INDIAN INSTITUTE OF ENGINEERING SCIENCE AND TECHNOLOGY, SHIBPUR

Design of a Smart Graphics System

by

Anupam Sanidhya

Examination Roll Number : 510817049 of 2017-21
under the supervision of
Dr. Sukanta Das

A report submitted in fulfilment of the requirements for the
Major project of Bachelor of Technology in Information Technology (8th Semester)

Department of Information Technology
Indian Institute of Engineering Science and Technology, Shibpur
P.O. Botanic Garden, Howrah 711103, WB, India
June 2021

CONTENTS

Certificate of Approval 3
Forward 4
Acknowledgement 5
Design of A Smart Graphics System 6
Chapter 1 : Proposed Methodology 7
    1.1 Circuit of a Single Pixel 8
Chapter 2 : Working 9
    2.1 : Working of a Single Pixel 9–11
    2.2 : Counter Decrement Circuit 12
    2.3 : Transfer of Control from one pixel to next 13
Chapter 3 : Estimations 14
    3.1 : Size Estimate for a 256x256 screen 14
    3.2 : Number of Transistors for a 256x256 screen 15
    3.3 : Theoretical minimum pixel size 15
Challenges and Limitations 16
Bibliography 17

Certificate of Approval

This is to certify that the thesis synopsis entitled **"Design of a Smart Graphics System"** is a record of bona fide work carried out by Mr. Anupam Sanidhya under my supervision and guidance.

The report has fulfilled the requirements for the completion of major project of degree of Bachelor of Technology in Information Technology from Indian Institute of Engineering Science and Technology, Shibpur, India.

He has duly completed the required course/research work with sincerity and the work has reached the standard necessary for submission.

Signed:

[Dr. Sukanta Das]
Associate Professor and Head
Department of Information Technology
Indian Institute of Engineering Science and Technology, Shibpur
Howrah - 711103, West Bengal, India

DEPARTMENT OF INFORMATION TECHNOLOGY
INDIAN INSTITUTE OF ENGINEERING SCIENCE AND TECHNOLOGY, SHIBPUR
P.O. BOTANIC GARDEN, HOWRAH 711103

FORWARD

I would like to forward the report entitled **Design of a Smart Graphics System** towards the examination committee as a record of bona fide work carried out by Anupam Sanidhya under my supervision and guidance.
In my opinion, the work for the report is satisfactory and it has reached the standard necessary for the submission in the fifth semester of *Bachelor of Technology in Information Technology* of Indian Institute of Engineering Science and Technology, Shibpur.

[Dr. Sukanta Das]
Associate Professor and Head
Department of Information Technology
Indian Institute of Engineering Science and Technology, Shibpur

Countersigned by:

[Dr. Sudip Kumar Roy]
Professor and Dean (Academic Affairs) [Acting]
Indian Institute of Engineering Science and Technology, Shibpur

ACKNOWLEDGEMENT

I would hereby like to avail this opportunity to extend my heartfelt gratitude to my project guide Dr. Sukanta Das, Associate Professor and Head, Department of Information Technology, Indian Institute of Engineering Science and Technology, Shibpur, for his incessant encouragement, sincere supervision, valuable advice and perfect guidance throughout the entire period without which it would not have been possible to complete the portion of the project scheduled for the semester and the term paper in time.

I would also like to thank all other faculty members and staff members for their support and help as and when required.

Finally, I cordially thank my team members and fellow classmates of Information Technology, Indian Institute of Engineering Science and Technology, my parents and all others who have been attached to this phase of the development of the project directly or otherwise.

[Anupam Sanidhya]
Examination Roll Number - 510817049
B.Tech. Eight Semester
Department of Information Technology
Indian Institute of Engineering Science and Technology, Shibpur

INDIAN INSTITUTE OF ENGINEERING SCIENCE AND TECHNOLOGY, SHIBPUR

Design of a Smart Graphics System

Department of Information Technology
Bachelor of Technology in Information Technology
by Anupam Sanidhya

The modern graphics systems are heavily dependent on a centralized processing unit(GPU) which takes a considerable toll on the performance and provides no intelligent way to draw required images. In this project we aim to resolve this issue by making our graphics system divide its entire task of processing into several connected processing units each taking up the place of a pixel. Thus every pixel serves to be its own unit that feeds data to the next pixel as required.

