ALPS Touchpad Protocol

Introduction

Currently the ALPS touchpad driver supports seven protocol versions in use by ALPS touchpads, called versions 1, 2, 3, 4, 5, 6, 7 and 8.

Since roughly mid-2010 several new ALPS touchpads have been released and integrated into a variety of laptops and netbooks. These new touchpads have enough behavior differences that the alps_model_data definition table, describing the properties of the different versions, is no longer adequate. The design choices were to re-define the alps_model_data table, with the risk of regression testing existing devices, or isolate the new devices outside of the alps_model_data table. The latter design choice was made. The new touchpad signatures are named: "Rushmore", "Pinnacle", and "Dolphin", which you will see in the alps.c code. For the purposes of this document, this group of ALPS touchpads will generically be called "new ALPS touchpads".

We experimented with probing the ACPI interface _HID (Hardware ID)/_CID (Compatibility ID) definition as a way to uniquely identify the different ALPS variants but there did not appear to be a 1:1 mapping. In fact, it appeared to be an mn mapping between the _HID and actual hardware type.

Detection

All ALPS touchpads should respond to the "E6 report" command sequence: E8-E6-E6-E6-E9. An ALPS touchpad should respond with either 00-00-0A or 00-00-64 if no buttons are pressed. The bits 0-2 of the first byte will be 1s if some buttons are pressed.

If the E6 report is successful, the touchpad model is identified using the "E7 report" sequence: E8-E7-E7-E7-E9. The response is the model signature and is matched against known models in the alps_model_data_array.

For older touchpads supporting protocol versions 3 and 4, the E7 report model signature is always 73-02-64. To differentiate between these versions, the response from the "Enter Command Mode" sequence must be inspected as described below.

The new ALPS touchpads have an E7 signature of 73-03-50 or 73-03-0A but seem to be better differentiated by the EC Command Mode response.

Command Mode

Protocol versions 3 and 4 have a command mode that is used to read and write one-byte device registers in a 16-bit address space. The command sequence EC-EC-EC-E9 places the device in command mode, and the device will respond with 88-07 followed by a third byte. This third byte can be used to determine whether the devices uses the version 3 or 4 protocol.

To exit command mode, PSMOUSE_CMD_SETSTREAM (EA) is sent to the touchpad.

While in command mode, register addresses can be set by first sending a specific command, either EC for v3 devices or F5 for v4 devices. Then the address is sent one nibble at a time, where each nibble is encoded as a command with optional data. This encoding differs slightly between the v3 and v4 protocols.

Once an address has been set, the addressed register can be read by sending PSMOUSE_CMD_GETINFO (E9). The first two bytes of the response contains the address of the register being read, and the third contains the value of the register. Registers are written by writing the value one nibble at a time using the same encoding used for addresses.

For the new ALPS touchpads, the EC command is used to enter command mode. The response in the new ALPS touchpads is significantly different, and more important in determining the behavior. This code has been separated from the original alps_model_data table and put in the alps_identify function. For example, there seem to be two hardware init sequences for the "Dolphin" touchpads as determined by the second byte of the EC response.

Packet Format

In the following tables, the following notation is used:

```
CAPITALS = stick, miniscules = touchpad
```

?'s can have different meanings on different models, such as wheel rotation, extra buttons, stick buttons on a dualpoint, etc.

PS/2 packet format

```
byte 0:  0    0 YSGN XSGN    1    M    R    L
byte 1: X7   X6   X5   X4   X3   X2   X1   X0
byte 2: Y7   Y6   Y5   Y4   Y3   Y2   Y1   Y0
```

Note that the device never signals overflow condition.

For protocol version 2 devices when the trackpoint is used, and no fingers are on the touchpad, the M R L bits signal the combined

status of both the pointingstick and touchpad buttons.

ALPS Absolute Mode - Protocol Version 1

```
byte 0:  1    0    0    0    1   x9   x8   x7
byte 1:  0   x6   x5   x4   x3   x2   x1   x0
byte 2:  0    ?    ?    l    r    ?  fin  ges
byte 3:  0    ?    ?    ?    ?   y9   y8   y7
byte 4:  0   y6   y5   y4   y3   y2   y1   y0
byte 5:  0   z6   z5   z4   z3   z2   z1   z0
```

ALPS Absolute Mode - Protocol Version 2

```
byte 0:  1    ?    ?    ?    1  PSM  PSR  PSL
byte 1:  0   x6   x5   x4   x3   x2   x1   x0
byte 2:  0  x10   x9   x8   x7    ?  fin  ges
byte 3:  0   y9   y8   y7    1    M    R    L
byte 4:  0   y6   y5   y4   y3   y2   y1   y0
byte 5:  0   z6   z5   z4   z3   z2   z1   z0
```

Protocol Version 2 DualPoint devices send standard PS/2 mouse packets for the DualPoint Stick. The M, R and L bits signal the combined status of both the pointingstick and touchpad buttons, except for Dell dualpoint devices where the pointingstick buttons get reported separately in the PSM, PSR and PSL bits.

