CAN We Talk Python?

Talking to your vehicle over the CAN bus with Python

@shnewto

CAN?


PEAK

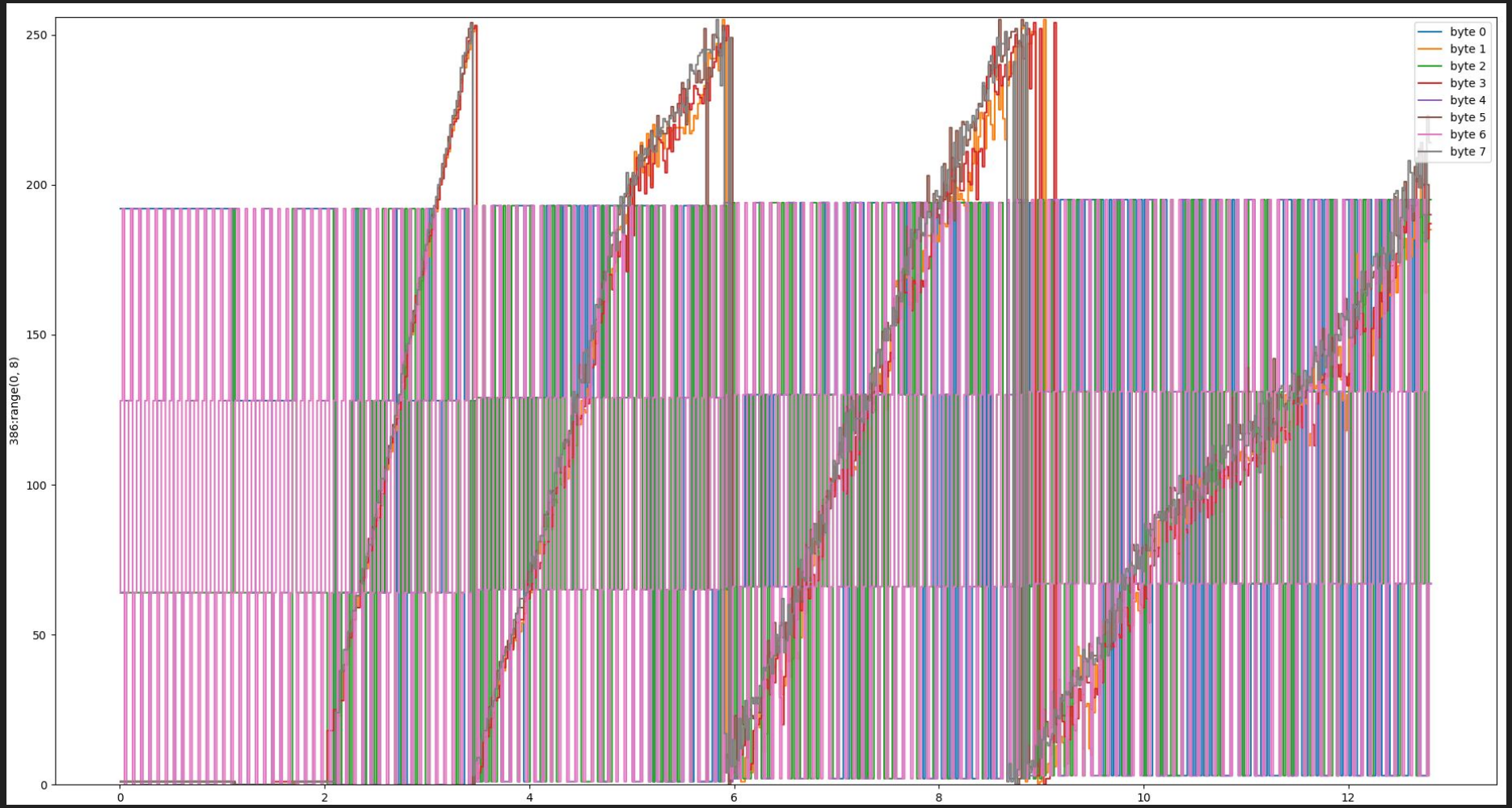
The standard CAN frame (mostly)

ID: 0x123

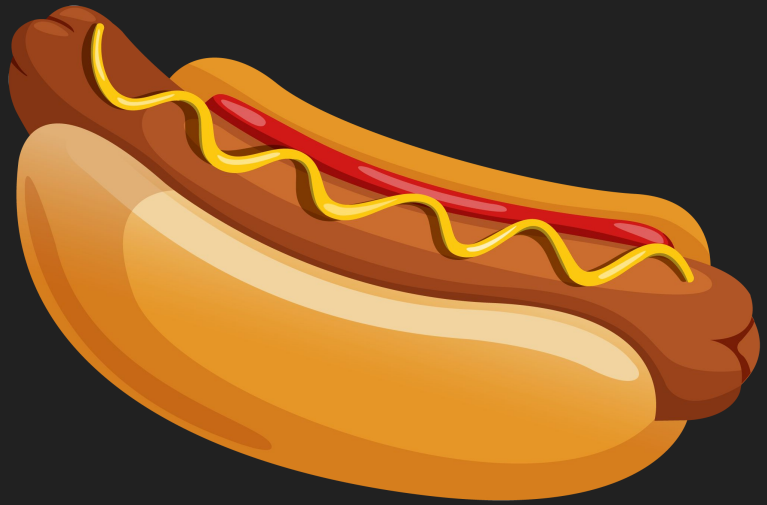
DLC: 8

Data: 8 bytes / 64 bits

```
[0] 0x00 0x11
[1] 0x22 0x33
[2] 0x44 0x55
[3] 0x66 0x77
[4] 0x88 0x99
[5] 0xAA 0xBB
[6] 0xCC 0xDD
[7] 0xEE 0xFF
```


byte 0
byte 1
byte 2
byte 3
byte 4
byte 5
byte 6
byte 7
386:range(0, 8)
0
50
100
150
200
250
0
2
4
6
8
10
12

```
byte1 = (data[offset + 1] & 0x0F) << 8
byte0 = data[offset]
value = byte1 | byte0
wheel_speed = value / 10.0
```



The OBD-II Port

(My 2006 Hyundai Elantra)

A LOT of chatter!

can_we_talk.py

github.com/shnewto/can-we-talk

```python
# pip install python-can
import can

bus = can.interface.Bus(
    bustype='socketcan_native', channel='can0', bitrate=500000)

pid = 0x2F  # fuel tank level

query_frame = [0x02, 0x01, pid, 0x55, 0x55, 0x55, 0x55, 0x55]

query_arbitration_id = 0x7DF # Defined by OBD-II standard

msg = can.Message(
    arbitration_id=query_arbitration_id, data=query_frame, extended_id=False)

bus.send(msg)

reply_arbitration_id = 0x7E8 # Defined by OBD-II standard

reply = bus.recv(timeout)

if (reply is not None) and (reply.arbitration_id == reply_arbitration_id):
    print(reply.data)
```

oscc-check.py

github.com/PolySync/oscc-check

```python
import oscccan

bus = oscccan.CanBus()

report = bus.check_wheel_speed()

print(report.value)
```

The CAN bus and your car

Questions?

Resources

sheas.blog/talks (slide deck)

Code:

- github.com/shnewto/can-we-talk OBD-II reader
- github.com/PolySync/oscc-check Vehicle actuation and status reports

Twitter: @shnewto
Blog: sheas.blog
GitHub: shnewto
Email: shnewto@gmail.com