Proposed Methodology

Circuit of a Single Pixel

The workflow of this project involves taking coordinates of end-points as input from the user. We have created the circuit for a line drawing algorithm for now, this organization can be later modified to any other figure we want to create. Once the input is provided, it will be fed to the CPU to calculate the necessary precomputed values which will take significantly lower computation power as compared to controlling each pixel. Then the data is processed and transferred to the neighboring pixels accordingly.

A. Input:- The circuit contains two 8 x 256 decoders for which each of the 256 outputs address a row and a column. After the input has been provided, its binary will be fed to the grid which will set the input pixel as high, depending on the outputs of the two decoders.

B. Pre-computes :- The algorithm that we are using in the current case is a modified version of Bressanham's Line Drawing algorithm. After the end point's coordinates are provided we will be calculating the pre-computes for a screen of 256 x 256 pixels. The precomputes will include

(i) dx : Difference in x-axis (always positive)
(ii) dy: Difference in y - axis(always positive)
(iii) sx : 0 if x0<x1 else 1
(iv) sy : 0 if y0<y1 else 1
(v) err: dx-dy.

These pre-computed values will be computed by the CPU and then will be fed to the starting pixel. Then from there on the values will be processed accordingly and fed the neighboring pixels.

C. Algorithm :-

```
while(true){
    path.push([y0,x0,err])
    state.grid.grid[y0][x0].err = err
    if (x0===x1 && y0===y1){
        break;
    }
    let e2 = 2*err
    if(e2 > -dy){
        err = err-dy
        x0 = x0+sx
    }
    if(e2 < dx){
        err = err + dx
        y0 = y0 + sy
    }
}
```

Variables :
x0,y0 : Coordinates of starting pixels.

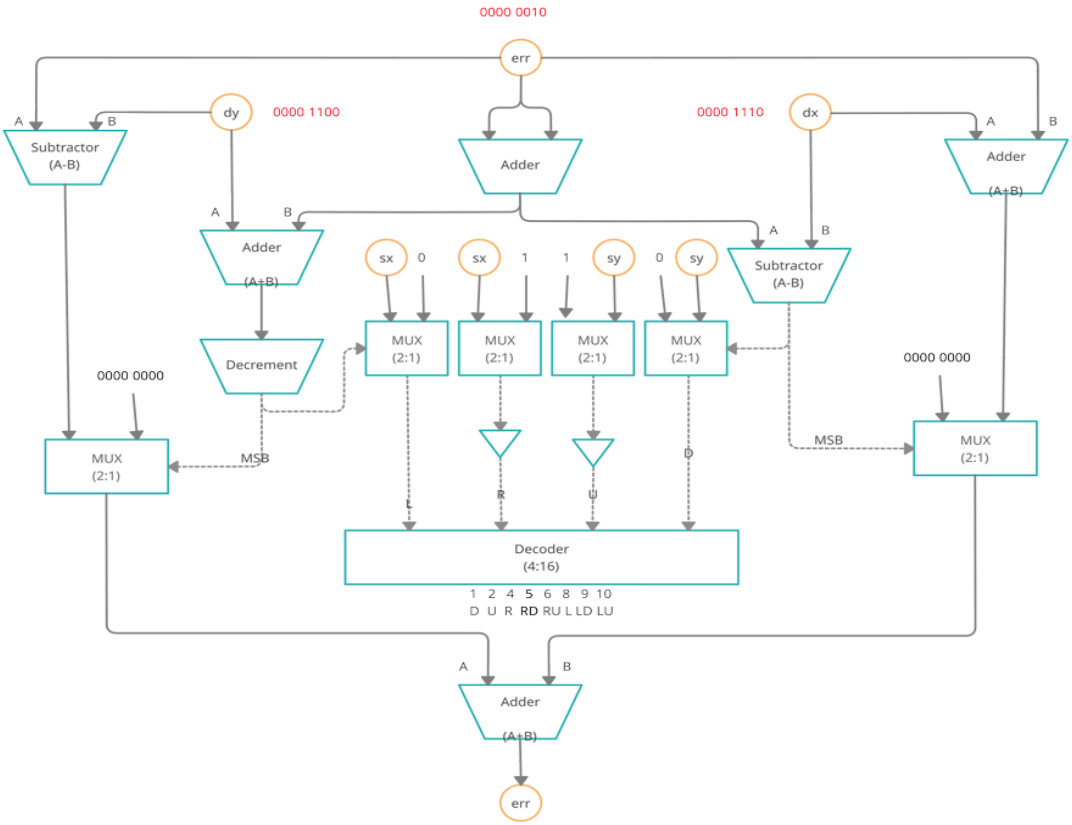
x1,y1 : Coordinate of the ending pixel.

