上田研　ゼミ

Hybrid系の計算アルゴリズム

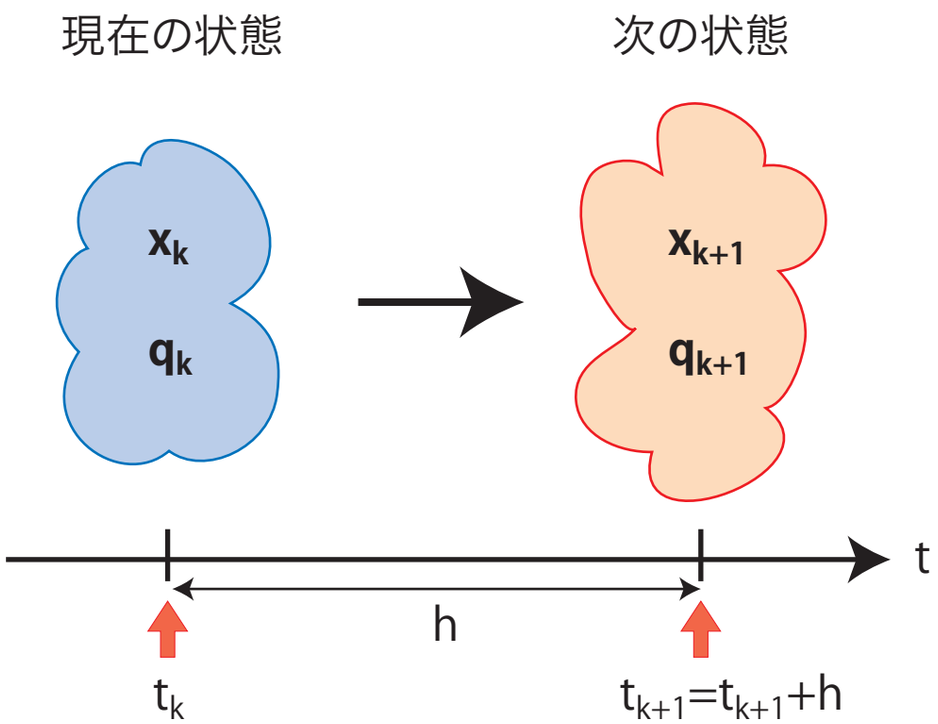
ーLEDホタルの結合系の解析にむけてー

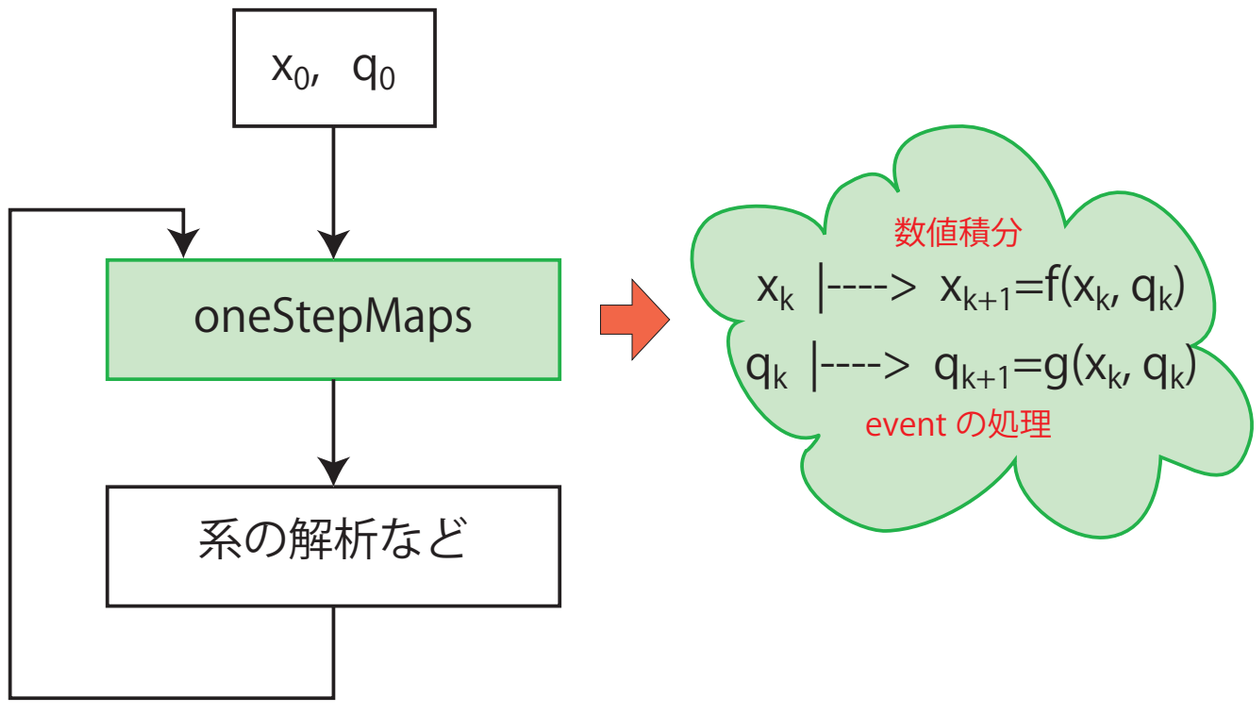
川上　博

2014(H26).04.21

話の概要

