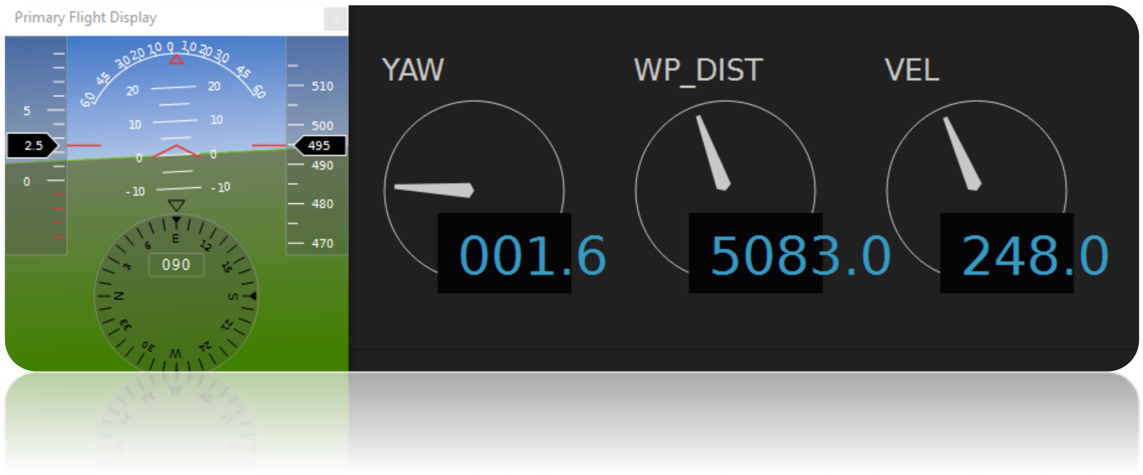
Custom Gauges in QgroundControl_AQ

Summary by afernan. Oct 16.

Based on QGC1.7beta3



QGC > Open Tool Widgets > Custom Gauges

To create new gauge: (click right mouse button)

Format

Let say we´re monitoring the vehicle: **MAV 129**

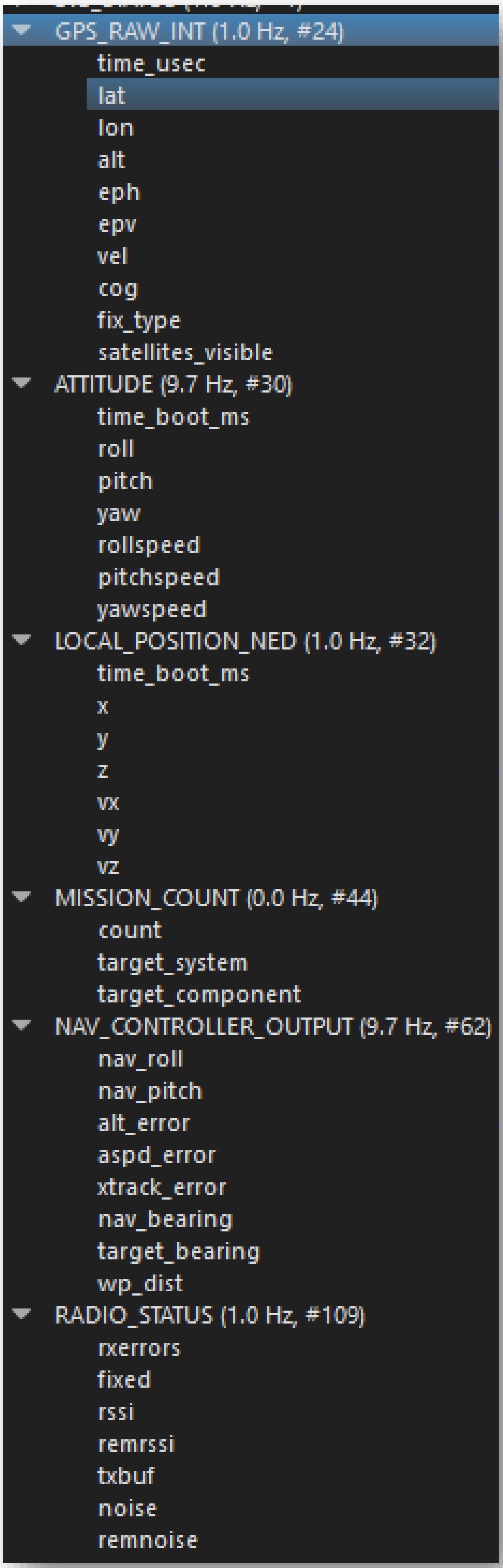
```
min,M129:message.type,format,max,tittle
```

Example:

to show a "yaw" gauge from 0-360 deg (0 – 6.28 rad)

```
0,M129:ATTITUDE.yaw,float,6.28,Yaw
```

