无线通信实验在线开放课程

主讲人：吴光 博士

广东省教学质量工程建设项目

An example in our daily life

Lab 16: LDPC Code

主讲人： 吴光 博士
Email: wug@sustech.edu.cn

Algorithm Analysis and Verification

1. Hamming Code Simulation
2. LDPC Design and Simulation
3. USRP Verification
4. Analysis and Discussion

Noisy-channel coding theorem

- Given a noisy channel with channel capacity C and information transmitted at a rate R, then if

$$\boxed{R < C}$$

- There exists a coding technique which allows the probability of error at the receiver to be made arbitrarily small.

(7,4) Hamming Encoding

Code word

$$(\underbrace{C_6, C_5, C_4, C_3}_{\text{Transmitted bits}}, \underbrace{\mathbf{C_2}, \mathbf{C_1}, \mathbf{C_0}}_{\text{Redundant bits}})$$

Transmitted bits

Redundant bits

$$\begin{cases} C_2 = C_6 + C_5 + C_4 \\ C_1 = C_5 + C_4 + C_3 \\ C_0 = C_6 + C_5 + C_3 \end{cases}$$

0000 →	0 0 0 0 **0 0 0**
0001 →	0 0 0 1 **0 1 1**
0010 →	0 0 1 0 **1 1 0**
0011 →	0 0 1 1 **1 0 1**
0100 →	0 1 0 0 **1 1 1**
0101 →	1 0 0 1 **1 0 0**
... →	......
1111 →	1 1 1 1 **1 1 1**

Generator Matrix

$$\mathrm{m}_{1\times4}\mathbf{G}_{4\times7} = \mathrm{C}_{1\times7}$$

Transmitted bits

$$\mathbf{G}_{4\times7} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

Generator matrix

$$\begin{cases} C_2 = \boxed{C_6 + C_5 + C_4} \\ C_1 = C_5 + C_4 + C_3 \\ C_0 = C_6 + C_5 + C_3 \end{cases}$$

Example

$$m_{1\times4} = [0 \quad 1 \quad 0 \quad 1]$$

$$[0 \quad 1 \quad 0 \quad 1]\begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} = [0 \quad 1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 0]$$

Transmitted bits

Generator matrix

Code word

Parity-check Matrix

$$\begin{cases} C_2 = C_6 + C_5 + C_4 \\ C_1 = C_5 + C_4 + C_3 \\ C_0 = C_6 + C_5 + C_3 \end{cases} \Rightarrow \begin{cases} C_6 + C_5 + C_4 + 0 + C_2 + 0 + 0 = 0 \\ 0 + C_5 + C_4 + C_3 + 0 + C_1 + 0 = 0 \\ C_6 + C_5 + 0 + C_3 + 0 + 0 + C_0 = 0 \end{cases}$$

Parity-check matrix

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

Parity-check Matrix

$$\begin{cases} C_2 = C_6 + C_5 + C_4 \\ C_1 = C_5 + C_4 + C_3 \\ C_0 = C_6 + C_5 + C_3 \end{cases} \Rightarrow \begin{cases} C_6 + C_5 + C_4 + 0 + C_2 + 0 + 0 = 0 \\ 0 + C_5 + C_4 + C_3 + 0 + C_1 + 0 = 0 \\ C_6 + C_5 + 0 + C_3 + 0 + 0 + C_0 = 0 \end{cases}$$

Parity-check matrix

Received bits

Syndrome vector

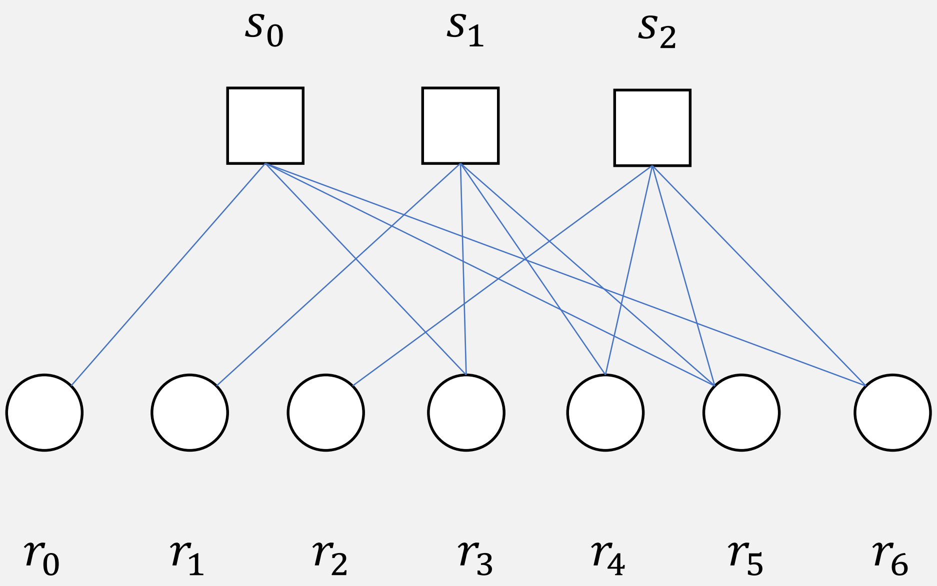
$$\mathrm{S}^T = \mathrm{H} \cdot \mathrm{R}^T = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_6 \\ r_5 \\ r_4 \\ r_3 \\ r_2 \\ r_1 \\ r_0 \end{bmatrix} = \begin{bmatrix} r_6 + r_5 + r_4 + r_2 \\ r_5 + r_4 + r_3 + r_1 \\ r_6 + r_5 + r_3 + r_0 \end{bmatrix} = \begin{bmatrix} s_2 \\ s_1 \\ s_0 \end{bmatrix} \stackrel{?}{=} \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Another View--Tanner Graph

$$\begin{bmatrix} s_2 \\ s_1 \\ s_0 \end{bmatrix} = \begin{bmatrix} r_6 + r_5 + r_4 + r_2 \\ r_5 + r_4 + r_3 + r_1 \\ r_6 + r_5 + r_3 + r_0 \end{bmatrix}$$

$$\begin{array}{cc} \mathrm{H} = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} & \begin{array}{c} s_2 \\ s_1 \\ s_0 \end{array} \\ \quad\;\; \begin{array}{ccccccc} r_6 & r_5 & r_4 & r_3 & r_2 & r_1 & r_0 \end{array} & \end{array}$$

