Conclusão de Curso do aluno Mikhail Anatholy Koslowski do curso de Engenharia Industrial Elétrica com ênfase em Eletrônica e Telecomunicações da UTFPR.

Inicialmente é apresentado o contexto do projeto, os problemas e as premissas envolvidas no desenvolvimento do projeto e a metodologia escolhida. Na sequência são descritas as tecnologias escolhidas e o desenvolvimento de cada uma das etapas. A seguir são apresentados os resultados obtidos e a situação atual do projeto.

Por fim, é apresentado o plano de negócio desenvolvido para este projeto, contemplando análise de mercado e a viabilidade econômica do projeto.

2 MICROCONTROLADOR AT89C5131A-M

O AT89C5131A-M é uma versão de microcontrolador da família MCS51 desenvolvido pela ATMEL com memória flash interna para código, controlador USB *Full Speed* (USB-IF, 2014) além uma série de periféricos integrados para facilitar o desenvolvimento de aplicações.

Além de todas as características comuns aos microcontroladores da família MCS51, algumas das principais características específicas deste microcontrolador são:

- Modo X2, executando uma instrução a cada 6 ciclos de clock;
- Três timers/contadores de 16bits;
- Controlador USB integrado compatível com os padrões 1.1 e 2.0 *Full Speed*;
- PLL interno para geração de clock para USB;
- 32kB de memória flash interna para gravação de código via USB;
- 3kB de memória flash interna pré gravada com programa *bootloader* USB;
- 1kB de memória EEPROM para dados;
- 1kB de memória RAM interna;
- Programmable Counter Array de 5 canais com contador interno de 16 bits e capacidade de geração de PWM;
- Interface TWI compatível com padrão I2C;
- Interface SPI com suporte aos modos mestre e escravo;
- 4 drivers de led com fontes internas de corrente programável;
- 34 pinos de IO.

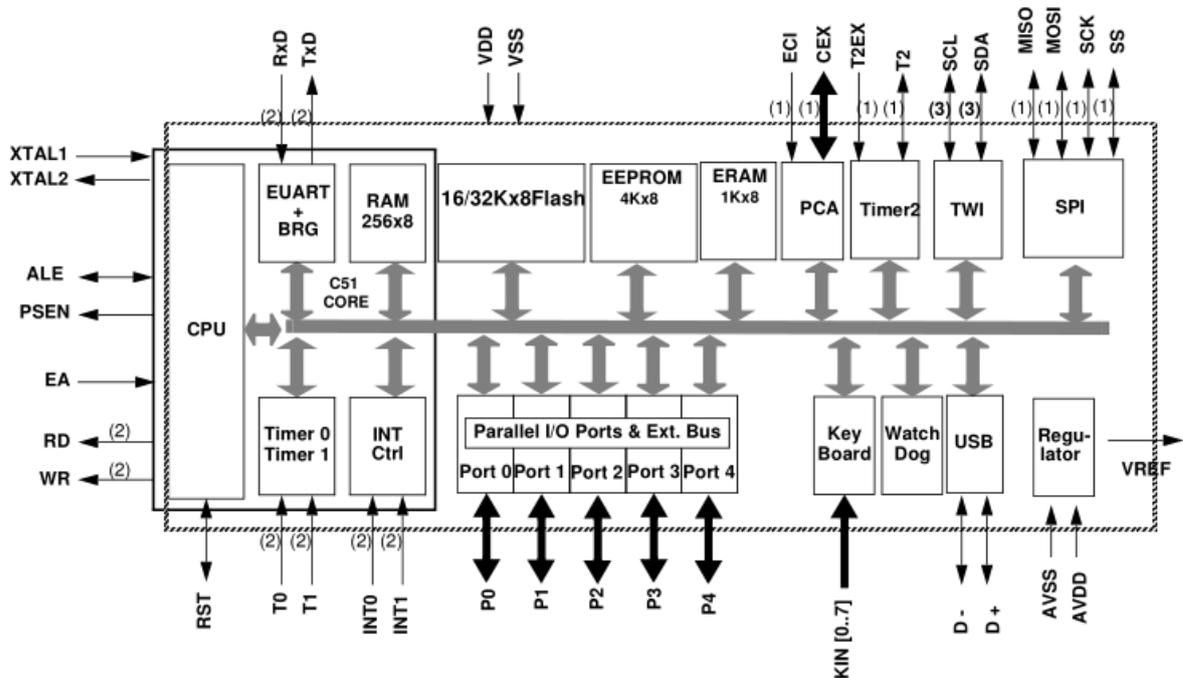


Figura 2: Diagrama em blocos do microcontrolador AT89C5131A-M.

Fonte: (ATMEL, 2008)

A figura 2 apresenta o diagrama em blocos do AT89C5131A-M.

A descrição de todas as funcionalidades deste microcontrolador pode ser encontrada no seu *datasheet* (ATMEL, 2008). Nas próximas seções serão detalhadas aquelas que foram utilizadas na composição do *kit*.

2.1 CONTROLADOR USB 2.0 FULL SPEED

O barramento serial universal, USB, foi desenvolvido a partir de meados da década de 90 com o objetivo de ser um substituto aos barramentos que eram então utilizados para conexão de periféricos ao computador como a porta serial, porta paralela, PS/2, entre outros que além de limitar o uso dos periféricos não forneciam suporte a configuração automática por meio do sistema operacional(SO) (AXELSON, 2001). As principais características do barramento USB são:

Padronização do conector - Um único tipo de conector pode ser utilizado para conectar diversos tipos diferentes de periféricos ao computador host;

Configuração automática - O SO é capaz de se comunicar com todos os dispositivos USB

através de uma interface padrão e através desta comunicação inicial decidir quais ações deverão ser tomadas para o correto funcionamento deste dispositivo;

Facilidade para o usuário final - Uma vez que o conector é padrão e o SO se responsabiliza pela configuração, o usuário fica isento destas responsabilidades;

Fornecimento de Energia - A USB 2.0 funciona em 5V e é capaz de alimentar dispositivos, desde que o consumo total de corrente dos dispositivos conectados a um mesmo barramento não ultrapasse 500mA.

Atualmente a interface USB está presente em todos os computadores e notebooks, tendo substituído as portas serial, paralela e PS/2. As aplicações legadas que ainda dependem dessas portas tipicamente utilizam conversores para serem conectadas à porta USB.

O AT89C5131A-M possui um controlador USB 2.0 *Full Speed* capaz de operar apenas em modo dispositivo. Este controlador está disponível para ser programado pelo usuário para implementar diversas classes USB como *Human Interface Device (HID)*, *Communication Device Class (CDC)* e *Device Firmware Upgrade Class*. As duas primeiras são relevantes para aqueles que desejarem criar aplicações como controladores de jogos e outros dispositivos de interface humana e dispositivos de comunicação como portas seriais virtuais e modems. A classe *Device Firmware Upgrade Class* é importante para aqueles que buscam criar uma forma de atualização de *firmware* via USB.

Ainda, o AT89C5131A-M vem pré-gravado de fábrica com um *firmwarebootloader* que faz uso da classe *Device Firmware Upgrade Class* para gravação de códigos na memória flash através da interface USB.

2.2 MEMÓRIA FLASH E O BOOTLOADER

A memória flash é uma memória não volátil do tipo EEPROM que pode ser apagada e gravada eletricamente. O AT89C5131A-M conta com duas regiões de memória flash, sendo 32kB de memória para programa e 3kB para o programa *bootloader*. As principais características da memória flash do AT89C5131A-M são:

- 32kB de memória para programa, organizada em 256 páginas de 128 bytes;
- Programa *bootloader* em área reservada de 3kB, contendo também diversas funções para escrita da memória flash via software, possibilitando a atualização de *firmware* via In-System Programming (ISP);

- Geração interna das tensões necessárias para gravar e apagar a memória.

A ATMEL disponibiliza, pré-gravado de fábrica no AT89C5131A-M, uma série de funções para a gravação da memória flash que podem ser utilizadas por programas para atualizações de *firmware* sem a necessidade de utilização de circuitos específicos para gravação. Entretanto, essas funções são destinadas a usuários experientes e o seu uso incorreto pode levar ao apagamento completo da memória flash e a única maneira de poder voltar a utilizar o microcontrolador seria utilizando um circuito gravador externo.

Entretanto, a ATMEL também disponibiliza um programa *bootloader*, também pré-gravado de fábrica, que se comunica com um computador via porta USB e, utilizando o software FLIP também da ATMEL, permite a gravação de programas. Esta será a forma pela qual os programas deverão ser gravados no *kit*, permitindo assim o seu uso sem a necessidade de gravadores externos.

A descrição de todas as funcionalidades e do funcionamento do *bootloader* podem ser encontradas no *datasheet* do *bootloader* (ATMEL, 2007).

2.3 RESET

O *reset* do AT89C5131A-M, além de reiniciar o microcontrolador para o seu estado inicial padrão, tem uma grande importância pois é no momento do *reset* que algumas verificações internas são feitas para decidir se o microcontrolador deverá iniciar a execução do programa do usuário ou do programa *bootloader*.

A figura 3 apresenta o fluxo de verificações executadas após o *reset* para decidir qual programa será executado.

2.4 I/O

O AT89C5131A-M está disponível em encapsulamento PLCC de 52 pinos, dos quais 34 podem ser utilizados como I/O, sendo divididos em 4 *ports* de 8 bits (P0 ao P3) e um *port* de 2 bits (P4). Os *ports* são todos bidirecionais ou seja, podem ser tanto entrada como saída.

Os ports P1, P2 e P3 ainda contam com resistores internos de pull-up para manterem um nível lógico constante enquanto estiverem desconectados.

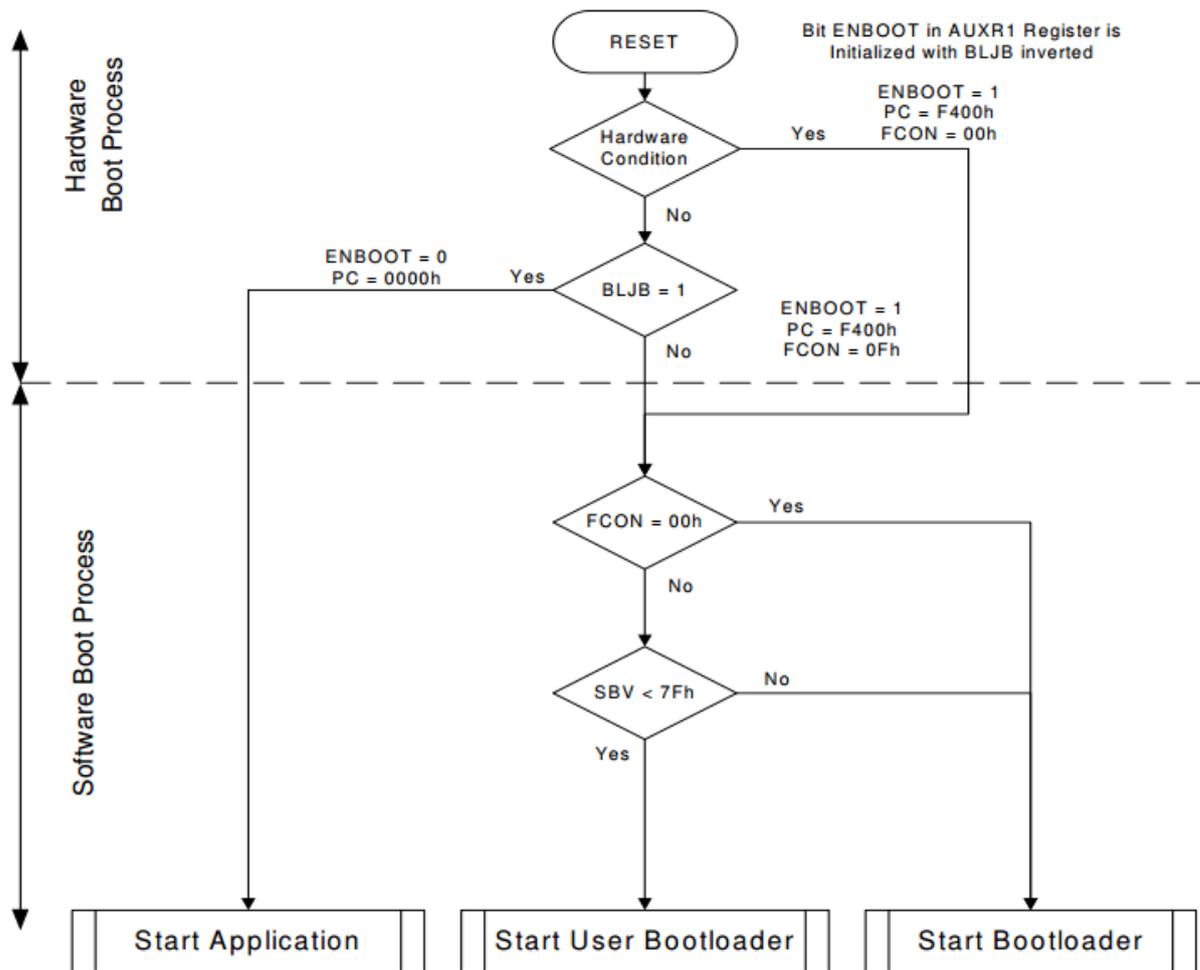


Figura 3: Fluxo de verificações após o reset.

Fonte: (ATMEL, 2007)

2.5 PROGRAMMABLE COUNTER ARRAY

O módulo *Programmable Counter Array* (PCA) do AT89C5131A-M é composto de um timer de 16 bits e cinco módulos de captura e comparação. A sua principal função é prover um mecanismo de temporização mais preciso e que possua menos dependência de intervenções da CPU do que os temporizadores convencionais, esse mecanismo pode ser utilizado para geração de sinais de modulação por largura de pulso (PWM).

2.6 DRIVER DE LEDS

O AT89C5131A-M conta com quatro drivers de LED de corrente programável em 4 níveis distintos, dispensando a necessidade de circuitos externos. Esses drivers são fontes de

corrente e portanto ajustam a tensão de acordo com a necessidade. A sua principal função é facilitar o uso de LEDs além de fornecer um mecanismo para alteração do nível de brilho.

2.7 SPI E I2C

Os protocolos SPI e I2C são protocolos seriais desenvolvidos para comunicação entre circuitos integrados. Em aplicações com microcontroladores, eles são tipicamente utilizados para a comunicação entre o microcontrolador e periféricos externos como conversores analógico-digitais (ADC), conversores digitais-analógicos (DAC), relógios de tempo real (RTC), sensores, memórias, entre outros.

2.8 EUART/RS232

A porta *Enhanced universal asynchronous receiver/transmitter* (EUART) do AT89C5131A-M possui todas as funcionalidades da porta *Universal asynchronous receiver/transmitter*(UART) padrão do 8051, além dos seguintes avanços:

- Capacidade de transmitir e receber dados simultaneamente em taxas distintas;
- Detecção automática da taxa de recepção;
- Detecção de erro de enquadramento dos dados;
- Detecção automática de endereço, quando o endereçamento for utilizado.

A principal função desta porta é, quando ligada a um conversor de nível, prover comunicação compatível com interfaces RS232 de computadores ou de outros dispositivos e com isso fornecer um canal de comunicação serial com o microcontrolador.

3 *KIT P51USB*

O *kit* P51USB é baseado no microcontrolador AT89C5131A-M e, uma vez que seu foco é didático, possui os componentes necessários para a utilização de suas interface de I/O, barramentos seriais e barramento para interface com display LCD 2x16. O *kit* completo pode ser visto na figura 4.

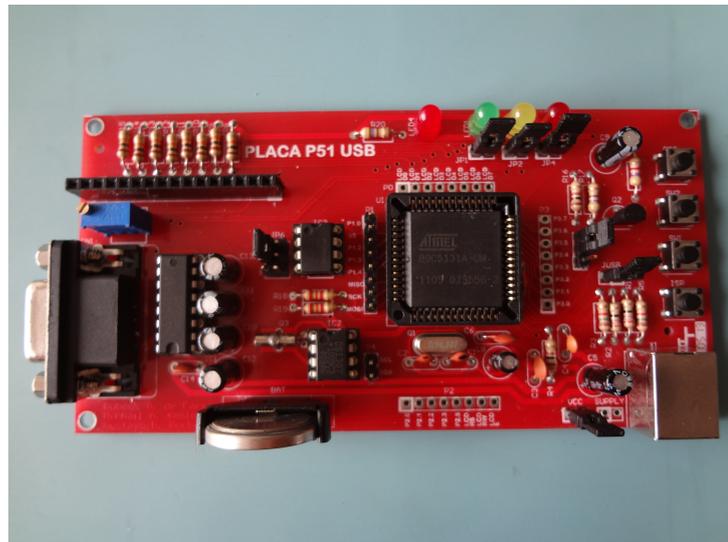


Figura 4: Fotografia do *kit* P51USB completo.

Fonte: Autoria própria.

