Transformation of the pixels in Tupper's self-referential formula

Prathamesh Deshmukh

24 may 2018

Introduction:

tupper's self-referential formula is really an amazing function because at a particular distance on y-axis it plots itself between k and k+17 on the y-axis and 0 to 106 on the x-axis

$$\frac{1}{2} < \left\lfloor \mathrm{mod}\left(\left\lfloor \frac{y}{17} \right\rfloor 2^{-17\lfloor x \rfloor - \mathrm{mod}(\lfloor y \rfloor, 17)}, 2 \right) \right\rfloor,$$

where the value of k is

960 939 379 918 958 884 971 672 962 127 852 754 715 004 339 660 129 306 651 505 519 271 702 802 395 266 424 689 642 842 174 350 718 121 267 153 782 770 623 355 993 237 280 874 144 307 891 325 963 941 337 723 487 857 735 749 823 926 629 715 517 173 716 995 165 232 890 538 221 612 403 238 855 866 184 013 235 585 136 048 828 693 337 902 491 454 229 288 667 081 096 184 496 091 705 183 454 067 827 731 551 705 405 381 627 380 967 602 565 625 016 981 482 083 418 783 163 849 115 590 225 610 003 652 351 370 343 874 461 848 378 737 238 198 224 849 863 465 033 159 410 054 974 700 593 138 339 226 497 249 461 751 545 728 366 702 369 745 461 014 655 997 933 798 537 483 143 786 841 806 593 422 227 898 388 722 980 000 748 404 719

But that is not the only fascinating thing about this formula ,it doesn't only plot itself but it plots every possible combinations of the pixels between k and k+17 on the y-axis and 0 to 106 on the x-axis

I.e. tupper's self-referential formula plots all possible combination of these1802 pixels and the value of k is used for sliding over the y-axis

Abstract:

My research idea start with a question that if there is some graphical formation at a particular value of k then what can do to change the graphical formation or to change its position

In his research paper ,I have shown that

- *How to change any graphical formation into other graphical formation by applying some kind of operation to the value of the k*
- *How can this different graphical formation be used as a frame to create a film/ motion picture*

Representation

consider a graph of tupper's self-referential function

$$\frac{1}{2} < \left\lfloor \mathrm{mod}\left(\left\lfloor \frac{y}{17} \right\rfloor 2^{-17\lfloor x \rfloor - \mathrm{mod}(\lfloor y \rfloor, 17)}, 2\right)\right\rfloor,$$

For X from 1 to 106

Y coordinate from k to k+17

The graphical structure can be represent as p_i

Where i represent the number of the pixel from the left bottom corner The pixel at left bottom corner of the graph will be represented by p_1

A pixel above it will be represented as p_2 and above it will be p_3

Pixel at right neighboring to the left bottom corner will be (neighboring column) will be p_{18}

Now ,

If the Pixel is black then the value of p_i value will be 1

If the Pixel is white then the value of p_i value will be 0

The series p_i will give us a binary value

k will be equals to

$k=17(p_1\times2^0 +p_2\times2^1 +p_3\times2^2 +p_4\times2^3 .....+p_{1802}\times2^{1801})$ ....(A)

(*the total number of pixels are106×17= 1802*)

the equation A can be represent as

$$k = 17\sum_{i=1}^{1802}((p_i)2^{i-1}$$

Transformations

- **To add or remove a pixel from the graph**

- **To remove any pixel the graph**

Consider any graph between the value of k and k +17 and let pi be any pixel position

If the pixel black i.e.it present in the graph then to remove it we have to subtract $17(2^{i-1})$ from the value of k

Proof:

By equation A

$k=17(p_1\times2^0 +p_2\times2^1 +p_3\times2^2 +p_4\times2^3 .....+p_{1802}\times2^{1801})$

$k=17\times p_1\times2^0 +17\times p_2\times2^1 +17\times p_3\times2^2 +17\times p_4\times2^3 .....+17\times p_{1802}\times2^{1801})$

i.e. value of k is addition of individual values of each pixel

Now consider any black pixel p_i

Consider this term

$k-17(2^{i-1})$

$=17\times p_1\times2^0 +17\times p_2\times2^1 +17\times p_3\times2^2 +17\times p_4\times2^3 ...+17\times p_i\times2^{i-1}...+17\times p_{1802}\times2^{1801}) -17(2^{i-1})$

In the pixel is present in the graph then $p_i= 1$

$k=17\times p_1\times2^0 +17\times p_2\times2^1 +17\times p_3\times2^2 +17\times p_4\times2^3 ..+17\times1\times2^{i-1}....+17\times p_{1802}\times2^{1801}) -17(2^{i-1})$

$k=(17\times p_{11}\times2^0 +17\times p_{12}\times2^1 +17\times p_{13}\times2^2 +17\times p_{14}\times2^3 + ...+17\times p_{106\ 17}\times2^{121})$

So pixel p_i get removed from the graph i.e. there is a transformation by removing a pixel

It represent as

$k' = k -17(2^{i-1})$

Where k is the original value

Let k' be the value after this transformation

$k' = k - 17(2^{i-1})$

To remove many pixels

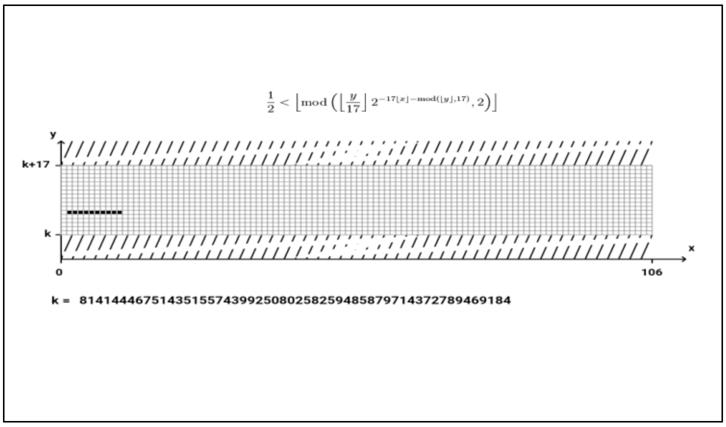
P_a , p_b ,p_c....

By same logic

$k' = k - (17(2^{a-1}) + 17(2^{b-1}) + 17(2^{c-1}) + ...)$

E.g:

Consider a This graph



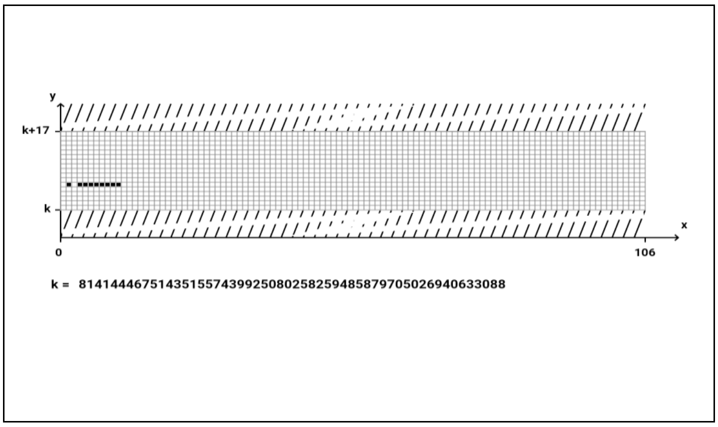
To remove its pixel p40

$k' = k - 17(2^{40-1})$

$k' = 8141444675143515574399250802582594858797143727894 69184 - 17(2^{40-1})$

$k' = 814144467514351557439925080258259485879714372789469184 - 18691697672192$

$k' = 814144467514351557439925080258259485879705026940633088$



- **To add any pixel to the graph**

To add a pixel in position of pi just add $17(2^{i-1})$ in the value of k

Proof: By equation A

$k=17(p_1\times2^0 + p_2\times2^1 + p_3\times2^2 + p_4\times2^3 \,.....+ p_{1802}\times2^{1801})$