$\Longleftrightarrow$



Error correction

$$R_i = C_i$$

$$i = 0,1, \dots, 6$$

$$S^T = H \cdot R^T = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0+1+0+1 \\ 1+0+1+0 \\ 0+1+1+0 \end{bmatrix} = \begin{bmatrix} 2 \\ 2 \\ 2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$$R_3 \neq C_3$$

$$S^T = H \cdot R^T = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0+1+0+1 \\ 1+0+0+0 \\ 0+1+0+0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} \neq \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Error pattern and Syndrome vector

Error pattern

1	0	0	0	0	0	0
0	1	0	0	0	0	0
0	0	1	0	0	0	0
0	0	0	1	0	0	0
0	0	0	0	1	0	0
0	0	0	0	0	1	0
0	0	0	0	0	0	1

Syndrome vector

1	1	1	0	1	0	0
0	1	1	1	0	1	0
1	1	0	1	0	0	1

The relationship between G and H

$$\begin{cases} \mathbf{G} = [\mathbf{I}_k \quad \mathbf{Q}_{k\times r}] \\ \mathbf{H} = [(\mathbf{Q}_{k\times r})^T \quad \mathbf{I}_r] \end{cases} \Leftrightarrow$$

$$\mathbf{G}_{(4,7)} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

$$\mathbf{H}_{(3,7)} = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

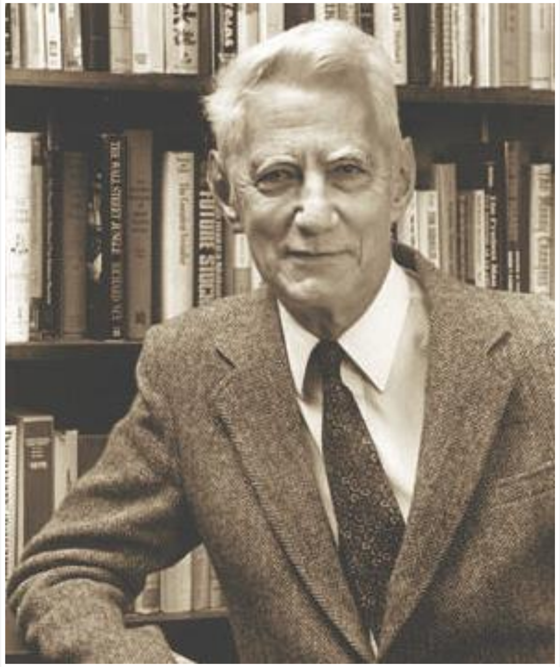
Exercise: (7,4) Hamming Encoding

Algorithm Analysis and Verification

1. Hamming Code Simulation
2. LDPC Design and Simulation
3. USRP Verification
4. Analysis and Discussion

Basic hypothesis of Shannon's coding theorem

- **Random** coding
- Codeword length tends to **infinity**
- **Maximum-Likelihood** Decoding



Regular LDPC

The parity-check matrix of a LDPC has ***sparse*** property, that is:

As for a $m \times n$ parity-check matrix H:

- The number of 1's in any column (the row weight w_r), is much less than row-length ($w_r \ << \ m$).
- The number of 1's in any row (the column weight w_c), is much less than column-length ($w_c \ << n$).
- w_c is constant for every column, w_r is constant for every row and $\frac{w_r}{m}=\frac{w_c}{n}$.

