**AGC Microcontroller Command and Data Format**

The microcontroller is connected to the control computer via RS485 serial link.

All 16 microcontrollers are connected in parallel on this link. Transmit data and receive data are in separate pairs. The data rate is 9600 baud, 8 bit, no parity and 1 stop bit,

All commands are initiated by the control computer. The control computer sends a packet containing a command number. This packet is checked for integrity by the target microcontoller and is ignored by all other microcontrollers. If the packet is error free, the command is executed and a reply is sent to the control computer. Corrupt packets are ignored and therefore the control computer should set a timeout whilst waiting for the reply. The microcontroller should reply within 20 msec.

Command Packet Format

Packets from the control computer are in the following format:

Byte 0 1 2 3 4

STX addr len cmd bcc

**STX** This is the start of the packet and is always 55 Hex

**addr** This is the target microcontroller address in the range 254 decimal. 55 Hex should be avoided.

**len** This is the length of the data field. For command packets this is always 1.

**cmd** This is the command number in the range 1-13 decimal

**bcc** This is the block checksum. This is the least significant 8 bits of the sum of bytes 1-3, i.e all bytes excluding STX and bcc.

Example of a packet that is sent to the AGC

55 2 1 1 4 (hex)

**Valid Commands that can be sent to AGC:**

1 Read Status Returns agc status in a special packet. Details below.

2 Reset Relay Closes the relay. Relay will trip again if fault still present.

3 Trip Relay Opens relay.

4 Close Loop Closes the agc loop.

5 Open Loop Opens the agc loop.

6 Close Cap 1 Puts damping capacitor 1 in circuit.

7 Open Cap 1 Removes capacitor 1.

8 Close Cap 2 Puts damping capacitor 2 in circuit.

9 Open Cap 2 Removes capacitor 2.

10 Restart Restarts the microcontroller by putting it into an infinite loop. A watchdog timer then forces a reset.

11 Ping Microcontroller responds with an ACK packet but does nothing else.

12 Enable Auto Reset.

13 Disable Auto Reset.

**Auto Reset**

With Auto Reset enabled the microcontroller Will attempt to reset the relay

after a fault trip. It will make 3 attempts. The first attempt will be made 5

seconds after the trip, the second attempt will be made 1 minute later and the

third attempt will be made 5 minutes later. After the third attempt the relay will

remain tripped until reset by command 2.

With Auto Reset disabled the relay will remain tripped until reset by command

2. This is the default state.

**Default State**

The microcontroller will power up in the following state:

Relay closed

AGC Loop closed

Cap 1 in

Cap 2 out

Auto Reset disabled

**Microcontroller Responses**

Commands 2-13 will provoke the following response, known as an ACK

packet, providing the command packet was received correctly:

STX addr

STX Start of Text character, 55 hex.

addr address of the microcontroller sending the ACK packet.

Command 1, Read Status, provokes a more complicated packet:

Byte O 1 2 3 4 5 6 7 8 9 10

STX addr len board ad0 ad1 ad2 ad3 ad4 ad5 ad6

Byte 11 12 13 14 15

ad7 port:A port:B port:C bcc

STX 55 hex

addr (this is the address of the control computer)

len 12 decimal. Bytes 3-14 inclusive constitute the data field.

board The address of the microcontroller sending this packet.

ad0 A/D converter channel 0: +5v sense.

ad1 A/D converter channel 1: +15v sense.

ad2 A/D converter channel 2: +500v sense.

ad3 A/D converter channel 3: -15v sense.

ad4 A/D converter channel 4: sense.

ad5 A/D converter channel 5: temperature.

ad6 A/D converter channel 6: forward power.

ad7 A/D converter channel 7: reflected power.

portA bit 0 1 if relay closed.

bit 1 1 if inhibit signal on.

bit 2 1 if power signal active.

bits 3-7 don't care.

portB bits 0-1 don't care.

bit 2 1 if Cap 1 in circuit.

bit 3 1 if Cap 2 in circuit.

bit 4 1 if loop closed.

bits 5-7 don't care.

portc bit 0 0 if bad duty cycle.

bit 1 0 if bad SWIR.

bit 2 0 if bad +5v.

bit 3 0 if bad +15v.

bit 4 0 if bad +500v.

bit 5 0 if bad -15v.

bit 6 0 if bad +50v.

bit 7 don't care.

**Note:** if the relay is tripped the values returned in port:C are the values taken at

the time of the trip. These values will not change until the relay is closed again.

bcc is the block checksum. This is the least significant 8 bits of the sum of bytes

1-14, ie all bytes excluding STX and bcc.

**AGC COMMANDER SOFTWARE**

The AGC COMMANDER software is a python script that is used to obtain diagnostic information about each of the 16 Transmitters. It is composed of 2 parts, 1: the commander and 2: the email client.

**1.AGC COMMANDER**

The commander runs on python 3.9 and has a few dependencies that need to be installed before it can run. Also the user needs to be added to the “dialout” group when on linux.

*git*

*Pyserial*

*PyQT5*

*Pandas*

*Python3.9*

Once these have been installed, the repository can be called by typing the following at the command teminal.

