



UFAM - Engenharia Elétrica

**RELATÓRIO DE ESTÁGIO SUPERVISIONADO: BANCO DE TESTES PARA
SENSORES DE IMAGEM DE LONGO ALCANCE DINÂMICO**

Manaus
Março de 2014

UNIVERSIDADE FEDERAL DO AMAZONAS - UFAM

LUCIANO LOURENÇO FURTADO DA SILVA

RELATÓRIO DE ESTÁGIO SUPERVISIONADO

DEPARTAMENTO : **DEPARTAMENTO DE ELETRÔNICA E COMPUTAÇÃO - DTEC**
TÍTULO DE TRABALHO : **BANCO DE TESTE PARA SENSORES DE IMAGEM DE LONGO
ALCANCE DINÂMICO**
ALUNO : **LUCIANO LOURENÇO FURTADO DA SILVA**
INSTITUIÇÃO : **UNIVERSIDADE FEDERAL DO AMAZONAS**
PROFº ORIENTADOR : **CARLOS AUGUSTO DE MORAES CRUZ**

Manaus – AM 2013

Manaus
Março de 2014

SUMÁRIO

INTRODUÇÃO	6
1. O CHIP.....	7
2. MATERIAIS E EQUIPAMENTOS ELETRÔNICOS UTILIZADOS E SUAS FUNÇÕES	9
Socket.....	9
Potenciômetros	10
Osciloscópio	11
CI –CD4007UB	11
3. INTEGRAÇÃO DO CHIP À PLATAFORMA DE TESTES	14
4. VERIFICAÇÃO DAS FUNÇÕES DO BANCO DE TESTES.....	15
Sinais de Controle.....	15
Sinais de endereçamento	18
5. TESTES DO FUNCIONAMENTO DO SENSOR UTILIZANDO A PLATAFORMA DE TESTES DESENVOLVIDA	19
CONSIDERAÇÕES FINAIS	25
REFERÊNCIAS.....	26
ANEXOS	27

LISTA DE FIGURAS

Figura 1.1: O CHIP	7
Figura 1.2: Layout das Matrizes de pixels e de seus respectivos circuitos de controle e endereçamento	8
Figura 1.3: Matriz de pixels.....	8
Figura 2.1: Mapeamento do socket	9
Figura 2.2: Osciloscópio	11
Figura 2.3: Disposição dos transistores DO CD4007UBE	11
Figura 2.4: CD4007UBE	12
Figura 3.1: Diagrama esquemático da plataforma de testes	14
Figura 3.2: Chip integrado ao banco de testes desenvolvido	14
Figura 4.1: Sinais do pixel	15
Figura 4.2: Arquitetura digital.....	16
Figura 4.3: Sinais de entrada do sensor gerados com o ARDUINO e regiões de trabalho do pixel.....	17
Figura 4.5: Sinais digitais gerados pelo Arduino vistos no osciloscópio.	17
Figura 4.6: Endereçamento de linha	18
Figura 4.7:Endereçamento de coluna	18
Figura 5.1: leituras da coluna 0	19
Figura 5.2: leituras da coluna 1	20
Figura 5.3: leituras da coluna 2	20
Figura 5.4: leituras da coluna 3	21
Figura 5.6: leituras da coluna 5	22
Figura 5.7: leituras da coluna 6	22

Figura 5.8: leituras da coluna 7	23
Figura 5.10: Leituras da coluna 5	23
Figura 5.10: Leituras dos pixels da ultima coluna	24

INTRODUÇÃO

Neste trabalho apresentamos o desenvolvimento da plataforma de testes para um sensor de imagem implementado no processo CMOS 0,35 μm . O sensor consiste de uma matriz 8x8 de circuitos APS trabalhando em modo tensão e os circuitos de controle e endereçamento necessários ao seu funcionamento. Serão apresentados os resultados experimentais dos testes realizados com o sensor, objetivando a comprovação experimental de uma teoria que consiste na implementação de uma técnica para redução de FPN (ruído de padrão fixo) quando o sensor esta no modo de funcionamento linear e logarítmico. Portanto, este trabalho é uma contribuição ao estudo de redução de ruído em sensores de imagem do tipo APS.

1. O CHIP

Na figura a seguir, podemos observar varios circuitos que foram integrados juntamente com o sensor, que é evidenciada pela quadrado vermelho. Esta região do chip é exposta através de uma lente para que haja incidência de luz sobre o sensor. Nela já estão incluídos a matriz de pixels e os circuitos de controle e endereçamento do sensor. As posições numeradas representam os pads de saída do sensor.

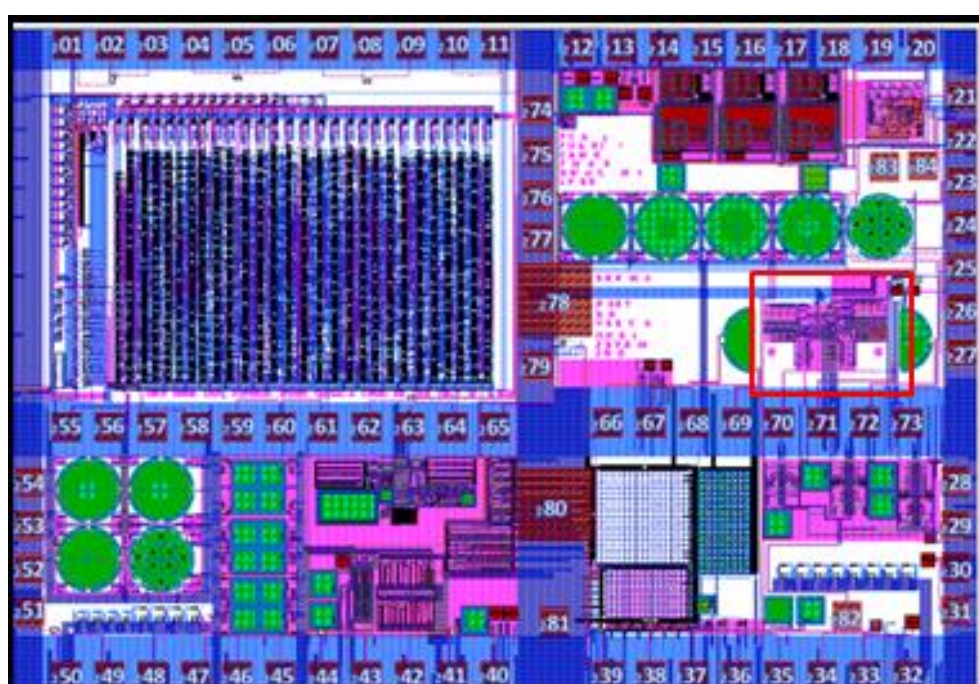


Figura 1.1: O CHIP

A figura 1.2 nos mostra uma ampliação da região de interesse. A parte da figura coberta por uma camada de cor rosa clara, indica a região onde localizam-se os circuitos de controle que não podem receber luz. Já a parte exposta (sem a camada de metal) representam a matrizes de pixels (note que são duas). A matriz localizada a esquerda (do leitor) é uma matriz de teste rápido. Neste trabalho buscamos os resultados dos testes realizados com a matriz de testes completo (matriz de sensores à direita do leitor).

