

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL  
ESCOLA DE ENGENHARIA  
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

CARLOS AUGUSTO DE ANDRADE REVERBEL

**Desenvolvimento do hardware para um *Network Attached Storage Device***

Porto Alegre

2018

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL  
ESCOLA DE ENGENHARIA  
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

CARLOS AUGUSTO DE ANDRADE REVERBEL

**Desenvolvimento do hardware para um *Network  
Attached Storage Device***

Projeto de Diplomação apresentado ao Departamento de Engenharia Elétrica da Escola de Engenharia da Universidade Federal do Rio Grande do Sul, como requisito parcial para Graduação em Engenharia Elétrica

Orientador: Prof. Dr. Ivan Muller

Porto Alegre

2018

CARLOS AUGUSTO DE ANDRADE REVERBEL

**Desenvolvimento do hardware para um *Network Attached Storage Device***

Projeto de Diplomação apresentado ao Departamento de Engenharia Elétrica da Escola de Engenharia da Universidade Federal do Rio Grande do Sul, como requisito parcial para Graduação em Engenharia Elétrica

---

**Prof. Dr. Ivan Muller**  
Orientador - UFRGS

---

**Prof. Dr. Ály Ferreira Flores Filho**  
Chefe do Departamento de Engenharia Elétrica (DELET) - UFRGS

Aprovado em 02 de Julho de 2018.

BANCA EXAMINADORA

---

**Prof. Dr. Hamilton Duarte Klimach**  
UFRGS

---

**Prof. Dr. Tiago Roberto Balen**  
UFRGS

# Resumo

Este trabalho visa apresentar o desenvolvimento do Hardware para um *Network Attached Storage* (NAS), apresentando suas especificações, escolha de componentes, layout de PCB e testes. Realizando um estudo dos dispositivos NAS existentes no mercado, percebe-se uma lacuna de mercado para um dispositivo que ofereça simplicidade de instalação e uso intuitivo. Com esse estudo foram escolhidas as especificações do protótipo e o processador central do circuito (i.MX6D). Pelas dimensões físicas definidas para o dispositivo, o circuito foi dividido em 2 placas, uma de 2 camadas, com os conectores, LEDs e HD e outra com 8 camadas, com o i.MX6D, memória SDRAM DDR3, regulador de tensão e outros periféricos. O projeto do hardware foi concluído e grande parte deste já foi testado e validado, restando a placa de 8 camadas, que está em processo de manufatura.

**Palavras-chave:** NAS; PCB multicamada; i.MX6; ARM Cortex-A9; Design de hardware; DDR3;

# Abstract

This work intends to present the Hardware development for a Network Attached Storage (NAS), including specs, component selection, PCB layout and tests. After researching existing NAS solutions a market gap was found for a "plug-and-play" device. Based on the existing NAS devices the specifications for a prototype were developed and its main component was chosen to be the i.MX6D system-on-chip. Because of mechanical constraints the circuit was divided in 2 printed circuit boards, a 2-layer board with the connectors, LEDs and hard drive and an 8-layer one, carrying the i.MX6D, SDRAM DDR3 memory, voltage regulator, etc. The hardware was fully designed and most of it was tested and validated, except for the 8-layer board that could not be tested because of its high fabrication cost.

**Keywords:** NAS; multi-layer PCB; i.MX6; ARM Cortex-A9; Hardware design; DDR3;

# Lista de Figuras

Figura 1 – Simples aplicação de um dispositivo NAS . . . . .	13
Figura 2 – Tipos de linha de transmissão para PCB usados no projeto . . . . .	16
Figura 3 – Etapas da realização de uma placa de 2 camadas . . . . .	19
Figura 4 – Exemplo de <i>stack-up</i> de uma placa de 4 camadas . . . . .	20
Figura 5 – Exemplo de uso de <i>teardrops</i> . . . . .	20
Figura 6 – Diferentes tipos de vias em uma placa de circuito . . . . .	21
Figura 7 – Diferença entre as topologias <i>T-Branch</i> e <i>Fly-by</i> . . . . .	23
Figura 8 – Exemplo de placa realizada com topologia <i>T-Branch</i> - 4 chips, 64bits .	23
Figura 9 – Exemplo de placa realizada com topologia <i>Fly-by</i> - 4 chips, 64bits . . .	24
Figura 10 – Fluxo de corrente em um conversor <i>boost</i> nos estados $T_{ON}$ e $T_{OFF}$ . . .	24
Figura 11 – Formas de onda de um conversor <i>boost</i> operando em modo de condução contínua . . . . .	25
Figura 12 – Renderização do envólucro do dispositivo . . . . .	27
Figura 13 – Modelo 3D das placas de circuito do protótipo . . . . .	28
Figura 14 – Localização do processador central no produto . . . . .	29
Figura 15 – Exemplo de aplicação do PMIC . . . . .	30
Figura 16 – Localização do PMIC no circuito . . . . .	30
Figura 17 – Esquemático do bloco Circuito de proteção de entrada . . . . .	31
Figura 18 – Localização do circuito de proteção de entrada no produto . . . . .	31
Figura 19 – Esquemático do bloco <i>Driver</i> dos LEDs indicadores . . . . .	32
Figura 20 – Localização do <i>driver</i> dos LEDs no produto . . . . .	33
Figura 21 – Conectores PCIe, SATA e <i>Ethernet</i> . . . . .	34
Figura 22 – Localização dos conectores no produto . . . . .	35
Figura 23 – Esquemático contendo o microprocessador responsável pelo controle dos LEDs . . . . .	35
Figura 24 – Localização do microcontrolador no produto . . . . .	36
Figura 25 – Localização dos demais componentes do produto . . . . .	37
Figura 26 – <i>Layers</i> da Placa Mãe . . . . .	38
Figura 27 – <i>Layers</i> da Placa IO . . . . .	39
Figura 28 – <i>Stack-up</i> da Placa Mãe . . . . .	40
Figura 29 – Design de referência para o i.MX6, <i>SABRE-AI</i> . . . . .	41
Figura 30 – Layout do grupo DRAM_SDCLK . . . . .	43
Figura 31 – Layout do grupo DRAM_ADDR . . . . .	43
Figura 32 – Layout do grupo DRAM_CTRL . . . . .	44
Figura 33 – Layout dos grupos DRAM_BANK0 a DRAM_BANK3 . . . . .	44
Figura 34 – Layout dos grupos DRAM_BANK4 a DRAM_BANK7 . . . . .	45

Figura 35 – Layout das interfaces RGMII e <i>Ethernet</i> . . . . .	46
Figura 36 – Layout das interfaces Flash e SATA . . . . .	46
Figura 37 – Layout dos níveis de tensão com maior capacidade de corrente da Placa Mãe . . . . .	47
Figura 38 – Layout dos níveis de tensão com maior capacidade de corrente da Placa Mãe . . . . .	48
Figura 39 – <i>Driver</i> dos LEDs e comparação com conversor <i>boost</i> . . . . .	49
Figura 40 – Nó com maior derivada de tensão do <i>driver</i> - quanto menor, melhor . .	49
Figura 41 – Malha com maior derivada de corrente do <i>driver</i> - quanto menor, melhor	50
Figura 42 – Um dos 4 braços de corrente do <i>driver</i> . . . . .	51
Figura 43 – Montagem para testes com PMIC e circuito de proteção de entrada . .	52
Figura 44 – Circuito de proteção de entrada na Placa IO . . . . .	53
Figura 45 – Placa desenvolvida para testes com o PMIC . . . . .	53
Figura 46 – Tabela de especificações - MMPF0100F0 e F4 (editado do <i>datasheet</i> ) .	54
Figura 47 – Montagem para teste com 16 LEDs . . . . .	56
Figura 48 – resultados obtidos com 16 LEDs . . . . .	56
Figura 49 – resultados obtidos com 32 LEDs . . . . .	57
Figura 50 – Montagem para teste do código do microcontrolador . . . . .	58
Figura 51 – Interface serial para transmissão de comandos SPI pelo <i>Arduino</i> . . . .	59
Figura 52 – Montagem para teste SATA - sem Placa IO . . . . .	60
Figura 53 – Montagem para teste SATA - com Placa IO . . . . .	61
Figura 54 – Montagem para teste Ethernet - sem Placa IO . . . . .	63
Figura 55 – Montagem para teste Ethernet - com Placa IO . . . . .	63
Figura 56 – Resultados dos testes Ethernet - sem Placa IO . . . . .	64
Figura 57 – Resultados dos testes Ethernet - com Placa IO . . . . .	64

# Lista de Tabelas

Tabela 1 – Vantagens e desvantagens dos tipos de linhas de transmissão em PCB .	17
Tabela 2 – Regras de design DDR3 - grupo DRAM_SDCLK . . . . .	41
Tabela 3 – Regras de design DDR3 - grupo DRAM_ADDR . . . . .	42
Tabela 4 – Regras de design DDR3 - grupo DRAM_CTRL . . . . .	42
Tabela 5 – Regras de design DDR3 - grupos DRAM_BANK[7:0] . . . . .	42
Tabela 6 – Medidas de comprimento de cada grupo da interface DDR3 . . . . .	45
Tabela 7 – Máximo consumo de corrente dos principais níveis de tensão da Placa Mãe . . . . .	47
Tabela 8 – Resultados - leitura do HD . . . . .	61
Tabela 9 – Resultados - escrita do HD . . . . .	61
Tabela 10 – Comparação dispositivos NAS Western Digital . . . . .	68
Tabela 11 – Comparação dispositivos NAS Synology . . . . .	69
Tabela 12 – Dispositivo NAS D-Link . . . . .	69
Tabela 13 – Dispositivo NAS ASUSTOR . . . . .	70
Tabela 14 – Dispositivo NAS Seagate . . . . .	70
Tabela 15 – Comparação dispositivos NAS QNAP . . . . .	71



# Lista de Abreviaturas e Siglas

ARM	<i>Advanced RISC Machine</i>
CAD	<i>Computer Aided Design</i>
DDR3	<i>Double Data Rate 3</i>
EUA	Estados Unidos da América
GB	<i>Gigabyte</i>
Gbps	<i>Gigabits por segundo</i>
GPU	<i>Graphics Processing Unit</i>
HD	<i>Hard Drive</i>
LAN	<i>Local Area Network</i>
LED	<i>Light-emitting diode</i>
MB	<i>Megabyte</i>
Mbps	<i>Megabit por segundo</i>
MT	<i>Megatransfer</i>
MVP	<i>Minimum Viable Product</i>
NAS	<i>Network Attached Storage</i>
PAM	<i>Pulse Amplitude Modulation</i>
PCB	<i>Printed Circuit Board</i>
PD1	Projeto de Diplomação 1
PD2	Projeto de Diplomação 2
PMIC	<i>Power Management Integrated Circuit</i>
RAM	<i>Random Access Memory</i>
RF	Radio frequência
RGMII	<i>Reduced Gigabit Media-Independent Interface</i>

RISC	<i>Reduced Instruction Set Computer</i>
SATA	<i>Serial AT Attachment</i>
SoC	<i>System on Chip</i>
SPI	<i>Serial Peripheral Interface</i>
TB	<i>Terabyte</i>
TEM	<i>Transverse electromagnetic mode</i>
USB	<i>Universal Serial Bus</i>

# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>12</b>
1.1	Sobre NAS	12
1.2	Motivação e descrição do trabalho	12
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>14</b>
2.1	Linhas de transmissão	14
2.1.1	O que são e para que servem linhas de transmissão	14
2.1.2	Modelo e parâmetros de uma linha de transmissão	15
2.1.3	Linhas de transmissão em placas de circuito impresso	16
2.1.4	Par diferencial	17
2.1.5	Resumo de linhas de transmissão	18
2.2	Construção de PCBs multi-camada	19
2.3	Tecnologia SDRAM DDR3	21
2.4	Layout de conversores CC-CC tipo <i>Boost</i>	24
<b>3</b>	<b>ESPECIFICAÇÕES DO PROTÓTIPO FINAL</b>	<b>26</b>
<b>4</b>	<b>DESENVOLVIMENTO DO CIRCUITO</b>	<b>28</b>
4.1	Processador central	28
4.2	Regulador de Tensão	29
4.3	Circuito de proteção de entrada	31
4.4	<i>Driver</i> dos LEDs indicadores	32
4.5	Conectores	33
4.6	Microcontrolador	35
4.7	Demais componentes	36
<b>5</b>	<b>LAYOUT DAS PLACAS DE CIRCUITO</b>	<b>38</b>
5.1	<i>Stack-up</i> da Placa Mãe	39
5.2	DDR3 e demais interfaces de alta velocidade da Placa Mãe	40
5.3	Alimentação na Placa Mãe	47
5.4	Interfaces de alta velocidade na Placa IO	48
5.5	<i>Driver</i> dos LEDs na Placa IO	48
<b>6</b>	<b>TESTES DO CIRCUITO</b>	<b>52</b>
6.1	Testes: Circuito de proteção de entrada e PMIC	52
6.2	Testes: <i>Driver</i> dos LEDs	55
6.3	Testes: Interface microcontrolador e i.MX6	57

6.4	Testes: Placa IO - SATA . . . . .	59
6.5	Testes: Placa IO - Ethernet . . . . .	62
7	CONCLUSÃO . . . . .	65
8	PROPOSTA DE TRABALHOS FUTUROS . . . . .	66
	REFERÊNCIAS BIBLIOGRÁFICAS . . . . .	67
	ANEXO A – DISPOSITIVOS NAS EXISTENTES NO MERCADO	68
A.1	Dispositivos NAS - Western Digital . . . . .	68
A.2	Dispositivos NAS - Synology . . . . .	69
A.3	Dispositivos NAS - D-Link . . . . .	69
A.4	Dispositivos NAS - ASUSTOR . . . . .	69
A.5	Dispositivos NAS - Seagate . . . . .	70
A.6	Dispositivos NAS - QNAP . . . . .	70
A.7	Conclusões do estudo . . . . .	71

# 1 Introdução

## 1.1 Sobre NAS

*Network Attached Storage* (NAS) é um computador simples, geralmente sem teclado, mouse ou tela (*headless*) ligado à rede e usado para armazenamento de dados. Contém no mínimo um HD, onde armazena os dados; Microprocessador; Memória RAM e um conector *Ethernet* para poder ser acessado por outros dispositivos na rede.

Diferente das soluções de armazenamento de arquivo baseadas em nuvem (e.g. *Dropbox*, *Google Drive*, etc...), os NAS apresentam como vantagem:

- Maior privacidade dos dados armazenados.

Os dados são armazenados no dispositivo NAS, não em um servidor da empresa de nuvem.

- Maior velocidade de acesso quando na rede local.

Estando na mesma LAN que o dispositivo NAS (maior parte dos casos) tem-se uma menor latência de acesso.

- Ausência de mensalidade.

Os serviços de nuvem oferecem espaço de armazenamento limitado gratuitamente. Para aumentar esse espaço, cobram uma taxa mensal.

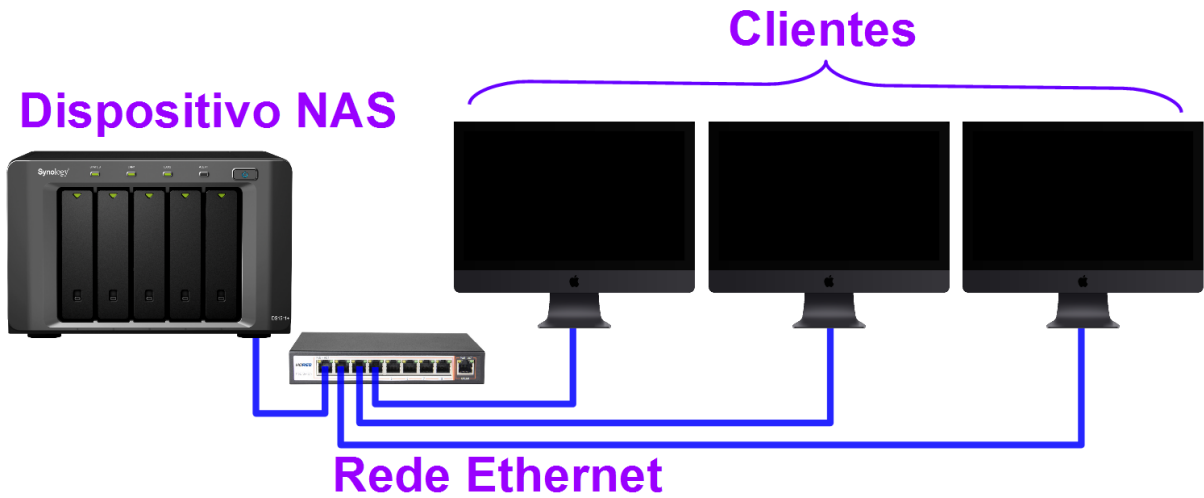
Dispositivos NAS são usados onde precisa-se de um servidor de arquivos de baixo custo e baixa latência, sendo os principais clientes desses mini computadores as pequenas e médias empresas. A Figura 1 apresenta o uso de um NAS em uma rede de computadores.

## 1.2 Motivação e descrição do trabalho

Após um estudo de algumas das várias soluções de NAS existentes hoje no mercado (detalhado no Anexo A), pode-se concluir que a maioria tem como foco o uso empresarial. Esses produtos, quando analisados sob a perspectiva de um usuário doméstico (geralmente leigo em tecnologia), sofrem de parte (ou totalidade) das deficiências apresentadas a seguir:

- Complexidade e (relativa) demora de instalação.
- Interface pouco intuitiva e com muitas funcionalidades e recursos desnecessários.

Figura 1 – Simple aplicação de um dispositivo NAS



- Experiência de acesso diferente para quem está dentro e fora da rede local (dentro da LAN podem ser acessados como dispositivos de rede, mas fora precisam ser acessados por um site ou aplicativo).

A partir desse estudo percebe-se uma lacuna de mercado para um dispositivo NAS que ofereça simplicidade de instalação (dispositivo *plug-and-play*) e uso intuitivo para os mais leigos.

Este documento descreve então as etapas envolvidas no desenvolvimento do Hardware de tal dispositivo, como especificações e restrições; escolha dos componentes; layout da PCB; etc.

## 2 Fundamentação teórica

Para auxiliar a leitura deste trabalho, neste capítulo foram tratados/revisados alguns conceitos pertinentes para se ter uma boa compreensão dos capítulos seguintes (especialmente do Capítulo 5).

Este projeto envolve a realização de placas de circuito multi-camada e com interfaces de alta velocidade como SATA, DDR3 e *Ethernet Gigabit*. Este capítulo inicia então com uma introdução a linhas de transmissão, estruturas capazes de transmitir sinais de alta frequência sem distorcer-los, seguido pela apresentação do processo de fabricação de placas de circuito multi-camadas, pois entendendo como são construídas é possível otimizar o design de uma placa para reduzir custos e minimizar falhas de produção.

Sendo a interface mais complexa desenvolvida nesse projeto, o capítulo segue com uma apresentação da tecnologia SDRAM DDR3 e suas principais topologias de layout em uma placa de circuito. O capítulo termina então apresentando algumas boas práticas de layout para conversores CC-CC (com ênfase no conversor tipo *boost*). Isso é importante pois, como será visto no Capítulo 5, o projeto envolveu o layout de um *driver* de LEDs, o qual se comporta como um conversor tipo *boost*.

Em resumo, este capítulo aborda os seguintes tópicos:

- Linhas de transmissão.
- Construção de PCBs multi-camada.
- Tecnologia SDRAM DDR3.
- Layout de conversores CC-CC tipo *Boost*.

### 2.1 Linhas de transmissão

#### 2.1.1 O que são e para que servem linhas de transmissão

Segundo (WIKIPEDIA, 2018b), uma linha de transmissão pode ser definida como uma estrutura destinada a conduzir correntes de frequências altas o suficiente para que a teoria de circuitos não mais produza resultados satisfatórios, sendo necessário levar em conta a sua natureza ondulatória.

Quando o tamanho de um circuito se torna comparável aos comprimentos de onda dos sinais que nele se propagam, esses sinais podem sofrer reflexão quando encontram descontinuidades, além de poderem perder energia ao longo do caminho por irradiação. A

teoria de linhas de transmissão trata do estudo desses fenômenos e permite a construção de estruturas capazes de transmitir um sinal do ponto A ao ponto B com distorções e perdas em níveis toleráveis.

### 2.1.2 Modelo e parâmetros de uma linha de transmissão

Como este trabalho trata da realização de um circuito digital, as perdas por não idealidade dos materiais podem ser desprezadas e o modelo ideal para linhas de transmissão pode então ser usado (o efeito de dispersão também será desprezado pois as frequências envolvidas são relativamente baixas e o circuito é pequeno).

Uma linha de transmissão ideal pode ser completamente descrita por 3 parâmetros:

- Impedância característica ( $Z_0$ ): razão entre tensão e corrente em qualquer ponto da linha, em  $\Omega$ .
- Velocidade de propagação ( $v_p$ ): velocidade com que um sinal se propaga no interior da linha, em  $m/s$ .
- Comprimento total da linha ( $l$ ), em  $m$ .

Para que não haja reflexão dos sinais, a linha deve ser terminada em suas extremidades por uma resistência com o mesmo valor da sua impedância característica, evitando assim descontinuidades.

Para extrair os parâmetros de uma linha de transmissão pode-se usar o modelo de elementos discretos de circuitos e resolver as equações de telegrafista, que descrevem a propagação dos sinais ao longo da linha. Resolvendo as equações para o caso ideal, vide (POZAR, 1998), tem-se os seguintes resultados:

$$Z_0 = \sqrt{\frac{L}{C}} \quad (2.1)$$

$$v_p = \frac{1}{\sqrt{LC}} \quad (2.2)$$

Onde L e C dependem, no caso ideal, apenas da geometria e do dielétrico que formam a estrutura da linha e são definidos como:

L = indutância série por comprimento unitário, em  $H/m$ .

C = capacitância *shunt* por comprimento unitário, em  $F/m$ .



Vale também ressaltar aqui que, para o caso de uma linha que propaga um sinal em modo transversal eletromagnético (TEM), a velocidade de propagação (vide (POZAR, 1998)) também é dada por:

$$v_p = \frac{c}{\sqrt{\mu_r \epsilon_r}} \quad (2.3)$$

Onde:

$c$  = velocidade da luz no vácuo  $\approx 3.10^8 m/s$ .

$\mu_r$  = permeabilidade magnética relativa.

$\epsilon_r$  = permissividade elétrica relativa (também chamado de constante dielétrica).

Esse resultado é importante pois as análises de linhas de transmissão em placas de circuito assumem geralmente propagação TEM ou *quasi*-TEM.

### 2.1.3 Linhas de transmissão em placas de circuito impresso

Para a transmissão de sinais de alta frequência em placas de circuito existem diversos tipos de linhas de transmissão que podem ser realizados, nesta seção são apresentados apenas 3, por serem os tipos usados durante o projeto. São eles *microstrip*, *stripline* e guia de onda coplanar (*coplanar waveguide*).

A Figura 2 mostra a forma de cada um dos tipos supracitados. A Tabela 1 descreve as principais vantagens e desvantagens de cada um.