$k=17\times p_1\times2^0 + 17\times p_2\times2^1 + 17\times p_3\times2^2 + 17\times p_4\times2^3 \,.....+ 17\times p_{1802}\times2^{1801})$

i.e. value of k is addition of individual values of each pixel

Now consider any white pixel p_i

I.e. the pixel is absent from the graph

So

$p_i= 0$

Consider

$k + 17(2^{i-1})$

$=17\times p_1\times2^0 + 17\times p_2\times2^1 + 17\times p_3\times2^2 + 17\times p_4\times2^3 \,...+ 17\times p_i\times2^{i-1} ...+ 17\times p_{1802}\times2^{1801}) + 17(2^{i-1})$

Pixel p_i is white which means

$p_i = 0$

$k + 17(2^{i-1})$

$=17\times p_1\times2^0 + 17\times p_2\times2^1 + 17\times p_3\times2^2 + 17\times p_4\times2^3 \,...+ 17\times0\times2^{i-1} ...+ 17\times p_{1802}\times2^{1801}) + 17(2^{i-1})$

$=17\times p_1\times2^0 + 17\times p_2\times2^1 + 17\times p_3\times2^2 + 17\times p_4\times2^3 \,...+ 17_i\times2^{i-1} ...+ 17\times p_{1802}\times2^{1801})$

So pixel p_i get added to the graph i.e. there is a transformation by adding a pixel.

This adding of a pixel can be represented by

$k' = k + 17(2^{i-1})$

Where k is the original value

Let k' be the value after this transformation

Similarly

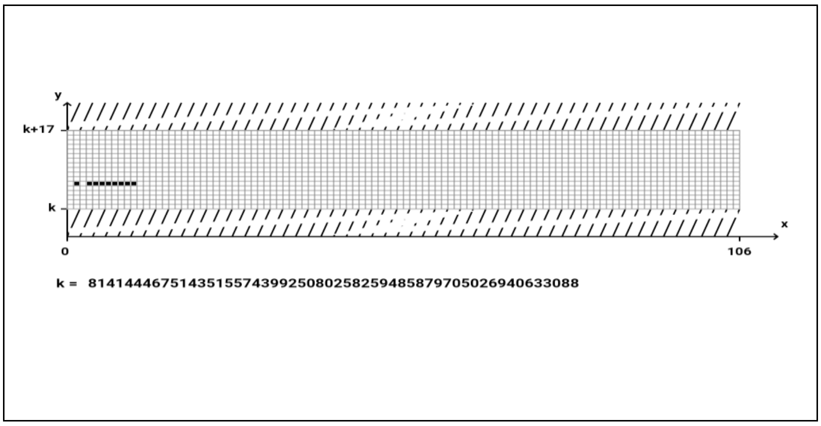
To remove many pixels

P_a , p_b , p_c....

$k' = k + (17(2^{a-b}) +17(2^{b-1}) +17(2^{c-1})+ .....)$

E.g:

Consider this graph



Where k=814144467514351557439925080258259485879705026940633088

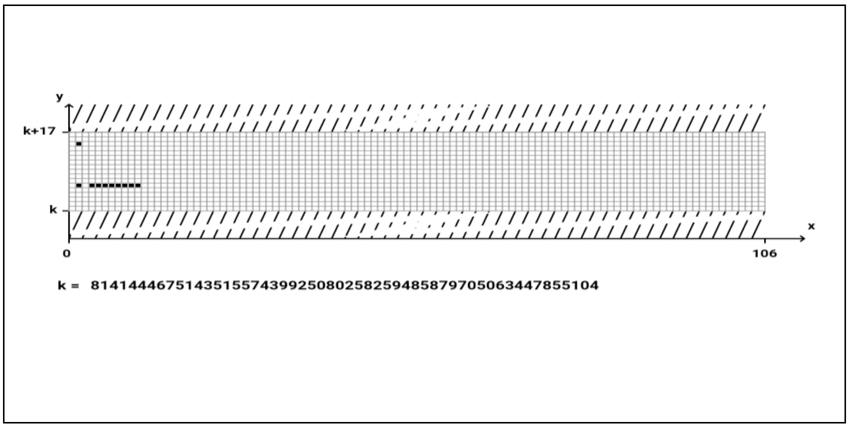
We will add a pixel p_{32}

$k'= k+17(2^{32-1})$

$k'=814144467514351557439925080258259485879705026940633088+17(2^{31})$

$k'=814144467514351557439925080258259485879705026940633088+36507222016$

$k'=814144467514351557439925080258259485879705063447855104$



- **In general to add or remove pixel**

$k' = k \pm 17(2^{i-1})$

In which addition is used to add a pixel to the graphical formation

And substation is used to remove the pixel from the graphical formation

- **spatial transformation**
  - **Transformation in the vertical direction**

To move the pixel in ***upward direction*** *we just have multiple it by 2*

Proof:

By equation A

$k=17(p_1\times2^0 +p_2\times2^1 +p_3\times2^2 +p_4\times2^3 .....+p_{1802}\times2^{1801})$

Now consider the term k×2

$k\times2=2\times\{17(p_1\times2^0 +p_2\times2^1 +p_3\times2^2 +p_4\times2^3 .....+p_{1802}\times2^{1801})\}$

$=17(p_1\times2^0\times2 +p_2\times2^1\times2 +p_3\times2^2 \times2+p_4\times2^3\times2 .....+p_{1802}\times2^{1801}\times2)$

$=17(p_1\times2^1 +p_2\times2^2 +p_3\times2^3 +p_4\times2^4 .....+p_{1802}\times2^{1802})$

It has all 2's power higher than that is in k

So all the pixels in the graph will move in upward direction

(*As Value of the upper pixel is twice of the just below pixel*)

So this upward transformation can be represented as k'= k*2

Where k is the original value

Let k' be the value after this upward transformation

Similarly

To move Two Steps in upward direction from the original position (by the same logic)

k'= k*2*2

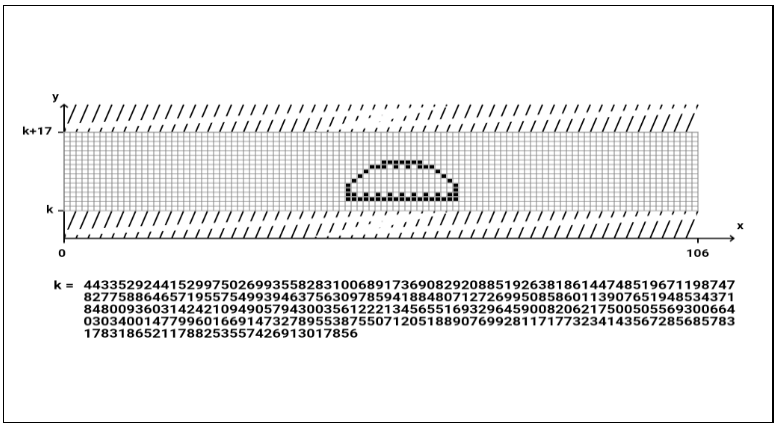
k'= k*4

k'= $k*2^2$

Generalizing this we get

k'= $k*2^{n.}$ ....(1)