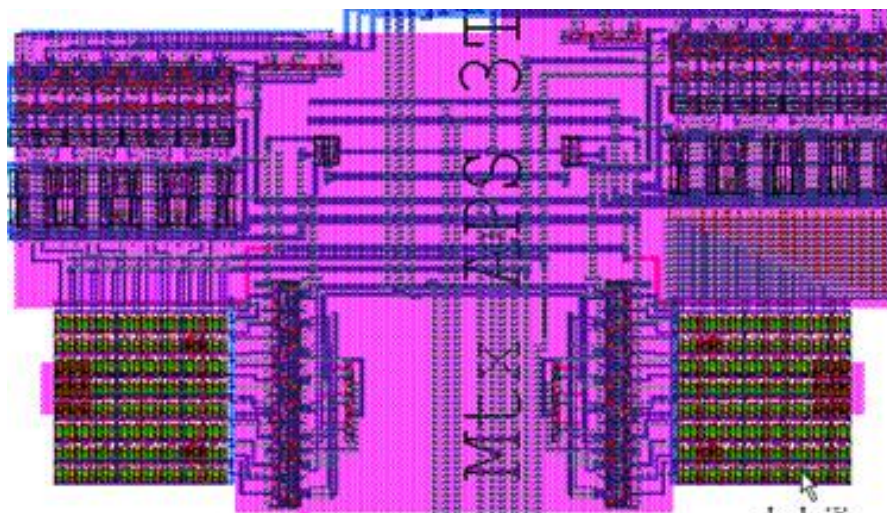


Figura 1.2: Layout das Matrizes de pixels e de seus respectivos circuitos de controle e endereçamento

Na figura 1.3 podemos observa a matriz de pixels que será testada. Repare que a uma camada de metal sobre alguns pixels. Isto foi feito propositalmente para induzir uma diferença na informação lida entre os pixels cobertos e os outros pixels. Repare ainda que alguns pixels estão parcialmente cobertos, enquanto outros estão totalmente cobertos.

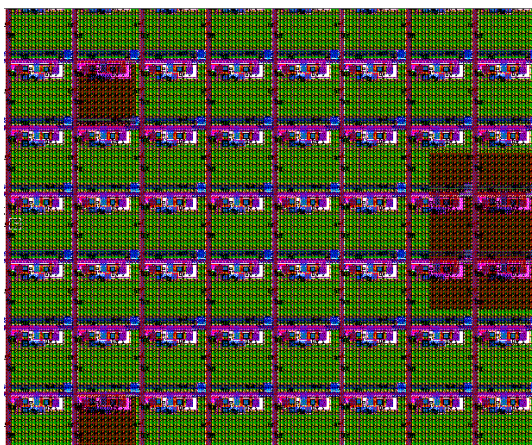


Figura 1.3: Matriz de pixels

2. MATERIAIS E EQUIPAMENTOS ELETRÔNICOS UTILIZADOS E SUAS FUNÇÕES

Os materiais e a descrição dos equipamentos utilizados para montagem da plataforma de testes são listados a seguir:

- 1 x socket;
- 3 x CI –CD4007UB (inversores);
- 6 x resistores de 330Ω ;
- 1 x microcontrolador (ARDUINO UNO);
- Potenciômetros;
- 1 x Osciloscópio;
- 2 x Protoboards;
- Jumpers;
- 6 x LED's

Socket

Utilizou-se o socket adequado conforme a estrutura dos pads de saída do chip. O mapeamento dos terminais do socket com o chip já acoplado é mostrado na figura a seguir:

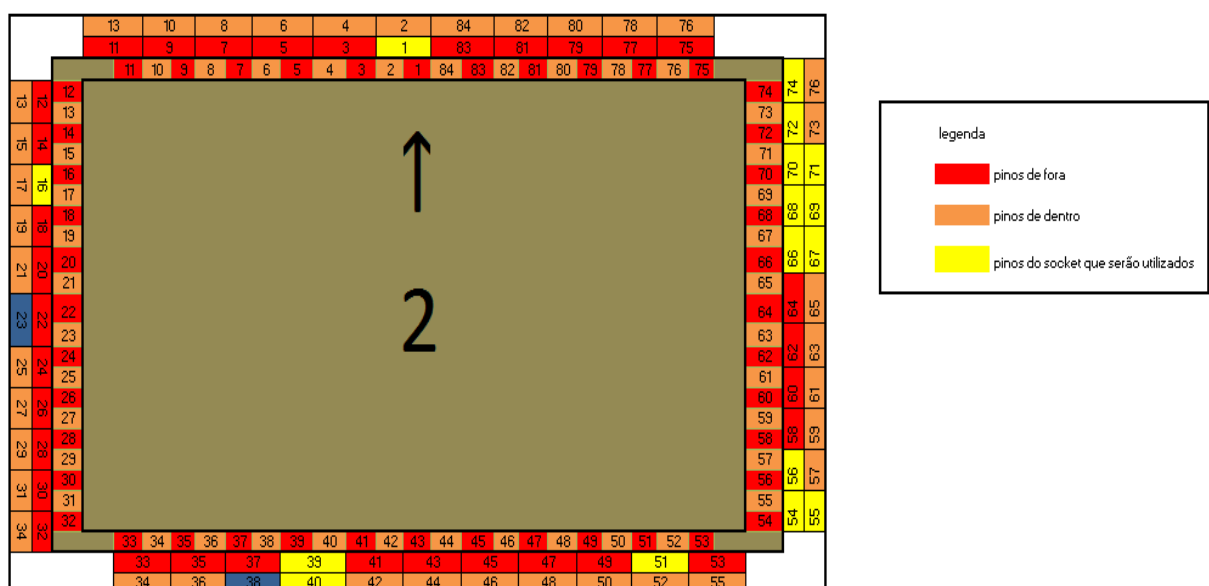


Figura 2.1: Mapeamento do socket

Observe na legenda que os pinos que serão utilizados tem a cor amarela e estão devidamente numerados. A numeração e a função de cada pino são elencadas a seguir:

- Pino 1: V_{DD} (alimentação em 3.3 Volts);
- Pino 16: V_{Load} (tensão de 0,8 Volts para os transistores de carga);
- Pino 39: GND (terra analógico);
- Pino 40: OUTG (saída global do sensor);
- Pino 51: RST D1 (sinal de reset digital);
- Pino 54: ROWSELDIG S30 (sinal de seleção de linha);
- Pino 55: RDR S29 (sinal de controle RDR);
- Pino 66: MOD 1 (Ativação da matriz)
- Pino 67: AC2 (endereçamento de coluna)
- Pino 68: AC1 (endereçamento de coluna)
- Pino 69: AC0 (endereçamento de coluna)
- Pino 70: A2 (endereçamento de linha)
- Pino 71: A1 (endereçamento de linha)
- Pino 72: A0 (endereçamento de linha)
- Pino 74: V_{DD} (outro ponto de alimentação em 3.3 Volts);
- Pino 38: E_RDR;
- Pino 23: E_RST.

Através do pino 23, podemos modular o sinal de Reset. Observe que é através da modulação do nível baixo de RST, tensão no pino 23, que podemos modificar o modo de operação do sensor. A modulação do nível baixo de RDR é feita através da alteração do valor de tensão no pino 38.

Potenciômetros

Utilizaremos três potenciômetros com o objetivo de estabelecer níveis de tensão predefinidos no projeto do sensor. Para isso, os potenciômetros serão ligados na configuração de divisor de tensão em paralelo com a alimentação de 3.3 Volts.

Osciloscópio

Utilizamos o osciloscópio para verificação das formas de onda de tensão dos sinais de entrada do sensor e análise da resposta do sensor quando submetidos a vários níveis de intensidade luminosa.



Figura 2.2: Osciloscópio

CI –CD4007UB

O CI – CD4007UB consiste de três transistores PMOS e três transistores NMOS dispostos na configuração mostrada na figura a seguir.

