Subject: Re: Using Pen with U++
Posted by mirek on Sat, 27 Mar 2021 19:23:43 GMT
View Forum Message <> Reply to Message

Tom1 wrote on Sat, 27 March 2021 12:20mirek wrote on Sat, 27 March 2021 10:11Tom1 wrote on Fri, 26 March 2021 18:52
However, e.g. RectTracker does not work as is with my Wacom tablet. The specific pen support (I just posted) is simply required for it to work. (I'm now under impression that RectTracker works with your tablet perfectly without these changes... ?? Perhaps XP-PEN drivers can avoids Windows Ink and work more mouse-like through the stack.)

Sorry, I might get lost in the code that we exchange here a lot...

Do you mean that my variant where Pen returns false and RectTracker is called in normal LeftDown does not work with Wacom? (My impression so far was that you are only testing RectTracker in Pen virtual method).

If so, does it mean RectTracker does not work even in pre-Pen U++? (With Wacom in "mouse mode")

Mirek

Hi,

With original pre-Pen U++, RectTracker worked with Wacom at correct coordinates. (It only suffered from the Windows Ink induced 2 cm starting delay, but the rectangle followed pen thereafter.)

With current trunk code and testing with:

```
bool Pen(Point p, const PenInfo& pn, dword keyflags) override {
  if(keyflags & K_SHIFT)
   return false;
  if(pn.pressure) {
   if((!!pn.pressure == !!pen.pressure) && drawing.GetCount())
    drawing.Top().Add(MakeTuple(pn.pressure, p));
   else
    drawing.Add().Add(MakeTuple(pn.pressure, p));
  }
  pen = pn;
  Refresh();
  return true;
 }

 void LeftDown(Point p, dword keyflags) override {
  if(keyflags & K_SHIFT) {
   RectTracker tracker(*this);
```

```
  tracker.MinSize(Size(-100000,-100000));
  tracker.Track(Rect(p,p));
 }
}
```

the tracking rect still of course has the 2 cm starting delay. However, now the rect dragged corner lags behind with a 2 cm constant dx,dy offset. Please see below a photo of a live 'while dragging' situation. So, actually, RectTracker is now worse than before introducing pen support:

Further on, what I really need with RectTracker, is that there is no 2 cm reaction delay. The tracking must start immediately from PEN_DOWN event. My RectTracker inherited sketching code requires immediate reaction for even a single pixel pen drags. This is why I need specific Pen() support in RectTracker.

Best regards,

Tom

OK, this works just fine with xp-pen...

Widgets in general do work? (Because if it is this way, I would just expect the offset to be everywhere...)

What happens if you remove return 0L in pen processing?

Mirek
