

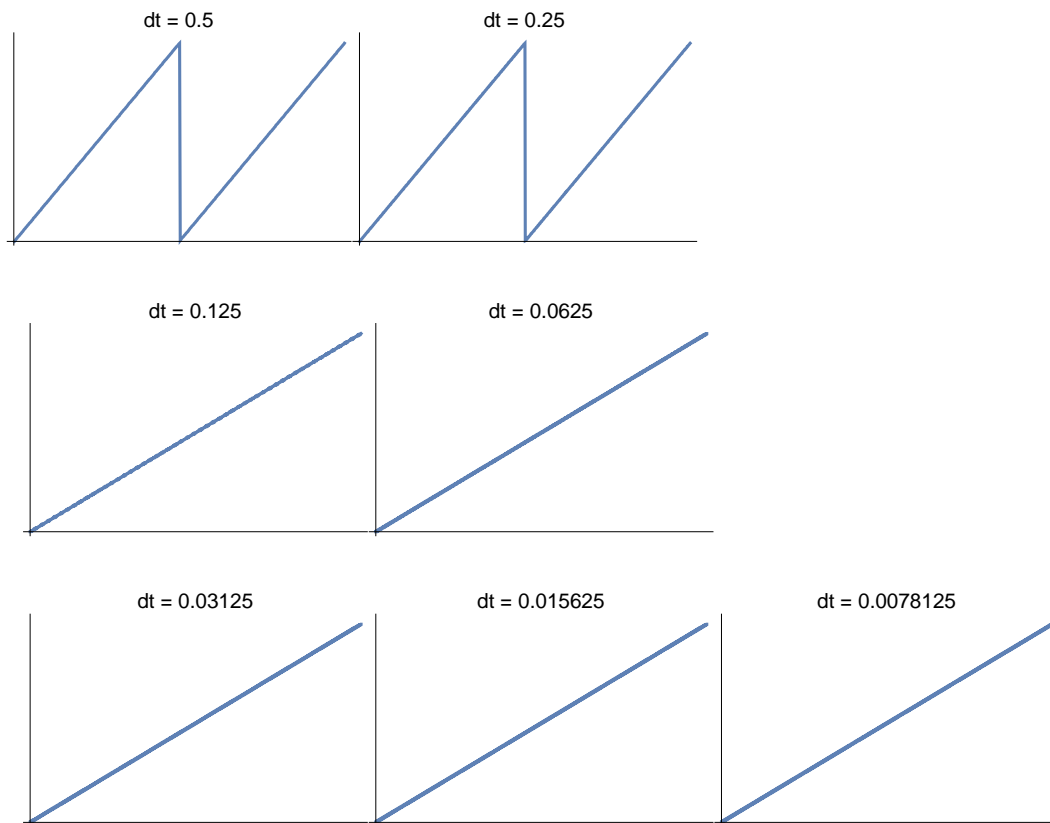
# Rounding Issues in Simple Euler Integration

---

## Using Machine Precision

### Basic Formulation

```
Table[
  Clear[dt, stock, inflow, outflow];
  dt := 1. / 2i;
  inflow[_] := 1. / 100;
  outflow[t_] := If[stock[t] == 1.,  $\frac{\text{stock}[t]}{\text{dt}}$ , 0.];
  stock[0.] := 0.;
  stock[t_] := stock[t] = stock[t - dt] + (inflow[t - dt] - outflow[t - dt]) × dt;
  ListLinePlot[
    Table[
      stock[τ],
      {τ, 0., 200., dt}
    ],
    Ticks → None,
    PlotLabel → "dt = " <> ToString[dt],
    ImageSize → Small
  ],
  {i, 1, 7}
] // Row
```