Dualpoint device -- interleaved packet format

```
byte 0:    1    1    0    0    1    1    1    1
byte 1:    0   x6   x5   x4   x3   x2   x1   x0
byte 2:    0  x10   x9   x8   x7    0  fin  ges
byte 3:    0    0 YSGN XSGN    1    1    1    1
byte 4:   X7   X6   X5   X4   X3   X2   X1   X0
byte 5:   Y7   Y6   Y5   Y4   Y3   Y2   Y1   Y0
byte 6:    0   y9   y8   y7    1    m    r    l
byte 7:    0   y6   y5   y4   y3   y2   y1   y0
byte 8:    0   z6   z5   z4   z3   z2   z1   z0
```

Devices which use the interleaving format normally send standard PS/2 mouse packets for the DualPoint Stick + ALPS Absolute Mode packets for the touchpad, switching to the interleaved packet format when both the stick and the touchpad are used at the same time.

ALPS Absolute Mode - Protocol Version 3

ALPS protocol version 3 has three different packet formats. The first two are associated with touchpad events, and the third is associated with trackstick events.

The first type is the touchpad position packet:

```
byte 0:    1    ?   x1   x0    1    1    1    1
byte 1:    0  x10   x9   x8   x7   x6   x5   x4
byte 2:    0  y10   y9   y8   y7   y6   y5   y4
byte 3:    0    M    R    L    1    m    r    l
byte 4:    0   mt   x3   x2   y3   y2   y1   y0
byte 5:    0   z6   z5   z4   z3   z2   z1   z0
```

Note that for some devices the trackstick buttons are reported in this packet, and on others it is reported in the trackstick packets.

The second packet type contains bitmaps representing the x and y axes. In the bitmaps a given bit is set if there is a finger covering that position on the given axis. Thus the bitmap packet can be used for low-resolution multi-touch data, although finger tracking is not possible. This packet also encodes the number of contacts (f1 and f0 in the table below):

```
byte 0:    1    1   x1   x0    1    1    1    1
byte 1:    0   x8   x7   x6   x5   x4   x3   x2
byte 2:    0   y7   y6   y5   y4   y3   y2   y1
byte 3:    0  y10   y9   y8    1    1    1    1
byte 4:    0  x14  x13  x12  x11  x10   x9   y0
byte 5:    0    1    ?    ?    ?    ?   f1   f0
```

This packet only appears after a position packet with the mt bit set, and usually only appears when there are two or more contacts (although occasionally it's seen with only a single contact).

The final v3 packet type is the trackstick packet:

```
byte 0:    1    1   x7   y7    1    1    1    1
byte 1:    0   x6   x5   x4   x3   x2   x1   x0
byte 2:    0   y6   y5   y4   y3   y2   y1   y0
byte 3:    0    1   TP   SW    1    M    R    L
```

```
byte 4:    0   z6   z5   z4   z3   z2   z1   z0
byte 5:    0    0    1    1    1    1    1    1
```

TP means Tap SW status when tap processing is enabled or Press status when press processing is enabled. SW means scroll up when 4 buttons are available.

ALPS Absolute Mode - Protocol Version 4

Protocol version 4 has an 8-byte packet format:

```
byte 0:    1    ?   x1   x0    1    1    1    1
byte 1:    0  x10   x9   x8   x7   x6   x5   x4
byte 2:    0  y10   y9   y8   y7   y6   y5   y4
byte 3:    0    1   x3   x2   y3   y2   y1   y0
byte 4:    0    ?    ?    ?    1    ?    r    l
byte 5:    0   z6   z5   z4   z3   z2   z1   z0
byte 6:    bitmap data (described below)
byte 7:    bitmap data (described below)
```

The last two bytes represent a partial bitmap packet, with 3 full packets required to construct a complete bitmap packet. Once assembled, the 6-byte bitmap packet has the following format:

```
byte 0:    0    1   x7   x6   x5   x4   x3   x2
byte 1:    0   x1   x0   y4   y3   y2   y1   y0
byte 2:    0    0    ?  x14  x13  x12  x11  x10
byte 3:    0   x9   x8   y9   y8   y7   y6   y5
byte 4:    0    0    0    0    0    0    0    0
byte 5:    0    0    0    0    0    0    0  y10
```

There are several things worth noting here.

