AllTimer Update

Frank Suits

Outline

- General needs of video timing
- Simple ways to do it
- Challenges of USB/Firewire video cameras
- Benefits of modular components
- Current AllTimer features
- Examples

Two aspects of video timing

- Which second did an event take place?
- What was the fraction of a second?
  - It's not too hard to do one or the other, but doing both at the same time is more challenging
  - You can usually use some manual method such as setting the videocamera clock accurately to know time to the nearest second
  - But something automatic is preferred given all the things that can go wrong in the heat of battle

Simplest timing

- Blinking LED attached to PPS output of GPS module
- Dangle LED in front of telescope before and after event
- Mark the minute using a flash, or get time from accurately set video clock – to know “which second”
- Low cost, reliable, some manual effort, works with analog and usb video cameras, all scopes

Other options for analog cameras

- If you don't mind running analysis software such as LiMovie, then don't need readable text to know time
- Just have bar at bottom of screen indicating fraction of second, and another indicating “which” second
- Just needs GPS and microcontroller to insert bar
- Easier to code and “read” than text on screen

AllTimer current version

- Main target is digital video cameras, e.g. USB
- But analog also supported
- Also accurately sets PC clock in the field using USB and GPS
- Includes “Occultation” LED for self-checks of timing accuracy
- Marks time on USB cameras with projected spot that ramps up and down each second
- “Which Second” shown by notches in the curve
- Continuous reading of GPS NMEA stream
- Venus GPS has internal clock and battery backup