A figura 5 apresenta o diagrama em blocos do hardware que compõem o *kit*. As seções a seguir descrevem os blocos do diagrama.

Além do AT89C5131A-M, o *kit* ainda contém um conversor de nível MAX232 para a interface RS232, 3 LEDs e 2 botões disponíveis para o usuário, um barramento para interface de display LCD 2x16 e ainda dois periféricos seriais escolhidos de acordo com a disponibilidade e preço no mercado local, além das suas funcionalidades, sendo eles um ADC, ADC0832, e um RTC, DS1307.

Os esquemáticos e layouts estão disponíveis nos apêndices A e B respectivamente.

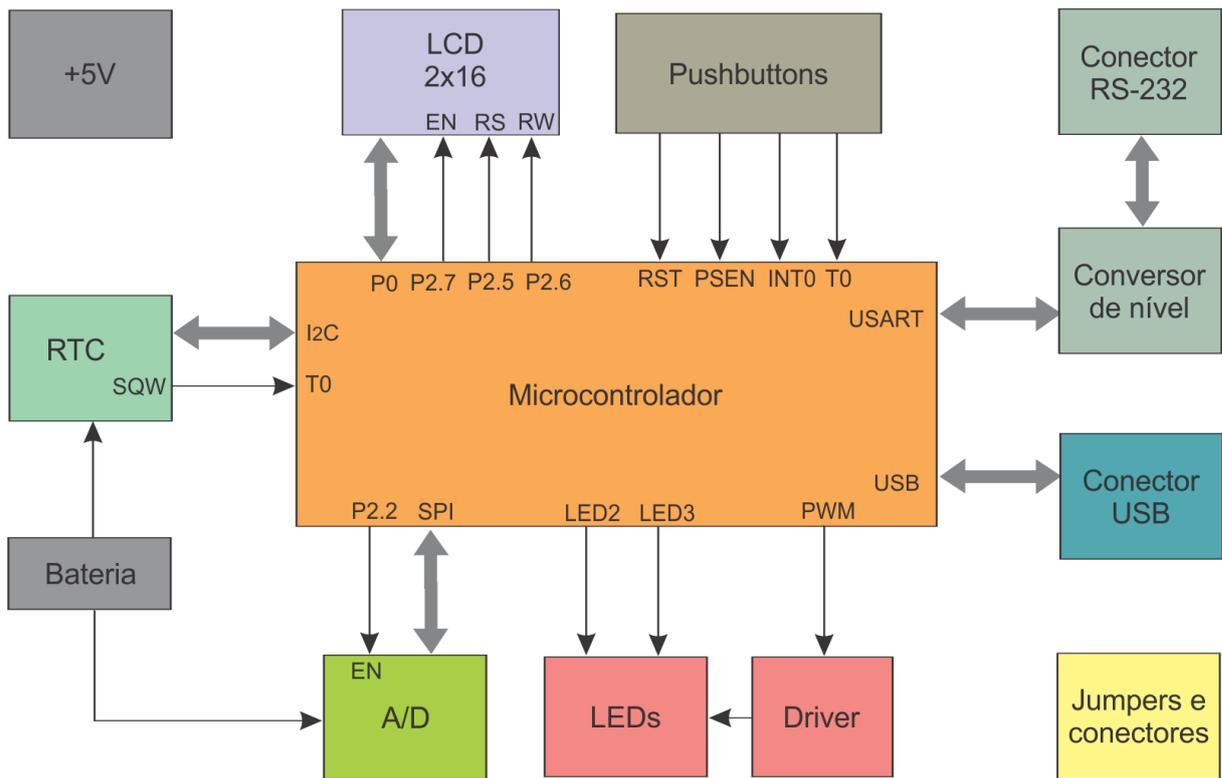


Figura 5: Diagrama em blocos do sistema.

Fonte: Autoria própria.

3.1 ADC0832

O ADC0832 é um conversor analógico digital de 8 bits, mesma largura do barramento de dados do microcontrolador utilizado, e poderá ser utilizado em experimentos envolvendo aquisição de sinais.

3.2 DS1307

O DS1307 é um relógio de tempo real que possui as seguintes funções:

- Relógio com contagem de segundos, minutos, horas, dia do mês, mês, dia da semana e ano com ano bissexto até 2100;
- 56 bytes de RAM de uso geral;
- Interface I2C;

- Consumo de corrente menor que 500nA quando não estiver em operação de comunicação, dependendo apenas de uma bateria;
- Gerador de uma onda quadrada de frequência programável em 1Hz, 4kHz, 8kHz ou 32kHz.

Ele poderá ser utilizado em uma série de experimentos que envolvam a necessidade de uma base de tempo externa e precisa, ou naqueles que envolvam saber o tempo absoluto de um determinado evento por exemplo.

3.3 BARRAMENTO PARA DISPLAY LCD 2X16

O *kit* P51USB conta com um barramento pronto para a conexão de displays LCD 2x16. Esses displays são facilmente encontrados no mercado local, além de terem uma operação simples. O barramento é composto de 16 pinos, dos quais 8 são utilizados para dados, 3 para sinais de controle, 2 para a luz de fundo, 1 para ajuste de contraste e 2 para alimentação, conforme o quadro 2.

Pino	Nome	Função
1	Led+	Luz de fundo
2	Led-	Luz de fundo
3	GND	Terra
4	VCC	Alimentação
5	VREF	Referência para ajuste de contraste
6	RS	Modo dados ou comandos
7	RW	Operação de escrita ou leitura
8	E	Envia os dados quando pulsado
9	DB0	Barramento de dados
10	DB1	
11	DB2	
12	DB3	
13	DB4	
14	DB5	
15	DB6	
16	DB7	

Quadro 2: Descrição dos pinos do barramento para display LCD.

Fonte: Autoria própria.

3.4 FLASHMON51

O FlashMon51 (KEIL, 2014) é um monitor par depuração de sistemas microcontrolados compatíveis com o 8051. Ele utiliza a porta serial do computador para estabelecer a conexão entre o software IDE Keil uVision e o microcontrolador. Para o caso do AT89C5131A-M e outros microcontroladores que contêm memória flash, o FlashMon51 já possui acesso às funções necessárias para gravar o programa do usuário na memória flash.

Para o *kit* P51USB, uma pequena modificação no código fonte do FlashMon51 foi necessária a fim de prover um mecanismo onde o usuário possa escolher entre os modos *bootloader*, “Debug” ou executar diretamente o seu programa.

O acesso a estes modos é feito através das chaves “RST”, “ISP” e “SW1” presentes no *kit*. A figura 6 apresenta a maneira correta de operação das chaves para acessar cada um dos três modos.

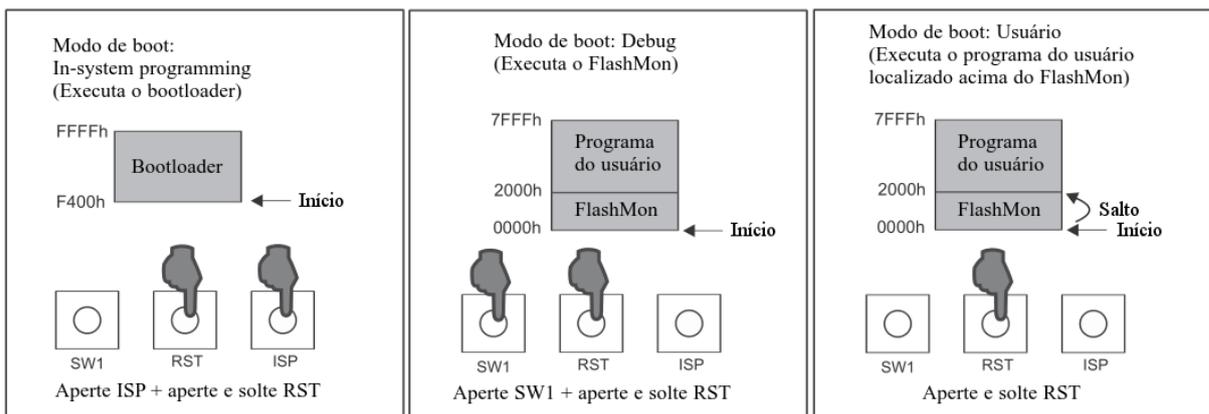


Figura 6: Utilização das chaves para acessar os três modos de operação do *kit*.

Fonte: Autoria própria.

3.5 BIBLIOTECA DE ROTINAS ASSEMBLY

As seções a seguir apresentam a descrição dos códigos em linguagem *assembly* que podem ser utilizados para explorar algumas funcionalidades do *kit* P51USB, além de um arquivo *header* que contém definições referentes ao *kit* P51USB. Os códigos estão disponíveis na forma de projetos do uVision em (KOSLOWSKI, 2014).

3.5.1 *HEADER* PADRÃO

Define nomes para alguns pinos da placa, para facilitar a leitura do código fonte e o seu entendimento.

3.5.2 LCD

Implementa as seguintes funções:

- `lcd_ini` - Faz a configuração inicial do display, deixando-o preparado para receber os dados a serem escritos;
- `lcd_busy` - verifica o flag que indica se o display está livre ou ocupado para receber um novo comando ou dado;
- `lcd_inst` - envia uma instrução ao display;
- `lcd_dado` - envia um dado ao display;
- `lcd_string` - envia uma *string* terminado em '0' ao display.

3.5.3 ADC VIA SPI

Implementa as seguintes funções:

- `ini` - Faz a configuração do periférico SPI para utilização com o ADC0832;
- `ADC_SAMPLE` - Função que faz uso do periférico SPI em modo *polling* para obter as leituras do ADC.

Existe também o arquivo `spi_adc.h` que define os nomes para algumas configurações alternativas que podem ser utilizadas com o ADC0832 como por exemplo o modo de leitura diferencial.

3.5.4 RTC VIA I2C

Implementa as seguintes funções:

- `init`] - Inicializa o controlador I2C e configura o relógio;

- `RTC_SET_TIME` - Utiliza as funções da I2C para configurar os registradores do RTC;
- `RTC_GET_TIME` - Lê os valores dos registradores do RTC;
- `i2c_int` - Função que implementa a rotina de atendimento à interrupção da I2C. Chama diversas sub-rotinas, uma para cada estado do controlador;
- `decode` - Função que opera o *Program Counter* de forma a ‘enganar’ o microcontrolador, fazendo-o retornar no ponto de entrada da sub-rotina de atendimento da interrupção.

As demais funções implementadas são sub-rotinas de atendimento a interrupção, sendo o mínimo necessário para que o dispositivo opere em modo mestre. Para operações em modo escravo ou para um tratamento diferenciado em caso de erros, algumas sub-rotinas deverão ser alteradas de acordo e outras deverão ser implementadas.

3.5.5 OUTROS EXEMPLOS

Outros exemplos podem ser encontrados no site da ATMEL, uma vez que eles disponibilizam exemplos genéricos de uso de todos os periféricos para todos os seus microcontroladores.

4 OPERAÇÃO DO KIT

A operação do *kit* será dividida em duas etapas, gravação e depuração, sendo cada uma delas descrita por um exemplo de utilização que pode servir como um guia a ser consultado por quem for utilizar o *kit*.

Os exemplos a seguir estão disponíveis também em (KOSLOWSKI, 2014) na forma de tutoriais.

4.1 GRAVAÇÃO

O processo de gravação é apresentado através de um exemplo introdutório *hello world*.

4.1.1 HELLO WORLD

Um exemplo introdutório, *hello world* (olá mundo), é tipicamente utilizado quando se vê pela primeira vez uma nova linguagem de programação ou plataforma. Em microcontroladores o *hello world* normalmente envolve a simples leitura de uma chave e o acender de um LED.

Neste exemplo, o programa fará a leitura da chave e de acordo com o resultado será feito um *toggle* (mudança de estado) de um LED, ou seja, cada vez que o botão for apertado o LED acende ou apaga.

4.1.2 REQUISITOS

- Hardware
 - Placa P51USB;
 - Cabo USB-B ou USB-miniB (de acordo com aquele montado na placa).
- Software

- Keil uVision 4 C51 versão de avaliação;
- Atmel FLIP 3.4.7.

A versão de avaliação do uVision possui todas as funcionalidades da versão completa porém é limitada em 4k de código por projeto.

O Atmel FLIP deverá ser instalado preferencialmente com o *Java Runtime Enviroment*(JRE) incluído.

4.1.3 CRIANDO UM PROGRAMA COM O UVISION

4.1.3.1 CRIANDO UM PROJETO NO UVISION

Ao iniciar o uVision o usuário será apresentado a uma tela semelhante a da figura 7.

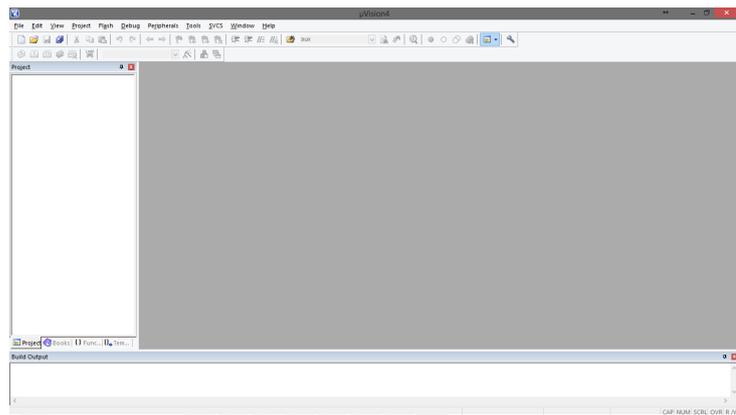


Figura 7: Tela inicial do Keil uVision 4.

Fonte: Autoria própria.

Para criar um projeto, deve-se utilizar o item “Project → New uVision Project...” e, quando solicitado, escolher um nome e um local para salvar o projeto conforme a figura 8.

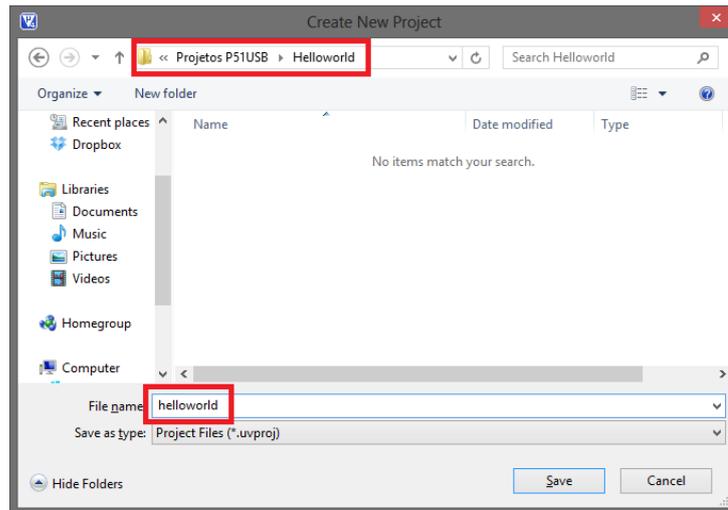


Figura 8: Janela para escolha do nome e local de armazenamento do projeto do uVision.

Fonte: Autoria própria.

A seguir, será necessário escolher o microcontrolador, conforme a figura 9

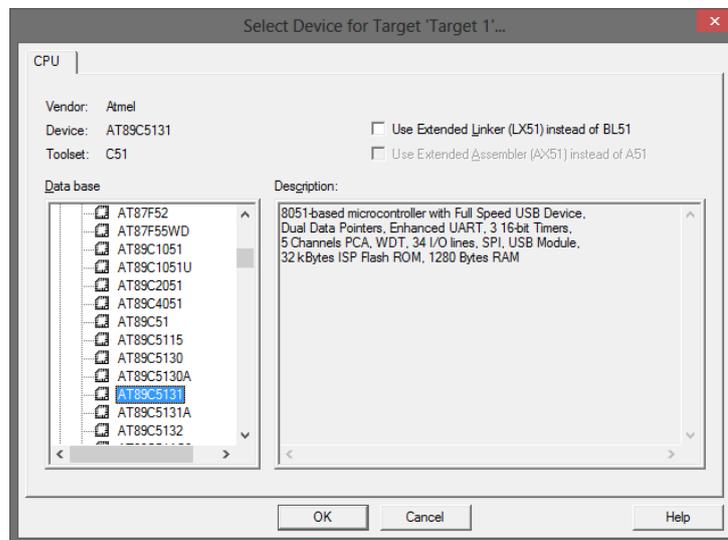


Figura 9: Tela para escolha do microcontrolador.

Fonte: Autoria própria.

Por fim, o usuário será questionado se deseja que seja incluído no projeto o arquivo “Startup.a51”, devendo selecionar a opção para não incluí-lo, conforme a figura 10.