We can monitor any sensor received in a MAVLINK message

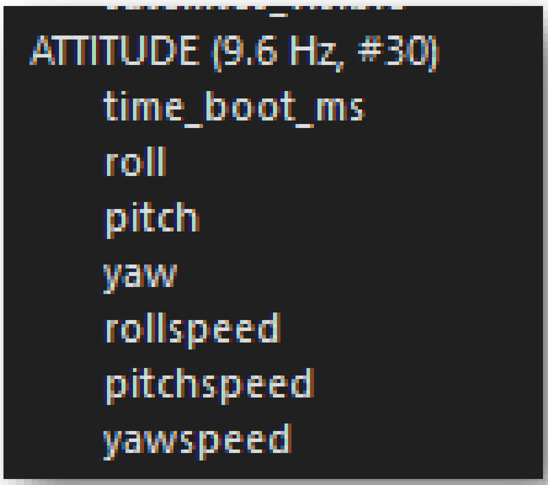


- GPS_RAW_INT (1.0 Hz, #24)
  - time_usec
  - lat
  - lon
  - alt
  - eph
  - epv
  - vel
  - cog
  - fix_type
  - satellites_visible
- ATTITUDE (9.7 Hz, #30)
  - time_boot_ms
  - roll
  - pitch
  - yaw
  - rollspeed
  - pitchspeed
  - yawspeed
- LOCAL_POSITION_NED (1.0 Hz, #32)
  - time_boot_ms
  - x
  - y
  - z
  - vx
  - vy
  - vz
- MISSION_COUNT (0.0 Hz, #44)
  - count
  - target_system
  - target_component
- NAV_CONTROLLER_OUTPUT (9.7 Hz, #62)
  - nav_roll
  - nav_pitch
  - alt_error
  - aspd_error
  - xtrack_error
  - nav_bearing
  - target_bearing
  - wp_dist
- RADIO_STATUS (1.0 Hz, #109)
  - rxerrors
  - fixed
  - rssi
  - remrssi
  - txbuf
  - noise
  - remnoise

ATTITUDE

Notes:

- The format must match the data
- If any wrong data is set, it will display “000”
- Don´t put spaces between commas



ATTITUDE (9.6 Hz, #30)

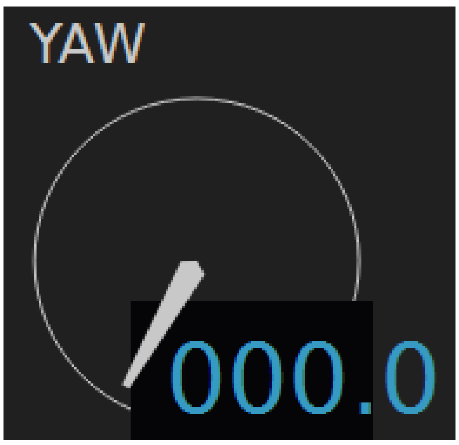
    time_boot_ms
    roll
    pitch
    yaw
    rollspeed
    pitchspeed
    yawspeed

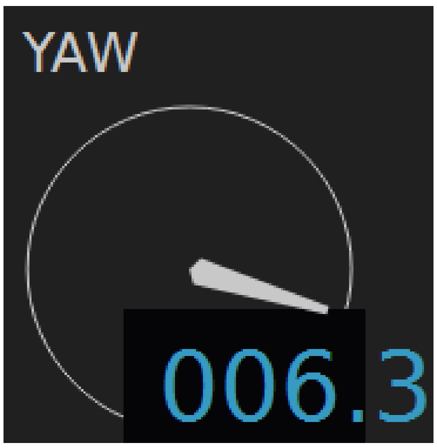
Yaw

```
0,M129:ATTITUDE.yaw,float,6.28,yaw
```

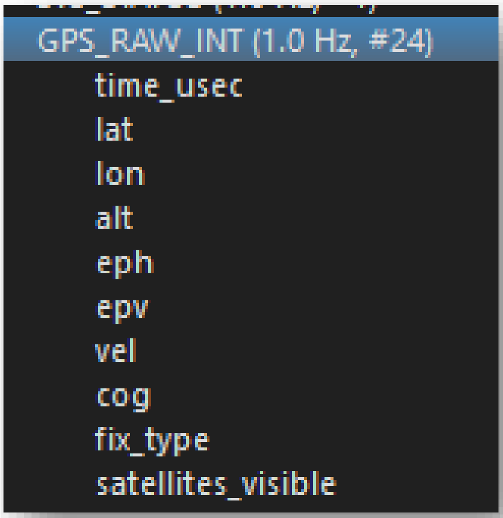
0deg = 0rad

359deg = 6.28rad



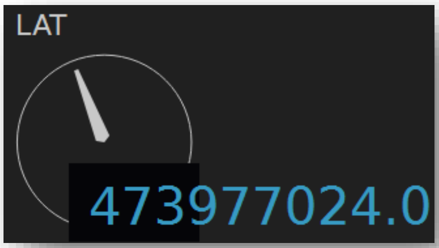


GPS_RAW_INT


GPS_RAW_INT (1.0 Hz, #24)
time_usec
lat
lon
alt
eph
epv
vel
cog
fix_type
satellites_visible