Modular construction

PIC18F4550 mcontroller

Venus 634 GPS module

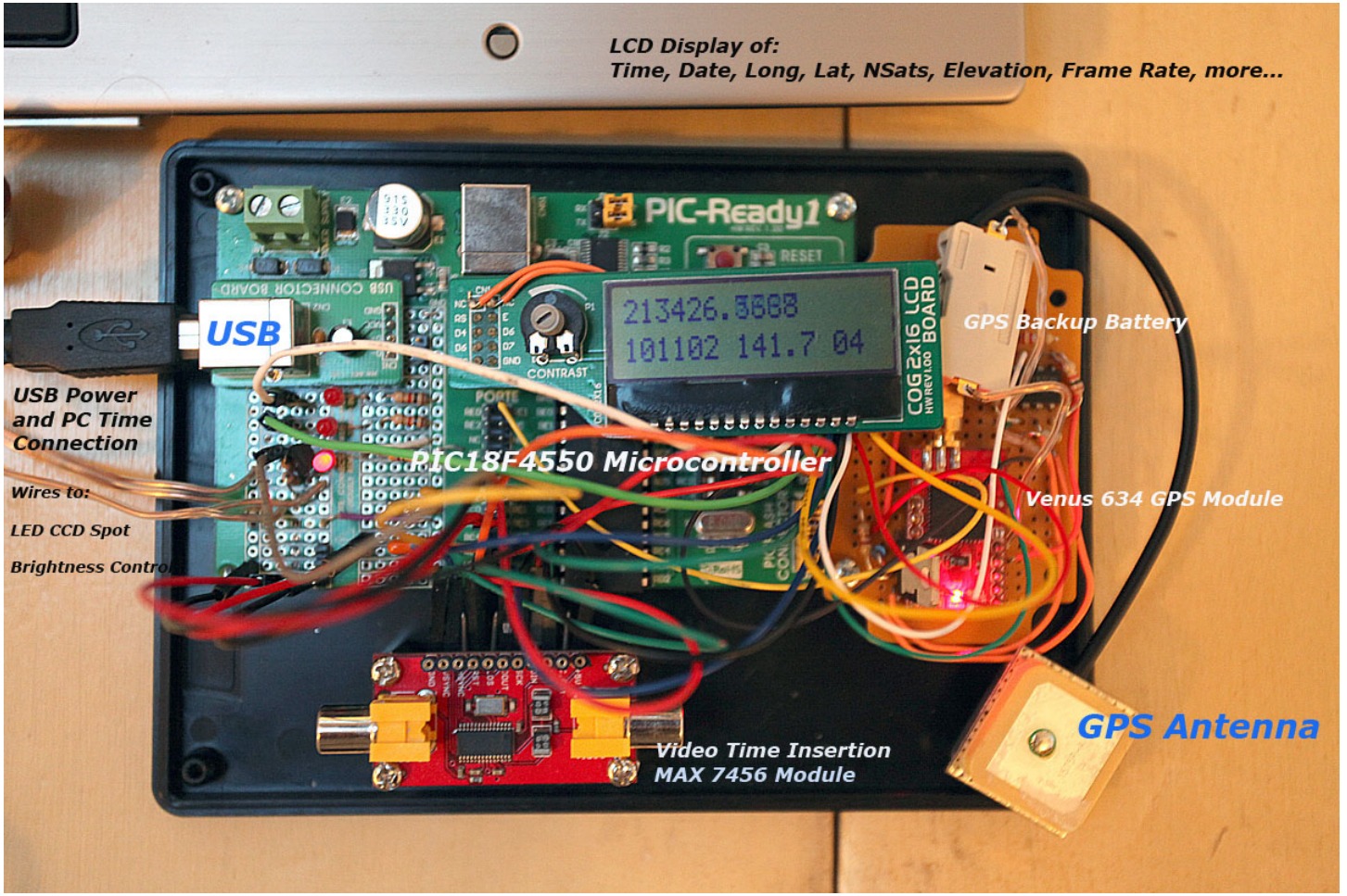
MAX 7456 Text insertion

PIC Ready1 board

GPS Antenna

LCD display

Power from USB



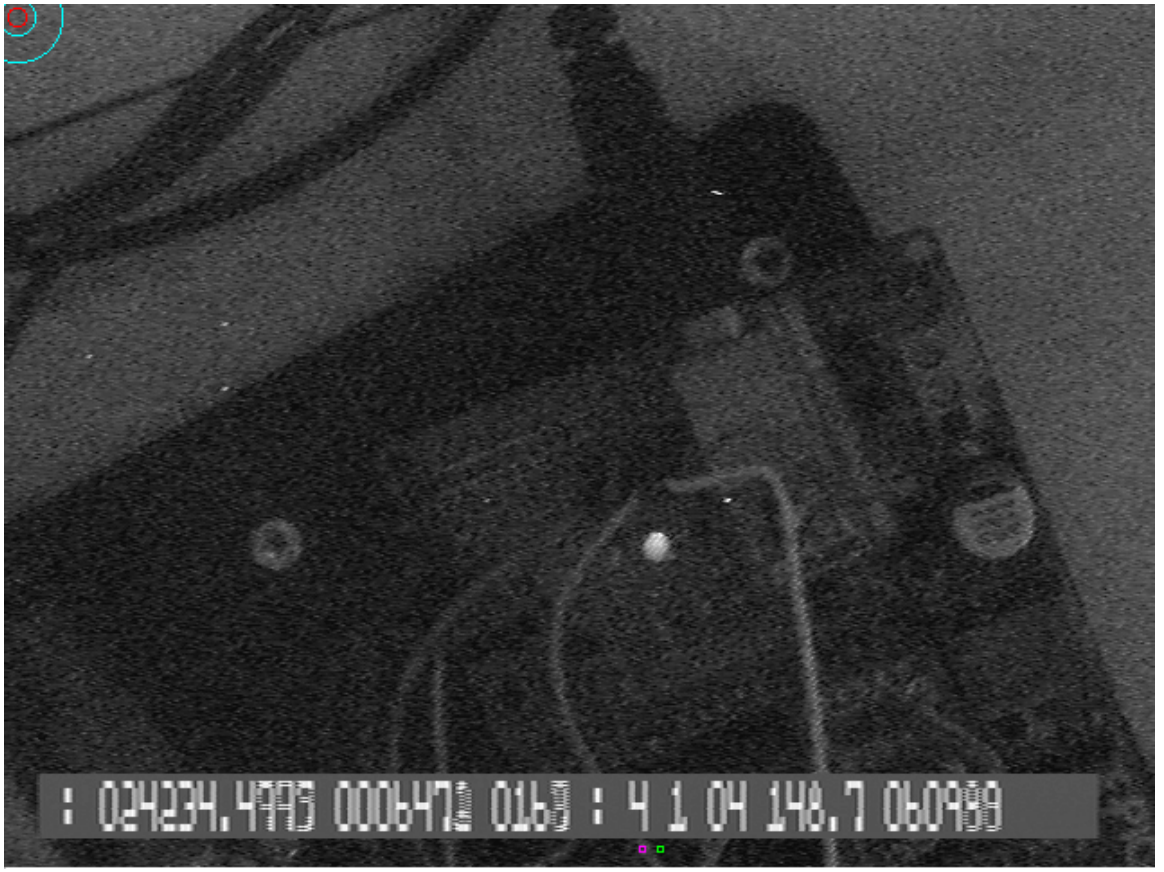
Ugly but effective. Simple parts strung together with wires under the control of the PIC