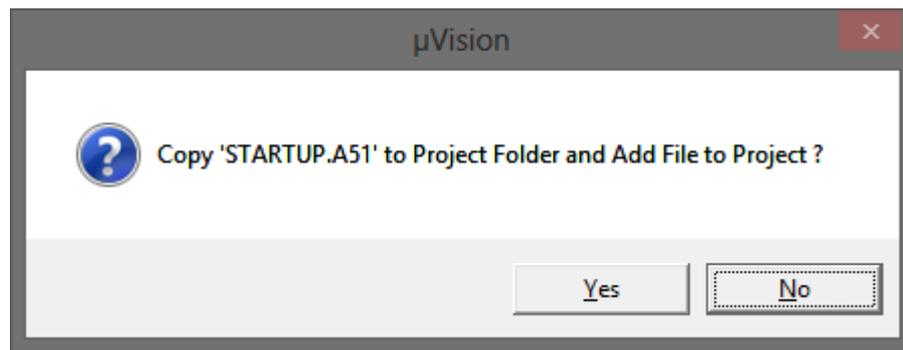


Figura 10: Tela para escolha do microcontrolador.

Fonte: Autoria própria.

4.1.3.2 CONFIGURANDO O PROJETO

O projeto criado anteriormente deverá ser configurado para gerar um arquivo do tipo Intel HEX como saída e também para não incluir os símbolos padrões do 8051 pois isso será feito no código por meio de um arquivo de cabeçalho.

Com o botão direito do mouse, o usuário terá acesso ao menu para a escolha da opção “Options for Target ‘Target 1’...” como na figura 11.

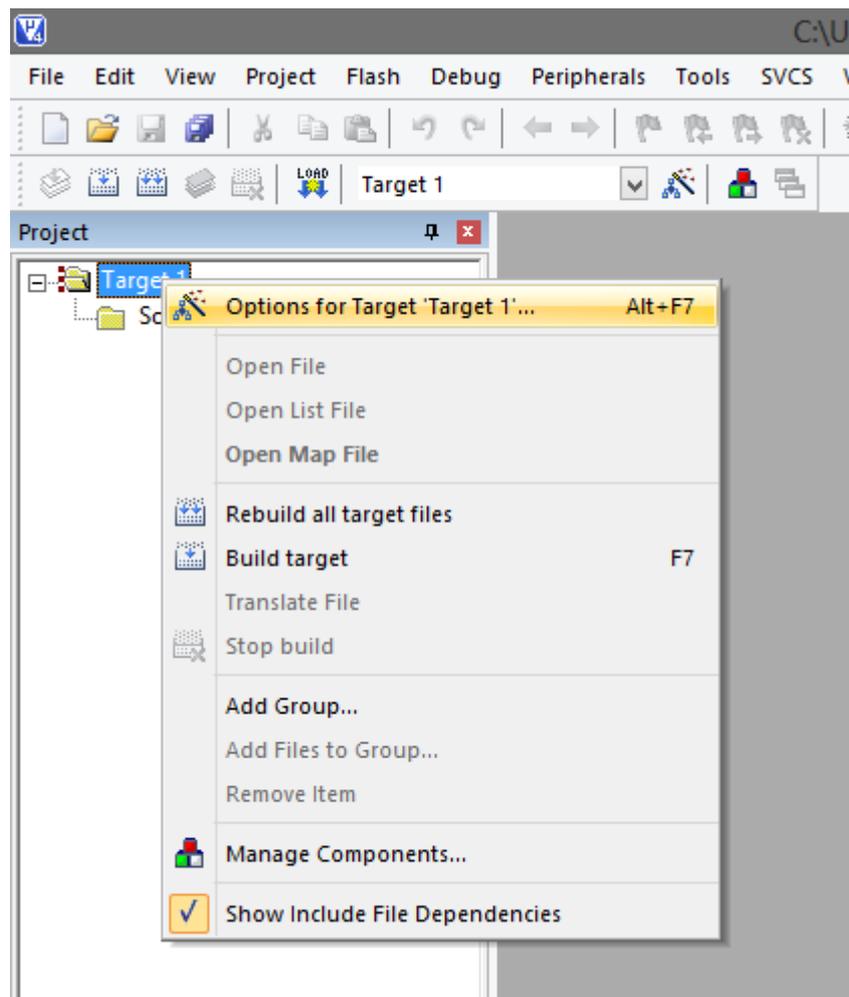


Figura 11: Menu para acessar as configurações do projeto.

Fonte: Autoria própria.

A primeira configuração deverá ser feita na aba “Output”, selecionando a opção “Create HEX File”, como destacado na figura 12

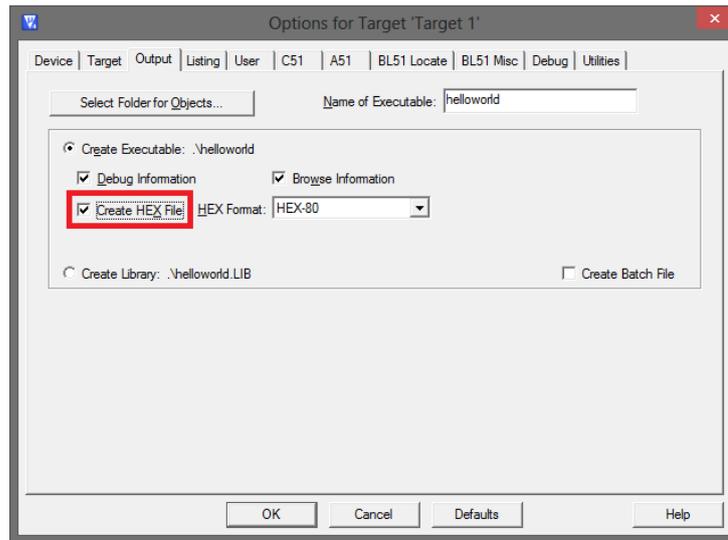


Figura 12: Configuração para criação de arquivo Intel HEX.

Fonte: Autoria própria.

A segunda configuração deverá ser feita na aba "A51", desmarcando a opção "Define 8051 SFR Names", como destacado na figura 13

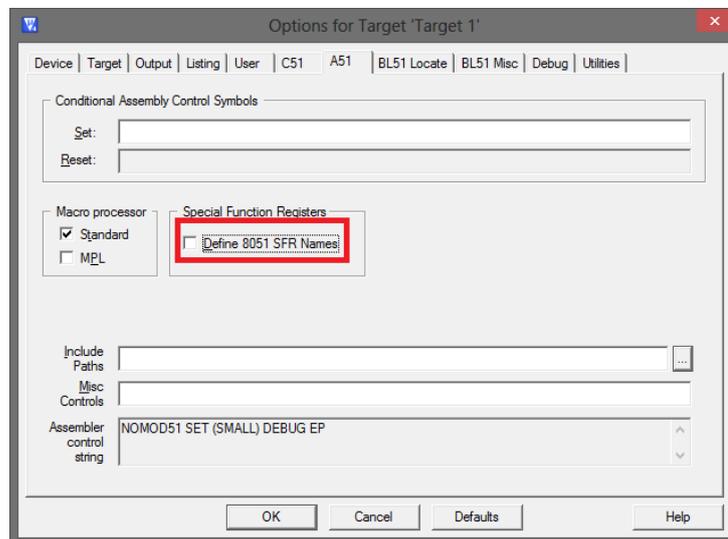


Figura 13: Configuração para a não inclusão automática dos símbolos do 8051.

Fonte: Autoria própria.

4.1.3.3 ADICIONANDO O ARQUIVO COM O CÓDIGO FONTE

Um novo arquivo deverá ser criado, utilizando o botão “New” ou o atalho “Ctrl+N”, como na figura 14

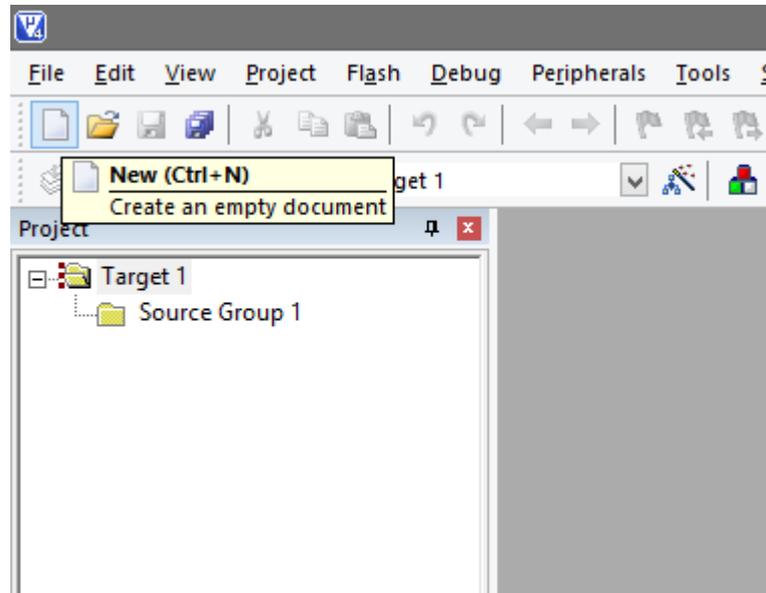


Figura 14: Criação de um novo arquivo.

Fonte: A autoria própria.

O seguinte código fonte será utilizado neste exemplo:

```
;Placa P51USB v1.0
;Programa de teste
;-----
;Chave SW1(P3.2) faz toggle do Led3(P1.4)

#include "at89c5131.h"

; Definições da P51USB
LED3 EQU P1.4
SW1 EQU P3.2

ORG 0x00
LJMP main
```

```

main:
    JNB SW1, toggle
    SJMP main
toggle:
    CPL LED3
    SJMP main
END

```

O arquivo deverá ser salvo na mesma pasta do projeto, com a extensão “.a51”, como na figura 15.

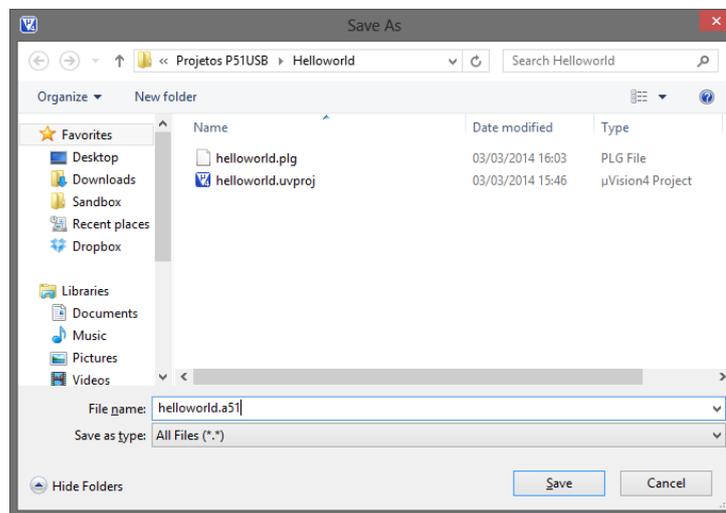


Figura 15: Janela com as opções para salvar arquivo.

Fonte: Autoria própria.

Por fim, o arquivo deverá ser adicionado ao projeto utilizando-se a opção “Add Files to Group ‘Source Group 1’...” acessível através do menu conforme a figura 16.

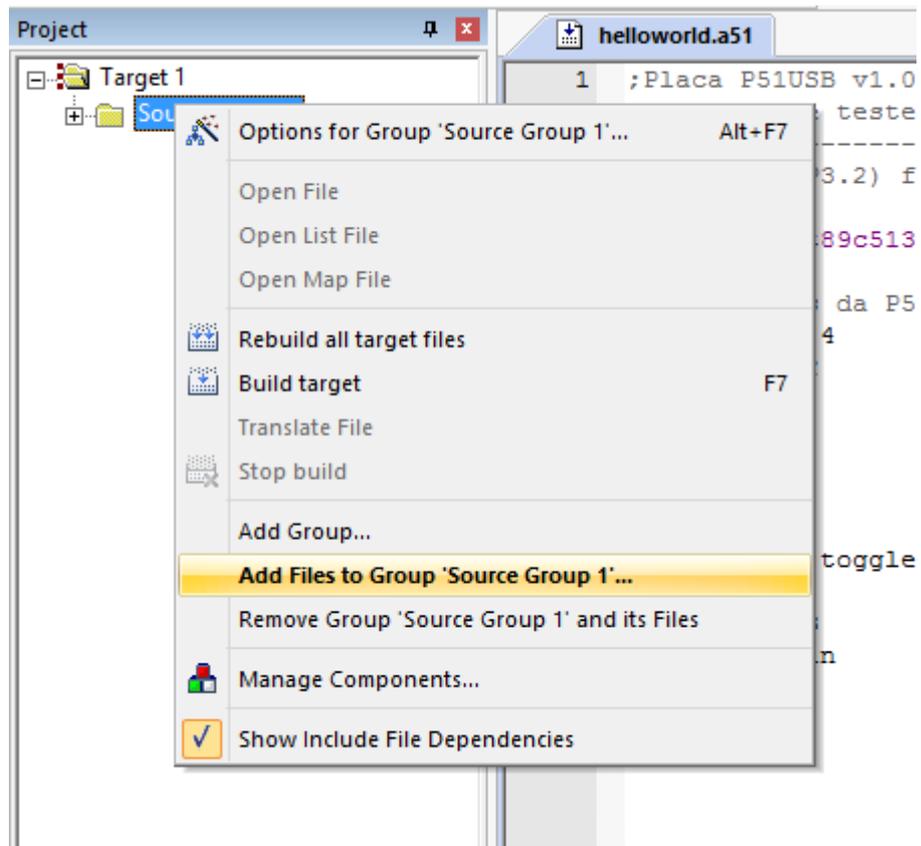


Figura 16: Menu para adicionar arquivos ao projeto.

Fonte: Autoria própria.

O tipo de arquivo deverá ser escolhido “Asm Source file”, e após a seleção do arquivo o usuário deverá pressionar os botões “Add” e “Close”, nesta ordem, conforme a figura 17.

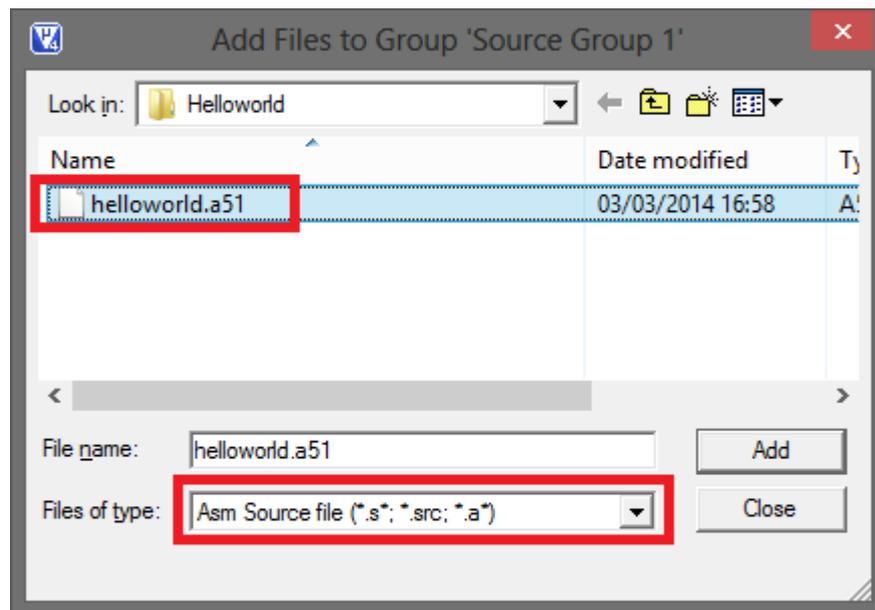


Figura 17: Opções para assembler e linkar o projeto.

Fonte: Autoria própria.

4.1.3.4 MONTANDO E LIGANDO

Para montar e ligar o projeto, pode-se utilizar o botão “Build”, ou o item de menu “Build target” ou a tecla de atalho “F7”, conforme a figura 18.

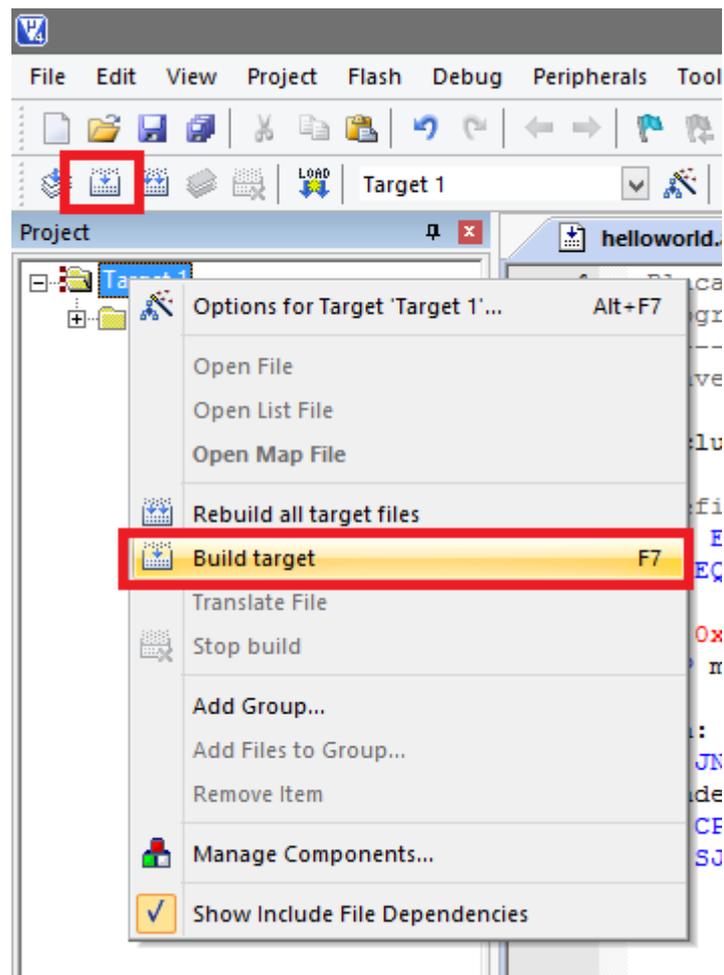


Figura 18: Opções para montar e ligar o projeto.

