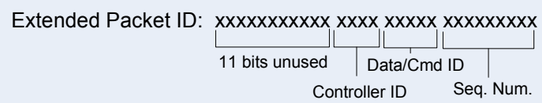
Description

Communication between the Boost Controller, the Pneumatic Spring controller, and the Buoy Controller is done via CAN Bus. Custom fields within the 29bit packet ID are used for data and command ID, as described here. Encoded in the CAN ID are three values, a controller ID that specifies either the origin of streamed data, or the destination of commands, a Data/Cmd ID that specifies the data or command packet contents, and a sequence number that groups data from one instant in time across packets, this counts up for streamed data and rolls over. It is intended that the Data/Cmd ID's are unique to each controller.

Request and response is avoided, instead each controller streams appropriate data repetitively at relatively low rates (10's of Hz). Packets can be sent to controllers change modes and state, but are not acknowledged, instead the complete controller state is represented in the streamed packets so the sender can verify that a change was accepted.

Packet ID Structure

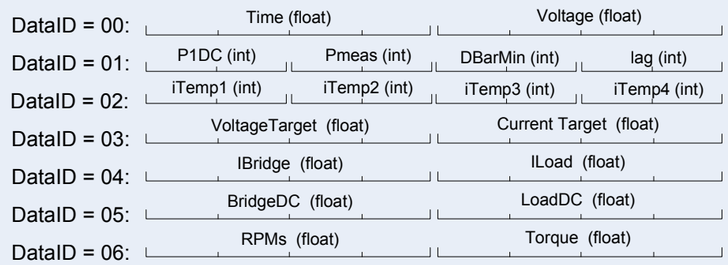


Sequence Number = 9 bits
Data/Cmd ID = 5 bits
Controller ID = 4 bits

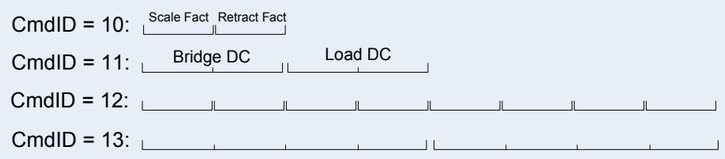
Higher ID values have higher priority

Boost Controller (ID = 02)

Data Packets Streamed Out



Cmd Packets Acceped



PneumaticSpring Controller (ID = 01)

Data Packets Streamed Out

Cmd Packets Acceped

Buoy Controller (ID = 00)

Data Packets Streamed Out

Cmd Packets Acceped

CAN Bus @ 500kbit/sec

Oscillator Setup
- Fosc = 79.2275MHz
- Fcy = 39.61375

QEI
- Encoder module is set up to set the counter to 0 if rotating clockwise at the index pulse, and 4096 if rotating counterclockwise.
- Calls _QEI_Interrupt on index pulse.

DMA
Three DMA channels are set-up.
- DMA0 transfers data from ADC to buffer
- DMA1 transfers CAN data from buffer to CAN1 when CAN1 is able to send.
- DMA2 transfers CAN data from CAN1 to buffer when packet is received and passes filters.

TMR1
- Based on Fosc and rolls over at approximately 25Hz
- T1CON = 0x8020 (1:64 prescaler)
- PR1 =2X12378-1

- _T1Interrupt is executed when this rolls over.

CAN1
- Set to 500kb/s
- Uses 6 buffers in DMA ram
  - 4 Tx buffers
  - 2 Rx buffers
- The TMR1 Interrupt loads the Tx buffers and initiates CAN transfers which proceed w/o CPU involvement.
- An interrupt occurs when packets that match the specified mask arrive.

MCPWM
- PWM Timebase = Tcy
- PWM Freq approximately 20kHz
- MC channels 1-3 drive FET pairs and the low side of channel 4 drives the load-dump FET.
- Deadtime on channel 1-3 set to: 1 microsecond
- Initial values are 100% for channel 1-3 and 0% for channel 4.
- PWM values are set by boost voltage PID controller (initiated by ADC sample set complete interrupt (_DMA0Interrupt).
- Override values are set by _CNInterrupt based upon rotor position via table look-up when appropriate (motor mode).

TMR2
- Based on Fosc and rolls over at 604.467Hz
- T1CON = 0x8000 (1:1 prescaler)
- PR1 =65535

- _T2Interrupt is executed when this rolls over.

UART2
- Set to 115200 baud 8N1
- Interrupts enabled for characters in Rx Buffer
- This UART is currently only used for debugging and will be disabled for deployment.

TMR3
- Based on Fosc and rolls over at 128.616 kHz
- T1CON = 0x8000 (1:1 prescaler)
- PR1 =308

- _T3Interrupt is executed when this rolls over.

ADC1
- 12 bit mode
- Set up to scan through specified inputs repeatedly and transfer resulting data to buffers via DMA
- DMA system is set up to cause interrupt when one half of the buffer is full.
- Example. The ADC scans 4 inputs at 128kHz and uses DMA to place the results in a big buffer, sequentially. When the set of four inputs has been sampled the specified number of times (32 for example), the DMA initiates an interrupt (_DMA0Interrupt). This resulting interrupt occurs at about 1kHz (128/(4*32)) and 32 samples of each channel are available then. The samples aren't simultaneous, the maximum skew across the channels is 4/20000 = 31.25 microseconds.

CN
- Set up to cause interrupt on any change of CN6 or CN7 which are the A and B channels of the encoder. Calls _CNInterrupt which adjusts the FET commutation based upon rotor position.
- This is enabled if commutation is needed (e.g. motor mode) and disabled in cases where the FET pwm signals don't depend on rotor position.

_T1Interrupt
- Updates “time” variable.
- Assembles and ships CAN packets over to CAN controller via DMA.

_T2Interrupt
- Computes rotor speed based on encoder.
- Interrupt is called at TMR2 rollover frequency (~600Hz), speed updates down slower based on postscaler defined for this purpose.

_T3Interrupt
- Initiates ADC scanning which happens in one-shot mode.

_CNInterrupt
- This is called every time one of the encoder inputs (A or B) changes state.
- This interrupt performs the needed commutation by setting the MCPWM over-ride bits appropriate to the rotor position.

_QEI_Interrupt
- This occurs on index pulse and is used to count revolutions in a signed 16bit counter.

DMA0Interrupt
- After N channels are read M times, the DMA system notices and causes this interrupt which performs filtering and subsequent processing (PID loops) based on the samples collected. This all needs to complete fast enough to be done before next set of samples is available. However, the ADC/DMA is working in “ping-pong” mode so the ADC is filling the second buffer while this interrupt is processing the first buffer.
- This interrupt
    1) Performs filtering of most recent set of ADC samples
    2) Executes PID loops
    3) Sets Boost and Load-Dump PWM values based on PID results.

_U2RxInterrupt
- This occurs on UART2 character received and responds to single character inputs that affect settings and modes.