Figura 2 – Tipos de linha de transmissão para PCB usados no projeto

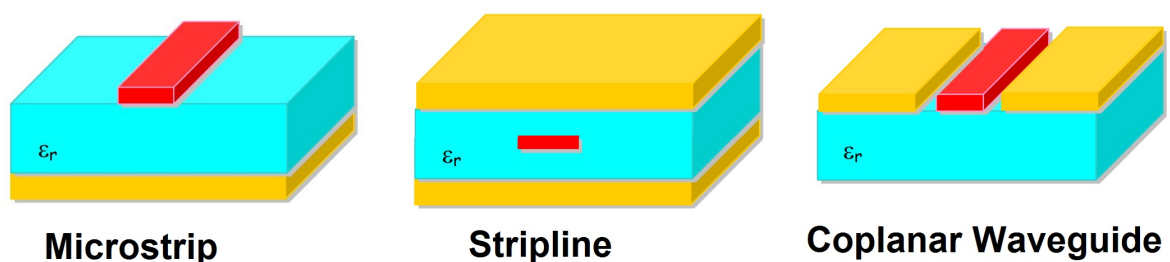


Tabela 1 – Vantagens e desvantagens dos tipos de linhas de transmissão em PCB

	Vantagem	Desvantagem
<b><i>Microstrip</i></b>	fácil de fabricar; mais popular para PCB; fácil de conectar componente;	difícil de analisar; mais dispersiva;
<b><i>Stripline</i></b>	fácil de analisar; menos dispersiva; irradia menos;	mais difícil de fabricar; mais difícil de conectar componentes;
<b><i>Coplanar Waveguide</i></b>	muito fácil de fabricar;	ocupa mais espaço;

Da Equação 2.3 conclui-se que a velocidade de propagação de um sinal em uma placa de circuito é inversamente proporcional à permissividade elétrica relativa ( $\epsilon_r$ ) do dielétrico da placa.

É possível analisar a geometria de cada uma dessas linhas, calcular sua indutância série e capacitância *shunt* por comprimento unitário e finalmente obter sua impedância com a Equação 2.1. Esse processo é, no entanto, bastante complicado, por isso foram desenvolvidas equações pseudo-empíricas que podem ser encontradas facilmente *online* (MICROWAVES101, 2018) para estimar a impedância de cada tipo de linha.

#### 2.1.4 Par diferencial

Um par diferencial é uma estrutura de linha de transmissão formado por duas linhas próximas o suficiente para que haja acoplamento de energia entre elas. Em um par diferencial, cada linha transporta um sinal de igual magnitude e sinais opostos.

Comparado a uma linha de transmissão normal, um par diferencial apresenta como vantagens uma maior imunidade à interferência eletromagnética externa, menor irradiação de ruído eletromagnético, possibilidade de trabalhar com níveis de tensão menores, entre outros.

Um parâmetro importante para um par diferencial é a sua impedância diferencial, que é o valor da impedância que, quando conectada entre as 2 linhas que o compõem, evita reflexões.

Para que 2 linhas de transmissão funcionem como um par diferencial é necessário que, em qualquer segmento infinitesimal do par, a impedância de cada linha seja a mesma e que as correntes sejam de igual módulo mas sinais opostos (também é necessário, é claro, que as linhas estejam próximas uma da outra para permitir o acoplamento de energia).

Respeitando essas condições temos de (BROOKS, 1998) as seguintes relações de tensão e corrente em qualquer segmento infinitesimal do par:

$$V_1 = Z_0.i.(1 - k) \quad (2.4)$$

$$V_2 = -Z_0.i.(1 - k) \quad (2.5)$$

Onde:

$Z_0$  = Impedância característica das linhas, em  $\Omega$ .

$V_1$  = Tensão na linha 1 com relação ao terra, em  $V$ .

$V_2$  = Tensão na linha 2 com relação ao terra, em  $V$ .

$i$  = Módulo da corrente (igual em ambas as linha), em  $A$ .

$k$  = Fator de acoplamento (depende da geometria e proximidade das linhas).

Analisando as equações 2.4 e 2.5, percebe-se que a relação entre tensão e corrente em cada uma das linhas, quando operam como par diferencial, é dada por:

$$\frac{V_1}{i} = Z_{odd} = Z_0.(1 - k) \quad (2.6)$$

Onde  $Z_{odd}$  é a impedância de modo ímpar, que é o valor de impedância com a qual deve-se terminar cada uma das linhas (com relação ao terra) para evitar reflexões. É fácil concluir então que a impedância diferencial do par ( $Z_{diff}$ ) é o dobro da impedância de modo ímpar.

$$Z_{diff} = 2.Z_{odd} \quad (2.7)$$

### 2.1.5 Resumo de linhas de transmissão

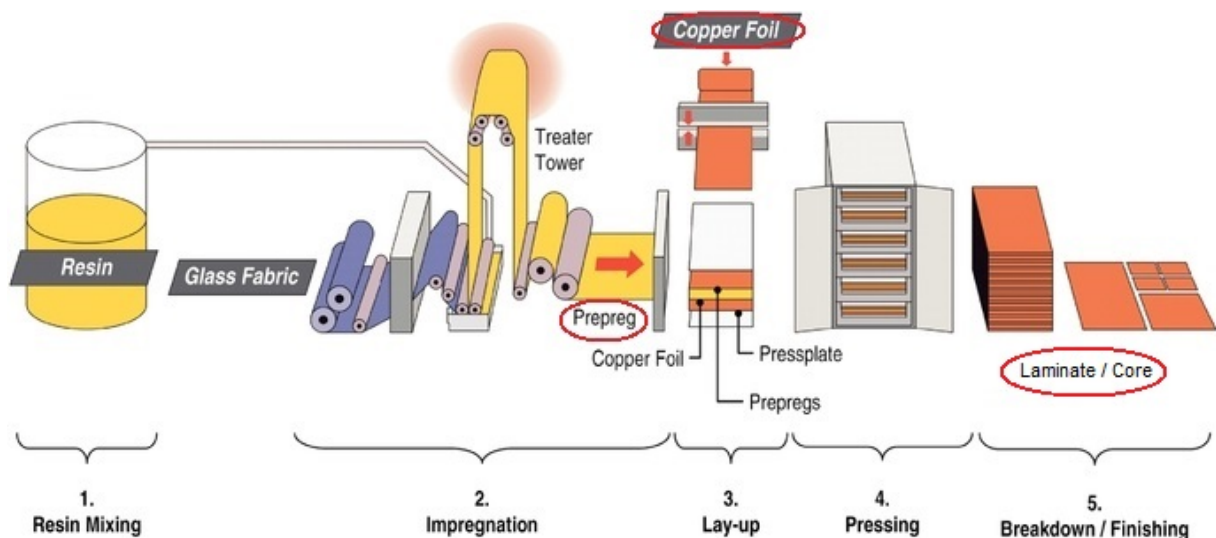
- A impedância característica de uma linha é a razão entre tensão e corrente em qualquer ponto da linha, em  $\Omega$ .
- Para evitar reflexões deve-se terminar uma linha de transmissão em seus 2 extremos com uma resistência do mesmo valor da sua impedância característica.
- Da equação 2.1, quanto maior a capacitância de uma estrutura (maior constante dielétrica, maior largura da trilha ou menor espessura do dielétrico), menor será a sua impedância.
- Da equação 2.3, quanto maior a constante dielétrica do meio onde a estrutura está inserida, menor será a sua velocidade de propagação.

- Um par diferencial é formado por duas linhas de transmissão próximas o suficiente para que haja acoplamento de energia entre elas.
- Para evitar reflexões deve-se terminar um par diferencial em seus 2 extremos com uma resistência do mesmo valor da sua impedância diferencial. Essa resistência deve ser conectada entre cada uma das linhas.

## 2.2 Construção de PCBs multi-camada

É interessante começar explicando como é feita uma placa de 2 camadas para em seguida abordar a construção de uma multi-camada. A Figura 3 apresenta as etapas envolvidas na fabricação de uma placa virgem de 2 camadas.

Figura 3 – Etapas da realização de uma placa de 2 camadas



Como pode ser visto na imagem, o processo inicia pela impregnação de um tecido de fibra de vidro por uma resina, formando o chamado *prepreg*. Em seguida, folhas de cobre, comumente abreviado para *foil*, são inseridas acima e abaixo do *prepreg*. Esse sanduíche é então aquecido e prensado para que a resina do *prepreg* derreta e cole as *foils*, criando assim um laminado que pode ser usado na fabricação de placas de circuito de duas camadas ou multi-camadas. Quando usados na construção de placas multi-camada, esses laminados são geralmente chamados de *core*.

Tendo o laminado, o processo de fabricação de uma placa de circuito de 2 camadas é bem simples. O processo inicia pela realização de todos os furos metalizados, como vias e *pads* de componentes *through-hole*. Em seguida é realizada a transferência do layout do circuito para a placa usando técnicas como transferência térmica ou (geralmente) fotolitografia. O circuito é então protegido contra oxidação através da deposição de uma máscara de solda e os contatos de solda recebem um acabamento de superfície, onde é

depositado uma liga de metais que oxidam menos, como estanho e chumbo, níquel e ouro, prata, etc. Em seguida são feitas a serigrafia e os furos e chanfros não-metalizados da placa.

A fabricação de uma placa multi-camada segue um processo semelhante, mas com mais etapas e materiais. Para a fabricação de uma placa multi-camada primeiro deve-se definir o seu *stack-up*, que nada mais é do que o arranjo de como a placa será construída, i.e. quantas camadas ela terá e qual o material e espessura de cada camada. A Figura 4 apresenta um exemplo de *stack-up* para uma placa de 4 camadas.

Figura 4 – Exemplo de *stack-up* de uma placa de 4 camadas

L1	Copper Foil	36 um - Copper
	PrePreg	0.17 mm - FR408 PrePreg
L2	Copper	36 um - Copper
	Core	1.2 mm - FR408 Core
L3	Copper	36 um - Copper
	PrePreg	0.17 mm - FR408 PrePreg
L4	Copper Foil	36 um - Copper

Neste caso primeiro é feita a transferência do circuito para as camadas L2 e L3, em seguida as camadas são aquecidas e prensadas, colando assim as camadas L1 e L4 ao *core*. A partir deste ponto o processo é idêntico ao de uma placa de 2 camadas.

Ao analisar cada uma das etapas descritas acima é possível identificar algumas boas práticas que possibilitam a redução de custos de fabricação e ajudam a minimizar defeitos de fabricação, por exemplo, usar o mesmo diâmetro de furo para todas as vias e *pads* de componentes *through-hole* para minimizar o tempo de furação; Usar *teardrops* (usado no encontro de trilhas com vias, como pode ser visto na Figura 5) para proteger a placa contra tolerâncias na furação; Só usar vias especiais, como *microvias*, vias cegas e vias enterradas, quando forem realmente necessárias (exemplos de cada uma delas na Figura 6); Manter a maior largura e espaçamento das trilhas possíveis; etc.

Figura 5 – Exemplo de uso de *teardrops*

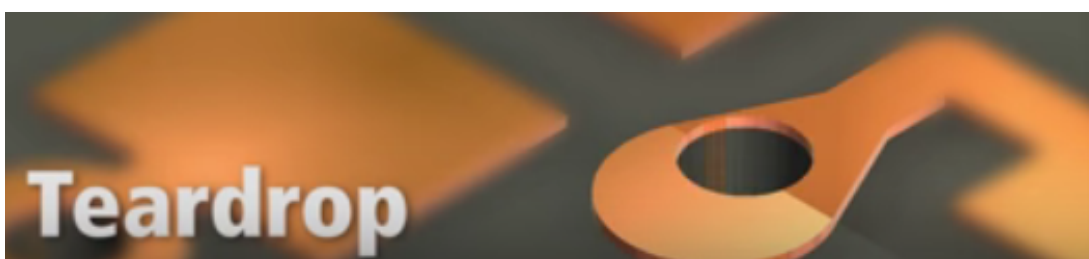
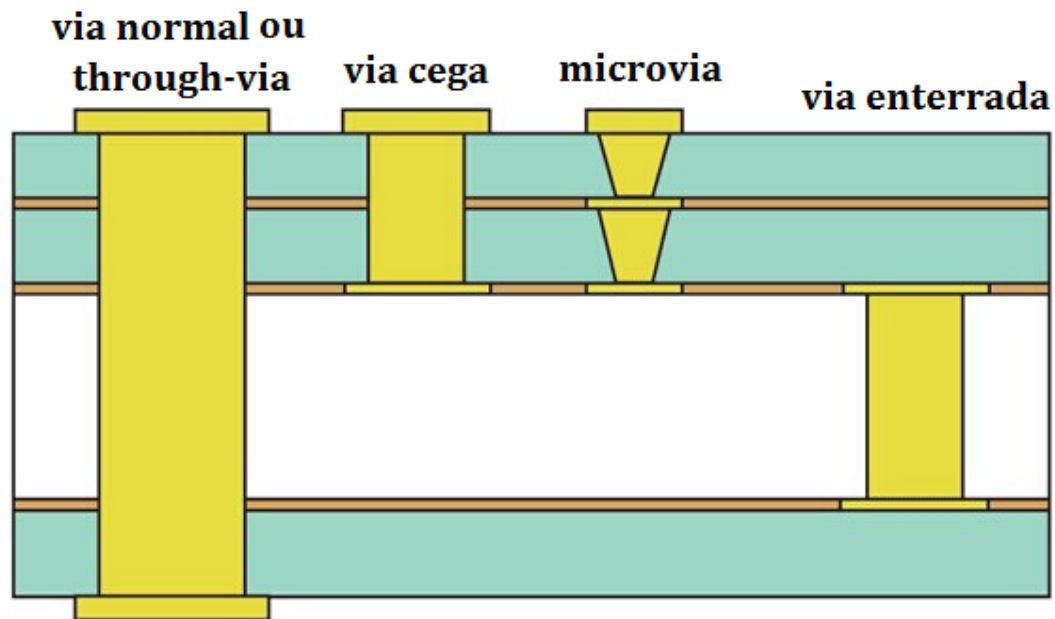


Figura 6 – Diferentes tipos de vias em uma placa de circuito



## 2.3 Tecnologia SDRAM DDR3

SDRAM DDR significa *synchronous dynamic random-access memory double data rate*, i.e. uma memória RAM síncrona (necessita de um sinal de *clock*), dinâmica (os dados são armazenados em elementos capacitivos que devem ser continuamente recarregados) e capaz de ler/escrever dados a uma velocidade duas vezes maior que o seu *clock*. Segundo (WIKIPEDIA, 2018a), no ano 2000 foram lançadas as especificações da primeira versão SDRAM DDR, o DDR1, sendo posteriormente atualizada para DDR2, DDR3 e DDR4, essa última sendo lançada comercialmente em 2014.

Durante o projeto foi realizado o layout de uma interface DDR3-1066 de 64bits, isso significa que a interface é capaz de realizar (teoricamente) 1066 *Megatransfer* por segundo (MT/s). Com um *clock* de 533 MHz e um barramento de 64 bits, cada transferência (leitura ou escrita) é feita com 64bits, o que dá uma taxa de leitura/escrita teórica de aproximadamente 68Gbps.

Para o layout de um circuito usando DDR3 é conveniente dividir os sinais em grupos, que serão apresentados a seguir. Por simplicidade, os grupos serão nomeados da mesma forma que no Capítulo 5:

Grupo DRAM\_SDCLK:

Esse grupo contém os pares diferenciais que transportam o *clock* para cada um dos chips de memória da placa.

Grupo DRAM\_ADDR:

Grupo de sinais responsável por especificar em qual endereço de memória ocorrerá a leitura/escrita.

Grupo *DRAM\_CTRL*:

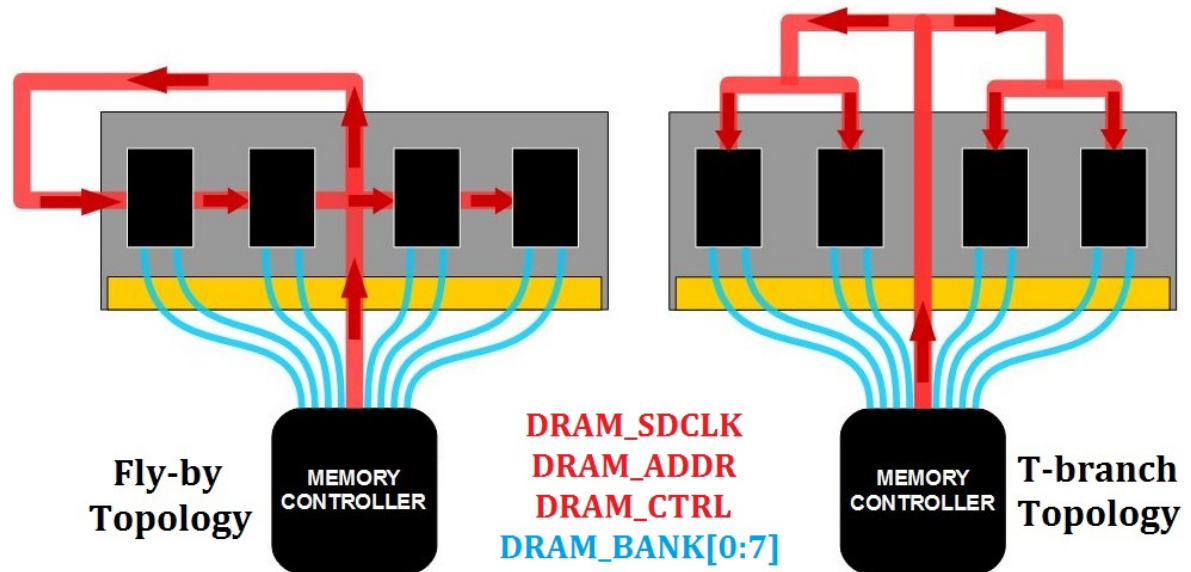
Grupo que contém sinais de controle, como *clock enable*, *chip select*, etc.

Grupo *DRAM\_BANK*:

Em um circuito com DDR3, cada banco é composto por 8 sinais de dados (1 byte), 1 sinal de *data mask* (que habilita ou não a leitura/escrita do banco) e 1 sinal (par diferencial) de *strobe* (responsável por indicar que o banco já pode ser lido). Por usar um controlador de 64bits, esse projeto apresenta 8 grupos *DRAM\_BANK*, *DRAM\_BANK0* a *DRAM\_BANK7*.

Duas topologias comuns para o layout de interfaces DDR3 com mais de 1 chip de memória são *T-Branch* e *Fly-by*, elas diferem apenas na forma como os sinais dos grupos *DRAM\_SDCLK*, *DRAM\_ADDR* e *DRAM\_CTRL* são conectados, conforme ilustra a Figura 7.

A Figura 7 apresenta 1 controlador e 4 chips de memória DDR3. As linhas azuis representam cada um dos grupos *DRAM\_BANK* (perceba que na figura, cada chip de memória recebe duas linhas azuis, significando que são chips de 16bit, 8 bits de dados por linha). A linha vermelha representa os grupos *DRAM\_SDCLK*, *DRAM\_ADDR* e *DRAM\_CTRL*. A única diferença entre as topologias *T-Branch* e *Fly-by* é a forma como os sinais desses grupos são roteados. Na topologia *T-Branch* (à direita) cada um dos sinais é bifurcado duas vezes, garantindo que os sinais cheguem ao mesmo tempo (se o roteamento for bem feito) em todos os chips de memória. Na topologia *Fly-by* (à esquerda), os sinais não sofrem bifurcação, chegando em cada um dos chips de memória em tempos diferentes (os sinais de *strobe* são usados para compensar essa diferença durante uma leitura/escrita). Cada topologia apresenta vantagens e desvantagens, para esse projeto, como será visto nos capítulos futuros, foi usada a topologia *T-Branch* por ocupar um espaço menor na placa.

Figura 7 – Diferença entre as topologias *T-Branch* e *Fly-by*

Em qualquer uma das topologias, é necessário que haja o casamento dos sinais de cada grupo, i.e. o comprimento de todos os sinais de um mesmo grupo deve ser aproximadamente o mesmo. Isso é necessário para que todos os sinais cheguem ao mesmo tempo ao chip, permitindo que o circuito possa operar em sua velocidade normal. Também é necessário que as trilhas apresentem controle de impedância para minimizar reflexões e distorções dos sinais. Segundo (FREESCALE SEMICONDUCTOR, INC., 2013), as trilhas devem apresentar impedância característica de  $50\Omega$  e os pares diferenciais,  $100\Omega$  diferencial.

Finalmente, a Figura 8 apresenta um exemplo de placa que usa topologia *T-Branch* e a Figura 9, um exemplo com topologia *Fly-by*. Essas placas são projetos *open-source* que usam o SoC i.MX6, que podem ser encontrados em (ACADEMY, 2018).

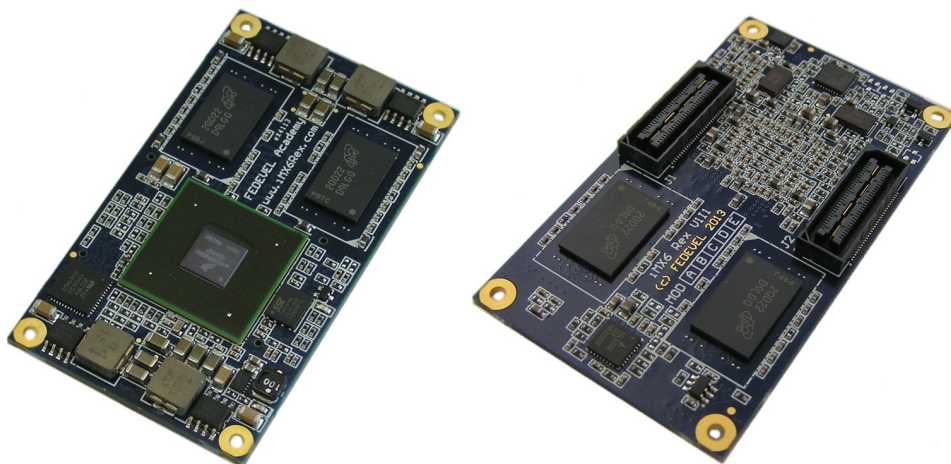
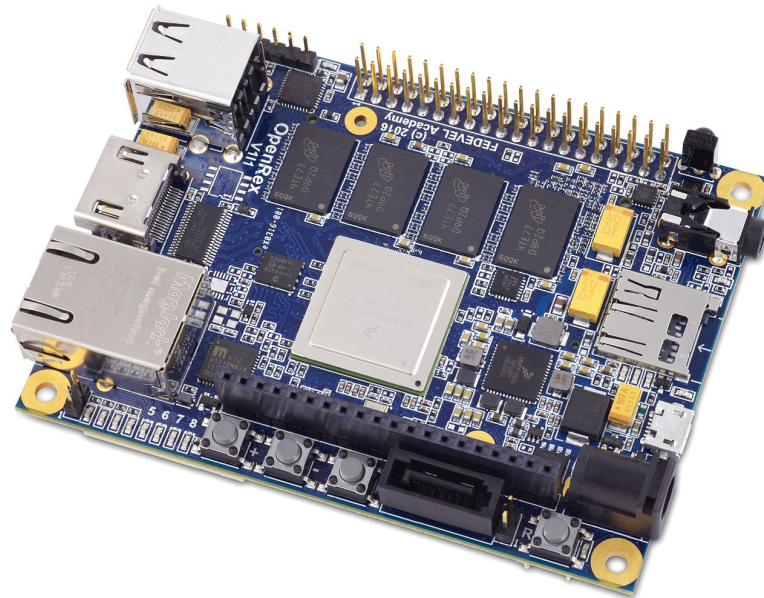
Figura 8 – Exemplo de placa realizada com topologia *T-Branch* - 4 chips, 64bits



Figura 9 – Exemplo de placa realizada com topologia *Fly-by* - 4 chips, 64bits



## 2.4 Layout de conversores CC-CC tipo *Boost*

Esta seção é baseada em (DEVICES, 2018) e (INTEGRATED, 2018) e visa apresentar algumas considerações a serem tomadas ao realizar o layout de um conversor CC-CC. As análises a seguir são feitas apenas para um conversor tipo *boost* por simplicidade, podendo ser estendidas para um tipo *buck* de forma semelhante.

A Figura 10 mostra o fluxo de corrente em um conversor *boost* em seus 2 estados de funcionamento ( $T_{ON}$  e  $T_{OFF}$ ) e a Figura 11 apresenta as formas de onda do conversor, assumindo que está operando em modo de condução contínua.