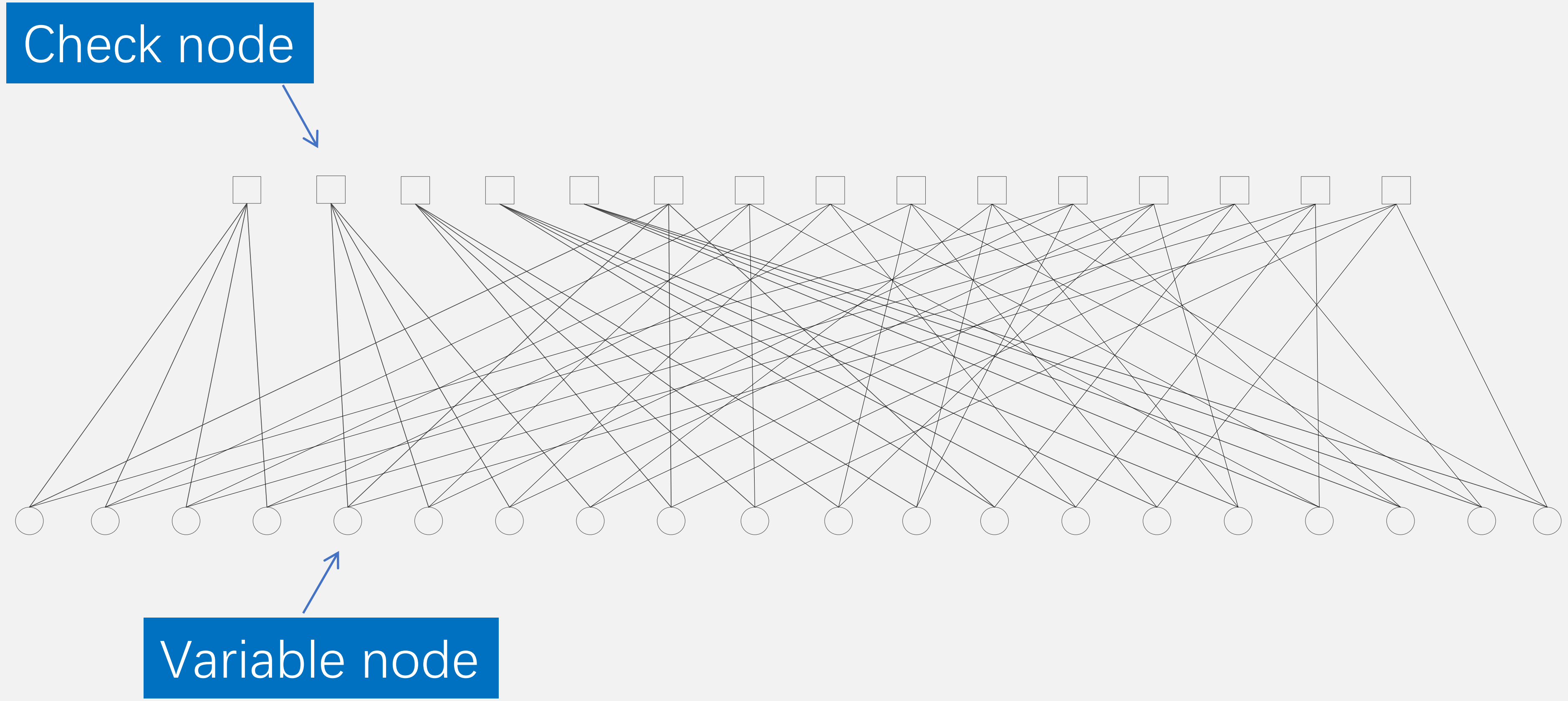
Here is an example of H

$$
\left[\begin{array}{ccccc:ccccc:ccccc:ccccc}
1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\
\hdashline
1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\
\hdashline
1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1
\end{array}\right]
$$

(20, 3, 4) $m = 15, n = 20, w_r = 4, w_c = 3$

Low-Density Parity-Check Codes, Robert G. Gallager,1963.

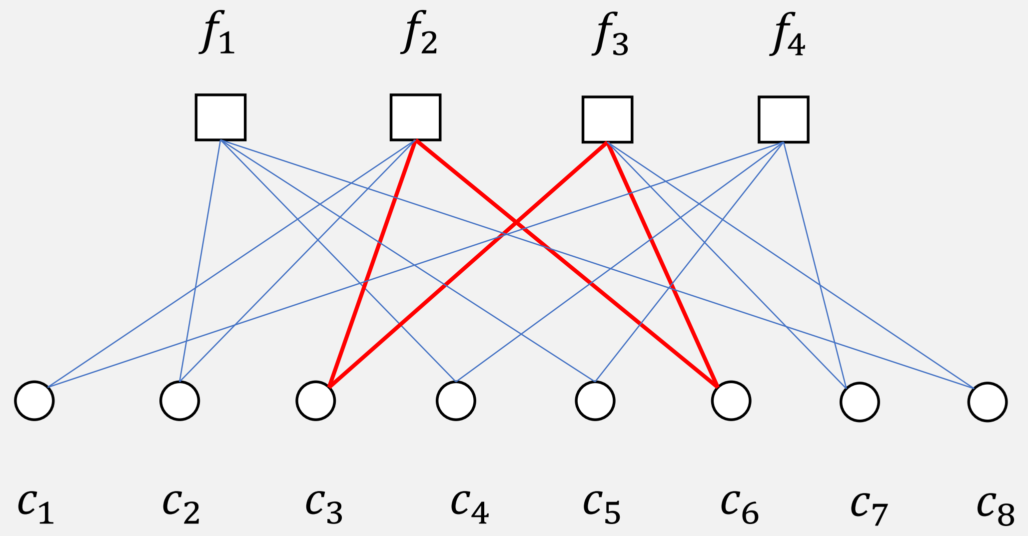
Tanner graph



Impact of girth 4

$$\mathbf{H} = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix}$$

(8, 2, 4)



Construction of Parity-check matrix

Initialization

↓

Random Construction

↓

Remove 4-cycles

Column permutation

$$\begin{array}{c} \\ H_1 \\ \\ \\ H_2 \\ \\ \\ H_3 \\ \\ \end{array} \left[\begin{array}{cccccccccccc} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ \hline 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ \hline 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \end{array}\right]$$

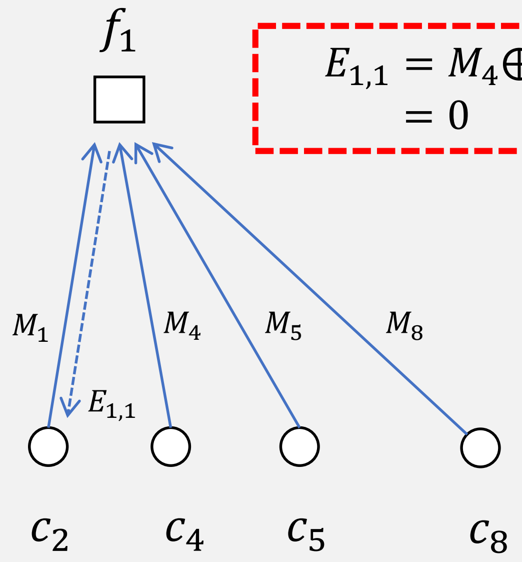
A length 12 (3,4)-regular Gallager parity-check matrix

Flowchart of Hard-decision Decoding

Initialization	All variable nodes send a message to their connected check nodes.
Check message	Every check nodes calculate a response to their connected variable nodes
Variable nodes update	Variable nodes use the messages they get from the check nodes to decide if the bit at their position is a 0 or a 1 by **majority rule**.
Syndrome decoding	Repeat step 2 until either exit at step 2 or a certain number of iterations has been passed.

