

Projeto de Microprocessadores e Microcontroladores Misturador Magnético para o Laboratório de Química da FGA

Josué Bezerra Bonfim Filho, Abhay Caran Das Baraky Dias

Resumo—De acordo com os requisitos necessários para o aprendizado sobre relações de balanceamento de equações químicas através de dados de titulação, este artigo tem o objetivo de desenhar um produto cuja função principal é auxiliar os estudantes da disciplina de Química Experimental a evitar acidentes. O documento explicita o projeto de software e hardware do sistema automatizado de mistura, controle de válvula e detecção do ponto de equivalência entre concentração de ácidos e bases. O produto usa o microcontrolador MSP430G2553 da fabricante Texas Instruments e sensores e periféricos de fabricantes variadas.

Keywords—LDR, microcontrolador, MSP430, display, titulação, misturador, detecção, cor.

I. INTRODUÇÃO

A UnB Gama oferece 5 cursos de graduação, os quais são Engenharia Automotiva, Engenharia Aeroespacial, Engenharia Eletrônica, Engenharia de Energia e Engenharia de Software[10]. Os alunos que ingressam na universidade cursam disciplinas que são comuns a todos até o 3º semestre. Dentre as disciplinas comuns, Química Geral Teórica e Experimental se preocupam em entregar uma abordagem geral dos princípios da Química e suas aplicações, usando exemplos de compostos orgânicos e inorgânicos[9]. Para que o objetivo seja alcançado, os estudantes são submetidos à uma série de procedimentos práticos e testes teóricos.

O experimento de "Balanceamento de uma equação química por meio de dados de titulação" tem como objetivo determinar a concentração exata de soluções aquosas de bases fortes e ácidos fracos. Para isso, os estudantes devem titular a solução presente na bureta com a solução no erlenmeyer através do gotejamento controlado por uma válvula, até que a solução do erlenmeyer mude de cor por conta do indicador ácido-base fenolftaleína. Assim, o experimento exige dos estudantes agilidade e coordenação motora para agitar o erlenmeyer e abrir a válvula da bureta[7].

Encorajados pelo corpo docente da UnB Gama, o projeto da disciplina de Microprocessadores e Microcontroladores visa a construção de um produto que realize a abertura da válvula que será feita por um motor de passo, a mistura das soluções, a qual será feita por um misturador magnético[8] e a detecção da mudança de cor por um sensor para realizar a parada do gotejamento. A fim de controlar o processo, será usado um microcontrolador MSP430G2553.

II. DESENVOLVIMENTO

A. Justificativa e Benefícios

Existe uma dificuldade entre os estudantes da Disciplina de Química Geral Experimental na UnB Gama de realizar o experimento de "Balanceamento de Equações Químicas através de dados de Titulação" devido ao fato que o mesmo exige destreza com o manuseio das vidrarias do laboratório. A falta de habilidade resulta na demora da realização do experimento com repetições, o extravio de material útil e até quebra das vidrarias. Outro aspecto pontual que o laboratório da faculdade enfrenta trata-se da pequena quantidade de equipamentos que implica na limitação da quantidade de pessoas realizando o experimento e dessa maneira a repetição ou erro ao realizar uma execução pode atrapalhar o andamento da aula.

Em conversas com os professores e técnicos, vendo a dificuldade dos estudantes e a falta de recursos para a compra de material para os laboratórios na FGA, o produto será destinado ao uso experimental. Em pesquisas de mercado, o preço de um misturador magnético chega a R\$1300,00.

O maior benefício que o produto traz é a facilidade do uso no ambiente laboratorial, o preço final do produto e sobre tudo a inovação da parada do gotejamento instantaneamente conforme a solução muda de cor. Este equipamento pode reduzir o período da aula e permitir que mais alunos completem o experimento sem demais complicações.

B. Revisão Bibliográfica

1) Experimento - Balanceamento de Equações Químicas Através de Dados de Titulação: A titulometria, ou titulação é um método de análise quantitativa que determina a concentração de uma solução[7]. Dosar uma solução é determinar a sua quantidade por intermédio de outra solução de concentração conhecida. Assim, a titulação ácido-base se torna uma importante ferramenta para análises laboratoriais e é dividida em:

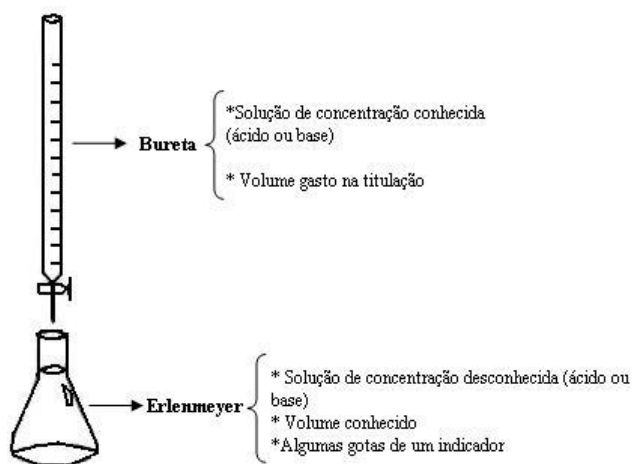
- **Acidimetria:** determinação da concentração de um ácido.
- **Alcalimetria:** determinação da concentração de uma base.

Existem também as substâncias que mudam de cor na presença de ácidos e bases, os quais são os **indicadores ácido-base**. [7] Os indicadores mais usados em laboratórios e as cores dos pontos de viragem estão descritos na tabela I:

Os equipamentos usados habitualmente em uma titulação são uma **bureta** e um **erlenmeyer**, representados na figura 1.

Tabela I. TABELA DOS INDICADORES DE TITULAÇÃO

Indicador	Meio Ácido	Meio Básico
Tornassol	róseo	azul
Fenolftaleína	incolor	vermelho
Alaranjado de metila	vermelho	amarelo
Azul de bromotimol	amarelo	azul

Figura 1. Esquemático de Titulação - Bureta e Erlenmeyer - Fonte <https://brasilescola.uol.com.br/quimica/titulacao.htm>

Ao abrir a torneira da bureta, começa a reação entre ácido e base no erlenmeyer. A titulação termina quando é evidenciada a mudança de cor da solução do erlenmeyer. A coloração obtida dependerá do tipo de indicador usado.