Figura 10 – Fluxo de corrente em um conversor *boost* nos estados  $T_{ON}$  e  $T_{OFF}$

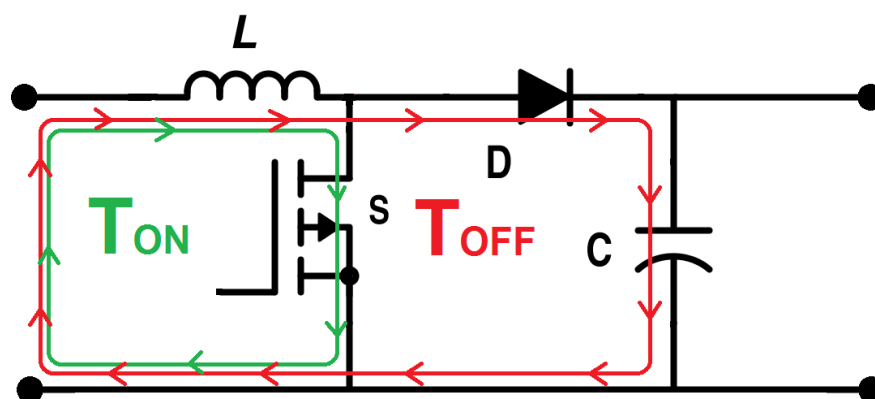
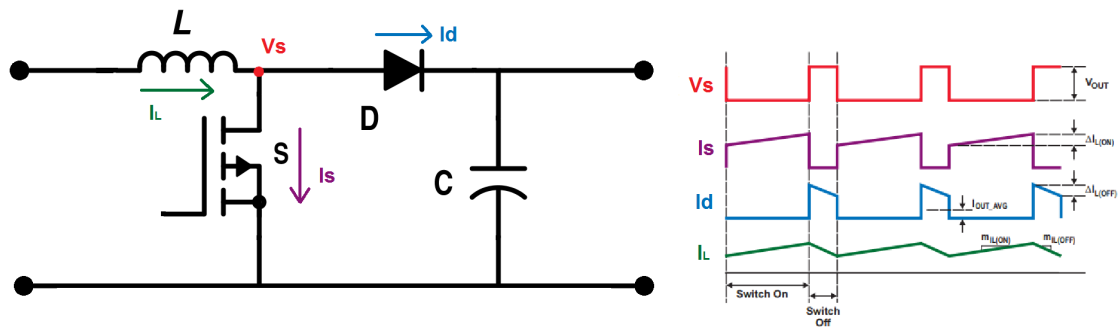


Figura 11 – Formas de onda de um conversor *boost* operando em modo de condução contínua



Analisando as figuras é possível perceber que a malha composta pelos elementos S, D e C é a que apresenta a maior derivada de corrente no circuito. Uma alta derivada de corrente pode induzir tensões indesejadas no circuito se a indutância total da malha for alta, podendo provocar o fenômeno conhecido como *ground bouncing*. Para evitar isso deve-se fazer o layout do conversor tentando minimizar ao máximo o tamanho dessa malha.

Analisando a Figura 11 pode-se constatar que o nó comum entre os elementos L, S e D, as vezes chamado de *switching node*, é o que apresenta a maior derivada de tensão, tornando-o um potencial emissor de ruído eletromagnético. Para evitar que o conversor polua o resto do circuito, deve-se reduzir ao máximo o tamanho deste nó.

### 3 Especificações do protótipo final

A lista de funcionalidades necessárias para o protótipo define o chamado Produto Minimamente Viável (sigla MVP - em inglês), que é o mínimo necessário para ser considerado um produto e poder ser vendido como tal.

A lista de funcionalidades mínimas de um MVP é muito específica para cada produto. Para o dispositivo cujo Hardware foi desenvolvido nesse projeto, essas funcionalidades foram definidas como:

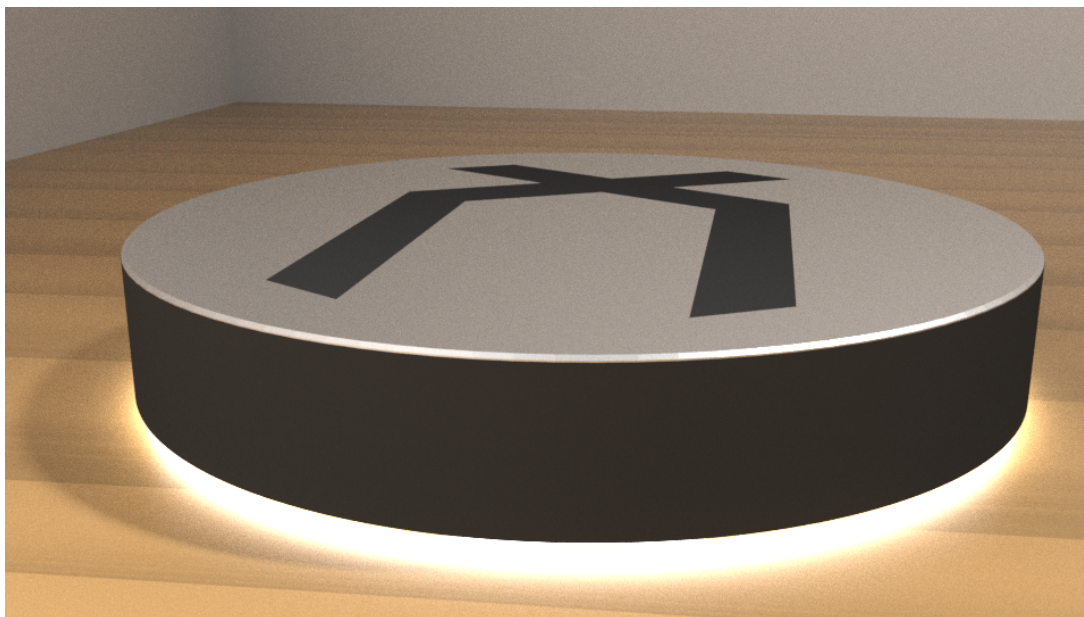
- Conectores *Ethernet* e *Jack* de alimentação acessíveis ao usuário para garantir conexão à Internet e alimentação, respectivamente, ao dispositivo.
- Aplicativos para as plataformas iOS, Android, Windows e macOS permitindo o acesso aos arquivos dentro do HD no interior do dispositivo.
- O HD deverá ser montado como um disco nos sistemas Windows e Mac (o HD aparecerá para o usuário como um pen-drive ou disco externo).
- Possibilidade de realizar *snapshots* dos arquivos, permitindo restaurá-los a versões anteriores.
- LED para indicação do status do aparelho (Iniciando o sistema; Standby; Leitura/escrita no HD).
- Criptografia para transporte de dados.

Apesar da maior parte dessas funcionalidades ser específica do Software (que não serão desenvolvidas nesse projeto), é interessante incluí-las aqui para que o leitor tenha uma ideia do produto final.

Quanto à aparência física do dispositivo, adotou-se (após algumas iterações de ideias) o modelo apresentado na Figura 12 como envólucro do produto. O formato é de um cilindro de 14cm de diâmetro, 2cm de altura e LEDs indicadores localizados na parte de baixo do cilindro, próximos à borda. O design foi assim escolhido para chamar a atenção dos possíveis compradores, além de combinar com o nome que o produto terá, *Allogica Disk*.

Por necessitar de um microprocessador operando em uma frequência em torno de 1GHz, além de uma quantidade considerável de memória RAM (de acordo com as demandas do software), o projeto envolveu o design de uma PCB multi-camada.

Figura 12 – Renderização do envólucro do dispositivo



Uma vez que o preço por área de uma placa multi-camada é muito maior do que o preço de uma placa de 2 camadas, o circuito foi realizado em 2 placas. Uma placa menor mas com muitas camadas, chamada Placa Mãe, e uma placa maior mas com poucas camadas, chamada Placa IO.

Uma grande vantagem de dividir o design em 2 placas é a possibilidade de testar cada uma individualmente, o que permite a correção de bugs em uma placa independentemente da outra, além de reduzir o custo de desenvolvimento do protótipo (caso uma das placas não funcione e seja preciso mandar fabricar outra), algo muito importante para uma *startup*.

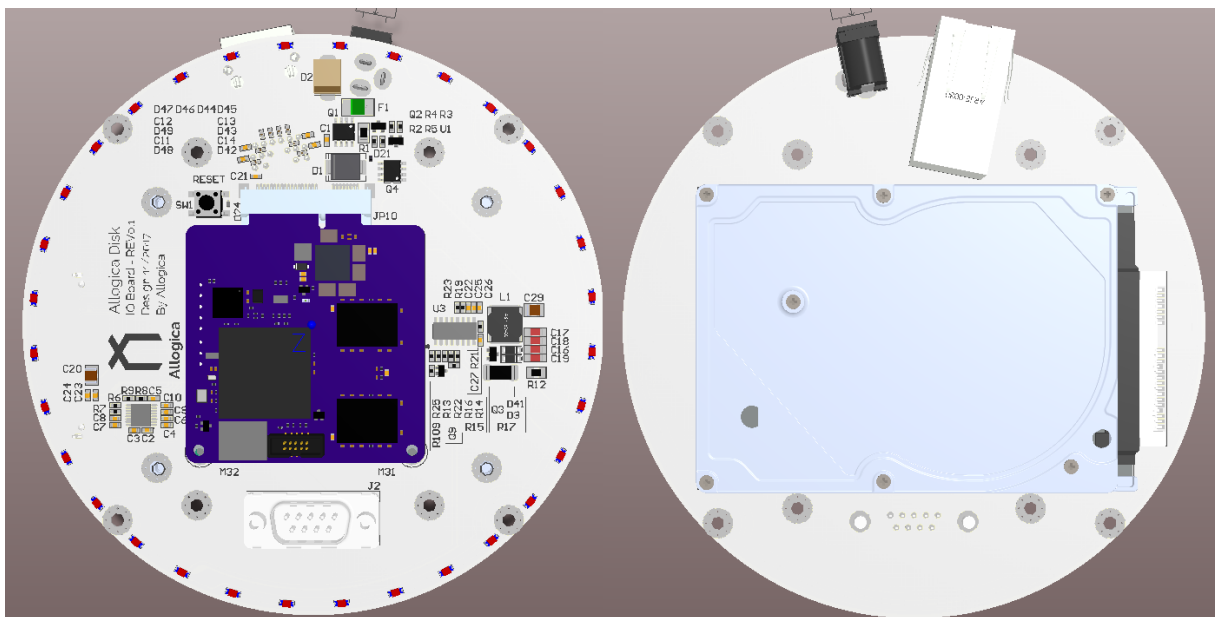
Realizado o estudo dos dispositivos disponíveis no mercado, as especificações para um primeiro protótipo e as restrições mecânicas do envólucro, pode-se elaborar uma lista de funcionalidades para o Hardware, que norteará a realização do circuito.

- Proteção contra descarga eletrostática nos conectores e proteção contra sobretensão e polaridade reversa no conector de alimentação.
- Luminosidade programável dos LEDs indicadores, com mais uniforme possível.
- Memória RAM mínima de 512MB (de acordo com as demandas do software).
- Memória flash mínima de 256MB para comportar o sistema operacional e o resto do software (com uma margem para futuras atualizações).
- Suporte para HD de 2.5" (preferencialmente sem o uso de cabos, por simplicidade de montagem e estética).

## 4 Desenvolvimento do circuito

Após algumas iterações realizando o *placement* dos componentes nas 2 placas, o arranjo final é apresentado na Figura 13, onde a placa maior, Placa IO, contém apenas 2 camadas e a placa menor, Placa Mãe, 8 camadas. A cor da Placa IO é branca para ajudar a refletir a luz dos LEDs na sua borda (que serão brancos), garantindo assim uma luminosidade mais uniforme.

Figura 13 – Modelo 3D das placas de circuito do protótipo



A seguir são apresentados os blocos que compõem o circuito do dispositivo um por um e a justificativa para a escolha de seus componentes. Os testes de validação são apresentados no capítulo seguinte.

### 4.1 Processador central

Estudando os dispositivos concorrentes percebe-se que a maioria utiliza um microprocessador *dual core* com arquitetura ARMv7 operando em torno de 1GHz. De acordo com as demandas do software, um processador com essas especificações é o suficiente para o dispositivo NAS aqui realizado. Infelizmente, o SoC usado pelos demais fabricantes não é vendido nos fornecedores de componentes eletrônicos mais conhecidos (como *Digi-Key*, *Arrow*, *Farnell*, *Mouser*, etc), tornando muito difícil a utilização de tal componente por uma empresa muito pequena, que fabrica produtos em baixa quantidade.

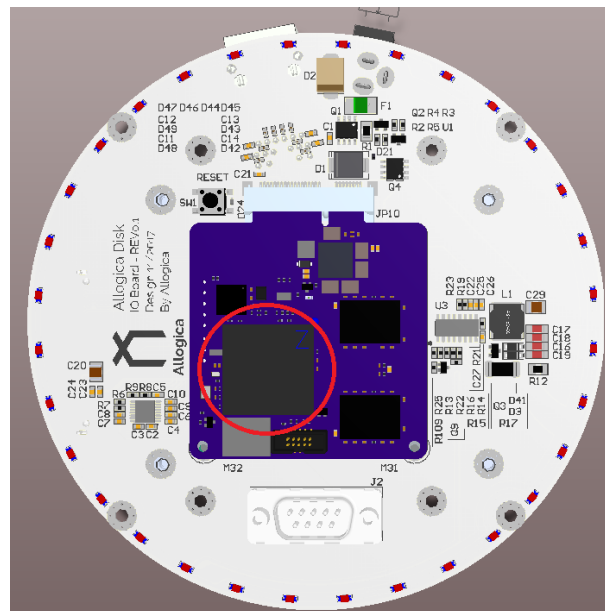
O processador central escolhido foi o i.MX6D, da fabricante NXP. Analisando seu *datasheet* (FREESCALE SEMICONDUCTOR, INC., 2015), vemos que o SoC contém 2 núcleos ARM Cortex-A9 @1GHz, SATA-II, *Ethernet Gigabit* e controlador de memória RAM DDR3-1066 de 64bits.

O processador foi escolhido por apresentar, entre outras, as seguintes vantagens:

- Preço relativamente baixo, em torno de US\$ 36,00 por unidade (para centenas de peças);
- Disponível nos maiores e mais conhecidos fornecedores de componentes eletrônicos;
- Controlador SATA e *Ethernet*;
- Possibilidade de expansão para 4 núcleos se necessário (i.MX6Q);
- Controlador DDR3 de 64bits;
- GPU que pode ser usada para acelerar a criptografia;
- Grande quantidade de designs e kits de referência;

A Figura 14 mostra a posição do processador central no produto.

Figura 14 – Localização do processador central no produto



## 4.2 Regulador de Tensão

Para alimentar o processador central (i.MX6D), são necessários diversos níveis de tensão, sendo que alguns deles precisam ser ligados em uma ordem específica para garantir a correta inicialização do processador.

Para simplificar o design de um circuito com o processador escolhido, o fabricante sugere a utilização de um PMIC (*Power Management Integrated Circuit*), capaz de gerar todos os níveis necessários e na ordem correta.

O PMIC escolhido foi o MMPF0100F4 (também pode-se usar a versão MMPF0100F0), do mesmo fabricante do processador. O circuito apresenta 5 conversores *Buck*, 1 conversor *Boost* e 6 reguladores lineares. A Figura 15 apresenta um exemplo de aplicação do PMIC e a Figura 16 mostra a posição do PMIC no circuito.

Figura 15 – Exemplo de aplicação do PMIC

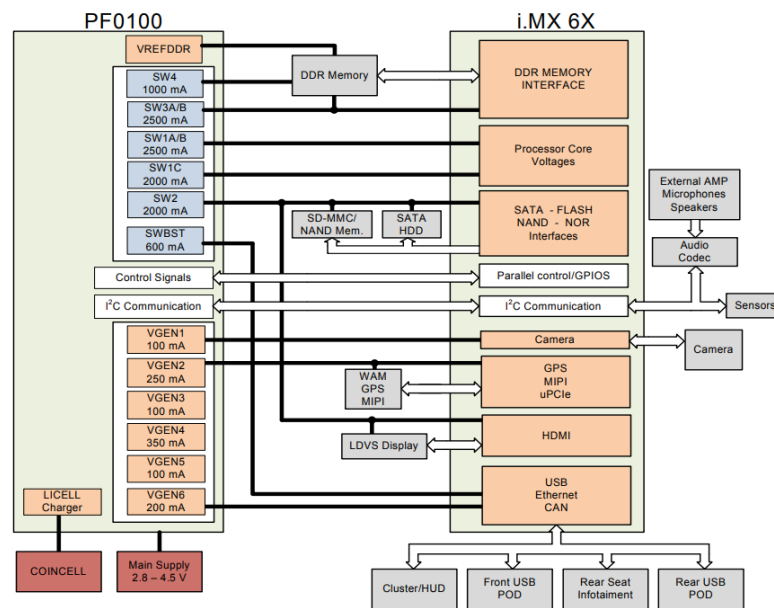
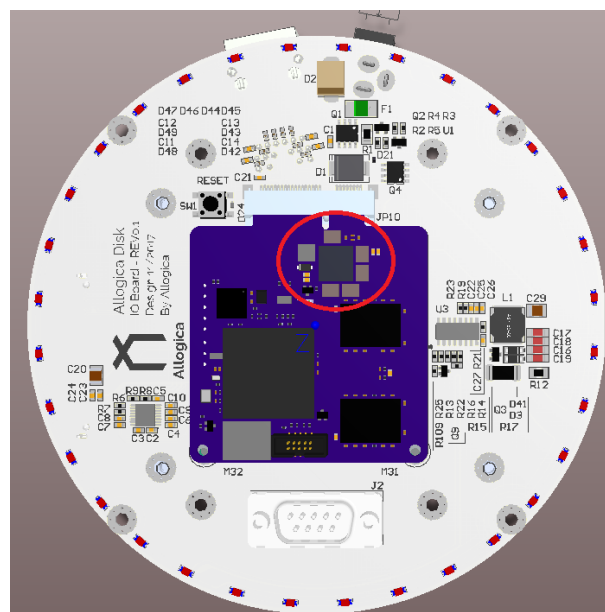


Figura 16 – Localização do PMIC no circuito

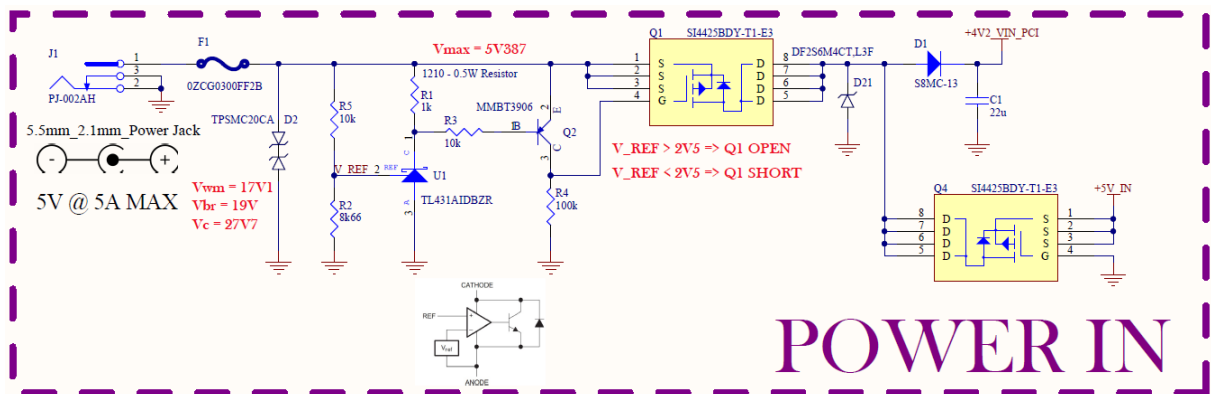


### 4.3 Circuito de proteção de entrada

O esquemático final para este bloco do circuito é apresentado na Figura 17. A função dessa parte do circuito é proteger o dispositivo contra sobretensão, descargas eletrostáticas e polaridade reversa.

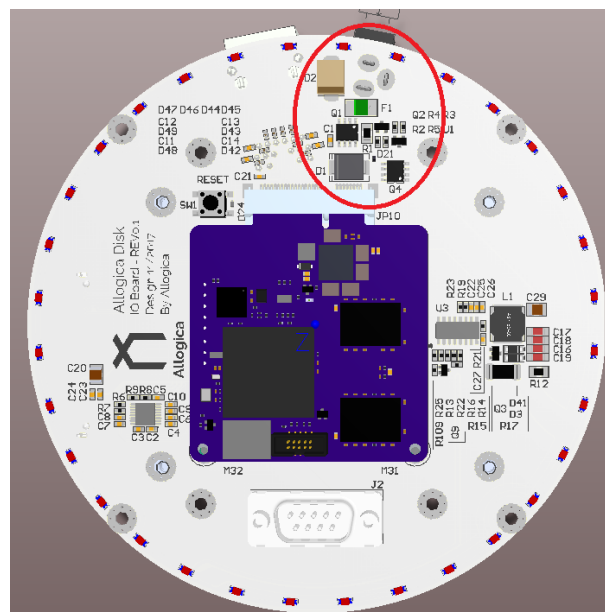
O componente J1 é o conector de alimentação do circuito. Os componentes R1-5, U1 e Q1-2 garantem proteção contra tensões acima de 5,4V. O componente D2 é um supressor de transientes de tensão (TVS, em inglês). O diodo D1 garante proteção contra polaridade reversa e rebaixa a tensão de entrada nominal de 5V para 4,2V para ser usado pelo regulador de tensão da Placa Mãe. O transistor Q4 garante proteção contra polaridade reversa e mantém a tensão nominal de 5V de entrada para alimentar o HD e os LEDs. O diodo D21 é um supressor de transientes que foi incluído no circuito por garantia.

Figura 17 – Esquemático do bloco Circuito de proteção de entrada



A Figura 18 mostra a posição do circuito de proteção de entrada no produto.

Figura 18 – Localização do circuito de proteção de entrada no produto





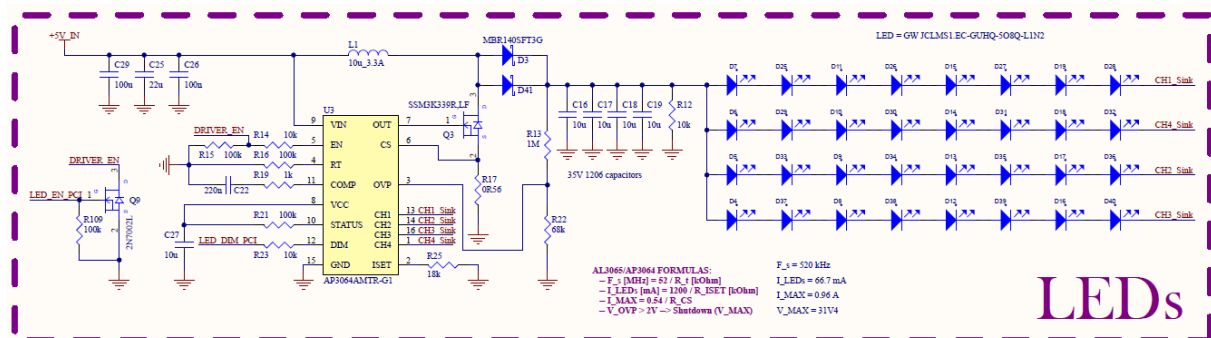
## 4.4 Driver dos LEDs indicadores

O esquemático final para este bloco do circuito é apresentado na Figura 19. A função dessa parte do circuito é permitir o controle da intensidade luminosa dos LEDs por PWM, provendo assim uma indicação visual do *status* do dispositivo para o usuário.

