

```
#ifndef _PACKET_H
#define _PACKET_H

#include <string>

// Packet bytes
#define NONE      (char)0x00
#define SOH      (char)0x01

// Flags
#define ENQ      (char)0x80
#define ACK0     (char)0x40
#define ACK1     (char)0x20
#define NACK     (char)0x10
#define RVI      (char)0x08
#define SEQ      (char)0x04
#define CON      (char)0x02
#define NTS      (char)0x01

class Packet {
public:
    Packet();
    Packet(const Packet& packet);    // cpconst
    Packet(char flag);              // control packet
    Packet(std::string data);        // data packet (from GUI)
    void append(char c);
    bool valid();                   // validates packet
    bool complete();               // checks if enough bits have been received (for input from
serial port)
    int flags();                   // flags byte from packet (seq bits set to 0!!!)
    std::string data();            // data section from packet
    std::string toString();        // entire packet (for sending)
    void seq(bool toggle);         // sets the seq bit
    bool seq();                    // checks the seq bit
    void clear();
    int length();
    bool cmd();
    void calcCRC();
private:
    std::string packet_;          // entire packet (header + data + crc)
    size_t length_;
    static const size_t minLength_ = 4;
};

#endif
```