Os estudantes da UnB Gama realizam o experimento de acordo com a metodologia a seguir:

- Pipetar 1mL de vinagre com pipeta volumétrica em um erlenmeyer de 250mL;
- Adicionar 50mL de água destilada com a proveta;
- Acrescentar 3 gotas de solução indicadora de fenolftaleína;
- Titular com a solução preparada anteriormente de NaOH 0,05M até o aparecimento de uma coloração levemente rosa;
- Anotar o volume gasto;
- Realizar a duplicata.

A partir das instruções acima, os estudantes, juntamente com todo o conhecimento teórico, podem calcular a concentração da solução presente no erlenmeyer.

2) **Agitadores Magnéticos - "Magnetic Stirrer"**: Um agitador magnético é um aparelho de laboratório destinado a agitar soluções por meio de uma pequena barra magnética movida por um campo magnético rotativo. Os agitadores magnéticos são utilizados para agitar líquidos ou soluções por longos períodos de tempo. São muito utilizados em trabalhos laboratoriais, mas não podem ser utilizados com líquidos muito viscosos ou se a mistura reacional for heterogênea com grande quantidade de sólidos dispersos[5].

O magnetismo é o fenômeno de atração ou repulsão entre ímãs ou materiais ferromagnéticos e é um dos principais funda-

mentos para que os agitadores magnéticos funcionem. A partir desta atração é possível mover objetos através de superfícies e barreiras que não contenham materiais ferromagnéticos.

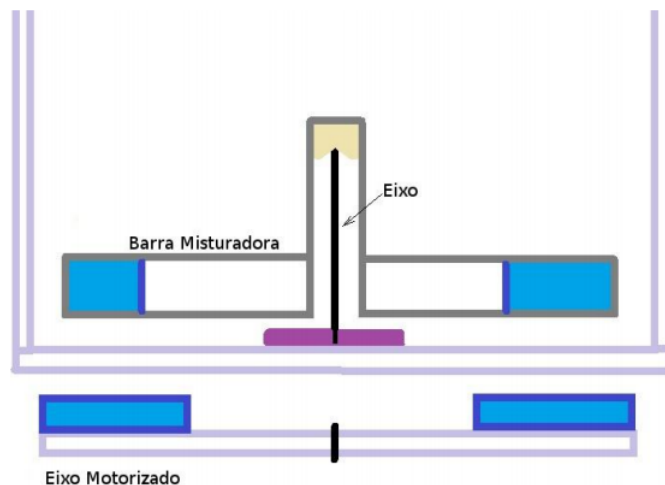


Figura 2. Esquemático de funcionamento interno do agitador magnético.

De acordo com a figura 2, o eixo motorizado possui 2 ímãs em suas pontas. Acima dele, existe uma superfície não magnetizada que separa o eixo da barra magnetizada, que por sua vez, é uma barra feita de material inerte com materiais ferromagnetizados nas pontas. (fonte: misturador magnetico)

3) **Sistema de Detecção Eletrônica de Cores**: Os sistemas de detecção eletrônica de cores são aqueles que utilizam sensores de cor para tomar decisões de maneira autônoma. São sistemas de ampla aplicação na indústria[8]. O sensor de cor consegue detectar fielmente diferenças sutis de cor que geralmente são difíceis de captar com sensores convencionais. Como eles geralmente se baseiam no padrão RGB (Red, Green e Blue), a detecção é pouco afetada por vibrações ou variação de posição.

Os sistemas podem ser usados para separação de objetos por cor, verificação de cor da tampa, etiquetas mal colocadas, detecção de cor para tomada de decisão, detecção de junção em aerossois ou latas, detecção de presença ou ausência de impressão, detecção de graxa em rolamentos de esfera, verificação de montagem correta, entre outras.

4) **Motor de Passo**: Um motor de passo é um tipo de motor elétrico usado quando algo tem que ser posicionado com precisão e ou rotacionado em um ângulo exato e que convém ao nosso projeto pois a abertura da válvula é pequena e não necessita de grande torque para a perfeita operação. Nesse tipo de motor a rotação do balancete é controlada por uma série de campos eletromagnéticos que são ativados e desativados eletronicamente. Além disso, é utilizado PWM para o controle da velocidade do motor e do tipo de movimento que eles efetuam, que relaciona consumo de energia, intensidade do Torque, velocidade e precisão.

Existem dois tipos de motores de passo: unipolares e bipolares. Em um motor unipolar, a corrente flui em apenas uma direção na bobina já nos bipolares, a corrente flui em ambas

as direções.

Pode-se definir 3 tipos principais de movimentação dos motores de passo:

Wave Drive Operation - este modo realiza a movimentação através de da conexão de uma bobina de cada vez que implica em rotações de 90° por ciclo. É uma opção para aplicações de baixo consumo e que um alto torque não seja necessário;

Full-Step Operation - esta modo também realiza rotações de 90°, porém são utilizadas duas bobinas simultaneamente de forma a intensificar o torque durante a movimentação. Assim, esta se torna uma opção para aplicações que exigem alto torque.

Half-Step Operation - este modo realiza passos de 45° e portanto consegue maior precisão para a movimentação do servo motor e para isto é utilizada uma onda de PWM de 3 (high) por 5 (low) para que ora sejam ligadas apenas uma bobina e ora duas de forma a se completar a rotação. Esta opção requer uma maior número de movimentos para a rotação e portanto consome mais energia e perde velocidade, todavia a resolução do passo aumenta e se torna uma boa para aplicações que demandam grande precisão.

5) **LCD**: Um display é um componente eletrônico onde a imagem é formada pela polarização da luz. Pode ser no formato de 7 segmentos ou matricial. No formato matricial é possível a representação de símbolos. É geralmente controlado por um circuito integrado Hitachi HD44780.