Bit-flipping decoding

$$c = [1\ 0\ 0\ 1\ 0\ 1\ 0\ 1]^T$$
$$y = [1\ 1\ 0\ 1\ 0\ 1\ 0\ 1]^T$$



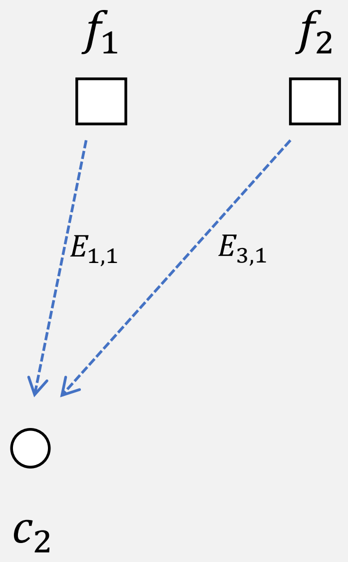
Check nodes update

$$E_{1,1} = M_4 \oplus M_5 \oplus M_8 = 0$$

check nodes	$E_{i,j}$				
f_1	receive	$c_2 \to 1$	$c_4 \to 1$	$c_5 \to 0$	$c_8 \to 1$
	send	$0 \to c_2$	$0 \to c_4$	$1 \to c_5$	$0 \to c_8$
f_2	receive	$c_1 \to 1$	$c_2 \to 1$	$c_3 \to 0$	$c_6 \to 1$
	send	$0 \to c_1$	$0 \to c_2$	$1 \to c_3$	$0 \to c_6$
f_3	receive	$c_3 \to 0$	$c_6 \to 1$	$c_7 \to 0$	$c_8 \to 1$
	send	$0 \to c_3$	$1 \to c_6$	$0 \to c_7$	$1 \to c_8$
f_4	receive	$c_1 \to 1$	$c_4 \to 1$	$c_5 \to 0$	$c_7 \to 0$
	send	$1 \to c_1$	$1 \to c_4$	$0 \to c_5$	$0 \to c_7$