1. In the bitmap data, bit 6 of byte 0 serves as a sync byte to identify the first fragment of a bitmap packet.
2. The bitmaps represent the same data as in the v3 bitmap packets, although the packet layout is different.
3. There doesn't seem to be a count of the contact points anywhere in the v4 protocol packets. Deriving a count of contact points must be done by analyzing the bitmaps.
4. There is a 3 to 1 ratio of position packets to bitmap packets. Therefore MT position can only be updated for every third ST position update, and the count of contact points can only be updated every third packet as well.

So far no v4 devices with tracksticks have been encountered.

ALPS Absolute Mode - Protocol Version 5

This is basically Protocol Version 3 but with different logic for packet decode. It uses the same alps_process_touchpad_packet_v3 call with a specialized decode_fields function pointer to correctly interpret the packets. This appears to only be used by the Dolphin devices.

For single-touch, the 6-byte packet format is:

```
byte 0:    1    1    0    0    1    0    0    0
byte 1:    0   x6   x5   x4   x3   x2   x1   x0
byte 2:    0   y6   y5   y4   y3   y2   y1   y0
byte 3:    0    M    R    L    1    m    r    l
byte 4:  y10  y9   y8   y7  x10   x9   x8   x7
byte 5:    0   z6   z5   z4   z3   z2   z1   z0
```

For mt, the format is:

```
byte 0:    1    1    1   n3    1   n2   n1  x24
byte 1:    1   y7   y6   y5   y4   y3   y2   y1
byte 2:    ?   x2   x1  y12  y11  y10   y9   y8
byte 3:    0  x23  x22  x21  x20  x19  x18  x17
byte 4:    0   x9   x8   x7   x6   x5   x4   x3
byte 5:    0  x16  x15  x14  x13  x12  x11  x10
```

ALPS Absolute Mode - Protocol Version 6

For trackstick packet, the format is:

```
byte 0:    1    1    1    1    1    1    1    1
byte 1:    0   X6   X5   X4   X3   X2   X1   X0
byte 2:    0   Y6   Y5   Y4   Y3   Y2   Y1   Y0
byte 3:    ?   Y7   X7    ?    ?    M    R    L
byte 4:   Z7   Z6   Z5   Z4   Z3   Z2   Z1   Z0
byte 5:    0    1    1    1    1    1    1    1
```

For touchpad packet, the format is:

```
byte 0:    1    1    1    1    1    1    1    1
byte 1:    0    0    0    0   x3   x2   x1   x0
byte 2:    0    0    0    0   y3   y2   y1   y0
byte 3:    ?   x7   x6   x5   x4    ?    r    l
byte 4:    ?   y7   y6   y5   y4    ?    ?    ?
byte 5:   z7   z6   z5   z4   z3   z2   z1   z0
```

(v6 touchpad does not have middle button)

ALPS Absolute Mode - Protocol Version 7

For trackstick packet, the format is:

```
byte 0:    0    1    0    0    1    0    0    0
byte 1:    1    1    *    *    1    M    R    L
byte 2:   X7    1   X5   X4   X3   X2   X1   X0
byte 3:   Z6    1   Y6   X6    1   Y2   Y1   Y0
byte 4:   Y7    0   Y5   Y4   Y3    1    1    0
byte 5:  T&P    0   Z5   Z4   Z3   Z2   Z1   Z0
```