Existem vários tipos de LCD no mercado atualmente e são descritos por AxBy, onde A é o número de colunas e B é o número de linhas. Existem também os modelos com e sem back-light, o qual é um conjunto de LEDs no fundo do display que permite que o escrito possa ser lido em ambientes de baixa ou nenhuma luminosidade. Os modelos mais comuns se comunicam através de pinos de I/O de seu microcontrolador chamados de LCDs paralelos. Outros modelos se comunicam em outros protocolos, como os I2C, SPI, etc.

O LCD é formado por "células" onde ficam os caracteres. O display 16x2 possui 32 células e cada célula é composta por uma matriz de 8x5 pixels. A ferraemnta dispõem de 16 portas das quais 8 portas são canais de dados. Entretanto, como o microprocessador utilizado demanda a economia de porta podemos utilizar a função do display para operar com apenas 4 canais.

A alimentação do display demanda 5V para os caracteres e para a Backlight, que é a luz verde de fundo que realça as informações do display. E além disso, também conta com uma porta de contraste para o ajuste da luz e mais 3 portas de controle.

Descrição dos Terminais:

- Vss - GND ou terra;
- Vdd - porta de alimentação 3 5V;
- Vo - Porta para controle do contraste através de uma resistência variável;
- RS - Canal "Register Select" que faz a seleção entre os modos de inserção de caracteres (nível lógico alto) e o modo de inserção de comandos (nível lógico baixo);
- RW - Porta que realiza a seleção entre receber texto para ser escrito no display (nível lógico baixo) e ler

os dados do display para fornecer ao microcontrolador (nível lógico alto);

- E - Canal que permite que todas as operações sejam concluídas e seu funcionamento requer que a porta passe do nível lógico alto para baixo e sem seguida retorne para o alto;
- D0 D7 - Canais de Dados para os modos de comando e de caracteres. Vale notar que ao selecionar a opção de 4 bits são utilizados apenas os bits mais significativos (D4 D7) e os dados são transmitidos seguindo a sequência de 4 bits mais significativos e depois os 4 bits menos significativos;
- A - É a porta de alimentação para a Backlight e é ligada ao 5V;
- k - Canal ligado ao GND para a alimentação do Backlight;

C. Descrição de Hardware

O produto é composto pelos componentes abaixo:

- Um fotoresistor LDR
- Um driver de Motor A4988
- Um motor de passo
- Um cooler
- Ímãs de neodímio
- MSP430FR2433
- Display LCD 16x2
- Botões

1) **LDR - Light Dependent Resistor**: É formado por duas células fotocondutivas com resposta espectral de comprimento de onda similar ao do olho humano. A resistência da célula cai de acordo com o incremento da intensidade luminosa captada por ele.

A sensibilidade de um fotodetector é a relação entre a queda de intensidade de luz captada pelo dispositivo e o sinal de saída. No caso do LDR, a relação é entre a incidência de luz, dada em [LUX], com a resistência da célula, dada em [Ω]. Esta relação pode ser visualizada de maneira efetiva na figura 3.

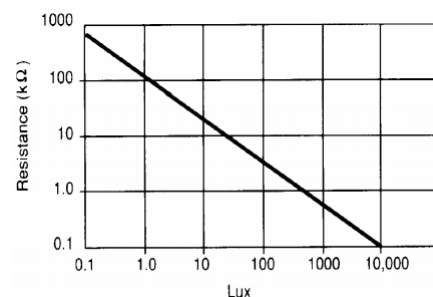


Figura 3. Resistência do LDR em função da intensidade luminosa

Como um olho humano, a sensibilidade relativa de uma célula fotocondutora é dependente do comprimento de onda da luz incidente. Cada tipo de material de fotocondutores tem uma curva característica para a resposta relativa da fotocélula

contra o comprimento de onda da luz incidente. A curva do LDR pode ser observada na figura 4.

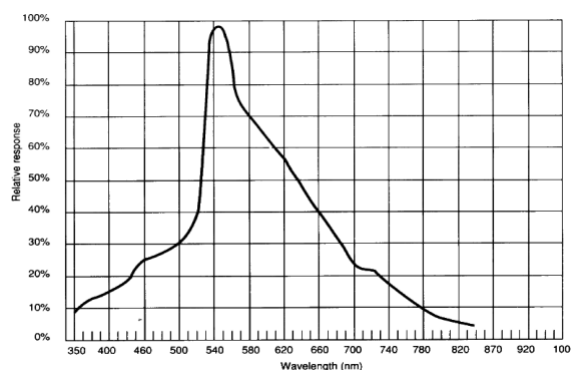


Figura 4. Curva de Resposta Relativa do LDR em função do comprimento de onda da luz incidente

O LDR pode ser implementado em uma série de aplicações. Algumas delas podem ser conferidos na lista a seguir.

- Controle de Exposição de Câmeras
- Máquinas Fotocopiadoras (Xerox)
- Equipamento de colorimetria
- Sistemas de Detecção de Cor
- Controle de Luz Noturna
- Controle de Iluminação Pública
- Sensor de Posição

Vejamos então algumas características elétricas na tabela II.

Tabela II. CARACTERÍSTICAS ELÉTRICAS DO LDR

Parâmetro	Condição	Máx	Típ	Mín	Unidade
Resistência	1000 LUX	-	400	-	Ω
	10 LUX	-	9	-	k Ω
R de escuro	-	-	1	-	M Ω
C de escuro	-	-	3,5	-	pF
Voltagem AC/DC	-	-	320	-	V max
Corrente	-	-	75	-	mA
Potência	-	-	100	-	mW
Temperatura de OP	-	-60	-	+75	°C

2) **Cooler:** Este projeto utilizará um PC cooler genérico para rotacionar os ímãs e realizar o agitador magnético. O equipamento em si é um motor DC de 12 volts e sem necessariamente o fio indicador de rotações por minuto.

3) **Driver de potência ULN2003:** Este driver, presente na figura 5, permite que o microcontrolador realize a operação do cooler eo controle do motor de passo. Ele possui 7 canais de operação que nada mais são um conjunto de portas inversoras que utilizam potência maior que a do MSP430. O driver possui também uma porta de alimentação e outra para o terra. Esta ferramenta consegue liberar no máximo 50V e receber 30V, além de transferir até 500mA de corrente para motores e outros periféricos.