Lat, Lon

```
473975000,M129:GPS_RAW_INT.lat,int32_t, 473975000,lat
```

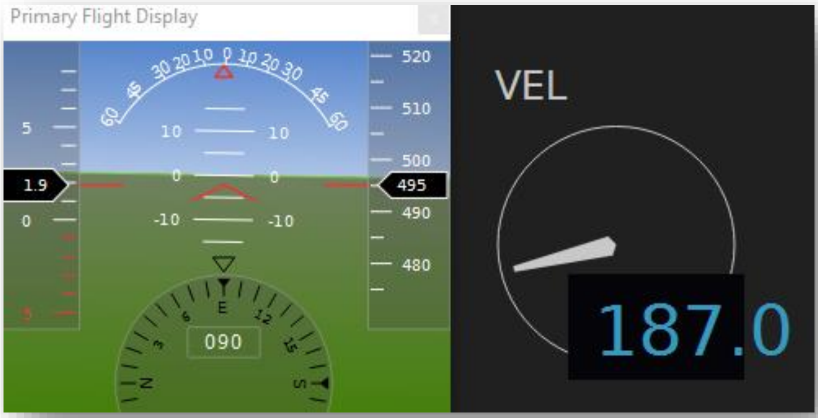


Velocity (cm/s)

To setuo a vel gauge from 0-5 m/s

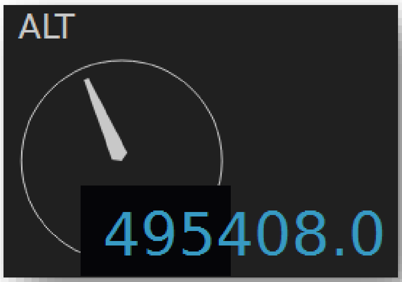
```
0,M129:GPS_RAW_INT.vel,uint16_t,500,vel
```

```
Example: flying at  1.9 m/s
```



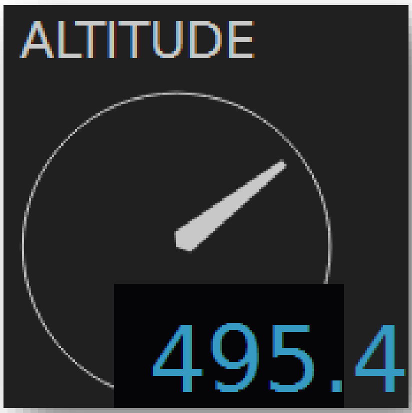
Altitude in mm

```
0,M129:GPS_RAW_INT.alt,int32_t,+100,alt
```



Altitude in m

```
0,altitude,m,+100,altitude
```



NAV_CONTROLLER_OUTPUT

wp_dist (cm)

to set a gauge from 0-100m:

```
0,M129:NAV_CONTROLLER_OUTPUT.wp_dist,int32_t,10000,WP_DIS
T
```

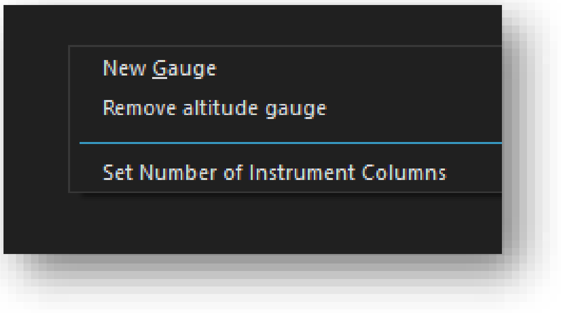
QGC source code

Implementation of Head Down Display (HDD)

HDDisplay.cc, line 302

```
    QString item = QInputDialog::getItem(this, tr("Add Gauge Instrument"),

                  tr("Format: min, data name, unit, max, label [,s]"), items, 0, true, &ok);

    if (ok && !item.isEmpty()) {

        addGauge(item);

    `
```

To create new gauge: (click right mouse button)



```
void HDDisplay::addGauge()
{
    QStringList items;
    for (int i = 0; i < values.count(); ++i) {
        QString key = values.keys().at(i);
        QString label = key;
        QStringList keySplit = key.split(".");
        if (keySplit.size() > 1)
        {
            keySplit.removeFirst();
            label = keySplit.join(".");
        }
        QString unit = units.value(key);
// si la gauge es de un ángulo, con unit = deg o rad, esto de abajo le añade al string 180°
        if (unit.contains("deg") || unit.contains("rad")) {
            items.append(QString("%1,%2,%3,%4,%5,s").arg("-180").arg(key).arg(unit).arg("+180").arg(label));
        } else {
            items.append(QString("%1,%2,%3,%4,%5").arg("0").arg(key).arg(unit).arg("+100").arg(label));
        }
    }
    bool ok;
    QString item = QInputDialog::getItem(this, tr("Add Gauge Instrument"),
                                         tr("Format: min, data name, unit, max, label [,s]"), items, 0, true,
&ok);
    if (ok && !item.isEmpty()) {
        addGauge(item);
    }
```

Data input to create the gauge:

```
void HDDisplay::addGauge(const QString& gauge)
{
    if (gauge.length() > 0) {
        QStringList parts = gauge.split(',');// leemos las partes de la gauge que
hemos definido y las asignamos a parts.at(1), etc

// parts.at(0) = min
// parts.at(1) = "data name" o key
// parts.at(2) = unit
```