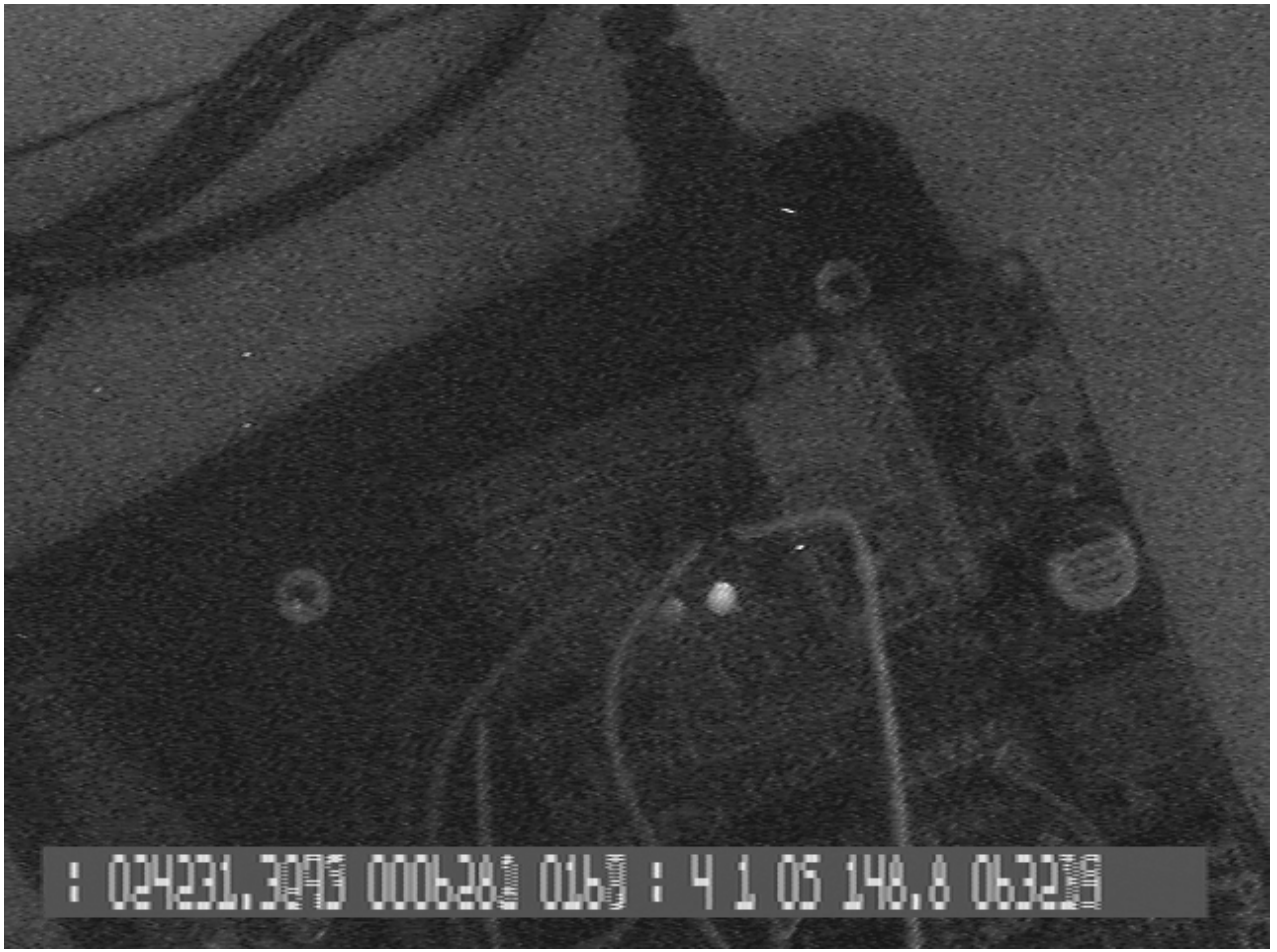
Analog output, frame view

Text shows standard items such as time and field count

Also cycles through items such as:
Nsatellites
Lat/Long
Elevation
… and more

Characters at half normal size using special character set since MAX text is large