oneStepMaps

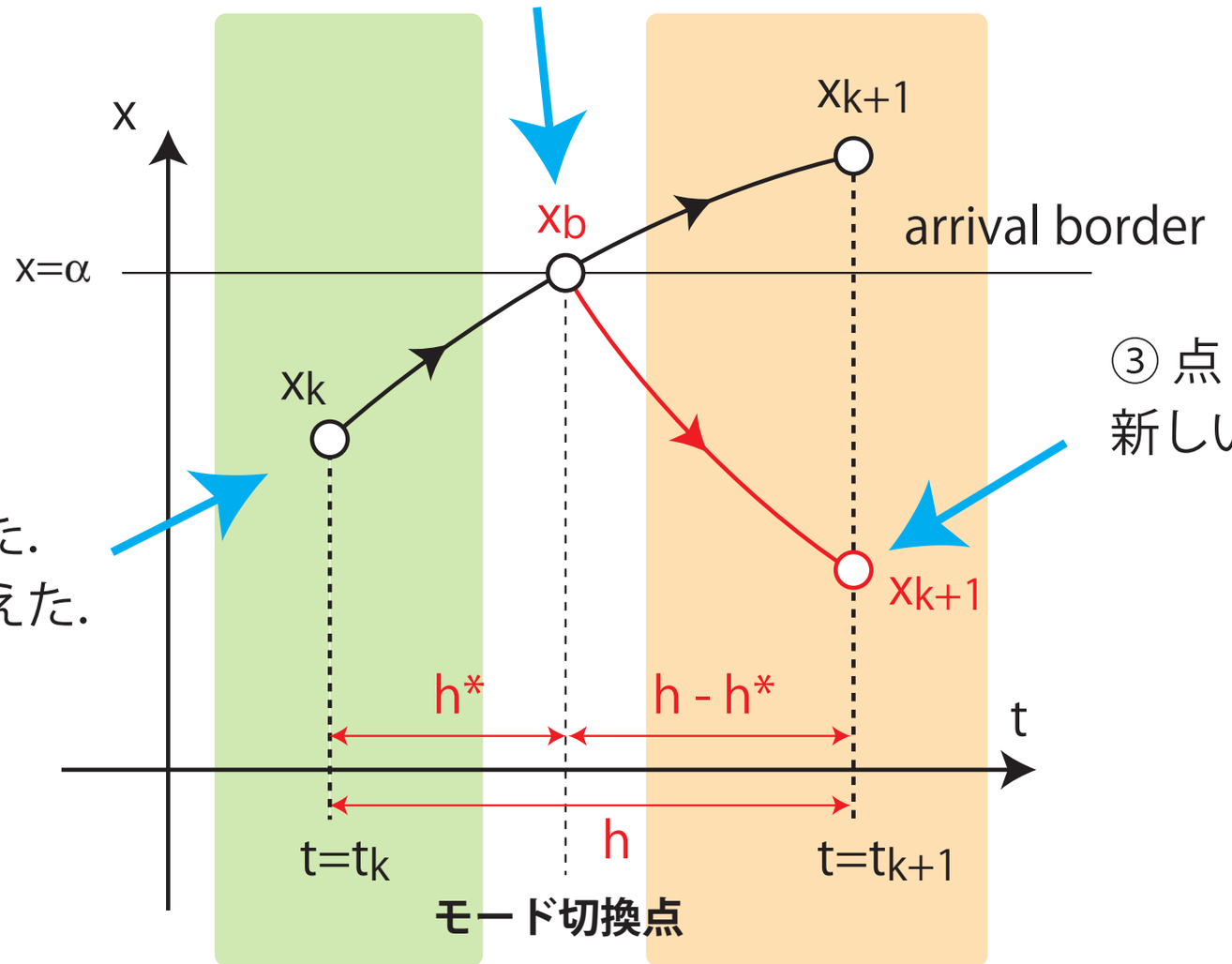




アナログ状態 xk とディジタルモードqkを１キザミ進めること

イベントの処理

② キザミ h* を求め，arrival border α 上の点 x_b を求める.

