

Decimal to Binary system

convert the integer to binary

$$64_{10} \rightarrow \begin{cases} 2)\underline{64} & \uparrow & 0 \\ 2)\underline{32} & \uparrow & 0 \\ 2)\underline{16} & 0 \\ 2)\underline{8} & 0 \\ 2)\underline{4} & 0 \\ 2)\underline{2} & 0 \\ 2)\underline{1} & 1 \end{cases} 1000000_{2} \tag{1}$$

 $64_{10} \Rightarrow 1 0 0 0 0 0 0$

$$17_{10} \rightarrow \begin{cases} 2)\frac{17}{2)8} & 1\\ 2)\frac{1}{4} & 0\\ 2)\frac{1}{2} & 0\\ 2)1 & 1 \end{cases} 10001_{2} \tag{2}$$

 $17_{10} \Rightarrow \boxed{1} \boxed{0} \boxed{0} \boxed{1}$

$$227_{10} \rightarrow \begin{cases} 2)\underline{227} & 1 \\ 2)\underline{113} & 1 \\ 2)\underline{56} & 0 \\ 2)\underline{28} & 0 \\ 2)\underline{14} & 0 \\ 2)\underline{7} & 1 \\ 2)\underline{3} & 1 \\ 2)\underline{1} & 1 \end{cases}$$
 11100011₂ (3)

$$227_{10} \Rightarrow |1||1||1||0||0||0||1||1$$

 $227_{10} \Rightarrow \boxed{1} \boxed{1} \boxed{0} \boxed{0} \boxed{1} \boxed{1}$ The matlab function to convert a integer to binary is dec2bin(N), where N is the integer.

Convert the fractional part to binary 1.2

• For 10.5

For the integer part

$$10_{10} \to \begin{cases} 2)\frac{10}{2)5} & 0 \\ 2)\frac{5}{2} & 0 \\ 2)\frac{1}{1} & 0 \\ 1 & 1 \end{cases} 1010_2 \tag{4}$$

For fractional part

$$\begin{array}{c|c}
\hline
 & . & 5 \\
 & \times 2 \\
\hline
 & 1 & . & 0
\end{array}$$

$$\begin{array}{c|c}
\hline
 & 1
\end{array}$$

$$\begin{array}{c|c}
\hline
 & 1
\end{array}$$

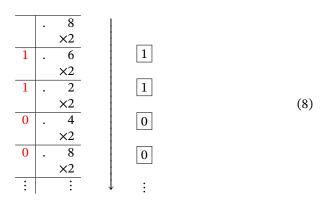
So

$$10.5_{[10]} = 1010.1_{[2]} \tag{6}$$

• For 12.8

$$12_{10} \to \begin{cases} \frac{2)12}{2)6} & 0\\ \frac{2)3}{2)1} & 1\\ 1 & 1 \end{cases} 1100_2 \tag{7}$$

For the fractional part



So

$$12.8_{[10]} = 1100.\bar{1}\bar{1}\bar{0}\bar{0}_{[2]} \tag{9}$$

• For 55.4

The integer and fractional parts convert:

$$55_{10} \rightarrow \begin{cases} 2)\underline{55} & \uparrow & 1\\ 2)\underline{27} & \uparrow & 1\\ 2)\underline{13} & 1\\ 2)\underline{6} & 2)\underline{3} & 1\\ 2)\underline{1} & 1 \end{cases} 110111_{2}, \qquad \begin{array}{c|c} & . & 4\\ & \times 2\\ \hline 0 & . & 8\\ & \times 2\\ \hline 1 & . & 6\\ & & \times 2\\ \hline 1 & . & 2\\ & & \times 2\\ \hline 0 & . & 4\\ & & \times 2\\ \hline \vdots & \vdots & & \vdots \end{cases} \qquad \begin{array}{c|c} & 0\\ \hline 1\\ \hline 1\\ \hline 0\\ \hline \end{array}$$

Hence $55.4_{10} = 110111.\bar{0}\bar{1}\bar{1}\bar{0}_2$

• For 1/3

We only need to convert the fractional part:

Hence
$$\frac{1}{3} = 0.\bar{0}\bar{1}_2$$

Since there is not built-in function in matlab to transform float number to binary, we write a function to calculate it. Usage:

```
1  >>  f = float2bin(55.4)
2  f =
    '110111.<0110>...'
```

The part included by "<>" is the repeated digits.

```
function u = float2bin(f)
  %This function converts a floating point number to a binary string.
  %Floating Point Binary Formats
  %Input checking
b = mod(f,1);
a = f - b;
                                   % the fractional part
                                   % the integer part
  %% for the integer part:
  u1 = dec2bin(a);
13
  %% for the fractional part
14
      %% initial u2 ,
      u2 = [];
                                  % empty vector
      B = [];
17
      % fractional part of 2*b while b~=0 && notin(b,B) % t
18
19
           B = [B,b];
b = 2*b;
20
           % calculate 2b
21
23
24
           u2 = [u2,t];
                                  % append a 0 or 1 to u if t is 0 or 1
25
      end
26
27
      if ~notin(b,B)
                                  % if repeated
          [~,i] = notin(b,B);
u2 = [u2(1:i-1),'<',u2(i:end),'>...'];
31
32
33
  u = [u1,'.',u2];
                                       % put integer and fractional parts together
35
  end
  function [y,j]=notin(c,d)
37
  %% fucntion to check whether c is in vector b
% if not, return 1
38
       otherwise, we also return the index the first one
      for j=1:length(d)
42
          if abs(d(j) - c) < 1e-8 \% may have rounding error
43
              y=0;
44
              break
45
          end
49
  end
```