1. Pendahuluan

Di era modern sekarang ini, teknologi informasi menjadi bagian penting bagi kehidupan manusia. Dalam hal pengiriman data, dengan kemajuan teknologi informasi pengiriman data menjadi lebih mudah, lebih cepat dan efektif. Seiring dengan kemajuan teknologi saat ini yang semakin cepat, keamanan dalam hal pengiriman data menjadi sangat penting, karena banyak sekali kasus-kasus pencurian data dan penyadapan data saat ini. Untuk menghindari hal tersebut dibutuhkan suatu sistem keamanan yang mampu menjaga kerahasiaan suatu data, sehingga data yang dikirim tetap aman. Salah satu cara agar data tetap aman yaitu dengan membuat data menjadi tidak dapat dimengerti oleh sembarang orang kecuali untuk penerima yang berhak menerima data tersebut [1].

Teknik pengamanan data tersebut dikenal dengan nama kriptografi, sebagai suatu ilmu untuk mengamankan data [2]. Pada kriptografi terdapat dua komponen utama yaitu enkripsi dan dekripsi, enkripsi merupakan proses merubah data asli *(Plaintext)* menjadi data acak yang tidak dapat dimengerti *(Ciphertext)* sedangkan dekripsi adalah kebalikan dari enkripsi yaitu merubah *ciphertext* menjadi bentuk semula *plaintext* [3].

Salah satu solusi yang dapat dilakukan adalah memodifikasi kriptografi yang sudah dipecahkan atau menciptakan kriptografi yang baru sehingga dapat menjadi alternatif untuk pengamanan pesan [4]. Kriptografi yang digunakan dalam penelitian ini bersifat simetris dengan menggunakan satu kunci untuk proses enkripsi dan dekripsi, digunakan kriptografi simetris karena tidak membutuhkan proses komputasi yang rumit untuk proses enkripsi dan dekripsi [3].

Balap sepeda *velodrome* adalah kegiatan balap sepeda yang dilakukan di dalam sebuah bangunan yang disebut *velodrome*. Dalam melakukan penelitian ini dirancang kriptografi *Block Cipher* baru yang berbasis pola lapangan balap sepeda (*velodrome*). Pada pola ini pemasukan bit dilakukan secara horizontal. Kemudian pengambilan bit dilakukan berdasarkan pola lapangan balap sepeda (*velodrome*) tersebut yaitu, secara memutar menyerupai bentuk lapangan balap sepeda *(velodrome)*.

Pemasukan bit pada blok-blok berjumlah 64 bit dilakukan sebanyak 20 putaran dimana setiap putaran memiliki 4 proses *plaintext* dan juga proses kunci *(key)*. Hasil dari *plaintext* akan di-XOR dengan kunci untuk menghasilkan *Ciphertext* kemudian dikombinasikan dengan *S-Box* lalu dilakukan *transpose* untuk menghasilkan *Avalance Effect* yang besar. Dalam penelitian ini *S-Box* yang digunakan adalah *S-Box* algoritma *AES (Advanced Encryption Standard)* [5].

2. Tinjauan Pustaka

Terdapat banyak penelitian tentang kriptografi *block cipher.* Salah satunya yaitu berjudul "Perancangan Kriptografi *Block Cipher* Berbasis Pada Teknik Formasi Permainan Bola" membahas tentang perancangan kriptografi baru berbasis pada *block cipher* yang dikombinasikan dengan XOR [4].

Penelitian kedua berjudul "Pengaruh *S-Box Advanced Encryption Standard (AES)* Terhadap *Avalanche Effect* Pada Perancangan Kriptografi *Block Cipher* 256 Bit Berbasis Pola Teknik Tarian Dansa Tali Dari Maluku" yang membahas tentang kriptografi menggunakan prinsip *s-box* dan juga *avalanche effect* yang diproses sebanyak 5 putaran [5].

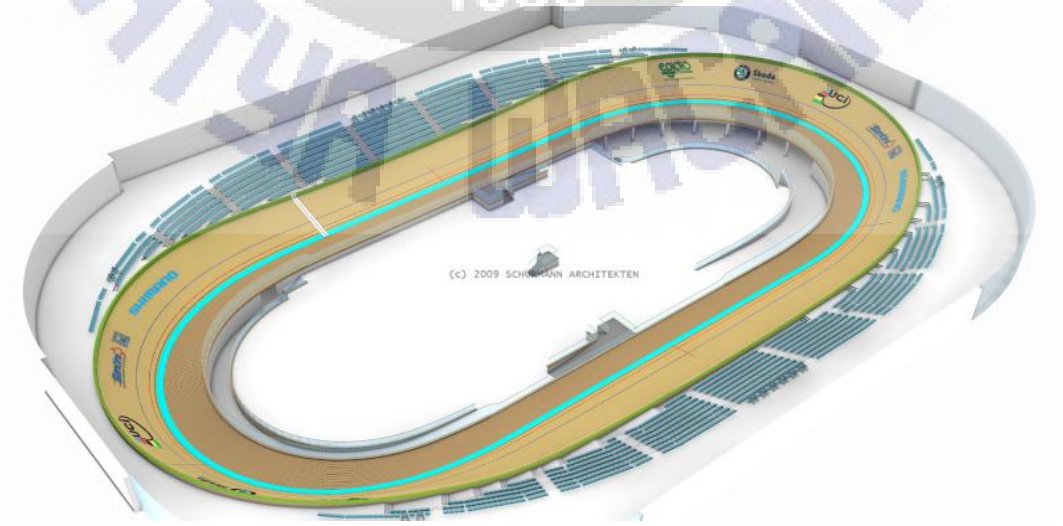
Penelitian ketiga dengan judul "Perancangan Kriptografi *Block Cipher* Berbasis Pola Ikan Berenang" membahas tentang perancangan kriptografi baru menggunakan prinsip *s-box, iterated cipher* dan jaringan *fiestel* yang dilakukan sebanyak 15 putaran dan diuji menggunakan *avalanche effect* dan nilai korelasi [6].

Kemudian penelitian keempat berjudul "Perancangan Kriptografi *Block Cipher* Berbasis Pola Bercocok Tanam Pada *Game Harvest Moon*" membahas tentang perancangan kriptografi baru menggunakan prinsip *s-box*, dan jaringan *fiestel* yang dilakukan sebanyak 20 putaran dan diuji menggunakan *avalanche effect* dan nilai korelasi [7].

Penelitian selanjutnya berjudul "Perancangan Kriptografi *Block Cipher* Berbasis pada Teknik Tanam Padi dan Bajak Sawah" membahas tentang implementasi algoritma *Block Cipher* untuk proses enkripsi dan dekripsi data yang dilakukan sebanyak 8 putaran [8].

Berdasarkan penelitian-penelitian yang terkait dengan algoritma *block cipher* tersebut, digunakan sebagai acuan dalam merancang penelitian tentang implementasi *block cipher* menggunakan pola lapangan balap sepeda *(velodrome)* yang dikombinasikan dengan *s*-box. Dalam penelitian ini dilakukan dengan proses enkripsi dan dekripsi sebanyak 20 putaran, kemudian diuji menggunakan nilai korelasi dari setiap putaran untuk mendapatkan hasil terbaik.

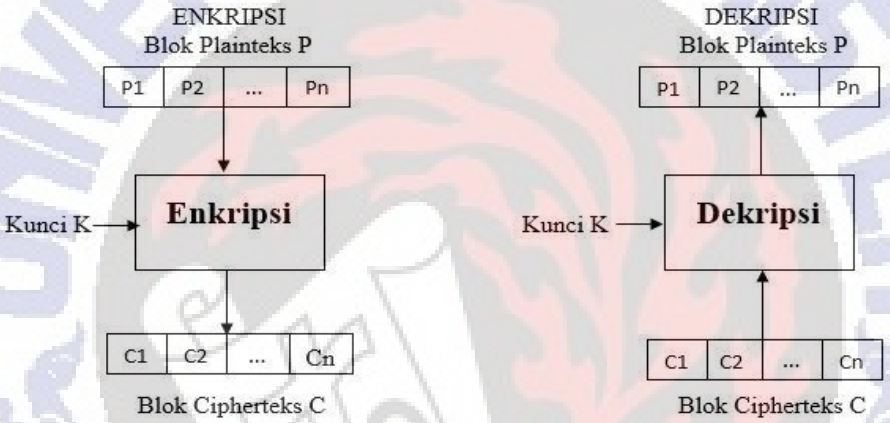
Balap sepeda trek di dalam arena yang disebut *velodrome* setidaknya sudah ada sejak tahun 1870. Dalam perkembangannya, lintasan yang digunakan awalnya terbuat dari bahan dengan material kayu yang berada pada bangunan yang disebut *velodrome*, yang terdiri dari dua lintasan lurus, serta lintasan berbentuk setengah lingkaran dengan kemiringan yang berbeda. Trek balap sepeda pada awal masa perkembangannya terdapat di beberapa kota besar di Inggris [9].