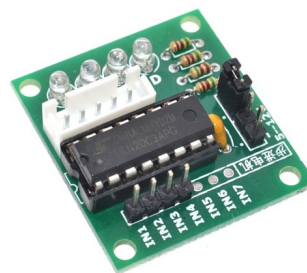


Figura 5. Drive de Potência ULN2003

4) **Driver de Motor de Passo A4989:** O CI A4988 da Allegro, figura 6, é um driver de motor de passo para controle de micropasso com um tradutor embutido para fácil operação. Foi projetado para operar motores de passo bipolares nos modos de passo completo, meio passo, um quarto de passo, um oitavo e um dezesseis avos de passo com a capacidade de operar em tensões até 35V e $\pm 2A$.

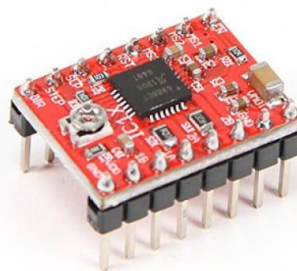


Figura 6. Driver de Motor A4988

A figura 7 nos mostra o diagrama de implementação do CI A4988.

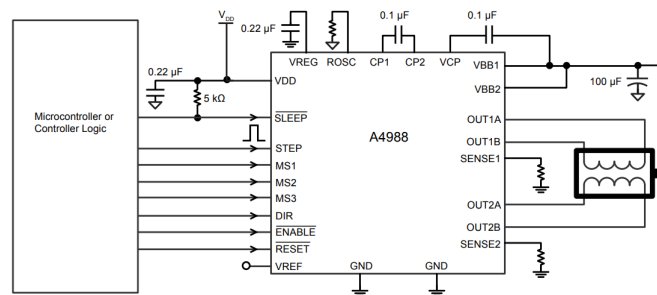


Figura 7. Diagrama de Implementação do Driver de Motor A4988

A Tabela III nos mostra as características de alimentação do driver A4988.

Tabela III. PARÂMETROS DE ALIMENTAÇÃO DE CI E DE CARGA DO DRIVER A4988

Característica	Símbolo	Valor Máximo	Unidade
Alimentação da Carga	V_{BB}	35	V
Corrente de Carga	I_{OUT}	± 2	A
Alimentação do CI	V_{DD}	5	V

A configuração de micropasso é apresentada pela tabela IV.

Tabela IV. PARÂMETROS DE MICROPASSO DO DRIVER A4988

MS1	MS2	MS3	Resolução
L	L	L	Passo Completo
H	L	L	Meio Passo
L	H	L	1/4 de Passo
H	H	L	1/8 de Passo
H	H	H	1/16 de Passo

5) **Motor de Passo 26BYJ-48:** O motor de passo utilizado no nosso projeto é o motor unipolar 26BYJ-48, mostrado na figura 8, com alimentação de 5V e com 64 passos, portanto possui um ângulo de passo de 5.625° . Sua frequência é de 100Hz, possui 4 bobinas magnéticas e sua alimentação é interligada internamente e por isso possui apenas 1 fio de alimentação Vcc.

Este projeto utilizará a configuração de meio passo, a qual necessita energizar as bobinas de 8 formas diferentes para realizar um ciclo completo. Dessa forma, o grupo realizou ensaios para determinar quantos ciclos de 8 passos são necessários para realizar uma volta completa.



Figura 8. Motor de Passo 26BYJ-48

6) **Display LCD 16x2:** Este Dispositivo é utilizado para informar o usuário quanto ao início e finalização do processo mecânico da titulação. O modelo usado é o ITM-16032BSTL da INTECH LCD GROUP, ilustrado na figura 9.

O modelo possui os pinos 1 e 2 para alimentação do LED BACKLIGHT+ e LED BACKLIGHT- respectivamente. Os pinos 3 e 4 são a alimentação do display. O pino 5 controla do contraste do display. O pino 6 é o RS ou Register Select. O pino 7 é o Read/Write select e o mesmo deve ser aterrado. O pino 8 é o pino de Enable do Read/Write. Os pinos 9 a 16 são os DATA BUS. Para este projeto, a comunicação será feita por nibble, então apenas 4 pinos serão usados e estes são os pinos 13 a 16.



Figura 9. LCD ITM-16032BSTL

7) **Ligação com a MSP430:** A figura 10 nos mostra um conceito de ligação entre os periféricos (sensores e motores) com a MSP430G2553. Espera-se que sejam usados 14 portas da MSP430. No ponto de controle anterior, esperava-se que fossem usadas 22 portas, então, com o aperfeiçoamento do projeto, foram reduzidas 8 portas.

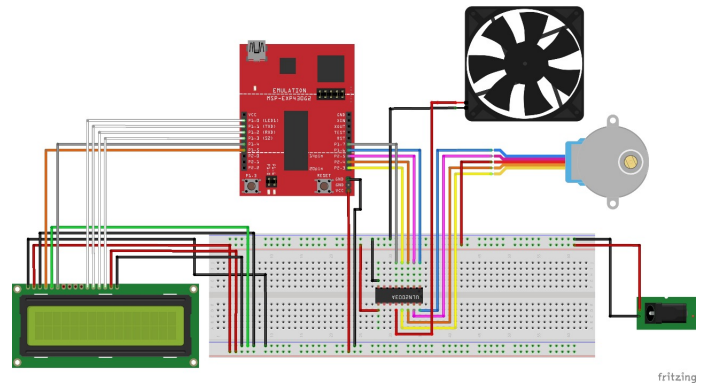


Figura 10. Ligação dos periféricos com a MSP430

D. Teste de detecção do ponto de virada pelo LDR

Atualmente está provado que a cor luz branca (diferente do pigmento branco) é a soma de todas as luzes do espectro visível. Este conceito foi afirmado primeiramente por Isaac Newton, através do experimento do prisma, onde a luz, ao passar pelo prisma, se decompõe em uma faixa de cores, como nos mostra a figura 11.