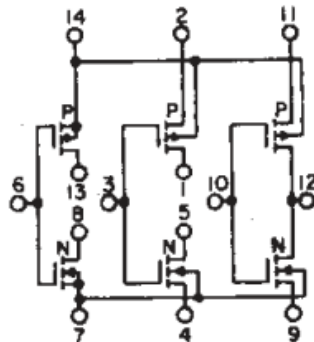


Figura 2.3: Disposição dos transistores DO CD4007UBE

O microcontrolador ARDUINO UNO fornece os níveis de tensão dos sinais de entrada variando de 0 a 5 Volts. Os inversores tem o papel de fazer com que essa variação ocorra de 0 a 3.3 Volts, visto que o circuito foi produzido para operar nessa faixa. Observe que a utilização deste CI implicará na operação dos sinais de entrada gerados pelo arduino, pois serão invertidos na saída do CD4007UB. Por esse motivo a programação que gera os sinais de controle endereçamento foi realizada de modo invertido.



Figura 2.4: CD4007UBE

ARDUINO UNO

O Arduino é uma plataforma de prototipagem eletrônica de hardware livre, projetada com um microcontrolador Atmel AVR de placa única, com suporte de entrada/saída embutido, uma linguagem de programação padrão, a qual tem origem em Wiring, e é essencialmente C/C++.

A principal finalidade do Arduino num sistema é facilitar a prototipagem, implementação ou emulação do controle de sistemas interativos, a nível doméstico, comercial ou móvel, da mesma forma que o CLP controla sistemas de funcionamento industriais. Com ele é possível enviar ou receber informações de basicamente qualquer sistema eletrônico, como identificar a aproximação de uma pessoa e variar a intensidade da luz do ambiente conforme a sua chegada. Ou abrir as janelas de um escritório de acordo com a intensidade da luz do sol e temperatura ambiente.



Figura 2.5: Arduino UNO

Neste trabalho utilizaremos o Arduino para gerar os sinais de controle e endereçamento necessários ao funcionamento do sensor. No capítulo 4 será realizada a verificação das formas de ondados sinais de controle e endereçamento utilizando um osciloscópio.

3. INTEGRAÇÃO DO CHIP À PLATAFORMA DE TESTES

Nas figuras abaixo podemos apreciar o diagrama esquemático da plataforma de testes com o chip acoplado a ela e uma foto do setup montado.

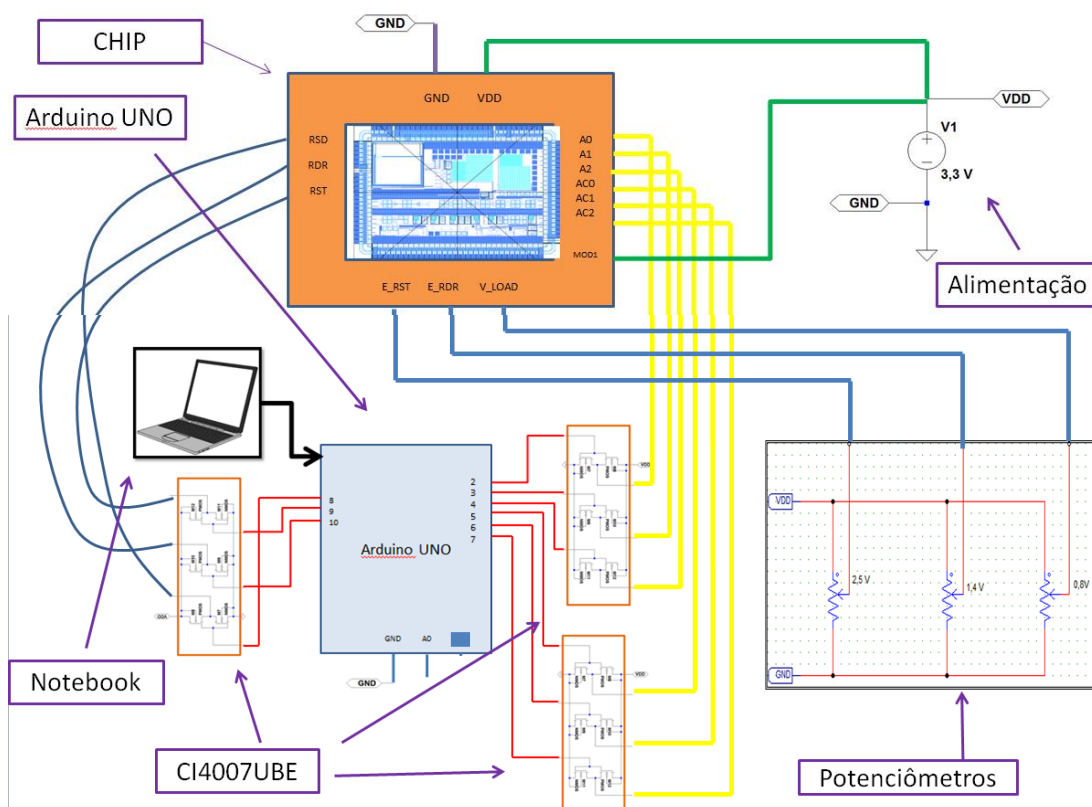


Figura 3.1: Diagrama esquemático da plataforma de testes

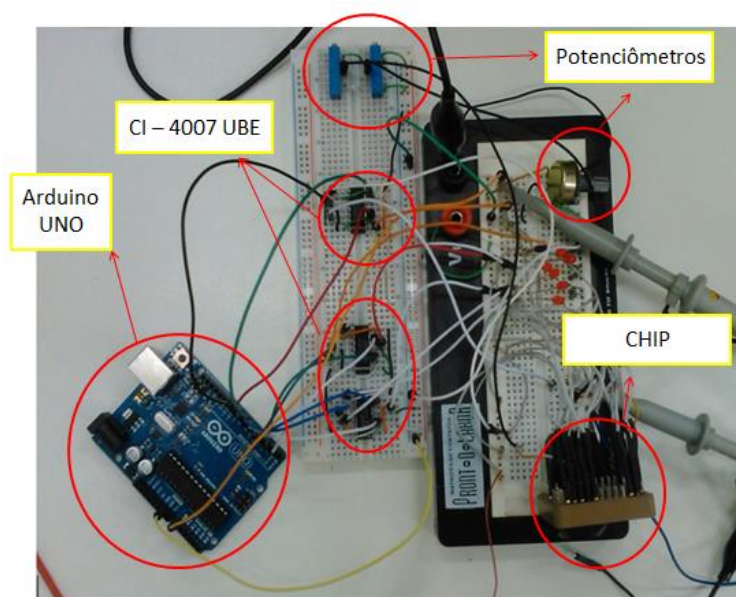


Figura 3.2: Chip integrado ao banco de testes desenvolvido

4. VERIFICAÇÃO DAS FUNÇÕES DO BANCO DE TESTES

Como já foi mencionado anteriormente, o banco de testes tem a função de fornecer todo o suporte necessário para que seja possível a realização dos testes com o sensor. As funções que devem ser desempenhadas pelo banco de testes são listadas a seguir:

- ✓ Gerar os sinais de controle;
- ✓ Gerar os sinais de endereçamento;
- ✓ Fornecer os níveis de tensão necessários;

Sinais de Controle

As formas de onda que devem chegar a cada pixel são mostradas na figura 4.1.

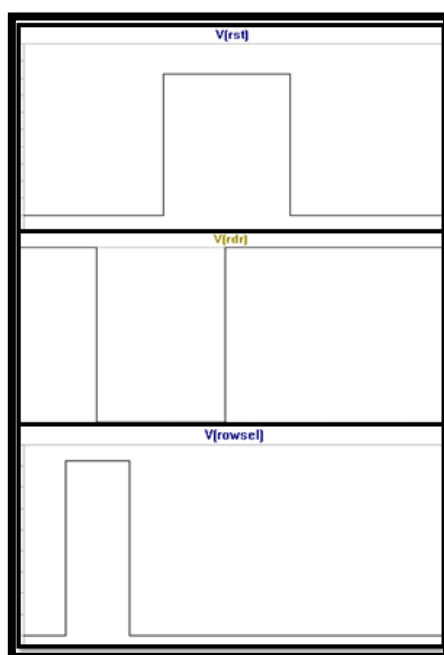


Figura 4.1: Sinais do pixel

Para gerar estes sinais utilizaremos o microcontrolador ARDUINO UNO. Para isso, foi desenvolvido um código na linguagem de programação do ARDUINO UNO, o C/C++. Para o desenvolvimento do código devemos considerar a arquitetura interna do sensor e os circuitos externos necessários.