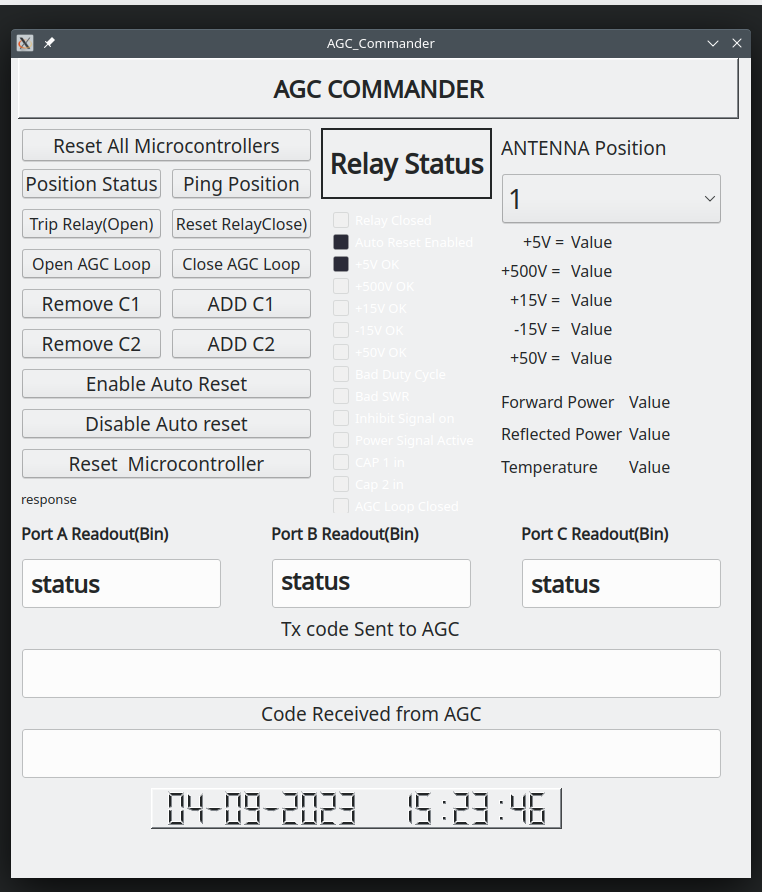
*cd $BOREALISPATH*

*git clone* [*https://github.com/Cassiopeia-Elizabeth/AGC\_COMMANDER.git*](https://github.com/Cassiopeia-Elizabeth/AGC_COMMANDER.git)

from the /borealis/AGC\_COMMANDER/ folder run:

*python3 AGC\_COMMANDER.py*

The following screen should be present:

**

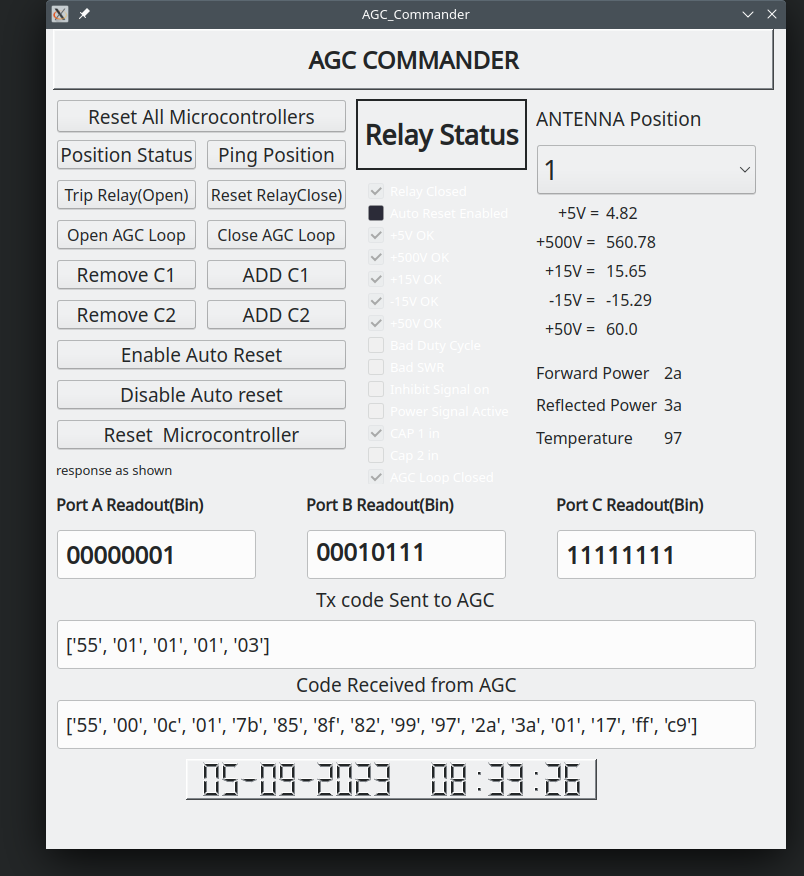
This program is pretty easy to use and follows the AGC COMMANDER instruction set.

The antenna position relates to the ACTUAL antenna number and not the transmitter. As transmitter could be swapped over there is a csv files where you would change the AGC number for the antenna position.

The first step would be to select the antenna position required and click the “position status” box.

This will send a packet of data to the transmitter and a response should be within 200ms. You will see the packet that you sent as well as the packet that was received in the bottom two rows. The packet received should fill most of the row up with 16 hex values. The check boxes should also indicate whether the voltages are correct.

Below is a screenshot of a position status screen:



As you can from the screenshot that the commander program sends a packet of 5 hex values. Only on the position status will you see the longer packet of 16 hex values.

You will note that the relay box does not do anything at the moment. That will be either red or green depending on whether or not the relay is closed or open. This will included at a later date. The forward power, reflected power and temperature boxes are not required. The temperature maybe needed for indicative purposes at some point. These are still shown in case at some point they are required they can be incorporated much easier.

The AGC COMMANDER also keeps a log, this will however be removed as this script is mainly for diagnostics. The AGC EMAIL script will be the main logging script as any faults will be noted and an email will be sent to notify selected users.