Test packet variation ideas :

What about sending you bad packets to test your logic/flow? Similar set to these I reckon

Should be configuration based so I can add my own as I go
request and expected response pairs? Order important for memory operationt tests
grouping of operations into test units? Still want to know what exactly failed though. hmm

flash writing tests should be optional and in a special section because flash writes are limited!

Test each reset type by read ram, write ram, read ram, reset, read ram

Use the echo function to ensure all the header variations work as expected.
Flags payloadid
ditto + ack
ditto + length
ditto + addrs
ditto + ack and length
ditto + ack and addrs
ditto + addrs and length

addr tests :
0 for source = error
ecu addr for source = error
ecu for target works
non ecu for target fails silently

// full test of every known combo of positive responses in a non combinatorial way plus :
one of each combo of header types and look for matching responses
one of each payload id and look for correct responses
one of each location id for memory actions :
    all known ram should work, only full blocks of flash and bigger should work
for writes and burns get, modify, put, get check :
    write straight to ram, check only ram
    write straight to flash, check ram and flash afterwards
    write to ram, tell it to burn that down, check both
table writing of blocks, axis rpm, axis load, cells validation errors – all table types, main and small ones

// ensure the escaping doesn't mess up the checksum check
packet with no escaped bytes and bad checksum
packet with escaped escape and bad checksum
packet with escaped start and bad checksum
packet with escaped stop and bad checksum
packet with escaped escape and escaped start and bad checksum
packet with escaped escape and escaped stop and bad checksum
packet with escaped start and escaped stop and bad checksum

// test that escapes function and produce good data that works and don't mess up the checksum check
// use echo function for this

packet with escaped escape and good checksum
packet with escaped start and good checksum
packet with escaped stop and good checksum
packet with escaped escape and escaped start and good checksum
packet with escaped escape and escaped stop and good checksum
packet with escaped start and escaped stop and good checksum

check that one byte +1 and one -1 don't fail checksum (only error type we can't catch...)

// test that various wrong payload IDs behave as expected
unimplemented payload IDs iterrated through
unknown even payload Ids return correct error
odd payload ids return correct error

// lengths
length doesnt match length header field
length doesn't match payload id and location id for blocks (one wrong sent for each location id)
length wrong because N zero bytes inserted which allow checksum to pass still