Tiva Network Tester

Project Titles: Integrating Tiva C Series Launchpad with Ethernet card for Network communication realization.

Group Members:

Name: Pallavi Kulkarni, Taylor Artunian

Email ID: pkulkarni1@mail.sfsu.edu, tartunian@mail.sfsu.edu

Abstract:

Intention of this project is to establish **communication** between a Computer and a Tiva C series Launch Pad in a **network** using **LAN**. The complete set up comprises of **Tiva C Series Launch Pad**, **Ethernet Breakout board - ENC28J60**, a **computer** and a **router** for network connectivity. Interfacing the Tiva C Series Launch Pad to an Ethernet Breakout board requires it to connect to a router through the Ethernet Breakout board on one side and a computer on the other side.

Interfacing ENC28J60 Ethernet Module with Tiva Launch Pad gives us the freedom of using the internet on Tiva Launch Pad.

Introduction:

Traditionally in low-cost microcontroller systems, simple communication protocols like RS-232 or RS-485 are used. Although simple to implement, they are characterized by low speed, short distance (for RS-232) and lack of communication and synchronization protocols. CAN is more advanced, but more complicated to implement. It is nondestructive bitwise arbitration protocol imposes hard limit on speed-length product. E.g., maximum CAN speed at 40 m link is 1Mbit/s.

Ethernet offers numerous advantages. Data rate of 100 Mbit/s or more is easily achieved, distance is practically unlimited, and standardization and widespread support makes communication possible with a broad spectrum of systems and devices. Implementation complexity could be overcome by using a specialized Ethernet controller chip, which implements Physical and Data Link layers of OSI model.

The ENC28J60 Ethernet Module:

The ENC28J60 Ethernet Module, used in our project, utilizes the Microchip ENC28J60 Stand-Alone Ethernet Controller IC. It features a host of features to handle most of the network protocol requirements. The board connects directly to most microcontrollers with a standard SPI interface with a transfer speed of up to 20MHz.

This Ethernet Breakout-Module is the simplest way to add LAN connectivity to a microcontroller-based product and projects.

- ✓ Use of this module enables an Ethernet interface. It works with any microcontroller operating at 3.3V or 5V.
- ✓ This module works at 3.3V and is compatible with 5V interface lines.
- ✓ Uses the SPI process to interface with it.
- ✓ Able to Host web server, ping the module or add it to home automation via internet.
- ✓ The heart of this module is the ENC28J60 Ethernet controller from Microchip.
- ✓ The use of RJ45 with Integrated magnetics has made it possible to reduce the size of the board.

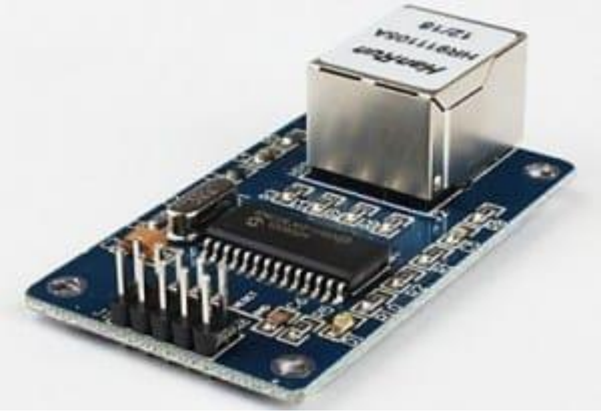


Fig 1. ENC28J60 Ethernet Module

Features:

- ✓ Ethernet LAN Module for various microcontroller boards like Tiva/Arduino/LPC/STM3
- ✓ ENC28J60 Ethernet chips
- ✓ Can be easily interfaced with the MCU
- ✓ Supply Voltage: 3.3 V (5V Tolerant DIO)
- ✓ 25Mhz crystal oscillator
- ✓ Size (L x W x H): Approx. 2.3 x 1.3 x 0.7 inch / 58 x 34 x 17 mm

TM4C123G LaunchPad:

The above-mentioned Ethernet module is interfaced, in our project, to TI’s Tiva™ C Series TM4C123G LaunchPad Evaluation Board (EK-TM4C123GXL) which is a low-cost evaluation platform for ARM® Cortex™-M4F-based microcontrollers. The Tiva C Series LaunchPad design highlights the TM4C123GH6PMI microcontroller USB 2.0 device interface, hibernation module, and motion control pulse-width modulator (MC PWM) module. The Tiva C Series LaunchPad also features programmable user buttons and an RGB LED for custom applications. The stackable headers of the Tiva C Series TM4C123G LaunchPad BoosterPack XL interface demonstrates how easy it is to expand the functionality of the Tiva C Series LaunchPad when interfacing to other peripherals on many existing BoosterPack add-on boards as well as future products. Fig 2. shows a photo of the Tiva C Series LaunchPad.