E.g: consider a pixilated UFO



k=443352924415299750269935582831006891736908292088519263818614474851967119874782775886465719557549939463756309785941884807127269950858601139076519485343718480093603142421094905794300356122213456551693296459008206217500505569300664030340014779960166914732789553875507120518890769928117177323414356728568578317831865211788253557426913017856

We want move it by 5 pixels

I.e. n=5

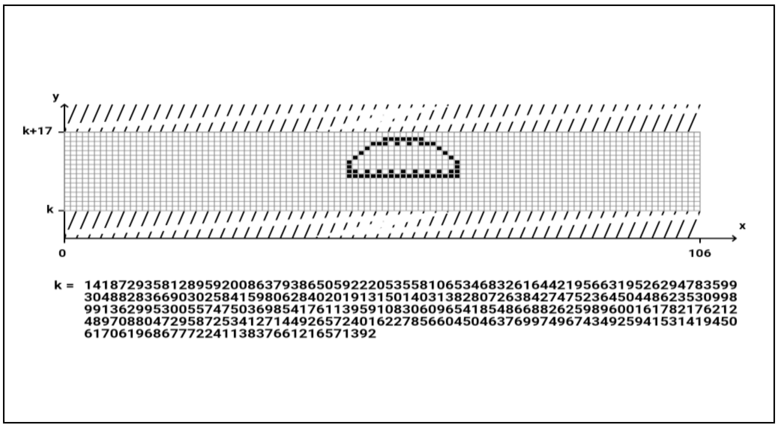
k'= $k*2^n$

k'= $k*2^5$

So

k'=443352924415299750269935582831006891736908292088519263818614474851967119
874782775886465719557549939463756309785941884807127269950858601139076519485
343718480093603142421094905794300356122213456551693296459008206217500505569
300664030340014779960166914732789553875507120518890769928117177323414356728
56857831783186521178825355742691301785 6×32

k'=141872935812895920086379386505922205355810653468326164421956631952629478
359930488283669030258415980628402019131501403138280726384274752364504486235
309989913629953005574750369854176113959108306096541854866882625989600161782
176212489708804729587253412714492657240162278566045046376997496743492594153
1419450617061968677722411383766121657 1392



★ ***To move the pixel in downward direction just have divide it by 2***

Similarly by above theorem for the downward transformation will be

k'= k/2

Where k is the original value

Let k' be the value after this downward transformation

Similarly

To move Two pixel in downward direction from the original position

k'= k/2*2

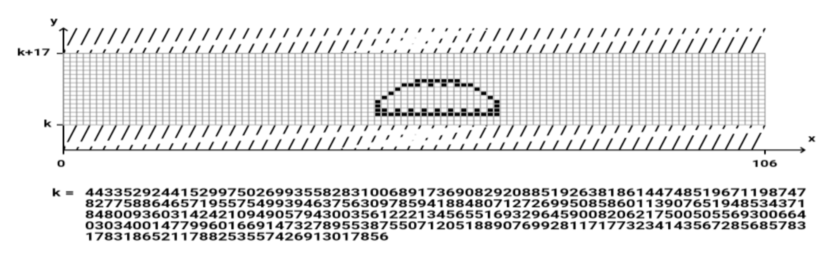
k'= k/4

$k' = k/2^2$

Generalizing this we get

$k' = k/2^n$ ....(2)

E.g: consider a pixelated UFO



Where

k=443352924415299750269935582831006891736908292088519263818614474851967119874782775886465719557549939463756309785941884807127269950858601139076519485343718480093603142421094905794300356122213456551693296459008206217500505569300664030340014779960166914732789553875507120518890769928117177323414356728568578317831865211788253557426913017856

We have To move the UFO (*graphical structure*) downward by 2 pixel is n=2

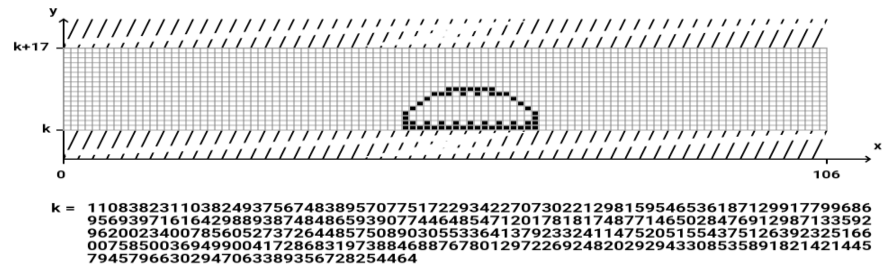
$k' = k/2^n$

$k' = k/2^2$

k'=443352924415299750269935582831006891736908292088519263818614474851967119874782775886465719557549939463756309785941884807127269950858601139076519485343718480093603142421094905794300356122213456551693296459008206217500505569300664030340014779960166914732789553875507120518890769928117177323414356728568578317831865211788253557426913017856÷4

k'=110838231103824937567483895707751722934227073022129815954653618712991779968695693971616429889387484865939077446485471201781817487714650284769129871

3359296200234007856052737264485750890305533641379233241147520515543751263923251660075850036949900417286831973884688767801297226924820292943308535891821421445794579663029470633893567282544464



- ❖ **Transformation to left or right.**
- ➢ Move the pixels in right direction

Consider any pixel p_i, where to move it to the right neighbouring place

we have to multiply original k by 2^{17}

Proof:

By equations A

$k=17(p_1\times2^0 +p_2\times2^1 +p_3\times2^2 +p_4\times2^3 .....+p_{1802}\times2^{1801})$

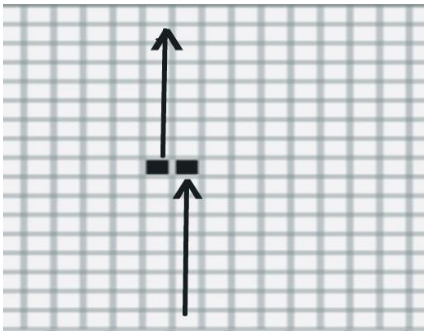
Consider

$k\times2^{17} =17(p_1\times2^0 +p_2\times2^1 +p_3\times2^2 +p_4\times2^3 .....+p_{1802}\times2^{1801})\times2^{17}$

$=17(p_1\times2^0 \times2^{17}+p_2\times2^1 \times2^{17}+p_3\times2^2 \times2^{17}+p_4\times2^3\times2^{17} .....+p_{1802}\times2^{1801}\times2^{17})$

$=17(p_1\times2^{17} +p_2\times2^{18} +p_3\times2^{19} +p_4\times2^{20} .....+p_{1802}\times2^{1818})$

But here 2's power are different than present in k ,which indicates the position of the arrangement of pixel is different (*location of the graphical structure is Changed*)



2's Power in k and $k \times 2^{17}$ are differ by 17

By equation 2

This implies there will be transformation in upward direction by 17 pixel from the original position

Now, as we know there are 17 pixel in a column

So by 17 pixel transformation in upward direction from the original position will result in the position right to the original position

This will be represented as

$k' = k*2^{17}$

To move the pixel pattern by Two pixel position in right from the original position

$k'=k*2^{2(17)}$

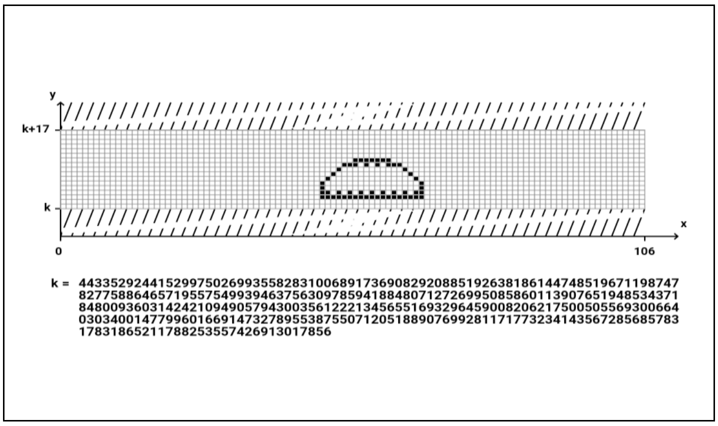
To move it n steps in right from the original position will be