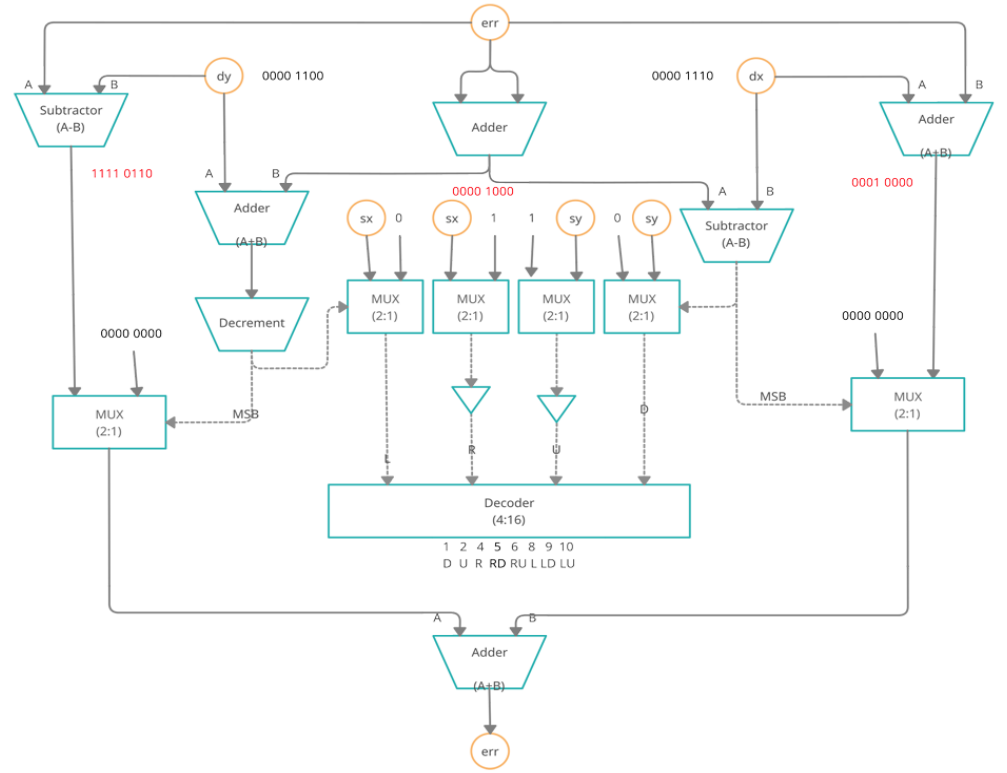
state : Matrix representing the 256 x 256 grid of pixels.