Bit-flipping decoding

$$c = [1\ 0\ 0\ 1\ 0\ 1\ 0\ 1]^T$$
$$y = [1\ 1\ 0\ 1\ 0\ 1\ 0\ 1]^T$$

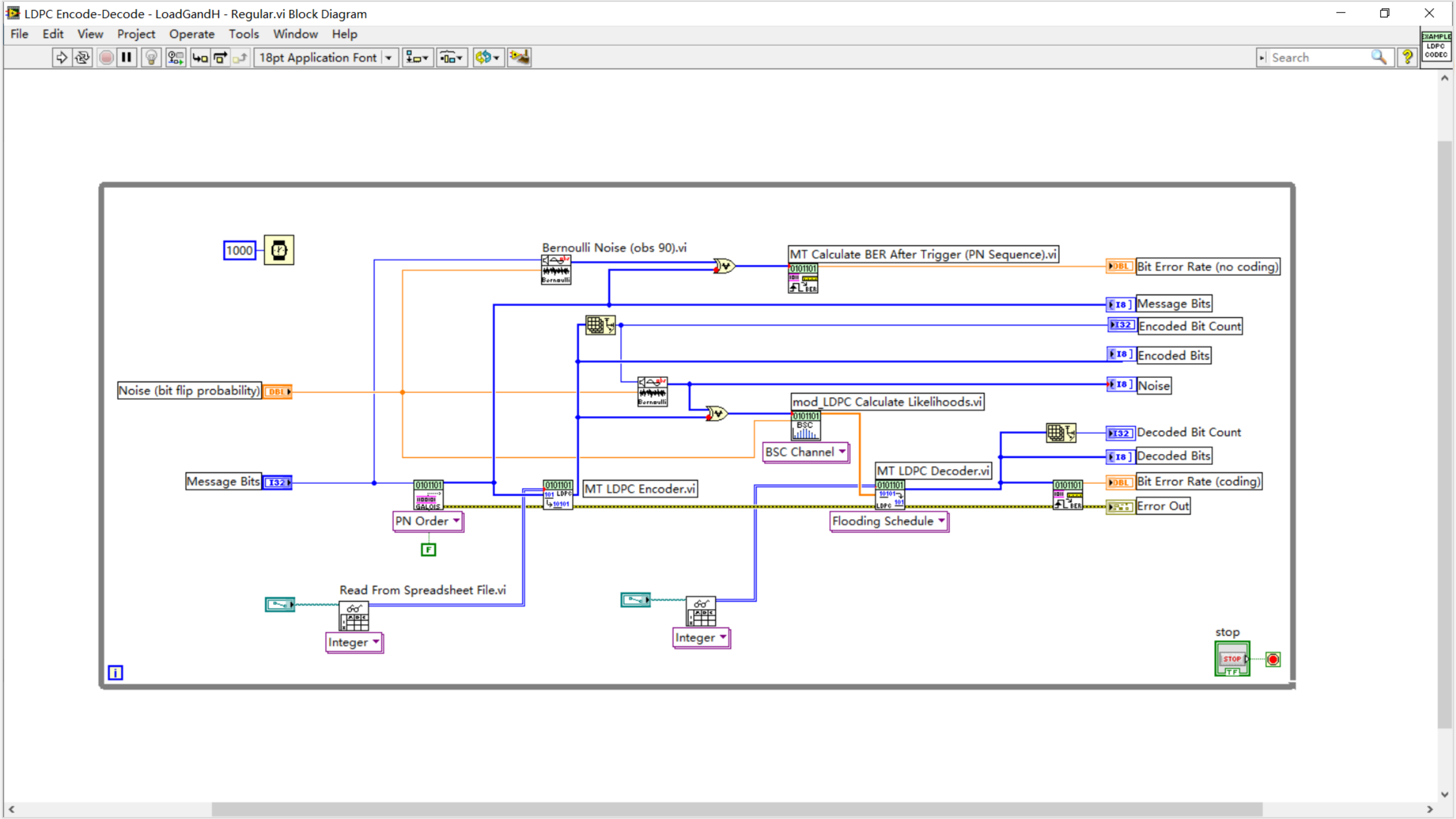


Variable nodes update

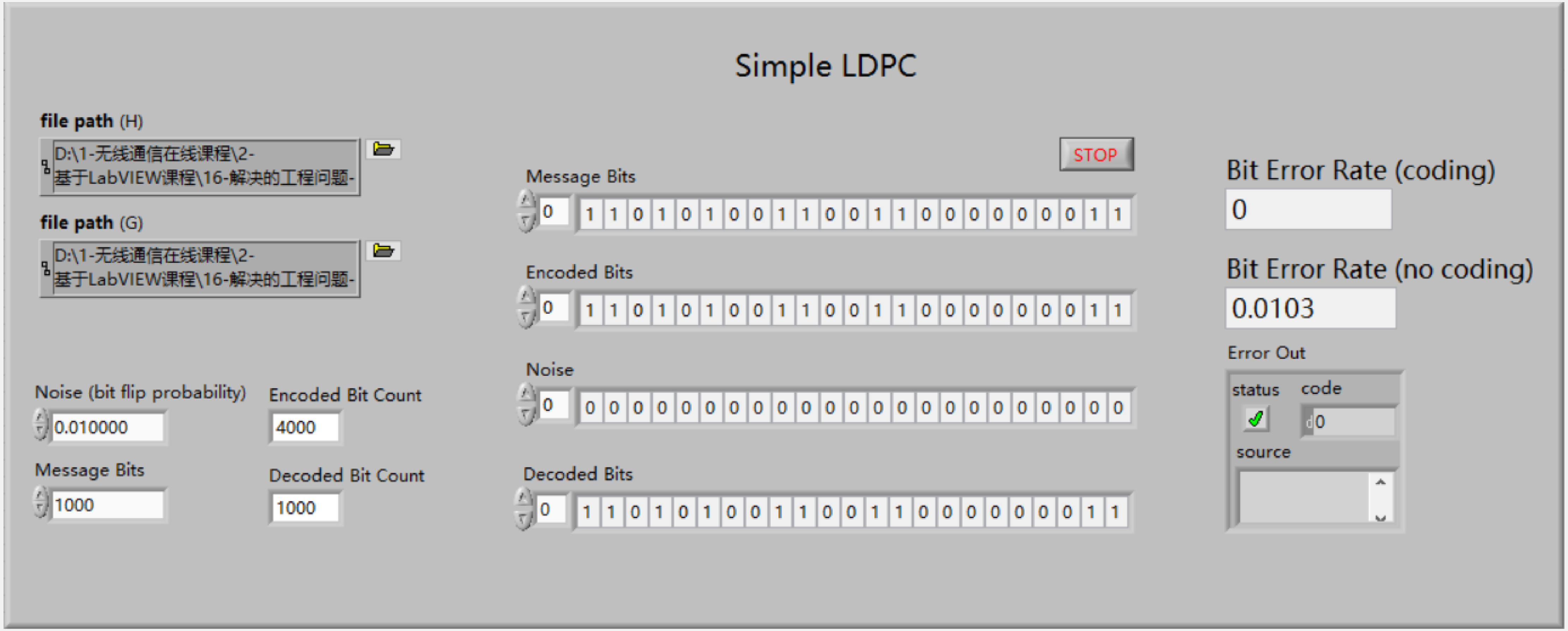
Majority rule:
if vote > 50%, flip
else, hold on

Variable nodes	y_i	messages from check nodes		decision
c_1	1	$f_2 \rightarrow 0$	$f_4 \rightarrow 1$	1
c_2	1	$f_1 \rightarrow 0$	$f_2 \rightarrow 0$	0
c_3	0	$f_2 \rightarrow 1$	$f_3 \rightarrow 0$	0
c_4	1	$f_1 \rightarrow 0$	$f_4 \rightarrow 1$	1
c_5	0	$f_1 \rightarrow 1$	$f_4 \rightarrow 0$	0
c_6	1	$f_2 \rightarrow 0$	$f_3 \rightarrow 1$	1
c_7	0	$f_3 \rightarrow 0$	$f_4 \rightarrow 0$	0
c_8	1	$f_1 \rightarrow 1$	$f_3 \rightarrow 1$	1