$k' = k*2^{n(17)}$

Where n is number of pixel It moved from its original position

E.g

Consider a UFO shaped graphical structure for which



k=4433529244152997502699355828310068917369082920885192638186144748519671198
7478277588646571955754993946375630978594188480712726995085860113907651948 53

437184800936031424210949057943003561222134565516932964590082062175005055693 006640303400147799601669147327895538755071205188907699281171773234143567285 68578317831865211788253557426913017856

We want To move it in right direction by 7 pixel, here n=12

So

$k' = k*2^{12(17)}$

Here n=12

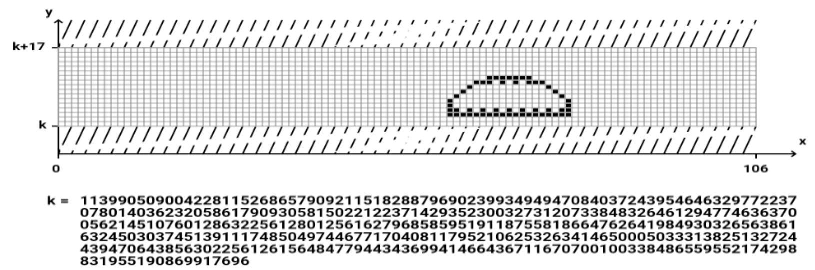
$k' = k*2^{7(17)}$

$k'=k\times 2^{204}$

k'=k×25,711,008,708,143,844,408,671,393,477,458,601,640,355,247,900,524,685,364,822,016

k'=4433529244152997502699355828310068917369082920885192638186144748519671198747827758864657195575499394637563097859418848071272699508586011390765194853437184800936031424210949057943003561222134565516932964590082062175005055693006640303400147799601669147327895538755071205188907699281171773234143567285685783178318652117882535574269130178 56×25,711,008,708,143,844,408,671,393,477,458,601,640,355,247,900,524,685,364,822,016

k'=11399050900422811526865790921151828879690239934949470840372439546463297722370780140362320586179093058150221223714293523003273120733848326461294774636370056214510760128632256128012561627968585951911875581866476264198493032656386163245030374513911174850497446771704081179521062532634146500050333138251327244394706438563022561261564847794434369941466436711670700100338486559552174298831955190869917696



★ To move the pixel in a left direction

Conversely to move it in a left direction we have to divided by 2^{17}

Which can consider by 17 pixel transformation in downward direction from the original position will result in the position left to the original position ....by (3)

$k' = k/(2^{17})$

To move 2 to pixels in left from the original position

$k' = k/(2^{2(17)})$

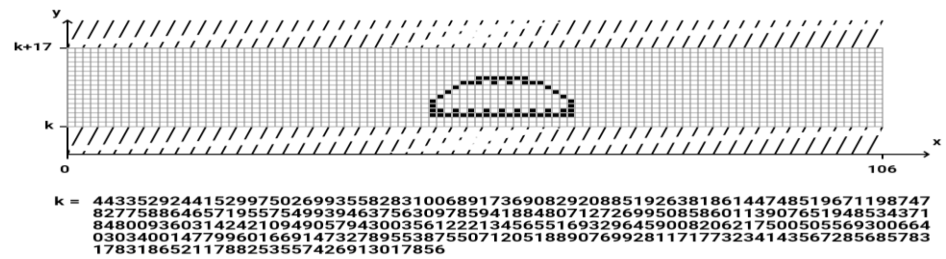
To move it n pixel in left from the original position

$k' = k/(2^{n(17)})$

Where n is number of pixels from the original position

E.g:

Consider a UFO shaped graphical structure for which



k=443352924415299750269935582831006891736908292088519263818614474851967119874782775886465719557549939463756309785941884807127269950858601139076519485343718480093603142421094905794300356122213456551693296459008206217500505569300664030340014779960166914732789553875507120518890769928117177323414356728568578317831865211788253557426913017856

We want To move it in right direction by 16 pixel, here n=16

So

$k' = k/2^{n(17)}$

Here n=16

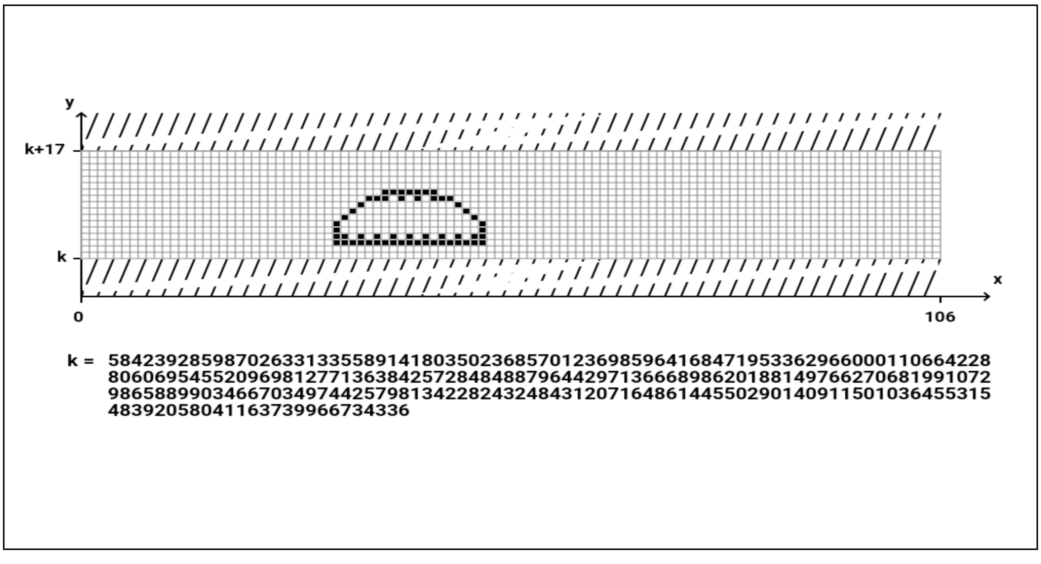
$k' = k/2^{16(17)}$

$k'=k/2^{272}$

k'=k×7588550360256754183279148073529370729071901715047420004889892225542594864082845696

k'=443352924415299750269935582831006891736908292088519263818614474851967119874782775886465719557549939463756309785941884807127269950858601139076519485343718480093603142421094905794300356122213456551693296459008206217500505569300664030340014779960166914732789553875507120518890769928117177323414356728568578317831865211788253557426913017856÷7588550360256754183279148073529370729071901715047420004889892225542594864082845696

k'=584239285987026331335589141803502368570123698596416847195336296600011066422880606954552096981277136384257284848879644297136668986201881497662706819910729865889903466703497442579813422824324843120716486144550290140911501036455315483920580411637399667343336



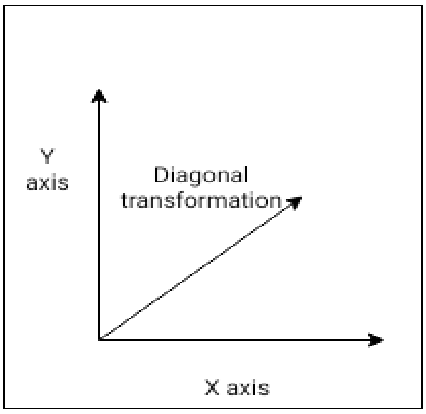
❖ Complex transformation

Now as we know how to move it in up or down and right or left we can do any Complex transformation from it

Like diagonal, elliptical transformation

For example

Consider diagonal directional transformation



It can be achieved applying both up and right transformation

So

$k' = (k*2^{n(17)})*2^{n^\circ}$

Where n represent number of pixels variation in right direction by from the original position

Where n° represent number of pixels variation in up direction by from the original position

Different value of n and n° will you different variation (i.e. different curve)

To make a continuous diagonal (or any) transformation the value of the k' should be put in the equation in the place of k to get value of k'' and continuing this process Will give us a series of k, k', k'', k'''......