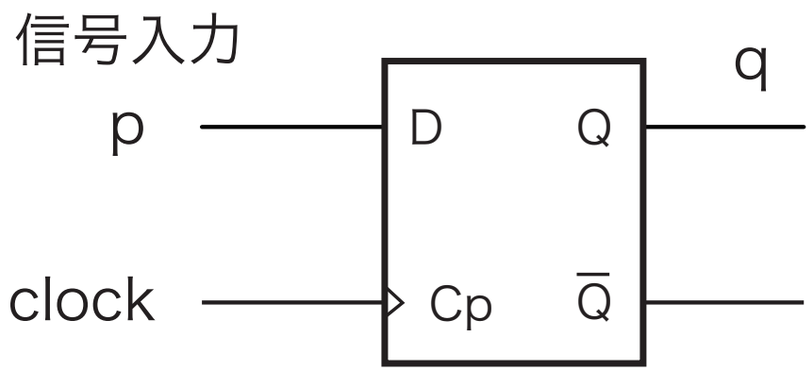


① $t=t_k$ で x_{k+1} を求めた.
x_{k+1} が border α を越えた.

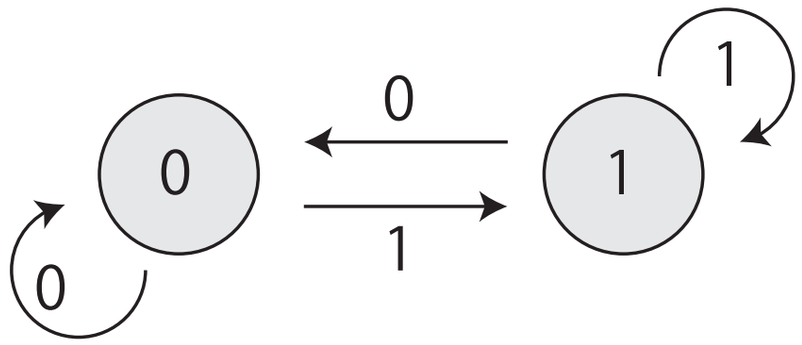
③ 点 x_b から改めて
新しい x_{k+1} を求める.

