Packetized Communications

Sharlene Katz, David Schwartz and James Flynn

Packet Communications

- Asynchronous vs. Synchronous
- Framing
- Zero- or Bit-stuffing
- Why use packets?
- Making a packet
- Kinds of packets
- Basic use of packets
- Some things to think about

Asynchronous vs. Synchronous Digital Transmission

- Asynchronous
  - Uses START and STOP bits.
  - Largely not used due to high overhead.

Start	Data	Stop
0	1 0 1 1 0	1 1
0	0 0 1 0 0	1 1
0	1 1 1 0 1	1 1
0	1 0 1 1 0	1 1

Efficiency =
Data bits / Total Bits =
5/8 = 63%

Asynchronous vs. Synchronous

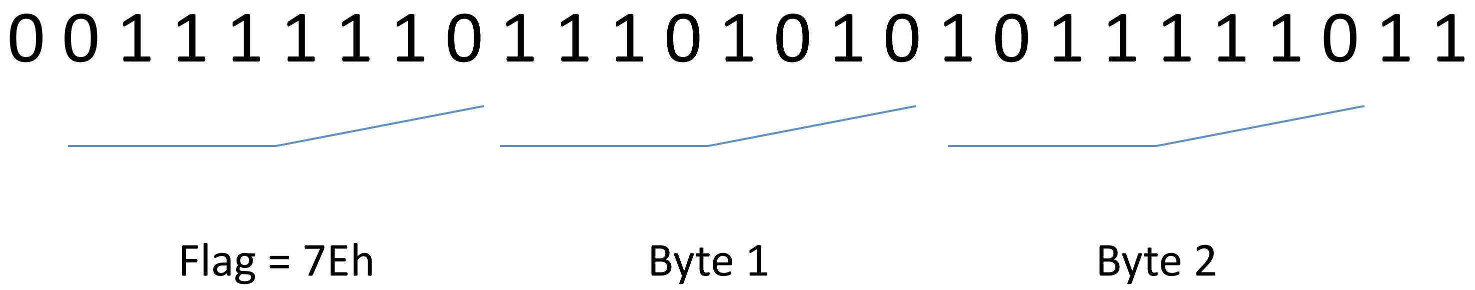
- Synchronous
  - No START or STOP bits.
  - Near 100% efficiency.
  - But how to identify bytes?

…0 1 1 0 1 0 0 1 1 1 1 0 0 1 0 1 0 0 0 0 1 1 0 1 0 1 …

Where are the byte boundaries?

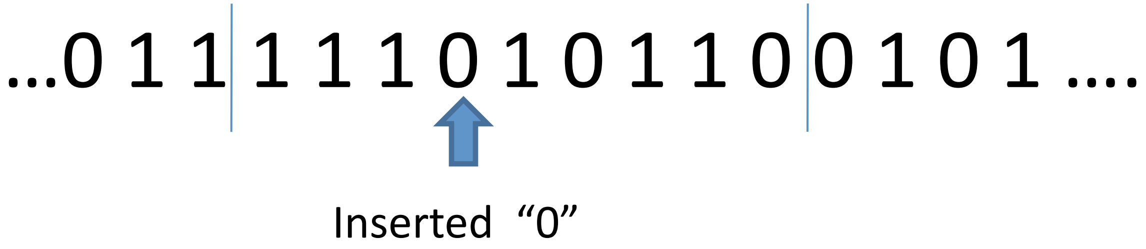
Framing

- Framing identifies bytes using two facts:
  – Bytes are fixed and known length: 8 bits.
  – A special byte or flag is sent to identify start of data.
  – Flag only appears at start or end; never sent otherwise.


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

Framing – zero stuffing

- To avoid sending a flag character at any other time, transmitter uses Zero- or Bit- stuffing.
- After sending five ones, transmitter inserts an extra zero.
- Zero inserted even if all five ones are not in the same byte.



…0 1 1 | 1 1 1 0 1 0 1 1 0 | 0 1 0 1 ….

Inserted “0”

Framing – zero stuffing

- Receiver deletes a zero if it follows five ones.

…0 1 | 1 1 1 1 ~~0~~ 1 0 1 1 | 0 0 1 0 1 …

Recovered stream becomes :

…0 1 | 1 1 1 1 1 0 1 1 | 0 0 1 0 1 …

Could be mistaken for flag without zero stuffing

Why use packets?

- Packets:
  - – divide data into smaller chunks. If errors occur in transmission, then only bad chunk needs to be repeated.
  - – allow error free transmission. Receiver acknowledges good packets, rejects bad ones.
  - – allow addressing and sequencing of data.

Making a packet

- Need to mark beginning and end: use flag bytes.
- Need to say from and to whom: use address field.
- Need to know if packet received okay: use error check byte(s).
- Need to know what packet contains or what it is for: use control and protocol ID.

Making a packet

- AX.25 standard packet

FLAG	ADDRESS	CONTROL & PID	DATA	FCS	FLAG

- Address: packet is to whom, from whom and what number is it in a sequence of packets.
- Control & Protocol ID: what kind of packet is it and what protocol is it using.
- Data: Just that. Variable length. Some kinds of packets contain no data.

Making a Packet (cont’d)

- FCS: Frame Check Sequence, 16 bit number calculated from all bytes before it. Transmitter calculates value based on transmitted data and inserts it here. Receiver calculates value from received bytes. If the values match, the data is accepted. If not, the packet is rejected. All packets have FCS.