- **Transformation on all pixel or individual pixel**
  - **Transformation on all pixel at the same time**

Consider any graphical formation, it can be considered as an addition of the k's value for the individual pixel.

$k=17(p_1\times2^0 +p_2\times2^1 +p_3\times2^2 +p_4\times2^3 .....+p_{1802}\times2^{1801})$

$k=17\times p_1\times2^0 +17\times p_2\times2^1 +17\times p_3\times2^2 +17\times p_4\times2^3 .....+17\times p_{1802}\times2^{1801})$

Which can be written as

$k=k_1+k_2+k_3.....k_{1802}$

Where

$k_1 =17\times p_1\times2^0$

$k_2=17\times p_2\times2^1$

$k_3=17\times p_3\times2^2$

And so on

Which are the value of k for individual pixel

Now apply a transformation,

$k'= k\times t$

Where k' is the value after transformation

k is value before transformation

And t is any transformation (up, down, diagonal etc.)

$k' = t\times(17\times p_1\times2^0 +17\times p_2\times2^1 +17\times p_3\times2^2 +17\times p_4\times2^3 .....+17\times p_{1802}\times2^{1801})$

$k'= t\times17\times p_1\times2^0 + t\times17\times p_2\times2^1 + t\times17\times p_3\times2^2 + t\times17\times p_4\times2^3 .....+ t\times17\times p_{1802}\times2^{1801}$

$k'=t\times k_1+t\times k_2+t\times k_3.....t\times k_{1802}$

By this we can see there will be a transformation on every individual pixel

So whole graphical formation will go under the transformation

Such type of transformation will result in same transformation on every individual pixel

- **Different transformation on the individual Pixar**
  (*Transformation on the individual pixels where the rest remains same or under different transformation*)

Consider a graph, the value of k can be written as

$k=k_1+k_2+k_3.....k_{1802}$

If you apply different transformations on every pixel we will get

$k' =t_1\times k_1+t_2\times k_2+t_3\times k_3.....+t_{1802}\times k_{1802}$

......................(B)

(*the total number of pixels are106×17= 1802*)

Where t_1 is the first transformation

t_2 is the second transformation

t_3 is the third transformation

And so on

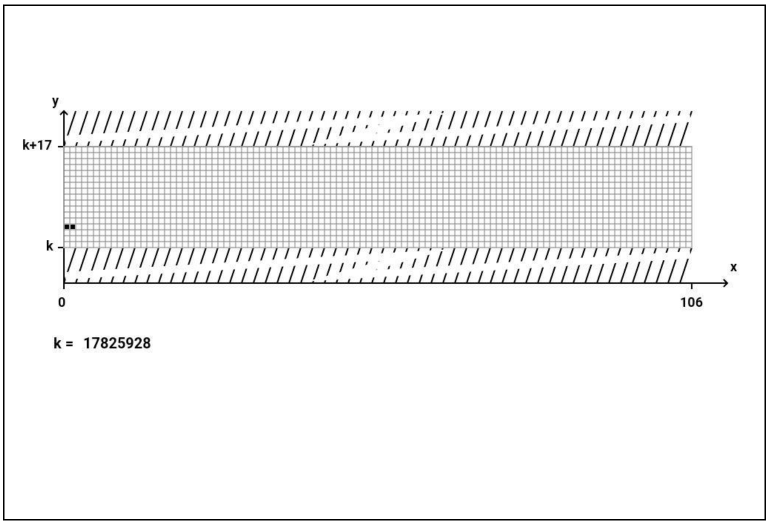
E.g

Consider two the pixel p_4, p_{21}

we will give them the different transformation (up transformation to the P_4 and down transformation to the p_{21}

Here $k= k_4+ k_{21}$

Where k = 17825928



$k' = t_1 \times k_4 + t_2 \times k_{21}$

Where t_1 is the first transformation

t_2 is the second transformation

k_4 value of the first pixel.

$k_4 = 17 \times (p_4 \times 2^{4-1}) = 136$

k_{21} value of the second pixel

$k_{21} = 17 \times (p_{21} \times 2^{21-1}) = 17825792$

Let t_1 be a downward spiral transformation (divide by 2)

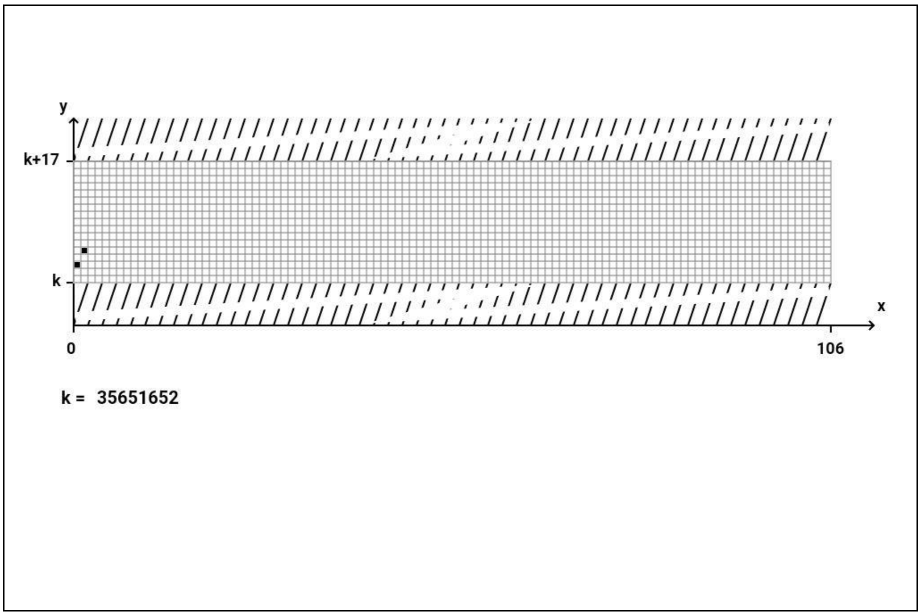
t_2 be a upward spiral transformation (multiply by 2)

$k' = k_4 t_1 + k_{21} t_2$

$k' = k_4/2 + k_{21} \times 2$

$k' = 136/2 + 17825792 \times 2$

$k' = 35651652$



- **Transformation and the addition or remove the pixel**

By equation (B)

$k' = t_1 \times k_1 + t_2 \times k_2 + t_3 \times k_3 ..... + t_{1802} \times k_{1802}$

By this equation, We can apply any spatial transformation on any pixels

we modify this equation with new feature such that to add or remove a pixel

$k' = t_1 \times k_1 + t_2 \times k_2 + t_3 \times k_3 .... + t_{1802} \times k_{1802} \pm (17 \times 2^{i-1})$

Where i is the coordinates of the pixel p_i

In which addition is used to add pixel in the graphical formation

And substation is used to remove pixel from the graphical formation

This can be generalized

$k' = t_1 \times k_1 + t_2 \times k_2 + t_3 \times k_3 .... + t_{1802} \times k_{1802} + \sum (17 \times 2^{i-1}) - \sum (17 \times 2^{j-1})$………(C)