## Using Approximate Equality-Check

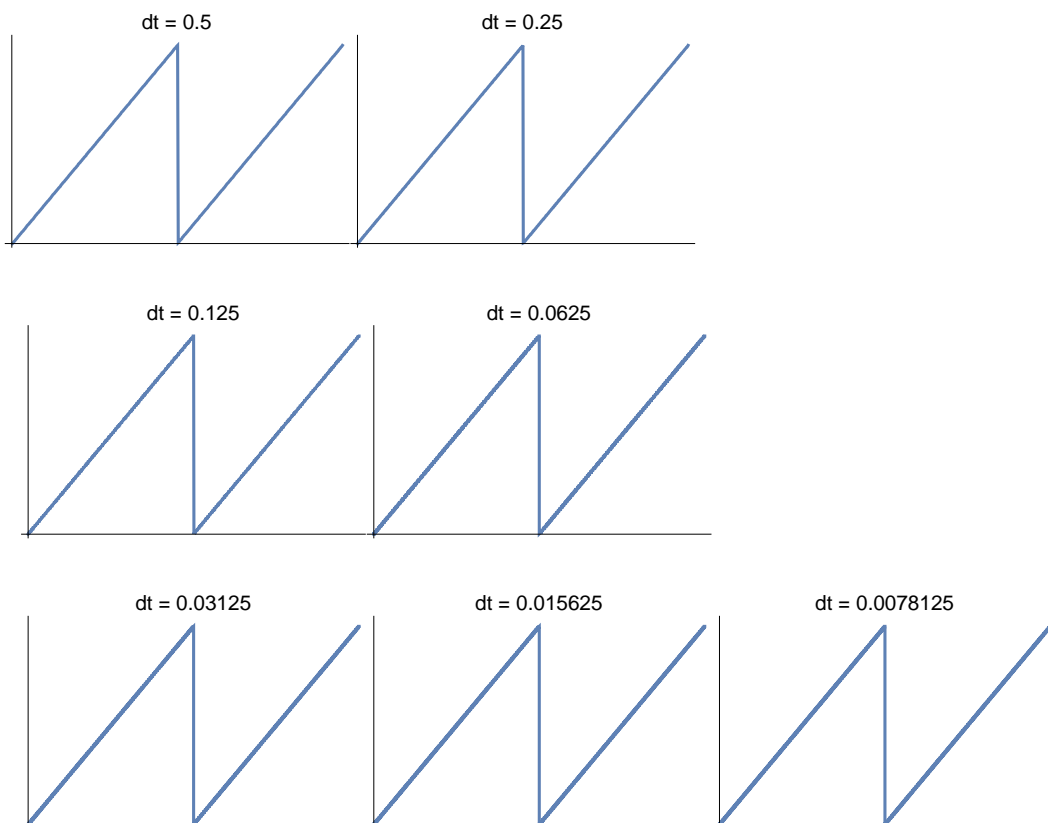
### ? Chop

`Chop[expr]` replaces approximate real numbers in *expr* that are close to zero by the exact integer 0. >>

```

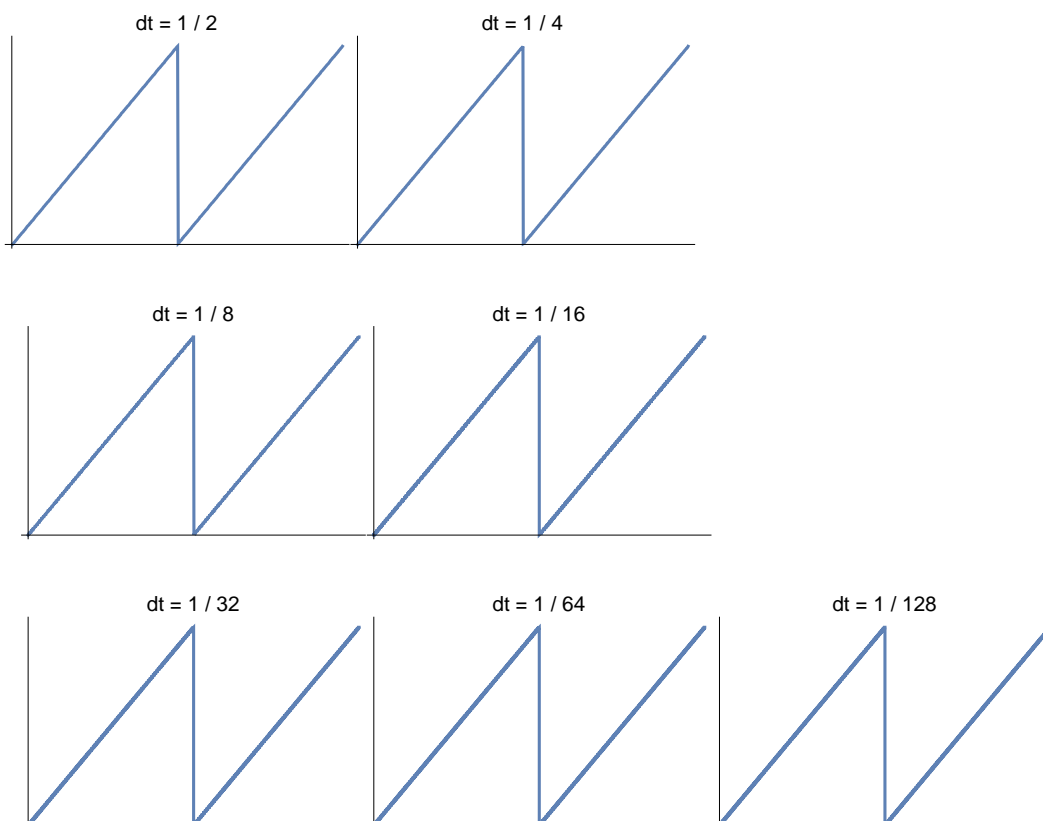
Table[
  Clear[dt, stock, inflow, outflow];
  dt := 1./2i;
  inflow[_] := 1./100;
  outflow[t_] := If[Chop[stock[t] - 1.] == 0.,  $\frac{\text{stock}[t]}{\text{dt}}$ , 0.];
  stock[0.] := 0.;
  stock[t_] := stock[t] = stock[t - dt] + (inflow[t - dt] - outflow[t - dt]) × dt;
  ListLinePlot[
    Table[
      stock[τ],
      {τ, 0., 200., dt}
    ],
    Ticks → None,
    PlotLabel → "dt = " <> ToString[dt],
    ImageSize → Small
  ],
  {i, 1, 7}
] // Row