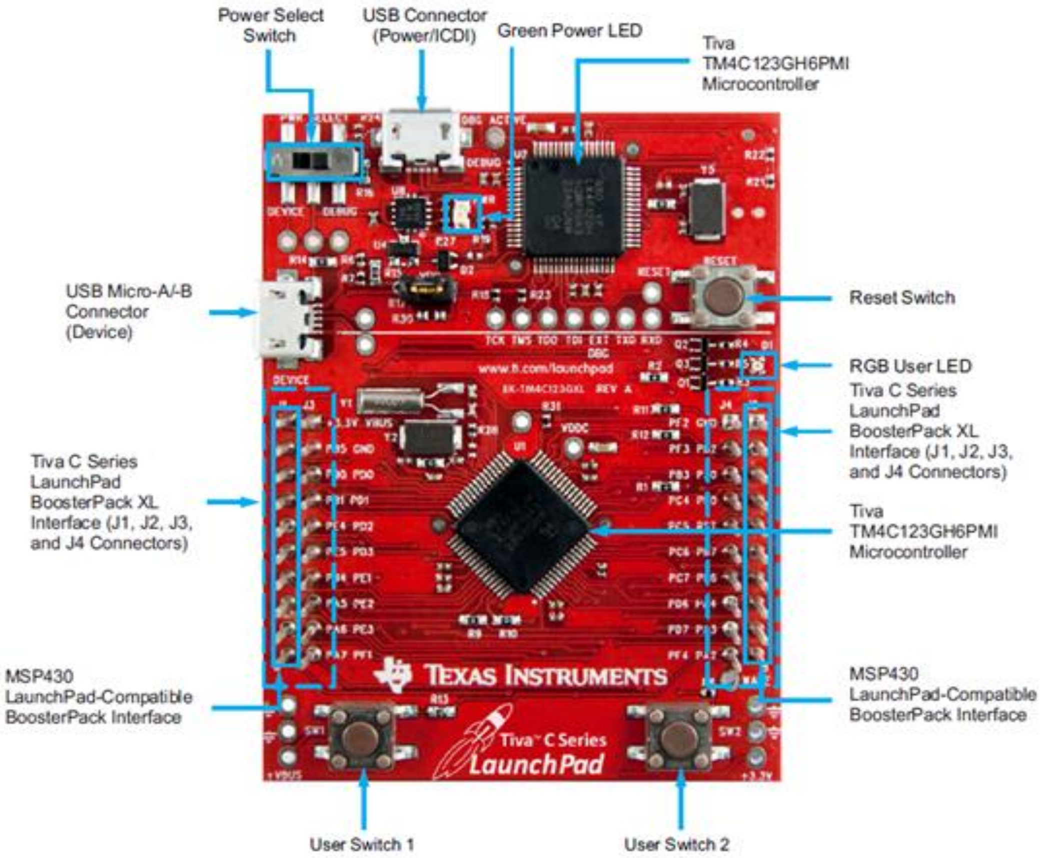

Power Select Switch
USB Connector (Power/ICDI)
Green Power LED
Tiva TM4C123GH6PMI Microcontroller
USB Micro-A/-B Connector (Device)
Reset Switch
RGB User LED
Tiva C Series LaunchPad BoosterPack XL Interface (J1, J2, J3, and J4 Connectors)
Tiva C Series LaunchPad BoosterPack XL Interface (J1, J2, J3, and J4 Connectors)
Tiva TM4C123GH6PMI Microcontroller
MSP430 LaunchPad-Compatible BoosterPack Interface
MSP430 LaunchPad-Compatible BoosterPack Interface
User Switch 1
User Switch 2

Fig 2. Tiva C Series TM4C123G LaunchPad Evaluation Board

Features

The Tiva C Series LaunchPad includes the following features:

- Tiva TM4C123GH6PMI microcontroller
- Motion control PWM
- USB micro-A and micro-B connector for USB device, host, and on-the-go (OTG) connectivity
- RGB user LED
- Two user switches (application/wake)
- Available I/O brought out to headers on a 0.1-in (2.54-mm) grid
- On-board ICDI
- Switch-selectable power sources:
  - ICDI
  - USB device
- Reset switch
- Preloaded RGB quickstart application
- Supported by TivaWare for C Series software including the USB library and the peripheral driver library

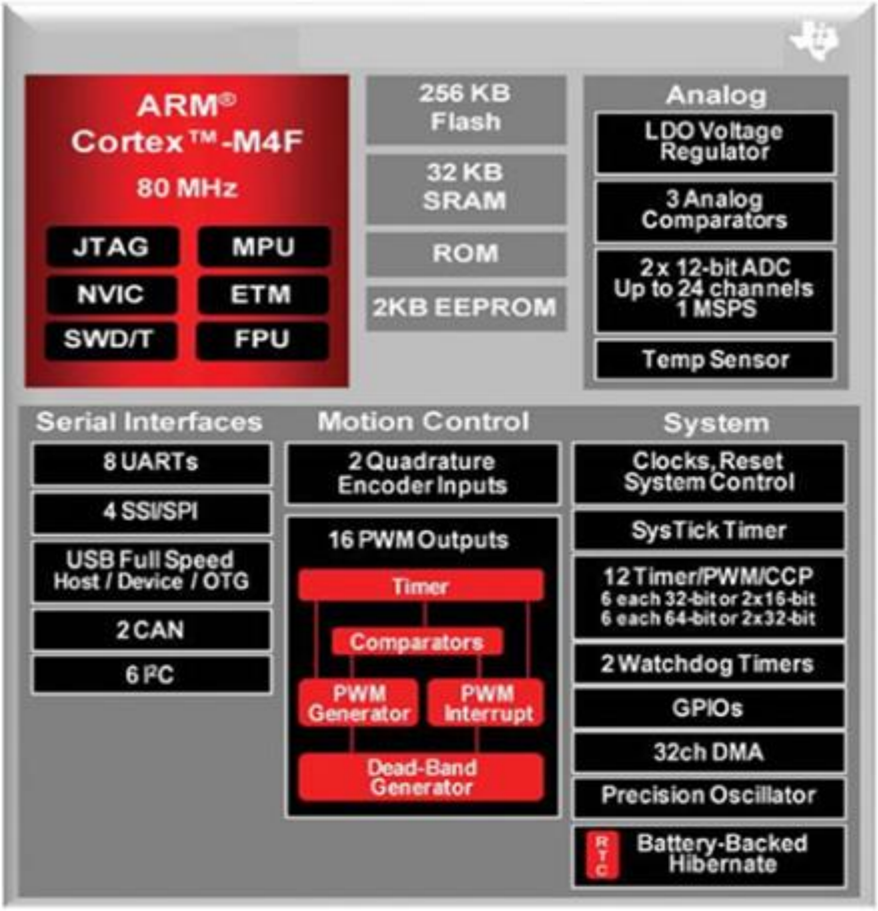


Fig 3. Tiva Core and FPU

Among the interfaces that the Tiva board provides, as shown in Fig. 7, we are using the following interfaces.

Tiva Peripherals/Modules Used:

- Interrupts
  - For handling the events.
- EEPROM
  - For storing the ethernet configurations, and commands implemented.
- UART
  - For displaying all the interactions between all the devices involved as well as the user.
- SPI
  - It is used for connecting Tiva board to the ethernet card.

ENC28J60 Ethernet Module is interfaced using Tiva's SPI interface. Interfacing ENC28J60 Ethernet Module with Tiva Launch Pad provisions usage of the internet on Tiva Launch Pad.

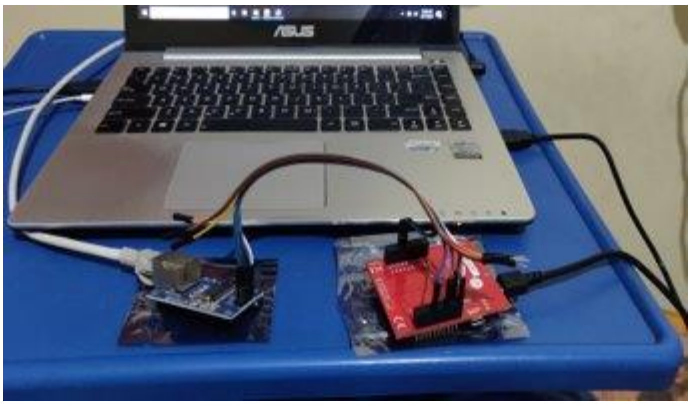


Fig 4. Physical Setup

Block Diagram: The block diagram is self-explanatory which gives the idea of a Tiva Launch Pad & ENC28J60 module interfaced with Router and Computer for communicating over the network.