In which $\sum (17\times2^{i-1})$ is for addition of pixel to the graph

and$\sum (17\times2^{j-1})$ is for removing pixel to the graph

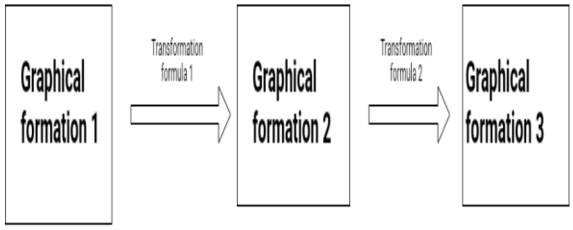
then by equation (C)

we can convert any graphical formation into any other graphical formation

So now we can apply any transformation on an individual pixel or group of pixel
Then by this equation(C) spatial transformation and addition & remove pixel transformation can be used to convert any graphical formation into any other graphical formation

And we can apply this transformation repeatedly creating different formations each time

There are two ways to do it



- **same transformational formula:**

In which graphical formation 1is converted the formation 2 by a transformation formula (in form of (A))

then the <u>same</u> transformational formula will be applied to graphical formation 2 to get graphical formation 3 and so on repeating this process

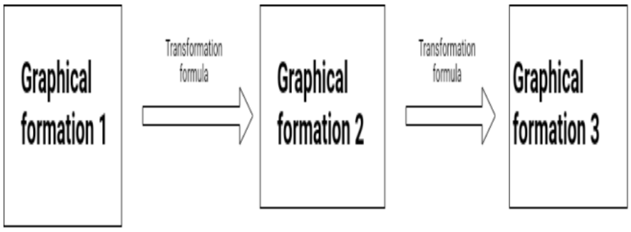
So by using the same transformation formula we will get a series of formation (movement in some definite pattern)

This series of formation can be use to create a film (as shown in eg 3.1)

- **variable transformational formula:**

In this method formation1get transferred to the formation 2 by the use of a transformational formula (formula 1) and then the formation 2 get converted to the formation 3 by using formula 2 And so on

Where all the transformation formula may be different



- **Framing**

Now the above series of formations can be use to create a motion picture /film

In which each formation created by the transformational formula is work as a frame of the film

To show this different formation in a particular sequence a variable f is used

Consider

$k° = f_1 \times k + f_2 \times k' + f_3 \times k'' + f_4 \times k''' .... f_n \times k^{n-1}$ ....(D)

$k°$ is the value which will be used to the representation of graphs /frames

Where k, k' ,k'' ,k ''' are different formation

f_1 f_2 $f_3 ... f_n$ are the framing variable

(Where n is a number of frames)

At a first time $f_1 = 1$, $f_2=0$ $f_3=0 .... f_n=0$

Which will give

$k° = 1 \times k + 0 \times k' + 0 \times k'' + 0 \times k''' .... 0 \times k^{n-1}$

$k° = k$

After that for the next frame

$f_1 = 0$, $f_2=1$ $f_3=0 .... f_n=0$

Which will result in

$k° = 0 \times k + 1 \times k' + 0 \times k'' + 0 \times k''' .... 0 \times k^{n-1}$

$k° = k'$

And so on

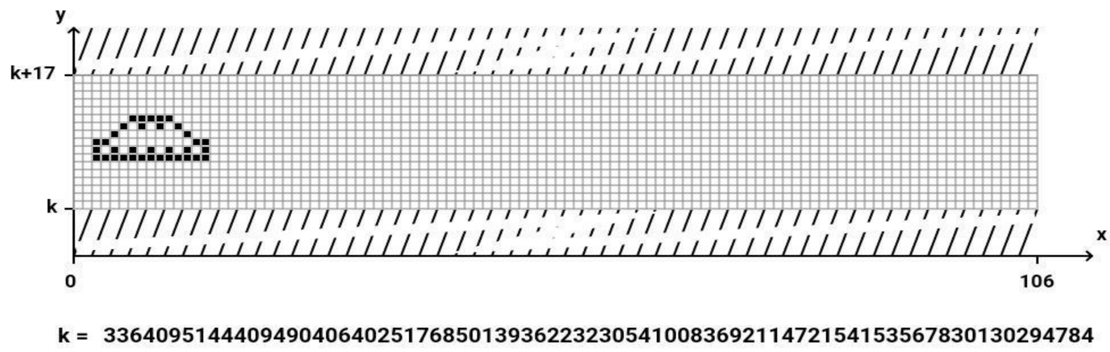
In f's 1 moves to on right side indicating the forward time flow

So by applying different transformation formulas and by applying frame variation the variation in the graphical formation can be shown as a film

Eg3.1

Consider a UFO given by

k=3364095144409490406402517685013936223230541008369211472154153567830130294784



We want to move it continuously in right direction

For this, we have to multiply by 2^{17}

$k' = k \times 2^{17}$

$k'' = k' \times 2^{17} = k \times 2^{2\times17}$

$k''' = k \times 2^{3\times17}$

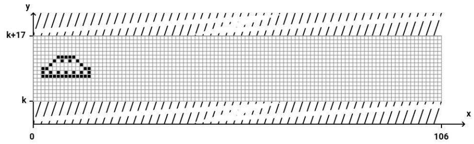
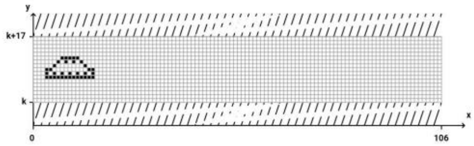
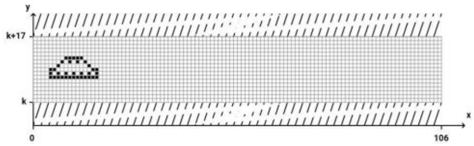
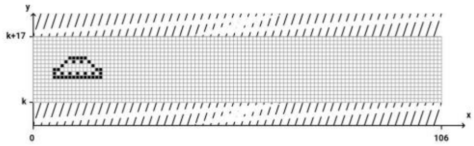
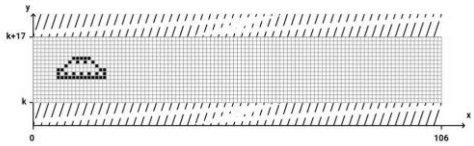
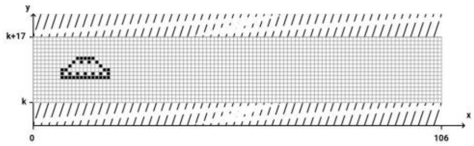
$k^{n} = k \times 2^{n\times17}$

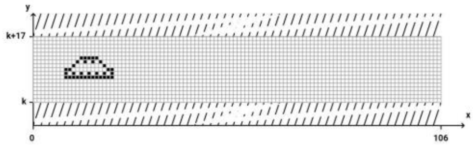
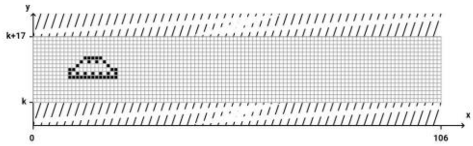
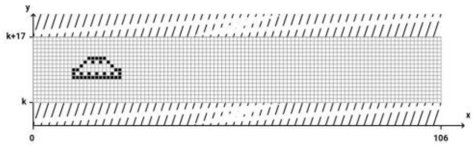
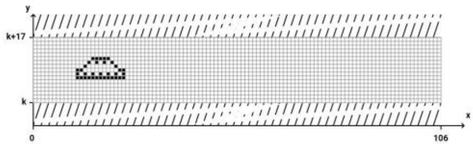
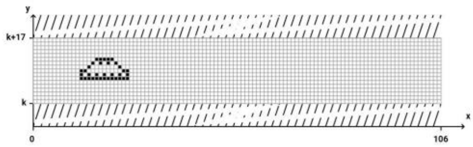
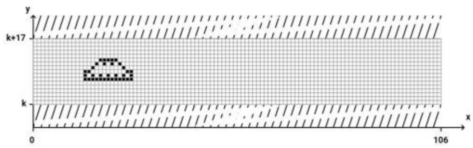
By the framing formula

