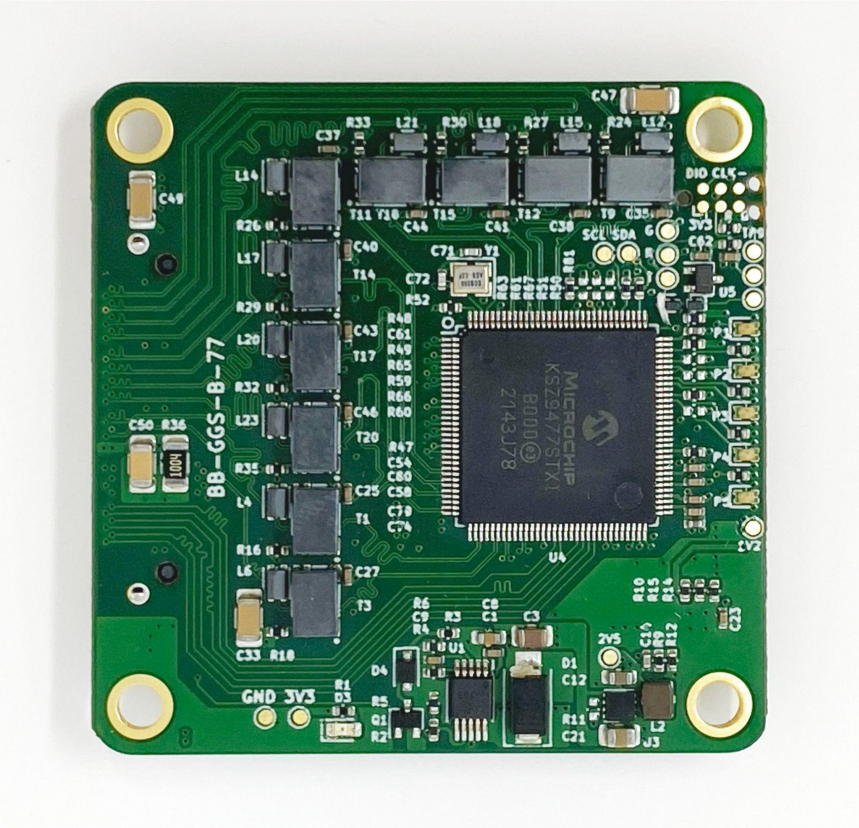


BB-AN-0004
GigaStax Rugged Programming Note

Application Note

September 2022

1 Introduction

GigaStax Rugged is a very small layer 2 unmanaged and managed ethernet switch that fits five 10/100/1000Mbps copper ethernet ports in an ultra-compact 48mm x 48mm form factor.

GigaStax ships without any firmware configuration in a fully unmanaged configuration, meaning it can be dropped into an application immediately with no hardware or software setup. This servers the majority of applications, however there are some applications where specific management functions required. In these cases the onboard management features of the KSZ9477S ethernet switch chip from Microchip can be leveraged by flashing custom firmware on the board.

For more information on what features can be achieved, please refer to the datasheet for the KSZ9477S from Microchip.

2 Firmware Theory of Operation

The KSZ9477S contains no non-volatile memory, and will lose configuration at power cycle. For this reason, an STM32L011D4P7 microcontroller is added to the board. This STM32 microcontroller is connected to the KSZ9477S via an I2C bus, giving it full access to the configuration registers inside the KSZ chip.

This STM32 microcontroller will store a firmware in its non-volatile memory, and run this firmware once at every power cycle. Thus, programming the firmware to send different I2C commands at power on can achieve different KSZ9477S configurations, and thus achieve management functions on GigaStax Rugged.

A template code that contains all pin definitions, along with examples of register writes can be obtained under NDA from Kapek Ltd. Please get in touch at info@botblox.org for this

3 Understanding KSZ9477S Registers

To understand how to decide which registers to write to on the KSZ9477S, an example is provided below.

Requirement: The switch will take a copy of all traffic **received** on port 1 and forward that copy to port 5.

To achieve this, first find the appropriate register in the KSZ9477S datasheet, shown below.

KSZ9477S

5.2.8 PORT N: PORT SWITCH INGRESS CONTROL REGISTERS (0xN800 - 0xN8FF)

5.2.8.1 Port Mirroring Control Register

Address: 0xN800 Port N: 1-7 Size: 8 bits

This register contains the port controls for port mirroring. The Global Port Mirroring and Snooping Control Register must also be properly configured.