Fonte: Autoria própria.

Não ocorrendo erros, a saída do programa deverá ser semelhante a figura 19.

```

Build Output
Build target 'Target 1'
assembling helloworld.a51...
linking...
Program Size: data=8.0 xdata=0 code=10
creating hex file from "helloworld"...
"helloworld" - 0 Error(s), 0 Warning(s).
  
```

Figura 19: Saída do programa em caso de sucesso.

Fonte: Autoria própria.

4.1.4 GRAVANDO O PROGRAMA COM O FLIP

4.1.4.1 COLOCANDO A PLACA EM MODO *BOOTLOADER*

O modo *bootloader* do microcontrolador permite a gravação de um *firmware* em formato Intel HEX a partir do FLIP utilizando a interface USB.

De acordo com o *datasheet* do AT89C5131A-M, para entrar em modo *bootloader* é necessário estar com o pino PSEN ligado, por meio de um resistor, ao GND durante um *reset*. A P51USB já possui uma chave para este fim, chamada ISP.

Com a P51USB já conectada ao computador host, execute as seguintes etapas:

1. Remova o *jumper* J_USB
2. Aperte e segure a chave ISP
3. Aperte momentaneamente a chave RST
4. Solte a chave ISP
5. Conecte o *jumper* J_USB

Note que na primeira conexão com o computador host poderá haver a necessidade de instalação dos drivers, que podem ser encontrados na pasta onde está instalado o FLIP.

Para certificar-se que a placa encontra-se em modo *bootloader* verifique os seguintes itens:

- O único LED aceso na placa é o LED4, indicando que a mesma está alimentada.
- O *jumper* J_USB permanece conectado durante toda a operação do *bootloader*. Caso ele seja removido, mesmo que acidentalmente, o processo deverá ser reiniciado.
- No Windows, verifique se o gerenciador de dispositivos mostra corretamente o dispositivo, conforme a figura 20.
- No Linux, verifique a saída dos comandos `lsusb` e `dmesg`.



Figura 20: Exemplo de dispositivo corretamente enumerado no gerenciador de dispositivos do Windows.

Fonte: Autoria própria.

4.1.4.2 CONECTANDO COM O FLIP

Durante a primeira utilização do FLIP, devemos selecionar o chip com o qual desejamos nos comunicar. No caso da P51USB o chip é o AT89C5131A-M.

Na tela inicial do FLIP esta função pode ser localizada conforme a figura 21.

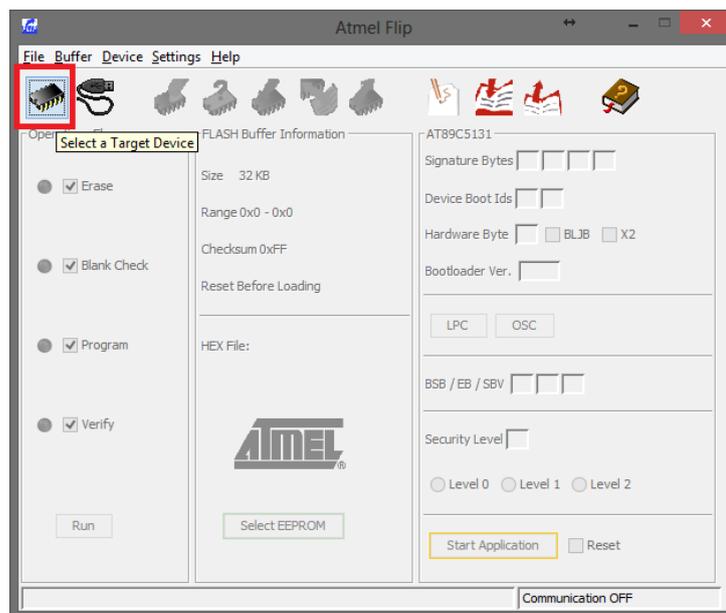


Figura 21: Tela inicial do FLIP, com destaque para a seleção do microcontrolador.

Fonte: Autoria própria.

Na sequência, o microcontrolador deverá ser escolhido conforme a figura 22.

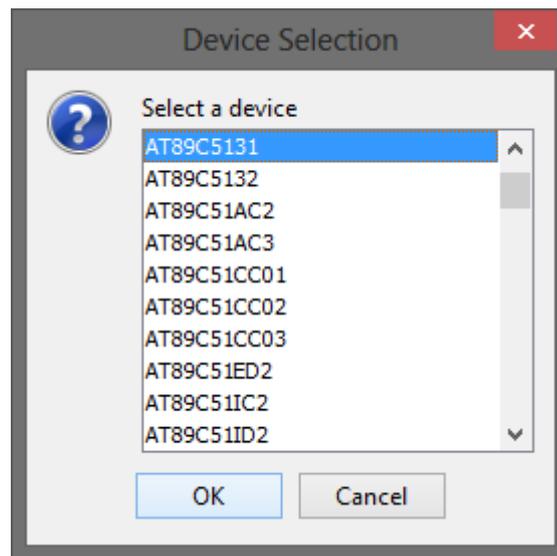


Figura 22: Tela para a seleção do microcontrolador.

Fonte: Autoria própria.

O próximo passo é a seleção do meio de comunicação, neste caso USB, conforme a figura 23.

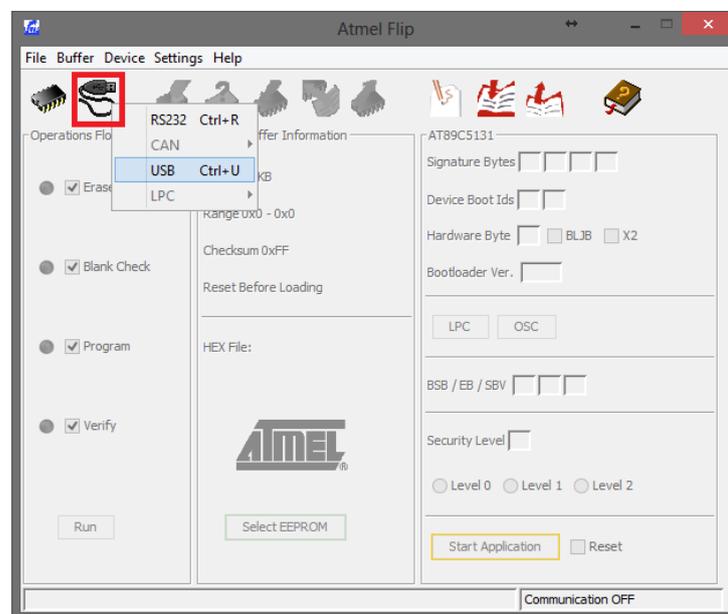


Figura 23: Seleção do meio de comunicação.

Fonte: Autoria própria.

Na janela apresentada na figura 24 será possível abrir ou fechar a conexão com a placa.

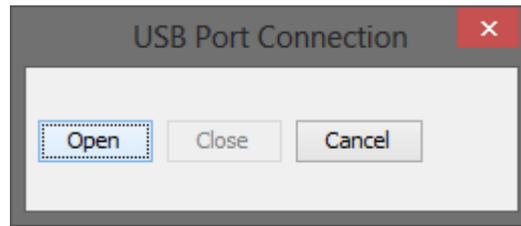


Figura 24: Tela da conexão USB.

Fonte: Autoria própria.

Caso a conexão não seja efetuada com sucesso, os seguintes itens deverão ser verificados:

- Desconecte a placa do computador, reconecte e tente novamente o procedimento;
- Verifique os cabos, dando preferência para a utilização de cabos blindados pois estes apresentam maior imunidade ao ruído;
- Utilize uma outra porta USB no computador host, pois algumas portas possuem maior capacidade de corrente e tensões mais estáveis do que outras.

4.1.4.3 CARREGANDO O ARQUIVO INTEL HEX

Após o estabelecimento da conexão com a placa, deve-se carregar o arquivo Intel HEX gerado anteriormente para que o mesmo seja posteriormente gravado. Para isso, utiliza-se um botão do FLIP conforme o destaque na figura 25.

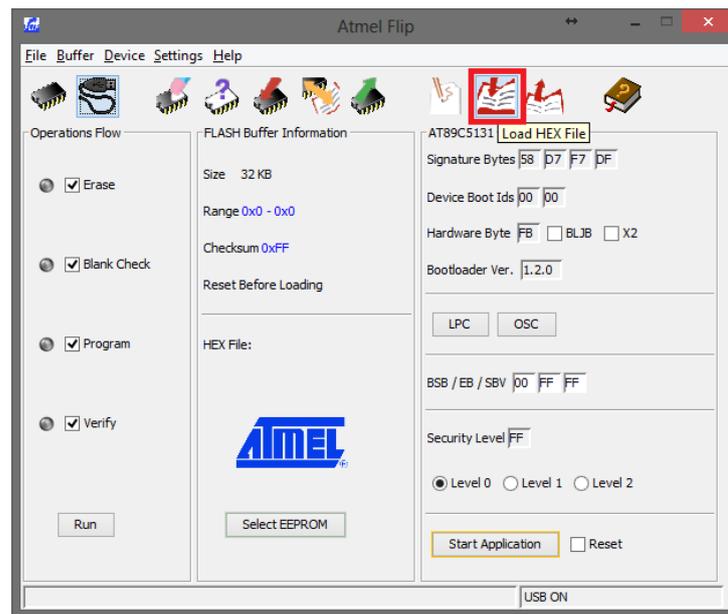


Figura 25: Item do menu para o carregamento do arquivo Intel HEX.

Fonte: Autoria própria.

Após a seleção e carga do arquivo, se não ocorrerem erros, algumas informações como *checksum* e o tamanho do programa serão exibidas conforme a figura 26

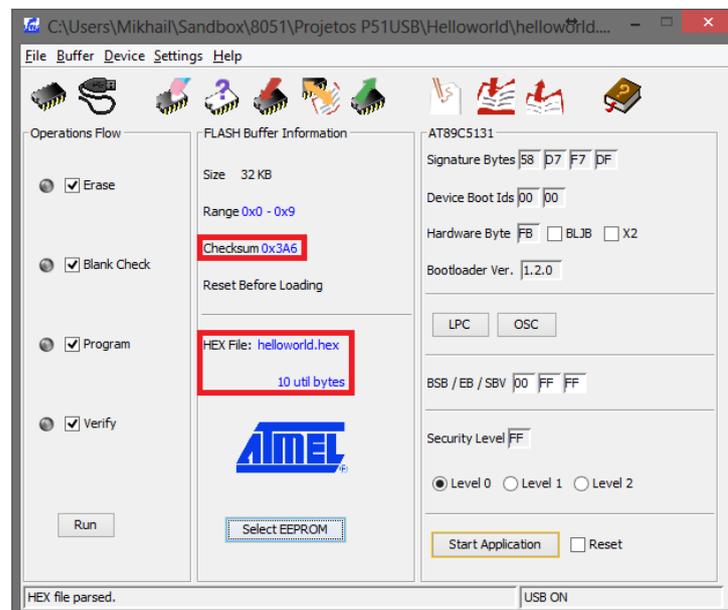


Figura 26: Informações sobre o programa carregado.

Fonte: Autoria própria.

Caso o arquivo hex não seja carregado corretamente, verifique os seguintes itens:

- O caminho até o arquivo não deve conter acentuações. Um bom teste é colocar o arquivo Intel HEX no 'C:' e tentar carregá-lo por lá.
- O arquivo Intel HEX pode estar corrompido. Neste caso, gera-lo novamente deverá solucionar o problema.

4.1.4.4 GRAVANDO

Para iniciar o processo de gravação, basta certificar-se que todos os *checkboxes* estão habilitados conforme a figura 27 e iniciar o processo com o botão 'Run'.

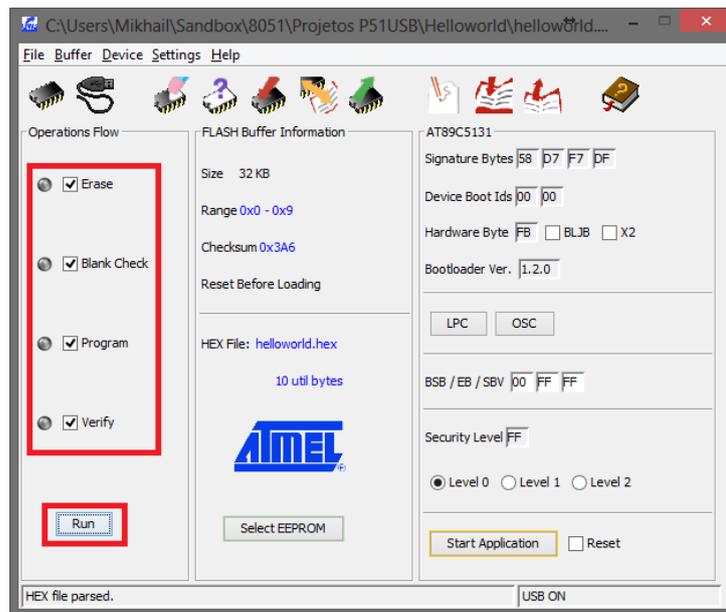


Figura 27: Início do processo de gravação.

Fonte: Autoria própria.

Se a gravação ocorrer normalmente, os ícones que representam cada etapa serão alterados para cor verde, indicando o sucesso.

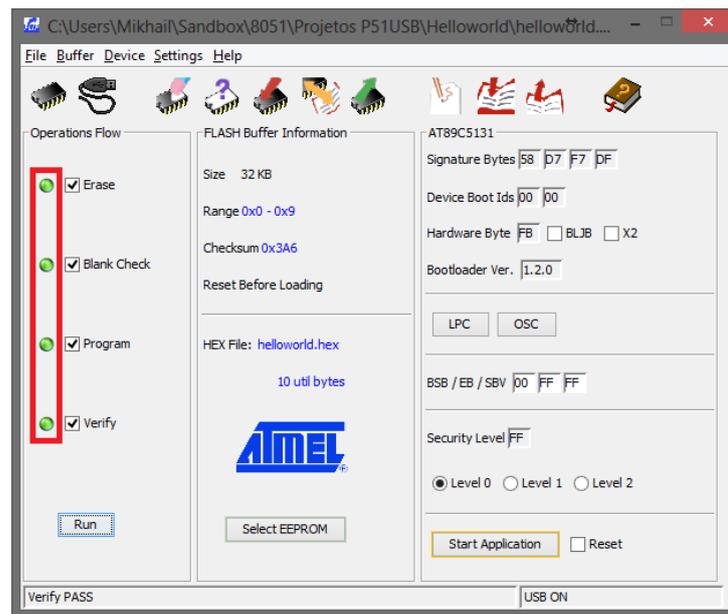


Figura 28: Fim do processo de gravação.

Fonte: Autoria própria.

Caso alguma das etapas não execute com sucesso, recomenda-se:

- Verificar todas as conexões entre a placa e o PC;
- Repetir individualmente cada uma das etapas nesta ordem: *Erase*, *Blank Check*, *Program*, *Verify*.

4.1.5 TESTANDO O PROGRAMA

Para testar o programa, desconecte o *jumper* 'J_USB' e reinicie a placa apertando momentaneamente a chave 'RST'.

Cada vez que a chave 'SW2' for pressionada, o 'LED3' deverá trocar de estado. O estado aceso é ilustrado na figura 29

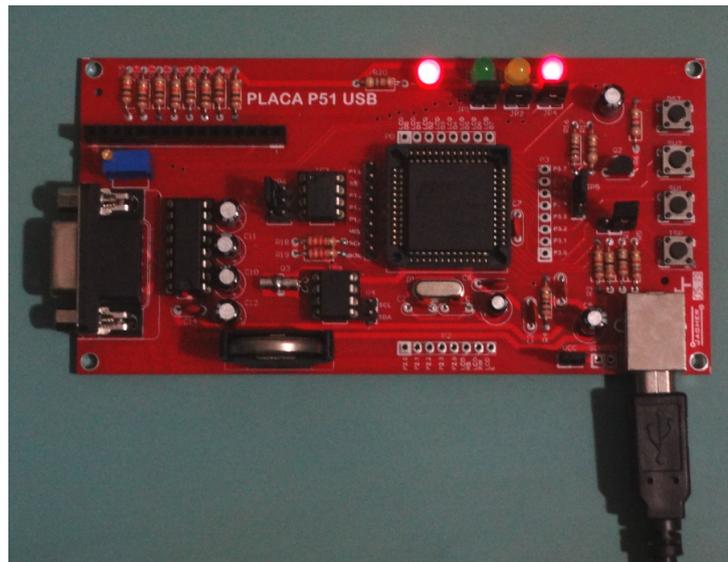


Figura 29: Placa executando o programa *hello world*.

