The AEAT-601B-F06 Incremental magnetic encoders measure motion and direction by using two sensors at two different positions relative that measure change across a code disk. The code disk has fixed shafts that are fixed distances from each other like the stokes on a bicycle wheel which have alternating magnetic poles and will output a voltage based on the magnetic pole type. This allows the measurement of speed of the wheel speed using the counts per second. The quadrature allows detection of direction also and detection of direction uses two sensors, A and B, that are placed so that when there’s motion, they detect a rising edge at a 90-degree phase difference. A will detect first when there’s anticlockwise movement and B will detect first when there’s clockwise movement. To use the quadrature interface mode, pin PA\_5 of channel 1 of TIM 2 and pin PB\_3 of channel 2 of TIM 2 will be used to input channel A and B, 5V connected to +5V and GND to GND both on CN6 and the index is not needed so it will remain disconnected. The TIM 2 timer peripheral will count the counts of rising edges without disrupting the CPU and we use a software timed interrupt that will return the sample the number of counts every specified period of time and store them in a variable which can be used to calculate angular speed which can then be used to calculate wheel speed using wheel radius and gear ratio.

Two motors need two different current sensing circuits. The motor drive board has a current amplifier circuit that is used to input voltage into the microcontroller for measurement. First the current of the motor flows across the 0.1 resistor in series with the motor. in order to get a more sensitive current sensor, we need a larger voltage so we use the aplc amplifier. This resistor is connected to the current amplifier circuit parallel to the voltage of the 0.1-ohm resistor so inputted into the aplc-c784 Vin+ by using isense + resistor path and isense + resistor path is Vin- which is essentially ground. The voltage that is outputted is the difference between Vin+ and Vin- is amplified to be between 0 – 5V with the gain being determined by VDDin of amplifier. Here, VDDin is fixed by the voltage regulator LM3480 – 5 at 5V to get a gain of 8.029 which is enough to use most of the ADC range. The microcontroller will connect to the pins Pin 10 Avago A(+) of jp1a to PC\_2 of CN7 and Pin 9 Avago A(–) of jp1a to PC\_3 of CN7 for Motor A, Pin 12 Avago B+ of jp1a to PC\_4 of CN10 Pin 11 Avago B- of jp1a to PC\_5 of CN10 for Motor B. To get an accurate relationship constant between voltage and current, we have to collect data of current through motor using an Ammeter and voltage using a volt meter across TP20 and TP18 for motor 1 for example, plot it and find the gradient. Using the gradient, the software can calculate the current from voltage input using voltage\*gradient.

The DS2781 chip will measure the current and voltage from the battery, place it in a register and output it to the microcontroller when software sends commands to it. The current is measured across the resistor Rsns that is placed between Vss and Rsns pins and the voltage is measured at Vin with respect to Vss pins. The chip is capable of reading the current down to a resolution of 1.56 microvolt/Rsns, Rsns here is 0.01 ohms so 156 micro Amps and up to a maximum of +- 5.11 A as 16 bits are used for current measurement and current measurement is updated every 3.515 s. The voltage is measured down to a resolution of 9.76 mV and up to a maximum of +9.9902V and the voltage register is updated every 440 ms. The microcontroller will connect to the pins of CN5.

The HM – 10 is a BLE module that allows the STM32 to connect to other BLEs. The BLE module services that allow the defining of their functional capabilities and roles are controlled or called using AT Commands that are sent to the HM – 10. The HM-10 will have a unique service profile that is a collection of services. The HM-10 can run at very high speeds of 32 MHz, allowing both Slave and Master roles, ability to send 20-bit character words ie. Hello would be 5 characters and can be put in sleep mode when not in use where it consumes much less power. Slave will mean that it will respond to master’s commands immediately and will have to process information at the same transmitter and receiver frequency as master so maintain a safe data exchange. The Master will initiate the connection using an AT command and the slave’s address and will decide when the slave will respond to commands ie. when to sleep or be active. It also means that the slave BLE will be able to be connected to by that master only. The HM-10’s ability to act as slave allows the buggy to be controlled from another BLE, our phones, that will be master to send a command to be processed by software of the STM32. The HM – 10 uses 32-bit commands that are sent to the module by software serially through a UART strip and the software can then control the module using SoftSerial.h. The UART connections between the STM32 are CN6 3.3 V to VCC, CN6 GND to GND, PA\_2 to TX and PA\_3 to RX.