Gambar 1 Lapangan Balap Sepeda (*Velodrome)* [10]

Lintasan trek balap sepeda berbentuk lingkaran pada ujung mengikuti hukum fisika, pada intinya bentuk lapangan balap sepeda ini lebih mendekati oval bukan bulat sempurna, didesain untuk menemukan kriteria pertandingan sepeda dan semua ilmu disiplin yang berkaitan, berdasar pada prinsip-prinsip keamanan dan kenyamanan berolahraga. Desain parameter harus menjamin bentuk yang dapat menerima kecepatan yang cukup untuk pertandingan dan menyediakan tingkat keamanan yang tinggi untuk pengguna, dalam hal ini mencakup para atlit sepeda [9].

Block Cipher merupakan rangkaian bit yang dibagi menjadi blok-blok yang panjangnya sudah ditentukan sebelumnya. Skema proses enkripsi-deskripsi *block cipher* secara umum dapat ditunjukkan pada Gambar 2.



Gambar 2 Skema Proses Enkripsi dan Dekripsi *Block Cipher*

Misalkan blok *plaintext* (P) yang berukuran n bit

$$P = (p_1, p_2, \ldots, p_n) \quad (1)$$

Blok *ciphertext* (C) maka blok C adalah

$$C = (c_1, c_2, \ldots, c_n) \quad (2)$$

Kunci (K) maka kunci adalah

$$K = (k_1, k_2, \ldots, k_n) \quad (3)$$

Sehingga proses Enkripsi adalah

$$E_k(P) = C \quad (4)$$

Proses dekripsi adalah

$$D_k(C) = P\,(\mathrm{C}) = \mathrm{P} \quad (5)$$

Sebuah sistem kriptografi terdiri dari 5-*tuple* (*Five Tuple*) (**P,C,K,E,D**) yang memenuhi kondisi [8] :

1. **P** adalah himpunan berhingga dari *plaintext*.
2. **C** adalah himpunan berhingga dari *ciphertext*.

3. **K** merupakan ruang kunci (*Key space*), himpunan berhingga dari kunci.
4. Untuk setiap $k \in \boldsymbol{K}$, terdapat aturan enkripsi $e_k \in \boldsymbol{E}$ dan berkorespodensi dengan aturan dekripsi $d_k \in \boldsymbol{D}$. Setiap $ek : P \rightarrow C$ dan $d_k : C \rightarrow P$ adalah fungsi sedemikian hingga $d_k(e_k(x)) = x$ untuk setiap *plaintext* $x \in \boldsymbol{P}$.

Dalam pengujian menggunakan korelasi yang merupakan teknik statistik untuk mengukur kekuatan hubungan antar dua variabel dan untuk mengetahui bentuk hubungan antara dua variabel tersebut dengan hasil yang bersifat kuantitatif. Kekuatan hubungan antar dua variabel itu disebut dengan koefisien korelasi. Nilai koefisien akan selalu berada di antara -1 sampai +1. Untuk menentukan kuat atau lemahnya hubungan antara variabel yang diuji, dapat digunakan Tabel 1 [6].

Tabel 1 Klasifikasi Koefisien Korelasi

Interval Koefisien	Tingkat Hubungan
0,00 – 0,199	Sangat Rendah
0,20 – 0,399	Rendah
0,40 – 0,599	Sedang
0,60 – 0,799	Kuat
0,80 – 1,000	Sangat Kuat

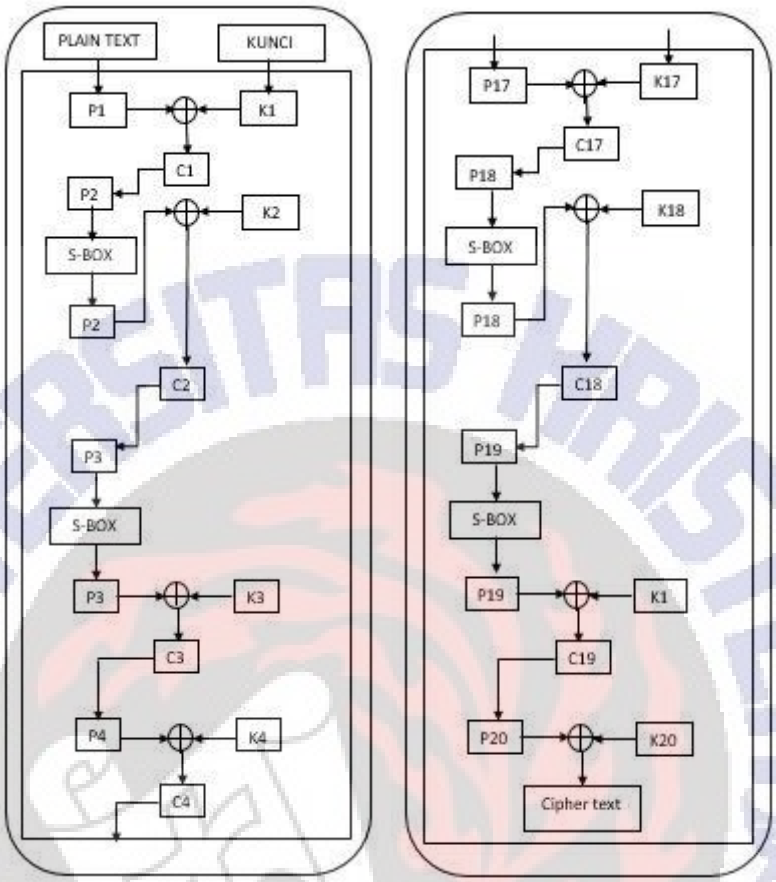
3. Metode Penelitian

Perancangan kriptografi ini dilakukan dengan beberapa tahap penyusunan yaitu : pengumpulan data, analisis kebutuhan, perancangan kriptografi, uji kriptografi, penulisan artikel ilmiah.

Gambar 3 Tahapan Penelitian

Berdasarkan pada Gambar 3, tahap penelitian dijelaskan sebagai berikut, tahap pertama adalah pengumpulan data, pada tahap ini dipersiapkan segala data yang dibutuhkan untuk merancang kriptografi menggunakan pola lapangan balap sepeda *(velodrome*). Data yang dimaksud seperti referensi jurnal-jurnal, buku-buku yang akan menjadi sumber untuk merancang kriptografi berbasis pola lapangan balap sepeda (*velodrome*). Tahap kedua adalah Analisi Kebutuhan. Pada tahap kedua ini dilakukan analisis kebutuhan untuk merancang kriptografi berbasis pola lapangan balap sepeda (*velodrome*). Kebutuhan yang dibutuhkan seperti laptop atau pc, aplikasi untuk merancang program kriptografi lapangan balap sepeda (*velodrome*), serta data yang disebutkan pada tahap pertama. Tahap ketiga, perancangan Kriptografi. Pada tahap ketiga ini adalah merancang kriptografi dengan pola lapangan balap sepeda (*velodrome*). Mengubah *plaintext* menjadi ASCII lalu mengubahnya menjadi bilangan biner, yang nantinya bilangan biner tersebut yang akan dilakukan proses enkripsi dan dekripsi, yang diterapkan ke dalam *block cipher* 64 bit. Tahap keempat, uji kriptografi. Tahap keempat ini adalah melakukan pengujian perhitungan *matematis*, mulai dari memasukkan *plaintext*, mengubah *text* ke dalam bit, melakukan proses enkripsi – dekripsi, mencari *avalanche effect* terbesar. Tahap kelima, penulisan artikel ilmiah. Tahap ini adalah tahap terakhir dalam penelitian ini, laporan ini sangat penting karena laporan ini bertujuan untuk menuliskan apa saja yang dibutuhkan, dilakukan, dan terjadi selama proses penelitian merancang kriptografi berbasis pola lapangan balap sepeda (*velodrome*).

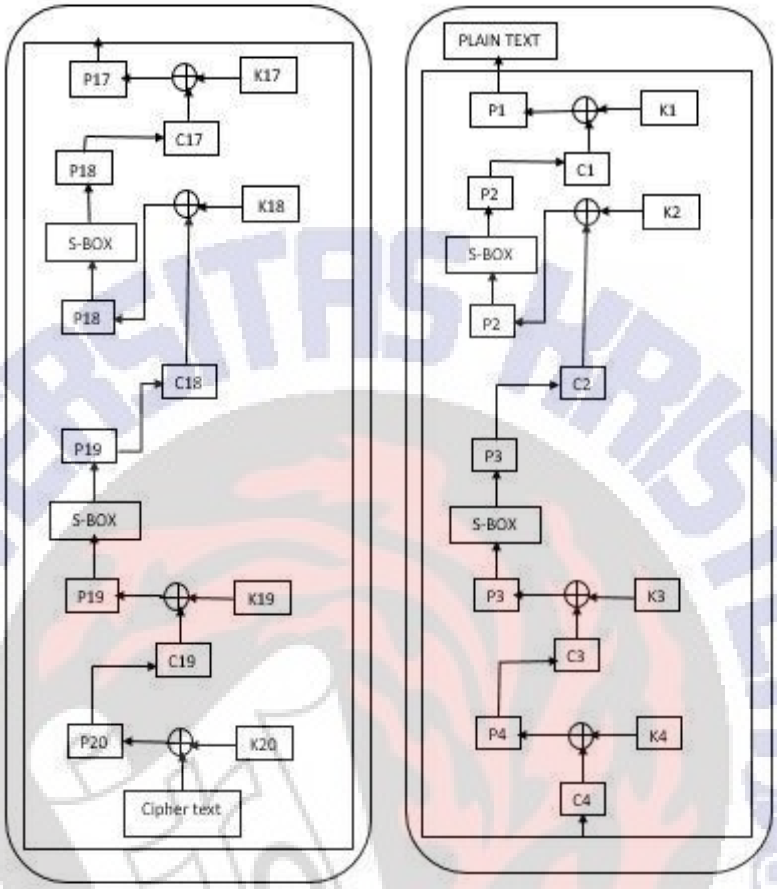
Dalam perancangan kriptografi *block cipher* 64 bit pada pola lapangan balap sepeda (*velodrome*) ini dilakukan dua proses yaitu proses enkripsi dan proses dekripsi. Enkripsi dan dekripsi itu sendiri dilakukan dalam 20 putaran, dan di setiap putaran terdapat 4 proses.