A figura 4.2 nos mostra de uma forma bem simples toda a arquitetura que estará após os sinais digitais de entrada, já incluindo os inversores introduzidos externamente. Quando o sinal no terminal DEMUX estiver no nível lógico baixo, não importa os níveis lógicos dos outros terminais, sempre teremos para os nós de saída RDRX, RSTX e ROWSELX, respectivamente níveis lógicos “0”, “1” e “0”, que indica a região de integração do pixel, ou seja, nenhuma operação de chaveamento ocorre. Quando o sinal no terminal DEMUX estiver no nível lógico alto, os circuitos de controle e endereçamento estarão habilitados realizar as devidas operações de chaveamento necessárias ao funcionamento do sensor, controladas pela temporização programada.

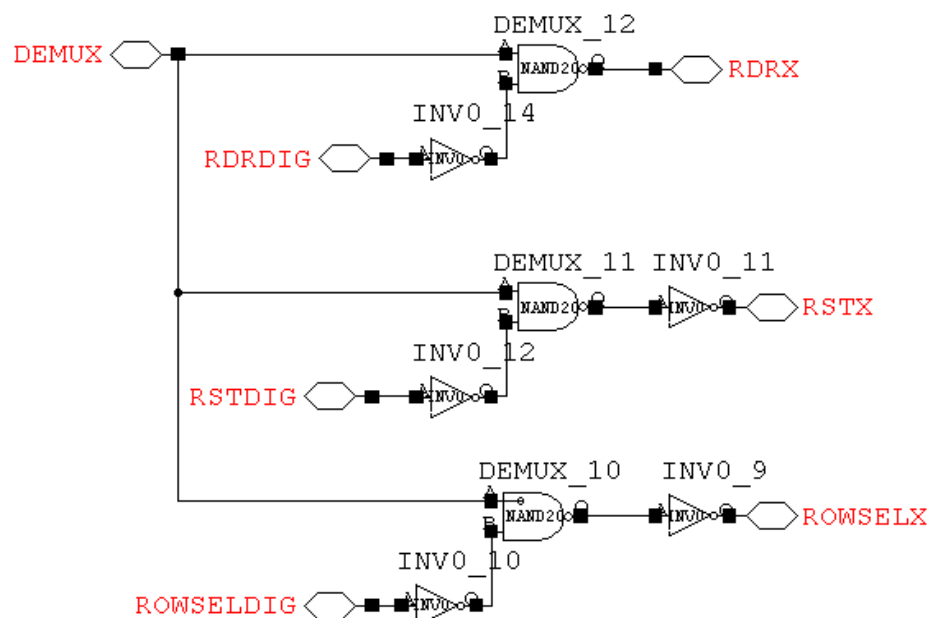


Figura 4.2: Arquitetura digital

O código desenvolvido já considera a arquitetura vista na figura 4.2 para que os pixels recebam os sinais vistos na figura 4.1. Foram desenvolvidas duas versões do código que podem ser encontrados nos anexos deste trabalho. Com estes códigos realizamos duas leituras em intervalos de tempo diferentes para cada um dos 64 pixels do sensor e imprimimos na tela. Os sinais de entrada do sensor gerados pelo arduino com a programação desenvolvida, são mostrados a seguir com alguns detalhes das regiões de funcionamento do sensor.

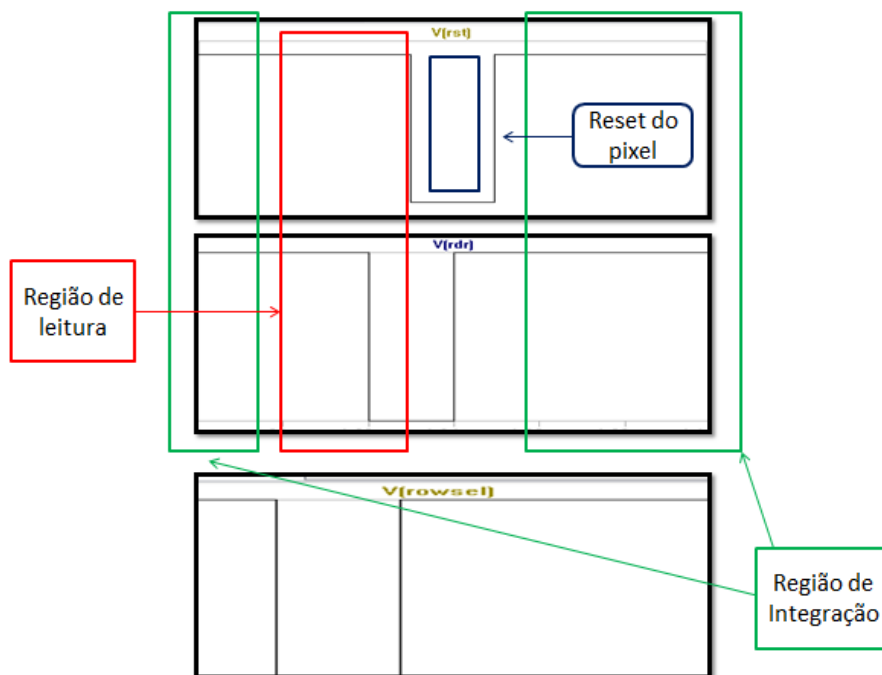


Figura 4.3: Sinais de entrada do sensor gerados com o ARDUINO e regiões de trabalho do pixel

Na figura 4.3 vimos as formas de onda de tensão dos sinais que devem ser gerados pelo ARDUINO com o código desenvolvido. Foram realizados os testes para a obtenção das formas de onda com osciloscópio, que podem ser vistas na figura 4.5:

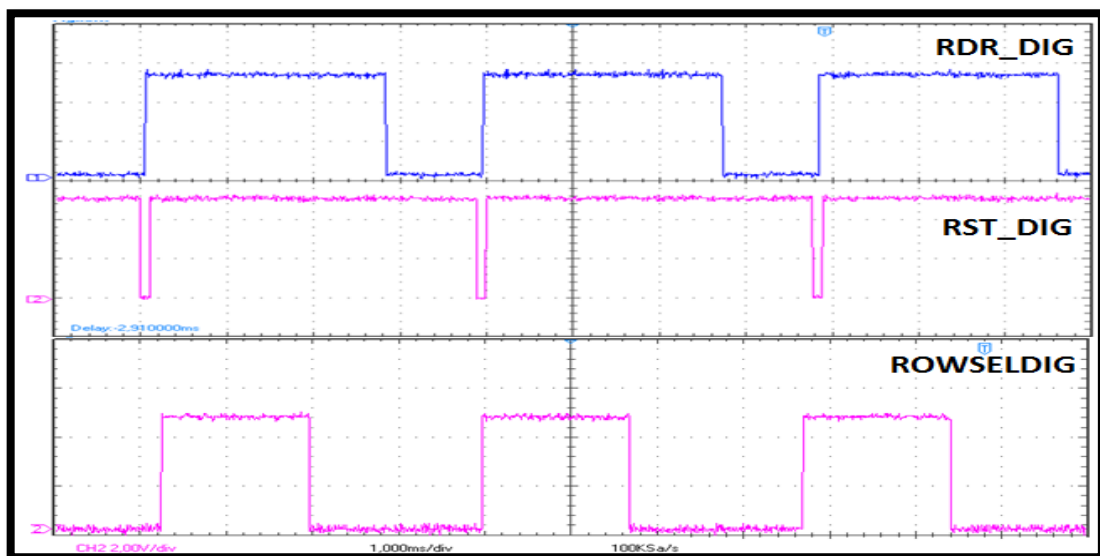


Figura 4.5: Sinais digitais gerados pelo Arduino vistos no osciloscópio

Sinais de endereçamento

O endereçamento gerado pelo arduino tem o objetivo de selecionar sequencialmente cada pixel para que seja habilitada a sua leitura. Foram verificados também as formas de onda de endereço de linha e coluna via osciloscópio, que podem ser observados nas figuras a seguir:

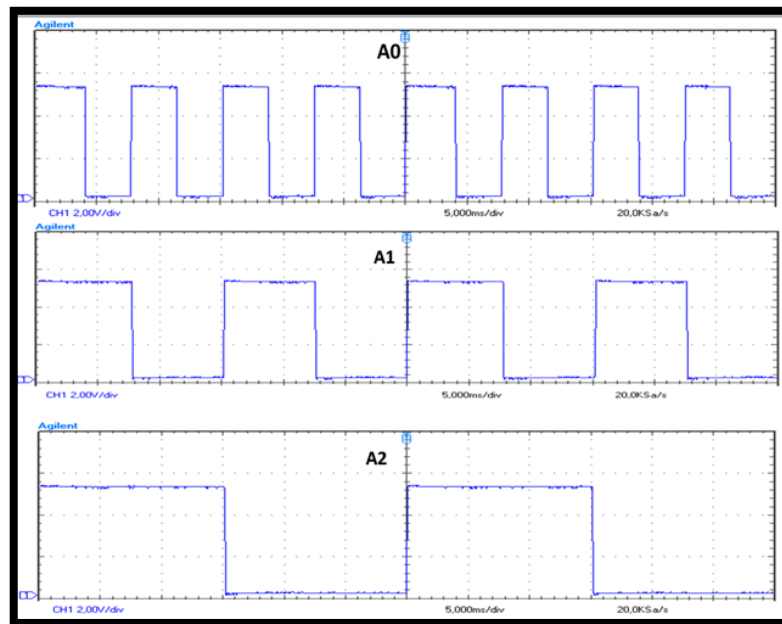


Figura 4.6: Endereçamento de linha

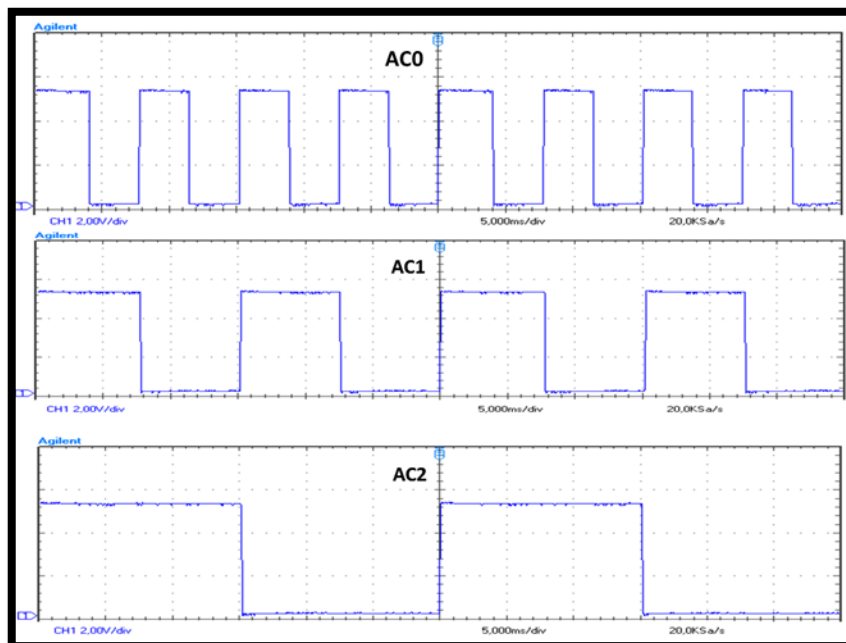


Figura 4.7: Endereçamento de coluna

5. TESTES DO FUNCIONAMENTO DO SENSOR UTILIZANDO A PLATAFORMA DE TESTES DESENVOLVIDA

Com o código desenvolvido, selecionamos uma coluna e realizamos as leituras de todos os pixels da mesma; em seguida passamos a ler as próximas colunas, até que toda a matriz seja varrida.

Neste primeiro teste a varredura da matriz foi realizada da seguinte forma: selecionamos uma coluna e realizamos as leituras de todos os pixels da mesma; em seguida passamos a ler as próximas colunas até que toda a matriz seja varrida.

Os resultados obtidos das leituras de cada coluna podem ser apreciados nas figuras 5.1 a 5.8. As colunas foram nomeadas de 0 a 7.