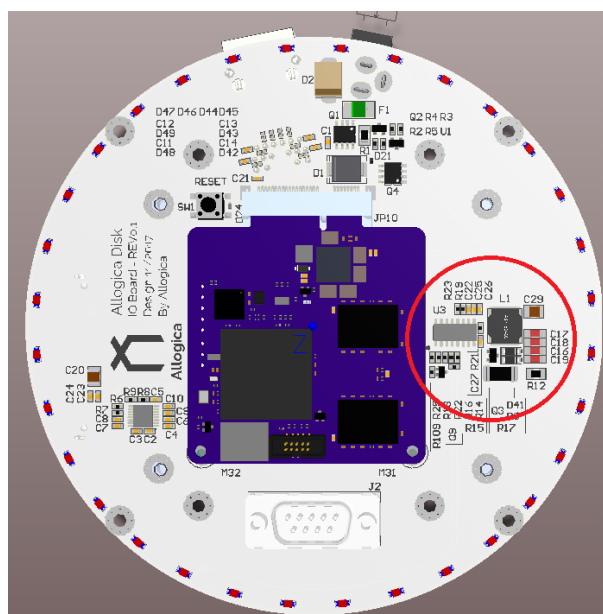
O *driver* escolhido foi o AP3064 por possibilitar o acionamento dos LEDs por fonte de corrente (o que garante que cada LED tenha aproximadamente a mesma intensidade luminosa), por sua capacidade de acionar uma grande quantidade de LEDs (máximo teórico de 80 LEDs de 3V cada) com uma fonte de apenas 5V e por possibilitar o controle da intensidade luminosa do arranjo de LEDs por PWM.

O circuito funciona como um conversor *Boost*, que eleva a tensão de entrada de 5V para um valor capaz de excitar os LEDs (a tensão depende do número de LEDs e, neste caso, ela está em torno de 24V). O CI tem também 4 pinos que funcionam como fontes de corrente (pinos CH1 - 4). Um sinal PWM na entrada DIM do circuito integrado (pino 12) controla o período em que as 4 fontes de corrente estão ligadas, possibilitando assim o controle do brilho de todos os LEDs ao mesmo tempo.

Figura 19 – Esquemático do bloco *Driver* dos LEDs indicadores



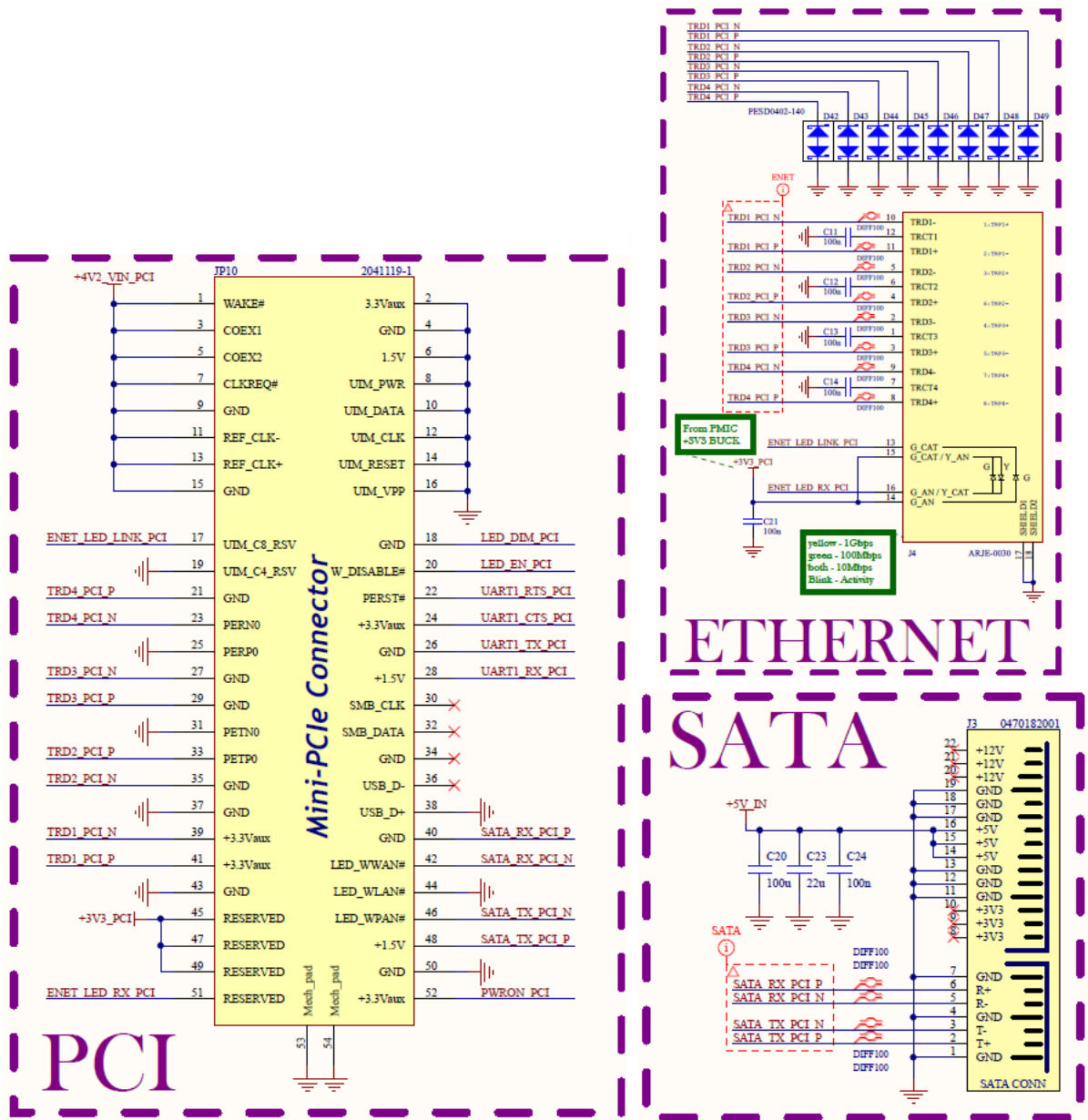
A Figura 20 mostra a posição do *driver* dos LEDs no produto.

Figura 20 – Localização do *driver* dos LEDs no produto

## 4.5 Conectores

Além do conector de alimentação (apresentado na Seção 4.3) e do conector de *debug* serial (que será comentado na Seção 4.7), a Placa IO tem ainda 3 conectores, apresentados na Figura 21 .

Figura 21 – Conectores PCIe, SATA e Ethernet



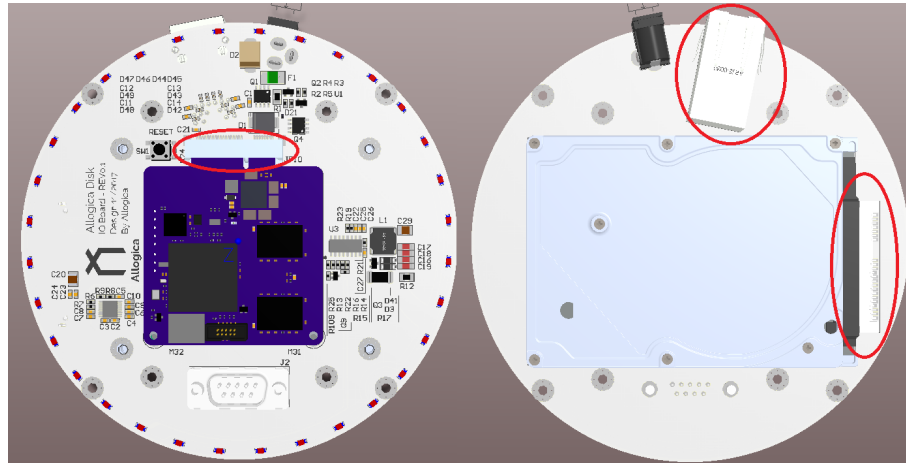
Esse modelo de conector SATA foi escolhido pois ele possibilita fixar o HD na parte de baixo da Placa IO sem a necessidade de cabos, o que garante uma boa estética e facilidade na montagem do produto.

O conector Ethernet foi escolhido por suas dimensões físicas. O modelo de conector escolhido garante que nenhum pino irá ficar no caminho dos LEDs, além de ser longo o suficiente para poder ser alinhado com as paredes externas do envólucro.

Um conector mini PCI express é usado para conectar a Placa Mãe e a Placa IO, pois suporta sinais de alta frequência, e ocupa pouco espaço na placa.

A Figura 22 mostra a posição dos conectores no protótipo.

Figura 22 – Localização dos conectores no produto



## 4.6 Microcontrolador

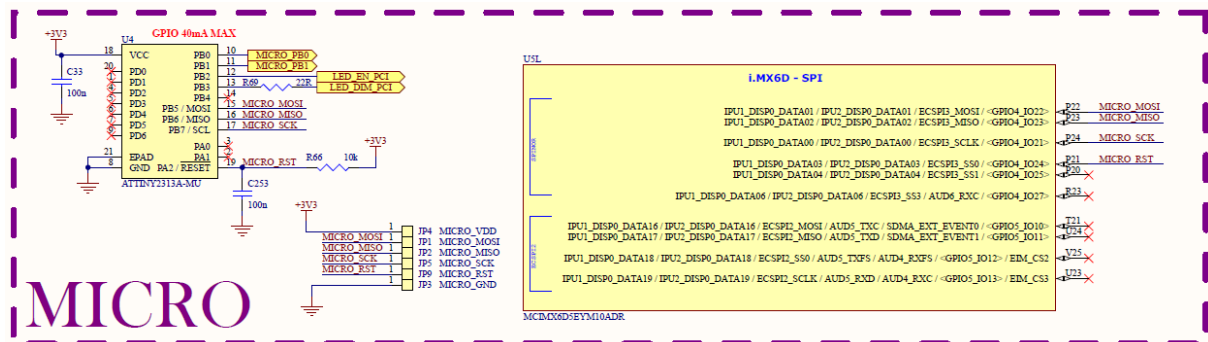
Uma das funcionalidades pensadas para o produto foi a possibilidade de indicar ao usuário quando o processador terminou o processo de *boot* e está funcionando normalmente. Para isso é necessário que o brilho dos LEDs possa ser controlado enquanto o processador principal está inicializando.

Uma forma simples de conseguir isso é acrescentando um microcontrolador ao circuito. Este microcontrolador, mais simples e barato que o processador central, ficará responsável pelo controle do brilho dos LEDs.

O microcontrolador escolhido foi o ATTINY2313 pelo seu preço e possibilidade de implementar um PWM em Hardware (basta escrever em um registrador para mudar o período do PWM). Esse micro se comunica com o processador central por SPI (*Serial Peripheral Interface*).

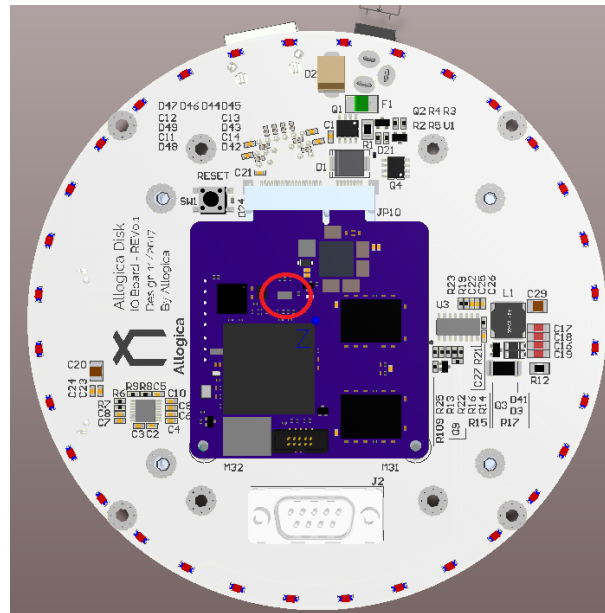
A Figura 23 mostra a parte do esquemático contendo o microprocessador.

Figura 23 – Esquemático contendo o microprocessador responsável pelo controle dos LEDs



A Figura 24 mostra a posição do microcontrolador no produto.

Figura 24 – Localização do microcontrolador no produto



## 4.7 Demais componentes

Além dos blocos apresentados anteriormente, o circuito também contém um PHY para Ethernet, memória Flash, SDRAM DDR3 e circuito para *debug* serial. Esses blocos serão apresentados a seguir.

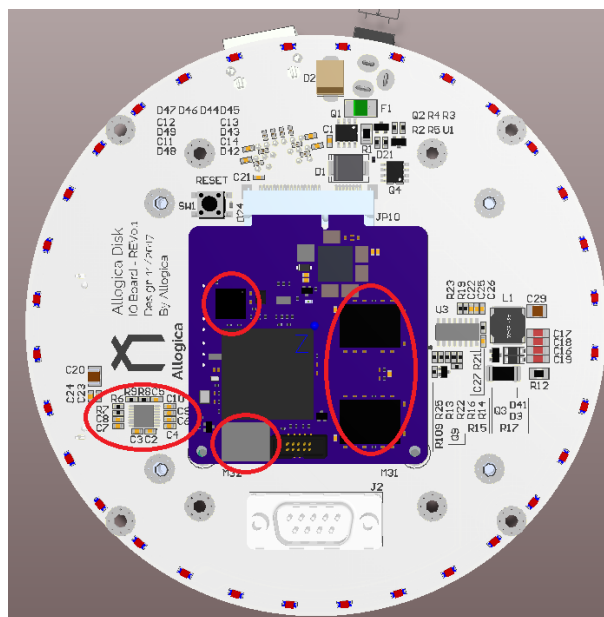
Um chip que implementa a camada física de um protocolo é geralmente chamado de PHY. Um PHY para Ethernet gera o sinal apropriado para ser transmitido por um conector Ethernet, sendo escolhido para esse serviço o circuito KSZ9031 do fabricante *Microchip*. Ele foi escolhido por ser usado em um dos designs de referência do i.MX6 e por apresentar um componente equivalente pino-a-pino (o KSZ9021) que pode ser usado em caso de falta de estoque.

Para a memória RAM, os testes realizados em software (não tratados nesse trabalho) mostram que 512MB é o suficiente para o dispositivo operar. O controlador SDRAM do i.MX6 suporta um barramento de dados de 64bits, sendo usados no projeto 4 chips DDR3 de 16bits por compatibilidade com o design de referência. A memória Flash comporta o sistema operacional e o *firmware*. Um chip de 256 MB de armazenamento é o suficiente (havendo ainda margem para futuras atualizações) de acordo com os testes de software.

Para fins de desenvolvimento e *debug*, o circuito conta com um conversor UART - RS232 e um conector serial (não fará parte do produto final), foi utilizado o mesmo esquemático do kit de desenvolvimento *WandBoard*.

A Figura 25 mostra a posição dos demais componentes do produto.

Figura 25 – Localização dos demais componentes do produto



## 5 Layout das placas de circuito

Conforme mencionado em capítulos anteriores deste trabalho, o protótipo conta com duas placas de circuito, uma de duas camadas, chamada de Placa IO, e outra de oito camadas, chamada de Placa Mãe. Este capítulo descreve como foi realizado o layout dessas placas.

A maior parte do capítulo será destinada à Placa Mãe, haja vista sua complexidade. Tendo explicado como foi realizado o layout desta, o roteamento da Placa IO torna-se uma tarefa bastante trivial, podendo ser resumido ao final.

Para ter uma ideia do resultado final, a Figura 26 mostra todos os *layers* da Placa Mãe e a Figura 27, os da Placa IO.

Figura 26 – *Layers* da Placa Mãe

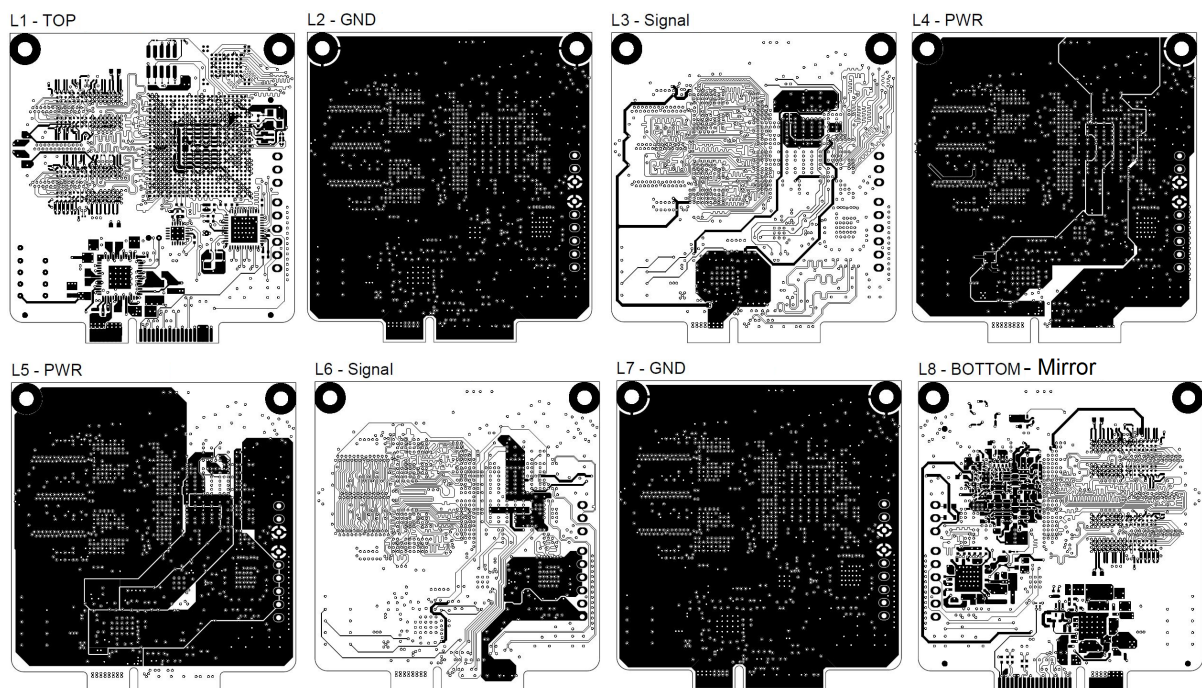
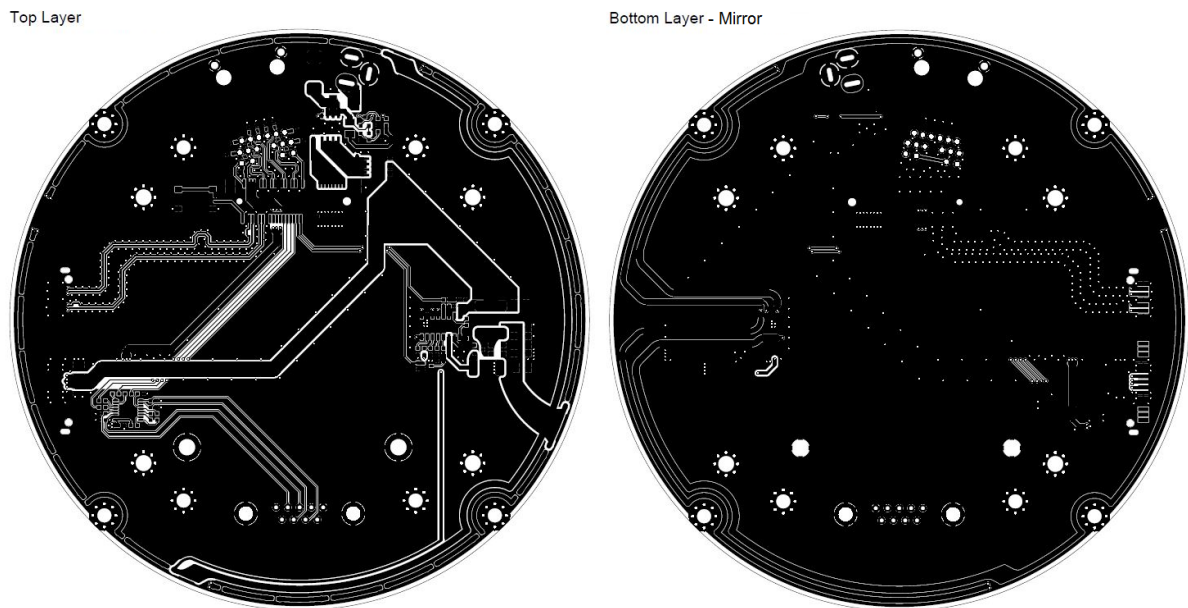


Figura 27 – Layers da Placa IO



## 5.1 Stack-up da Placa Mãe

Ao desenvolver uma placa de circuito multi-camada é recomendado começar com o planejamento do *stack-up*, que é simplesmente o arranjo de como a placa será construída, i.e. quantas camadas ela terá e qual o material e espessura de cada camada.

A construção do *stack-up* começou com a escolha de usar 8 camadas na placa. Esse número foi escolhido por ser o recomendado pelo fabricante do SoC usado no projeto, vide (FREESCALE SEMICONDUCTOR, INC., 2013), além de ser o número de camadas do design de referência para o mesmo.

Conforme mencionado no Capítulo 2, para que as interfaces de alta velocidade presentes no circuito (SDRAM, SATA, *Ethernet*) funcionem adequadamente é necessário que as trilhas destas mantenham uma impedância característica constante em seu percurso pela placa. O valor dessa impedância depende das especificações da interface. Para esse projeto, os únicos valores de impedância necessários são de  $50\Omega$ ,  $100\Omega$  diferencial e  $90\Omega$  diferencial.

Escolhidos o número de camadas e os valores de impedância necessárias para cada interface presente no circuito é possível então escolher o tipo de dielétrico que será usado na placa. Para isso, o mais recomendado é perguntar ao fabricante de placas de circuito que pretende contratar com quais laminados ele trabalha e quais ele recomendaria para o seu design, evitando assim ter que modificar o layout posteriormente. Para o projeto da Placa Mãe, o fabricante aconselhou o uso dos laminados da família Megtron 4, da fabricante Panasonic, cujo *datasheet* pode ser visto em (PANASONIC, 2013).



Tendo o número de camadas, os valores de impedância e o tipo de laminado é possível escolher a espessura de cada camada da placa. Essa parte foi realizada através de um processo iterativo, onde a espessura dos laminados foi escolhida arbitrariamente, seguido do cálculo da largura (**W**) e espaçamento (**S**) das trilhas em cada camada de sinal da placa a fim de se obter os valores de impedância desejados. O processo é então repetido até que se tenha trilhas com largura e espaçamento aceitáveis. O resultado desse processo pode ser visto na Figura 28, que mostra o *stack-up* final da placa. Pode-se perceber na figura que a espessura final da placa deve ser próxima de 1mm, isso é necessário pois, examinando a Figura 13, vemos que a Placa Mãe é conectada à Placa IO através de um conector mini PCIe, que exige uma placa com essa espessura.

Figura 28 – *Stack-up* da Placa Mãe

8 Layer Stack-up			Diff 90Ohm		Diff 100Ohm		Single 50Ohm
	Thickness (um)	Type	W	S	W	S	W
L1	18	Signal	0.1	0.1	0.09	0.17	0.093
	50	PrePreg - Dk = 3.55					
L2	18	GND					
	100	CORE - Dk = 3.84					
L3	18	Signal	0.08	0.1	0.08	0.195	0.085
	150	PrePreg - Dk = 3.57					
L4	18	PWR / GND					
	250	CORE					
L5	18	PWR / GND					
	150	PrePreg - Dk = 3.57					
L6	18	Signal	0.08	0.1	0.08	0.195	0.085
	100	CORE - Dk = 3.84					
L7	18	GND					
	50	PrePreg - Dk = 3.55					
L8	18	Signal	0.1	0.1	0.09	0.17	0.093
TOTAL =		0,994	mm		Target = 1 mm @ 10% tol		

Na Figura 28, os valores de largura (**W**) e espaçamento (**S**) estão em mm e **Dk** é a permissividade elétrica relativa (também chamada de constante dielétrica). Nas camadas **L1** e **L8** as linhas de transmissão foram feitas em topologia *microstrip* e nas camadas **L3** e **L6**, em topologia *stripline*. Os cálculos de impedância foram feitos com o software Si9000.