Kinds of Packets

- Information or I Frame: Packet with data in it.
- Supervisory or S Frame: Used to control the link between stations.
  - Direction of transmission
  - Reject out of sequence packets
  - Reject bad packets
  - Establish or break communications.
- Unnumbered or U Frame: used to “broadcast” information or establish communications.

Basic Use of Packets

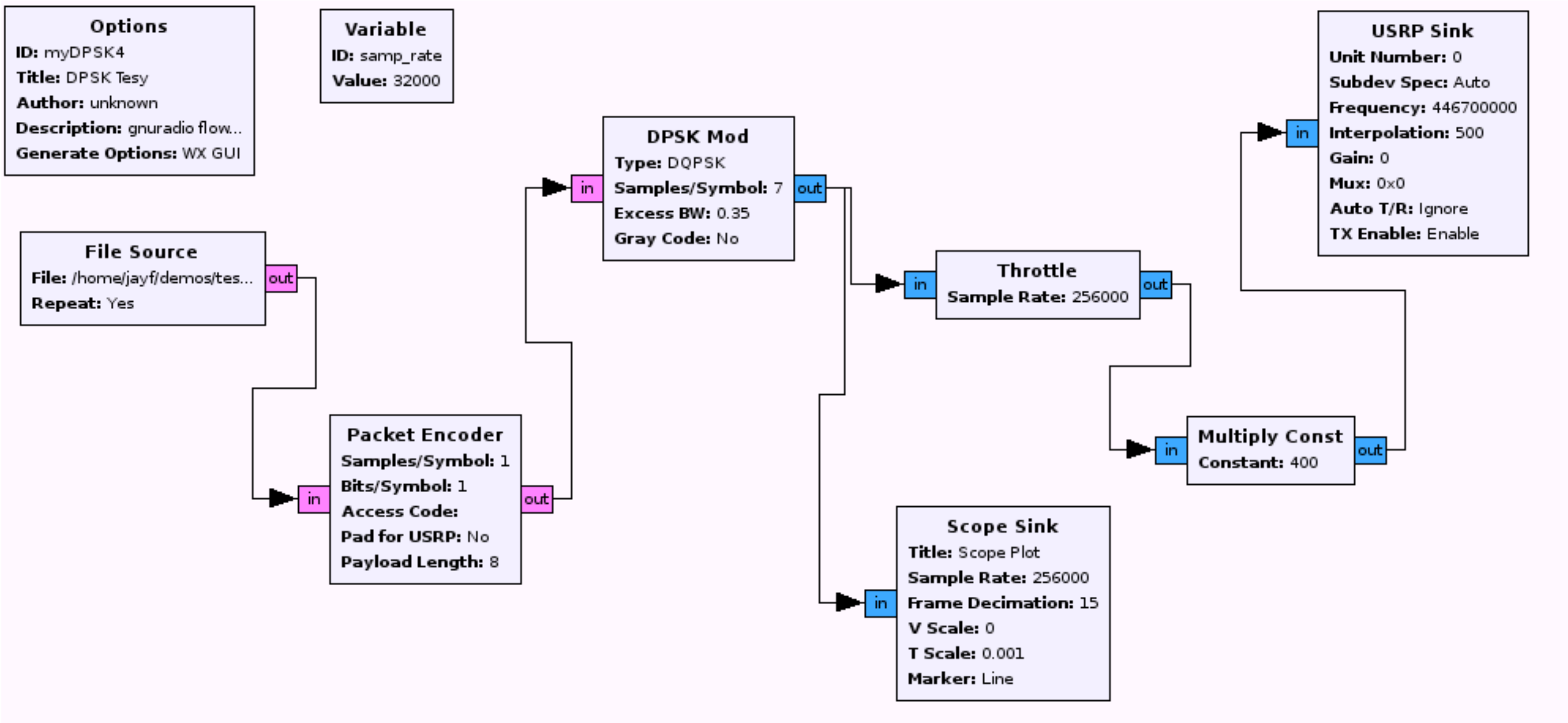
STATION A		STATION B
“Here’s data packet #1. (Data)”		
		“Got data packet #1.”
“Here’s data packet #2. (Data)”	<INTERFERENCE>	
		“Got data packet #1.”
“Here’s data packet #2. (Data)”		
		“Got data packet #2.”
“No more data. Ready to receive.”		
		“Here’s data packet #1. (Data)”
“Got data packet #1.” ….		

Packets – some things to think about:

- There are a number of protocols in use. Each tailored for a particular application or environment.
- Through-put: The “efficiency” of a packet link. I.e. number of information bits / total bits.
- Through-put is a function of packet data length and bit error rate.

Packet Demonstration

- Sending small packets over short path, UHF link.



Options
ID: myDPSK4
Title: DPSK Tesy
Author: unknown
Description: gnuradio flow...
Generate Options: WX GUI

Variable
ID: samp_rate
Value: 32000

File Source
File: /home/jayf/demos/tes...
Repeat: Yes

Packet Encoder
Samples/Symbol: 1
Bits/Symbol: 1
Access Code:
Pad for USRP: No
Payload Length: 8

DPSK Mod
Type: DQPSK
Samples/Symbol: 7
Excess BW: 0.35
Gray Code: No

Throttle
Sample Rate: 256000

Scope Sink
Title: Scope Plot
Sample Rate: 256000
Frame Decimation: 15
V Scale: 0
T Scale: 0.001
Marker: Line

Multiply Const
Constant: 400

USRP Sink
Unit Number: 0
Subdev Spec: Auto
Frequency: 446700000
Interpolation: 500
Gain: 0
Mux: 0x0
Auto T/R: Ignore
TX Enable: Enable

Questions?