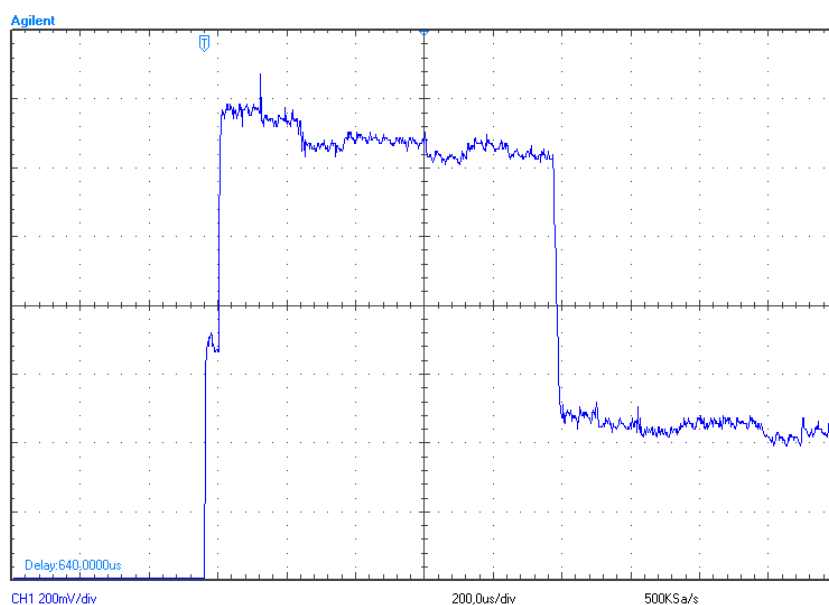


Figura 5.1: leituras da coluna 0

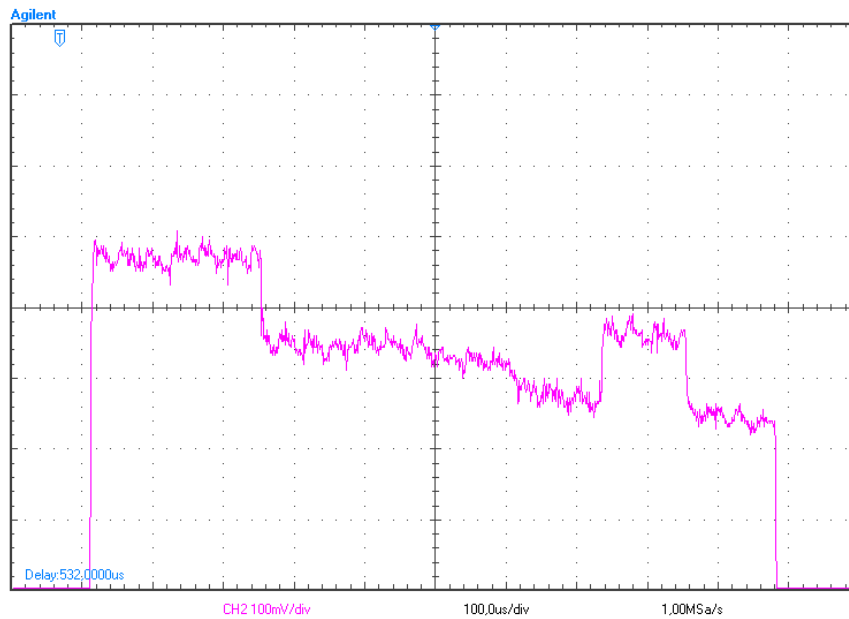


Figura 5.2: leituras da coluna 1

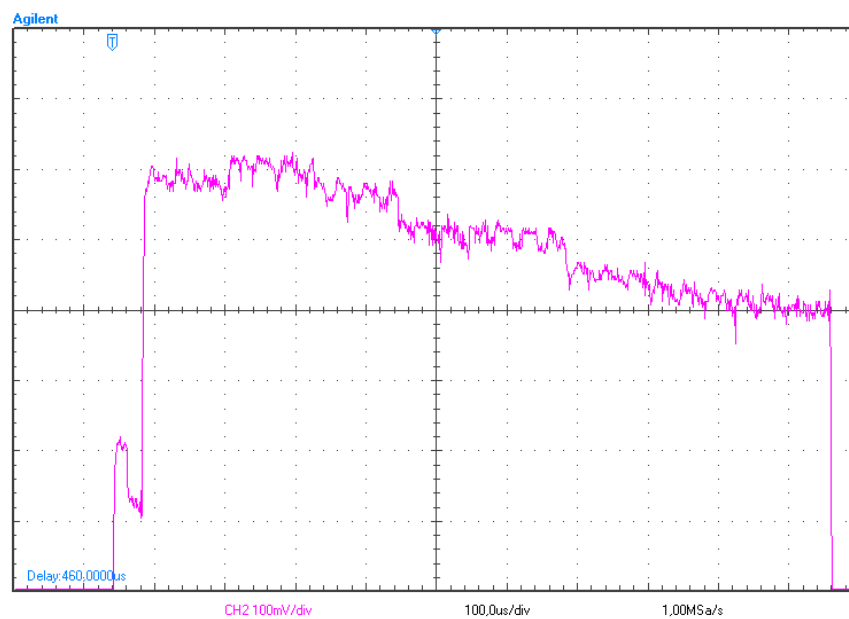


Figura 5.3: leituras da coluna 2

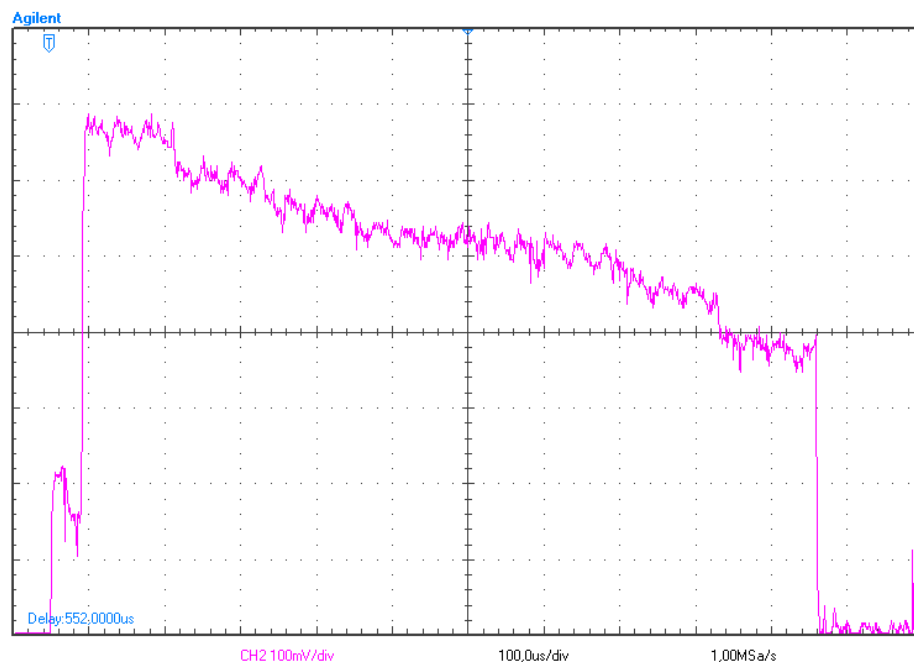


Figura 5.4: leituras da coluna 3

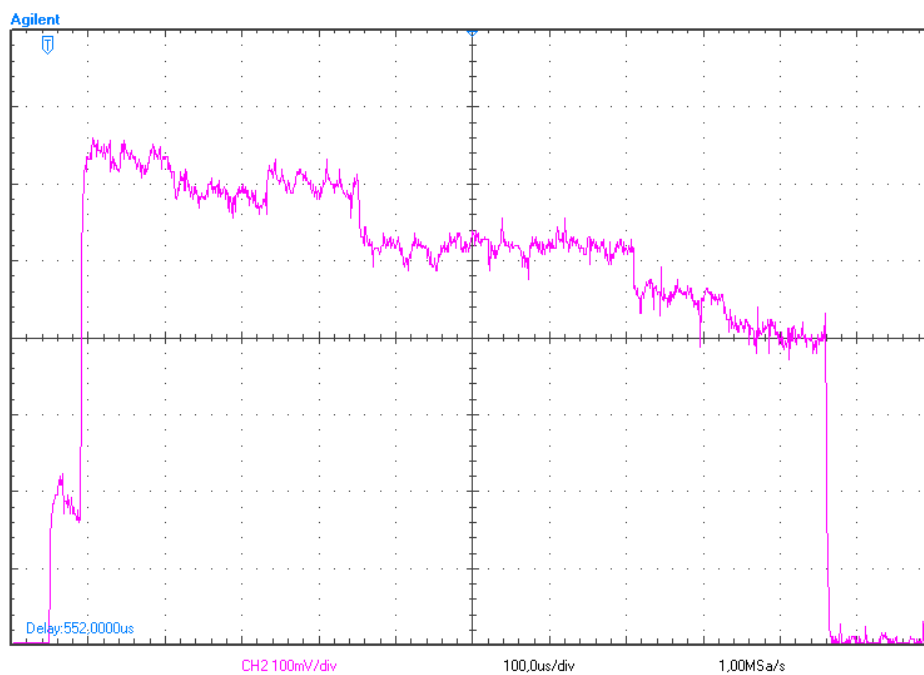


Figura 5.5: leituras da coluna 4

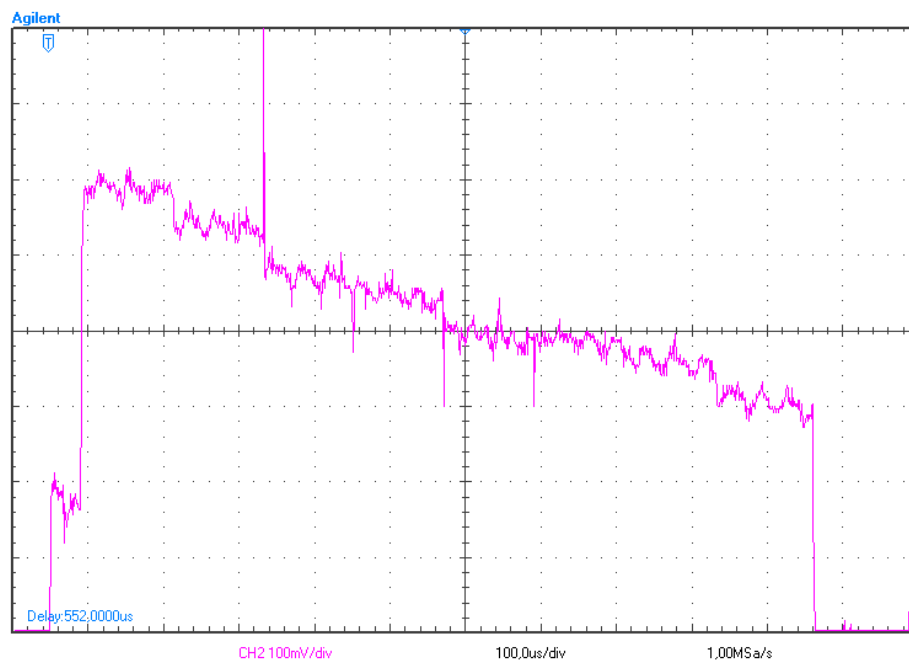


Figura 5.6: leituras da coluna 5

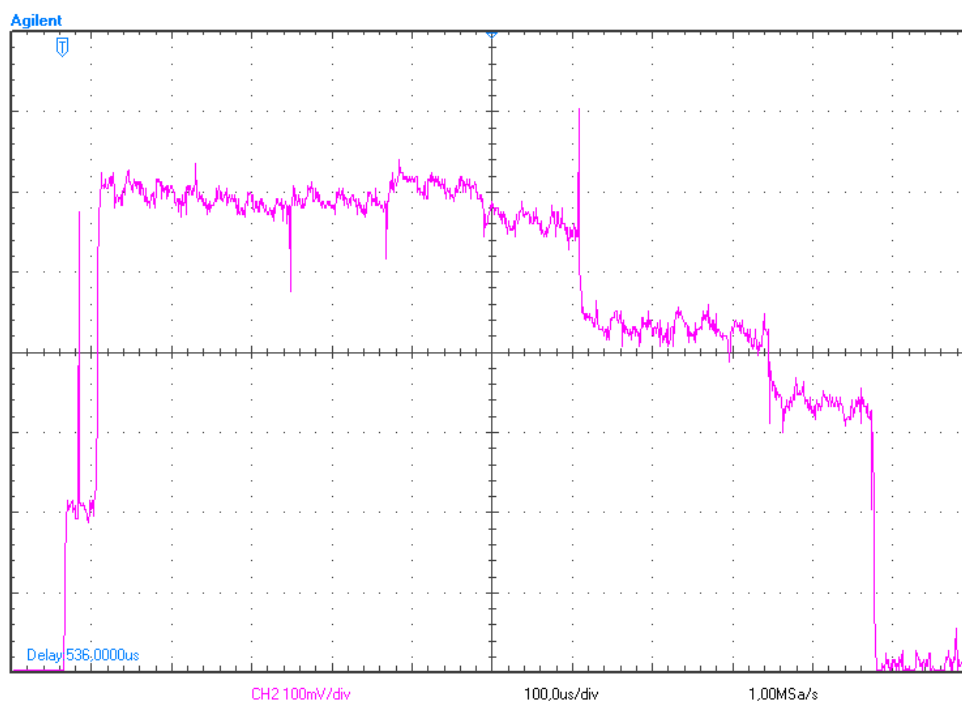


Figura 5.7: leituras da coluna 6

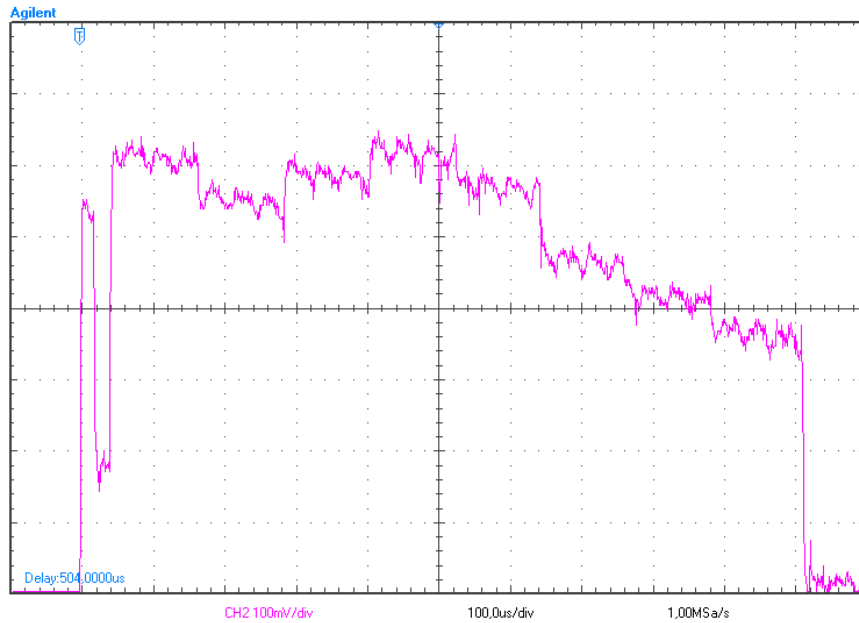


Figura 5.8: leituras da coluna 7

Podemos tirar as seguintes conclusões observando estas leituras:

- As leituras indicam o comportamento de uma queda linear. Isto nos informa que o sensor esta recebendo uma baixa intensidade luminosa, como podemos observar, por exemplo, na figura 5.9 que indica as leituras dos pixels da sexta coluna (coluna nomeda “coluna 5”).