Gambar 4 Proses Alur Enkripsi

Gambar 4 menunjukkan alur proses enkripsi, langkah-langkah proses enkripsi itu sendiri dijelaskan sebagai berikut :

a) Menyiapkan *plaintext* dan kunci.
b) Mengubah *plaintext* dan kunci menjadi biner sesuai dalam tabel ASCII.
c) Dalam perancangan enkripsi, *plaintext* dan kunci akan melewati empat proses pada setiap putaran.
d) Putaran pertama Plaintext 1 (P1) diproses dengan pola dan di-XOR dengan Kunci 1 (K1) menghasilkan C1.
e) C1 ditransformasikan dengan pola menjadi P2 kemudian di S-BOX, setelah itu P2 di-XOR dengan K2 menghasilkan C2
f) C2 ditransformasikan dengan pola menjadi P3 kemudian di S-BOX, setelah itu P3 di-XOR dengan K3 menghasilkan C3.
g) C3 ditransformasikan dengan pola menjadi P4 dan di-XOR dengan K4 menghasilkan C4.
h) Masuk pada putaran dua, C4 ditransformasikan menjadi P5 kemudian dilakukan proses yang sama dengan putaran pertama, dan dilakukan sampai putaran ke-20 hingga menghasilkan *Ciphertext* (C).



Gambar 5 Proses Alur Dekripsi

Gambar 5 menunjukkan alur proses dekripsi, langkah-langkah proses dekripsi tersebut dijelaskan sebagai berikut :

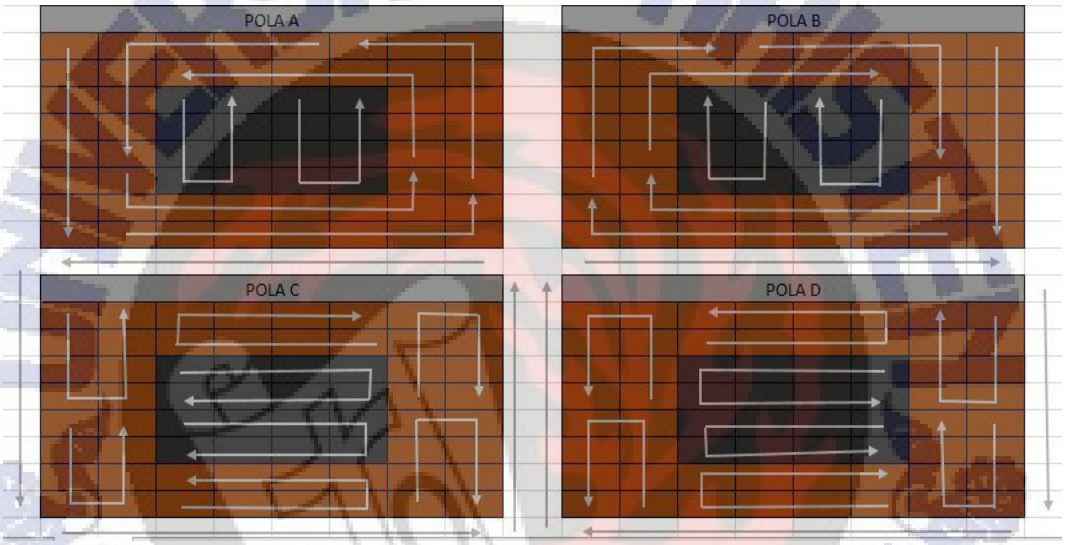
a) Menyiapkan *ciphertext* dan kunci.
b) Mengubah *ciphertext* dan kunci menjadi biner sesuai dalam tabel ASCII.
c) dalam perancangan dekripsi, *ciphertext* dan kunci akan melewati empat proses pada setiap putaran.
d) Putaran pertama *Ciphertext* (C) diproses dengan pola dan di-XOR dengan Kunci 20 (K20) menghasilkan P20.
e) P20 ditransformasikan dengan pola menjadi C19 kemudian di-XOR dengan K19 dan diproses menggunakan *S-BOX*, dan mendapatkan hasil akhir P19.
f) P19 ditransformasikan dengan pola menjadi C18 kemudian di-XOR dengan K18, menghasilkan P18 kemudian diproses menggunakan *S-BOX*, dan mendapatkan hasil akhir P18.
g) P18 ditransformasikan dengan pola menjadi C17 kemudian di-XOR dengan K17, menghasilkan P17.

h) Masuk pada putaran dua, P17 ditransformasikan menjadi C16 kemudian dilakukan proses yang sama dengan putaran pertama, dan dilakukan sampai putaran ke-20 hingga menghasilkan *Plaintext* (P).

4. Hasil dan Pembahasan

Pada bagian ini akan membahas tentang algoritma perancangan kriptografi *block cipher* 64 bit berbasis pola lapangan balap sepeda (*velodrome*) secara lebih rinci.

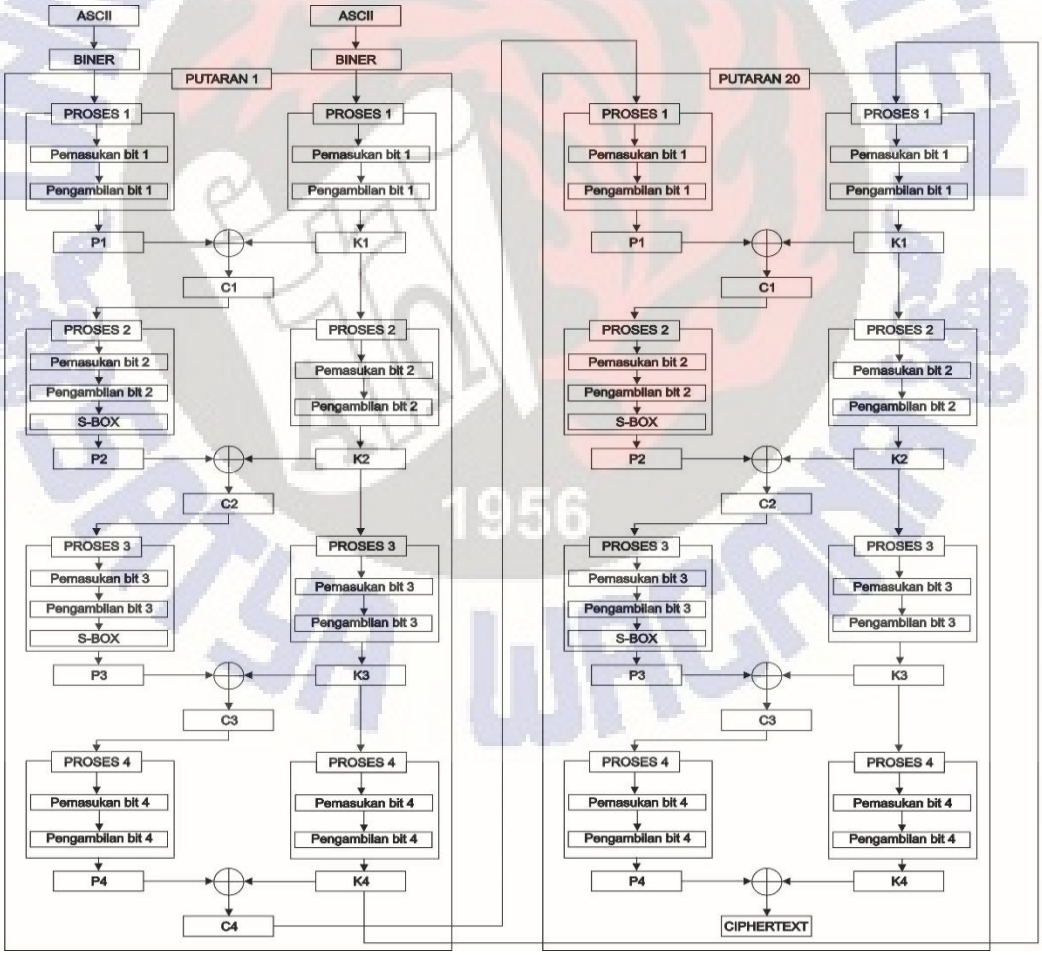
Dalam algoritma ini pola lapangan balap sepeda (*velodrome*) digunakan sebagai proses pemasukan dan pengambilan bit. Pola tersebut ditunjukkan pada Gambar 6.