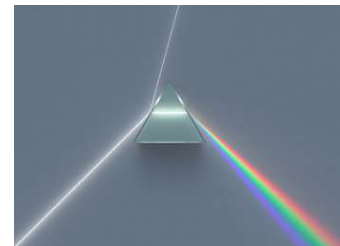


Figura 11. Luz branca passando pelo prisma e se decompondo nas cores do espectro visível

No começo do século XX, o físico alemão Max Planck (1858-1947), provou, através de sua teoria de radiação de corpo negro que a luz branca é a soma de todas outras luzes-cor, como nos mostra a equação 1.

$$E(\lambda) = \int_{400nm}^{700nm} \frac{2\pi h \lambda^3}{c^2 (e^{\frac{h\lambda}{kT}} - 1)} d\lambda \quad (1)$$

Onde:

- E - emitância espectral
- h - Constante de Planck [$6,62 \cdot 10^{-34} m^2 kg/s$]
- λ - comprimento de onda [nm]
- e - contante de Euler [adimensional]
- k - número de onda [rad]
- T - temperatura [K]
- c - velocidade da luz $\approx 2,98 \cdot 10^8$ m/s

Tendo em vista estas informações, podemos afirmar que quando a luz branca passa por um objeto translucido colorido, todas as outras cores luzes são filtradas, exceto a cor luz referente ao objeto translucido. Isso significa que a emitância espectral diminui e portanto, a quantidade de luz que chega no observador é menor que a quantidade que chegaria se não existisse esse filtro.

Essas informações é crucial para que o teste de detecção do ponto de virada da titulação. O esquemático da figura 12' nos mostra como o funcionamento da detecção de ponto de viragem deverá funcionar.

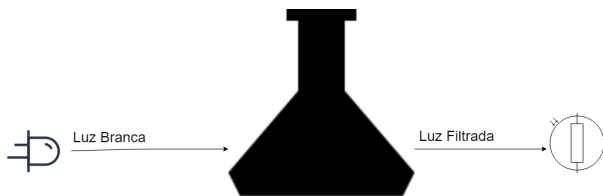


Figura 12. Diagrama de Funcionamento da Detecção do Ponto de Viragem pelo LDR

De acordo com o diagram da figura 12, um LED de cor branca está posicionado em um dos lados do erlenmeyer com a solução incolor de NaOH. Uma vez que o ponto de virada é atingido, a solução agora presente no erlenmeyer funciona como um filtro cor-de-rosa, impedindo que todas as outras luzes-cor cheguem ao LDR, fazendo com que a resistência da célula aumente.

Para que a MSP430 saiba que o ponto de virada ocorreu, será implementado um divisor de tensão, como nos mostra a figura II-D.

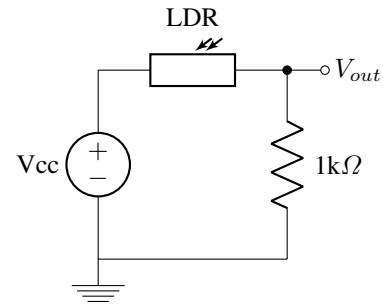


Figura 13. Circuito Divisor de Tensão do LDR

O MSP430 usará seu conversor analógico-digital de 10bits embutido, o ADC10 para que o mesmo seja capaz de enviar a informação de parada para os outros periféricos.

E. Descrição de Software

1) **Diagrama de Blocos:** A figura 14 ilustra o diagrama de blocos do software a ser desenvolvido.

Vimos nela que, enquanto o botão de início não é pressionado, a programa fica preso no estado de espera. Assim que é pressionado o botão de início, o mesmo faz o programa entrar no estado que liga o misturador, setando a flag mixer para nível alto.

Na função que liga o misturador o programa então espera 2s para que a flag início seja setada. Isto acontece para que o mixer tenha tempo hábil para iniciar e estabilizar na velocidade de rotação. Com a flag início setada, abrimos a torneira da bureta dando uma quantidade "n" de passos para que o gotejamento aconteça. Este número de passos ainda será definido.

Assim, o LDR espera até que o ponto de viragem seja alcançado. Uma vez que o ADC10 detectou mudança na voltagem do divisor de tensão do LDR, a flag detecção é setada e assim o programa manda instruções para fechar a bureta e desligar o misturador, da mesma forma que fez para ligá-los.

Para finalizar, o LCD informará o usuário quando o mesmo poderá repetir o procedimento.

2) **Teste do Motor de Passo:** O programa primeiramente define as saídas da porta P1 como:

- P1.0 para a bobina 1
- P1.1 para a bobina 2
- P1.2 para a bobina 3
- P1.3 para o botão da MSP430
- P1.4 para a bobina 4

Como variáveis globais foram definidos o inteiro sem sinal $i = 0$ e um inteiro direção = 0;

O programa define também uma variável para todas as bobinas, chamada COILS.

O clock do sistema é definido em 1MHz. O programa define P1.3 como entrada e as demais ports de P1 como saídas. Também é habilitado o resistor de pull-up do botão. O programa também utiliza de interrupção do botão.

O timer A é configurado utilizando o SMCLK dividido por 1 no modo up e que habilita interrupção do timerA.