The quadrature allows detection of direction also and detection of direction uses two sensors, A and B, that are placed so that when there’s motion, they detect a rising edge at a 90- degree phase difference so one would detect a rising edge after a ¼ of the time period. When rotation is clockwise, the 90-degree difference is positive meaning that A is detecting after B and anticlockwise so 90-degree difference is negative so A is detecting before B.

The STM32 microcontroller will get the signals from the encoder via the TIM2 or TIM5 general purpose timers in which are chosen because of their 32-bit quadrature encoder support interface. The quadrature will connect to +5V on VDD and GND, the A channel will connect to

If I plan to use them then how will I implement them

The quadrature will connect to the stm through its rail with A, B, index connecting to … and GND connecting to

The software will do the following; using input capture mode, the TIM 2 counter will intialise the timer to 0 and counter cleared. The timer will start and counts recorded and after a specific period an interrupt occurs that stops the counting and the time of input is recorded. The process is restarted for a second time and then we have two count-time values and using output compare mode and equation 1 we get speed. For direction, we use output compare to compare the times of rising edges of A and B to see which one leads and hence we get the direction.

Quadrature encoders:

The TIM 2 timer peripheral will count the counts of rising edges and we use a timed interrupt that will return the amount of interrupts store them somewhere which can be used to calculate speed later. To use the quadrature interface mode, pin PA\_5 of channel 1 of TIM 2 and pin PB\_3 of channel 2 of TIM 2 will be used to input channel A and B, 5V connected to +5V and GND to GND both on CN6 and the index is not needed so it will remain disconnected

TIM 2 32 oscilloscope in channel pin = PA\_5 and PB\_3

Current sensing:

Two motors need two different current sensing circuits. The motor drive board has a current amplifier circuit that is used to input voltage into the microcontroller for measurement. First the current of the motor flows across the 0.1 resistor in series with the motor. in order to get a more sensitive current sensor, we need a larger voltage so we use the aplc amplifier. This resistor is connected to the current amplifier circuit parallel to the voltage of the 0.1-ohm resistor so inputted into the aplc-c784 Vin+ by using isense + resistor path and isense + resistor path is Vin- which is essentially ground. The voltage that is outputted is the difference between Vin+ and Vin- is amplified to be between 0 – 5V with the gain being determined by VDDin of amplifier. Here, VDDin is fixed by the voltage regulator LM3480 – 5 at 5V to get a gain of 8.029 which is enough to use most of the ADC range. The microcontroller will connect to the pins Pin 10 Avago A(+) of jp1a to PC\_2 of CN7 and Pin 9 Avago A(–) of jp1a to PC\_3 of CN7 for Motor A, Pin 12 Avago B+ of jp1a to PC\_4 of CN10 Pin 11 Avago B- of jp1a to PC\_5 of CN10 for Motor B. To get an accurate relationship constant between voltage and current, we have to collect data of current through motor using an Ammeter and voltage using a volt meter, while the motor is turned on, across TP20 and TP18 for motor 1 for example, plot it and find the gradient. Using the gradient, the software can calculate the current from voltage input.

Battery monitor:

The DS2781 chip will measure the current and voltage from the battery, place it in a register and output it to the microcontroller when commands are sent to it. The current is measured across the resistor Rsns that is placed between Vss and Rsns pins and the voltage is measured at Vin with respect to Vss pins. The chip is capable of reading the current down to a resolution of 1.56 microvolt/Rsns, Rsns here is 0.01 ohms so 156 micro Amps and up to a maximum of +- 5.11 A as 16 bits are used for current measurement and current measurement is updated every 3.515 s. The voltage is measured down to a resolution of 9.76 mV and up to a maximum of +9.9902V and the voltage register is updated every 440 ms.