## 5.2 DDR3 e demais interfaces de alta velocidade da Placa Mãe

O SoC i.MX6 apresenta um controlador DDR3-1066 de 64bits, fazendo desta a interface mais complexa de todo o circuito. A topologia escolhida para o layout foi a *T-Branch* pois ela permite ocupar menos espaço na placa. Para o layout foram seguidas as

recomendações do fabricante em (FREESCALE SEMICONDUCTOR, INC., 2013) e o design de referência *SABRE-AI*, que pode ser visto na Figura 29.

Figura 29 – Design de referência para o i.MX6, *SABRE-AI*



As regras de design para a topologia *T-Branch* de acordo com (FREESCALE SEMICONDUCTOR, INC., 2013) estão resumidas nas tabelas abaixo, onde **CLK\_MIN** representa o menor comprimento de um sinal do grupo **DRAM\_SDCLK**.

Tabela 2 – Regras de design DDR3 - grupo DRAM\_SDCLK

Sinais	Comprimento		Recomendações
	Min (mm)	Max (mm)	
<b>DRAM_CLK0</b>	menor	57	casar os sinais diferenciais em $\pm 0,127$ mm;
<b>DRAM_CLK1</b>	possível		

Tabela 3 – Regras de design DDR3 - grupo DRAM\_ADDR

Sinais	Comprimento		Recomendações
	Min (mm)	Max (mm)	
<b>DRAM_A[15:0]</b> <b>DRAM_SDBA[2:0]</b> <b>DRAM_RAS</b> <b>DRAM_CAS</b> <b>DRAM_WE</b>	CLK_MIN - 5	CLK_MIN	casar os sinais em $\pm 0,6\text{mm}$ ;

Tabela 4 – Regras de design DDR3 - grupo DRAM\_CTRL

Sinais	Comprimento		Recomendações
	Min (mm)	Max (mm)	
<b>DRAM_CS0</b> <b>DRAM_SDCKE</b> <b>DRAM_SDOdT0</b>	CLK_MIN - 5	CLK_MIN	casar os sinais em $\pm 1,27\text{mm}$ ;

Tabela 5 – Regras de design DDR3 - grupos DRAM\_BANK[7:0]

Sinais	Comprimento		Recomendações
	Min (mm)	Max (mm)	
<b>DRAM_D[63:0]</b> <b>DRAM_DQM[7:0]</b> <b>DRAM_SDQS[7:0]</b>	-	CLK_MIN	casar os sinais de cada banco em $\pm 0,6\text{mm}$ ; casar os sinais diferenciais em $\pm 0,25\text{mm}$ ;

Seguindo essas regras e tendo como base o design de referência *SABRE\_AI*, o layout da interface foi realizado em etapas. Na primeira etapa, todos os pinos do i.MX6 pertencentes ao controlador DDR3 foram conectados aos seus respectivos pinos nos chips de memória, sem se preocupar em manter o mesmo comprimento para os sinais de um mesmo grupo.

Tendo a certeza de que todos os sinais estão conectados e que há espaço suficiente na placa para alterar o comprimento das trilhas, iniciou-se a segunda etapa, onde o casamento dos sinais é realizado para cada grupo em um processo iterativo.

O processo inicia pelo sinal de maior comprimento de um grupo, tentando encurtá-lo até que deixe de ser o sinal mais comprido, reiniciando o processo para o novo sinal no topo da lista até que não seja mais possível encurta-lo. Não sendo mais capaz de reduzir o comprimento das trilhas inicia-se o processo inverso, começando pelo sinal mais curto do grupo, tenta-se alonga-lo para que tenha o mesmo comprimento da maior trilha. O processo é então repetido para todos os outro sinais, até que todo o grupo esteja dentro da tolerância.

As Figuras 30 a 34 mostram o resultado do processo descrito acima, o layout de todos os grupos da interface DDR3 na Placa Mãe. Nas figuras, cada cor representa uma camada da placa.

Figura 30 – Layout do grupo DRAM\_SDCLK

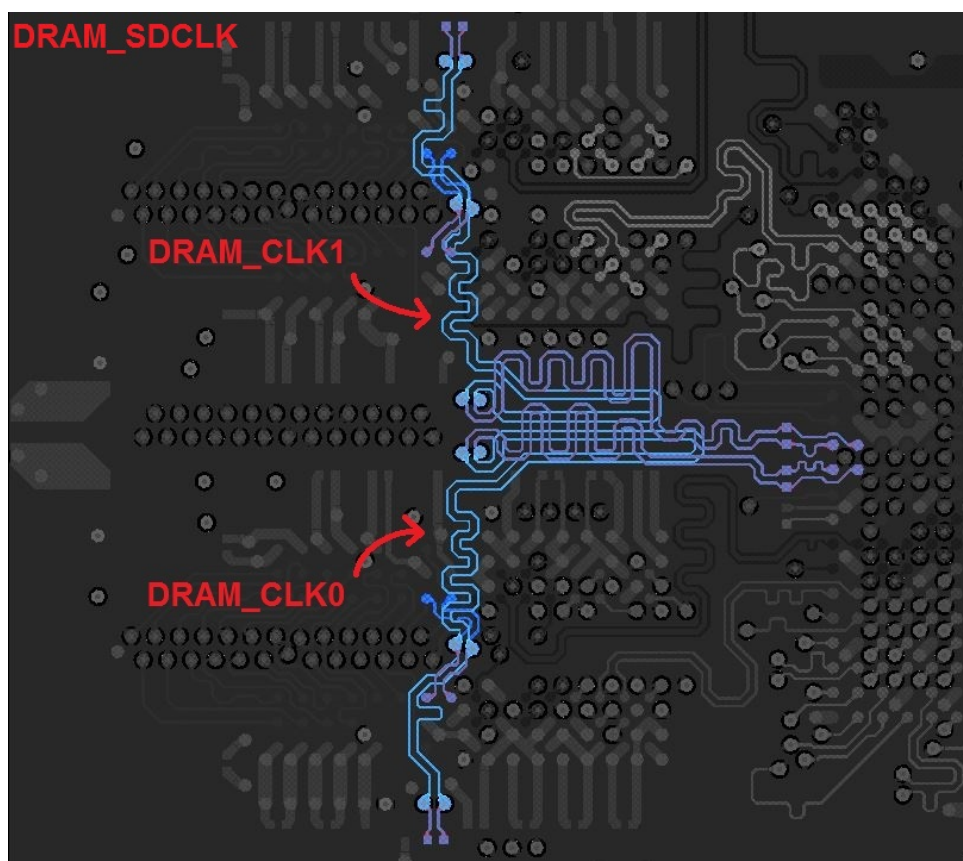


Figura 31 – Layout do grupo DRAM\_ADDR

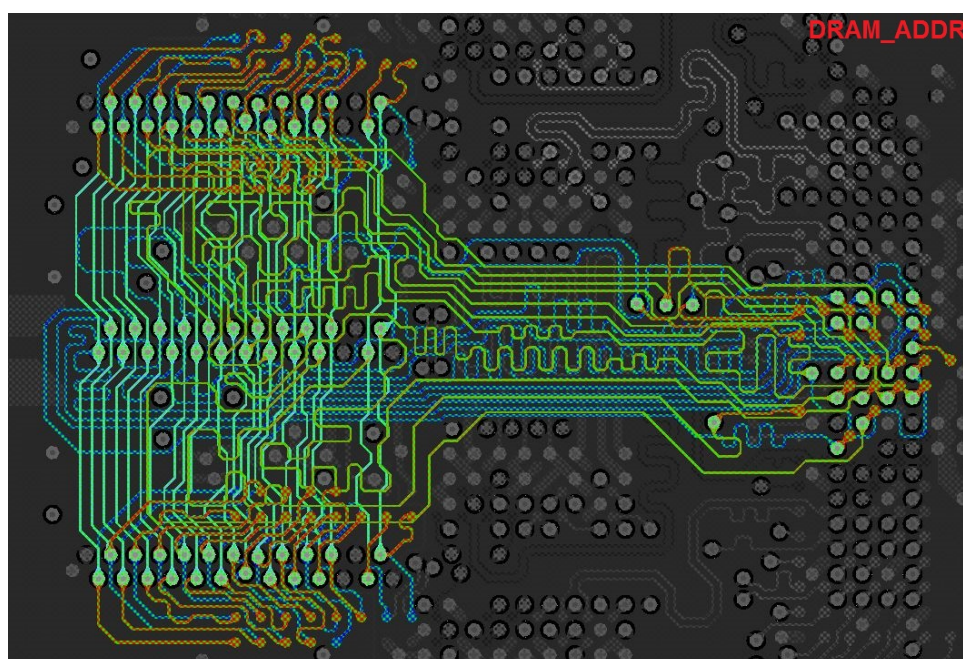


Figura 32 – Layout do grupo DRAM\_CTRL

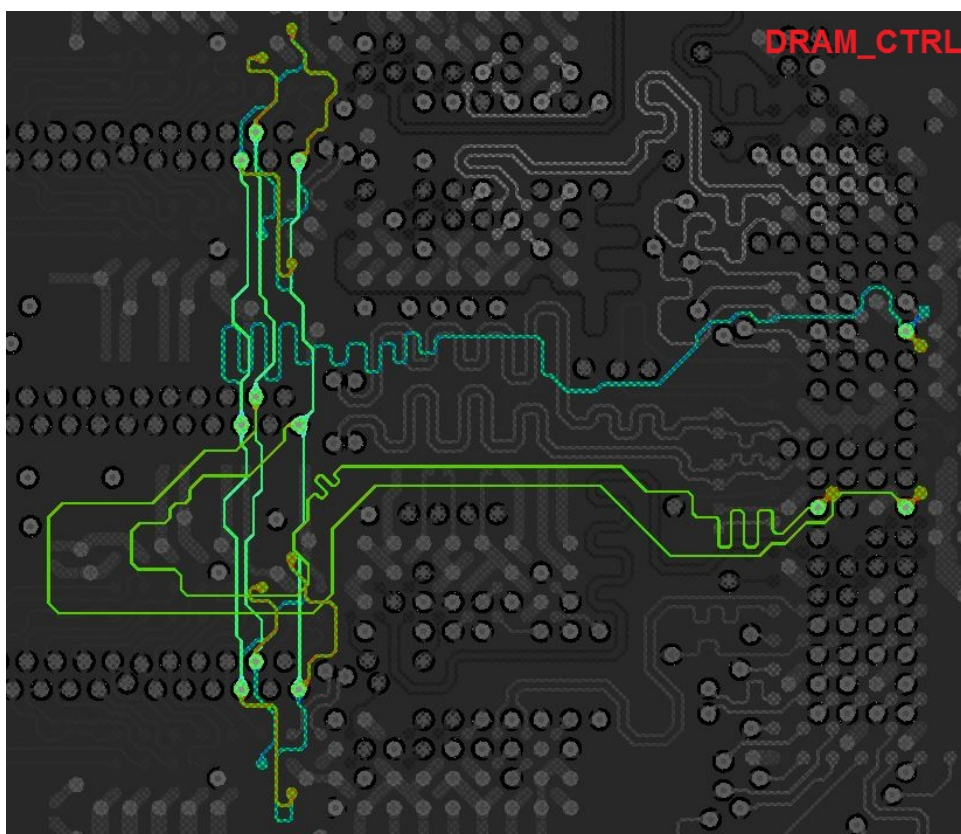


Figura 33 – Layout dos grupos DRAM\_BANK0 a DRAM\_BANK3

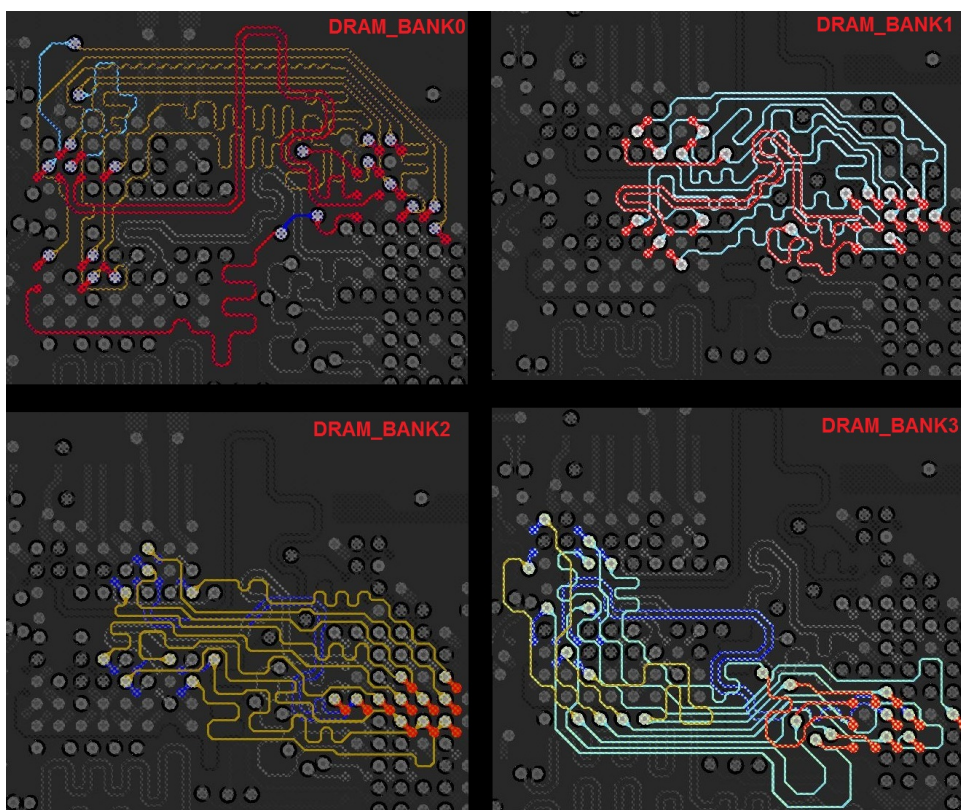
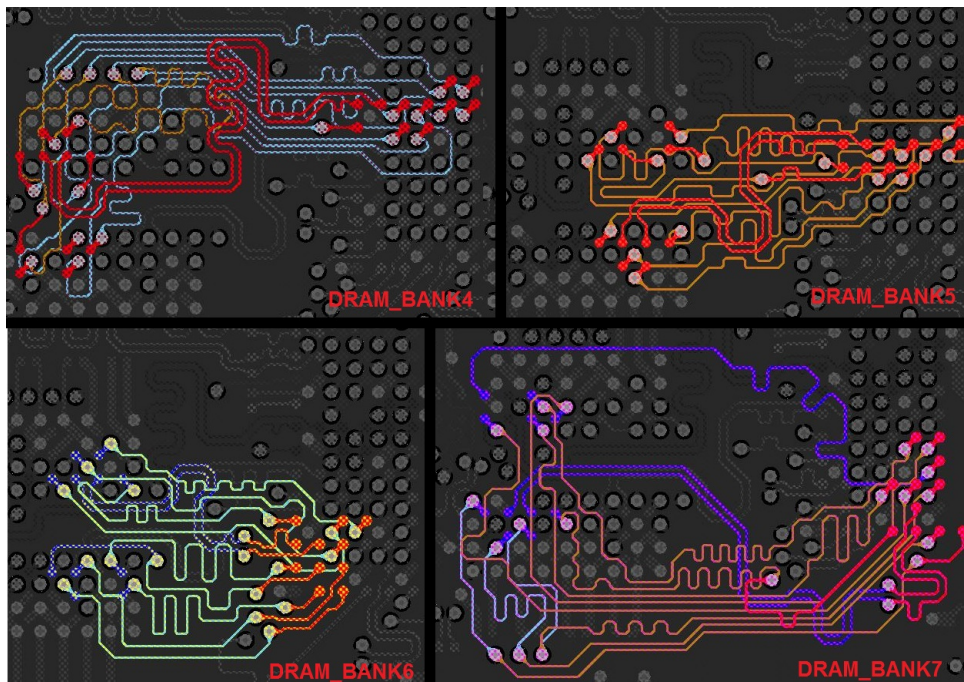


Figura 34 – Layout dos grupos DRAM\_BANK4 a DRAM\_BANK7



A Tabela 6 resume as medidas dos sinais de cada grupo. O design e medidas foram feitos no software de CAD *Altium Designer*.

Tabela 6 – Medidas de comprimento de cada grupo da interface DDR3

grupo	menor comprimento	maior comprimento	diferença de comprimento	diferença máxima permitida
DRAM_SDCLK	51,55mm	51,638mm	0,088mm	0,127mm
DRAM_ADDR	50,338mm	50,75mm	0,412mm	0,6mm
DRAM_CTRL	50,593mm	50,647mm	0,054mm	1,27mm
DRAM_BANK0	26,507mm	26,7mm	0,193mm	0,6mm
DRAM_BANK1	17,978mm	18,2mm	0,222mm	0,6mm
DRAM_BANK2	16,582mm	16,833mm	0,251mm	0,6mm
DRAM_BANK3	23,688mm	23,933mm	0,245mm	0,6mm
DRAM_BANK4	25,056mm	25,305mm	0,249mm	0,6mm
DRAM_BANK5	14,926mm	15,017mm	0,091mm	0,6mm
DRAM_BANK6	14,172mm	14,325mm	0,153mm	0,6mm
DRAM_BANK7	26,711mm	26,908mm	0,197mm	0,6mm

Realizada a interface SDRAM DDR3, o layout das demais interfaces que requerem casamento de sinais e impedância controlada torna-se trivial. O layout final das demais interfaces é apresentado a seguir, nas Figuras 35 e 36.

Figura 35 – Layout das interfaces RGMII e Ethernet

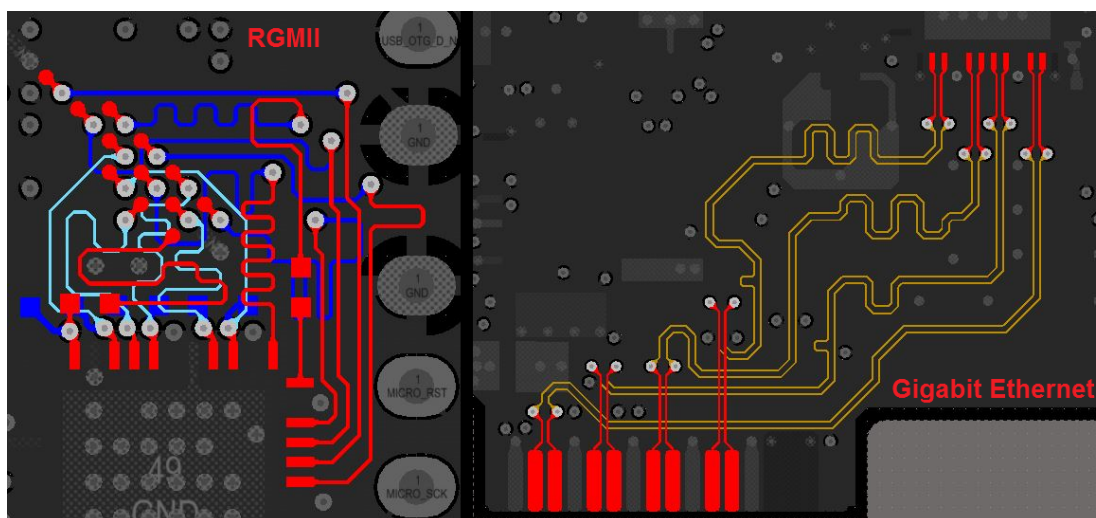
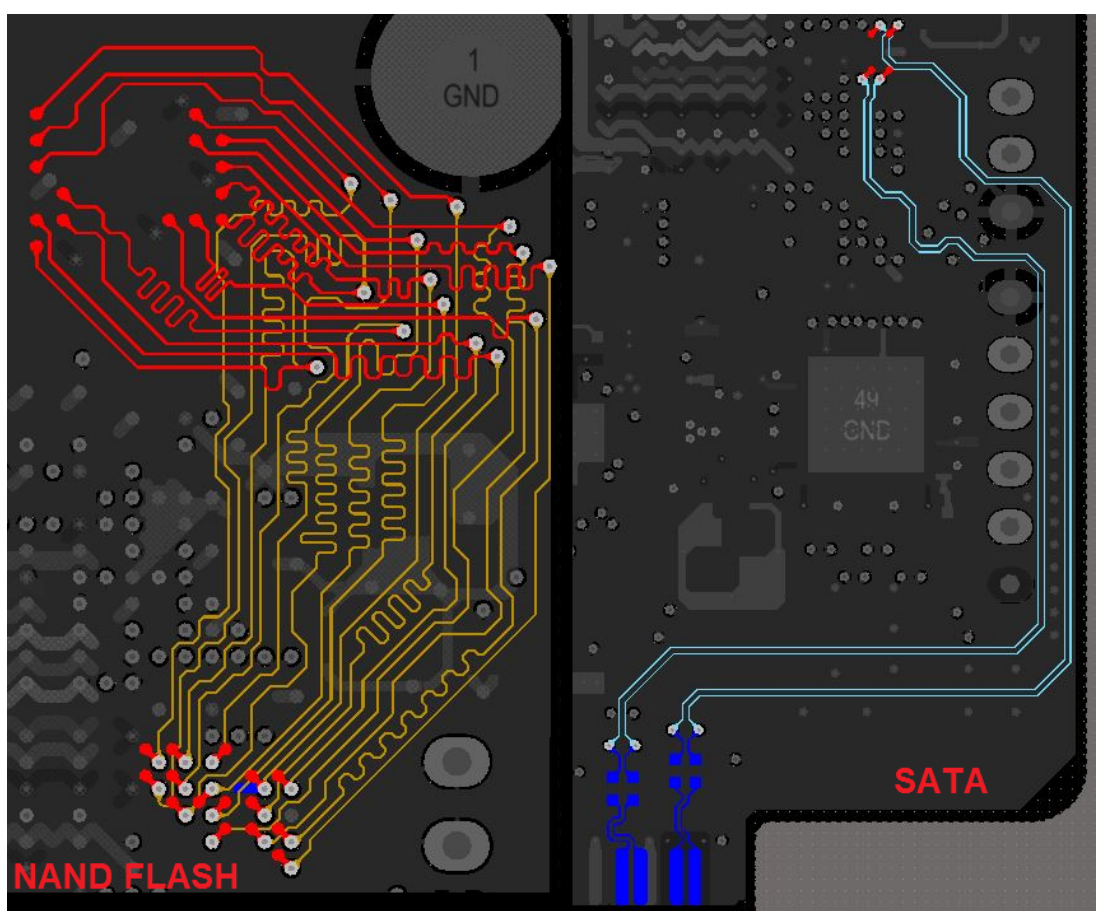


Figura 36 – Layout das interfaces Flash e SATA



### 5.3 Alimentação na Placa Mãe

O documento AN4509 (FREESCALE SEMICONDUCTOR, INC., 2012) apresenta o consumo de corrente de cada nível de tensão requerido pelo i.MX6. Resumido na Tabela 7 está o consumo de corrente máximo dos principais níveis de tensão da placa.