Bits	Description	Type	Default
7	**RESERVED**	RO	0b
6	**Receive Sniff** 1 = All packets received on this port are designated as “monitored packets” and will be forwarded to the designated “sniffer port”. 0 = No receive monitoring.	R/W	0b
5	**Transmit Sniff** 1 = All packets transmitted on this port are designated as “monitored packets” and will be forwarded to the designated “sniffer port”. 0 = No transmit monitoring.	R/W	0b
4:2	**RESERVED**	RO	0_00b
1	**Sniffer Port** 1 = This port is designated as the sniffer port and will transmit monitored packets. 0 = Not a sniffer port. Normal operation.	R/W	0b
0	**RESERVED**	RO	0b

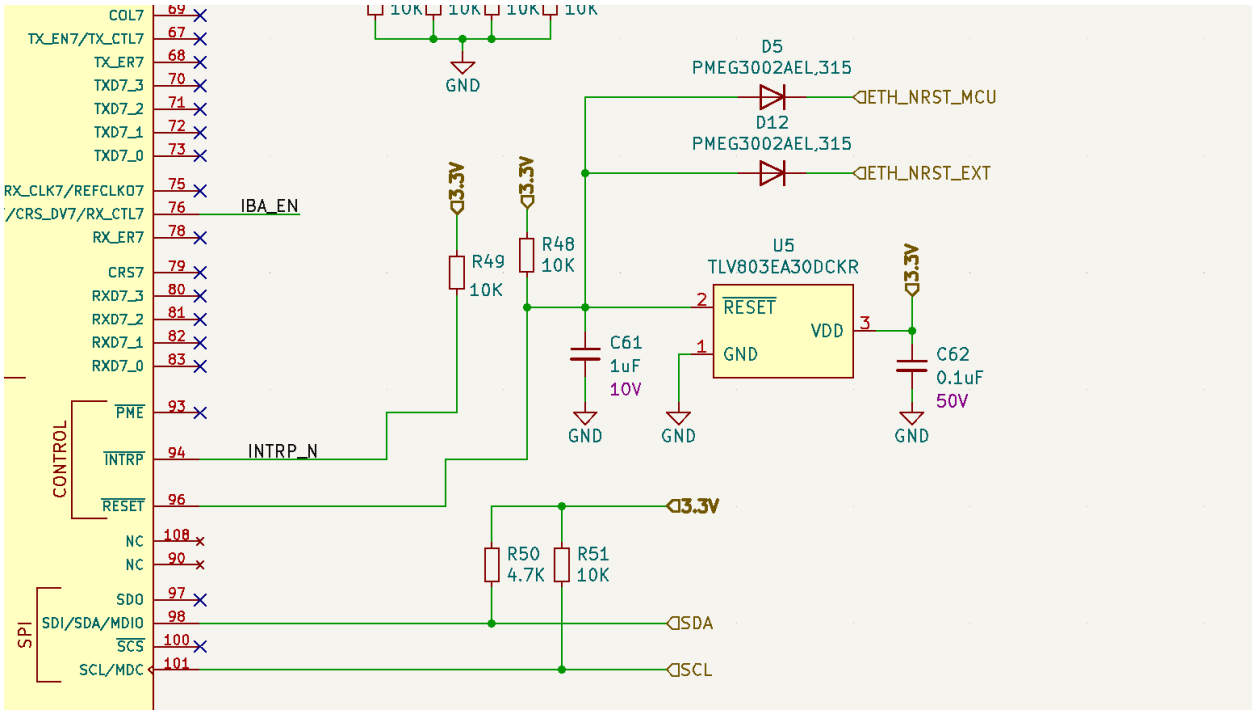
In this case, there are two configurations needed. The first is to setup Receive Sniff on Port 1 and the second the setup port 5 as the sniffer port. In this case, the pseudo code would be…

All address and data values shown below can be inferred from the datasheet. While this is a simple example, the same logic applies to any configuration on the KSZ chip, it is simply a case of finding the correct registers and writing to them successively.