Target uC: TM4C123GH6PM **System Clock:** 40 MHz

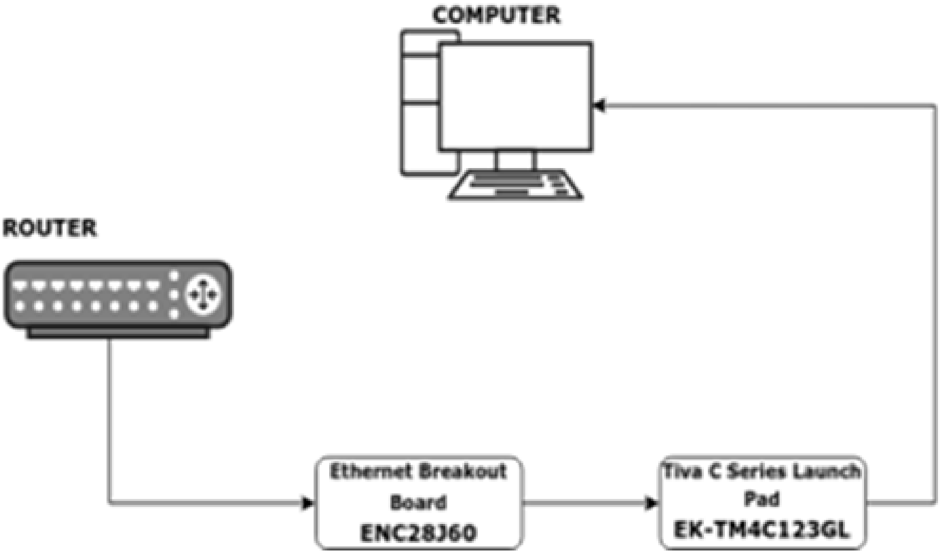

COMPUTER
ROUTER
Ethernet Breakout Board ENC28J60
Tiva C Series Launch Pad EK-TM4C123GL

Fig 5: Block Diagram

Connections: The Router should be LAN connected and should have multiple input-output ports. From the router, one Ethernet cable should be connected to **an ENC28J60 Module** that in turn is connected to the **Tiva Launch Pad**. The **Tiva Launch Pad** is directly connected to **Computer,** as shown in the block diagram.

The **connection between Tiva Launch Pad & ENC28J60** is as given below.

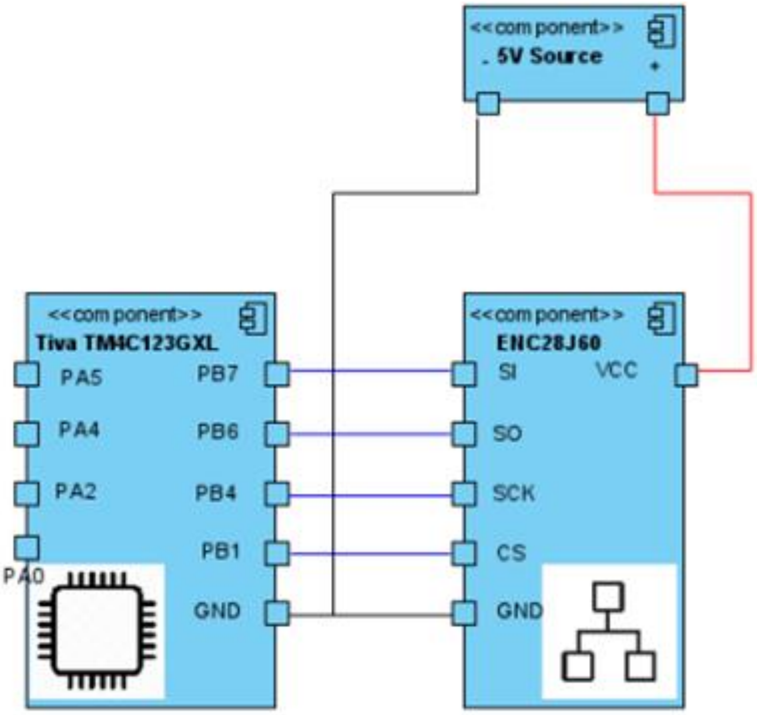


Fig 6: Connection between Tiva launch pad and two Ethernet breakout boards

Proposed Networking Features for realization:

- Ping
  - The Tiva will respond to ICMP requests
- Man-in-the-middle Ethernet packet sniffer
  - The device can be connected inline between two devices to monitor traffic. As an example, we will collect MAC addresses or IP addresses.
- Network uptime report generator
  - The Tiva will ping a particular IP and record the responses or lack of responses to create an uptime report.
- MAC address filtering avoidance via brute force testing of addresses (optional)
  - The Tiva will attempt to find MAC addresses which are allowed through a firewall's MAC filter
- Bandwidth/speed test (optional)
  - Tiva will attempt to download a static file from the web and use this to determine the network speed. The upper limit will be determined by the speed of the Tiva.

Expected Outcome:

We had proposed to be able to demonstrate Communication between a computer and a Tiva Launch Pad over the network using the Ethernet Breakout Board by pinging, uptime report generation and sniffing.

Might be able to demonstrate network speed and MAC address filtering if time and research allow.

Development:

We began with a basic ENC28J60 driver written for the Tiva TM4C from “nihit30” on GitHub. Below are some of the changes/improvements we made:

- Created separate buffers for RX and TX packets
- Swapped out constants for sizeof
- Moved the “check for new packet” logic from while loop to timed interrupt
- Created structs for common headers within a packet. These can be used to typecast the packet.
    - Ethernet
    - IP
    - ARP
    - ICMP
    - UDP

Realized Outcome:

- Ping (bidirectional)
- ARP
- Configuration saved in persistent memory (EEPROM)
- System uptime
- Set IP, gateway and subnet addresses
- Ipconfig
- Printing ethernet frames in raw format
- Display received UDP packet (broadcasted)

System Flow:

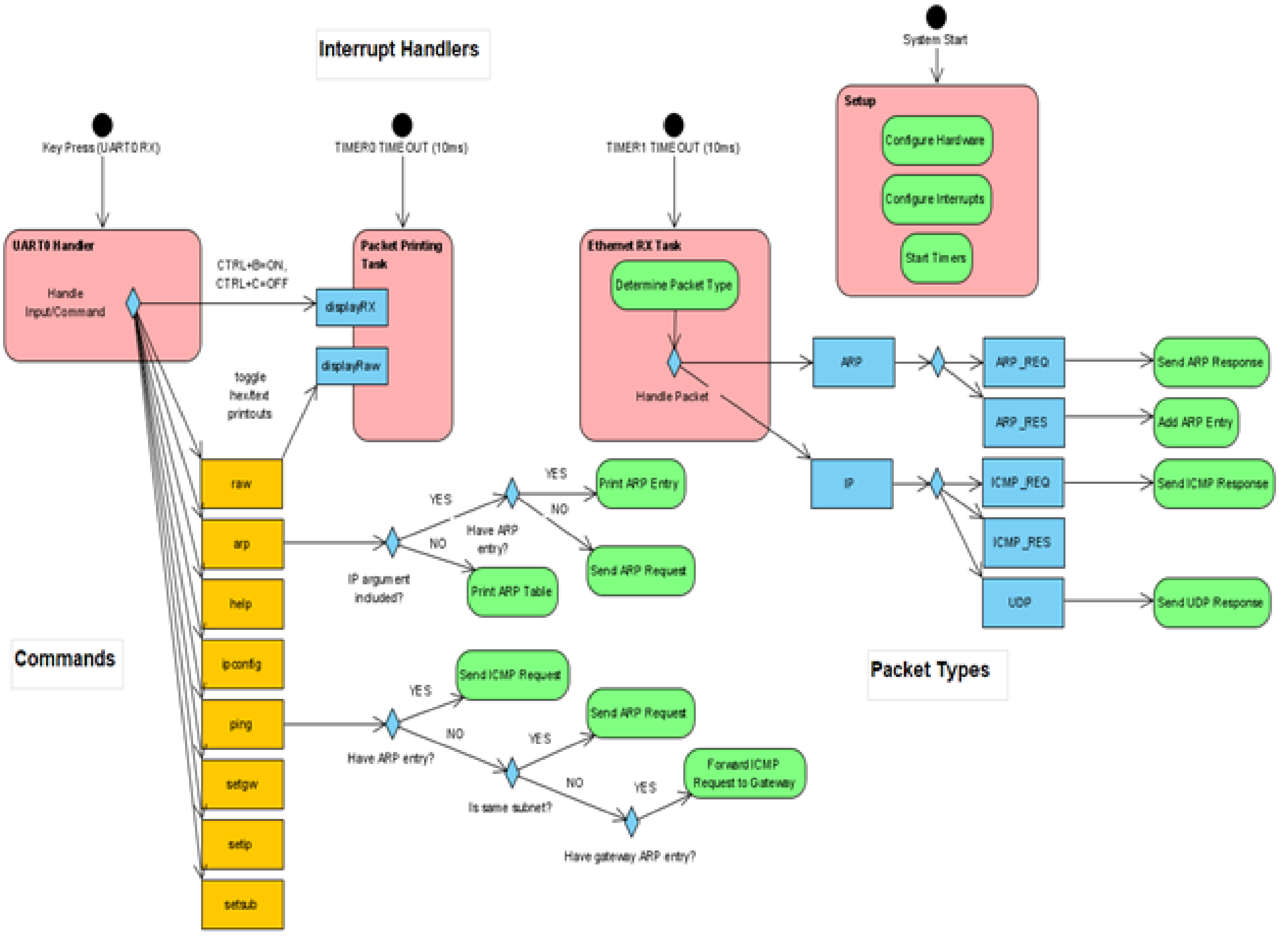


Fig 7: Implementation Flow Chart

ENC28J60 Driver API:

Types:

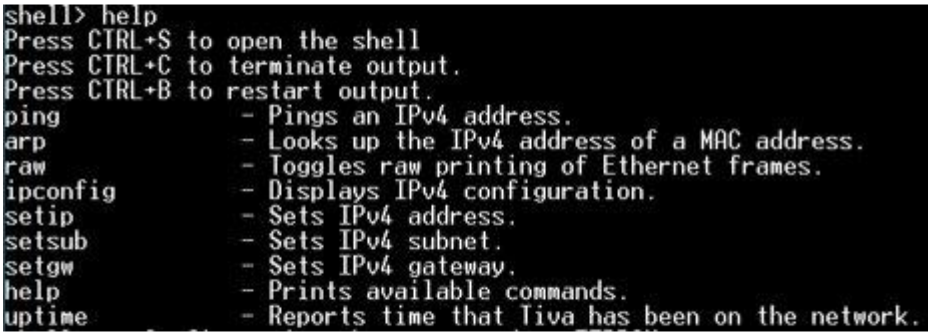
- ✓ ENCHeader
- ✓ EthernetHeader
- ✓ IPHeader
- ✓ ICMPHeader
- ✓ ARPHeader
- ✓ UDPHeader
- ✓ Packet
- ✓ PacketType
- ✓ ARPEntry

Function Prototypes:

- void etherInit(uint8_t mode, uint8_t*, uint8_t*, uint32_t);
- void etherWritePhy(uint8_t, uint16_t);
- uint16_t etherReadPhy(uint8_t);

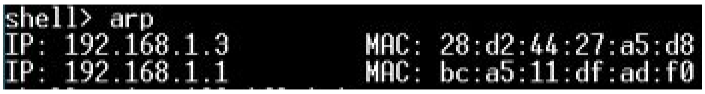
- uint8_t etherKbhit(void);
- uint16_t etherGetPacket(uint8_t[], uint16_t);
- uint8_t etherIsOverflow(void);
- bool etherPutPacket(uint8_t[], uint16_t);
- uint32_t etherGetPacketCount(void);
- void etherIncrementPacketCount(void);
- uint8_t getARPTableCount(void);
- uint8_t* etherGetIpAddress(void);
- uint8_t* etherGetGatewayIpAddress(void);
- uint8_t* etherGetSubnetMask(void);
- void etherSetIpAddress(uint8_t, uint8_t, uint8_t, uint8_t);
- void etherSetGateway(uint8_t, uint8_t, uint8_t, uint8_t);
- void etherSetSubnetMask(uint8_t, uint8_t, uint8_t, uint8_t);
- uint8_t etherIsSameSubnet(uint8_t[], uint8_t[]);
- uint8_t etherIsArp(void);
- uint8_t etherIsArpReq(void);
- uint8_t etherIsArpResp(void);
- uint8_t etherIsIp(void);
- bool etherIsIpUnicast(void);
- bool etherIsValidIp(void);
- uint8_t etherIsPingReq(void);
- uint8_t etherIsPingRes(void);
- uint8_t etherIsUdp(void);
- void etherSendPingResp(void);
- void etherSendPingReq(void);
- void etherSendArpResp(void);
- void etherSendArpReq(uint8_t[]);
- PacketType_t getPacketType(void);
- ARPEntry* getARPEntry(uint8_t*);
- void addARPEntry(ARPEntry);
- uint8_t* etherGetUdpData(void);
- void etherSendUdpData(uint8_t*, uint8_t);
- uint16_t htons(uint16_t);