For touchpad packet, the format is:

```
        packet-fmt     b7     b6     b5     b4     b3     b2     b1     b0
byte 0: TWO & MULTI     L      1      R      M      1   Y0-2   Y0-1   Y0-0
byte 0: NEW             L      1   X1-5      1      1   Y0-2   Y0-1   Y0-0
byte 1:             Y0-10   Y0-9   Y0-8   Y0-7   Y0-6   Y0-5   Y0-4   Y0-3
byte 2:             X0-11      1  X0-10   X0-9   X0-8   X0-7   X0-6   X0-5
byte 3:             X1-11      1   X0-4   X0-3      1   X0-2   X0-1   X0-0
byte 4: TWO         X1-10    TWO   X1-9   X1-8   X1-7   X1-6   X1-5   X1-4
byte 4: MULTI       X1-10    TWO   X1-9   X1-8   X1-7   X1-6   Y1-5      1
byte 4: NEW         X1-10    TWO   X1-9   X1-8   X1-7   X1-6      0      0
byte 5: TWO & NEW   Y1-10      0   Y1-9   Y1-8   Y1-7   Y1-6   Y1-5   Y1-4
byte 5: MULTI       Y1-10      0   Y1-9   Y1-8   Y1-7   Y1-6    F-1    F-0

L:         Left button
R / M:     Non-clickpads: Right / Middle button
           Clickpads: When > 2 fingers are down, and some fingers
           are in the button area, then the 2 coordinates reported
           are for fingers outside the button area and these report
           extra fingers being present in the right / left button
           area. Note these fingers are not added to the F field!
           so if a TWO packet is received and R = 1 then there are
           3 fingers down, etc.
TWO:       1: Two touches present, byte 0/4/5 are in TWO fmt
           0: If byte 4 bit 0 is 1, then byte 0/4/5 are in MULTI fmt
              otherwise byte 0 bit 4 must be set and byte 0/4/5 are
              in NEW fmt
F:         Number of fingers - 3, 0 means 3 fingers, 1 means 4 ...
```

ALPS Absolute Mode - Protocol Version 8

Spoken by SS4 (73 03 14) and SS5 (73 03 28) hardware.

The packet type is given by the APD field, bits 4-5 of byte 3.

Touchpad packet (APD = 0x2):

```
          b7   b6   b5   b4   b3   b2   b1   b0
byte 0:  SWM  SWR  SWL    1    1    0    0   X7
byte 1:    0   X6   X5   X4   X3   X2   X1   X0
byte 2:    0   Y6   Y5   Y4   Y3   Y2   Y1   Y0
byte 3:    0  T&P    1    0    1    0    0   Y7
byte 4:    0   Z6   Z5   Z4   Z3   Z2   Z1   Z0
byte 5:    0    0    0    0    0    0    0    0
```

SWM, SWR, SWL: Middle, Right, and Left button states

Touchpad 1 Finger packet (APD = 0x0):

```
          b7   b6   b5   b4   b3   b2   b1   b0
byte 0:  SWM  SWR  SWL    1    1   X2   X1   X0
byte 1:   X9   X8   X7    1   X6   X5   X4   X3
byte 2:    0  X11  X10  LFB   Y3   Y2   Y1   Y0
byte 3:   Y5   Y4    0    0    1 TAPF2 TAPF1 TAPF0
byte 4:  Zv7  Y11  Y10    1   Y9   Y8   Y7   Y6
byte 5:  Zv6  Zv5  Zv4    0  Zv3  Zv2  Zv1  Zv0
```

TAPF: ??? LFB: ???

Touchpad 2 Finger packet (APD = 0x1):

```
            b7   b6   b5   b4   b3   b2   b1   b0
byte 0:  SWM  SWR  SWL    1    1  AX6  AX5  AX4
byte 1: AX11 AX10  AX9  AX8  AX7  AZ1  AY4  AZ0
byte 2: AY11 AY10  AY9  CONT AY8  AY7  AY6  AY5
byte 3:    0    0    0    1    1  BX6  BX5  BX4
byte 4: BX11 BX10  BX9  BX8  BX7  BZ1  BY4  BZ0
byte 5: BY11 BY10  BY9    0  BY8  BY7  BY5  BY5
```

CONT: A 3-or-4 Finger packet is to follow

Touchpad 3-or-4 Finger packet (APD = 0x3):

```
            b7   b6   b5   b4   b3   b2   b1   b0
byte 0:  SWM  SWR  SWL    1    1  AX6  AX5  AX4
byte 1: AX11 AX10  AX9  AX8  AX7  AZ1  AY4  AZ0
byte 2: AY11 AY10  AY9  OVF  AY8  AY7  AY6  AY5
byte 3:    0    0    1    1    1  BX6  BX5  BX4
byte 4: BX11 BX10  BX9  BX8  BX7  BZ1  BY4  BZ0
byte 5: BY11 BY10  BY9    0  BY8  BY7  BY5  BY5
```

OVF: 5th finger detected