Fonte: Autoria própria.

4.2 DEPURAÇÃO

4.2.1 GRAVAÇÃO DO FLASHMON51

Inicialmente, é necessário gravar o FlashMon51 na memória do microcontrolador. Este procedimento deverá ser feito obtendo-se o arquivo Intel HEX do FlashMon51 para o AT89C5131A-M em (KOSLOWSKI, 2014) e gravando-o utilizando o FLIP conforme procedimento descrito na seção 4.1.4.

4.2.2 CONFIGURAÇÃO DO UVISION

Posteriormente, é necessário efetuar a configuração do uVision para utilizar o FlashMon51. Nas opções do projeto, na aba 'Debug' deverá ser selecionado o item 'Use Keil Monitor-51 Driver', conforme a figura 30.

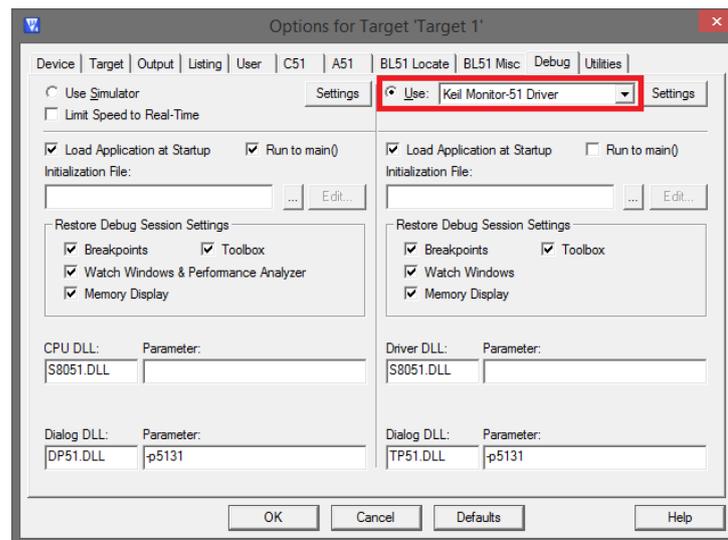


Figura 30: Configuração para utilização do FlashMon51.

Fonte: Autoria própria.

A seguir, será necessário escolher a porta serial, conforme a figura 31.

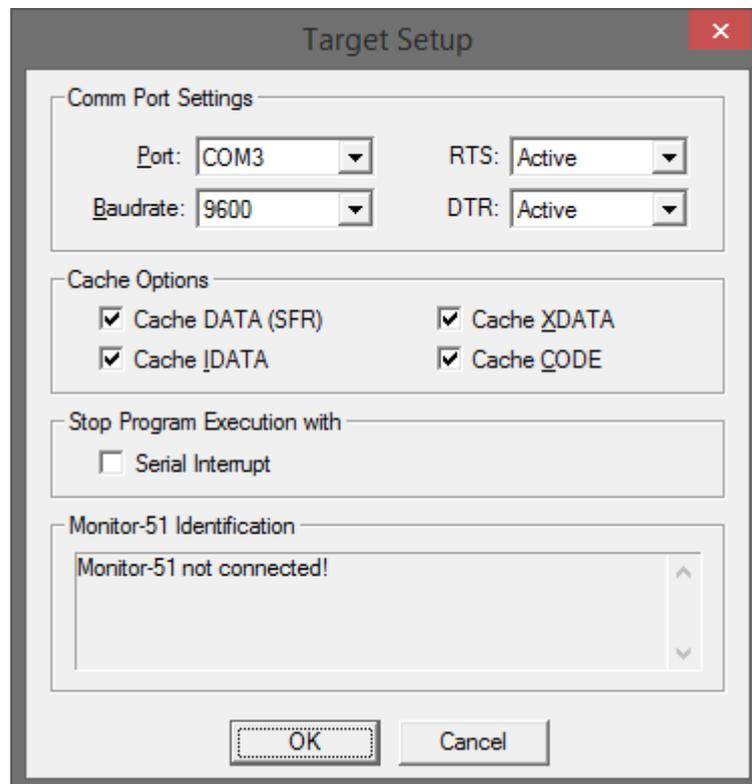


Figura 31: Escolha dos parâmetros da porta serial.

Fonte: Autoria própria.

Esta escolha deverá ser feita de forma que a porta utilizada seja aquela listada no gerenciador de dispositivos do sistema, como o exemplo da figura 32.

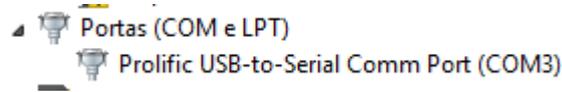


Figura 32: Detalhe da porta serial no gerenciador de dispositivos do Windows.

Fonte: Autoria própria.

4.2.3 COLOCAR A PLACA EM MODO *DEBUG*

Para que o programa seja carregado corretamente pelo FlashMon51, inicialmente é necessário criar o arquivo Intel HEX deslocado de 0x2000 bytes, uma vez que o próprio FlashMon51 reside nos endereços mais baixos. Após alterar o programa, basta monta-lo e liga-lo novamente.

Para iniciar o processo de depuração, a placa deve ser colocada em modo *bootloader* seguindo as seguintes etapas:

1. Ligue a placa e conecte o cabo serial;
2. Aperte e segure a chave 'SW1';
3. Aperte momentaneamente a chave 'RST';
4. Solte a chave 'SW1'

Como indicativo de que o modo *debug* está ativado, o LED 'LED2' ficará aceso, como na figura 33. (Note que o LED da figura é amarelo, e por ter brilho baixo não aparenta estar aceso.)

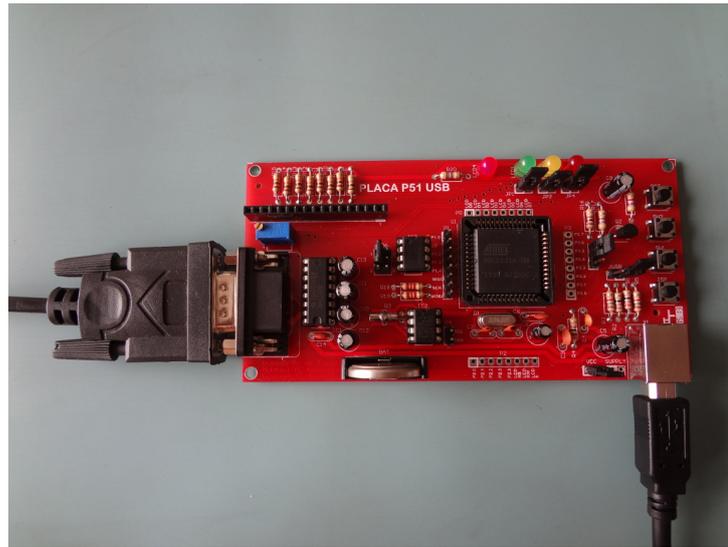


Figura 33: P51USB em modo *debug*.

Fonte: Autoria própria.

4.2.4 CARGA DO PROGRAMA VIA FLASHMON51

Uma vez que a placa estiver em modo *debug*, para efetuar a carga do programa, basta iniciar a sessão de *debug* através do item de menu ‘Start Debug Session’ conforme a figura 34

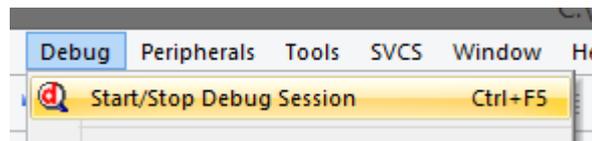


Figura 34: Opção do menu para iniciar sessão de download.

Fonte: Autoria própria.

A figura 35 apresenta um computador e uma placa P51USB durante o processo de depuração.

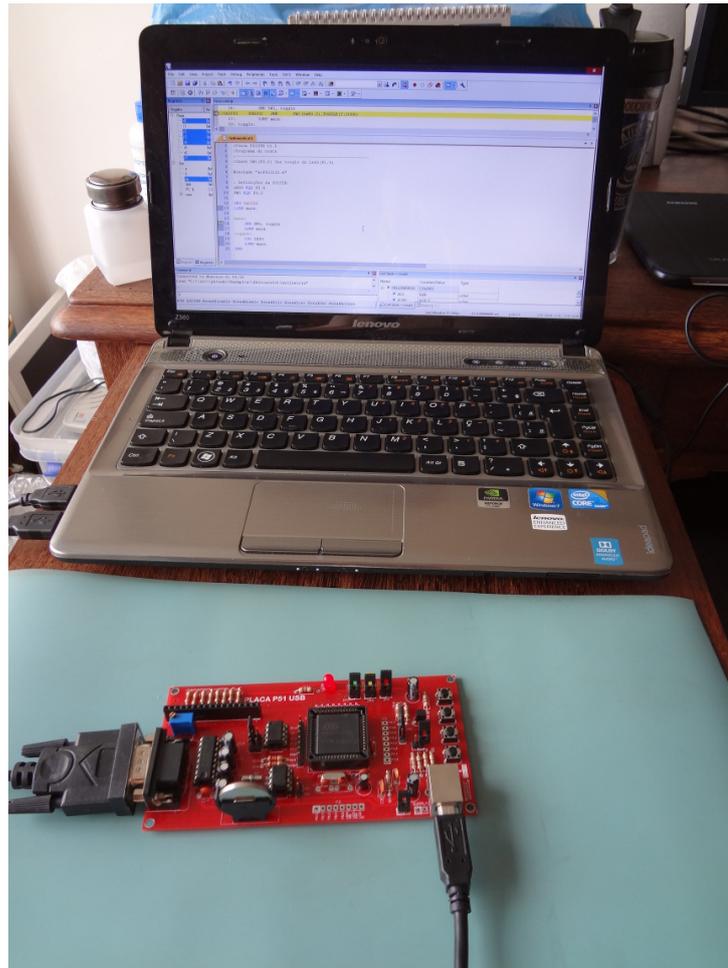


Figura 35: Computador e P51USB em processo de depuração.

Fonte: Autoria própria.

5 CONSIDERAÇÕES FINAIS

Os objetivos propostos inicialmente para o projeto foram atingidos, resultando na produção do *kit* P51USB, tecnologicamente atualizado e de fácil utilização. Além de sua documentação técnica, um site contendo toda a documentação, tutoriais de utilização e links para os softwares necessários foi produzido.

Quanto à possibilidade de utilização da porta USB para comunicação direta com o FlashMon51, sem a necessidade de utilização da porta serial e de um conversor USB, após diversas tentativas buscando várias abordagens diferentes para este problema, não foi possível executar simultaneamente o programa necessário para a comunicação USB, o FlashMon51 e o programa alvo de depuração. Caso o código fonte do FlashMon51 estivesse disponível, seria possível tentar ainda uma nova abordagem onde a comunicação USB e o FlashMon51 seriam um único programa e talvez fosse possível efetuar a depuração, mas com severas limitações temporais pois a USB demanda uma grande fatia do processamento e o FlashMon51, utilizando as interfaces de escrita e leitura de memória flash, também necessita de uma quantidade razoável de recursos.

Parte do resultado deste trabalho, a placa, a documentação técnica e um esboço dos tutoriais, foi utilizada de forma piloto na turma de Microcontroladores do segundo semestre de 2013 do curso de Engenharia Eletrônica do câmpus Curitiba da UTFPR. A utilização do *kit* P51USB por esta turma, bem como as contribuições dos professores que ministram a disciplina foram de fundamental importância para o desenvolvimento deste projeto, tendo um grande impacto no resultado final. A utilização deste *kit* pelas próximas turmas poderá contribuir para a identificação de melhorias a serem implementadas em versões futuras e para o desenvolvimento da qualidade de ensino.

6 PUBLICAÇÕES

- Borba, Gustavo B., Soviersovski, Miguel A., Koslowski, Mikhail A., Schwarz, Leandro. KIT DE DESENVOLVIMENTO PARA O ENSINO DE MICROCONTROLADORES MCS51 COM COMUNICAÇÃO USB. III Simpósio Nacional de Ensino de Ciência e Tecnologia. Ponta Grossa, PR, Brasil. Setembro de 2012.
- Koslowski, Mikhail A., Borba, Gustavo B., Faria, Rubens A. de, Goes, Rafael E. de. AN EASY-TO-USE DEVELOPMENT KIT FOR A MICROCONTROLLER COURSE BASED ON THE 8051. XIII Conferência Internacional em Educação em Engenharia e Tecnologia - INTERTECH'2014. Guimarães, Portugal. Março de 2014.

PLANO DE NEGÓCIO

7 PLANO DE NEGÓCIO

7.1 SUMÁRIO EXECUTIVO

O mercado de trabalho busca cada vez mais profissionais qualificados e principalmente com experiência prática. Neste cenário, alunos com uma formação apenas teórica acabam ficando em desvantagem com relação aqueles que possuem, além da formação teórica, um maior contato com a prática.

Para promover uma melhor experiência no contato dos alunos com a prática faz-se necessário em algumas disciplinas o desenvolvimento de ferramentas didáticas que reduzam a lacuna entre teoria e prática.

O *kit* P51USB foi desenvolvido para atender às demandas das disciplinas que utilizam microcontroladores da família MCS51 em suas atividades práticas. Ele possui interface USB para comunicação com o computador, não dependendo assim da interface RS232, tipicamente utilizada para este fim. A porta USB também é utilizada para acessar um sistema bootloader e efetuar a gravação do microcontrolador, não dependendo de gravadores externos.

O *kit* foi desenvolvido após estudo dos *kits* previamente existentes para que as suas principais características fossem mantidas e as suas principais desvantagens fossem contornadas.

Para o desenvolvimento e a produção do *kit*, um investimento inicial de R\$12.735,00 será necessário. O break-even é previsto para 3 anos e 4 meses após o investimento inicial e a taxa interna de retorno do projeto estimada em 114% no quinto ano. Para isso, o preço de venda estimado de cada *kit* é de cerca R\$ 87,00, corrigido anualmente pela inflação, estimada em 6% durante todo o período.

7.2 DEFINIÇÃO DO NEGÓCIO

7.2.1 VISÃO

Ser referência no desenvolvimento de ferramentas didáticas nas áreas tecnológicas.

7.2.2 MISSÃO

Desenvolver e entregar ferramentas didáticas acessíveis e eficazes para com isso promover o desenvolvimento tecnológico.

7.2.3 VALORES

- Ética e profissionalismo em todos os processos;
- Desenvolver e entregar produtos e serviços de qualidade;
- Estar sempre atualizado tecnologicamente;
- Manter um diálogo sempre aberto com o cliente.

7.2.4 DESCRIÇÃO DO NEGÓCIO

O mercado no qual o negócio está sendo proposto é o de educação tecnológica, na indústrias de educação e tecnologia, na área de ensino, pesquisa e desenvolvimento tecnológico. O objetivo do produto é promover uma maior eficácia no ensino tecnológico, proporcionando ao aluno (usuário) uma ferramenta acessível e que promova o aprendizado. A venda do produto será feita diretamente ao cliente, por meio de website próprio, ou através de distribuidores como sites especializados e até mesmo as próprias instituições de ensino.

A expectativa é que após a inserção inicial do produto no mercado, este alcance cada vez mais clientes, tornando-se assim uma referência nacional. A expansão se dará através de parcerias com professores e instituições de ensino para promover o *kit*.

O baixo custo do *kit* e as facilidades que ele provê fazem com que o processo de ensino e aprendizagem seja mais eficaz, uma vez que professor e aluno podem focar nos seus objetivos sem ter que se preocupar com questões secundárias como por exemplo ter que montar e manter atualizados seus próprios conjuntos de ferramentas de desenvolvimento, depender de outras ferramentas como gravadores. Essas características, entre outras, são os maiores diferenciais com relação aos concorrentes e são fundamentais para atingir o reconhecimento no mercado.

7.3 OBJETIVOS

7.3.1 OBJETIVO PRINCIPAL

Ser a maior empresa no ramo de desenvolvimento de ferramentas didáticas nas áreas tecnológicas.