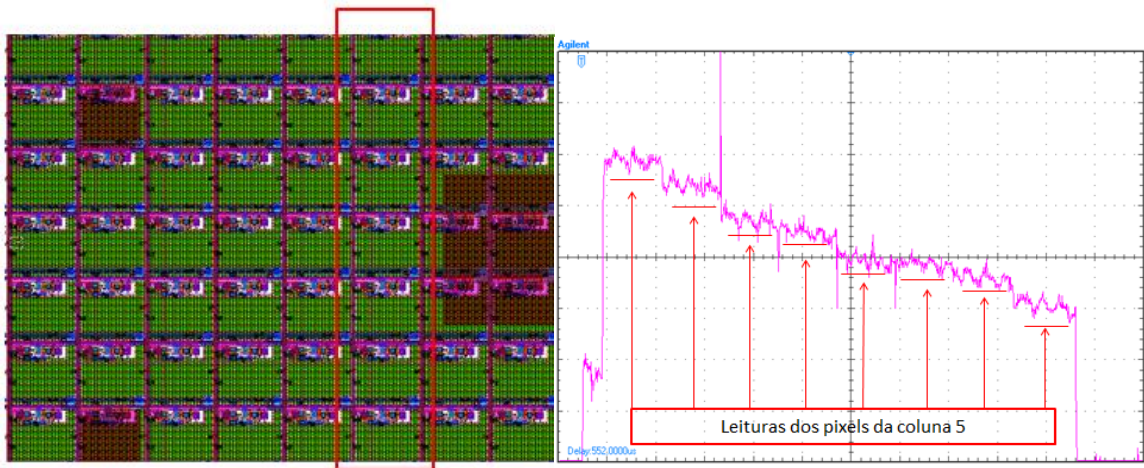


Figura 5.10: Leituras da coluna 5

- Podemos observar que para as leituras das figuras 5.2, 5.7 e 5.8, as leituras possuem um salto em alguns pontos das leituras. Observe a figura 5.10, onde a última coluna é evidenciada. Observe que essa

coluna possui o terceiro e o quinto pixels parcialmente cobertos e o quarto pixel totalmente coberto. Perceba que nas leituras dos pixels parcialmente cobertos ocorreram saltos na curva que deveriam ter um comportamento linear descendente. Isto se deve ao fato de que a luz incidente sobre esses pixels não é suficiente para produzir uma corrente fotogerada que reduza o nível de tensão do elemento fotosensor presente nestes pixels e, por isso, são lidos maiores níveis de tensão. Isto também pode ser observado na figura 5.10.