Using a pwm generator that connects to pin 3 and pin 6 (V+) to JP5 port GND, the motor can be turned on. Once turned on an ammeter in series with the motor and the pins tp18 and 19 (motor 1) and tp 20 and 21 (motor 2) will measure the current and voltage. The voltage across the tps result will be compared to the current of the ammeter to know how much voltage represents the current through the 0.1-ohm resistor. The voltage at different currents through the motor are measured and plotted to know the relation constant. Based on the control pwm chosen as uni or bipolar we can choose the signal to be tested while current sensing.

Battery monitor:

The DS2781 chip will measure the current and voltage from the battery, place it in a register and output it to the microcontroller when commands are sent to it. The current is measured across the resistor Rsns that is placed between Vss and Rsns pins and the voltage is measured at Vin with respect to Vss pins. The chip is capable of reading the current down to a resolution of 1.56 microvolt/Rsns, Rsns here is 0.01 ohms so 156 micro Amps and up to a maximum of +- 5.11 A as 16 bits are used for current measurement and current measurement is updated every 3.515 s. The voltage is measured down to a resolution of 9.76 mV and up to a maximum of +9.9902V and the voltage register is updated every 440 ms.

Bluetooth:

The HM – 10 is a BLE module that allows the STM32 to connect to other BLEs. The BLE module characteristics and roles are controlled

using AT Commands that are sent to the HM – 10. The BLE module will make use of

services that are all part of its service profile which is a summary of its services that they offer. The HM – 10 uses 32-bit commands that are sent to the module by software and are sent serially through a UART strip. The HM-10 can run at very high speeds of 32 MHz, allowing both Slave and Master roles, ability to send 20-bit character words ie. Hello would be 5 characters and can be put in sleep mode when not in use. The HM-10’s ability to act as slave allow the buggy to be controlled from another ble, our phones, that will be master to send a command to be processed by software. Slave will mean that it will respond to master’s commands immediately and will have to process information at the same transmitter and receiver frequency as master so maintain a safe data exchange. It also means it can’t connect to other master devices. The UART connections between the STM32 are CN6 3.3 V to VCC, CN6 GND to GND, PA\_2 to TX and PA\_3 to RX. Using these inputs the software can then control the module using SoftSerial.h

magnetic incremental encoder has a plate that has alternating magnetic poles that have fixed distance between them. The sensor will react to the magnetic pole type north or south by a change in voltage.

Mechanical (max) 12000 RPM

VDD Supply Current IDD 16 20 mA

Output High Voltage VOH VDD-0.5 V

Output Low Voltage VOL VSS+0.4 V

Output Current IO 4 mA

The HM – 10 is a BLE module that allows the STM32 to connect to other BLEs. The

BLE module characteristics and roles are controlled Using AT Commands that are sent to the HM – 10. The BLE module will make use of

services that are all part of its service profile which is a summary of its services that they offer. The HM – 10 uses 32-bit commands that are sent to the module by software and are sent serially through a UART strip. The HM-10 can run at very high speeds of 32 MHz, allowing both Slave and Master roles, ability to send 20-bit character words ie. Hello would be 5 characters and can be put in sleep mode when not in use. The HM-10’s ability to act as slave allow the buggy to be controlled from another ble, our phones, that will be master to send a command to be processed by software. Slave will mean that it will respond to master’s commands immediately and will have to process information at the same transmitter and receiver frequency as master so maintain a safe data exchange. It also means it can’t connect to other master devices. The UART connections between the STM32 are CN6 3.3 V to VCC, CN6 GND to GND, PA\_2 to TX and PA\_3 to RX. Using these inputs the software can then control the module using SoftSerial.h