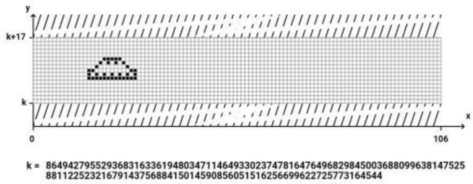
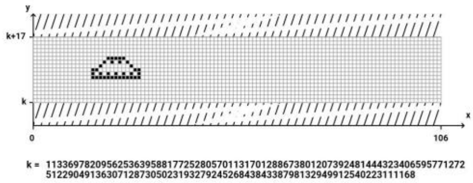
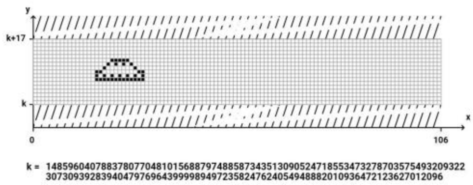
$k^{\circ} = f_1 \times k + f_2 \times k' + f_3 \times k'' + f_4 \times k''' .... f_n \times k^{n-1}$

$k^{\circ} = f_1 \times k + f_2 \times k \times 2^{17} + f_3 \times k \times 2^{2\times17} + f_4 \times k \times 2^{3\times17} .... f_n \times k^{n-1}$

this Framing formula will give us different frame's

Frame 1	 k = 3364095144409490406402517685013936223230541008369211472154153567830130294784
Frame 2	 k = 440938678768040726547990798010146648651273471048969286078189216442630837997928448
Frame 3	 k = 5779471450348463411009824987678594153201971639733050226484041697756850919806 4477536256
Frame 4	 k = 7575268819400737962078797807850086928484888267630903592857163134083859637608 707199632146432
Frame 5	 k = 9929056346964935261655921862705265938903712750149177957229740863106396504206 48470070184697135104
Frame 6	 k = 1301421273509387994615764990388504617143987437587553053210016594409081602599 35236269039248622892351488

Frame 7	 k = 17057988916142250323027754882020207717829672141947575379034329506238714381590231288255512395499746294235136
Frame 8	 k = 22358247232165970343398938878961526659913707869893526000807876370417207714237947954142265207029427462779877745792
Frame 9	 k = 29305401812144580648499857167432452223682095179226842399778899716233242495205963142453349852157611164014840981644 9024
Frame 10	 k = 38411176263214144747601732786497063778624595793316206870238199436061235603316360010076454718220024104897532371461606473728
Frame 11	 k = 50346296951720043803576543197917431435918830198215418668958612764834182729978819392407410728265349994771293629922156837244 76416
Frame 12	 k = 65989898340558495814223846700374335731687529117404913557777432923123459987837838154016241389751959545146629986611569409713457 2797952

Frame 13	
Frame 14	
Frame 15	

And so on

We will get a smooth motion film of the UFO traveling towards the right direction.

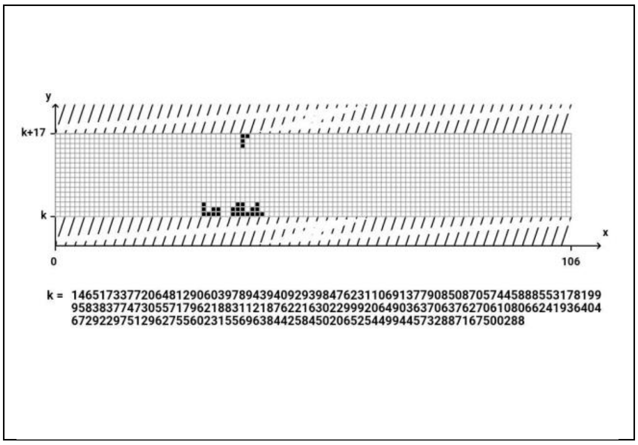
The video of this frames variation is on this link

https://kten.tk/Example1

Eg3.2

We will create a Tetris game film by the equation in the form of (D)

Consider the first frame



Wherek=146517337720648129060397894394092939847623110691377908508705744588855317819995838377473055717962188311218762216302299920649036370637627061080662419364046729229751296275560231556963844258450206525449944573288716750028 8

We want to move the L shape structure at the top (as in Tetris game)

We have to calculate the value for the L shaped structure say k_s

$k_s=17(p^{661}*2^{660}+p_{662}*2^{661}+p_{663}*2^{662}+p_{714}*2^{713})$

(as shown in figure)

Now

Let k-ks represent as kr (for graphical structure in rest at the bottom)

So

k=ks+kr

First We have to move the structure down one pixel per frame

If we have to divide k_s by for every frame

k'=kr+(ks/2)

k''=kr+(ks/4)

k'''=kr+(ks/8)

and soon

So the framing formula will be

$k°=f_1×k+f_2×k'+f_3×k''+f_4×k'''........+f_n×k^{n-1}$

$k°=(f_1×k)+f_2×(kr+ks/2)+f_3×(kr+ks/4)+f_4×(kr+ks/8)+........+f_{15}×k^{15-1}$

$k°=(f_1×k)+f_2×(kr+ks/2)+f_3×(kr+ks/4)+f_4×(kr+ks/8)+.......+f_{15}×(kr+ks/2^{14})$

(*Here there will be 15 frames as the bottom of the graph 14 pixel away from the structure)*

We just don't want to it freely falling ,we want to fall it wherever we want

So we have to apply the right or left transformation

The suitable position to fit it(*in this particular situation*) is3pixelsaway in the left direction.

So we havetoapplythe3timeslefttransformationbyonepixelatatime

For the left transformation ,we have to divide by 2^{17}

So we will move the structure left and downward in first 3 steps

We know

$k°=f_1×k+f_2×k'+f_3×k''+f_4×k'''........+f_{15}×k^{14}$

Where

k=ks+kr

wewillmovethestructureleftanddownwardinfirst3frames

Which gives

$k'=kr+\{(ks/2)/2^{17}\}$

$k'''=kr+\{(ks/4)/2^{2\times17}\}$

$k'''=kr+\{(ks/8)/2^{3\times17}\}$

Then after that we just have to move the structure downward by one pixel at a time from the latest position i.e. k'''

$k''''=k'''/2$

$k'''''=k'''/4$

and so on

$k^{14}=k'''/2^{10}$

Which gives us

$k°=(f_1\times k)+f_2\times(kr+ks/2)/2^{17}+f_3\times(kr+ks/4)/2^{2\times17}+f_4\times(kr+ks/8)/2^{3\times17}+f_5\times(k'''/2)+f_6\times(k'''/4).......+f_{15}\times(k'''/2^{10})$