Tabela 7 – Máximo consumo de corrente dos principais níveis de tensão da Placa Mãe

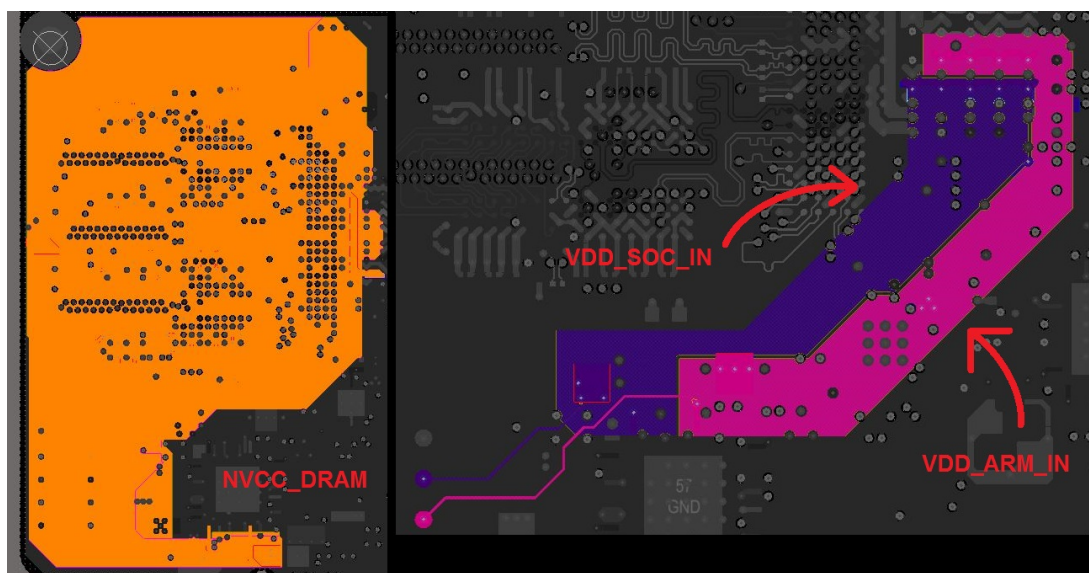
domínio de alimentação	tensão	corrente
VDD_ARM_IN	1,37V	1450mA
VDD_SOC_IN	1,37V	1250mA
NVCC_DRAM	1,5V	825mA

Como pode ser observado, a corrente é consideravelmente elevada, sendo necessário o uso de planos de alimentação e uma estimativa da quantidade de vias necessárias para conduzir a corrente.

Usando o software *Saturn PCB Design* é possível estimar a resistência elétrica de uma via e sua capacidade de corrente. Assumindo uma via com furo de diâmetro 0,25mm, espessura de metalização de 0,018mm (mesma espessura das camadas da placa) e comprimento total de 1mm (espessura total da placa), tem-se uma resistência de aproximadamente 1,2mΩ e uma capacidade de corrente de 830mA por via. O software usa como base de cálculo o documento IPC-2152 (BUNEA P. SVASTA, 2010).

Segundo essa estimativa, 2 vias por nível de tensão seriam o suficiente para a corrente máxima, mas por precaução foram utilizadas no mínimo 3 vias por nível de tensão. A Figura 37 apresenta o layout desses 3 níveis de tensão na Placa Mãe.

Figura 37 – Layout dos níveis de tensão com maior capacidade de corrente da Placa Mãe

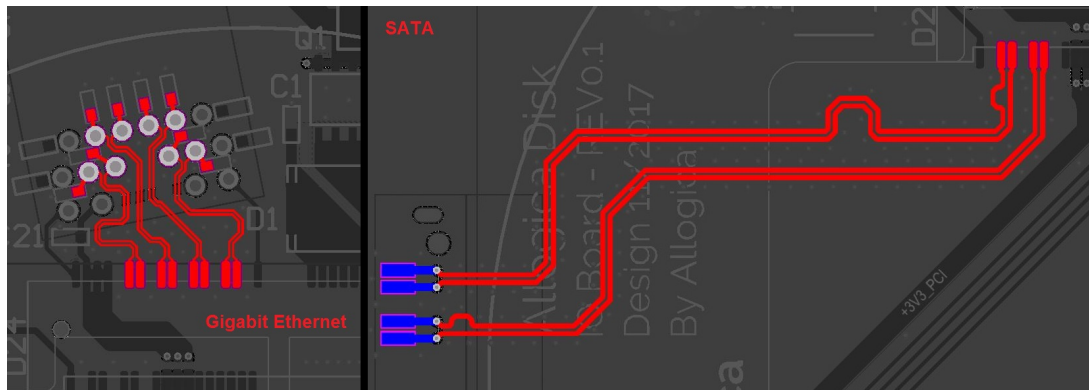




## 5.4 Interfaces de alta velocidade na Placa IO

Por apresentar apenas 2 camadas, a topologia guia de onda coplanar (*coplanar waveguide*) foi escolhida para as interfaces SATA e *Ethernet*, permitindo assim atingir o valor de impedância desejada com trilhas não muito largas. O resultado do layout consta na Figura 38.

Figura 38 – Layout dos níveis de tensão com maior capacidade de corrente da Placa Mãe



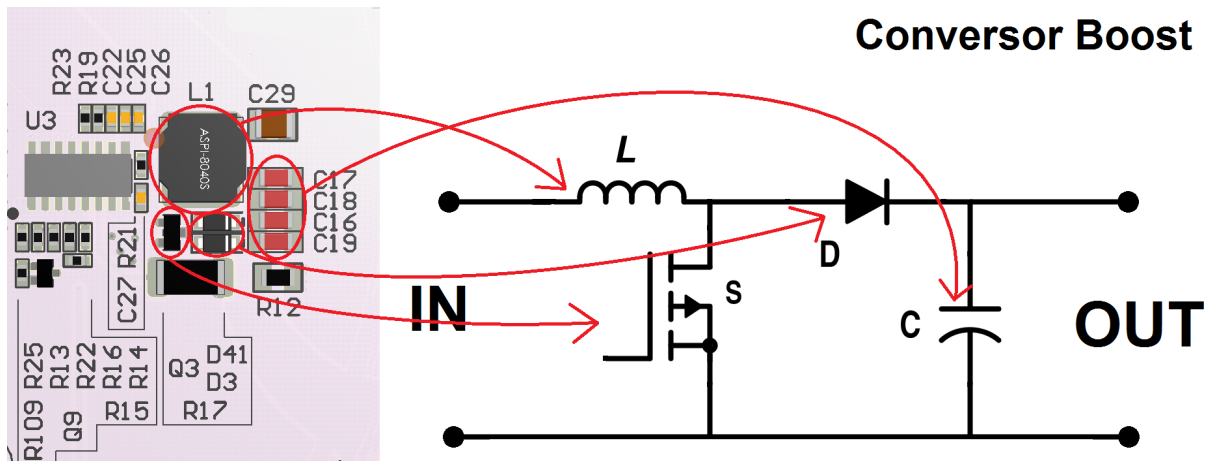
## 5.5 Driver dos LEDs na Placa IO

Conforme mencionado no Capítulo 4, o *driver* funciona como um conversor *boost* com 4 fontes de corrente (braços de corrente) comandadas por PWM para o acionamento dos LEDs. Para o layout deste circuito foram tomados 3 cuidados principais:

- Manter o nó comum ao indutor, transistor e diodo (onde existe a maior derivada de tensão) o mais curto possível para reduzir emissões eletromagnéticas;
- Manter a malha formada pelo transistor, diodo e capacitor de saída (onde existe a maior derivada de corrente) o menor possível para evitar o fenômeno de *ground bouncing*;
- Minimizar a indutância dos 4 braços de corrente do *driver*;

A Figura 39 mostra o *placement* dos componentes do *driver*, identificando as partes que compõem o conversor *boost*.

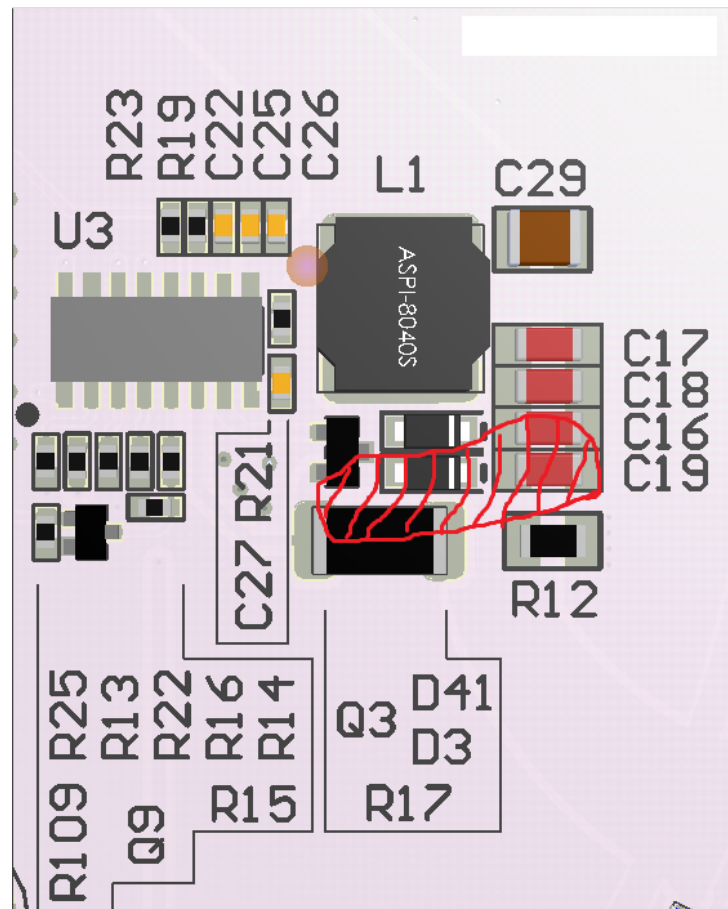
Figura 39 – Driver dos LEDs e comparação com conversor *boost*



A Figura 40 apresenta o nó com maior derivada de tensão do circuito e a Figura 41, a malha com maior derivada de corrente do circuito.

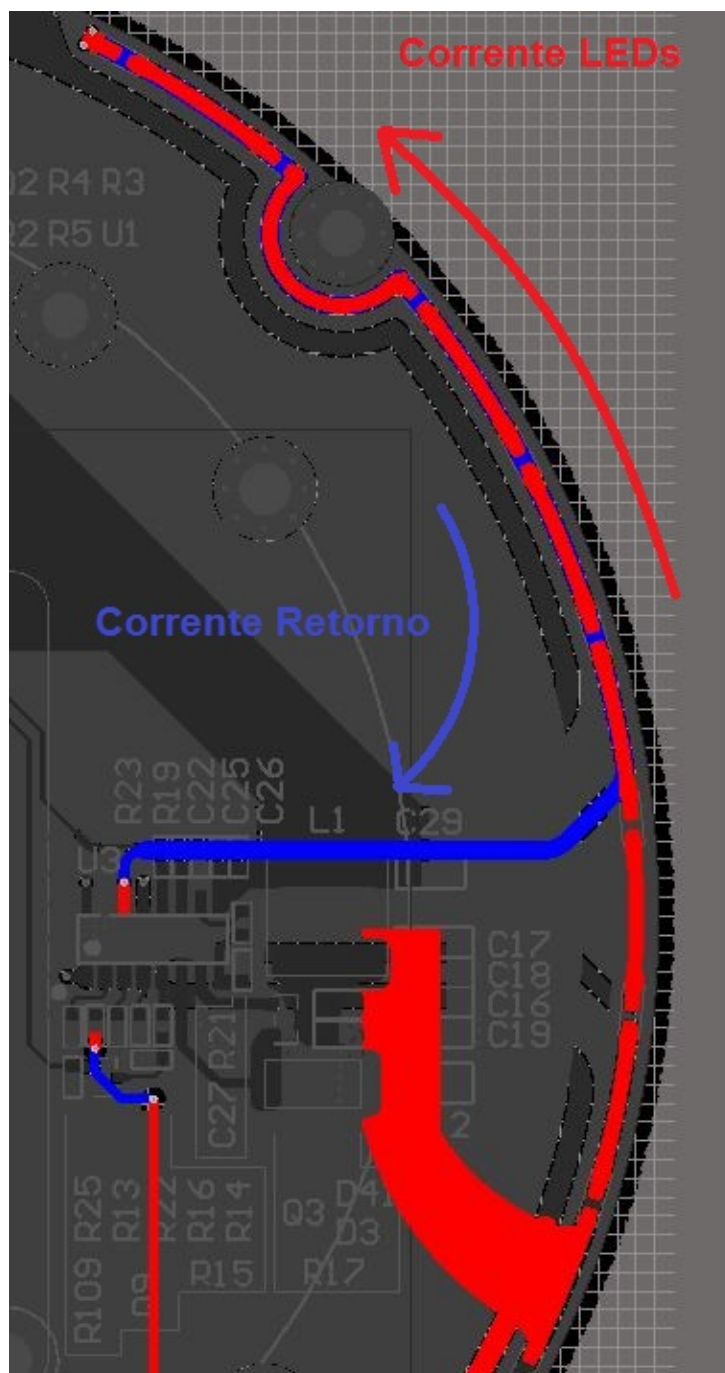
Figura 40 – Nó com maior derivada de tensão do *driver* - quanto menor, melhor



Figura 41 – Malha com maior derivada de corrente do *driver* - quanto menor, melhor

Por fim, a Figura 42 apresenta um dos braços de corrente do *driver*. As trilhas em vermelho estão na camada de cima da placa e as trilhas azuis, na camada de baixo. Pode-se perceber que o retorno da corrente (em azul) se dá por baixo dos LEDs, minimizando assim a indutância total da malha.

Figura 42 – Um dos 4 braços de corrente do *driver*



## 6 Testes do circuito

Durante a realização do projeto muitos testes e alterações foram realizados no circuito até chegar na sua versão atual. Esta seção sintetiza os testes mais relevantes realizados durante o projeto para a validação dos blocos do circuito.

### 6.1 Testes: Circuito de proteção de entrada e PMIC

#### Objetivos:

O PMIC (*Power Management Integrated Circuit*) funciona como regulador de tensão para o processador central e demais periféricos da Placa Mãe. Por este não apresentar um dispositivo de proteção próprio, foi desenvolvido também um circuito de proteção de entrada.

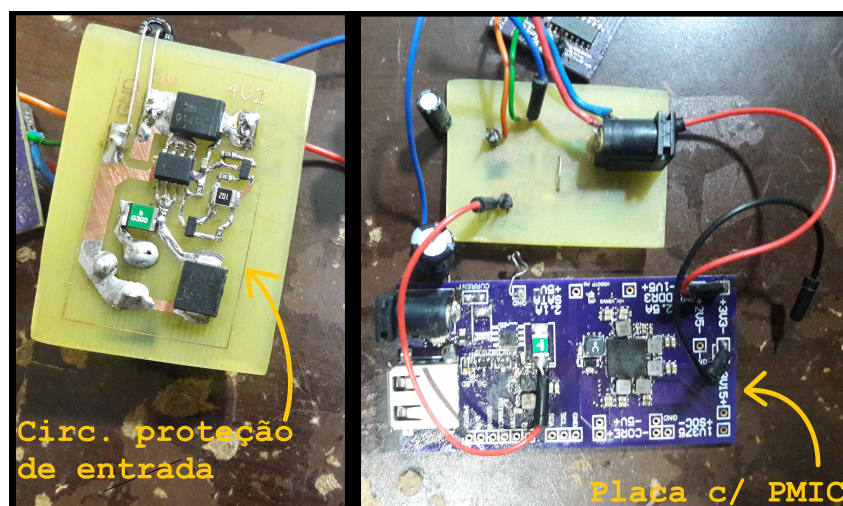
Os objetivos dos testes apresentados nesta seção são:

- Validar a proteção contra sobre-tensão;
- Validar a proteção contra polaridade reversa;
- Validar o bom funcionamento do PMIC;

#### Montagem e Etapas:

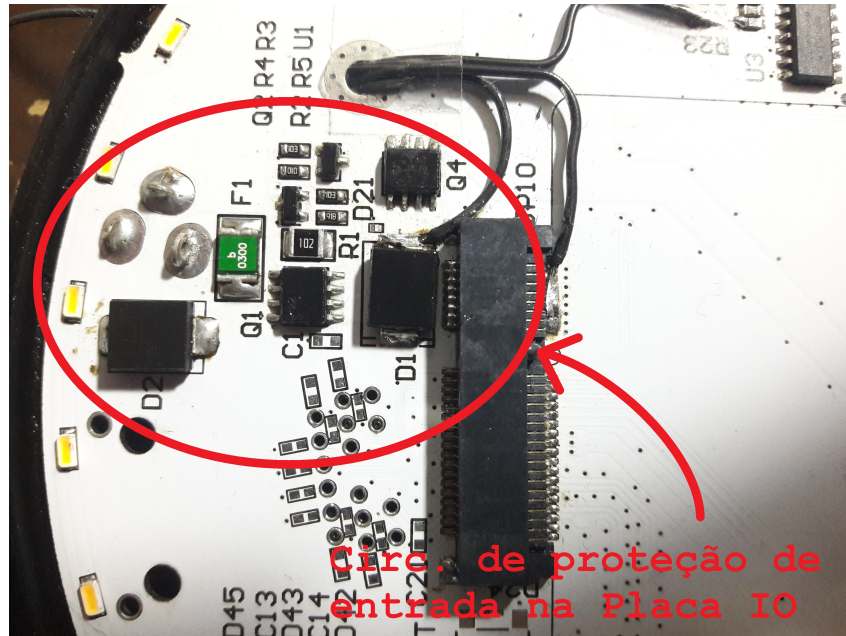
A Figura 43 mostra a montagem para os testes com o circuito de proteção de entrada e com o PMIC antes da realização da Placa IO.

Figura 43 – Montagem para testes com PMIC e circuito de proteção de entrada



Depois da realização da Placa IO, o circuito de proteção de entrada pôde ser testado novamente, como pode ser visto na montagem da Figura 44.

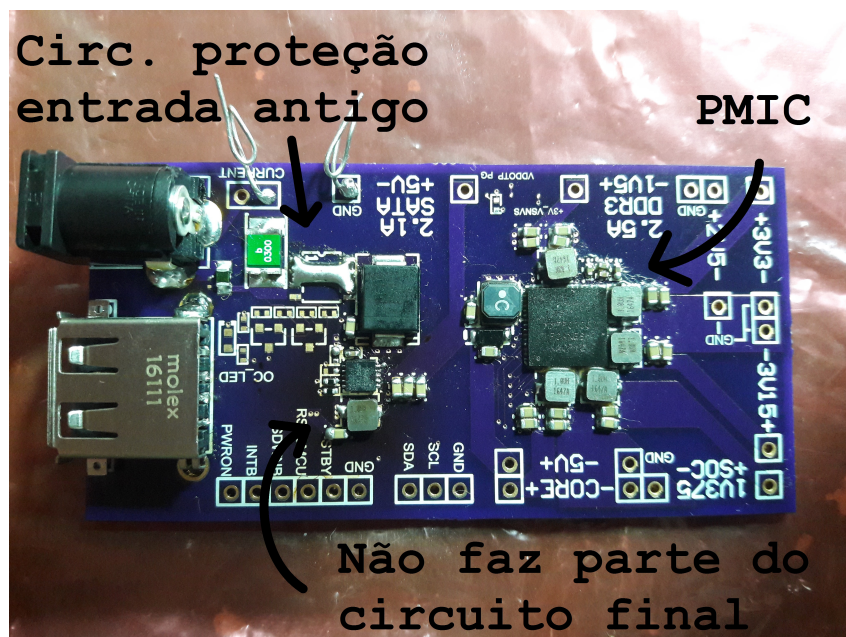
Figura 44 – Circuito de proteção de entrada na Placa IO



Testes e Resultados:

Para testar o PMIC, foi desenvolvida a placa da Figura 45. Nela pode ser visto o PMIC (MMPF0100F0) e uma primeira versão do circuito de proteção de entrada (também está presente na placa um circuito que não faz parte da versão final do produto).

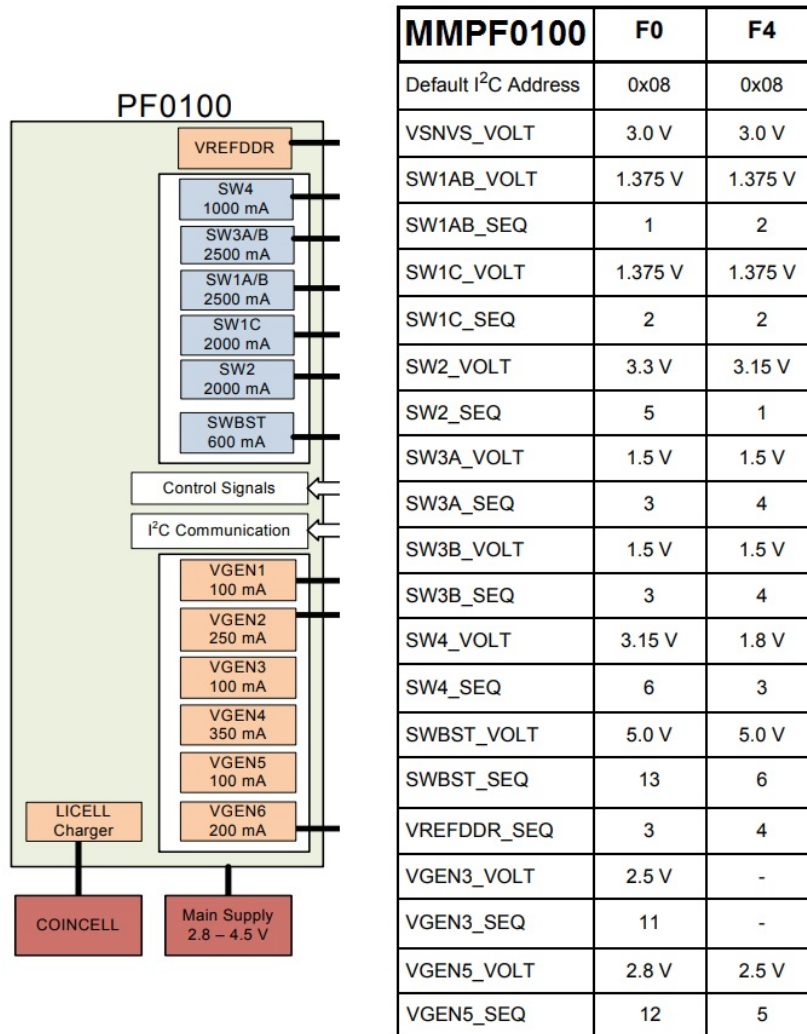
Figura 45 – Placa desenvolvida para testes com o PMIC



O PMIC foi então alimentado com uma tensão em torno de 4.5V, sendo medidos o nível de tensão e a ordem de acionamento das suas saídas com e sem carga.

Todos os reguladores do PMIC se comportaram conforme especificado no *datasheet* (NXP SEMICONDUCTORS, 2017), conforme mostra a Figura 46.

Figura 46 – Tabela de especificações - MMPF0100F0 e F4 (editado do *datasheet*)



Para os testes do circuito de proteção de entrada, em ambos os casos (antes e depois da realização da Placa IO), o circuito foi capaz de proteger o PMIC de sobre-tensões e polaridade reversa. Também foi incluído no circuito um supressor de transientes de tensão (TPSMC20CA), para proteção contra descargas eletrostáticas.

Discussão dos Resultados:

Ambos os circuitos funcionaram sem muitas surpresas. Antes dos testes acreditava-se que o PMIC, por ser um circuito complexo e com um encapsulamento difícil de soldar, apresentaria problemas, no entanto, ele acabou sendo o bloco mais simples de testar.

Os valores de alguns resistores do circuito de proteção de entrada tiveram que ser alterados durante os testes para levar em conta a tolerância das fontes de tensão de 5V comerciais (algumas fontes especificadas como sendo de 5V apresentavam uma tensão de 5,3V).

## 6.2 Testes: *Driver* dos LEDs

### Objetivos:

Os LEDs da Placa IO servem de indicação visual do *status* do dispositivo para o usuário. O *driver* deve ser capaz de controlar a luminosidade de todos os LEDs através de um sinal PWM vindo do microcontrolador.