Gambar 6 Pola Lapangan Balap Sepeda *(Velodrome)*

Pada Gambar 6, pola A dan B menjelaskan ketika seorang atlet balap sepeda sedang melakukan balap sepeda di lapangan balap sepeda (*velodrome*) dilakukan secara berputar memutari lapangan tersebut untuk mendapatkan catatan waktu tercepat, karena dari itu pola A dan B dibuat memutar seperti pada gambar, sedangkan kotak di tengah tersebut merupakan sisa bagian dari lapangan yang tidak digunakan untuk balap sepeda, yang biasanya digunakan untuk alih fungsi lainnya, sehingga pola yang di tengah dibuat sedikit berbeda. Sedangkan pola C dan D dalam melakukan pengacakan pemutaran binernya dilakukan pada setiap 8 kotak seperti pada gambar, pola C di mulai dari kiri atas lalu ke bawa setelah kotak ke 4 pindah ke kotak sebelah kanannya dan menuju ke arah atas begitu pula dengan pola D berjalan seperti Pola C hanya dimulai dari kanan atas kebalikan dari pola C, sedangkan maksud dari panah di luar kotak tersebut adalah untuk menunjukkan arah putaran setiap 8 bitnya, setelah pemasukan 8 bit pertama, untuk memasukkan 8 bit kedua dan seterus nya akan mengikuti arah panah yang ada di luar kotak tersebut. Berdasarkan pola-pola yang sudah dirancang, dilakukan pengujian korelasi dengan mengkombinasikan urutan pola untuk menemukan nilai korelasi terbaik. Pengujian dilakukan menggunakan contoh *plaintext* “gbipUKSW” menggunakan kunci “mAiP2472”.

Dalam melakukan perancangan kriptografi *block cipher* berbasis pola lapangan balap sepeda *(velodrome)* ini dilakukan sebanyak 20 putaran, dan di setiap putaran memiliki 4 proses untuk mendapatkan hasil akhir yaitu *ciphertext.* Proses pertama *plaintext* dan kunci diubah ke dalam bentuk ASCII kemudah diubah lagi ke dalam biner. Kemudian bit-bit *plaintext* diproses dengan pola pemasukan dan pengambilan ke dalam kolom matriks 8x8 menggunakan pola lapangan balap sepeda *(velodrome)* yang berbeda-beda pada setiap proses. Setelah mendapatkan hasil XOR dari proses pertama lalu diproses dengan pola pengambilan, kemudian ditransformasikan menggunakan S-BOX pada proses kedua. Hasil XOR dari proses kedua akan diproses dengan pola pengambilan, kemudian ditransformasikan lagi dengan S-BOX pada proses ketiga, kemudian akan dimasukkan lagi pada blok matriks dengan pola pemasukan sehingga akan menghasilkan XOR dari proses ketiga yang kemudian akan digunakan pada proses keempat dan diulang terus-menerus hingga putaran ke-20 untuk menghasilkan *ciphertext.*



Gambar 7 Proses Enkripsi

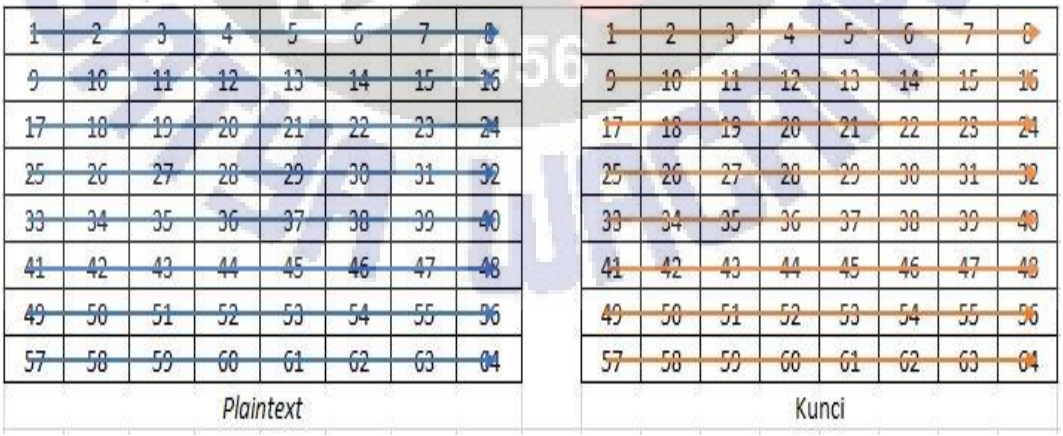
Berdasarkan hasil pengujian korelasi, maka hasil terbaiklah yang akan digunakan sebagai acuan perancangan dalam proses enkripsi dan dekripsi.

Tabel 2 Tabel Rata Korelasi

RATA-RATA NILAI KORELASI			
POLA	**RATA-RATA**	**POLA**	**RATA-RATA**
A-B-C-D	0,763597484	**C-A-B-D**	-0,478710731
A-B-D-C	0,306723103	**C-A-D-B**	-0,548017918
A-C-B-D	-0,609164992	**C-B-A-D**	-0,191640745
A-C-D-B	0,039529351	**C-B-D-A**	0,348277313
A-D-B-C	0,034455493	**C-D-A-B**	0,183663961
A-D-C-B	-0,576722376	**C-D-B-A**	0,461418785
B-A-C-D	0,621482097	**D-A-B-C**	0,169129293
B-A-D-C	0,232476754	**D-A-C-B**	0,215012307
B-C-A-D	0,125809618	**D-B-A-C**	0,078789357
B-C-D-A	-0,477513757	**D-B-C-A**	0,07264371
B-D-A-C	-0,047145594	**D-C-A-B**	-0,037481101
B-D-C-A	0,401694122	**D-C-B-A**	-0,29109936

Tabel 2 menunjukkan hasil kombinasi pola dan mendapatkan nilai korelasi terbaik pada kombinasi pola A-D-B-C. Kombinasi A-D-B-C yang akan digunakan untuk melanjutkan proses enkripsi hingga putaran ke-20 untuk menghasilkan *ciphertext*.

Pada bagian ini menjelaskan secara detail proses pemasukan bit dalam matriks maka diambil proses 1 pada putaran 1 sebagai contoh. Misalkan angka 1 merupakan inisialisasi setiap bit yang merupakan hasil konversi *plaintext* maka urutan bit adalah sebagai berikut 1, 2, 3, 4, …..64.



Gambar 8 Pola Pemasukan Awal *Plaintext* dan Kunci *(key)*

Gambar 8 menjelaskan pola pemasukan bit *plaintext* dan kunci. Setiap 8 bit dari setiap karakter *plaintext* dan kunci dimasukkan pada setiap blok yang dimulai dari

bagian kiri terlebih dahulu kemudian ke kanan mengikuti garis anak panah dan juga sesuai dengan urutan angka seperti pada Gambar 8.

1	28	27	26	25	24	23	22
2	29	48	47	46	45	44	21
3	30	49	56	57	64	43	20
4	31	50	55	58	63	42	19
5	32	51	54	59	62	41	18
6	33	52	53	60	61	40	17
7	34	35	36	37	38	39	16
8	9	10	11	12	13	14	15

Pengambilan

1	2	3	4	5	6	7	8
16	15	14	13	12	11	10	9
17	18	19	20	21	22	23	24
32	31	30	29	28	27	26	25
33	34	35	36	37	38	39	40
48	47	46	45	44	43	42	41
49	50	51	52	53	54	55	56
64	63	62	61	60	59	58	57

Pemasukan

Gambar 9 Pola Pengambilan dan Pola Pemasukan *Plaintext* dan Kunci Proses 1

Gambar 9 merupakan pola pengambilan dan pemasukan bit pada proses 1, proses pengambilan bitnya sesuai dengan urutan angka tersebut. Setelah bit diambil menggunakan pola, kemudian hasil pengambilan dimasukkan ke dalam ke dalam blok matriks dengan menggunakan pola pemasukan proses 1 dan akan menghasilkan P1 lalu nanti P1 dan K1 akan di-XOR untuk menghasilkan C1. Begitu juga dengan Pola kunci setelah dilakukan pola pengambilan K1 lalu dimasukan ke dalam blok matriks dengan menggunakan pola pemasukan proses 1 agar menghasilkan hasil akhir K1 lalu K1 akan di-XOR dengan P1 untuk menghasilkan C1.