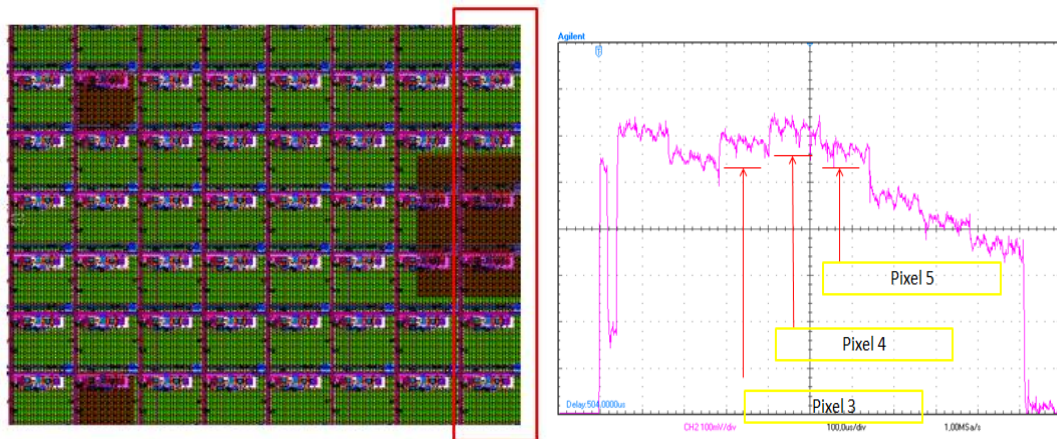


Figura 5.10: Leituras dos pixels da ultima coluna

CONSIDERAÇÕES FINAIS

No presente trabalho vimos como foi realizada etapa final de um fluxo de projeto para desenvolvimento de um circuito integrado. Nesta etapa foi realizado o desenvolvimento do setup para a realização dos testes das estruturas internas do chip. Para isso, foi desenvolvido código para geração dos sinais de excitação do sensor e de endereçamento da matriz. Em seguida fizemos o mapeamento dos terminais do socket ao qual seria acoplado o chip para que fossem realizadas as conexões corretas. Por fim utilizamos vários equipamentos eletrônicos, como microprocessadores, fontes de alimentação, osciloscópio, entre outros, para finalizar o desenvolvimento desta plataforma de testes.

Finalizado o desenvolvimento do setup, iniciamos a realização dos testes. Os resultados das leituras geraram matrizes de valores analógicos, chamados de mascaras, que foram convertidos em uma escala de cinza, gerando assim uma imagem em escala de cinza 8x8. Podemos observar que as imagens geradas através desses dados não podem ser bem definidas devido à baixa resolução resultante da pequena quantidade de pixels. Porém, estas máscaras geraram dados mais que suficientes para que análise e implementação da técnica de redução do ruído de padrão fixo. Os dados obtidos de redução de ruído são visíveis. A obtenção destes dados foi o principal objetivo do desenvolvimento do setup de testes.

REFERÊNCIAS

1. Carlos A. de Moraes Cruz, Davies W. de Lima Monteiro, Gilles and Alexandre K. P. Souza, "Simple Technique to Reduce FPN in a linear-logarithm APS", "In 2013 Internacional Image Sensor Workshop (IISW)", Snowbird, Utah, USA, June, 2013.
2. SOUZA, Alexandre Kennedy Pinto, Relatório de Estágio Supervisionado, **"Banco de Testes em uma Matriz de sensores de Imagem"**.
3. <http://www.ti.com/lit/ds/symlink/cd4007ub.pdf>, **DATASHEET do CI-CD4007UB**, acessado em 16 de março de 2014.
4. SEDRA, Adel Smith, SMITH, K. C., **Microeletrônica**. 5ª edição, São Paulo: earson Prentice Hall, 2007.
5. BROWN, Stephen, VRANESIC, Zvonko, **Fundamentals of DIGITAL LOGIC with VHDL design**. 2ª edição, Mc Graw Hill Company, 2005.
6. S.N.M, Mestanza, I.F., Silva, J.E.C., Queiroz, J.W Swart, **Sensor de Pixel Ativo no processo CMOS 0,35 μm** , CCS/FEEC-UNICAMP, Campinas-São Paulo, Brasil.

ANEXOS

CÓDIGO “VERSÃO 1”

//LEITURA DA MATRIZ DE PIXELS

int sensorPin = A0;

float sample_s1 [2][64];

int i, j, k;

int junk;

void setup()

{

DDRD=B11111100;

DDRB=B00000111;

Serial.begin(9600);

}

void loop()

{

//Espera para pressionar uma tecla para iniciar as k leituras da matriz

while (*Serial.available*() == 0)

{

delay(100);

}

junk = Serial.read();

//Fim da espera para pressionar uma tecla para iniciar as k leituras da matriz

PORTD = B00000000;

PORTB = B00000111;

//Segundo loop de varredura das colunas. Realiza as duas leituras para cada pixel de uma coluna

```
for (j=0;j<8;j++)
```

```
{
```

```
    delayMicroseconds(1654);
```

```
    PORTB = B00000011; //RST, RDR e RSD
```

```
    delayMicroseconds(20);
```

```
    //Custo de tempo: 20-24 microssegundos
```

```
    PORTD = ~PORTD;
```

```
    PORTD = (j) << 2;
```

```
    delayMicroseconds(20);
```

```
    //Primeira leitura. Custo de tempo: 968 microssegundos
```

```
    for(i=0;i<8;i++)
```

```
    {
```

```
        PORTD = (j+(8*i)) << 2;
```

```
        PORTD = ~PORTD;
```

```
        sample_s1 [0][j+(8*i)] = analogRead(sensorPin); // 120us
```

```
    }
```

```
    delayMicroseconds(10);
```

```
    //Custo de tempo: 20-24 microssegundos
```

```
    PORTB = B00000001; //RST baixo, RDR baixo, RSD alto
```

```
    delayMicroseconds(10);
```

```
    //Segunda leitura. Custo de tempo: 960-972 microssegundos.
```

```

for(i=0;i<8;i++)
{
    PORTD = (j+(8*i)) << 2;
    PORTD = ~PORTD;
    sample_s1 [1][j+(8*i)] = analogRead(sensorPin);
}
delayMicroseconds(20);

```

//Custo de tempo: 12-16 microssegundos

```

PORTB = B00000101; //RST baixo, RDR baixo, ROWSEL DIG
baixo
delayMicroseconds(10);
PORTB = B00000100; //RST alto, RDR baixo, ROWSEL DIG baixo
delayMicroseconds(20);
PORTB = B00000110; //RST alto, RDR alto, ROWSEL DIG baixo
delayMicroseconds(20);
PORTB = B00000111; //RST baixo, RDR alto, ROWSEL DIG
baixo
}

```

```

Serial.println("Amostragem s1");
for (j=0;j<8;j++) {
    for(i=0;i<8;i++) {
        if(i<8){
            Serial.print((sample_s1[0][(i+(8*j))])*0.00489,3);
            Serial.print(" ; ");
        }
    }
}

```

```

}
Serial.println();
}
Serial.println();
Serial.println("Amostragem s2");

for (i=0;i<8;i++) {
    for(j=0;j<8;j++) {
        if(j<8){
            Serial.print((sample_s1[1][(j+(8*i))])*0.00489,3);
            Serial.print(" ; ");
        }

    }
    Serial.println();

}
Serial.println();
Serial.println();
}

```

Fotos do Banco de testes desenvolvido