O objetivo dos testes apresentados nesta seção é:

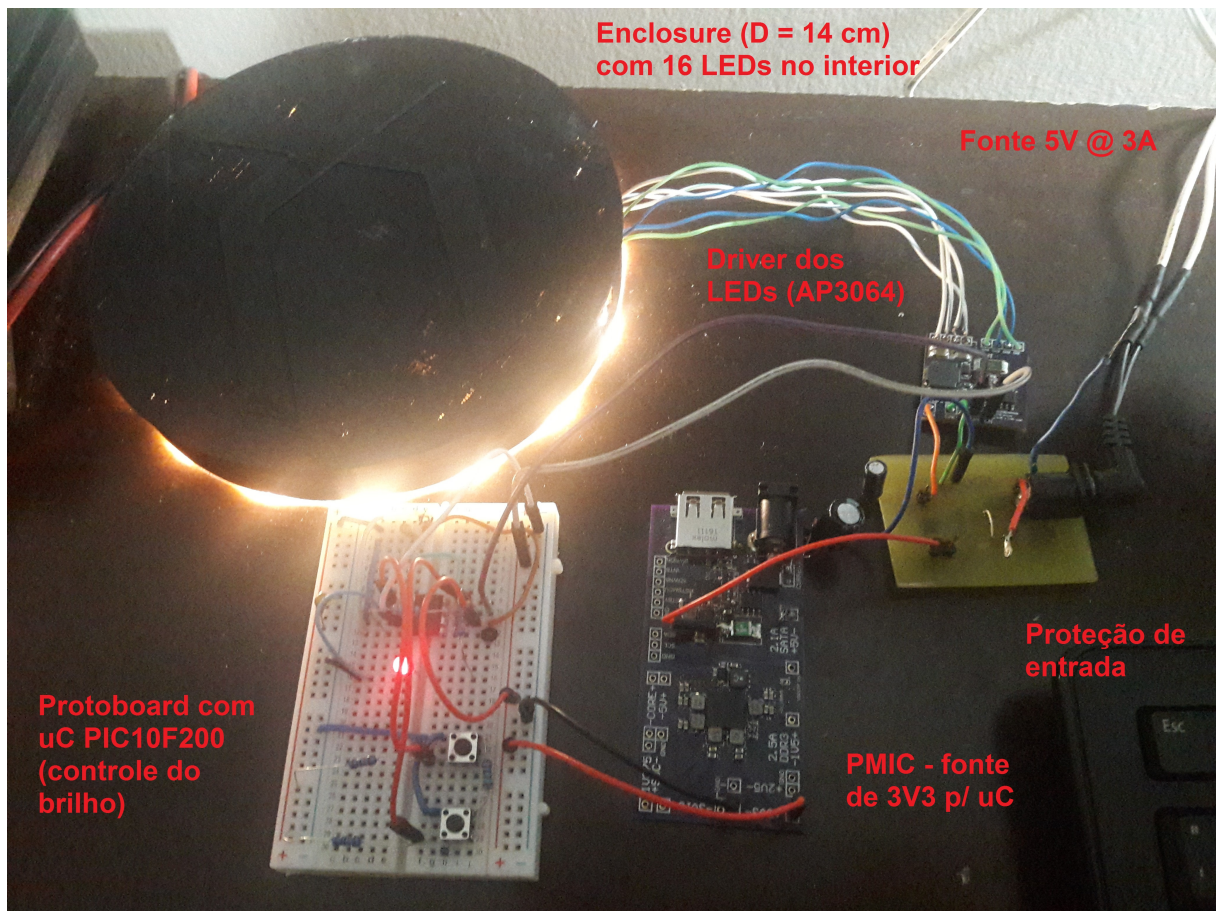
- Validar que o *driver* escolhido é capaz de controlar os 32 LEDs da placa com um sinal PWM;

### Montagem e Etapas:

Os primeiros testes com o circuito de *driver* foram realizados com 16 LEDs (vide Figura 47), como o número de LEDs não era suficiente para ter um brilho uniforme, o número de LEDs foi aumentado para 32.



Figura 47 – Montagem para teste com 16 LEDs



#### Testes e Resultados:

A Figura 48 apresenta os resultados com 16 LEDs e a Figura 49, os resultados com 32 LEDs.

Figura 48 – resultados obtidos com 16 LEDs

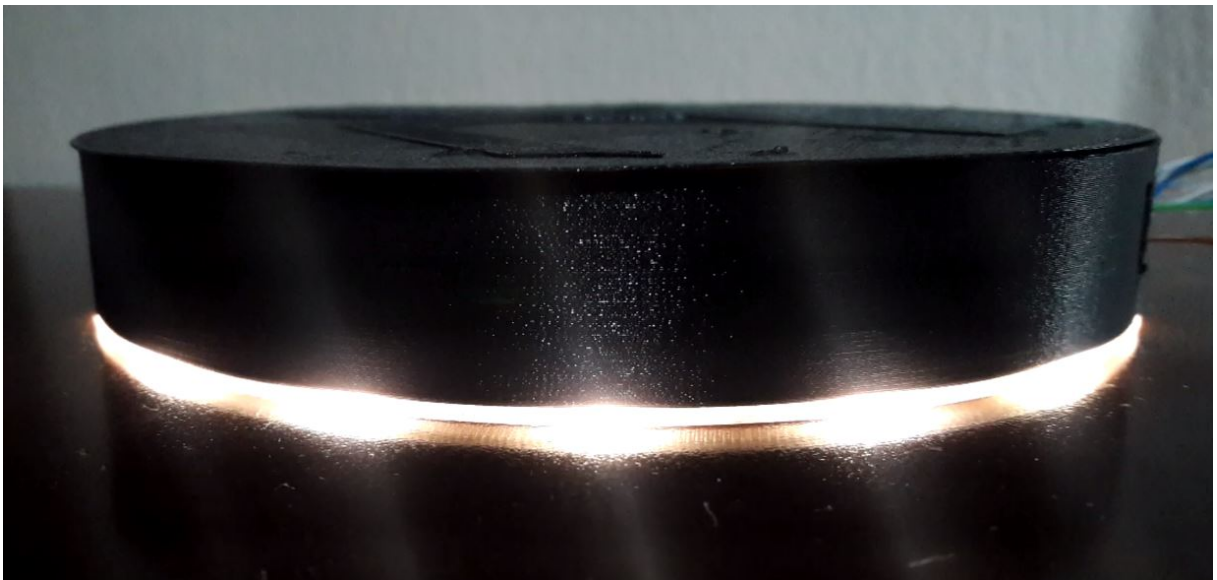
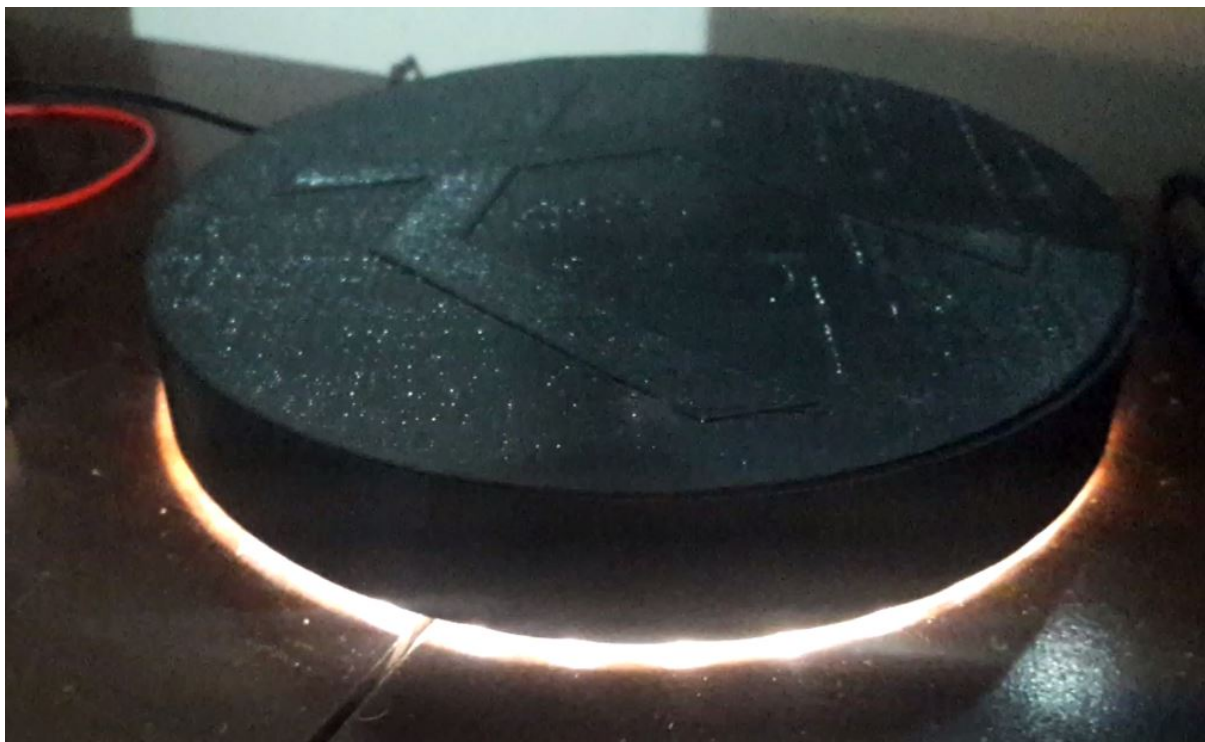


Figura 49 – resultados obtidos com 32 LEDs



#### Discussão dos Resultados:

O circuito com 32 LEDs não funcionou na primeira tentativa, sendo necessário aumentar a frequência do conversor de 520kHz para (aproximadamente) 765kHz. A melhor explicação encontrada para esse fenômeno é que uma frequência de 520kHz combinada com um indutor de  $10\mu\text{H}$  não seria suficiente para que o conversor opere em Modo de Condução Contínua com 32 LEDs, de acordo com (BCD SEMICONDUCTOR MANUFACTURING LIMITED, 2011).

Também foi incluída uma resistência na saída do *driver* (em paralelo com os LEDs) para descarregar os capacitores de saída quando o PWM tendia a zero, garantindo que os LEDs sejam instantaneamente desligados.

Corrigidos esses problemas, o circuito funcionou como esperado, como pode ser visto na Figura 49, validando assim esse bloco do circuito.

### 6.3 Testes: Interface microcontrolador e i.MX6

#### Objetivos:

Para garantir o controle dos LEDs enquanto o processador central (i.MX6) está inicializando, foi utilizado o microcontrolador de 8bits ATTINY2313. O microcontrolador pode se comunicar com o processador central através do protocolo SPI (*Serial Peripheral Interface*) para saber como controlar os LEDs.

O objetivo dos testes apresentados nesta seção é:

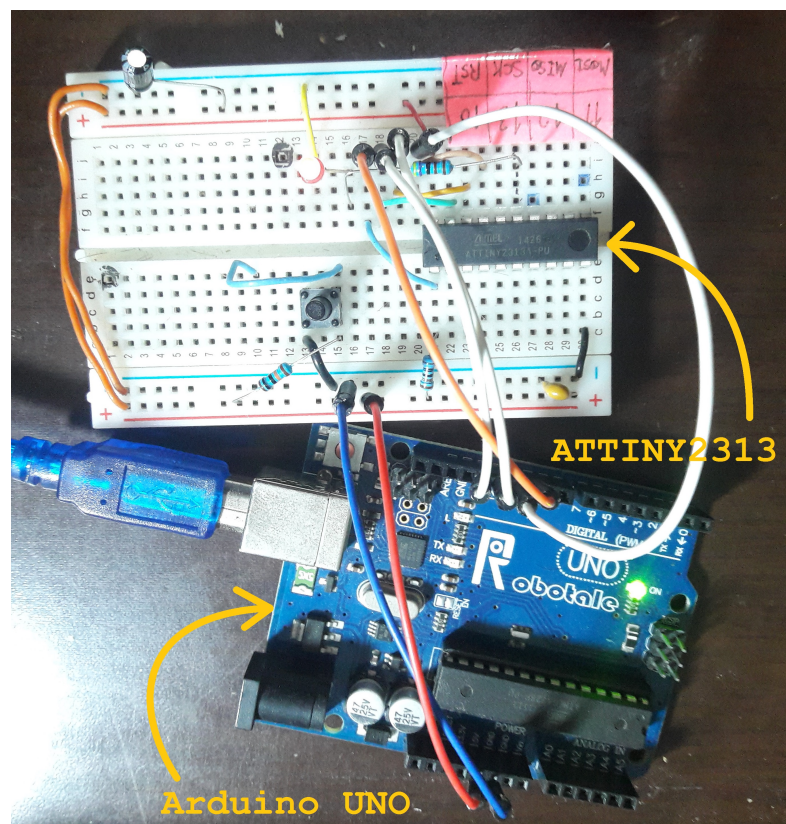
- Validar o funcionamento do código do microcontrolador.

#### Montagem e Etapas:

Para esse teste foi utilizado o microcontrolador ATTINY2313 em uma *protoboard* e um *Arduino UNO* para simular o processador central.

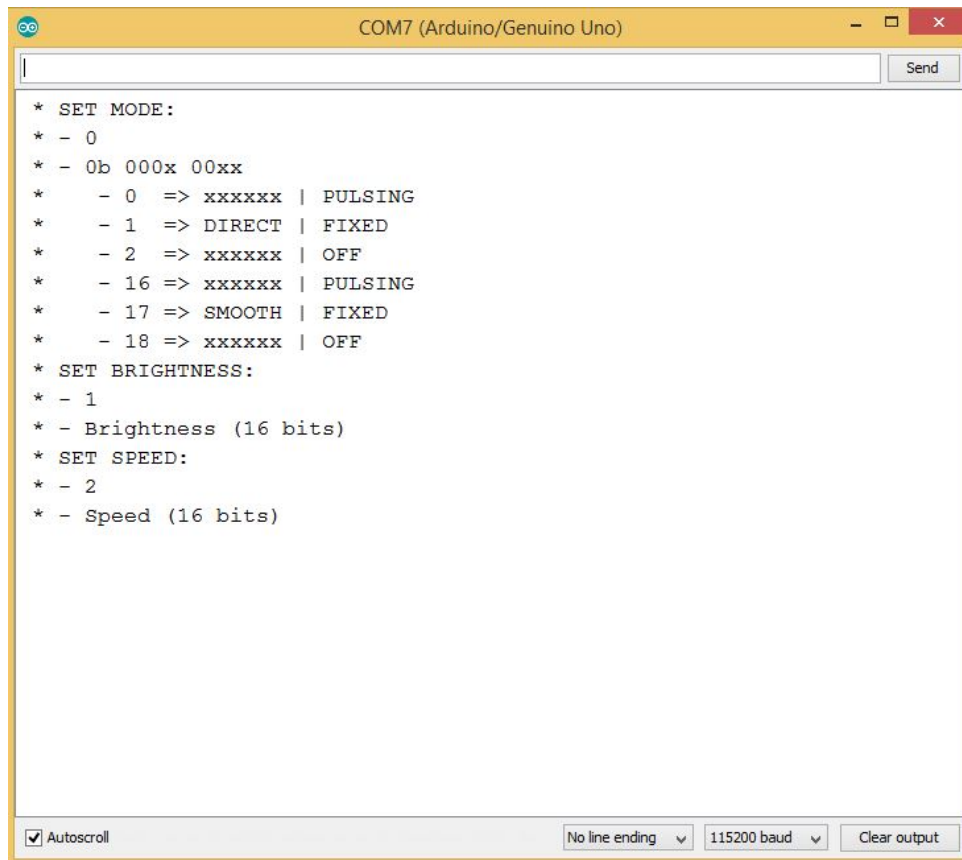
A Figura 50 mostra a montagem para o teste.

Figura 50 – Montagem para teste do código do microcontrolador



#### Testes e Resultados:

Para o teste foi desenvolvido um programa que permite o envio de comandos pelo protocolo SPI através do *Arduino UNO* usando a interface serial apresentada na Figura 51.

Figura 51 – Interface serial para transmissão de comandos SPI pelo *Arduino*

```
COM7 (Arduino/Genuino Uno)
Send

* SET MODE:
* - 0
* - 0b 000x 00xx
* - 0 => xxxxxx | PULSING
* - 1 => DIRECT | FIXED
* - 2 => xxxxxx | OFF
* - 16 => xxxxxx | PULSING
* - 17 => SMOOTH | FIXED
* - 18 => xxxxxx | OFF
* SET BRIGHTNESS:
* - 1
* - Brightness (16 bits)
* SET SPEED:
* - 2
* - Speed (16 bits)

 Autoscroll
No line ending 115200 baud Clear output
```

Com a interface serial os vários modos de operação do microcontrolador puderam ser facilmente testados. Durante os testes foi observado o bom funcionamento do código responsável pelo controle dos LEDs, validando assim essa parte do circuito.

#### Discussão dos Resultados:

O uso da interface serial foi bastante útil para a validação do código, uma vez que facilitou a identificação e posterior correção de *bugs*.

O código do microcontrolador ATTINY2313 para o acionamento dos LEDs ocupou quase toda a memória de programa deste (de 2 KB).

## 6.4 Testes: Placa IO - SATA

#### Objetivos:

As trilhas da interface SATA conectam o processador central (i.MX6) e o HD. Durante todo o percurso entre os 2 dispositivos a integridade do sinal deve ser mantida para garantir uma transferência de dados estável.

Os objetivos dos testes apresentados nesta seção são:

- Validar que as trilhas da Placa IO não degradam os sinais da interface SATA;
- Em caso de perda de performance com a Placa IO, encontrar as causas para tal fenômeno;

#### Montagem e Etapas:

Para esse teste foi utilizado o kit de desenvolvimento para o chip i.MX6, *WandBoard*, e 2 montagens foram realizadas, uma sem a Placa IO e outra com. Para ambos os casos foram medidas a velocidade de leitura e escrita no HD. Os resultados foram então comparados.

A Figura 52 mostra a montagem sem a Placa IO (como medida de referência) e a Figura 53, a montagem com a Placa IO.

Na montagem de referência a *WandBoard* foi conectada diretamente ao HD com um cabo SATA. Na segunda montagem, a Placa IO é colocada entre a *WandBoard* e o HD. Uma placa de adaptação (chamada Placa PCI) foi realizada para ligar o cabo SATA à Placa IO, como pode ser visto na Figura 53.

Figura 52 – Montagem para teste SATA - sem Placa IO

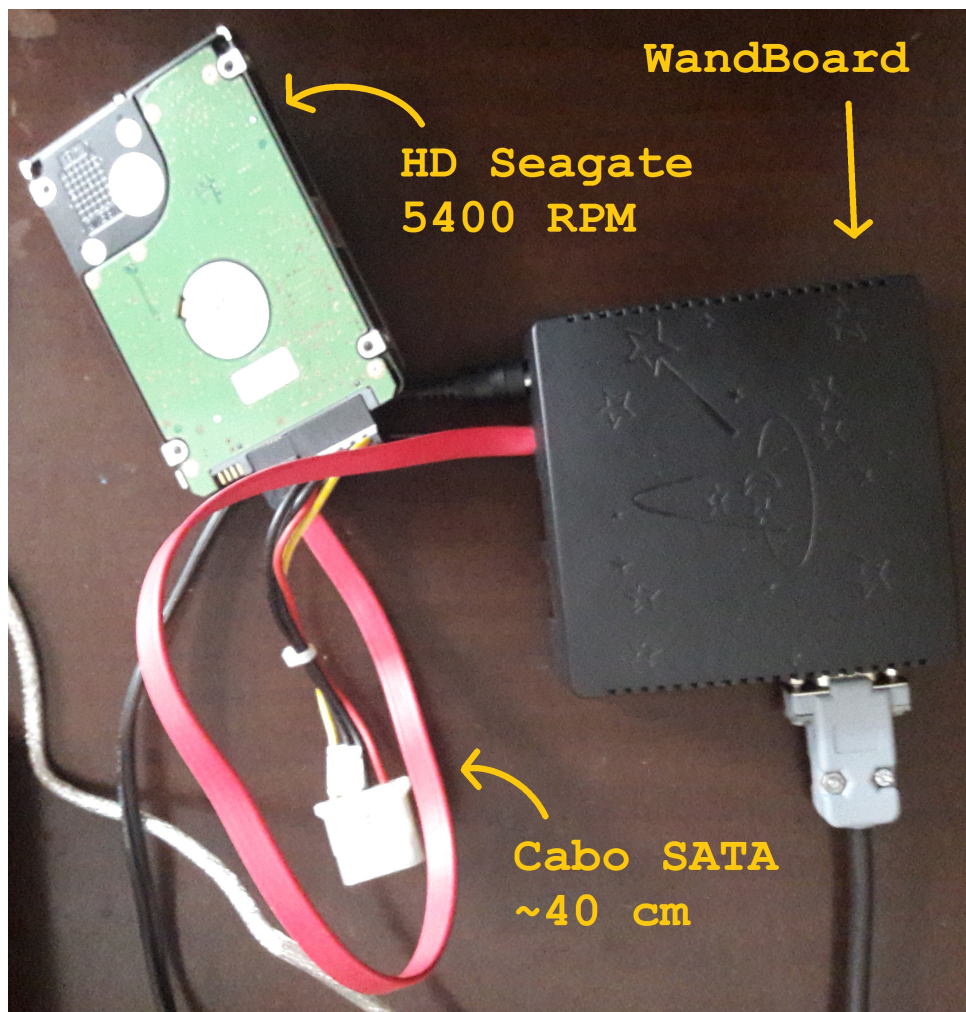
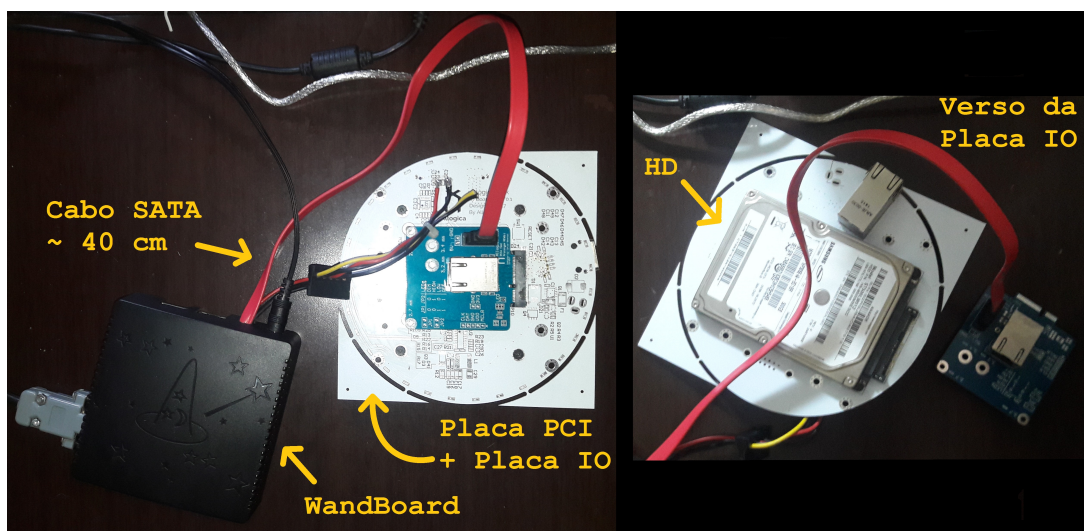


Figura 53 – Montagem para teste SATA - com Placa IO

Testes e Resultados:

Para os testes foi utilizado o comando `<dd>`, um utilitário de linha de comando de sistemas Unix (e.g. Linux e FreeBSD) que permite converter e copiar arquivos, que foi utilizada nesse teste para medir a velocidade de leitura/escrita no HD. Os comando usados nos testes seguem a seguinte estrutura: `<dd if=/dev/zero of=/dev/ada0 bs=BYTES count=NB_BLOCOS>`.

Ele copia o conteúdo de `/dev/zero` (arquivo contendo zeros) para `/dev/ada0` (para o HD) em `NB_BLOCOS` blocos de `BYTES` Bytes.

Os resultados dos testes de leitura constam na Tabela 8 e os resultados dos testes de escrita, na Tabela 9.