37	36	48	47	46	45	8	1
38	35	41	42	43	44	7	2
39	34	52	51	50	49	6	3
40	33	53	54	56	56	5	4
29	28	60	59	58	57	16	9
30	27	61	62	63	64	15	10
31	26	21	22	23	24	14	11
32	25	20	19	18	17	13	12

Pengambilan

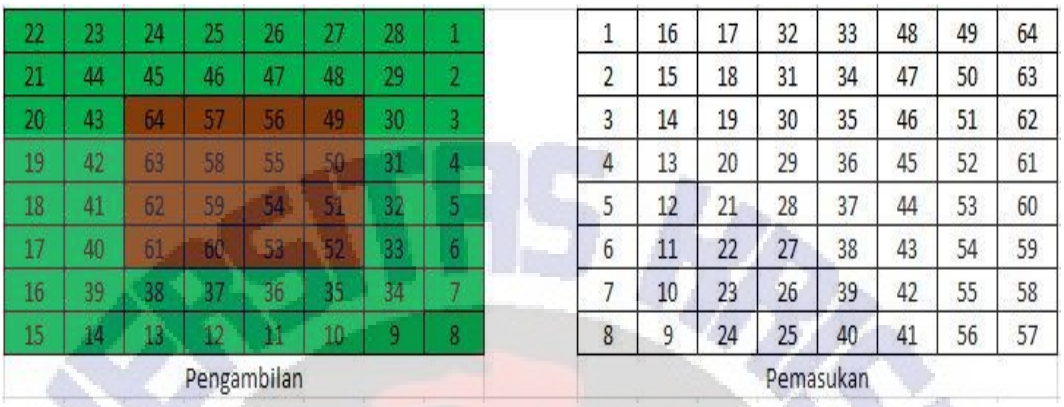
8	9	24	25	40	41	56	57
7	10	23	26	39	42	55	58
6	11	22	27	38	43	54	59
5	12	21	28	37	44	53	60
4	13	20	29	36	45	52	61
3	14	19	30	35	46	51	62
2	15	18	31	34	47	50	63
1	16	17	32	33	48	49	64

Pemasukan

Gambar 10 Pola Pengambilan dan Pola Pemasukan *Plaintext* dan Kunci Proses 2

Gambar 10 merupakan pola pengambilan dan pemasukan bit *plaintext* dan kunci proses 2, dimana C1 pada proses 1 digunakan sebagai P2 dan K1 digunakan sebagai K2. Kemudian P2 diproses menggunakan S-BOX lalu berikut dimasukkan ke dalam blok matriks dengan menggunakan pola pemasukan proses 2, setelah itu hasil akhir dari P2 di-XOR dengan K2 untuk menghasilkan C2. Untuk proses Kunci sendiri tidak dilakukan S-BOX jadi setelah pola pengambilan bit dimasukkan ke dalam blok matriks dengan menggunakan pola pemasukan proses 2 untuk

menghasilkan K2 lalu K2 akan dilakukan XOR dengan *plaintext* agar menghasilkan C2.



22	23	24	25	26	27	28	1
21	44	45	46	47	48	29	2
20	43	64	57	56	49	30	3
19	42	63	58	55	50	31	4
18	41	62	59	54	51	32	5
17	40	61	60	53	52	33	6
16	39	38	37	36	35	34	7
15	14	13	12	11	10	9	8

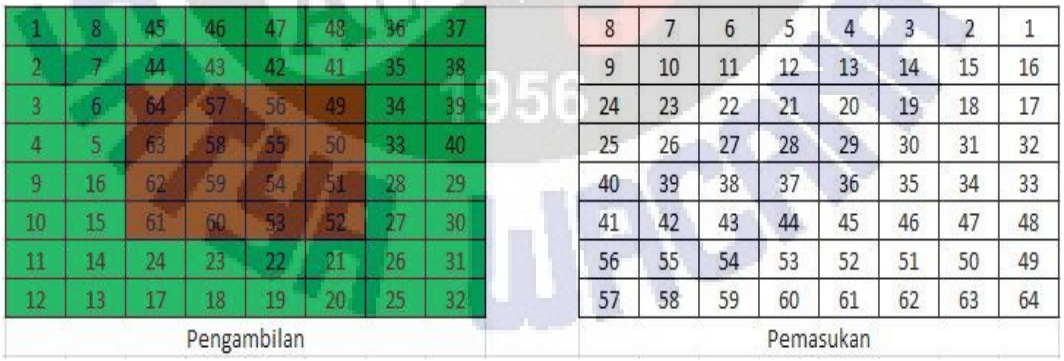
Pengambilan

1	16	17	32	33	48	49	64
2	15	18	31	34	47	50	63
3	14	19	30	35	46	51	62
4	13	20	29	36	45	52	61
5	12	21	28	37	44	53	60
6	11	22	27	38	43	54	59
7	10	23	26	39	42	55	58
8	9	24	25	40	41	56	57

Pemasukan

Gambar 11 Pola Pengambilan Pola Pemasukan *Plaintext* dan Kunci Proses 3

Gambar 11 merupakan pola pengambilan dan pemasukan bit *plaintext* dan kunci proses 3, dimana C2 pada proses 2 digunakan sebagai P3 dan K2 digunakan sebagai K3. Kemudian P3 diproses menggunakan S-BOX lalu berikut akan dimasukkan ke dalam blok matriks dengan menggunakan pola pemasukan proses 3, hasil akhir dari P3 di-XOR dengan K3 untuk menghasilkan C3. Untuk proses kunci sendiri tidak dilakukan S-BOX jadi setelah pola pengambilan akan dimasukkan ke dalam blok matriks dengan menggunakan pola pemasukan proses 3, lalu setelah itu akan dilakukan XOR dengan *plaintext* agar menghasilkan C3. Proses pengambilan P3 dan K3 dilakukan sesuai seperti nomer yang tertera pada Gambar 11.



1	8	45	46	47	48	36	37
2	7	44	43	42	41	35	38
3	6	64	57	56	49	34	39
4	5	63	58	55	50	33	40
9	16	62	59	54	51	28	29
10	15	61	60	53	52	27	30
11	14	24	23	22	21	26	31
12	13	17	18	19	20	25	32

Pengambilan

8	7	6	5	4	3	2	1
9	10	11	12	13	14	15	16
24	23	22	21	20	19	18	17
25	26	27	28	29	30	31	32
40	39	38	37	36	35	34	33
41	42	43	44	45	46	47	48
56	55	54	53	52	51	50	49
57	58	59	60	61	62	63	64

Pemasukan

Gambar 12 Pola Pengambilan dan Pola Pemasukan *Plaintext* dan Kunci Proses 4

Gambar 12 menjelaskan tentang tahap akhir pola pengambilan dan pola pemasukan *plaintext* dan kunci. P4 dan K4 diambil menggunakan pola sesuai dengan urutan nomor yang tertera, hasil pengambilan P4 kemudian dimasukkan kembali ke dalam blok matriks dengan menggunakan pola pemasukan proses 4 begitu juga dengan K4 untuk menghasilkan *plaintext* dan kunci akhir. Hasil akhir dari P4 dan K4 kemudian di-XOR agar menghasilkan C4. Proses enkripsi putaran

1 telah selesai, dan dilakukan proses yang sama secara terus-menerus hingga putaran ke-20 untuk mendapatkan *ciphertext.*

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FFD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	6	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

Gambar 13 Tabel S-BOX AES *(Advanced Encryption Standard)*

Gambar 13 menunjukkan tabel *S-BOX* yang digunakan pada proses enkripsi dan dekripsi. Cara pensubstitusiannya adalah sebagai berikut: untuk setiap *byte* pada *array state*, misalkan *S*[*r*, *c*] = *xy*, yang dalam hal ini *x*,*y* adalah digit heksadesimal dari nilai *S*[*r*, *c*], maka nilai substitusinya, dinyatakan dengan *S*'[*r*, *c*], adalah elemen di dalam *S-box* yang merupakan perpotongan baris *x* dengan kolom *y*. Misalnya *S*[0, 0] = 47, maka S'[0, 0] = A0.

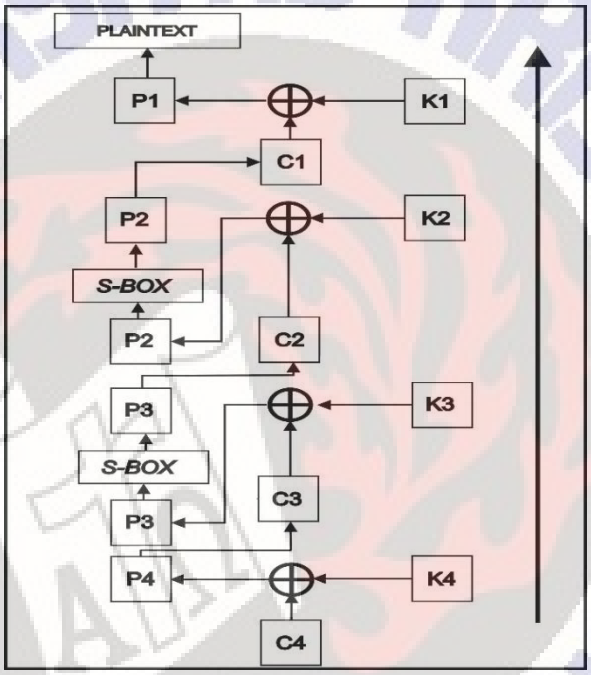
Untuk pengujian algoritma dilakukan dengan mengambil contoh *plaintext* gbipUKSW dan kunci mAiP2472. Kemudian dilakukan proses enkripsi sebanyak 20 putaran, dan di setiap putaran enkripsi akan mendapatkan *ciphertext* (C) dan dikonversi ke dalam nilai *hexadecimal.* Hasil enkripsi dari putaran ke 20 adalah *final ciphertext* ditunjukkan pada Tabel 3.