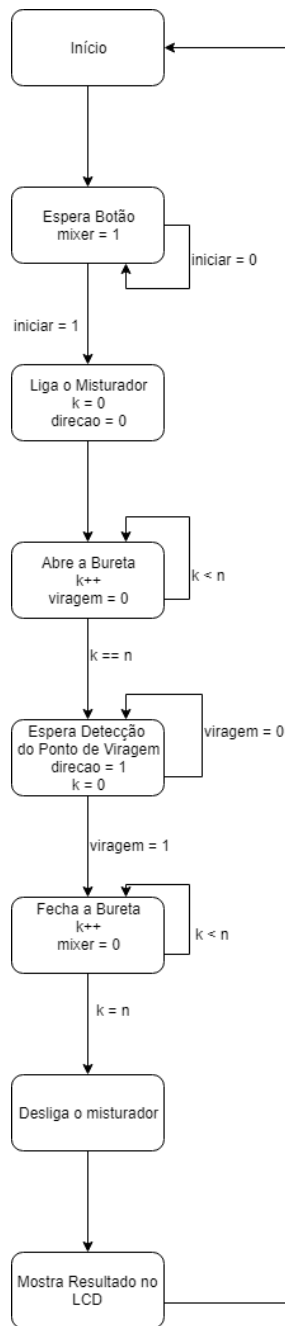


Figura 14. Diagrama de Blocos do Software do Produto

Na função da interrupção do timerA, a variável global *i* realiza o controle de qual bobina será energizada. Se *i* é zero, a bobina 1 é energizada, as outras bobinas não são e se a direção = 0, *i* = 7, senão *i*++. Se *i* = 7 as bobinas 1 e 4 são energizadas, enquanto as demais não são e se direção = 0, *i* é decrementado, senão *i*=0. Se *i* = 6 apenas a bobina 4 é energizada e se direção for = 0, *i*++, senão *i*-. Assim, seguindo a lógica de energizar a bobina *n*, no próximo passo, energiza a bobina *n* e a bobina *n*±1, até que *n* <= 1. A flag direção

seleciona se a próxima bobina energizada será a bobina *n*-1 ou a bobina *n*+1.

Na função de interrupção do botão, se o botão é pressionado, a flag de direção é invertida, ou seja, se era 0, recebe 1, se era um recebe 0.

3) **Misturador:** O mesmo programa que liga o motor de passo, também liga o misturador, com apenas uma diferença. Enquanto o motor de passo tem uma quantidade limitada de passos e gira tanto para um lado quanto para o outro, o cooler fica girando até que a flag de detecção seja setada.

4) **Display LCD 16x2:** Para economia de portas, o modo de comunicação escolhido foi o de 4 bits. Para a comunicação via software, durante a pesquisa sobre o funcionamento do display, foi achada uma biblioteca pronta em C que realiza a comunicação de forma eficaz.

Tendo em vista que o projeto tem como objetivo servir a comunidade acadêmica da FGA, o mesmo deverá ser o mais intuitivo possível. Portanto, o display LCD desempenhará o papel importantíssimo de guiar o usuário nas etapas do processo. De acordo com a figura 15, podemos entender como o LCD servirá de apoio ao usuário.

A porta P1 foi setada inteiramente como saída, ou seja, $P1DIR = 0xFF$; e todos os pinos foram resetados, ou seja, $P1OUT = 0x00$; . Na porta P2, o bit 3, referente ao botão na MSP430FR2433 foi setado como entrada, ou seja $P2DIR \& \sim BTN$; . O Resistor de pull-up/down foi habilitado com $P2REN$ or $P2REN, BTN$; e foi setado como resistor de pull-up a partir de $P2OUT$ or $P2REN, BTN$; .

Então o LCD é inicializado com `lcd16x2_Init()`; O programa então realiza a escrita no LCD de uma string e então, se o usuário apertar o botão, o programa realiza outra escrita no LCD e fica preso no laço infinito.

5) **Detecção de mudança de cor com o LDR:** O programa define o bit1 para a entrada de referência do conversor AD e define o LED1 como o bit0 e o LED2 como o bit6. Existe também uma variável que seta os dois leds ao mesmo tempo, ou seja $LEDS = (LED1|LED2)$.

Na função main, desabilitamos o watchdog timer e setamos o clock do sistema para 1MHz, ou seja, $BCSCTL1 = CALBC1_1MHZ$; e $DCOCTL = CALDCO_1MHZ$; Então os leds são setados para a porta P1 como saída.

Para a configuração do ADC10, usaremos como a fonte de referência $V^+ = V_{cc}$ e $V^- = V_{ss}$, usaremos também o 4 períodos de clock para fazer aproximações sucessivas com o conversor ligado o tempo todo. Usaremos para a fonte de conversão a porta P1.1. Selecionamos a entrada analógica A1, selecionamos a divisão de clock por 1, escolhemos o subsystem master clock como a fonte de clock, usamos a conversão única por canal único e toda vez que uma conversão é solicitada o bit referente ao início da conversão, $ADC10SC$ será setado.

Entramos então em um loop infinito, em que iniciamos uma conversão, esperamos a mesma ficar pronta e se a voltagem no LDR for menor que o threshold do ponto de viragem, o LED1 pisca, senão, ele não pisca.

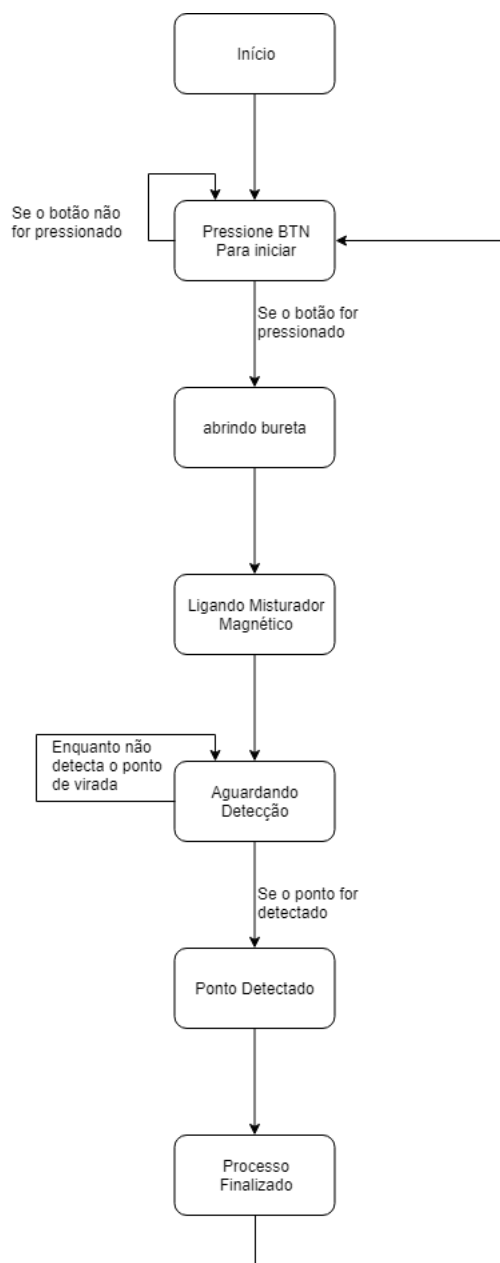


Figura 15. Diagrama do Display LCD 16x2

III. RESULTADOS E ANÁLISES

A. Objetivos

O objetivo geral se baseia na construção de um produto que realize a titulação automática das soluções ácidas e alcalinas a fim de auxiliar os estudantes, técnicos e professores do laboratório de química da UnB Gama e oferecer alta precisão durante o gotejamento. Ademais, o projeto busca realizar as funcionalidades dos aparelhos de alto custo do mercado utilizando periféricos baratos.

