In Comms library:

boolean sendMessage(Message m) - send a new message (of any type), can be called anywhere

packet\_t[] packageMessage(Message m) - splits message into packets

packet\_t createPacket(byte[] content, Service sender, … )

byte calculateCRC(byte[] p)

boolean sendPacket(packet\_t p)

boolean writePacketToSerial(packet\_t p) - write packet to serial including byte

stuffing

void checkMessages() - read messages from message queue and process them, called in main loop

Message fetchMessageFromQueue() - pop first message off queue

void readMessage(Message m) - parse actual message contents

mergeServiceTable(Service[]) (if message is a service table)

void readArduinoSerial() - triggered on interrupt, reads bytes from hardware serial and handles

packages

void writeToSoftwareBuffer(byte b)

packet\_t getPacketFromSerial() - including deserialise/reverse byte stuffing

boolean checkPacketCRC(packet\_t p)

void writePacketToSerial(packet\_t p) - cf. sendMessage()

void addPacketToQueue(packet\_t p)

Message buildMessage()

void addMessageToQueue(Message m)

void initialiseNode(Port[] ports) - set up the node and look-up table and advertise services

createNewServiceTable()

addLocalServices()

sendServiceTable(Port port)

sendMessage()

Application functions:

ping(Service target)

sendMessage(...)

In DCS library: (to be implemented later)

subscribe()

request()

unsubscribe()