Tabel 3 Hasil *Ciphertext* Setiap Putaran

Putaran	Hasil *Hexadecimal*	Putaran	Hasil *Hexadecimal*
1	18C888F991AB98E	11	FBDDE63C967F74A
2	842F908CA605739	12	C939D83328BEDA77
3	1F3FECE61B239374	13	9BD94D52BD9513F0
4	B4C9F3AA7B83FAFE	14	DABF46FC9E7E7E8
5	9A5214A8AC2782AC	15	65B9CBEAFB2330D0
6	B2E32F66DF813EE0	16	954131373B53CE4D
7	542979D57DDD19C5	17	F48B4D2530B841A6
8	98BFA8ABC15D8411	18	2B113A6D1C9ECD27
9	BF5E2A579B6DB96	19	8F4DBF3F377CB

10	34E1913357BF013	20	64898CB3ECE77857

Proses dekripsi merupakan proses merubah *ciphertext* menjadi *plaintext* awal. Dekripsi dilakukan sama seperti enkripsi, tetapi dekripsi dimulai dari putaran ke-20 menuju putaran ke-1 untuk mendapatkan *plaintext* awal. Pada proses dekripsi ini alurnya dibalik untuk mendapatkan kembali hasil *plaintext* yang sudah diubah dalam bentuk *ciphertext* tadi. Pola pengambilan pada proses enkripsi akan menjadi pola pemasukan pada proses dekripsi, sedangkan pola pemasukan pada proses enkripsi akan mejadi pola pengambilan pada proses dekripsi.



Gambar 14 Alur Proses Dekripsi

Gambar 14 menjelaskan alur dekripsi. Pola pengambilan pada proses enkripsi akan menjadi pola pemasukan pada proses dekripsi, sedangkan pola pemasukan pada enkripsi akan digunakan sebagai pola pengambilan pada dekripsi. Proses dekripsi dimulai dari memasukkan *ciphertext* ke kolom matrik C4 kemudian di-XOR dengan K4 pada proses keempat menghasilkan P4. Kemudian P4 digunakan sebagai C3, lalu C3 di-XOR dengan K3 dan menghasilkan P3, P3 ditransformasikan dengan S-BOX untuk mendapatkan hasil akhir P3, P3 digunakan sebagai C2 pada proses berikutnya. Setelah itu, C2 di-XOR dengan K2 untuk menghasilkan P2. P2 ditransformasikan dengan S-BOX untuk mendapatkan hasil akhir P2, lalu hasil dari P2 digunakan sebagai C1 pada proses selanjutnya. C1 kemudian di-XOR dengan K1 untuk menghasilkan P1, proses itu dilakukan berulang-ulang sebanyak 20 putaran sesuai dengan banyaknya putaran enkripsi dan hasil akhir dari dekripsi putaran ke-20 adalah *plaintext* awal.

Tabel 4 Algoritma Enkripsi dan Dekripsi

Proses Enkripsi	Proses Dekripsi
1. Masukkan *plaintext*	1. Masukkan *ciphertext*
2. *Plaintext* diubah ke ASCII	2. *Ciphertext* diubah ke ASCII
3. ASCII diubah ke BINER	3. ASCII diubah ke BINER
4. Bit BINER dimasukkan ke kolom matrik P1 dengan pola pemasukan proses 1	4. Bit BINER dimasukkan ke kolom matrik C4 dengan pola pemasukan awal
5. Bit pada kolom matrik diambil menggunakan pola pengambilan.	5. C4 di-XOR dengan K4 menghasilkan P4
6. Bit pengambilan dimasukkan lagi ke dalam matrik untuk mendapatkan hasil akhir P1	6. P4 diproses dengan pola pengambilan proses 4
7. P1 di XOR dengan K1 menghasilkan C1	7. P4 = C3
8. C1 = P2 diambil menggunakan pola pemasukan awal	8. C3 di XOR dengan K3 menghasilkan P3
9. P2 diproses dengan pola pengambilan proses 2	9. P3 diproses dengan pola pengambilan proses 3
10. BINER diubah menjadi DEC	10. Hasil pengambilan diubah ke bentuk HEX
11. DEC diubah menjadi HEX	11. Hasil HEX ditransformasikan menggunakan S-BOX
12. HEX ditransformasikan menggunakan S-BOX	12. Hasil konversi diubah menjadi BINER
13. Hasil HEX diubah menjadi BINER	13. BINER dimasukkan ke blok matriks P3
14. Hasil biner dimasukkan ke blok matriks P2	14. P3 = C2
15. P2 di XOR dengan K2 menghasilkan C2	15. C2 di XOR dengan K2 menghasilkan P2
16. C2=P3 diambil dengan pola pemasukan awal	16. P2 diproses dengan pola pengambilan proses 2
17. P3 diproses dengan pola pengambilan proses 3	17. Hasil pengambilan diubah ke bentuk HEX
18. BINER diubah menjadi DEC	18. Hasil HEX ditransformasikan dengan S-BOX
19. DEC diubah menjadi HEX	19. Hasil konversi diubah menjadi BINER
20. HEX ditransformasikan menggunakan S-BOX	20. BINER dimasukkan ke blok matriks P2
21. Hasil HEX diubah menjadi BINER	21. Hasil akhir P2 = C1
22. Hasil BINER dimasukkan ke blok matriks P3	22. C1 di XOR dengan K1 menghasilkan P1
23. P3 di XOR dengan K3 menghasilkan C3	23. P1 diproses dengan pola pengambilan proses 1
24. C3=P4 diambil dengan pola pemasukan awal	24. Bit pengambilan dimasukkan lagi ke dalam kolom matrik P1
25. P4 diproses dengan pola pengambilan proses 4	25. Hasil akhir pemasukan bit diproses dengan pola pengambilan
26. P4 di XOR dengan K4 menghasilkan C4	26. Hasil akhir BINER P1 diubah ke DEC
27. C4 diubah ke DEC	27. DEC diubah ke HEX
28. DEC diubah ke HEX	28. HEX diubah ke CHAR

Tabel 4 merupakan algoritma proses enkripsi dan dekripsi secara menyeluruh. Proses enkripsi menghasilkan C4 (*Cipertext*), dan proses dekripsi menghasilkan P1 *(Plaintext)* awal.

Algortma proses Kunci (*key*), dijelaskan sebagai berkut:

1. Masukkan Kunci.
2. Kunci diubah ke dalam ASCII.

3. ASCII diubah menjadi BINER.
4. Bit BINER dimasukkan ke blok K1 dengan pola pemasukan Kunci.
5. Bit kunci diambil dengan pola pengambilan Kunci.
6. BINER hasil pengambilan dimasukkan ke dalam kolom matrik K1.
7. K1 = K2.
8. K2 dimasukkan ke dalam blok matriks K2 dengan menggunakan pola pemasukan awal.
9. Bit K2 pada blok matriks diambil menggunakan pola pengambilan proses 2.
10. BINER hasil pengambilan dimasukkan ke dalam blok matriks K2 dengan pola pemasukan proses 2.
11. K2 = K3.
12. K3 dimasukkan ke blok matriks dengan pola pemasukan awal.
13. K3 pada blok matriks diambil menggunakan pola pengambilan proses 3.
14. BINER hasil pengambilan dimasukkan ke dalam blok matriks K3.
15. K3 = K4
16. K4 dimasukan ke blok matriks K4 dengan menggunakan pola pemasukan proses 4.
17. K4 pada blok matriks diambil dengan pola pengambilan proses 4.
18. BINER hasil pengambilan dimasukkan ke dalam blok matriks K4.

Pada bagian ini menjelaskan proses enkripsi dan dekripsi salah satu huruf *plaintext* yaitu huruf "g" secara rinci. Pada proses enkripsi huruf awal *plaintext* adalah "g". Setelah dilakukan proses enkripsi dan menghasilkan hasil akhir *ciphertext* yaitu "01". Pada proses dekripsi, hasil dari *ciphertext* diproses dengan proses dekripsi dan menghasilkan hasil akhir huruf "g". Langkah-langkah proses enkripsi dan dekripsi dapat dilihat pada Tabel 5.

Tabel 5 Contoh Enkripsi dan Dekripsi Huruf

	Contoh Proses Enkripsi		**Contoh Proses Dekripsi**
1.	*Plaintext* contoh huruf "g"	1.	*Ciphertext* huruf "g" yaitu "01"
2.	Huruf "g" diubah ke ASCII menjadi "103"	2.	*Ciphertext* "01" diubah ke DEC menjadi "103"
3.	ASCII diubah ke BINER menjadi "01100111"	3.	DEC diubah ke BINER menjadi "00000001"
4.	Bit BINER dimasukkan ke blok matriks P1 dengan pola pemasukan proses 1	4.	Bit BINER dimasukkan ke blok matriks C4 dengan pola pemasukan awal
5.	Bit pada blok matriks diambil menggunakan pola pengambilan	5.	C4 di-XOR dengan K4 menghasilkan P4
6.	Bit pengambilan dimasukkan lagi ke dalam blok matriks mendapatkan hasil akhir P1	6.	P4 diproses dengan pola pengambilan proses 4
7.	P1 di-XOR dengan K1 menghasilkan C1	7.	P4 = C3
8.	C1 = P2 diambil dengan pola pemasukan proses 2	8.	C3 di-XOR dengan K3 menghasilkan P3