: 024231.3075 0006287 0163 : 4 1 05 148.8 063275

Field view

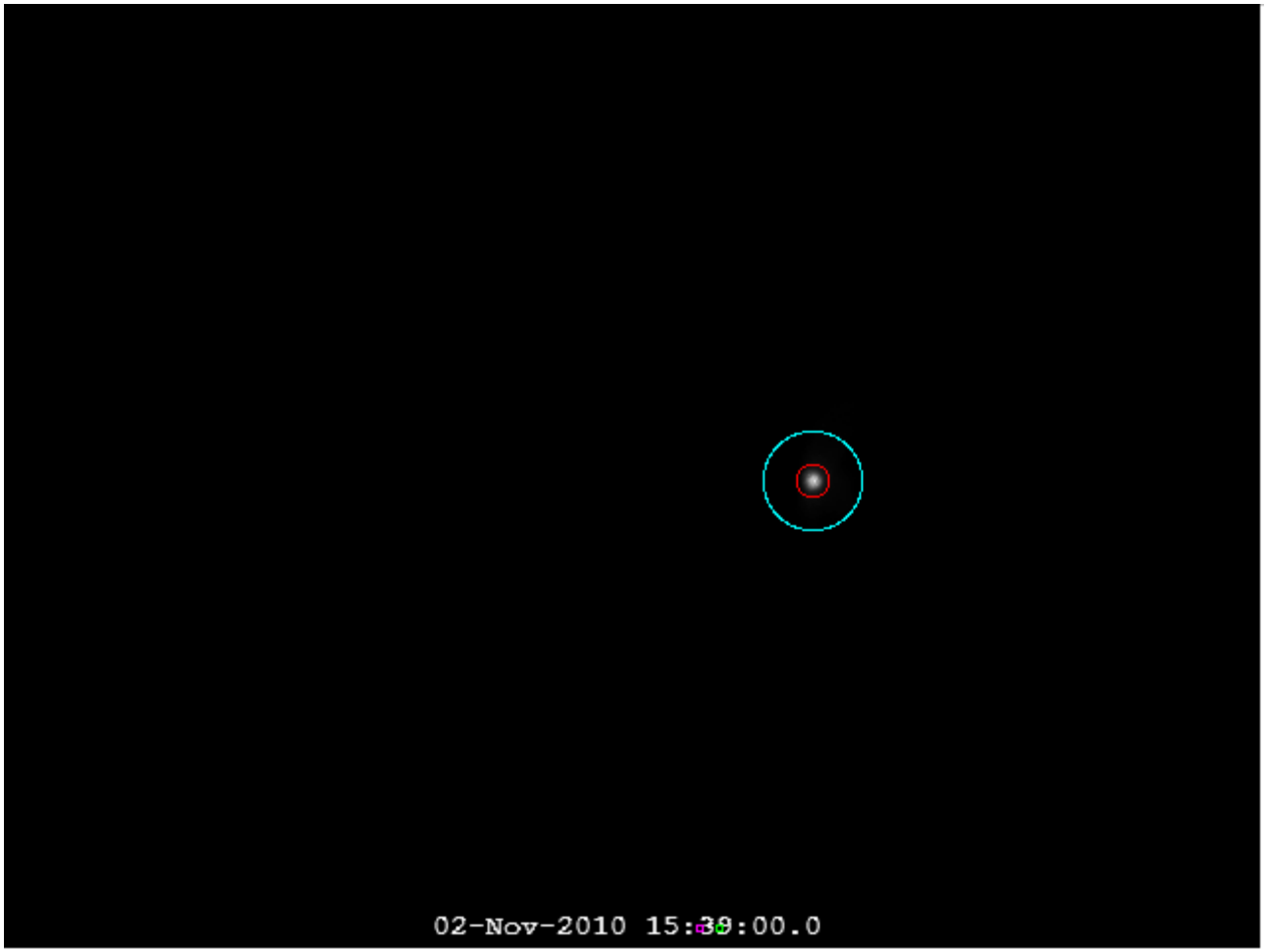
Character set designed to be easily read by software in each field

Optical time inserter for USB cameras

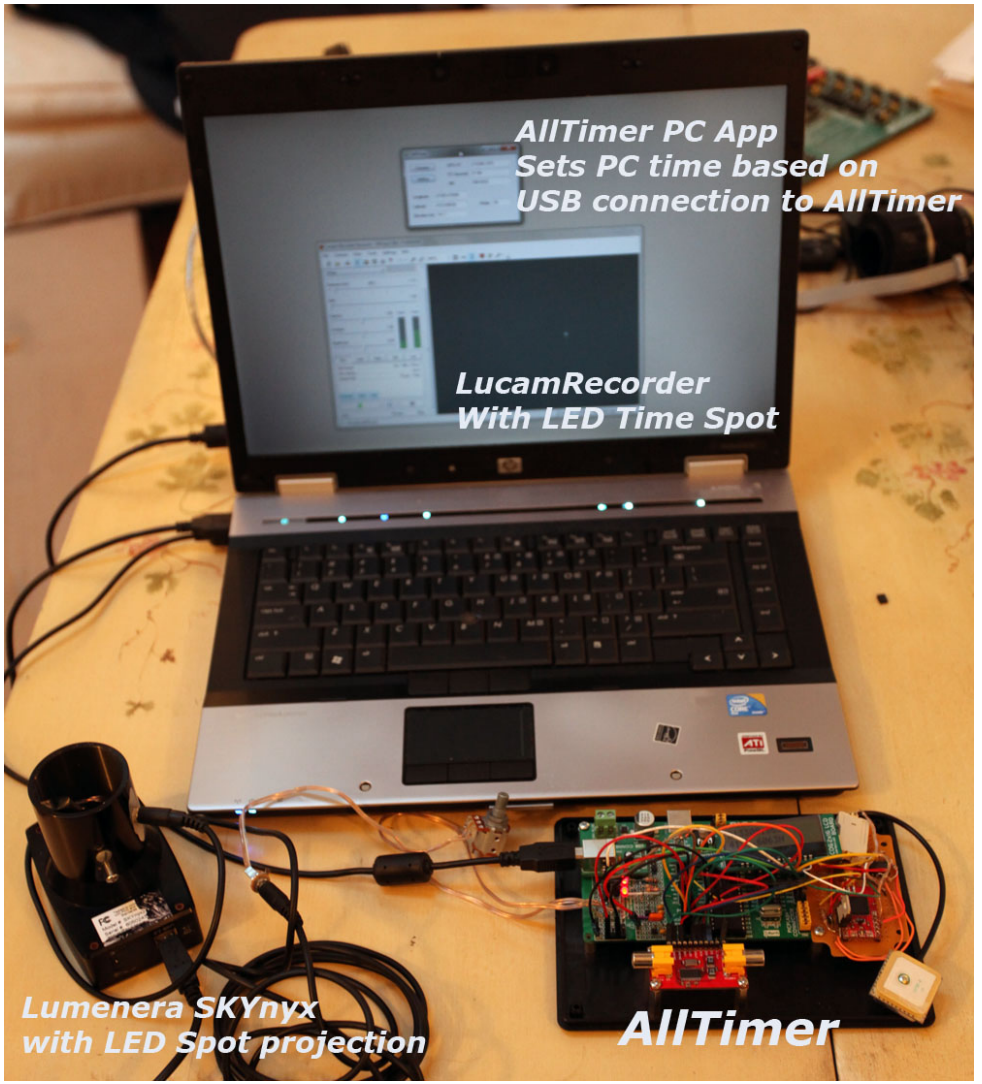
Pinhole projected onto CCD through small lens