ディジタル部分系の扱い

D FF



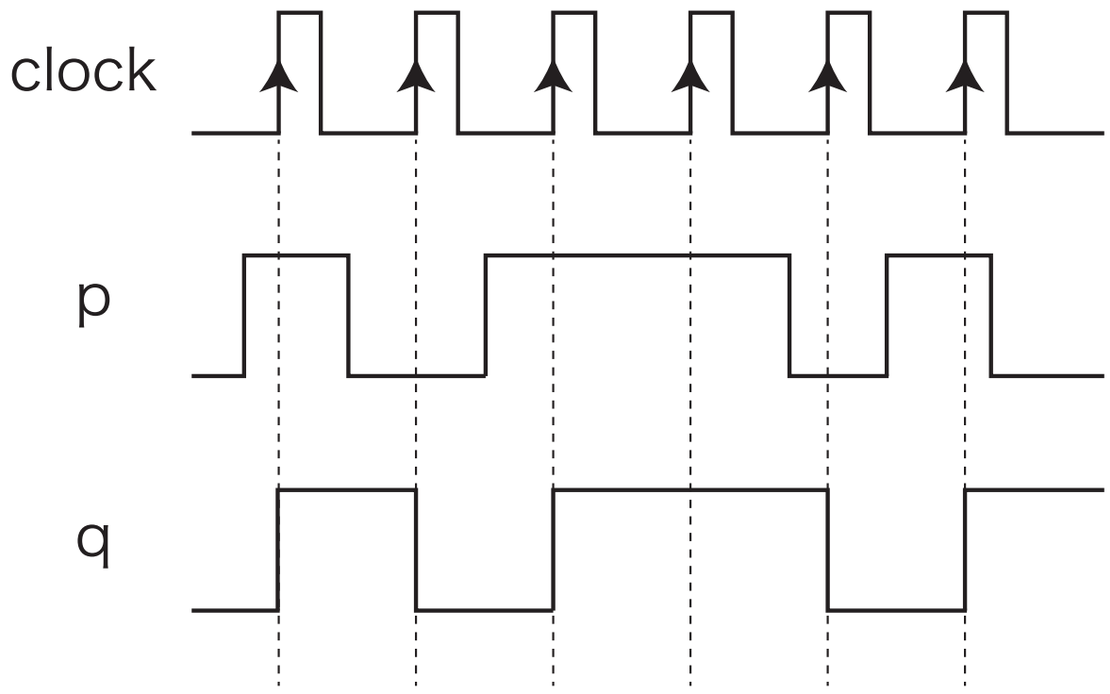
mode transition diagram



mode transition table

現在のモード q_k	入力 p_k	次のモード q_{k+1}
0	0	0
	1	1
1	0	0
	1	1

wave forms



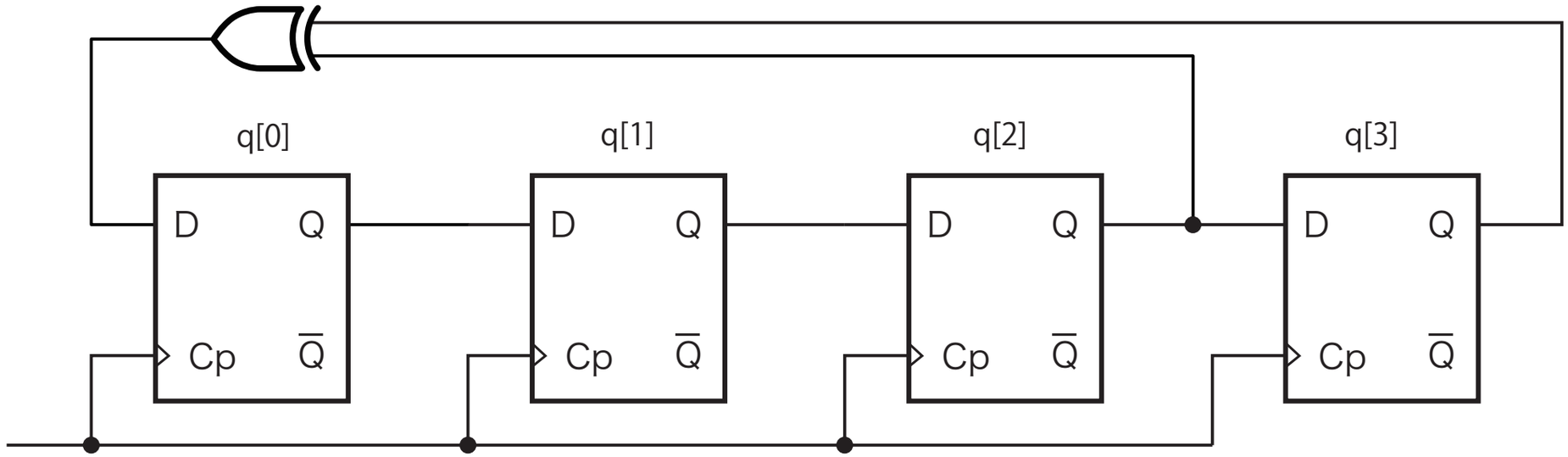
D Flip Flop

```
int oneStepMaps(int q, int p){

    switch (q) {
        case 0:
            if (p==1) { q=1; }
            break;
        case 1:
            if (p==0) { q=0; }
            break;
        default:
            break;
    }
    return q;
}


int oneStepMaps(int q, int p){
   return p;
}
```