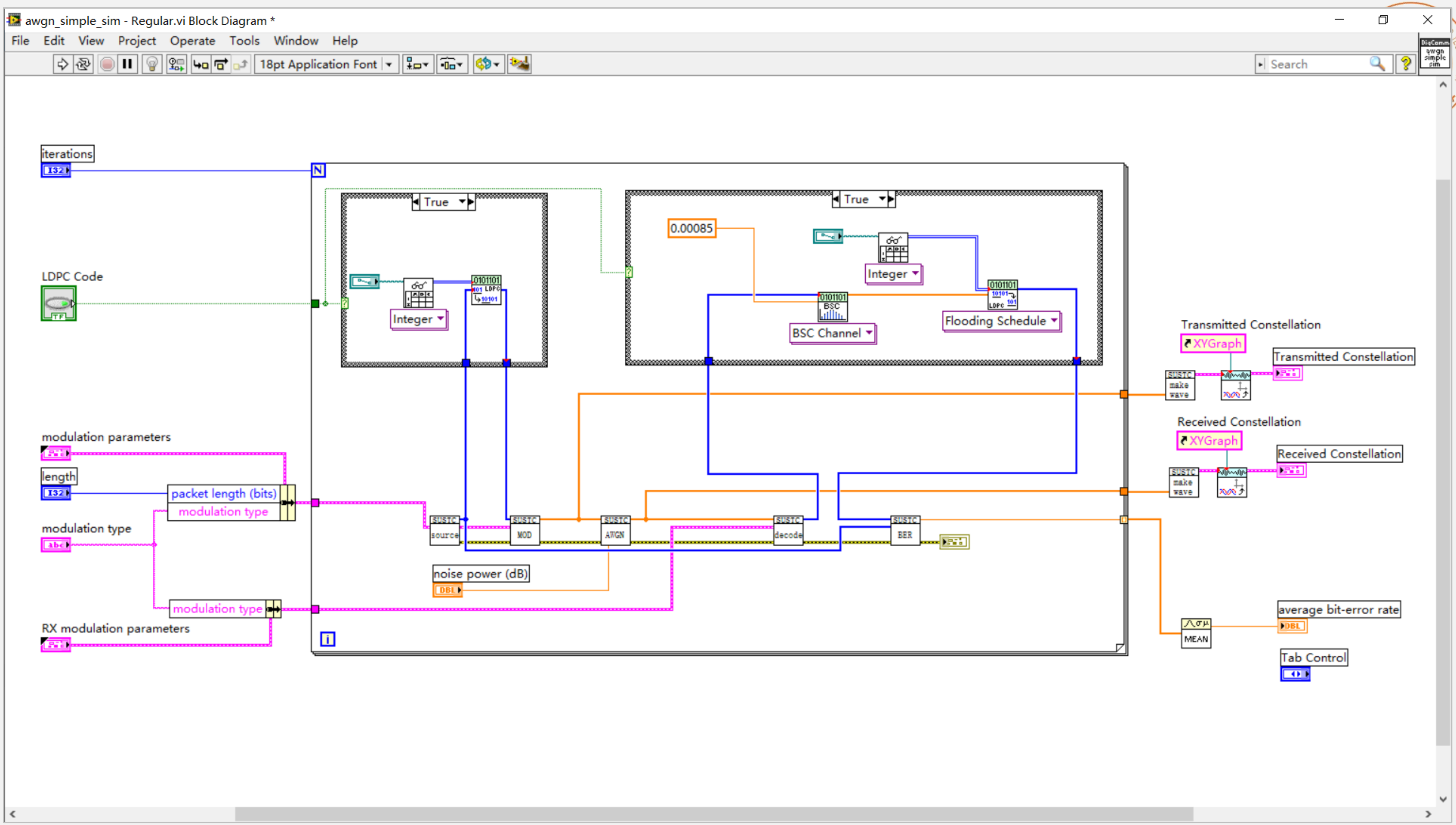
Exercise: Simple LDPC (BSC)



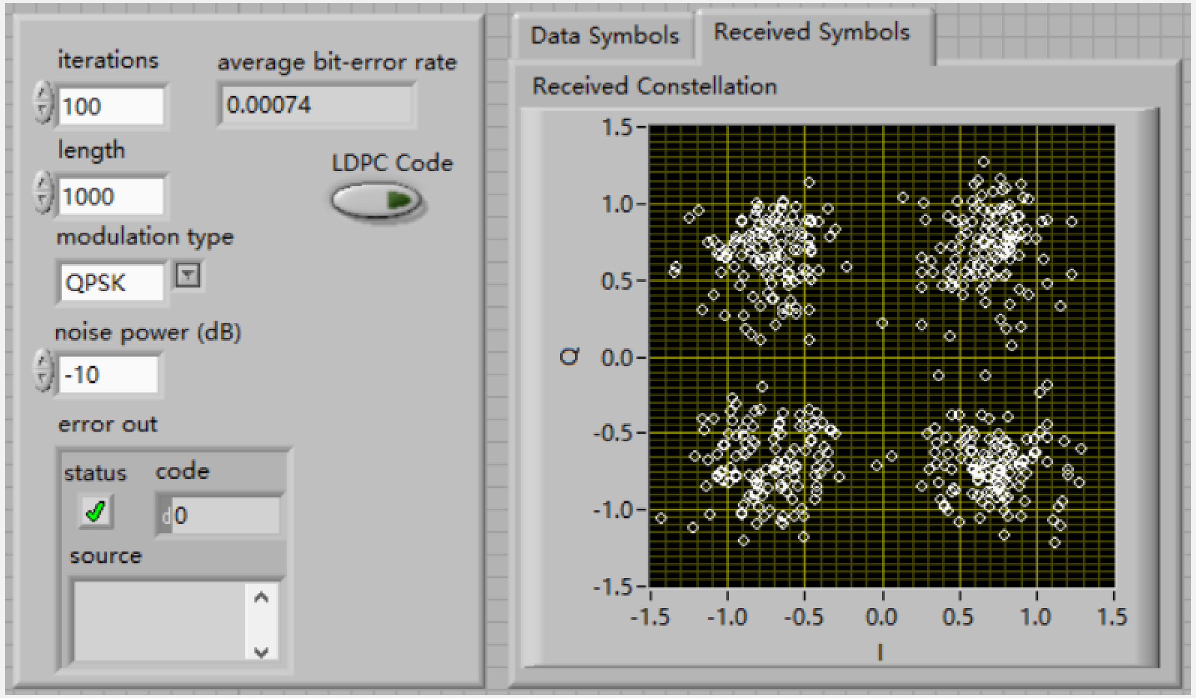
LDPC Encode-Decode - LoadGandH - Regular.vi Block Diagram
File Edit View Project Operate Tools Window Help
18pt Application Font
Search
1000
Bernoulli Noise (obs 90).vi
MT Calculate BER After Trigger (PN Sequence).vi
Bit Error Rate (no coding)
Message Bits
Encoded Bit Count
Encoded Bits
Noise (bit flip probability)
Noise
mod_LDPC Calculate Likelihoods.vi
Decoded Bit Count
BSC Channel
Decoded Bits
MT LDPC Decoder.vi
Message Bits
MT LDPC Encoder.vi
Bit Error Rate (coding)
PN Order
Flooding Schedule
Error Out
Read From Spreadsheet File.vi
Integer
Integer
stop