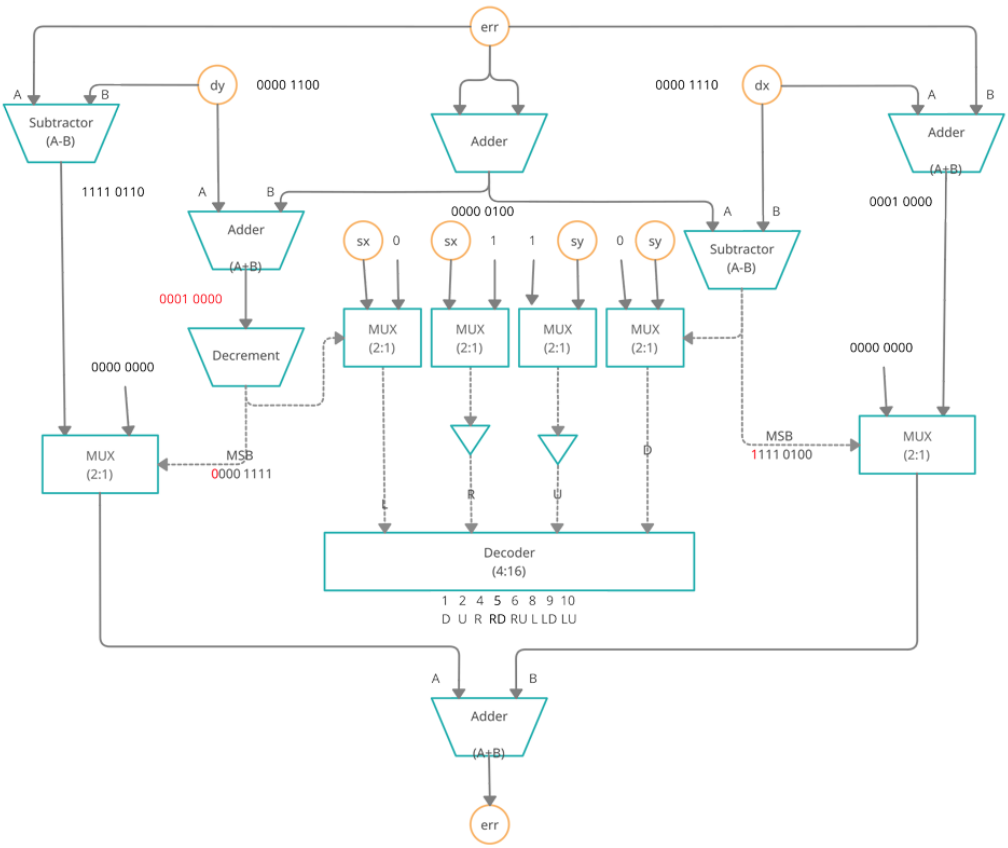
Working

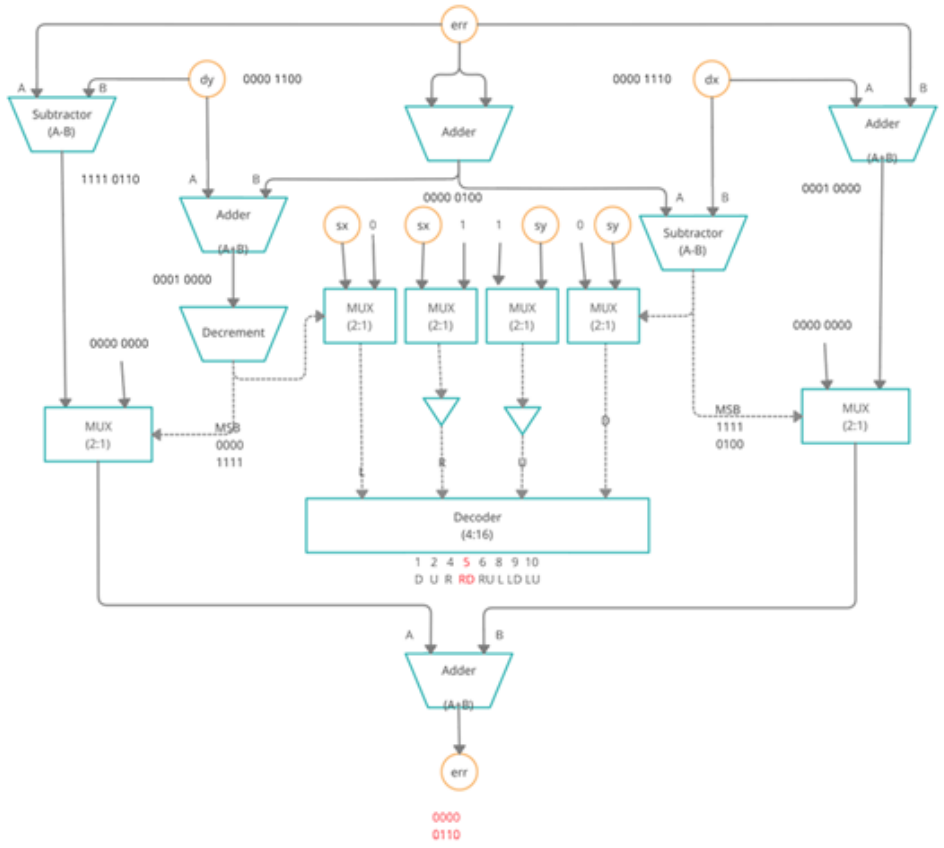
Working of a single pixel :