```



## Using Higher Precision

```
Table[
  Clear[dt, stock, inflow, outflow];
  dt := 1 / 2i;
  inflow[_] := 1 / 100;
  outflow[t_] := If[stock[t] == 1,  $\frac{\text{stock}[t]}{\text{dt}}$ , 0];
  stock[0] := 0;
  stock[t_] := stock[t] = stock[t - dt] + (inflow[t - dt] - outflow[t - dt]) × dt;
  ListLinePlot[
    Table[
      stock[τ],
      {τ, 0, 200, dt}
    ],
    Ticks → None,
    PlotLabel → "dt = 1 / " <> ToString[2i],
    ImageSize → Small
  ],
  {i, 1, 7}
] // Row
```



# Taking a deeper look at Precision

## Precision and Accuracy in *Mathematica*

Precision (the number of significant digits) and Accuracy (the number of decimal digits known to be true) are complementary concepts in *Mathematica* and separated from each other.

So we might take the Microsoft example:

```
1. × (0.5 - 0.4 - 0.1)
- 2.77556 × 10-17
```

While all the numbers have the same Precision...

```
Precision /@ {1., 0.5, 0.4, 0.1}
{MachinePrecision, MachinePrecision, MachinePrecision, MachinePrecision}
```

their Accuracy differs:

```
Accuracy /@ {1., 0.5, 0.4, 0.1}
{15.9546, 16.2556, 16.3525, 16.9546}
```

*Mathematica* will keep track of Precision and Accuracy during numerical calculations:

```
Precision[1. × (0.5 - 0.4 - 0.1)]
MachinePrecision
```

```
Accuracy[1. × (0.5 - 0.4 - 0.1)]
32.5112
```

We may set the Accuracy (or Precision) arbitrarily if we feel we know “better”:

```
Accuracy[1.``50 × (0.5``50 - 0.4``50 - 0.1``50)]
49.5229
```

... which will influence the result:

```
1.``50 × (0.5``50 - 0.4``50 - 0.1``50)
0. × 10-50
```

in *Mathematica* we will also be able to use infinite Precision and Accuracy:

```
1 × (5/10 - 4/10 - 1/10)
0
```

```
Through[{Precision, Accuracy}[%]]
{∞, ∞}
```

## Machine Precision in *Mathematica*

```
Block[
  {
    stock,
    dt
  },
  Table[
    stock = 0.;
    While[ stock < 1.,
      (* do *)
      stock = stock +  $\frac{1.}{100.} \times dt$ 
    ];
    {dt, stock},
    {dt, {0.5, 0.25, 0.125, 0.0625}}
  ]
] // TableForm[#, {None, {"dt", "last value of stock"}}] &
```

0.5	1.
0.25	1.
0.125	1.00125
0.0625	1.00062

## Higher Precision in *Mathematica*

We now set the precision  $p$  to 32 digits.



```
Block[
{
  stock,
  dt,
  a = 32
},
Table[
  stock = SetAccuracy[0., a];
  While[stock < SetAccuracy[1., a],
    (* do *)
    stock = stock + SetAccuracy[ $\frac{1}{100}$ , a] × dt
  ];
  {dt, stock},
  {dt, SetAccuracy[#, a] & /@ {0.5, 0.25, 0.125, 0.0625}}
]
] // TableForm[#, {None, {"dt", "last value of stock"}}] &
```

## Infinite Precision and Accuracy

Entering exact fractions using integer numbers gives infinite precision and accuracy:



```

Block[
  {
    stock,
    dt,
  },
  Table[
    stock = 0;
    While[stock < 1,
      (* do *)
      stock = stock +  $\frac{1}{100} \times dt$ 
    ];
    {dt, stock},
    {dt, { $\frac{1}{2}$ ,  $\frac{1}{4}$ ,  $\frac{1}{8}$ ,  $\frac{1}{16}$ }}
  ]
] // TableForm[#, {None, {"dt", "last value of stock"}}] &

```

$\frac{1}{2}$	1
$\frac{1}{4}$	1
$\frac{1}{8}$	1
$\frac{1}{16}$	1