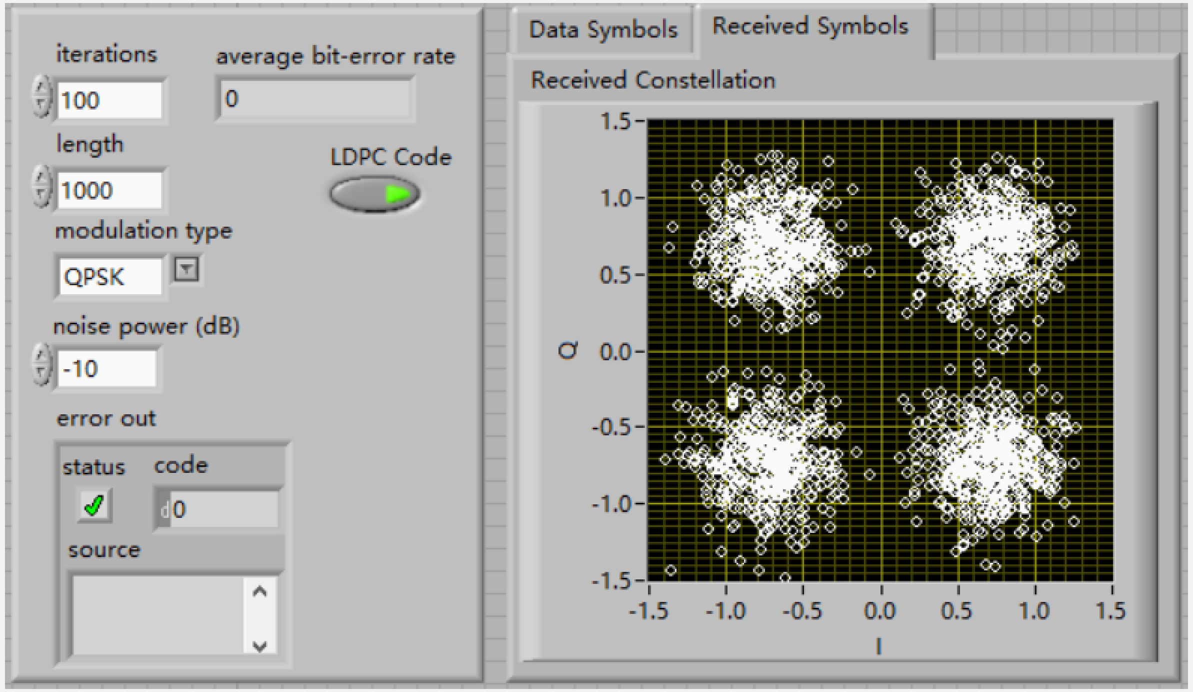

Simple LDPC
file path (H)
D:\1-无线通信在线课程\2-基于LabVIEW课程\16-解决的工程问题-
file path (G)
D:\1-无线通信在线课程\2-基于LabVIEW课程\16-解决的工程问题-
Noise (bit flip probability)
0.010000
Message Bits
1000
Encoded Bit Count
4000
Decoded Bit Count
1000
STOP
Message Bits
0
1 1 0 1 0 1 0 0 1 1 0 0 1 1 0 0 0 0 0 0 0 1 1
Encoded Bits
0
1 1 0 1 0 1 0 0 1 1 0 0 1 1 0 0 0 0 0 0 0 1 1
Noise
0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
Decoded Bits
0
1 1 0 1 0 1 0 0 1 1 0 0 1 1 0 0 0 0 0 0 0 1 1
Bit Error Rate (coding)
0
Bit Error Rate (no coding)
0.0103
Error Out
status
code
0
source

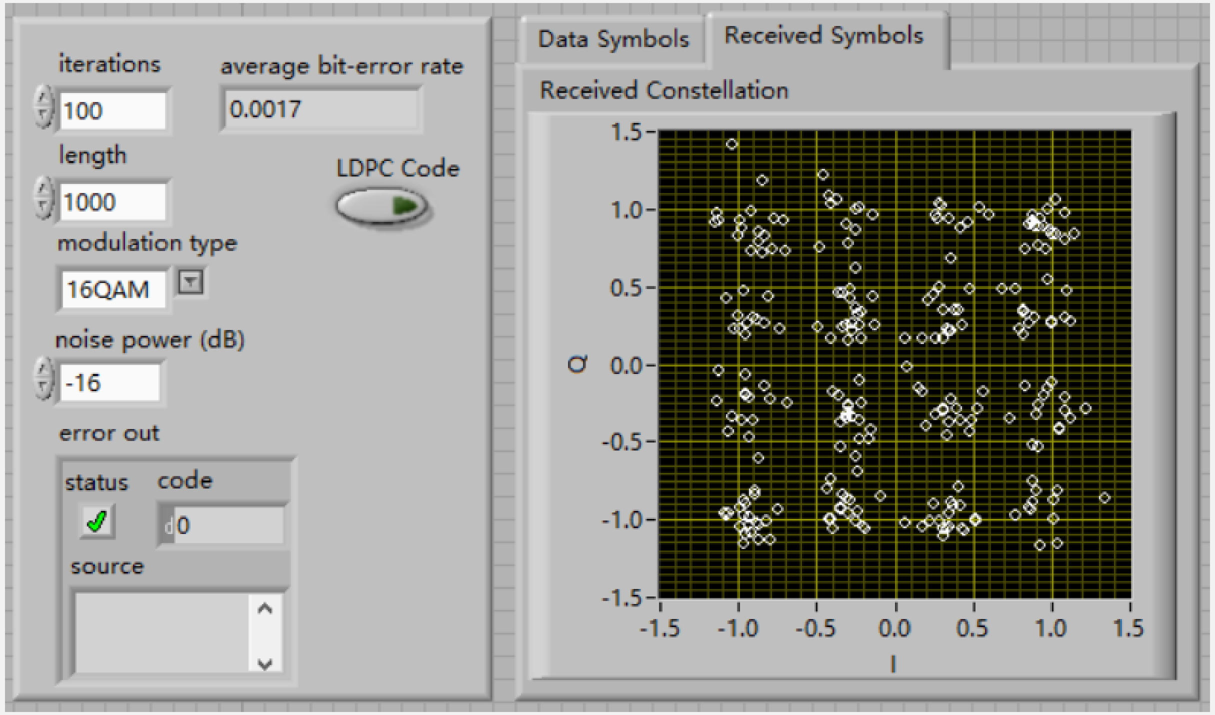
Exercise: Simple LDPC (AWGN)