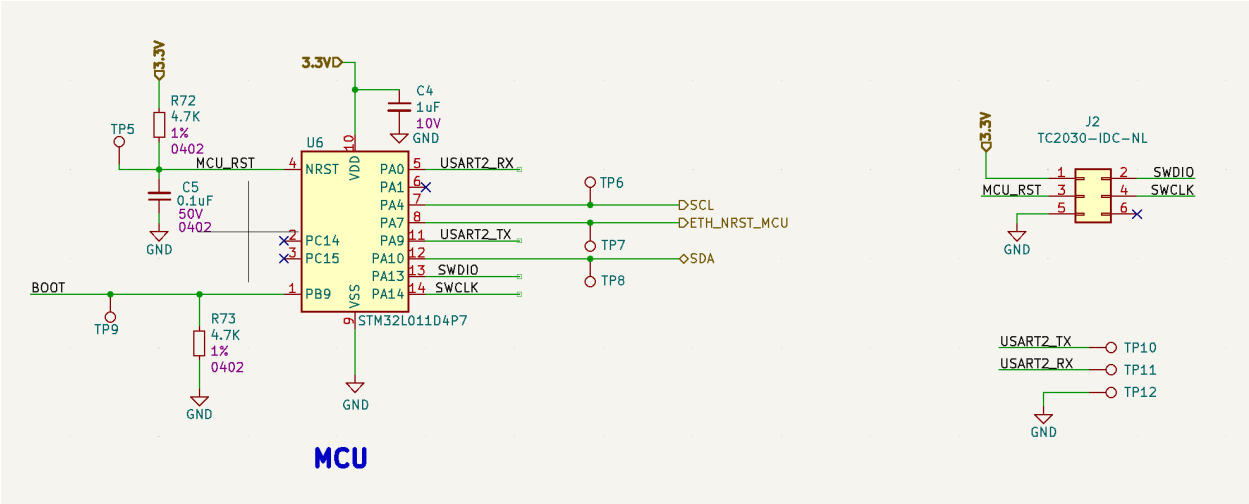
```
I2C deviceAddress = 0b1011111;
port1RegisterAddress = 0x1800;
port5RegisterAddress = 0x5800;
receiveSniffConfig = 0b0100_0000;
snifferPortConfig = 0b0000_0010;

I2C_Write(I2CdeviceAddress, port1RegisterAddress,
receiveSniffConfig);
I2C_Write(I2CdeviceAddress, port5RegisterAddress, snifferPortConfig);
```

4 Physical Mappings

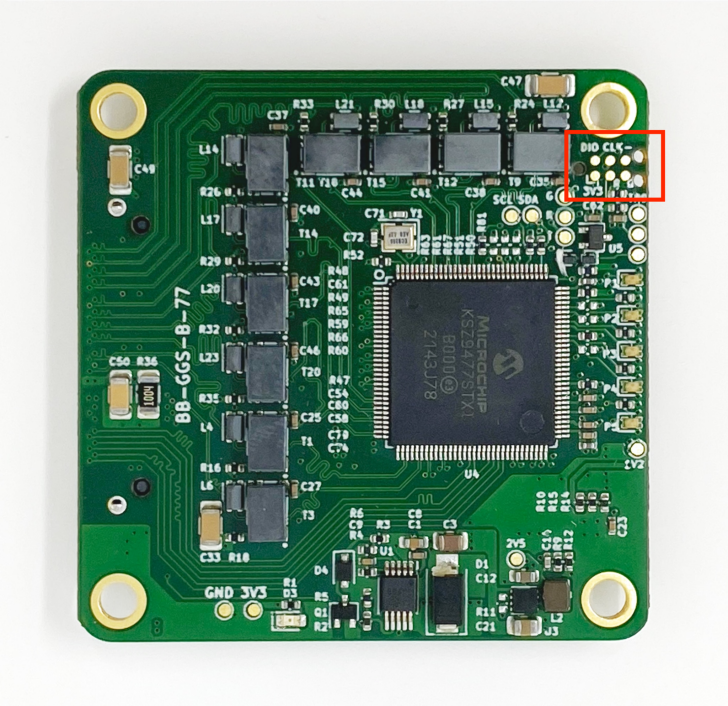
The image below shows the I2C pins on KSZ9477S. Note that pullup resistors are already present on GigaStax Rugged.



The image below shows the corresponding connections on the onboard STM32 microcontroller. The microcontroller also has a connection to the reset line of the KSZ chip, which may be necessary for loading specific configurations.



Note that the programming header for the microcontroller is J2. It is this header through which custom firmware can be loaded onto the microcontroller. The image below shows the location of this programming header. It is a 6 pin Tag Connect footprint.



5 Requirements

In addition to a working and powered GigaStax Rugged board, the following hardware and software is required.

- Segger J-link programmer + USB Cable (link)
- 6 Pin Needle Adapter (link)
- Need adapter retaining clip (optional) (link)
- STM32IDE or equivalent IDE for STM32 microcontrollers

6 Steps

1) Assemble the Segger J-link programmer with the 6 pin Needle Adapter
2) With GigaStax Rugged powered from an external supply, insert the Needle Adapter into the holes around J2
3) Ensure that voltage is seen on the programmers voltage sense pin
4) Flash the firmware. Do not disconnect the programming cable during flashing as this can brick the microcontroller.

Step 2 can be a bit tricky. It is made more difficult due to the close proximity of the transformers. It is best to practice first while the board is off.

6 Datasheet Changelog

Date	Datasheet Version	Author	Notes
20/09/2022	A_A	Josh Elijah	Initial release

7 Contact

If you have any questions regarding this product, please contact us:

info@botblox.org
4 Pavilion Court 600 Pavilion Drive,
Northampton Business Park,
Northampton,
England
NN4 7SL