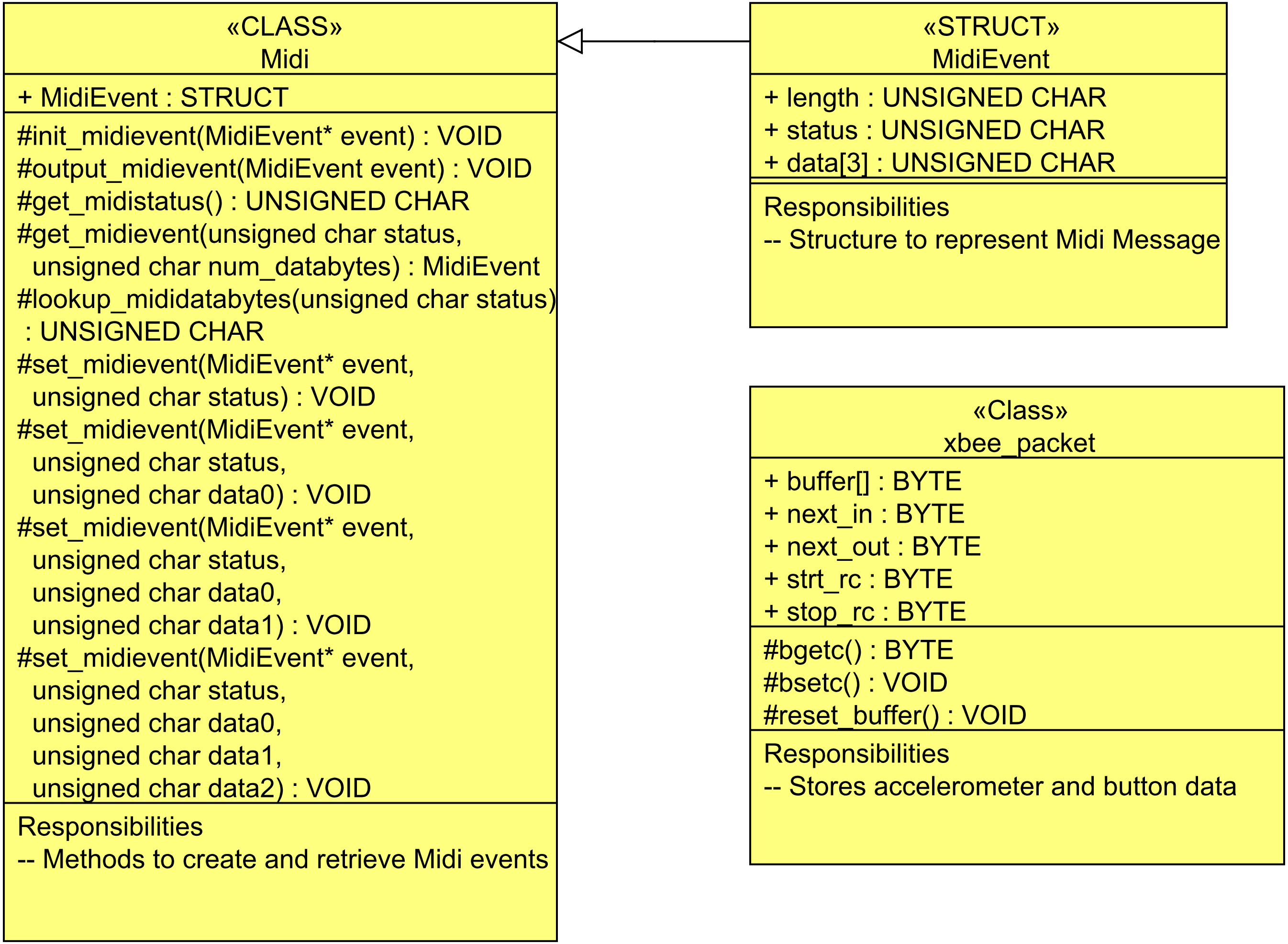

«CLASS»
Midi

+ MidiEvent : STRUCT

#init_midievent(MidiEvent* event) : VOID
#output_midievent(MidiEvent event) : VOID
#get_midistatus() : UNSIGNED CHAR
#get_midievent(unsigned char status,
  unsigned char num_databytes) : MidiEvent
#lookup_mididatabytes(unsigned char status)
 : UNSIGNED CHAR
#set_midievent(MidiEvent* event,
  unsigned char status) : VOID
#set_midievent(MidiEvent* event,
  unsigned char status,
  unsigned char data0) : VOID
#set_midievent(MidiEvent* event,
  unsigned char status,
  unsigned char data0,
  unsigned char data1) : VOID
#set_midievent(MidiEvent* event,
  unsigned char status,
  unsigned char data0,
  unsigned char data1,
  unsigned char data2) : VOID

Responsibilities
-- Methods to create and retrieve Midi events

«STRUCT»
MidiEvent

+ length : UNSIGNED CHAR
+ status : UNSIGNED CHAR
+ data[3] : UNSIGNED CHAR

Responsibilities
-- Structure to represent Midi Message

«Class»
xbee_packet

+ buffer[] : BYTE
+ next_in : BYTE
+ next_out : BYTE
+ strt_rc : BYTE
+ stop_rc : BYTE

#bgetc() : BYTE
#bsetc() : VOID
#reset_buffer() : VOID

Responsibilities
-- Stores accelerometer and button data