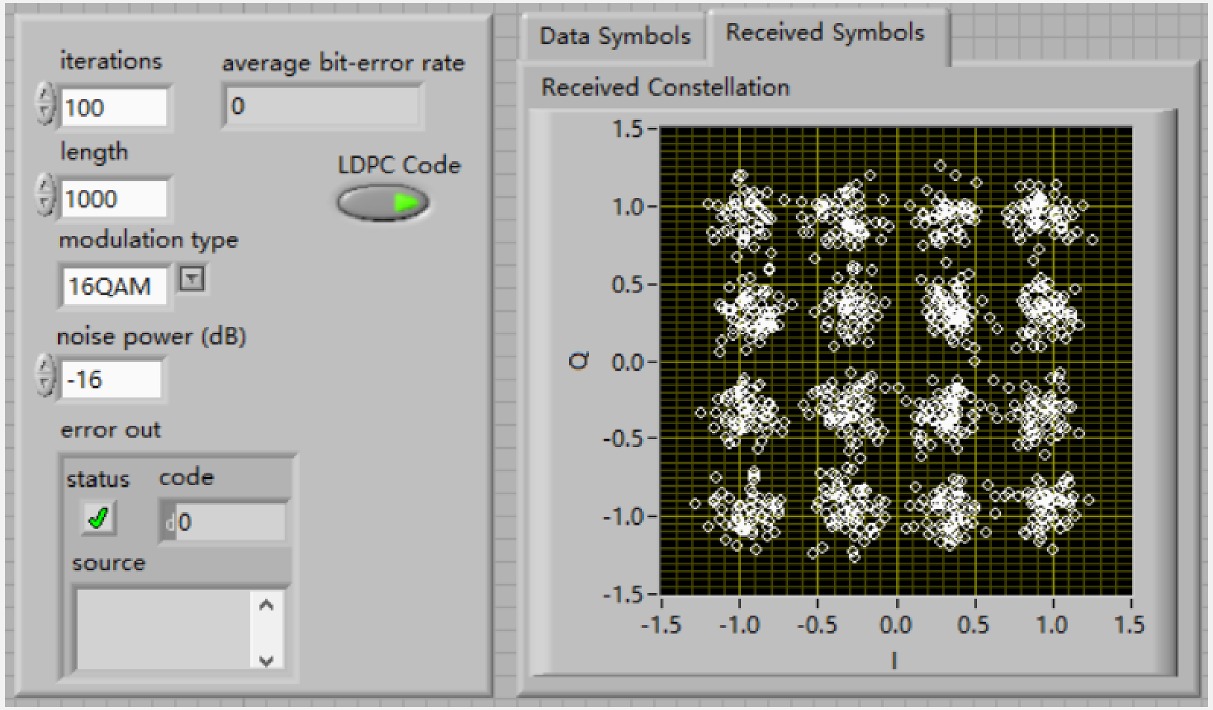


awgn_simple_sim - Regular.vi Block Diagram *
File Edit View Project Operate Tools Window Help
18pt Application Font
Search
iterations
I32
N
True
True
0.00085
LDPC Code
Integer
Integer
BSC Channel
Flooding Schedule
Transmitted Constellation
XYGraph
Transmitted Constellation
Received Constellation
XYGraph
Received Constellation
modulation parameters
length
packet length (bits)
modulation type
modulation type
source
MOD
AWGN
decode
BER
noise power (dB)
DBL
modulation type
RX modulation parameters
make wave
make wave
MEAN
average bit-error rate
DBL
Tab Control


iterations
100
average bit-error rate
0.00074
length
1000
LDPC Code
modulation type
QPSK
noise power (dB)
-10
error out
status
code
0
source
Data Symbols
Received Symbols
Received Constellation
Q
I
1.5
1.0
0.5
0.0
-0.5
-1.0
-1.5


iterations
100
average bit-error rate
0
length
1000
LDPC Code
modulation type
QPSK
noise power (dB)
-10
error out
status
code
0
source
Data Symbols
Received Symbols
Received Constellation
Q
I
1.5
1.0
0.5
0.0
-0.5
-1.0
-1.5


iterations
100
length
1000
modulation type
16QAM
noise power (dB)
-16
error out
status
code
0
source
average bit-error rate
0.0017
LDPC Code
Data Symbols
Received Symbols
Received Constellation
Q
I
1.5
1.0
0.5
0.0
-0.5
-1.0
-1.5


iterations
100
length
1000
modulation type
16QAM
noise power (dB)
-16
error out
status
code
0
source
average bit-error rate
0
LDPC Code
Data Symbols
Received Symbols
Received Constellation
Q
I
1.5
1.0
0.5
0.0
-0.5
-1.0
-1.5

Algorithm Analysis and Verification

1. Hamming Code Simulation
2. LDPC Design and Simulation
3. USRP Verification
4. Analysis and Discussion

- Question ?