B. Resultados Esperados

A partir das premissas do projeto, o grupo pretende:

- Realizar a abertura controlada da torneira da bureta;
- Misturar as soluções ácida e alcalina através de um agitador magnético;
- Detectar o ponto de viragem através de um sensor de cor;
- Informar os usuários quanto ao início e o término do procedimento de titulação.

C. Comunicação - Componentes e MSP430G2553

1) **Display LCD 16x2:** Durante as pesquisas sobre o funcionamento e comunicação com microcontroladores, foi encontrada uma biblioteca pronta para o MSP430FR2433, a "lcd16x2_msp43x.h", que realiza comunicação de maneira eficaz. Sua vantagem é que ela não é para o Energia e sim para a linguagem C.

A partir disso, foi montada uma sequência que mostra como o LCD ajudará o usuário, apresentando as frases de instrução para o usuário.

2) **LDR para Detecção do Ponto de Viragem:** O LDR será o substituto definitivo para o sensor TCS3200, uma vez que o fotoreistor se mostrou eficiente na detecção do ponto de viragem de acordo com as premissas teóricas apresentadas na seção de Desenvolvimento.

3) **Motores:** O motor de passo entrou em funcionamento através da plataforma Energia e realizou o movimento com perfeição. E evidenciou que podemos configurar a direção de rotação, a velocidade e o torque do motor adquirido, o 26BYJ-48.

Ademais, o motor entrou em funcionamento agora pelo Code Composer Studio na linguagem C, assim como está descrito no apêndice. O teste do motor implementou a rotação com direção controlada pelo botão da porta P1.3 e com velocidade controlada pela TACCR0 e pela frequência do SMCLK, pois eles ditam a frequência em que a interrupção do TA acontece. É importante comentar também que o ensaio sobre a quantidade de ciclos de 8 passos necessários para completar 1 volta completa resultou em 4112 passos ($64 * 64 + 16$) e isto implica em uma resolução de 0,07 graus por passo. Desta forma, ficou evidenciado que foi uma boa escolha o tipo de movimentação por "Half-step", pois está com grande resolução e torque mais forte, duas características essenciais para a implementação do projeto.

4) **Cooler:** O Cooler foi implementado com sucesso através do modo de captura do Timer A da porta P1.

IV. CONCLUSÕES

Dado que o objetivo do projeto é desenvolver um produto que solucione um problema do laboratório de Química na UnB Gama, é possível notar que o mesmo se encontra bem embasado e explica de maneira clara os objetivos e requisitos.

Pode-se destacar que, enquanto a comunicação dos periféricos com o microcontrolador, o display LCD e os motores foram bem sucedidos nos testes. O sensor de cor não comunicou com a MSP430, porém houve sucesso na comunicação com o Arduino Uno.

Para o próximo ponto de controle, espera-se que todos os periféricos se comuniquem com a MSP430 e haja um protótipo funcional do produto final.

REFERÊNCIAS

- [1] TCS3200 Datasheet - Texas Advanced Optoelectronic Solutions - <https://www.mouser.com/catalog/specsheets/TCS3200-E11.pdf>
- [2] ITM-1602BSTL - INTECH LCD GROUP - Display LCD 16x2 - http://www.intech-lcd.com/image/Character_LCM/1602b-c.pdf
- [3] COLOR DETECTION DEVICE using MSP430 launch pad - <https://www.youtube.com/watch?v=zj8cD8tLFYo>
- [4] LCD 16x2 with MSP430 TUTORIAL - <https://www.youtube.com/watch?v=P3i4YHqJeMs>
- [5] Essential Magnetic Stirrer Terms - <https://www.catscientific.com/magnetic-stirrer-glossary-of-terms/>
- [6] Magnetic Stirrers - http://www.prosense.net/files/Magnetic_Stirrers.pdf
- [7] Titulação - Brasil Escola - <https://brasilecola.uol.com.br/quimica/titulacao.htm>
- [8] OLIVEIRA, Leonardo Gabriel de ; FERREIRA, Marcelo Sutil , SOUZA, Carlos Marques (orientador). **Misturador magnético digital de fluidos**. Revista Tecnológica da FATEC-PR, v.1, n.6, p. 204 - 226, jan./dez., 2015.
- [9] UnB Gama - Graduação - <https://fga.unb.br/unb-gama/graduacao>
- [10] Química Geral Teórica - <https://matriculaweb.unb.br/graduacao/disciplina.aspx?cod=114626>

