EXPERIMENT CM11: MODULATION AND DEMODULATION FOR ASK AND FSK

Related course: KIE2008 (Communication System)

OBJECTIVES:

To investigate the modulation and demodulation for ASK and FSK implemented in MATLAB

EQUIPMENT:

PC with CommLab-T module console and MATLAB, CommLab-T device by Benchmark

REFERENCE(S):

J.G. Proakis, M. Salehi, "Fundamental of Communication Systems", 2nd Ed., Pearson, 2014

TEST:

TEST 1: Amplitude Shift Keying (ASK) modulation and demodulation TEST 2: Frequency Shift Keying (FSK) modulation and demodulation

TEST 1: Amplitude Shift Keying (ASK)

Amplitude-shift keying (ASK) is a form of modulation that represents digital data as variations in the amplitude of a carrier wave. The process involved in ASK modulation is similar to the type of modulation done in the Amplitude Modulation (analog) case except that the message signal is a digital signal.

PRINCIPLE OF OPERATION

i. ASK Modulation

The generated analog message signal is first converted to a digital signal. The digital signal generated is then multiplied by a carrier to generate the amplitude modulated signal (Double-sideband suppressed-carrier (DSB-SC) signal).



Figure 1: Block Diagram of ASK Modulator

ii. ASK Demodulation

Synchronous Detection is used to demodulate the signal. For this, the carrier needs to be generated in the receiver side. The process is similar to Amplitude Modulation.



Figure 2: Block diagram of an ASK demodulator

iii. MATLAB implementation of ASK

At the transmitter, a digital signal (Fig. 3) is generated and then multiplexed with the sinusoidal wave to generate the modulated signal (Fig. 4). The generated signal is given to the CommLAB-T Tx block and is sent through the kit as shown in Fig. 5. At the receiver, samples from the kit are passed through CommLAB-T Rx interface blocks to obtain blocks of data. The signal is then amplitude demodulated using the synchronous detection method to detect the message signal.

CM11



Figure 7: Spectrum of the signal

PROCEDURES:

- 1. Set up CommLAB-T. Refer to Part I of Appendix on how to set up CommLAB-T.
- Generate the transmitter modem samples. Refer to Part II of Appendix on how to generate the modem samples. Choose Amplitude Shift Keying (ASK) Experiment in selecting your experiment (in Step 2, Part II).
- 3. Transmit the modem samples through CommLAB-T. Refer to Part III of Appendix on how to transmit and receive the samples through CommLAB-T.
- 4. Analyze and observe various plots generated by MATLAB.

QUESTIONS:

- 1. Describe all the output figures obtained from the experiment.
- 2. What are the advantages and disadvantages of ASK?

demodulated signal

TEST 2: Frequency Shift Keying (FSK) Modulation

The digital signals generated by PAM, PPM, etc. are called baseband signals as their bandwidth is concentrated around the zero frequency. These signals can be transmitted via a wired medium. However, to transmit these signals via a band pass channel, a mechanism to shift the signal to that band is required. For this purpose, Frequency Modulation is one of the choices and this frequency modulation is called Frequency Shift Keying (FSK).

PRINCIPLE OF OPERATION

i. Modulation

Frequency shift keying (FSK) is a form of modulation that represents digital data as variations in the frequency of a carrier wave. The process involved in this type of modulation is similar to the type of modulation done in the Frequency Modulation (analog) case, except that the message signal is a digital Signal.

- 1. The generated analog message signal is first converted to a digital signal.
- 2. The digital signal generated is modulated using the narrow band frequency modulation technique to generate the FSK signal.



Figure 9: Block diagram of FSK modulator

ii. Demodulation

Improved zero crossing detection is used to demodulate the signal. The process is similar to the one done in Frequency Modulation.



Figure 10: Block diagram of FSK demodulator

iii. MATLAB implementation of FSK

For the transmitter, a digital signal is generated and then frequency modulated before sending it to the CommLAB-T Tx block and to the kit. Fig. 11 shows the transmitter block diagram implemented in MATLAB. For the receiver, samples from the kit are passed through CommLAB-T Rx interface blocks to obtain blocks of data. The signal is then frequency demodulated using the improved zero crossing method to detect the message signal.





Figure 12: Receiver block diagram of FSK in MATLAB



Figure 17: Demodulated signal and its spectrum

PROCEDURES:

- 1. Setup CommLAB-T Refer Part I of Appendix on how to setup CommLAB-T
- Generate the transmitter modem samples. Refer Part II of Appendix on how to generate the modem samples. Choose Frequency Shift Keying (FSK) Experiment in selecting your experiment (in Step 2, Part II)
- 3. Transmit the modem samples through CommLAB-T. Refer Part III of Appendix on how to transmit and receive the samples through CommLAB-T
- 4. Analyze and observe the various plots generated by MATLAB.

QUESTIONS:

- 1. Describe all the output figures obtained from the experiment.
- 2. What are the advantages and disadvantages of FSK?