7.3.2 OBJETIVOS INTERMEDIÁRIOS

- Introduzir o *kit* P51USB em no mínimo 5 diferentes cursos no próximo ano;
- Desenvolver nos próximos dois anos um outro *kit* que atenda outros cursos mais avançados, com o suporte de um sistema operacional de tempo real;
- Firmar parcerias com fabricantes de microcontroladores e IDEs;

7.4 PRODUTOS E SERVIÇOS

7.4.1 DESCRIÇÃO DO PRODUTO E DOS SERVIÇOS

O *kit* P51USB é a base para aulas práticas de microcontroladores. Utilizando apenas interface USB é possível alimentar e gravar código no microcontrolador, facilitando a sua utilização em qualquer ambiente, seja ele em casa ou na aula.

O *kit* apresenta dimensões pequenas, podendo ser facilmente transportado, além de não ter dependência do uso de qualquer outro equipamento como fonte, gravador ou placa base.

Além disso, o *kit* conta com documentação completa, exemplos constantemente atualizados, e ambiente de desenvolvimento disponível gratuitamente. O que o torna uma excelente ferramenta no processo didático.

7.4.2 ANÁLISE COMPARATIVA

Os principais concorrentes do *kit* P51USB atualmente presentes no mercado são:

- Cerne Tecnologia (CERNE, 2014)
- *Kit*8051 (KIT8051, 2014)
- MicroGenios (MICROGENIOS, 2014)

Em uma análise simples, é possível destacar os seguintes pontos fracos destes *kits* com relação do P51USB:

- Dependem de fonte externa, enquanto o P51USB é alimentado via USB;
- Dependem de porta serial ou paralela para gravação, ambas não presentes em computadores modernos e notebooks, enquanto o P51USB utiliza a interface USB nativa;
- Ofecerem a opção de gravação via USB com conversor próprio que custa o mesmo preço do *kit*, dobrando o valor total, ou através de conversor USB-Serial;
- Não incluem incluem componentes para utilização de protocolos seriais de comunicação entre circuitos integrados (CIs) como I2C e SPI, apenas nas versões “avançadas” dos *kits* que chegam a custar mais de 3 vezes o valor do *kit* básico. O *kit* P51USB contém dois CIs para serem utilizados com tais protocolos.

7.4.3 TECNOLOGIA

As tecnologias utilizadas no *kit* P51USB são:

AT89C5131 - Microcontrolador da família 8051 com suporte nativo a USB, I2C e SPI, 32kB de memória flash interna e bootloader USB para gravação de programas;

ADC0832 - Conversor Analógico Digital de 8 bits com suporte ao protocolo SPI;

DS1307 - Relógio de tempo real com calendário e suporte ao protocolo I2C;

USB - Barramento Serial Universal, interface serial que atualmente provê as funcionalidades da antiga interface RS232 em computadores e notebooks, além de outras funcionalidades.

7.4.4 PRODUTOS E SERVIÇOS FUTUROS

De acordo com a análise de mercado apresentada a seguir, e alinhado aos objetivos da empresa, a expansão da linha de produtos e serviços se dará inicialmente com a produção de novos *kits*, atendendo às necessidades de disciplinas mais avançadas e produção de seu material didático, e posteriormente existe a possibilidade da oferta de cursos especializados, fazendo uso dos *kits* já produzidos.

7.5 ANÁLISE DE MERCADO RESUMIDA

A análise de mercado busca contextualizar o mercado atual de *kits* didáticos e apresentar os potenciais clientes. Além disso, são apresentados os principais concorrentes e a situação atual de oferta e demanda. Para todas as análises do mercado, assumiu-se que cada turma de Engenharia Eletrônica e de Engenharia de Computação possui um professor de Microcontroladores e 40 alunos que fariam uso dos *kits*.

7.5.1 SEGMENTAÇÃO DE MERCADO

De acordo com dados obtidos junto ao Ministério da educação (MEC) através da plataforma e-MEC (MEC, 2014), existem atualmente, em atividade no Brasil, 52 cursos de Engenharia Eletrônica e 135 cursos de Engenharia de Computação ofertados por 141 Instituições de ensino superior.

Considerando o pior caso em que cada instituição possua um único professor para ministrar a disciplina Microcontroladores ou equivalente e assumindo que as turmas tenham em média 40 alunos, chegamos aos valores apresentados pela tabela 1 para os potenciais compradores do *kit*.

Tabela 1: Estimativa dos tipos de clientes e suas quantidades.

Categoria	Total	%
Instituições de Ensino	141	1,82%
Professores	141	1,82%
Alunos	7480	96,36%

Fonte: Autoria própria.

Observando os dados da tabela 1 torna-se evidente que o segmento de mercado a ser atingido é aquele composto pelos alunos, uma vez que são a maioria e são aqueles que de fato irão fazer o uso do *kit*. Entretanto, devido a relação existente entre as instituições e professores com os alunos, devemos buscar esses dois grupos como parceiros, aumentando a visibilidade e com isso atingindo mais clientes em potencial.

7.5.2 SEGMENTO ALVO

Conforme já observado anteriormente, o segmento alvo de mercado é aquele composto pelos alunos dos cursos de Engenharia Eletrônica e Engenharia de Computação.

Inicialmente o *kit* será utilizado na instituição de ensino para a qual foi desenvolvido (UTFPR). Na sequência, será iniciado um processo de divulgação e formação de parcerias, para aumentar a abrangência.

Neste âmbito, o *kit* resolve os problemas atuais de custo e simplicidade de utilização.

7.5.2.1 NECESSIDADES DO MERCADO

As necessidades atuais do mercado estão diretamente relacionadas com a qualidade e eficiência no processo de ensino, sendo elas:

- Custo baixo, para que o *kit* possa ser adquirido por todos os alunos;
- Facilidade na montagem e utilização do *kit*, para que seja possível focar na atividade a ser realizada e não em procedimentos secundários para funcionamento e operação do *kit*;
- Tamanho reduzido sem o comprometimento da funcionalidade, para que seja fácil para o aluno transportar o *kit*.

7.5.2.2 TENDÊNCIAS DO MERCADO

As tendências atuais de *kits* de microcontroladores, inclusive alguns com propostas didáticas, tem sido a de *kits* compactos, com funcionalidades mínimas embutidas mas com capacidade de expansão através de outras placas encaixadas sobre a placa base. Nessa categoria se enquadram *kits* como o Arduino (ARDUINO, 2014) e o Launchpad (TI, 2014).

A principal característica desejada para este tipo de *kit* é que o funcionamento do microcontrolador em si seja abstraído e com isso a programação do mesmo seja simplificada ao ponto de possibilitar o uso até mesmo por quem tem pouca experiência em programação. Essa característica é pouco desejada em uma disciplina introdutória de Microcontroladores em um curso de Engenharia, uma vez em que um dos objetivos é a compreensão do funcionamento do microcontrolador.

7.5.2.3 CRESCIMENTO DO MERCADO

De acordo com as tendências do mercado atual, o crescimento deste mercado deverá se dar na direção de *kits* que abstraíam o funcionamento do microcontrolador, permitindo que pessoas com pouca experiência de programação o façam. Neste cenário haverá a criação

de uma lacuna na produção de *kits* com o objetivo de servir como base para o ensino do microcontrolador em questão.

Esta lacuna deverá ser preenchida com o *kit* P51USB e seus sucessores.

7.5.3 ANÁLISE DA INDÚSTRIA

7.5.3.1 PLAYERS

Os players são divididos em categorias, de acordo com a sua função no mercado, e são eles:

Fornecedores - Fabricantes e Fornecedores de componentes, desenvolvedores de IDEs e Fabricantes de placas de circuito impresso;

Clientes - Alunos;

Parceiros - Professores, Universidades e sites especializados;

Concorrentes - Outros fabricantes de *Kits*.

7.5.3.2 MODELO DE DISTRIBUIÇÃO

Inicialmente dois modelos de distribuição serão adotados, sendo eles:

1. Quando a venda for direta, por meio de website, a entrega será feita por meio postal ou transportadora.
2. Quando a venda for por meio de algum representante, sejam os sites especializados ou os professores, a distribuição será de acordo com a definição do representante.

7.5.3.3 MODELO DE COMPETITIVIDADE

O modelo de competitividade adotado é o de apresentar recursos inovadores, em comparação aos concorrentes, enquanto se mantem o menor preço. Sendo o menor preço um fator limitante, vários recursos que poderiam estar presentes como extensões ao *kit* foram omitidos.

O preço foi estabelecido tomando-se por base o preço de todos os componentes do *kit*, acrescido de impostos, despesas comerciais e pouco ou nenhum lucro, buscando não gerar prejuízos no final da conta.

7.5.3.4 PRINCIPAIS PLAYERS

Os principais players, por categoria, são:

Fornecedores - ATMEL, Mouser e Keil;

Clientes - Alunos da UTFPR;

Parceiros - Professores da UTFPR;

Concorrentes - MicroGenios.

7.6 PLANO FINANCEIRO

7.6.1 CONSIDERAÇÕES

Os tópicos a seguir apresentam a análise financeira para este projeto, considerando uma inflação anual de 6% e um investimento inicial sem incidência de juros, uma vez que o montante não é tão alto e pode ser proveniente de capital próprio.

7.6.2 INVESTIMENTO INICIAL

O investimento inicial, no valor de R\$ 12.735,00 foi calculado levando-se em consideração a necessidade de produção de 200 *kits* para venda imediata, além dos custos com desenvolvimento, e sua composição detalhada é apresentada na tabela 2.

7.6.3 PROJEÇÃO DE RESULTADOS E DE FLUXO DE CAIXA

A projeção do fluxo de caixa foi feita tendo como base uma projeção de vendas de *kits* P51USB com um aumento progressivo de acordo com a quantidade de turmas que farão adesão ao uso do *kit*, sendo 200 *kits* no primeiro ano, 500 no segundo, 700 no terceiro, 800 no quarto e 900 no quinto.

Ainda, foi considerado um custo de R\$ 5,00 por venda para questões logísticas, custo este que poderia ser reduzido buscando-se alternativas como distribuidores.

7.6.4 INDICADORES FINANCEIROS

Para as análises financeiras feitas a seguir, os seguintes indicadores foram utilizados:

Tabela 2: Detalhamento dos custos iniciais.

Desenvolvimento			
Produto	Quantidade	Valor unitário	Valor total
Protótipo	3	R\$ 70,00	R\$ 210,00
Placas para os professores	5	R\$ 65,00	R\$ 325,00
Subtotal		R\$ 135,00	R\$ 535,00
Produção			
Produto	Quantidade	Valor unitário	Valor total
PCB	200	R\$ 8,00	R\$ 1.600,00
AT89C5131	200	R\$ 20,00	R\$ 4.000,00
ADC0832	200	R\$ 3,00	R\$ 600,00
DS1307	200	R\$ 5,00	R\$ 1.000,00
Demais componentes	200	R\$ 10,00	R\$ 2.000,00
LCD (opcional)	200	R15,00R 3.000,00	
Subtotal		R\$ 61,00	R\$ 12.200,00
Total		R\$ 196,00	R\$ 12.735,00

Fonte: Autoria própria.

- Payback simples
- Valor presente líquido
- Taxa interna de retorno

7.6.5 ANÁLISE DE BREAK-EVEN

A Análise de Break-even é uma maneira de avaliar o tempo que leva para que um projeto se pague. Para isso foi aplicado o método de payback simples, onde o valor do dinheiro no tempo não é levado em consideração. Apesar de simples, esta abordagem demonstra rapidamente um valor aproximado de tempo no qual o projeto estará equilibrado.

O cálculo do payback simples apresenta o equilíbrio no final de 1 anos e 4 meses.

7.6.6 VIABILIDADE DO EMPREENDIMENTO

Analisando o fluxo de caixa apresentado na tabela 3 através do Valor Presente Líquido e da Taxa interna de retorno, encontramos valores que parecem muito atrativos para a execução do projeto após o break-even. O Valor presente líquido no quinto ano é de R\$ 1.625,62, enquanto a Taxa interna de retorno é de 659%.

Tabela 3: Projeção do fluxo de caixa.

Receita Bruta em R\$					
Vendas de Produtos	17361.96	46009.2	68277.66	82713.5	98635.85
Total da Receita Bruta	17361.96	46009.20	68277.66	82713.50	98635.85
(-)Imposto sobre vendas (18%)	3125.15	8281.66	12289.98	14888.43	17754.45
(=)Receita Líquida	14236.81	37727.55	55987.68	67825.07	80881.40
Custos dos Produtos Vendidos em R\$					
Matéria Prima e Materiais Diretos	12735.00	33747.75	50081.66	60670.36	72349.49
(-)Total dos Custos dos Produtos Vendidos	12735.00	33747.75	50081.66	60670.36	72349.40
(=)Margem de Contribuição	1501.81	3979.80	5906.02	7154.72	8532.00
Despesas de Vendas em R\$					
Fixas	1000.00	2500.00	3500.00	4000.00	4500.00
3% sobre Receita Bruta	520.86	1380.28	2048.33	2481.41	2959.08
(-)Total de Despesas de Vendas	1520.86	3880.28	5548.33	6481.41	7459.08
Despesas Fixas em R\$					
(=)Lucro Operacional	-19.05	99.52	357.69	673.31	1072.93
(-)Imposto de Renda	0	9.95	35.77	67.33	107.29
(=)Receita Líquida após IR	-19.05	89.57	321.92	605.98	965.63
Fluxo de caixa em R\$					
(=)Fluxo de Caixa Livre	-19.05	89.57	321.92	605.98	965.63
(=)Caixa Gerado no Período	-19.05	89.57	321.92	605.98	965.63
(=)Caixa Gerado Acumulado	-19.05	70.52	392.44	998.42	1964.05

Fonte: Autoria própria.

Entretanto, deve-se verificar que estes valores não representam adequadamente o capital de giro necessário para a execução do projeto, uma vez que a produção é sob demanda e os custos são imediatamente cobertos com as vendas.

Portanto, a menos que exista uma outra maneira para manter este capital de giro, o projeto será viável apenas em escalas pequenas onde um investimento inicial seria considerado e em algum tempo ele estaria coberto, propondo a oportunidade de ampliação do negócio.

Uma das principais causas da falta de capital de giro foi a consideração da existência de pouco ou nenhum lucro no cálculo do preço praticado, sendo esta uma das condições para manter o preço o mais baixo possível sem que houvesse prejuízo.

REFERÊNCIAS

- ARDUINO. **Arduino home page**. 2014. Disponível em: <<http://www.arduino.cc>>. Acesso em: 5 de março de 2014.
- ATMEL. **AT89C5131A USB Bootloader**. 2007. Disponível em: <<http://www.atmel.com/Images/doc4287.pdf>>. Acesso em: 5 de março de 2014.
- ATMEL. **8-bit Flash Microcontroller with Full Speed USB Device**. 2008. Disponível em: <<http://www.atmel.com/Images/doc4337.pdf>>. Acesso em: 5 de março de 2014.
- AXELSON, J. 2. ed. [S.l.]: Lakeview Research, 2001.
- BOLANAKIS, D. E. et al. A teaching approach for bridging the gap between low-level and high-level programming using assembly language learning for small microcontrollers. **Computer Applications in Engineering Education**, Wiley Subscription Services, Inc., A Wiley Company, v. 19, n. 3, p. 525–537, 2011. ISSN 1099-0542. Disponível em: <<http://dx.doi.org/10.1002/cae.20333>>.
- CERNE. **Cerne Tecnologia**. 2014. Disponível em: <<http://cerne-tec.com.br/kit8051.htm>>. Acesso em: 5 de março de 2014.
- DAELN. **Matrizes curriculares**. 2014. Disponível em: <<http://engenharia.daeln.ct.utfpr.edu.br/curriculo>>. Acesso em: 5 de março de 2014.
- DAINF. **Matrizes curriculares**. 2014. Disponível em: <<http://www.utfpr.edu.br/curitiba/cursos/bacharelados/Ofertados-neste-Campus/engenharia-de-computacao/matriz>>. Acesso em: 5 de março de 2014.
- FARIA, R. A.; LOPES, H. S. **Manual da placa P51N**. 2014. Disponível em: <<http://paginapessoal.utfpr.edu.br/rubens/microcontroladores/anexos/placa-didatica-p51n>>. Acesso em: 5 de março de 2014.
- GAL, R. et al. Fpga implementation of 8-bit risc microcontroller for embedded systems. In: **Mixed Design of Integrated Circuits and Systems (MIXDES), 2011 Proceedings of the 18th International Conference**. [S.l.: s.n.], 2011. p. 323–328.
- HOVEMEYER, D.; LINK, G. Teaching Embedded Systems Using AVR Microcontrollers: Tutorial Presentation. **J. Comput. Sci. Coll.**, Consortium for Computing Sciences in Colleges, USA, v. 26, n. 3, p. 104–105, jan. 2011. ISSN 1937-4771.
- HSU, C.-M.; CHAO, H.-M. Constructing u-controller teaching tool with the integration of hardware and software technology. **International Conference on New Trends in Information and Service Science (NISS '09)**, p. 269–276, 2009.
- IAR. **IAR Embedded Workbench for 8051**. 2014. Disponível em: <<http://supp.iar.com/Download/SW/?item=EW8051-EVAL>>. Acesso em: 5 de março de 2014.