Which is equals to

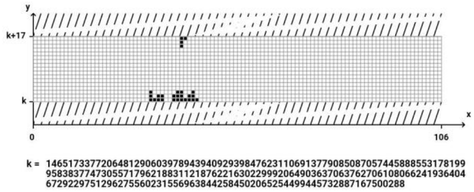
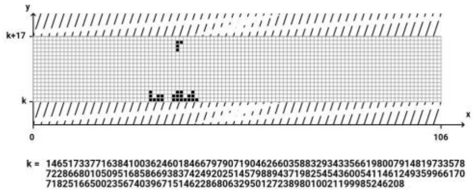
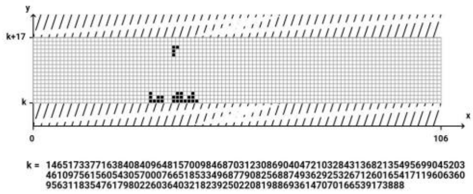
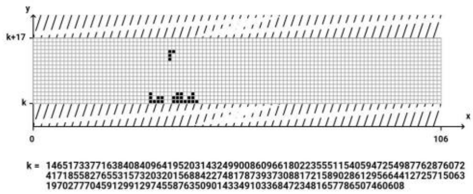
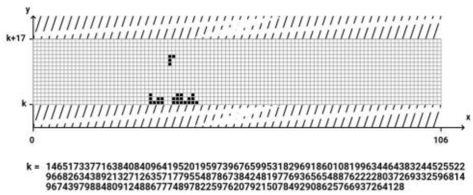
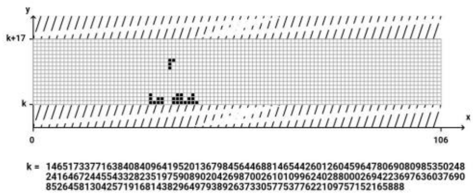
$k°=f_1\times\{k\}+f_2\times\{kr+(ks/2^{18})\}+f_3\times\{kr+(ks/2^{36})\}+f_4\times\{kr+(ks/2^{54})\}+f_5\times(kr+\{ks/2^{54}\}/2)+f_6\times(kr+\{ks/2^{54}\}/4).......+f_{15}\times(kr+\{ks/2^{54}\}/2^{10})$

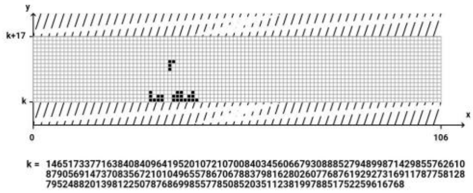
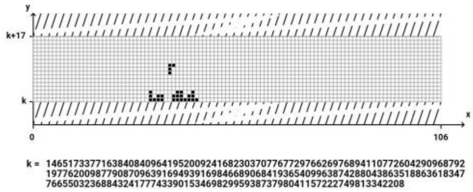
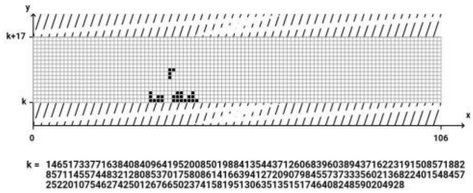
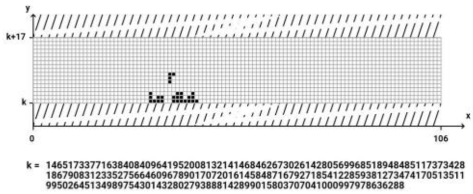
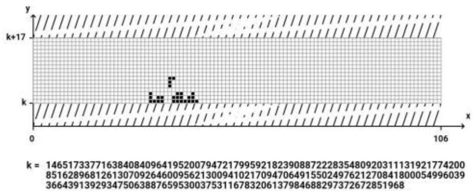
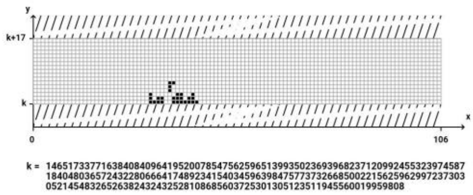
$k°=f_1\times\{k\}+f_2\times\{kr+(ks/2^{18})\}+f_3\times\{kr+(ks/2^{36})\}+f_4\times\{kr+(ks/2^{54})\}+f_5\times(kr+\{ks/2^{55}\})+f_6\times(kr+\{ks/2^{56}\}).......+f_{15}\times(kr+\{ks/2^{64}\})$

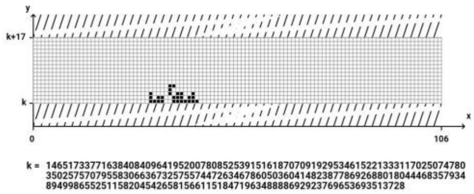
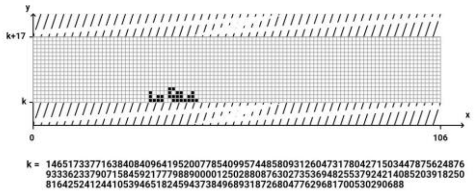
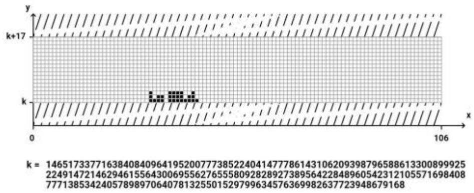
Following images shows the framing

Kr=1,465,173,377,163,840,840,964,195,200,776,229,452,338,097,479,160,165,100,993,321,673,778,726,174,973,516,467,105,018,733,852,068,234,022,255,298,658,975,702,245,055,915,087,238,316,041,012,593,947,856,673,785,182,723,705,258,476,099,103,213,618,090,667,396,472,262,335,593,747,367,067,648

Ks=42,640,449,639,783,743,164,699,946,138,133,627,753,613,984,093,735,772,109,827,003,226,441,916,669,711,823,327,553,649,089,932,323,504,047,296,961,434,448,619,137,954,569,794,030,245,783,793,507,114,789,257,497,126,216,466,424,824,493,834,668,782,237,110,139,139,800,432,640

<table>
<tr><td>Frame 1</td><td>
</td></tr>
<tr><td>Frame 2</td><td>
</td></tr>
<tr><td>Frame 3</td><td>
</td></tr>
<tr><td>Frame 4</td><td>
</td></tr>
<tr><td>Frame 5</td><td>
</td></tr>
<tr><td>Frame 6</td><td>
</td></tr>
</table>

<table>
<tr><td>Frame 7</td><td>
</td></tr>
<tr><td>Frame 8</td><td>
</td></tr>
<tr><td>Frame 9</td><td>
</td></tr>
<tr><td>Frame 10</td><td>
</td></tr>
<tr><td>Frame 11</td><td>
</td></tr>
<tr><td>Frame 12</td><td>
</td></tr>
</table>

Frame 13	
Frame 14	
Frame 15	

the video film on the link

https://kten.tk/Example2

Result:

The equation C

$k' = t_1\times k_1+t_2\times k_2+t_3\times k_3........+t_{1802}\times k_{1802} + \sum(17\times 2^{i-1}) - \sum(17\times 2^{j-1})$………(C)

Can be use to convert any graphical formation into any other graphical formation

and by equation D

$k° = f_1\times k+f_2\times k'+f_3\times k''+f_4\times k'''........f_n\times k^{n-1}$…………….(D)

This transformation in the graphical formation can be represent as a film/ Motion Picture

So the equations C and D together represent all the possible graphical formation and all its possible variation

Conclusion:

So the transformation on the pixels by changing the value of k is possible

And this variation of this graphical formation can be used as a frame to create a film / motion picture

bibliography:

1) Numberphile video on YouTube "the 'everything' formula numberphile"

2) Tupper's Self-Referential Formula Explained on https://www.petervis.com/mathematics/tuppers_self-referential_formula/tuppers_self-referential_formula.html

3) Tupper's Self-Referential Formula Margaret Fortman June 2, 2015

4) Graphs of Tupper's Self Referential are generated by http://tuppers-formula.tk