9.	P2 diproses menggunakan pola pengambilan proses 2	9.	P3 diproses dengan pola pengambilan
10.	BINER diubah menjadi DEC	10.	Hasil pengambilan diubah ke bentuk HEX
11.	DEC diubah menjadi HEX	11.	Hasil HEX ditranformasikan menggunakan tabel *S-box*
12.	HEX ditranformasikan menggunakan *S-box*	12.	Hasil konversi diubah menjadi BINER
13.	Hasil HEX diubah menjadi BINER	13.	BINER dimasukkan ke blok matriks P3
14.	Hasil BINER dimasukkan ke blok matriks P2	14.	P3 = C2
15.	P2 di-XOR dengan K2 menghasilkan C2	15.	C2 di-XOR dengan K2 menghasilkan P2
16.	C2 = P3 diambil dengan pola pemasukan proses 3	16.	P2 diproses dengan pola pengambilan
17.	P3 diproses menggunakan pola pengambilan proses 3	17.	Hasil pengambilan diubah ke bentuk HEX
18.	BINER diubah menjadi DEC	18.	Hasil HEX ditranformasikan menggunakan tabel *S-box*
19.	DEC diubah menjadi HEX	19.	Hasil konversi diubah menjadi BINER
20.	HEX ditransformasikan menggunakan *S-box*	20.	BINER dimasukkan ke blok matriks P2
21.	Hasil HEX diubah menjadi BINER	21.	P2 = C1
22.	Hasil BINER dimasukkan ke blok matriks P3	22.	C1 di-XOR dengan K1 menghasilkan P1
23.	P3 di-XOR dengan K3 menghasilkan C3	23.	P1 diproses dengan pola pengambilan
24.	C3 = P4 diambil dengan pola pemasukan proses 4	24.	Bit pengambilan dimasukkan lagi ke dalam blok matriks P1
25.	P4 diproses menggunakan pola pengambilan proses 4	25.	Hasil akhir pemasukan bit diproses dengan pola pengambilan proses 1
26.	P4 di XOR dengan K4 menghasilkan C4	26.	Hasil akhir dekripsi BINER huruf "g" yaitu "01100111" pada blok matriks P1
27.	Hasil enkripsi bit biner dari C4 huruf "g" adalah "00000001" diubah ke DEC menjadi "1"	27.	Hasil BINER diubah ke DEC menjadi "103"
28.	DEC diubah ke HEX menjadi "01" yang merupakan hasil *ciphertext.*	28.	Hasil DEC "103" diubah ke CHAR menjadi "g"

Dalam bagian ini, menjelaskan tentang *Pseudocode* pada perancangan kriptografi *block cipher* berbasis pada pola lapangan balap sepeda *(velodrome).* Proses enkripsi dan proses dekripsi, dijelaskan sebagai berikut:

Proses Enkripsi
{Program ini digunakan untuk melakukan proses enkripsi data}

```
Kamus
P,K,P1,K1,P2,K2,P3,K3,P4,K4 = integer
C1,C2,C3,C4 = integer
Start
        C1 <- P1 ⊕ K1
```

Input P
Read P
P to ASCII
ASCII to BINER
Dari BINER = blok matriks P1, masukan BINER
P1 menggunakan Pola pemasukan awal
Dari blok matriks P1 = BINER, ambil bit
P1 dengan pola lapangan balap sepeda *(velodrome)* A
Dari BINER = blok matriks P1, masukan BINER
P1 dengan pola pemasukan proses 1
Output P1

Input K
Read K
K to ASCII
ASCII to BINER
Dari BINER = blok matriks K1, masukan BINER
K1 menggunakan Pola pemasukan awal
Dari blok matriks K1 = BINER, ambil bit K1
K1 dengan pola pengambilan A
Dari BINER = blok matriks K1, masukan BINER
K1 dengan pola pemasukan proses 1
Output K1
Print C1
C1 = P2
C2 <- P2 ⊕ K2
Dari C1 = blok matriks P2, masukan C1
P2 menggunakan Pola pemasukan awal
Dari blok matriks P2 = BINER, ambil bit
P2 dengan pola pengambilan lapangan balap sepeda *(velodrome)* D
BINER to HEXA
Dari HEXA = Tabel *S-box*, masukan HEXA
HEXA dikonversi menggunakan *S-box*
Print BINER *S-box*
Dari BINER = blok matriks P2, masukan BINER
P2 dengan pola pemasukan proses 2
Output P2
Dari K1 = blok matriks K2, masukan K1
Dari blok matriks K2 = BINER, ambil bit
K2 dengan pola pengambilan D
Dari BINER = blok matriks K2, masukan BINER
K2 dengan pola pemasukan proses 2
Output K2
Print C2
C2 = P3
C3 <- P3 ⊕ K3
Dari C2 = blok matriks P3, masukan C2
P3 menggunakan Pola pemasukan awal
Dari blok matriks P3 = BINER, ambil bit
P3 dengan pola lapangan balap sepeda *(velodrome*) B
BINER to HEXA
Dari HEXA = Tabel *S-box*, masukan HEXA
HEXA konversi menggunakan *S-box*
Print BINER *S-box*
Dari BINER = blok matriks P3, masukan BINER
P3 menggunakan pola pemasukan proses 3
Output P3
Dari K2 = blok matriks K3, masukan K2
Dari blok matriks K3 = BINER, ambil bit K3
K3 dengan pola pengambilan B
Dari BINER = blok matriks K3, masukan BINER K3
K3 menggunakan pola pemasukan proses 3
Output K3
Print C3

C3 = P4

C4 <- P4 ⊕ K4

Dari C3 = blok matrik P4, masukan C3

P4 menggunakan Pola pemasukan awal

Dari blok matriks P4 = BINER, ambil bit

P4 dengan pola lapangan balap sepeda *(velodrome*) C

Dari BINER = blok matriks P4, masukan BINER

P4 dengan pola pemasukan proses 4

Output P4

Dari K3 = blok matriks K4, masukan K3

Dari blok matrik K4 = BINER, ambil bit K4

K4 dengan pola pengambilan C

Dari BINER = blok matriks K4, masukan BINER

K4 dengan pola pemasukan proses 4

Output K4

Print C4

Repeat

End

Proses Dekripsi

{Program ini digunakan untuk melakukan proses enkripsi data}

Kamus

P,K,P1,K1,P2,K2,P3,K3,P4,K4 = integer

C1,C2,C3,C4 = integer

Start

Input K

Read K

K to ASCII

ASCII to BINER

Dari BINER = blok matriks K1, masukan BINER

K1 menggunakan Pola pemasukan awal

Dari blok matriks K1 = BINER , ambil bit K1

K1 dengan pola lapangan balap sepeda *(velodrome)* A

Dari BINER = blok matriks K1, masukan BINER K1

Output K1

K1 = K2

Dari K1 = blok matriks K2, masukan Bit

K1 dengan pola pemasukan awal

Dari blok matriks K2 = BINER, ambil bit

K2 dengan pola pengambilan D

Dari BINER = blok matriks K2, masukan BINER K2

Output K2

K2 = K3

Dari K2 = blok matriks K3, masukan Bit K2

K2 dengan pola pemasukan awal

Dari blok matriks K3 = BINER, ambil bit K3

K3 dengan pola pengambilan B

Dari BINER = blok matriks K3, masukan BINER K3

Output K3

K3 = K4

Dari K3 = blok matriks K4, masukan K3

Dari blok matriks K4 = BINER, ambil bit K4

K4 dengan pola pengambilan C

Dari BINER = blok matriks K4, masukan BINER K4

Output K4

P4 <- C4 ⊕ K4

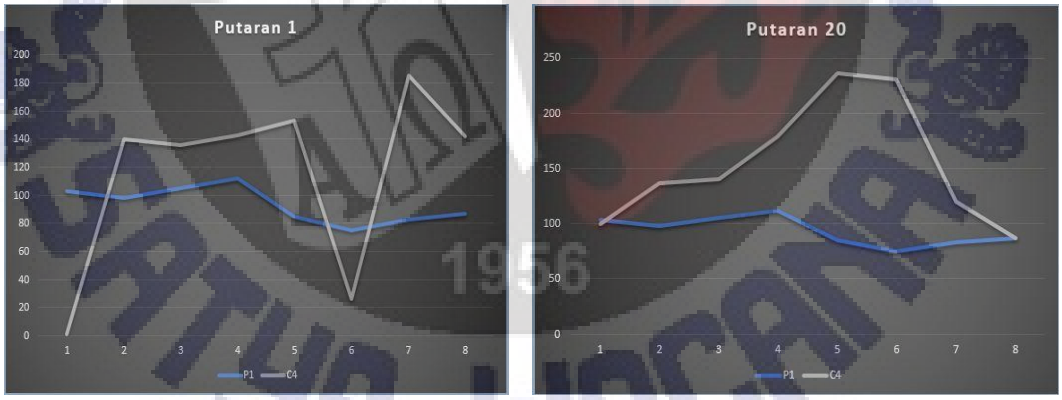
```
Input C
Read C
        C4 to ASCII
        ASCII to BINER
        Dari BINER = blok matriks C4, masukan BINER
        C4 ⊕ K4
Print P4
        Dari blok matriks P4 = BINER, ambil bit P4
                Dari BINER P4 = blok matrik P4, masukan BINER
                Menggunakan pola pengambilan A
Output P4
P4 = C3
        P3 <- C3 ⊕ K3
        Dari P4 = blok matriks C3, masukan BINER
        C3 ⊕ K3
Print P3
        Dari blok matriks P3 = BINER, ambil bit P3
        BINER to HEXA
                Dari HEXA = Tabel S-Box, masukan HEXA
                HEXA ditranformasi menggunakan S-Box
                        Dari BINER P3 = blok matriks P3, masukan BINER
                        Menggunakan pola pengambilan D
Output P3
P3 = C2
        P2 <- C2 ⊕ K2
        Dari P3 = blok matriks C2, masukan BINER
        C2 ⊕ K2
Print P2
        Dari blok matriks P2 = BINER, ambil bit P2
        BINER to HEXA
                Dari HEXA = Tabel S-Box, masukan HEXA
                HEXA ditranformasi menggunakan S-Box
                        Dari BINER P2 = blok matriks P2, masukan BINER
                        Menggunakan pola pengambilan C

Output P2
P2 = C1
        P1 <- C1 ⊕ K1
        Dari P2 = blok matriks C1, masukan BINER
        C1 ⊕ K1
Print P1
        Dari blok matriks P1 = BINER, ambil bit P1
                Dari BINER = blok matrik P1, masukan BINER
                Menggunakan pola pengambilan B
Output P1
P1 to BINER
BINER to ASCII
ASCII to CHAR
Print P
End
```