J, R.-T. de et al. 8-bit CISC Microprocessor Core for Teaching Applications in the Digital Systems Laboratory. In: **Reconfigurable Computing and FPGA's, 2006. ReConFig 2006. IEEE International Conference on**. [S.l.: s.n.], 2006. p. 1–5.

KEANE, B. et al. Developing a 32-bit microcontroller board for embedded applications. In: **Southeastcon, 2012 Proceedings of IEEE**. [S.l.: s.n.], 2012. p. 1–4. ISSN 1091-0050.

KEIL. **FlashMon51 Target Monitor**. 2014. Disponível em: <<http://www.keil.com/c51/flashmon51.asp>>. Acesso em: 5 de março de 2014.

KEIL. **Keil Tools By ARM: C51 Software - Development tools for Classic and Extended 8051 Microcontrollers**. 2014. Disponível em: <<https://www.keil.com/c51/demo/eval/c51.htm>>. Acesso em: 5 de março de 2014.

KIT8051. **Kit8051 pode ser programado pela porta USB**. 2014. Disponível em: <http://www.kit8051.com.br/product_info.php?cPath=23&products_id=125>. Acesso em: 5 de março de 2014.

KOSLOWSKI, M. A. **P51USB - Kit didático para ensino de microcontroladores**. 2014. Disponível em: <<http://mikhailkosowski.github.io/p51usb/>>. Acesso em: 5 de março de 2014.

MA, C. et al. Low cost AVR microcontroller development kit for undergraduate laboratory and take-home pedagogies. In: **Education Technology and Computer (ICETC), 2010 2nd International Conference on**. [S.l.: s.n.], 2010. v. 1, p. V1–35–V1–38.

MEC. **e-MEC**. 2014. Disponível em: <<http://emec.mec.gov.br/>>. Acesso em: 5 de março de 2014.

MICROGENIOS. **8051Study - Módulo didático com Microcontrolador AT89C51ED2 - Família 8051 ATMEL**. 2014. Disponível em: <<http://www.microgenios.com/?1.34.0.0,351,8051study-modulo-didatico-com-microcontrolador-at89c51ed2-familia-8051-atmel.html>>. Acesso em: 5 de março de 2014.

RAISONANCE. **RKit-51**. 2014. Disponível em: <http://support-raisonance.com/extranet/tools/index.php?param=tool_id_6>. Acesso em: 5 de março de 2014.

REESE, R.; JONES, B. Improving the effectiveness of microcontroller education. In: **IEEE SoutheastCon 2010 (SoutheastCon), Proceedings of the**. [S.l.: s.n.], 2010. p. 172–175.

SCHUBERT, S. **MC HCK: The hacker's microcontroller board: small. cheap. versatile. open source**. 2014. Disponível em: <<https://mchck.org/>>. Acesso em: 5 de março de 2014.

SELVAKUMAR, R. S. K. et al. Learning embedded systems with multi-microcontroller evaluation tool. In: **Electrical, Control and Computer Engineering (INECCE), 2011 International Conference on**. [S.l.: s.n.], 2011. p. 288–292.

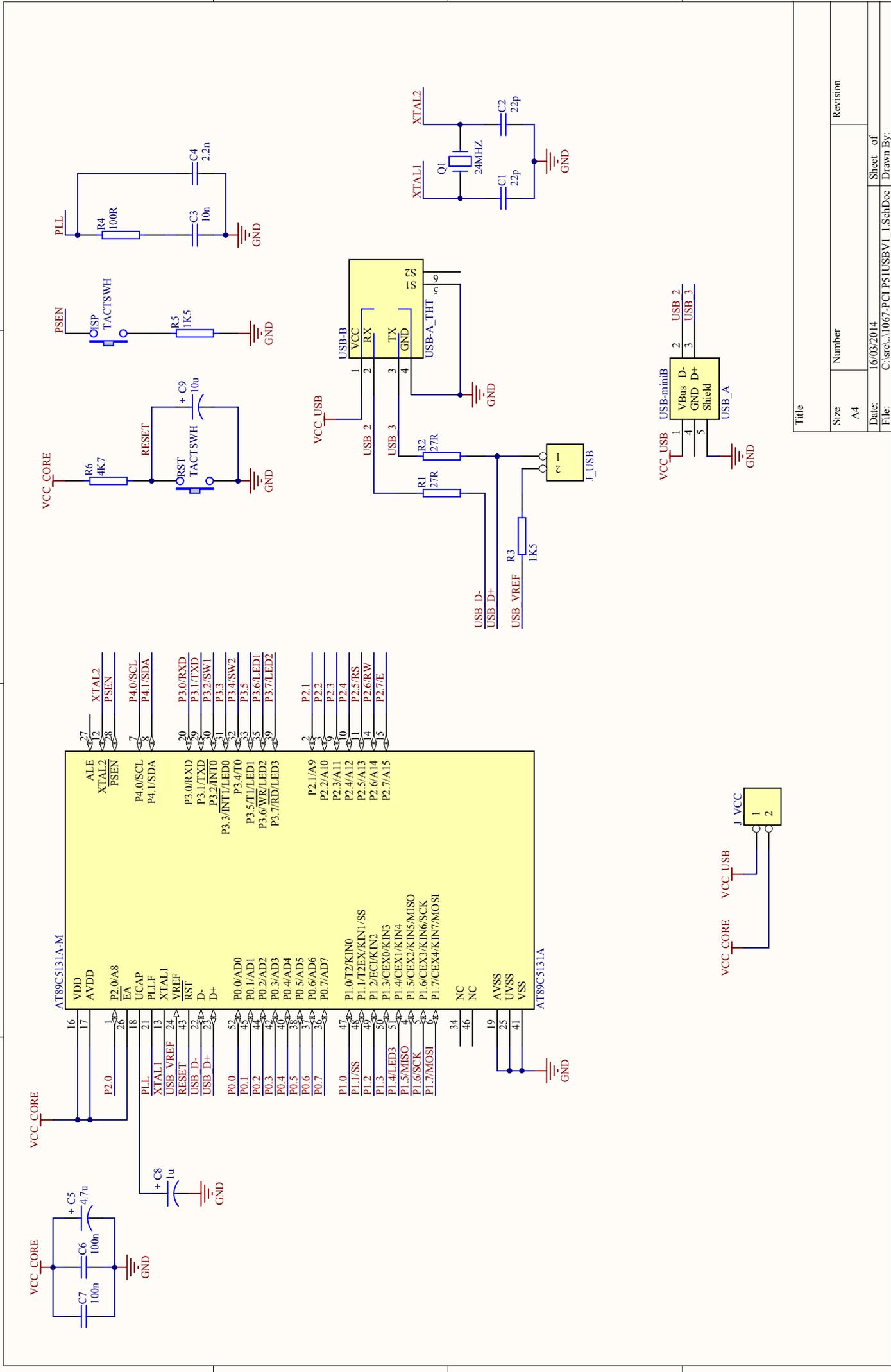
ST. **Press release: STMicroelectronics' STM32 Discovery Kit Establishes New Industry Standard for Low-Cost Tools for 32-bit Microcontrollers**. 2014. Disponível em: <http://www.st.com/internet/com/press_release/p3065.jsp>. Acesso em: 5 de março de 2014.

TI. **MSP430 LaunchPad Value Line Development Kit**. 2014. Disponível em: <<http://www.ti.com/tool/msp-exp430g2>>. Acesso em: 5 de março de 2014.

USB-IF. **USB 2.0 Specification.** 2014. Disponível em: <http://www.usb.org/developers/docs/usb20_docs/#usb20spec>. Acesso em: 5 de março de 2014.

ZHANG, X. Enhancing teaching of computer organization through concrete examples and laboratory experiences. In: **Computer Science Education, 2009. ICCSE '09. 4th International Conference on.** [S.l.: s.n.], 2009. p. 1357–1362.

APÊNDICE A - ESQUEMÁTICO



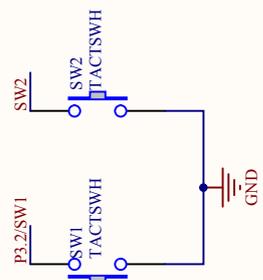
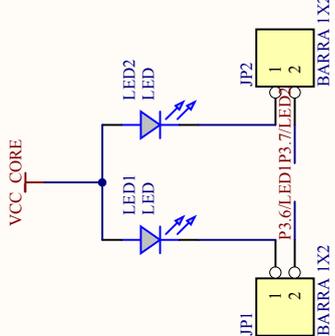
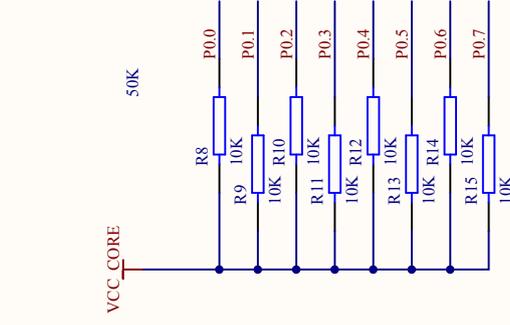
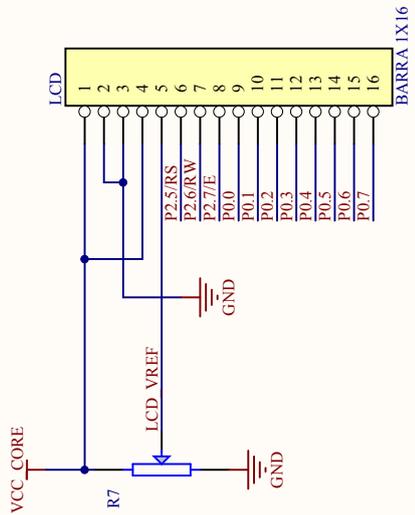
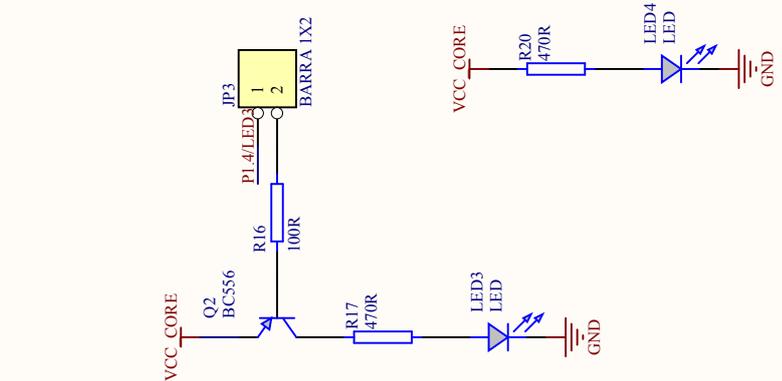
Title	
Size	Number
A4	
Date:	Revision
16/03/2014	
File:	Sheet of
C:\src\11067-PCI P51USBV1_1_SchDoc	Drawn By:

A

B

C

D



Title

Size	Number	Revision
A4		

Date:	Sheet of	Drawn By:
16/03/2014		
File:	C:\src\1067-PCI P51USBVI_2.SchDoc	

A

B

C

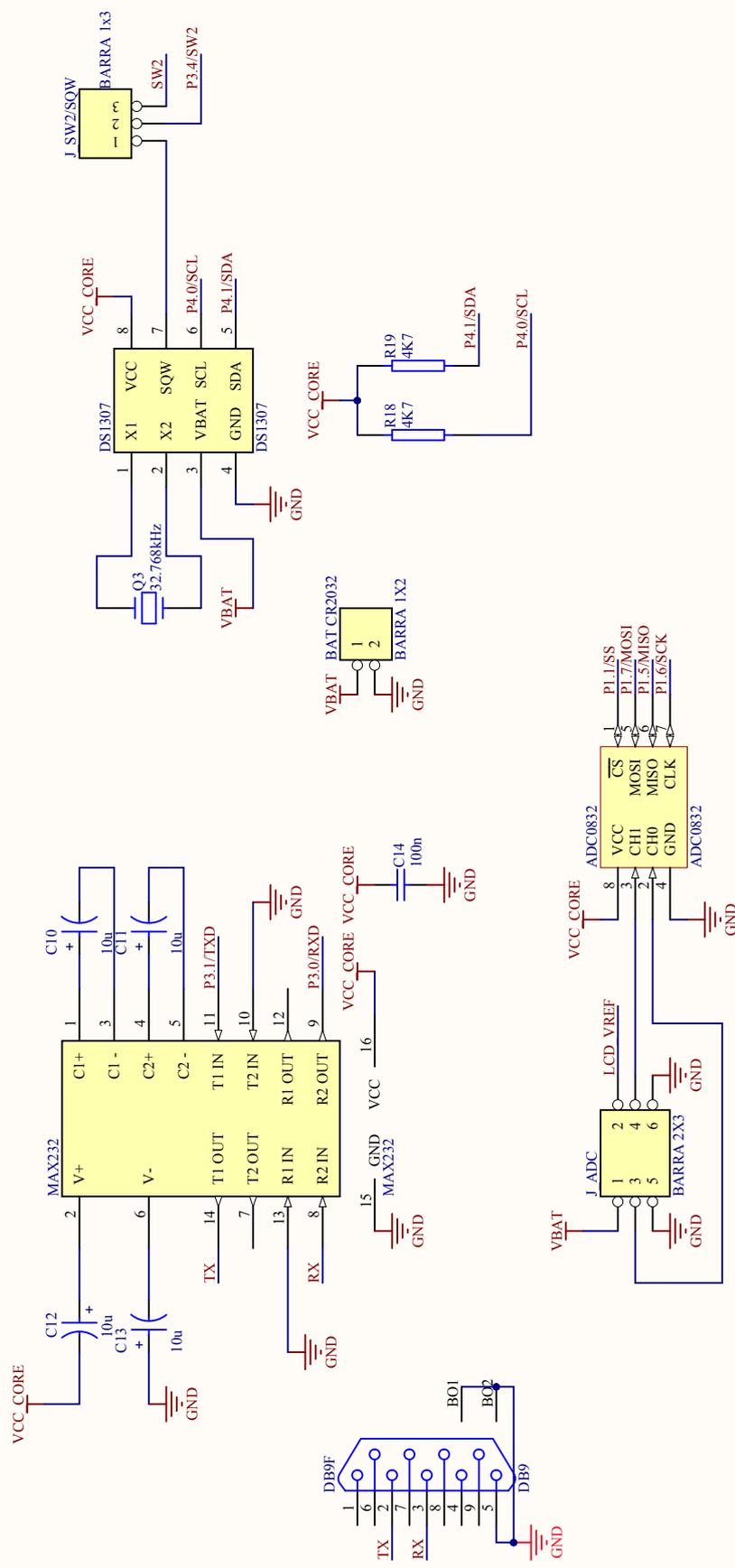
D

A

B

C

D



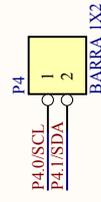
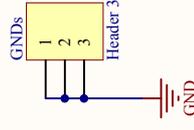
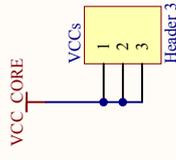
Title	
Size	Number
A4	
Date:	Revision
16/03/2014	
File:	Sheet of
C:\src\...\1067-PCI P51USBVI_3.SchDoc	Drawn By:

A

B

C

D



Title

Size

A4

Date:

16/03/2014

Number

Revision

Sheet of

4

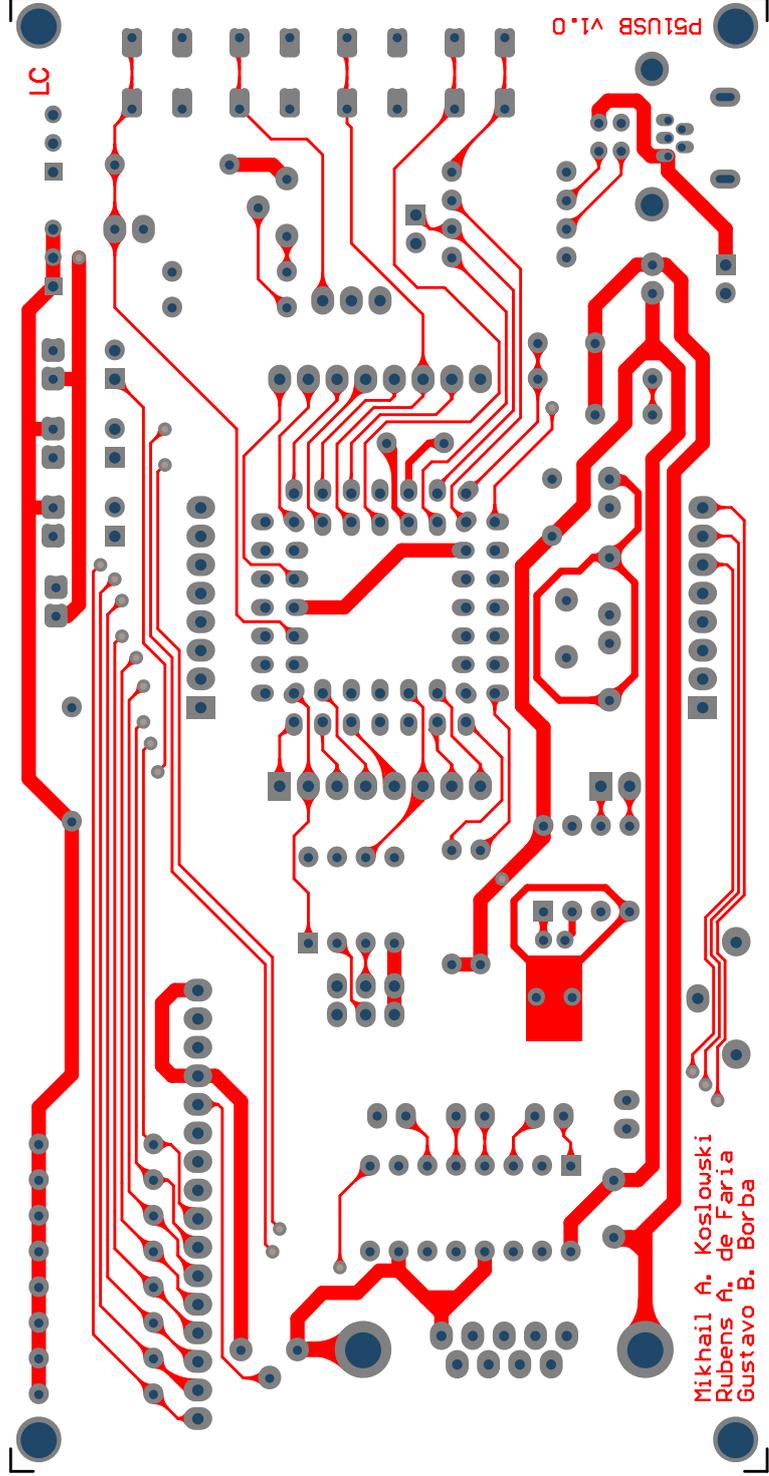
File:

C:\src\...\1067-PCI P5 IUSBV1_4.SchDoc

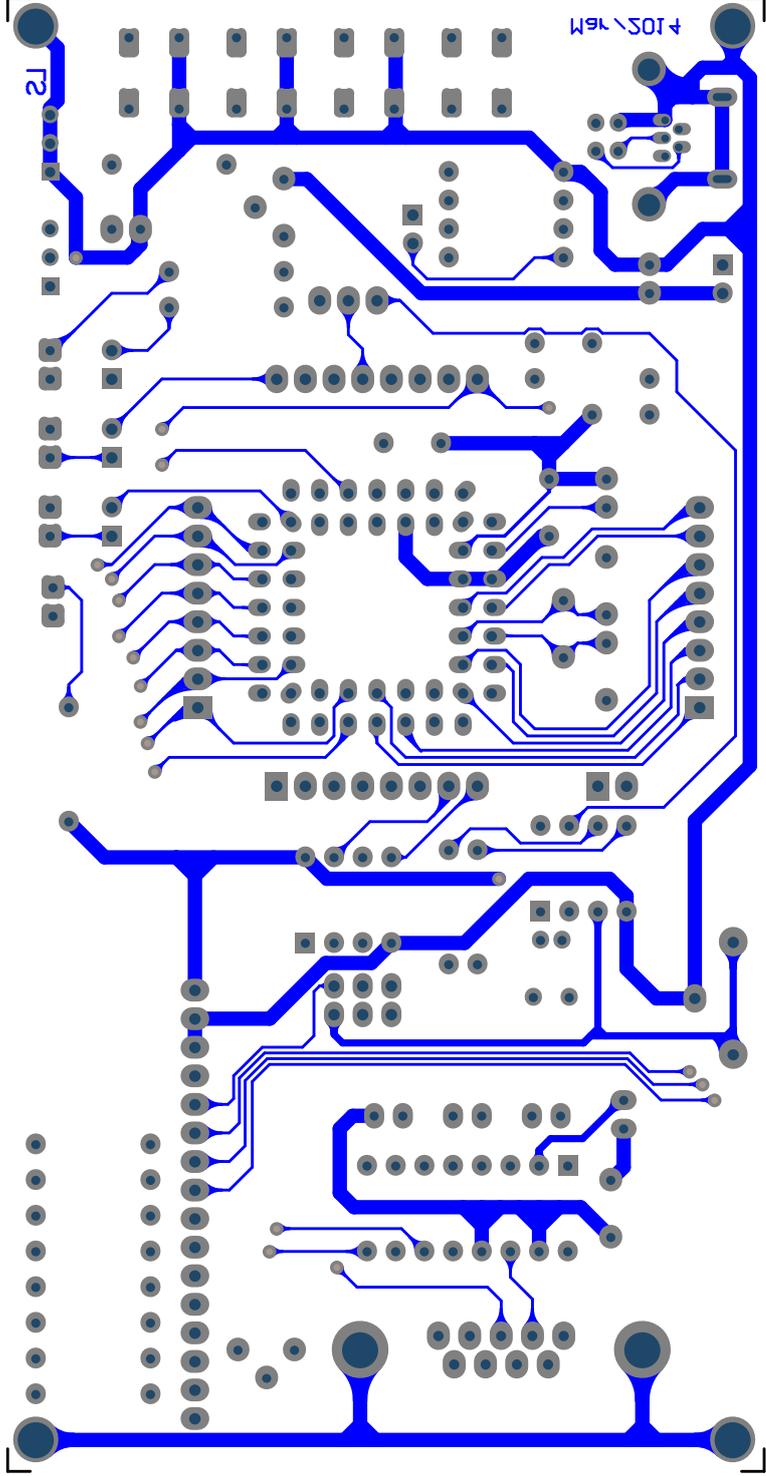
Drawn By:

APÊNDICE B – ARQUIVOS GERBER

PCB FILE NAME:	1067-PCI PS1USBV0.PcbDoc
PLOT FILE NAME:	Top Layer
LAYER NAME:	Top Layer
DATE:	16/03/2014

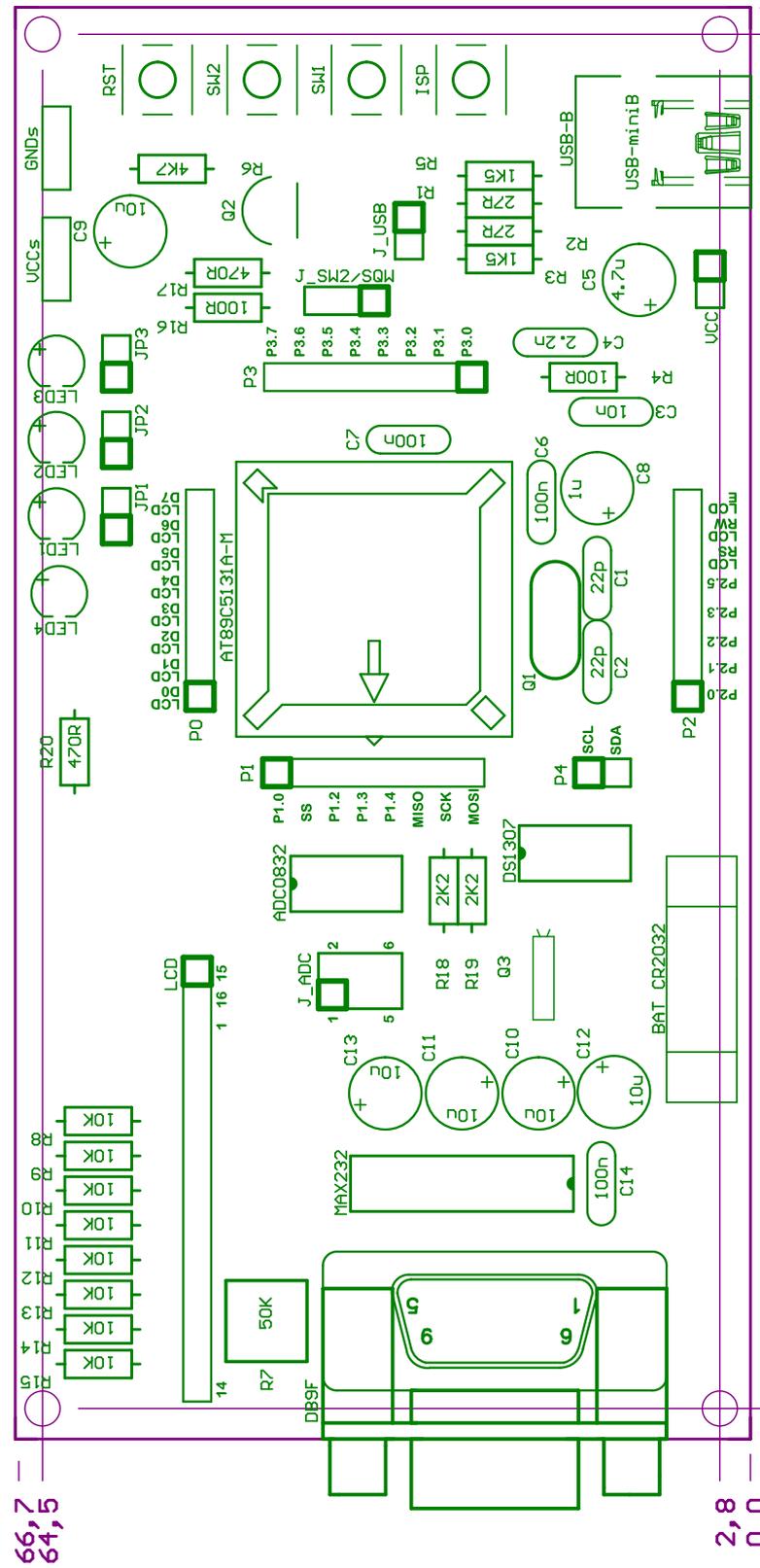


PCB FILE NAME:	1067-PCI PS1USBV0.PcbDoc
PLOT FILE NAME:	Bottom Layer
LAYER NAME:	Bottom Layer
DATE:	16/03/2014



SUMARIO DO DESENHO (DESIGN SUMMARY)	
NUM. DE LAYERS (NUMB. OF LAYERS)	2
ESPESSURA DA PLACA (BOARD THICKNESS)	1,6mm +/-10%
TAMANHO (TOTAL SIZE)	130,9 x 66,7mm
NUM. TOTAL DE FUROS (TOTAL HOLE COUNT)	310
MENOR/MAIOR DIAM. (SMALLEST/LARGEST HOLE DIA.)	0,3 / 4,0mm
TRACO / ESPACO (TRACE AND SPACE WIDTHS)	0,254 / 0,2mm
COMPONENTES SMD (SURFACE MOUNT COMPONENTS)	SIM (YES)
MENOR PITCH (MINIMUM PITCH)	0,5mm
NUM. PADS SMD (SMT PADS)	487

PCB FILE NAME:	1067-PCI PS1USBV0.PcbDoc
PLOT FILE NAME:	Top Silkscreen Overlay
LAYER NAME:	1067-PCI PS1USBV0
DATE:	16/03/2014

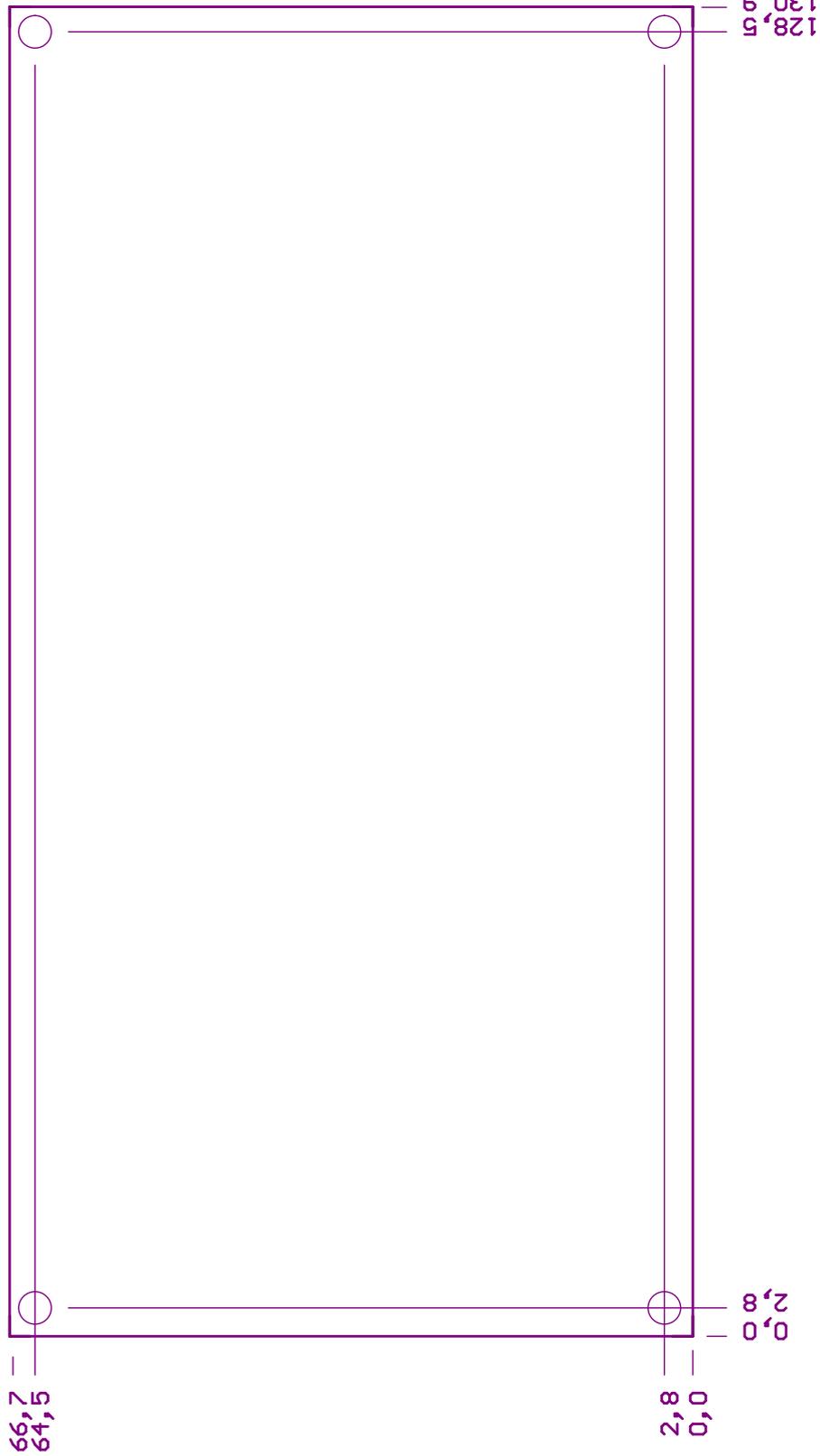


128,5
130,9

2,8
0,0

SUMARIO DO DESENHO (DESIGN SUMMARY)	
NUM. DE LAYERS (NUMB. OF LAYERS)	2
ESPESSURA DA PLACA (BOARD THICKNESS)	1,6mm +/-10%
TAMANHO (TOTAL SIZE)	130,9 x 66,7mm
NUM. TOTAL DE FUROS (TOTAL HOLE COUNT)	310
MENOR/MAIOR DIAM. (SMALLEST/LARGEST HOLE DIA.)	0,3 / 4,0mm
TRACO / ESPACO (TRACE AND SPACE WIDTHS)	0,254 / 0,2mm
COMPONENTES SMD (SURFACE MOUNT COMPONENTS)	SIM (YES)
MENOR PITCH (MINIMUM PITCH)	0,5mm
NUM. PADS SMD (SMT PADS)	487

PCB FILE NAME:	1067-PCI PS1USBV01.PcbDoc
PLOT FILE NAME:	Mechanical 1
LAYER NAME:	Mechanical 1
DATE:	16/03/2014



Symbol	Hit Count	Tool Size	Physical Length	Route Path Length	Plated	How
□	20	23.622mil (0.6mm)			PTH	Ro
☆	5	27.559mil (0.7mm)			PTH	Ro
○	138	31.496mil (0.8mm)			PTH	Ro
▽	67	35.433mil (0.9mm)			PTH	Ro
✕	72	39.37mil (1mm)			PTH	Ro
◇	2	78.74mil (2mm)			PTH	Ro
⊕	6	125.984mil (3.2mm)			PTH	Ro
⊗	2	27.559mil (0.7mm)	66.929mil (1.7mm)	39.37mil (1mm)	PTH	Slo
	312	Total				

Slot definitions : Route Path Length = Calculated from tool start centre position to tool end centre position
Physical Length = Route Path Length + Tool Size = Slot length as defined in the

PCB FILE NAME:	1067-PCI PS1USBV0.PcbDoc
PLOT FILE NAME:	
LAYER NAME:	
DATE:	16/03/2014