Tabela 8 – Resultados - leitura do HD

	sem Placa IO	com Placa IO	unidade
<b>5 blocos de 512 MB</b>	78	78	<i>Mbytes/s</i>
<b>1000 blocos de 1 MB</b>	78	79	<i>Mbytes/s</i>
<b>5 blocos de 800 MB</b>	78	78.7	<i>Mbytes/s</i>

Tabela 9 – Resultados - escrita do HD

	sem Placa IO	com Placa IO	unidade
<b>5 blocos de 512 MB</b>	90	89.5	<i>Mbytes/s</i>
<b>1000 blocos de 1 MB</b>	80	79	<i>Mbytes/s</i>
<b>5 blocos de 800 MB</b>	95	94.3	<i>Mbytes/s</i>

Discussão dos Resultados:

Pode-se perceber que a presença da Placa IO não afeta significativamente a velocidade de leitura/escrita do HD. As trilhas estão então validadas.

Também foi testada a performance das trilhas com o software de *benchmark* IOzone. Os resultados com e sem a Placa IO foram bastante similares, por isso os valores medidos foram omitidos.

## 6.5 Testes: Placa IO - Ethernet

### Objetivos:

De forma similar ao que foi feito nos testes SATA, nesta seção foram realizados os testes que validam as trilhas da interface Ethernet da Placa IO.

Os objetivos dos testes apresentados nesta seção são:

- Validar que as trilhas da Placa IO não degradam os sinais da interface Ethernet;
- Em caso de perda de performance com a Placa IO, encontrar as causas para tal fenômeno;

### Montagem e Etapas:

Para esse teste também foi utilizado o kit de desenvolvimento para o chip i.MX6, *WandBoard*, e 2 montagens foram realizadas, uma sem a Placa IO e outra com. Para ambos os casos foi medida a velocidade de transferência entre a *WandBoard* e o PC. Os resultados foram então comparados.

A Figura 54 mostra a montagem sem a Placa IO (como medida de referência) e a Figura 55, a montagem com a Placa IO.

Na montagem de referência a *WandBoard* é conectada diretamente ao PC com um cabo de rede. Na segunda montagem, a Placa IO é colocada entre a *WandBoard* e o PC (para ligar a *WandBoard* à Placa IO foi usada a Placa PCI, como pode ser visto na Figura 55).

Figura 54 – Montagem para teste Ethernet - sem Placa IO

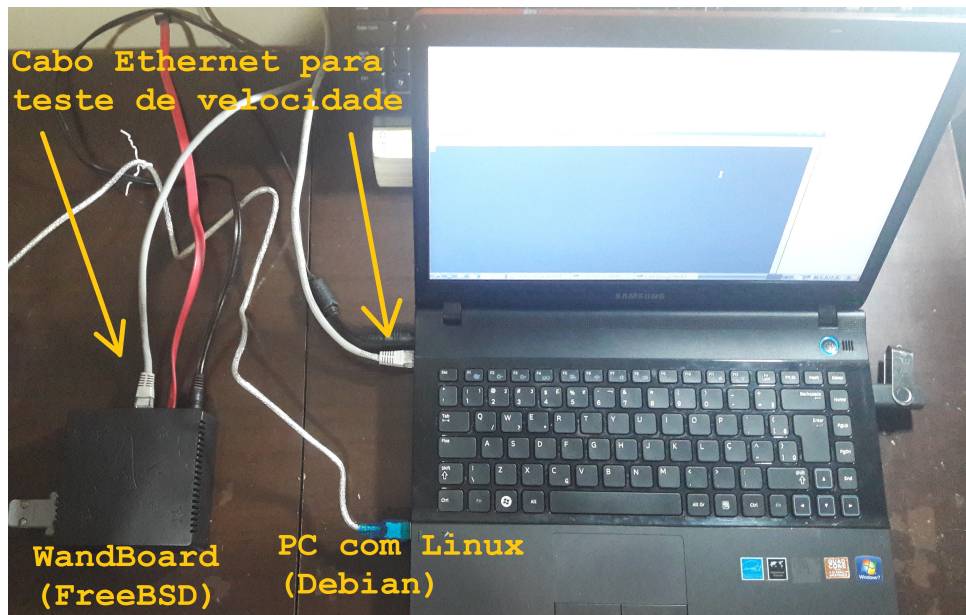
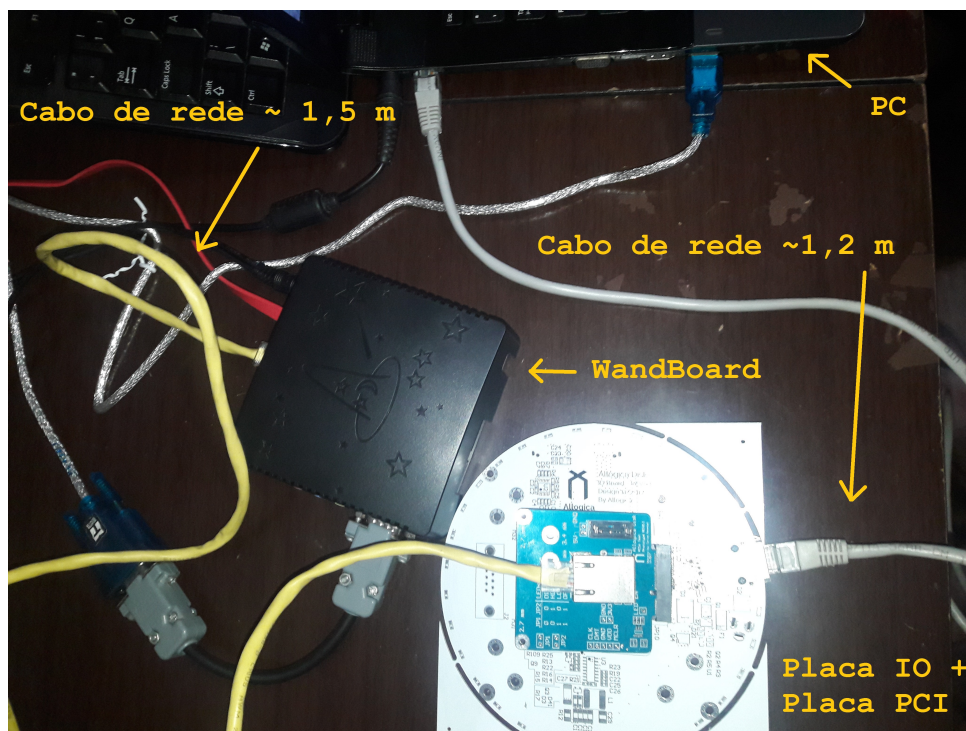


Figura 55 – Montagem para teste Ethernet - com Placa IO



### Testes e Resultados:

Para os testes, o PC foi configurado como um servidor TCP com o comando Linux `<iperf -s -w 56k>`. Para testar a velocidade de conexão foi utilizado, na *WandBoard*, o comando `<iperf -c 192.168.10.1 -w 56k -t 50 -i 5>`, que conecta ao servidor (de IP = 192.168.10.1) como cliente e executa um teste de velocidade de 50 segundos, reportando o resultado a cada 5 segundos.



Os resultados sem Placa IO constam na Figura 56 e com a Placa IO, na Figura 57.

Figura 56 – Resultados dos testes Ethernet - sem Placa IO

```

root@Rogiel:~# iperf -c 192.168.10.1 -w 56k -t 50 -i 5
-----
Client connecting to 192.168.10.1, TCP port 5001
TCP window size: 55.6 KByte (WARNING: requested 54.7 KByte)
-----
[ 3] local 192.168.10.2 port 64091 connected with 192.168.10.1 port 5001
[ ID] Interval      Transfer      Bandwidth
[ 3]  0.0- 5.0 sec    262 MBytes    440 Mbits/sec
[ 3]  5.0-10.0 sec   263 MBytes    441 Mbits/sec
[ 3] 10.0-15.0 sec   263 MBytes    441 Mbits/sec
[ 3] 15.0-20.0 sec   263 MBytes    441 Mbits/sec
[ 3] 20.0-25.0 sec   263 MBytes    441 Mbits/sec
[ 3] 25.0-30.0 sec   263 MBytes    441 Mbits/sec
[ 3] 30.0-35.0 sec   263 MBytes    441 Mbits/sec
[ 3] 35.0-40.0 sec   263 MBytes    441 Mbits/sec
[ 3] 40.0-45.0 sec   263 MBytes    441 Mbits/sec
[ 3] 45.0-50.0 sec   263 MBytes    441 Mbits/sec
[ 3] 0.0-50.0 sec   2.57 GBytes    441 Mbits/sec
root@Rogiel:~#

```

Figura 57 – Resultados dos testes Ethernet - com Placa IO

```

root@Rogiel:~# ifconfig ffec0 192.168.10.2
root@Rogiel:~# iperf -c 192.168.10.1 -w 56k -t 50 -i 5
-----
Client connecting to 192.168.10.1, TCP port 5001
TCP window size: 55.6 KByte (WARNING: requested 54.7 KByte)
-----
[ 3] local 192.168.10.2 port 16005 connected with 192.168.10.1 port 5001
[ ID] Interval      Transfer      Bandwidth
[ 3]  0.0- 5.0 sec    262 MBytes    440 Mbits/sec
[ 3]  5.0-10.0 sec   263 MBytes    441 Mbits/sec
[ 3] 10.0-15.0 sec   263 MBytes    441 Mbits/sec
[ 3] 15.0-20.0 sec   263 MBytes    441 Mbits/sec
[ 3] 20.0-25.0 sec   263 MBytes    441 Mbits/sec
[ 3] 25.0-30.0 sec   263 MBytes    441 Mbits/sec
[ 3] 30.0-35.0 sec   263 MBytes    441 Mbits/sec
[ 3] 35.0-40.0 sec   263 MBytes    441 Mbits/sec
[ 3] 40.0-45.0 sec   262 MBytes    439 Mbits/sec
[ 3] 45.0-50.0 sec   262 MBytes    440 Mbits/sec
[ 3] 0.0-50.0 sec   2.57 GBytes    441 Mbits/sec
root@Rogiel:~#

```

#### Discussão dos Resultados:

Mesmo com a Placa IO e um cabo de rede a mais, os resultados dos testes foram muito próximos, validando assim as trilhas da interface Ethernet da Placa IO.

Analisando a Errata do chip i.MX6: (NXP SEMICONDUCTORS, 2016), percebe-se que a velocidade da interface *Ethernet* está limitada a um valor máximo de 470 Mbps por conta de um bug no hardware, o que explica as velocidades obtidas durante os testes.

## 7 Conclusão

Este trabalho apresentou o projeto e desenvolvimento de um *Network Attached Storage* (NAS). No Capítulo 3 as especificações do produto foram definidas baseado na pesquisa realizada no Anexo A. Um circuito capaz de atender as especificações foi desenvolvido e apresentado em blocos funcionais no Capítulo 4.

O layout das duas placas que compõem o hardware do dispositivo foi completamente finalizado e as partes mais importantes de cada uma foram apresentadas no Capítulo 5, onde as interfaces de alta velocidade do circuito (como SDRAM DDR3, SATA e *Ethernet*) foram roteadas seguindo as recomendações de layout do fabricante do processador central do circuito.

Os testes realizados na Placa IO validaram o funcionamento da totalidade do circuito. O *driver* é capaz de acionar todos os 32 LEDs da placa; O circuito de proteção de entrada oferece proteção contra sobretensão e polaridade reversa; O microcontrolador é capaz de se comunicar com o i.MX6 através do protocolo SPI para decidir como controlar a luminosidade dos LEDs; As linhas de transmissão das interfaces SATA e *Ethernet* mantêm a integridade do sinal até os seus respectivos conectores.

O design da Placa Mãe foi totalmente finalizado e está em desenvolvimento, restando como trabalho futuro a integração para obter o produto minimamente viável. Validado o primeiro protótipo, testes automatizados do circuito podem ser pensados em conjunto com a empresa escolhida para fazer a montagem em escala do produto. Durante a produção, as placas devem ser programadas pela empresa montadora, carregando, junto do *firmware*, um programa capaz de testar todas as funcionalidades do produto automaticamente. Testes de estresse serão provavelmente realizados depois de receber as placas montadas.

## 8 Proposta de trabalhos futuros

Apesar de bastante popular e de contar com extensa documentação, o processador escolhido para o projeto, i.MX6D, é caro e seu uso acaba forçando escolhas de design que encarecem ainda mais o produto final (como o uso de componentes de dimensões 0201 e placa de circuito impresso de 8 camadas). A principal proposta para projetos futuros seria encontrar um microprocessador de menor custo. Bons candidatos para substituto seriam o SoC LS1012, do fabricante NXP e o SoC A20, do fabricante *Allwinner*.

Outra proposta para futuras versões do produto seria substituir os LEDs indicadores atuais, que são brancos e só podem ser controlados em conjunto, por LEDs multi-cor e capazes de ser controlados individualmente (uma solução usando os LEDs WS2812 seria viável, mas também aumentaria o custo final do produto). Para isso seria necessário substituir também o microcontrolador por um mais potente.

Finalmente, uma última proposta para futuras versões seria incluir uma porta USB *Host* para possibilitar a integração do dispositivo com um HD externo.

# Referências Bibliográficas

ACADEMY, F. *REX, NXP (Freescale) iMX6 Opensource Projects*. 2018. Disponível em: <<http://www.imx6rex.com/>>.

BCD SEMICONDUCTOR MANUFACTURING LIMITED. *WLED Backlighting Solution for Medium-sized LED Panel Designed with AP3064*. [S.l.], 2011. Rev. 1.0.

BROOKS, D. Differential impedance. *Printed Circuit Design, a Miller Freeman publication*, 1998.

BUNEA P. SVASTA, N. C. I. P. C. I. R. Ipc-2152 – standard for determining current-carrying capacity in printed board design. *Circuit World, Vol. 36 Issue: 1*, 2010.

DEVICES, A. *Reducing Ground Bounce in DC-to-DC Converters—Some Grounding Essentials*. 2018. Disponível em: <<http://www.analog.com/en/analog-dialogue/articles/reducing-ground-bounce-in-dc-to-dc-converters.html>>.

FREESCALE SEMICONDUCTOR, INC. *i.MX 6Dual/6Quad Power Consumption Measurement*. [S.l.], 2012. Rev. 0.

FREESCALE SEMICONDUCTOR, INC. *Hardware Development Guide for i.MX 6Quad, 6Dual, 6DualLite, 6Solo Families of Applications Processors*. [S.l.], 2013. Rev. 1.

FREESCALE SEMICONDUCTOR, INC. *i.MX 6Dual/6Quad Applications Processors for Consumer Products*. [S.l.], 2015. Rev. 4.

INTEGRATED, M. *Layout Considerations for Non-Isolated DC-DC Converters*. 2018. Disponível em: <<https://www.maximintegrated.com/en/app-notes/index.mvp/id/735>>.

MICROWAVES101. *Microwaves101 calculator*. 2018. Disponível em: <<https://www.microwaves101.com/calculators/>>.

NXP SEMICONDUCTORS. *Chip Errata for the i.MX 6Dual/6Quad and i.MX 6DualPlus/6QuadPlus*. [S.l.], 2016. Rev. 6.1.

NXP SEMICONDUCTORS. *14 channel configurable power management integrated circuit*. [S.l.], 2017. Rev. 17.0.

PANASONIC. *High Speed, Low Loss Multi-layer Materials*. [S.l.], 2013. Megtron 4.

POZAR, D. M. *Microwave Engineering*. 2nd. ed. [S.l.]: John Wiley & Sons, Inc., 1998. ISBN 0-471-17096-8.

WIKIPEDIA. *DDR SDRAM*. 2018. Disponível em: <[https://en.wikipedia.org/wiki/DDR\\_SDRAM](https://en.wikipedia.org/wiki/DDR_SDRAM)>.

WIKIPEDIA. *Transmission line*. 2018. Disponível em: <[https://en.wikipedia.org/wiki/Transmission\\_line](https://en.wikipedia.org/wiki/Transmission_line)>.

# ANEXO A – Dispositivos NAS existentes no mercado

É sempre bom ter como ponto de partida para qualquer projeto o estudo das soluções já existentes para o problema enfrentado. Uma pesquisa de mercado foi então realizada para permitir uma melhor definição das funcionalidades, escolha dos componentes principais e precificação do produto final.

Devido à grande quantidade de soluções NAS que as empresas concorrentes apresentam, cada seção desse capítulo foi destinada a uma empresa, Western Digital, Synology, D-Link, ASUSTOR, Seagate e QNAP.

Em cada seção, o(s) dispositivo(s) NAS que mais se aproxima(m) do dispositivo desenvolvido neste projeto (Allogica Disk) foram detalhados (especificações técnicas, preço, funcionalidades, etc).

O capítulo termina então com uma comparação entre todos eles e as conclusões tiradas desse estudo.

## A.1 Dispositivos NAS - Western Digital

Western Digital é uma empresa norte-americana fabricante de dispositivos de armazenamento, como HDs e seus derivados (HDs externos, dispositivos NAS, etc).

Dos dispositivos NAS da empresa, é interessante destacar aqui os produtos da linha *My Cloud*, discriminados na Tabela 10.

Tabela 10 – Comparação dispositivos NAS Western Digital

	My Cloud	My Cloud Home	My Cloud Ex2
<b>Preço EUA (U\$)</b>	150,00 (2TB)	160,00	160,00
<b>Preço BR (R\$)</b>	1400,00 (4TB)	Não vendido no Brasil	1 222,00
<b>Processador</b>	Armada 385 - Dual Core @ 1,3GHz	Não informado	Armada 385 - Dual Core @ 1,3GHz
<b>RAM (GB)</b>	Não informado	Não informado	1
<b>Nº de baias</b>	1	1	2
<b>Nº conectores Ethernet</b>	1	1	1
<b>Nº USB 2.0</b>	0	0	0
<b>Nº USB 3.0</b>	1	1	2

## A.2 Dispositivos NAS - Synology

A Synology é uma empresa que, entre outros, fabrica dispositivos NAS, roteadores e oferece soluções de vigilância IP (sistema de monitoramento baseado em Internet). Dentre seus muitos dispositivos NAS, destaco aqui o DS115j, DS116 e DS216.

Tabela 11 – Comparação dispositivos NAS Synology

	DS115j	DS116	DS216
<b>Preço EUA (U\$)</b>	117,00	180,00	260,00
<b>Preço BR (R\$)</b>	Não vendido no Brasil	Não vendido no Brasil	1 860,00
<b>Processador</b>	Armada 370 - Dual Core @ 800 MHz	Armada 385 - Dual Core @ 1,8GHz	Armada 385 - Dual Core @ 1,3GHz
<b>RAM (GB)</b>	0,256	1	0,512
<b>Nº de baias</b>	1	1	2
<b>Nº conectores Ethernet</b>	1	1	1
<b>Nº USB 2.0</b>	0	0	1
<b>Nº USB 3.0</b>	1	2	2

## A.3 Dispositivos NAS - D-Link

A empresa D-Link começou como fabricante de adaptadores de rede e hoje (2018), dentre outras coisas, oferece soluções de rede para os mercados doméstico e empresarial.

O único dispositivo NAS da empresa (atualmente) destinado ao mercado doméstico é o modelo DNS-320L, apresentado abaixo, na Tabela 12.

Tabela 12 – Dispositivo NAS D-Link

	DNS-320L
<b>Preço EUA (U\$)</b>	166,00
<b>Preço BR (R\$)</b>	690,00
<b>Processador</b>	Armada 370 - 1GHz
<b>RAM (GB)</b>	0,256
<b>Nº de baias</b>	2
<b>Nº conectores Ethernet</b>	1
<b>Nº USB 2.0</b>	1
<b>Nº USB 3.0</b>	0

## A.4 Dispositivos NAS - ASUSTOR

Fundada em 2011, a ASUSTOR é uma empresa fabricante de dispositivos NAS e provedora de serviços de vídeo vigilância.

Para esse estudo considerei apenas o seu (até então) modelo mais barato, o AS1002T.

Tabela 13 – Dispositivo NAS ASUSTOR

	AS1002T
<b>Preço EUA (U\$)</b>	191,00
<b>Preço BR (R\$)</b>	1 016,00
<b>Processador</b>	Armada 385 - Dual Core @ 1GHz
<b>RAM (GB)</b>	0,512
<b>Nº de baias</b>	2
<b>Nº conectores Ethernet</b>	1
<b>Nº USB 2.0</b>	0
<b>Nº USB 3.0</b>	2

## A.5 Dispositivos NAS - Seagate

A Seagate (segundo o site da empresa) é líder global no fornecimento de soluções de armazenamento de dados. Assim como a Western Digital, a Seagate fabrica dispositivos de armazenamento e seus derivados (HDs, HDs externos, dispositivos NAS, etc).

No quadro abaixo apresento o dispositivo NAS da empresa voltado para o usuário doméstico, o Seagate Personal Cloud.

Tabela 14 – Dispositivo NAS Seagate

	Personal Cloud
<b>Preço EUA (U\$)</b>	140,00 (3TB)
<b>Preço BR (R\$)</b>	1294,00 (3TB)
<b>Processador</b>	Armada 370 - 1,2GHz
<b>RAM (GB)</b>	0,512
<b>Nº de baias</b>	1
<b>Nº conectores Ethernet</b>	1
<b>Nº USB 2.0</b>	1
<b>Nº USB 3.0</b>	0

## A.6 Dispositivos NAS - QNAP

QNAP é uma empresa que oferece soluções de armazenamento de arquivos privado (NAS) e gravação de vídeo.

Os dispositivos mais próximos de uso doméstico oferecidos pela QNAP estão listados na tabela abaixo.

Tabela 15 – Comparação dispositivos NAS QNAP

	TS-128	TS-131P	TS-251
<b>Preço EUA (U\$)</b>	138,00	169,00	319,00
<b>Preço BR (R\$)</b>	Não vendido no Brasil	1 363,00	2 353,00
<b>Processador</b>	ARM v7 - Dual Core @ 1.1GHz	Cortex-A15 – Dual Core @ 1.4GHz	Intel Celeron – Dual Core @ 2.41GHz
<b>RAM (GB)</b>	1	1	1
<b>Nº de baias</b>	1	1	2
<b>Nº conectores Ethernet</b>	1	2	2
<b>Nº USB 2.0</b>	1	2	2
<b>Nº USB 3.0</b>	1	2	2

## A.7 Conclusões do estudo

Com o estudo dos produtos apresentados nesse capítulo, pode-se extrair algumas conclusões interessantes com relação ao hardware e ao preço de mercado deles.

Com relação ao hardware, observa-se que está presente em quase totalidade dos dispositivos um *System on Chip* (SoC) Armada, desenvolvido pela empresa Marvell. Este SoC apresenta um microprocessador baseado na arquitetura ARMv7 com 2 núcleos (Dual Core) e opera em uma frequência em torno de 1GHz.

Ainda com relação ao hardware, vemos que os dispositivos apresentam uma quantidade de memória RAM que varia entre 256 Megabytes (MB) e 1 Gigabyte (GB), a maioria apresentando o valor de 1 GB.

Quanto ao preço de mercado, observa-se que, para as unidades comercializadas no Brasil, o preço varia de R\$690,00 a R\$2.353,00 sem HD e R\$990,00 a R\$2.653,00 com HD (considerando a compra de um HD 3.5"de 2TB por R\$300,00). Temos então uma média (aproximada) de R\$1.600,00 para a compra no Brasil de um dispositivo NAS com um HD de 2TB instalado.

Analisando os preços no exterior (Estados Unidos), vemos que ele varia de U\$ 140,00 a U\$319,00 sem HD e U\$ 150,00 a U\$369,00 com HD (considerando a compra de um HD 3.5"de 2TB por U\$50,00). Temos então uma média (aproximada) de U\$210,00 para a compra nos EUA de um dispositivo NAS com um HD de 2TB instalado. No momento em que esse estudo foi realizado o dólar turismo estava em R\$3,35, logo o valor de U\$210,00 equivale a R\$703,50.

Apesar disso não ter relação com o presente trabalho, é interessante perceber que, além de termos uma maior variedade de escolha, pagamos (em média) menos da metade do preço por um dispositivo NAS nos EUA do que no Brasil.