APÊNDICE

CÓDIGO DO LCD

```
#include <msp430.h>
#include <stdint.h>
#include "lcd16x2_msp43x.h"

#define BTN BIT3

void main(void)
{
    WDTCTL = WDTPW + WDTHOLD;
    PM5CTL0 &= ~LOCKLPM5;

    P1DIR = 0xFF; //Todas as portas da P1
    ↪ como saída
    P1OUT = 0x00;
    P2DIR &= ~BTN; //Botao setado como
    ↪ entrada
    P2REN |= BTN; //Habilita o resistor de
    ↪ pull up/down
    P2OUT |= BTN; //Coloca como pull-up

    while(1) //Loop infinito
    {
        lcd16x2_Init(); //limpa o LCD
        lcd16x2_SetPosition(0,0); //
        ↪ seleciona a primeira linha
        lcd16x2_PrintString("Pressione BTN")
        ↪ ; //escreve
        lcd16x2_SetPosition(1,0); //
        ↪ seleciona a segunda linha
```

```
lcd16x2_PrintString("Para iniciar");
    ↪ //escreve

while(P2IN&BTN); //espera o botao
    ↪ ser pressionado
while((P2IN & BTN) == 0); //espera
    ↪ soltar o botao

lcd16x2_Init(); //limpa o LCD
lcd16x2_SetPosition(0,0); //
    ↪ seleciona a primeira linha
lcd16x2_PrintString(" abrindo bureta
    ↪ "); //escreve

while(P2IN&BTN); //espera o botao
    ↪ ser pressionado
while((P2IN & BTN) == 0); //espera
    ↪ soltar o botao

lcd16x2_Init();
lcd16x2_SetPosition(0,0);
lcd16x2_PrintString("Ligando mistura
    ↪ -");
lcd16x2_SetPosition(1,0);
lcd16x2_PrintString("dor magnetico")
    ↪ ;

while(P2IN&BTN); //espera o botao
    ↪ ser pressionado
while((P2IN & BTN) == 0); //espera
    ↪ soltar o botao

lcd16x2_Init();

lcd16x2_SetPosition(0,0);
lcd16x2_PrintString("Aguardando");

lcd16x2_SetPosition(1,0);
lcd16x2_PrintString(" Deteccao");

while(P2IN&BTN); //espera o botao
    ↪ ser pressionado
while((P2IN & BTN) == 0); //espera
    ↪ soltar o botao

lcd16x2_Init();
lcd16x2_SetPosition(0,0);
lcd16x2_PrintString("Ponto Detectado
    ↪ ");

while(P2IN&BTN); //espera o botao
    ↪ ser pressionado
while((P2IN & BTN) == 0); //espera
    ↪ soltar o botao

lcd16x2_Init();
lcd16x2_SetPosition(0,0);
```

```

    lcd16x2_PrintString(" Processo");
    lcd16x2_SetPosition(1,0);
    lcd16x2_PrintString(" Finalizado");

    while(P2IN&BTN); //espera o botao
        ↳ ser pressionado
    while((P2IN & BTN) == 0); //espera
        ↳ soltar o botao
}
}

```

CÓDIGO DO MOTOR DE PASSO E COOLER

```

#include <msp430g2553.h>
#include <legacymsp430.h>

#define COIL1 BIT0
#define COIL2 BIT1
#define COIL3 BIT2
#define BTN BIT3
#define COIL4 BIT4
#define COOLER BIT6
#define COILS (COIL1 + COIL2 + COIL3 +
    ↳ COIL4)
#define Speed_1 1000
#define Speed_2 5000
#define Speed_3 10000
#define Speed Speed_3
#define Dcycle 50

/*
    Frequencias: 100Hz 20Hz 1Hz 0,5Hz
    Tloop: 5 25 500 1000
*/

volatile unsigned int i = 0;
int Direcao = 0;
volatile unsigned int count = 0;

void main (void)
{
    WDTCTL = WDTPW + WDTOLD;

    BCSCTL1 = CALBC1_1MHZ;
    DCOCTL = CALDCO_1MHZ;

    P1DIR &= ~BTN; // Habilita resistor de
        ↳ pull-up
    P1REN |= BTN; // Habilita resistor de
        ↳ pull-up
    P1OUT |= BTN; // Define estado inicial
        ↳ do port1

    P1DIR |= COILS;
    P1OUT &= ~COILS;

//-----
    P1DIR |= COOLER;

```

```

    P1SEL |= COOLER;
    P1SEL2 &= ~(COOLER);
    P1OUT &= ~(COOLER);

//-----

    P1IES |= BTN; // Definicao da borda
        ↳ subida/descida
    P1IE |= BTN; // Habilitando a
        ↳ interrupcao p/ P1.3
    P1IFG = 0x00; // Registrador referente
        ↳ a flag

    TA0CTL = TACLK;
    TA0CTL = MC_0;
    TA0CCR0 = Speed-1;
    TA0CTL = TASSEL_2 + MC_1 + ID_0 + TAIE;

//-----

    TA0CCR1 = Speed/(100/Dcycle)-1;
    TA0CCTL0 &= ~CAP;
    TA0CCTL1 = OUTMOD_7;

//-----

    _BIS_SR(GIE + LPM0_bits);

}

interrupt (TIMER0_A1_VECTOR) TA0_ISR(void)
{
    switch(i){
        case(0):
            P1OUT = (P1OUT|COIL1)&(~(COIL2+
                ↳ COIL3+COIL4));
            if (Direcao) i++;
            else i=7;
            break;
        case(1):
            P1OUT = (P1OUT|(COIL1+COIL2))&(~(
                ↳ COIL3+COIL4));
            if (Direcao) i++;
            else i--;
            break;
        case(2):
            P1OUT = (P1OUT|COIL2)&(~(COIL1+
                ↳ COIL3+COIL4));
            if (Direcao) i++;
            else i--;
            break;
        case(3):
            P1OUT = (P1OUT|(COIL2+COIL3))&(~(
                ↳ COIL1+COIL4));
            if (Direcao) i++;
            else i--;
            break;
        case(4):
            P1OUT = (P1OUT|COIL3)&(~(COIL1+

```

```

        ↪ COIL2+COIL4));
    if (Direcao) i++;
    else i--;
    break;
case(5):
    P1OUT = (P1OUT|(COIL3+COIL4))&(~(
        ↪ COIL1+COIL2));
    if (Direcao) i++;
    else i--;
    break;
case(6):
    P1OUT = (P1OUT|COIL4)&(~(COIL1+
        ↪ COIL2+COIL3));
    if (Direcao) i++;
    else i--;
    break;
case(7):
    P1OUT = (P1OUT|(COIL1+COIL4))&(~(
        ↪ COIL2+COIL3));
    if (Direcao) i=0;
    else i--;
    break;
}

TA0CTL &= ~TAIFG;
}

interrupt(PORT1_VECTOR) Interrupcao_P1(
    ↪ void)
{
    Direcao ^= 1; // Troca a direcao
    P1IFG = 0x00; // Zera a flag
}

```