The calculated data from the previous pixels are being transferred to the current pixel. For the first pixel this data is the precomputes calculated by the CPU.



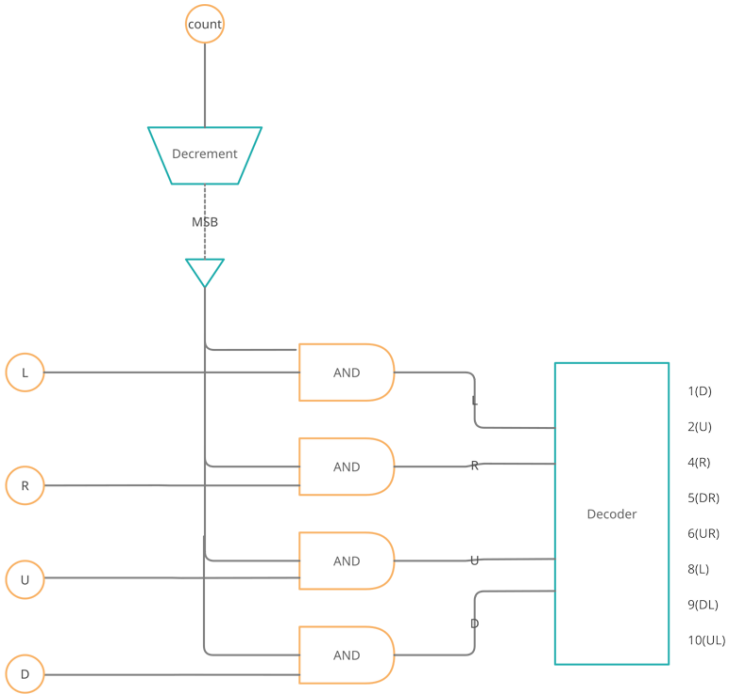

err
dy
0000 1100
Subtractor
(A-B)
1111 0110
Adder
0000 1000
0000 1110
dx
Adder
(A+B)
0001 0000
Adder
(A+B)
Decrement
sx
sy
MUX
(2:1)
Subtractor
(A-B)
0000 0000
MSB
Decoder
(4:16)
1 2 4 5 6 8 9 10
D U R RD RU L LD LU
Adder
(A+B)
err


err
dy
0000 1100
Subtractor
(A-B)
1111 0110
Adder
0000 0100
0000 1110
dx
Adder
(A+B)
0001 0000
Adder
(A+B)
0001 0000
Decrement
sx
sy
MUX
(2:1)
Subtractor
(A-B)
0000 0000
MSB
0000 1111
MSB
1111 0100
Decoder
(4:16)
1 2 4 5 6 8 9 10
D U R RD RU L LD LU
Adder
(A+B)
err



In the above 3 levels calculate if 2*err + dy >0 or 2*err-dx<0 then err changes to the above provided algorithm. We use the MSB of the above calculated value to select the value of sx or sy and update the value of ‘err’ as needed according to the algorithm. Depending on the values of sx and sy being fed to the decoder the next pixel is selected and the updated ‘err’ value is transferred to it.