Front view with Lumenera USB video camera

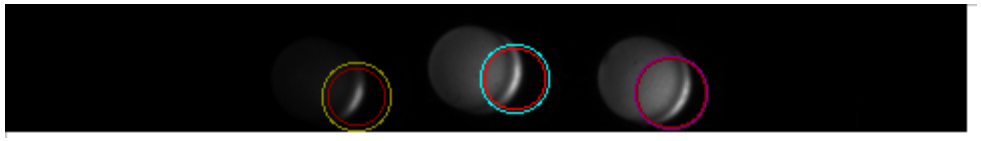
View of LED spot in frame

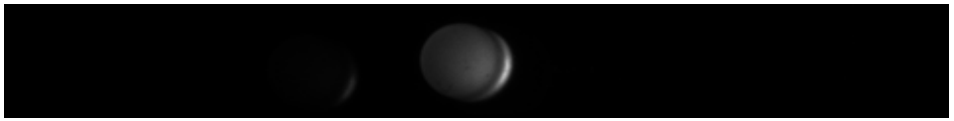


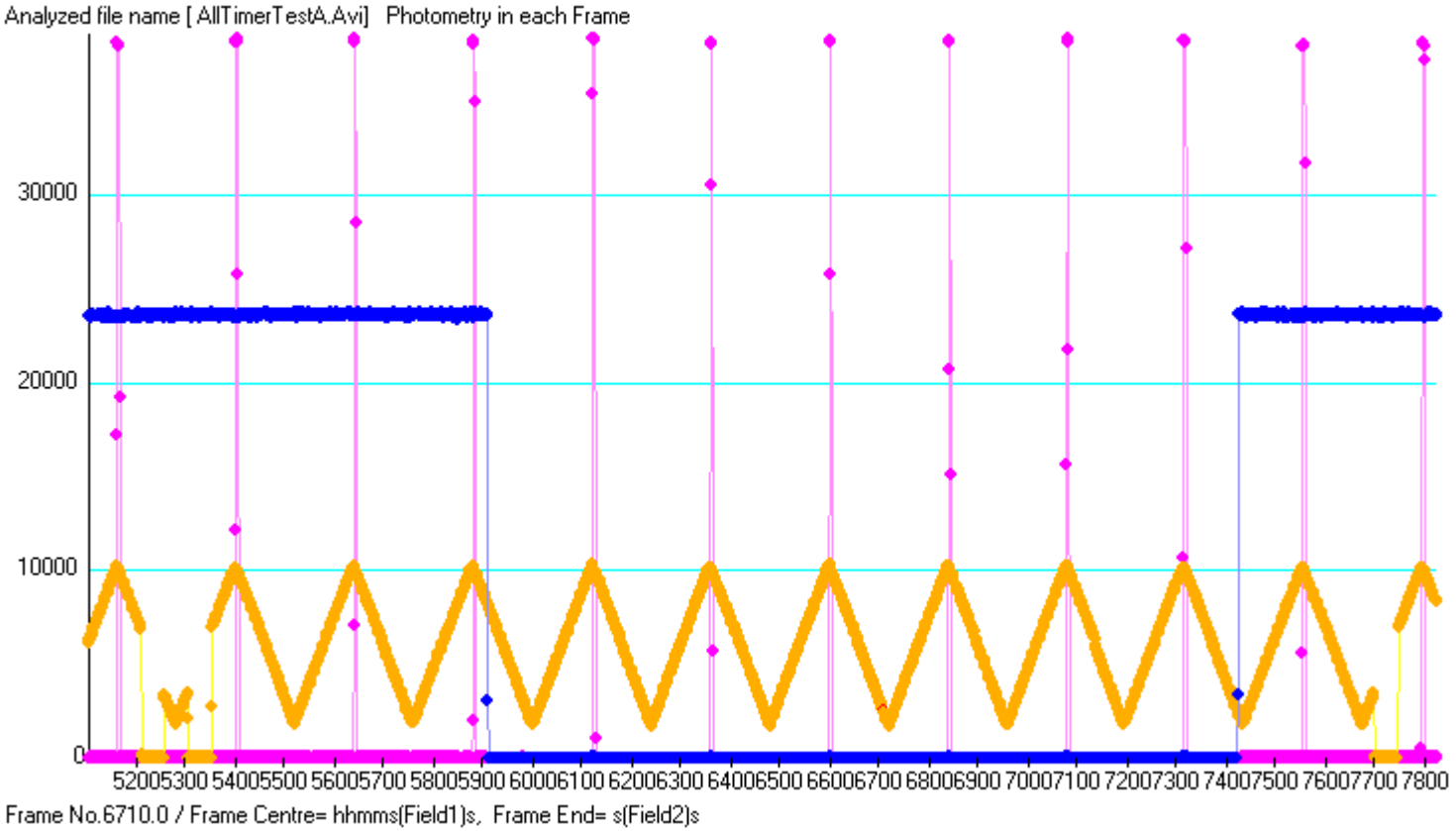
Complete setup



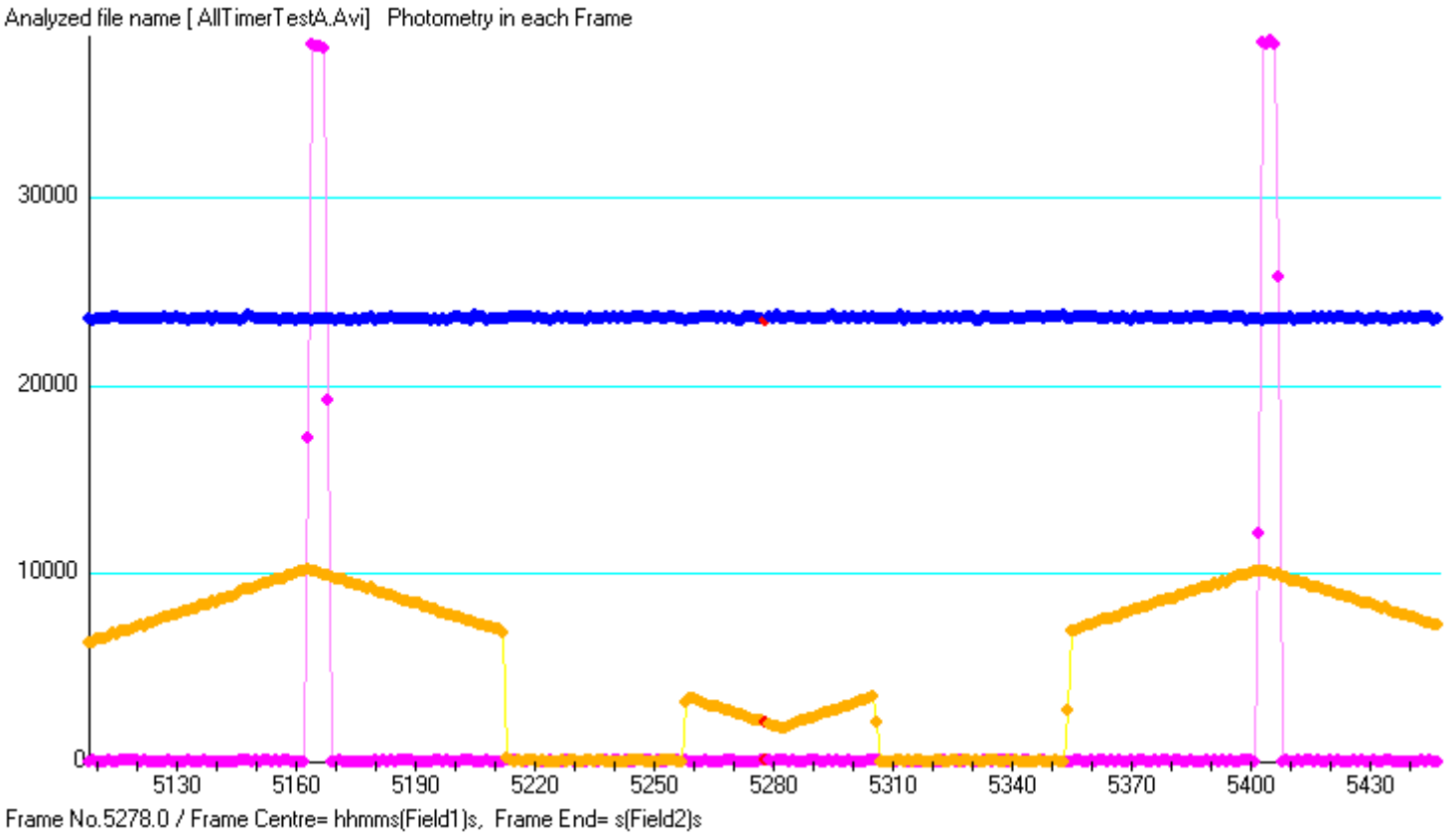
LED outputs: ramp with minute, 10s notches, and simulated occultation