Results and Outputs of the implemented commands:



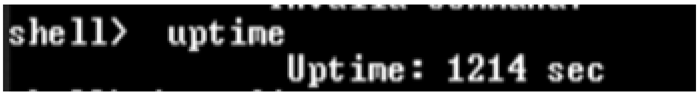
```
shell> help
Press CTRL+S to open the shell
Press CTRL+C to terminate output.
Press CTRL+B to restart output.
ping           - Pings an IPv4 address.
arp            - Looks up the IPv4 address of a MAC address.
raw            - Toggles raw printing of Ethernet frames.
ipconfig       - Displays IPv4 configuration.
setip          - Sets IPv4 address.
setsub         - Sets IPv4 subnet.
setgw          - Sets IPv4 gateway.
help           - Prints available commands.
uptime         - Reports time that Tiva has been on the network.
```

Help command



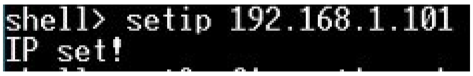
```
shell> arp
IP: 192.168.1.3          MAC: 28:d2:44:27:a5:d8
IP: 192.168.1.1          MAC: bc:a5:11:df:ad:f0
```

Arp command



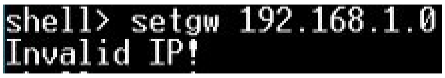
```
shell>  uptime
             Uptime: 1214 sec
```

Uptime command



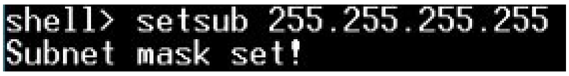
```
shell> setip 192.168.1.101
IP set!
```

Set IP command



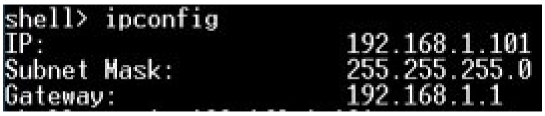
```
shell> setgw 192.168.1.0
Invalid IP!
```

Setting gateway command



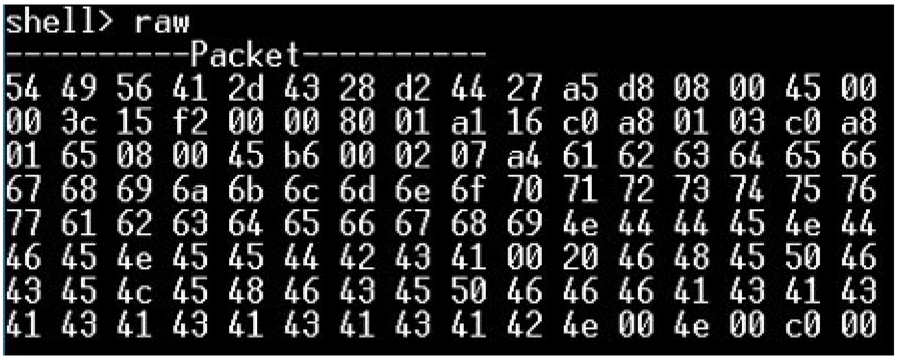
```
shell> setsub 255.255.255.255
Subnet mask set!
```

Set subnet mask command



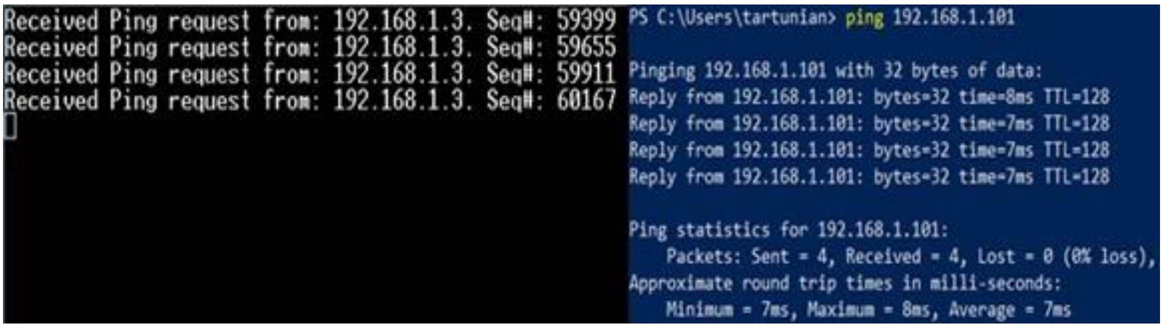
```
shell> ipconfig
IP:                       192.168.1.101
Subnet Mask:              255.255.255.0
Gateway:                  192.168.1.1
```