Counter Decrement Circuit:

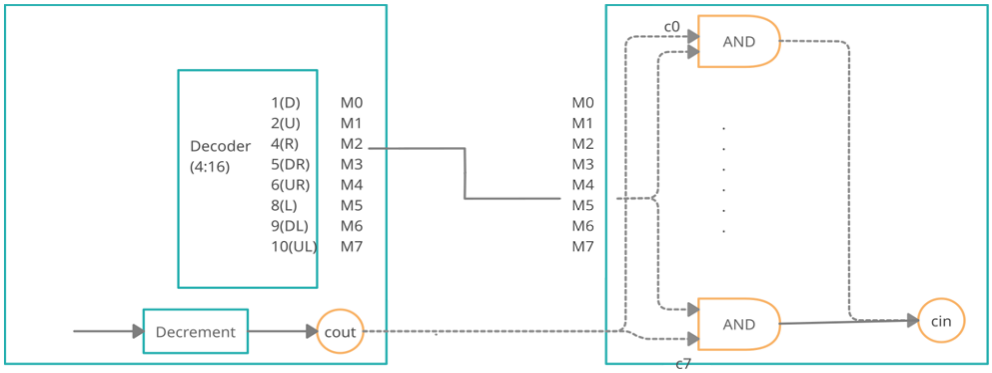


Count depicts the number of pixels that need to be lit next. Count Decrement circuit is used to decrement the counter and also to stop the flow to next pixels when the counter value hits 0.

The AND operation with the LRUD bits ensures that the decoder gets 0000 as input when the decremented counter value hits -1 whereas the calculated LRUD for any positive counter value.

The decoder's 0 output is not a part of M_{out} as a result no data is forwarded to the neighbouring pixels.

Transfer of Control from one pixel to next :



The next pixel that has to be lit up is being selected from the previous pixel as their corresponding pins are connected to each other.

Here an AND operation is performed between decremented count(c_{out}) value and M_{out} from the previous pixel. For the intended pixel this value will be the same as the C_{out} value, but for other neighboring pixels this value will be 0 so they won't be lit up. Only the intended pixel will be lit up.

Estimations

Size estimate for a 256x256 Screen :

A big concern of our project is its scalability. The circuit has to fit within a size of a pixel.

Average Area of a Pixel 6.76 x 10^4 nm^2.
Size of a small transistor is 100 nm^2 , so there can be 2704 transistors per pixel.
Number of two input transistors in an AND Gate is 2
Number of transistors needed to make a 2x1 Multiplexer is 10.
Number of transistors needed to make an 8-bit Full Adder is 96.
Number of transistors needed to make a 4 x 16 Decoder is 40.
Total number of transistors needed for our circuit is 840.

Thus the supposed circuit can be made possible given we have the possible fabrication method for transistors (MOSFET).

Number of Transistors for a 256x256 Screen :

Number of Multiplexers in one pixel = 6

Number of Adders in one pixel = 7

Number of Decoder in one pixel = 1

Number of 2 input and gates for input needed = number of bits in precomputes x 2

= 34 x 2 = 68.

Total number of transistors needed = 46x 10 + 7 x 96 + 40 + 68

= 840

Theoretical minimum Pixel Size :

Resolution	Avg. number of transistors per pixel	Minimum pixel size
256 x 256	840	287 nm
512 x 512	932	302 nm
1024 x 1024	1044	320 nm
2048 x 2048	1172	335 nm

Challenges and Limitations

- Collision of 2 messages when more than one inputs are being fed into a single pixel.
- Deletion of a specific line, in case of overlapping lines
- Current hardware design has been implemented only for line drawing and need to be expanded for curve drawing
- Accommodating space between pixel and circuit for avoidance of noise interference may compromise with the predicted resolution

Bibliography

- Algorithm for a computer control of a digital plotter, IBM Systems Journal
- New Efficient Design for XOR Function on the Transistor Level by Tripti Sharma, K.G.Sharma, B.P.Singh and Neha Arora
- Transistor Gates (gsu.edu)
- Pixel - Wikipedia