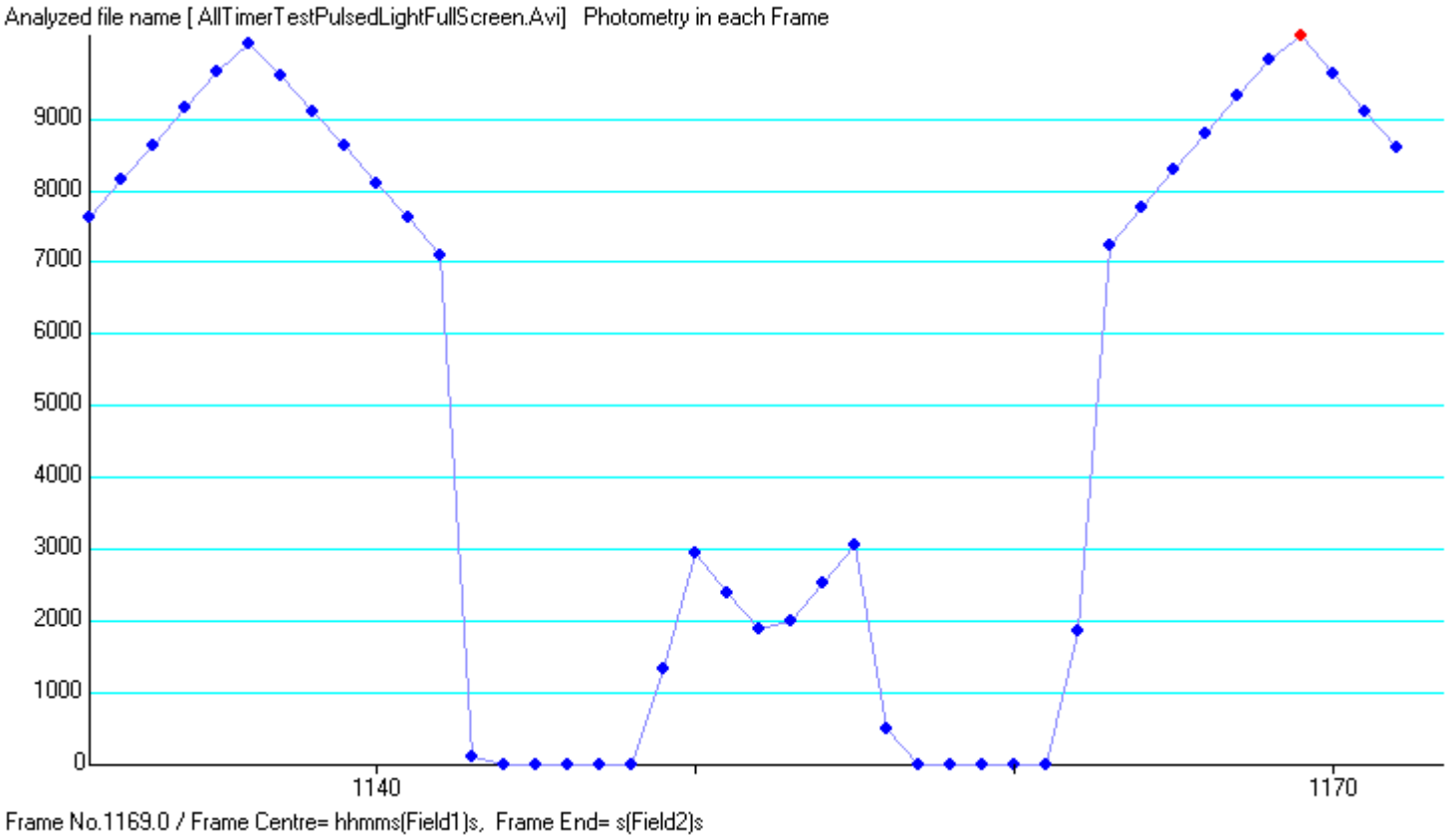




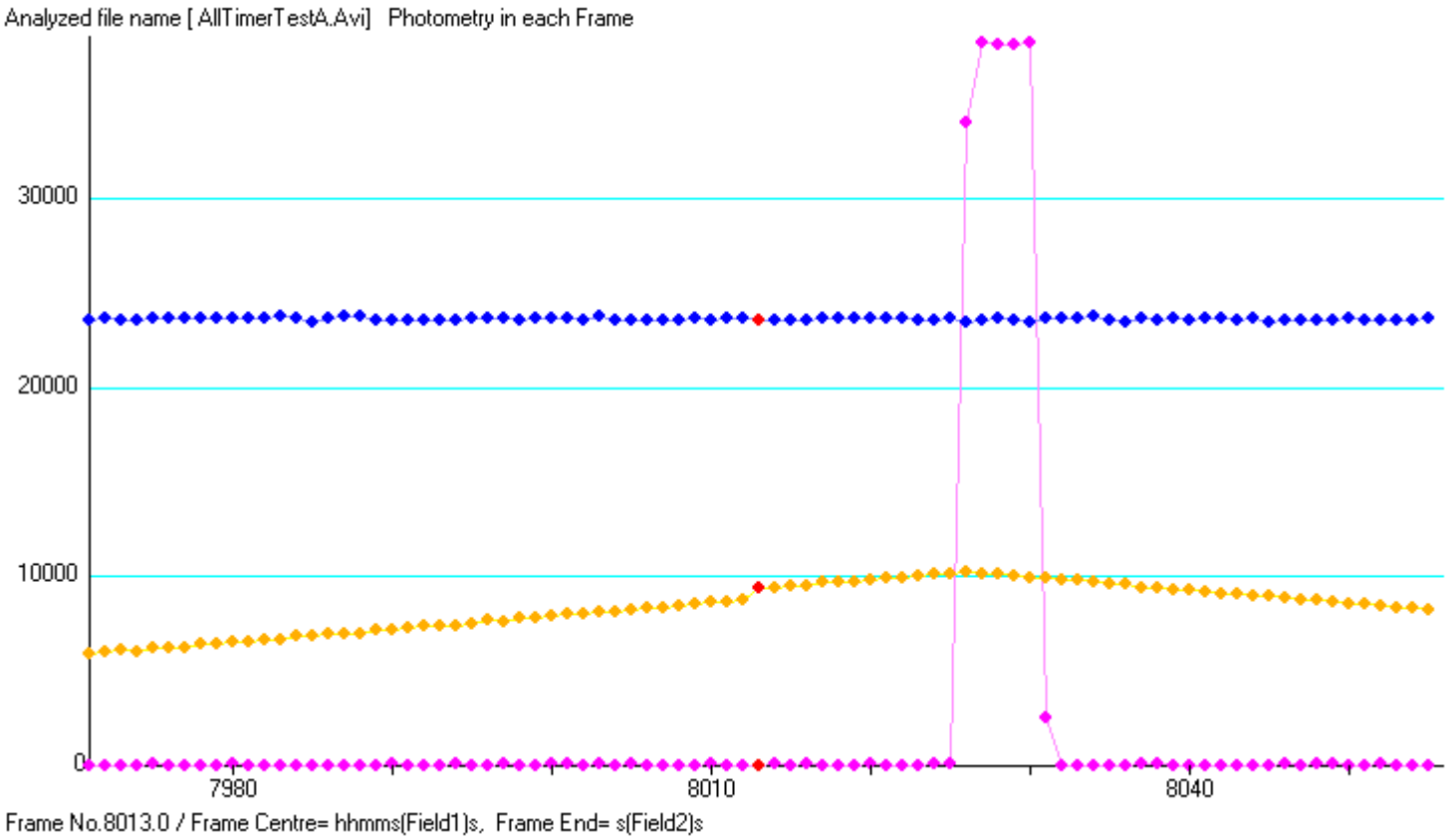
Close up of minute notch at 245 fps



Minute notch at 30 fps



Ramp reveals dropped frames
Dropped frames can be detected and corrected with no impact on timing



USB PC clock setting via GPS/PPS

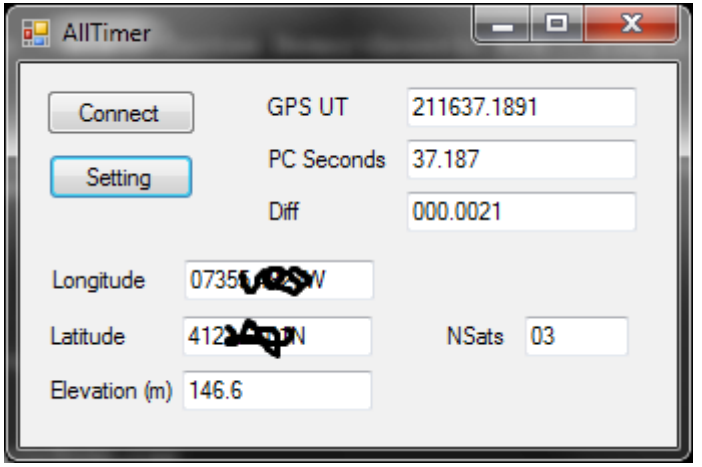
AllTimer sends exact time as text every 2 seconds

Lightweight utility waits for this message and immediately sets time

Accurate to 5-10ms with no cpu burden

Useful for many applications outside of occultations

Adequate accuracy for many events without need for the LED spot



AllTimer timing app running on PC
Reading time and satellite info over USB

Remarks

Main microcontroller options are PIC and Arduino

Very similar, but PIC has advantage for direct USB connection

Next version will be much smaller

Could be extremely small with discrete components and custom circuit board

Could mix and match each of the features:

- On screen display for analog cameras not needed if main usage is for USB cameras
- Don't need LED or text if just setting PC time in the field

But if you just want to time occultations with analog video – could just have bars on screen telling you the time, with no text

Thanks!