APPENDIX

Part I: Setup CommLAB-T

- 1. Double click on the CommLab-T icon on the desktop to activate the software. Wait until the CommLab-T Console appears. Leave this window active and do not close it. Also make sure that the CommLab-T device is switched ON.
- 2. On this Comm:LAB-T console (Fig. 18), click the 'INITIALIZE' button to initialize CommLAB-T. Press the RESET button once on the CommLab-T device (Fig. 19).
- 3. On the new window as shown in Fig.20, select Options and then EZ-USB Interface. The CommLab-T initialization window will appear, as shown in Fig. 21. Click download and choose WBU.hex file (Fig. 22).
- 4. After downloading is completed, you can exit the EZ-USB interface window. In the Cypress USB Console window, select 'Alt setting' as '2' in the 'Configuration Interfaces field' as highlighted in Fig. 23. You can then close this window (Fig. 23).
- 5. Press 'RUN' button from the CommLAB-T console (refer Fig. 18). This will open the WBU console as shown in Fig. 24. WBU console is used to transmit and receive modem samples through CommLAB-T. Choose the 2MBPS sampling rate using the sampling rate pull down menu as shown, press icon label 1 and make sure that the "Tx & Rx" is selected under 'Direction' and click 'OK' (Refer Fig. 25). Now CommLAB-T is ready to transmit and receive the samples.

Notes: Do not change any other fields or configuration options. Changing them will result in improper functioning of the system. Results obtained for CommLAB-T may not be correct if sampling rate is chosen as any other value except 2MBPS.



Figure 18: CommLab-T Console

File Options Help Image: EZ-USB Interface Image: EZ-USB Interface Image: EZ-USB Interface Selec Show EZ UI on Startup Image: EZ-USB Interface Image: EZ-USB Interface USE Verbose Output Image: EZ-USB Interface Image: EZ-USB Interface Image: EZ-USB Interface Device Properties Control Endpt Xiers Other Endpt Xiers Misc. Image: EZ-USB Interface VerdorID	🕤 Сур	ress USB Con	sole					
EZ-USB Interface X @ C # Selec Show EZ UI on Startup USE Verbose Output Interface Name in Windows Device Mgr (from.inf) Cypress EZ:USB PX2LP No EEPROM(3. Device Properties Control Endpt Xiers Other Endpt Xiers Misc. VendorID 0x8613 Subclass 0xFF Product/D 0x8613 Subclass 0xFF Product/D 0x8613 Subclass 0xFF Product/D 0x8613 Subclass 0xFF Product/D 0x8613 Subclass 0xFF Product wXITBENCHMARK Protocol 0xFF Device Configurations (1) Value Value Attributes Max Power [0x01 0x80 0x32(100 mA) Configuration Interfaces (4) Endplayed for the faces (4)	File Options Help							
Selec Show EZ UI on Startup USE Verbose Output Name in Windows Device Mgr (from .inf) Cyprest E2/USB PX2LP No EEPROM(3 Device Properties Control Endpt Xiers Other Endpt Xiers Misc. VendorID 0x0484 Class 0xFF ProductID 0x6613 Subclass. 0xFF ProductID 0x6613 Subclass. 0xFF Product. 0xVITBENCHMARK Protocol 0xFF Device Configurations (1) Value Value Attributes 0x30 0x32 (100 mA) Configuration Interfaces (4)	è	EZ-USB I	interface 🛛 🔫		🗶 🛞 😷 👂	7		
USE Verbose Output Name in Windows Device Mgr (from .inf) Cyperst EZUSB FX2LP No EEPROM(3) Device Properties Control Endpt Xiers Other Endpt Xiers VendorID	Selec	Show EZ	UI on Startup					
1 Cypress E2:USB PX2LP No EEPROM(3.) Device Properties Control Endpt Xfers Other Endpt Xfers Misc. VendotD 0x0484 Class 0xFF ProductD 0x6613 Subclass. 0xFF Mandacturer EBNCHMARK Protocol0xFF 0x6Device 0x4001 Setial Number VXITEENCHMARK bedDevice 0x4001 Setial Number V Device Configurations (1) Value	USE	✓ Verbose	Output	Name in Wir	ndows Device Mgr (from .inf)			
Device Properties Control Endpt Xiers Other Endpt Xiers Misc. VendorID 0x0484 Class 0xFF ProductID 0x6613 Subclass 0xFF Manufacturer BENCHMARK Protocol 0xFF Product WRITBENCHMARK bedDevice 0x4001 Serial Number ? 2 2 Device Configurations (1) 1 1 1 Value _Attributes Max Power 1 1 Qx01 0x80 0x32 (100 mA) 1 1	1 Cypress EZ-USB FX2LP No EEPROM(3							
Device Properties Control Endpt Xfers Other Endpt Xfers Misc. VendorlD 0x0484 Class 0xFF ProductID 0x6613 Subclass 0xFF Manufacturer BENCHMARK Protocol 0xFF Product. vxIIIBENCHMARK bcdDevice 0xA001 Serial Number ? 2 Device Configurations (1) Value Attributes Max Power [0x01 0x80 0x32 (100 mA) 2 Configuration Interfaces (4) 2 2								
Device Propeties Control Endpt Xiers Other Endpt Xiers Misc. VendorlD 0x0484 Class 0xFF ProductID 0x0613 Subclass 0xFF Manufacturer DENCHMARK Protocol 0xFF Product.								
VendotD0x0484 Class0xFF ProductD0x6613 Subclass0xFF Manufacturer0xFF Protocol0xFF ProductvKITBENCHMARK Protocol0xFF Setial Number? Postocol0xFF Device Configurations (1) Value Value Attributes Max Power 0x01 0x80 0x32(100 mA) Configuration Interfaces (4)	Device Properties Control Endpt Xfers Other Endpt Xfers Misc.							
VendorID 0x0484 Class 0xFF ProductID 0x0613 Subclass 0xFF Manufacturer BENCHMARK Protocol. 0xFF Product								
ProductID	Ven	dorlD (0x04B4	Class	Class 0xFF			
Manufacturer	Proc	iuctID (0x8613	Subcl	assOxFF			
Voluce Attributes Max Power Dx01 0x80 0x32(100 mA) Configuration Interfaces (4)	Manufacturer BENCHMARK Protocol UxFF Device 0x4001							
Device Configurations (1) Value Attributes Max Power 0x01 0x80 0x32 (100 mA) Configuration Interfaces (4)	Serial Number 2							
Device Configurations (1) Value Attributes 0x01 0x80 0x01 0x80 0x01 0x80 0x01 0x32 (100 mA)								
Value Attributes Max Power 0x01 0x80 0x32 (100 mA) Configuration Interfaces (4)	Device Configurations (1)							
0x01 0x80 0x32(100 mA) Configuration Interfaces (4)	Value		Attributes	Max Power				
Configuration Interfaces (4)	0x01		0x80	0x32 (100 mA)				
Conliguration Interfaces (4)	Configuration Interfaces (A)							
	Conrig	Juration Interrat	88 (4) Col	1011				
Intro Air Setting Liass Subclass Protocol	Intro	Ait Setting	Ulass 0.55 (Mandard)	SUDCIASS				
0 0 0xFF (Vendor) 0xFF 0xFF	0	1	OxFF (Vendor)	0xFF 0xFF	0xFF			
0 2 0xFF (Vendor) 0xFF 0xFF T	Ő	2	0xFF (Vendor)	0xFF	0xFF 🔻			