Linear Feedback Shift Register

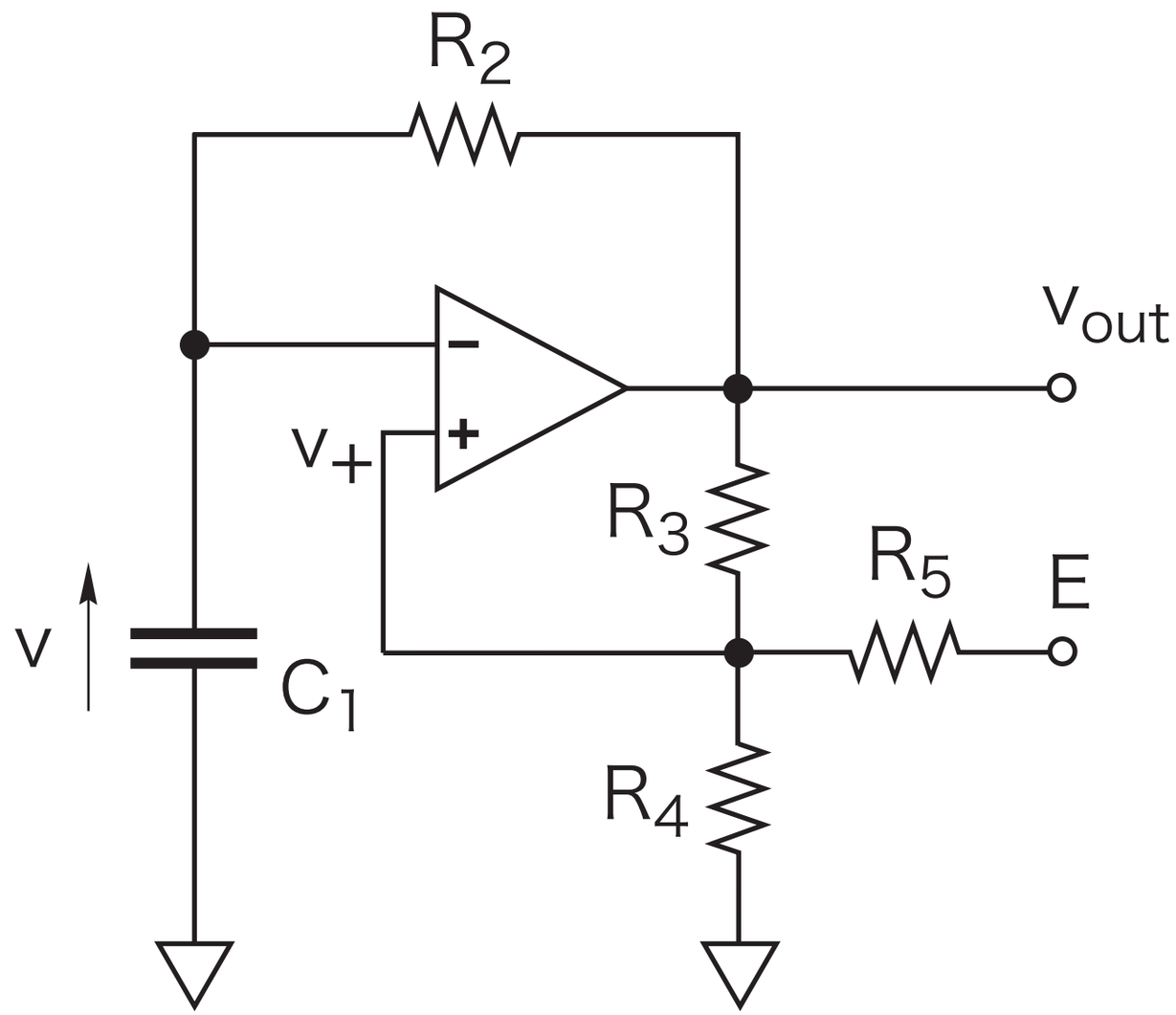


クロック入力

```
void nextMode(int q[]){
    int temp1,temp2;

    temp1=q[3]; temp2=q[2];
    q[3]=q[2]; q[2]=q[1]; q[1]=q[0];
    q[0]=temp1^temp2;
}
```

RC Square Wave Oscillator



```
void oneStepMaps(void){

    runge(1,h,x,tt, q);
    switch (q[0]) {    // define arrival border function
        case 0:
            bdrB[0]=betaOff;
            break;
        case 1:
            bdrA[0]=alphaOff;
            break;
        default:
            break;
    }
    switch (q[0]) {    // mode transition
        case 0:
            if (x[0]<bdrB[0]) { q[0]=1; }
            break;
        case 1:
            if (x[0]>bdrA[0]) { q[0]=0; }
            break;
        default:
            break;
    }
}
```