Pengujian korelasi digunakan untuk mengukur perbandingan antara *plaintext* dan *ciphertext*. Nilai korelasi berkisar antara 1 sampai -1, jika nilai korelasi mendekati angka 1 maka *plaintext* dan *ciphertext* memiliki hubungan yang kuat, sebaliknya jika mendekati angka 0 maka *plaintext* dan *ciphertext* memiliki hubungan yang lemah.

Tabel 6 Nilai Korelasi Setiap Putaran

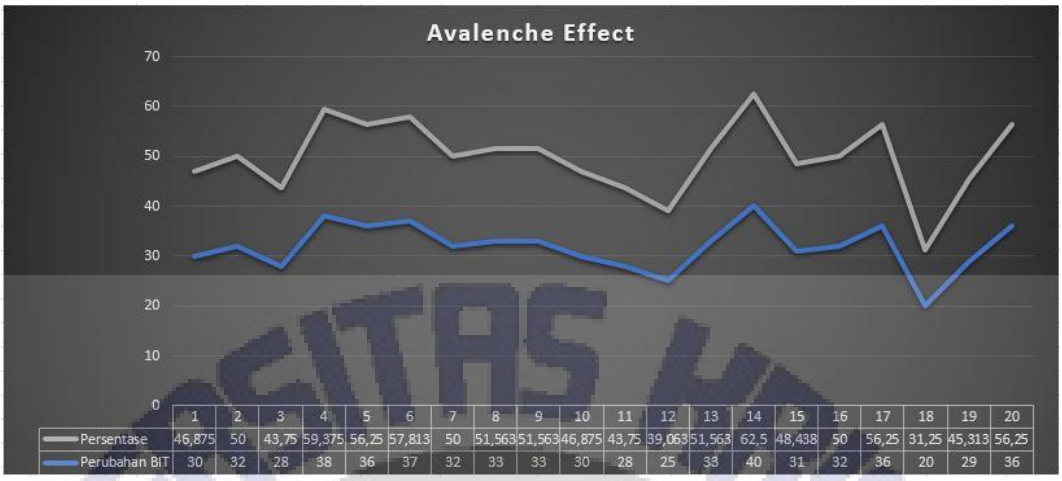
Putaran	Korelasi	Putaran	Korelasi
1	0,034455493	11	0,017193071
2	0,577888833	12	-0,187606999
3	0,561484671	13	-0,185842739
4	0,173065903	14	0,411705952
5	0,090436054	15	0,502344117
6	-0,18728265	16	-0,283087125
7	-0,055350837	17	-0,158353313
8	0,476449085	18	-0,390661874
9	-0,166470545	19	0,28500273
10	-0,612681667	20	-0,318861905

Tabel 6 menunjukkan nilai korelasi setiap putaran dan juga dapat disimpulkan bahwa algoritma kriptografi *block cipher* 64 bit menggunakan pola lapangan balap sepeda *(velodrome)* memiliki nilai korelasi yang lemah dan menghasilkan nilai korelasi yang acak.



Gambar 15 Grafik Perbandingan *Plaintext* dan *Ciphertext.*

Gambar 15 menunjukkan bahwa antara putaran satu dengan putaran yang lain memiliki perbedaan yang signifikan antara *plaintext* dan *ciphertext*. Pengujian *Avalanche Effect* dilakukan untuk mengetahui perubahan bit yang ada ketika *plaintext* diubah. Pengujian dilakukan dengan merubah karakter yang terdapat pada *plaintext* awal, sehingga akan menghasilkan perbedaan pada setiap putaran.



Gambar 16 Grafik Avalanche Effect

Pada Gambar 16 adalah hasil dari pengujian *Avalanche Effect*, pada penelitian ini *plaintext* awal adalah gbipUKSW yang kemudian diubah menjadi ReZa3010. Terjadi perubahan bit pada setiap putarannya, pada putaran ke- 14 perubahan bitnya terjadi cukup besar yaitu 62,5% dengan arti pada putaran ini terjadi perubahan bit yang baik, tetapi juga terjadi perubahan bit yang kecil pada putaran ke- 18 yaitu sebesar 31,25% ini berarti perubahan bitnya kurang baik. Berdasarkan hasil putaran pertama sampai dengan putaran ke dua puluh dapat disimpulkan bahwa rata-rata hasil pengujian *Avalanche Effect* ini yaitu sebesar 49,921875%. Presentase setiap putaran dapat dilihat pada Tabel 7.

Tabel 7 Tabel Presentase *Avalanche Effect*

Putaran	Presentase (%)	Putaran	Presentase (%)
1	46,875	**11**	43,75
2	50	**12**	39,0625
3	43,75	**13**	51,5625
4	59,375	**14**	62,5
5	56,25	**15**	48,4375
6	57,8125	**16**	50
7	50	**17**	56,25
8	51,5625	**18**	31,25
9	51,5625	**19**	45,3125
10	46,875	**20**	56,25

5. Simpulan

Berdasarkan penelitian yang telah dilakukan, dapat disimpulkan bahwa perancangan kriptografi *block cipher* berbasis pada pola lapangan balap sepeda *(velodrome)* dikatakan sebagai sistem kriptografi. Dalam proses enkripsi,

perancangan kriptografi *block cipher* berbasis pada pola lapangan balap sepeda *(velodrome)* ini menghasilkan *output* enkripsi yang acak, dan juga dengan adanya tabel subtitusi *S-BOX* yang dipasang pada proses kedua dan ketiga pada setiap putaran juga membuktikan bahwa hasil *ciphertext* tersebut menjadi semakin acak. sehingga dapat digunakan sebagai alternatif dalam pengamanan data. Dalam perancangan kriptografi *block cipher* 64 bit berbasis pada pola lapangan balap sepeda *(velodrome)*, pengujian *avalanche effect* yang dilakukan menunjukkan proses enkripsi di setiap putaran memiliki perubahan rata-rata sebesar 49,921875%.

6. Daftar Pustaka

[1] Husna, M. A., 2013, "Implementasi Algoritma MMB (*Modular Multiplication-Based Block Cipher*) Pada Pembuatan Aplikasi Manajemen Kata Sandi (*Password Management*). Ilmu Komputer, Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Sumatra Utara.

[2] Munir, R., 2006, "Kriptografi", Bandung: Informatika.

[3] Dipanegara, A., 2011, "*New Concept Hacking*". Jakarta: Agogos Publishig.

[4] Paliama, F. D., Wowor, A. D., 2016 "Perancangan Kriptografi *Block Cipher* Berbasis Pada Teknik Formasi Permainan Sepakbola". Teknik Infromatika, Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana.

[5] Tomasosa, E. L., Pakereng, M. A. I., 2016 "Pengaruh S-BOX *Advandce Encryption Standard* (AES) Terhadap *Avalanche Effect* Pada Perancangan Kriptografi *Block Cipher* 256 Bit Berbasis Pola Teknik Dansa Tali Dari Maluku". Teknik Informatika, Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana.

[6] Guntoro, Pakereng, M. A. I., 2016 "Perancangan Kriptografi *Block Cipher* Berbasis Pola Ikan Berenang". Teknik Informatika, Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana.

[7] Widodo, T., Pakereng, M. A. I., 2017"Perancangan Kriptografi Block Cipher Berbasis Pola Bercocok Tanam pada Game Harvest Moon". Teknik Informatika, Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana.

[8] Widodo, A, dkk., 2015 "Perancangan Kriptografi *Block Cipher* Berbasis pada Teknik Tanam Padi dan Bajak Sawah".Teknik Informatika, Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana.

[9] Suryonindito, A., 2012 " Arena Balap Sepeda *Velodrome* di Sleman, Daerah Istimewa Yogyakarta". Arsitektur, Fakultas Teknik, Universitas Atma Jaya Yogyakarta.

[10] http://www.velodromes.com/ Diakses tanggal 18 Juni 2017