Figure 20: Cyprus USB console



Figure 19: CommLab-T device

😙 Cypress USB Console	_ 🗆 🗙
File Options Help	
	• • • • • • • • • • • • • • • • • • •
Device WKITBENCHMARK	*ROM Sweet YON mon-ext-sio1-c0.hex
Get Dev Get Conf Get Pipes Get Strings Dounload Rolload Lg EE	PROM URB Stat HOLD RUN
Vand Rag Reg 0x00 Volue 0x0000 Index 0x0000 Length 0 Di	0 OUT • Hex Dyles C0 B4 04 81 00 01 00 •
Iso frank Pipe Packet Size	Packets
But trans Pipe Elength 64 Hex Bytes	5
Reset Ripe Abort Ripe File Trans File 9	
Set (Face Interface III AtSetting III	
	×
	×
	4! - I! 4!

Figure 21: CommLAB-T Initialisation window, EZ– USB Interface

- 6. Click icon labelled "2" in Fig. 24 which will open the statistics window (Fig. 26)
- 7. Click 'EXPERIMENT' button in CommLAB-T console (Fig. 18) to activate the CommLABT_EXPERIMENT Console (Fig. 27).



Figure 26: The statistics window

Figure 27: CommLab-T Experiment console

Part II: Generating the Samples

- 1. In experiment console (Fig. 27) select the Experiment to be run from the EXPERIMENT pull down menu. Upon selecting the experiment, the parameters that can be changed for transmitter and receiver will appear below the experiment as a pull down menu. If any parameters related to that experiment are to be changed, choose the available required value from the pull down menu.
- 2. Choose the type of experiment.
- 3. Click the 'GENERATE' button, which will generate the modem samples to be transmitted for the chosen experiment. This generated modem samples have to be transmitted and received through CommLAB-T using WBU console. Save all the figures that are output during this step.

Part III: Transmit and Receive the Samples

- Press START button in WBU console (Fig. 26) to start transmitting and receiving the modem samples through CommLAB-T. The Tx icon and the Rx icon in the WBU console will start blinking to indicate that CommLAB-T is transmitting and receiving properly. This can be ensured by looking at the statistics window (Fig. 26). If the transmission and reception are successful, then this field will keep incrementing and the Failed Packets field shows zero.
- 2. After around 50,000 packets are successfully received, press the STOP button to stop transmitting and receiving the samples. Exceeding too far beyond 50,000 may cause insufficient memory problems.
- 3. Whenever there is a link failure between Tx and Rx, the 'Failed Packets' field for Tx and Rx show non-zero values and the Tx and Rx icons will start blinking in Red. The 'Xfer Rate KB/s' field shows the sampling rate of the modem samples for Tx and Rx.
- 4. Click the ANALYZE button on the CommLab-T console (Fig. 27). Save the figures generated from the experiment.

END OF EXPERIMENT

7