Ipconfig command



```
shell> raw
----------Packet----------
54 49 56 41 2d 43 28 d2 44 27 a5 d8 08 00 45 00
00 3c 15 f2 00 00 80 01 a1 16 c0 a8 01 03 c0 a8
01 65 08 00 45 b6 00 02 07 a4 61 62 63 64 65 66
67 68 69 6a 6b 6c 6d 6e 6f 70 71 72 73 74 75 76
77 61 62 63 64 65 66 67 68 69 4e 44 44 45 4e 44
46 45 4e 45 45 44 42 43 41 00 20 46 48 45 50 46
43 45 4c 45 48 46 43 45 50 46 46 46 41 43 41 43
41 43 41 43 41 43 41 43 41 42 4e 00 4e 00 c0 00
```

Output showing raw packet printed in hex format



```
Received Ping request from: 192.168.1.3. Seq#: 59399
Received Ping request from: 192.168.1.3. Seq#: 59655
Received Ping request from: 192.168.1.3. Seq#: 59911
Received Ping request from: 192.168.1.3. Seq#: 60167
```

```
PS C:\Users\tartunian> ping 192.168.1.101

Pinging 192.168.1.101 with 32 bytes of data:
Reply from 192.168.1.101: bytes=32 time=8ms TTL=128
Reply from 192.168.1.101: bytes=32 time=7ms TTL=128
Reply from 192.168.1.101: bytes=32 time=7ms TTL=128
Reply from 192.168.1.101: bytes=32 time=7ms TTL=128

Ping statistics for 192.168.1.101:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 7ms, Maximum = 8ms, Average = 7ms
```

Output showing ICMP requests sent from a Windows PC (right) being received on the Tiva (left)

Test cases:

1. Any key combination presses other than CTRL + S, CTRL + C or CTRL + B will throw an "Invalid command" output.
2. Any commands other than the ones listed under "Help" command will throw a "Invalid command" output.
3. While in the middle of receiving or sending the packets/frames if the IP of the Tiva board is updated, then the "Host unreachable" or "Request timed out" is shown on the terminal of the destination system terminal.
4. While updating an IP, Gateway or a Subnet mask, giving a wrong format other than xxx.xxx.xxx.xxx then the Tiva UART shows error "Invalid command" as the output.

Future Work:

- Web server
- Network speed test
- Inline packet sniffer
- Connect to INT pin on ENC28J60 (not included on breakout board)
- Further improve the typecasting of the raw buffer
    - Currently to access headers you must offset by the sizeof the earlier headers manually. This could be added as helper functions.

Demo video link:
https://drive.google.com/drive/folders/1rmx86SMveK0oNM7cBM-Ol0BxNkA5pQAP

References:

(1) Schatzmann, Phil. (2021), ESP32-A2DP,

https://github.com/nihit30/ENC28J60-Library

(2) Datasheet of TIVA TM4C123G ARM-based cortex-M4F microcontroller -- https://www.ti.com/lit/ds/symlink/tm4c123gh6pm.pdf

(3) Datasheet of ENC28J60 Standalone 10/100 Ethernet Controller with SPI Interface. http://ww1.microchip.com/downloads/en/devicedoc/39662c.pdf

(4) Serial Peripheral Interface Bus. Wikipedia. [Online] 14 October, 2021.

https://en.wikipedia.org/wiki/Serial_Peripheral_Interface_Bus.

(5) Horvat, Goran, Damir Šoštarić, and Zoran Balkić. "Cost-effective Ethernet communication for low-cost microcontroller architecture." International journal of electrical and computer engineering systems Vol. 3, 1. (2012): 1-8.

https://scholar.google.com/scholar?cluster=17723064925696251262&hl=en&as_sdt=2005