FPGA-Based Feedback for Cooling the Motion of Levitated Nanospheres

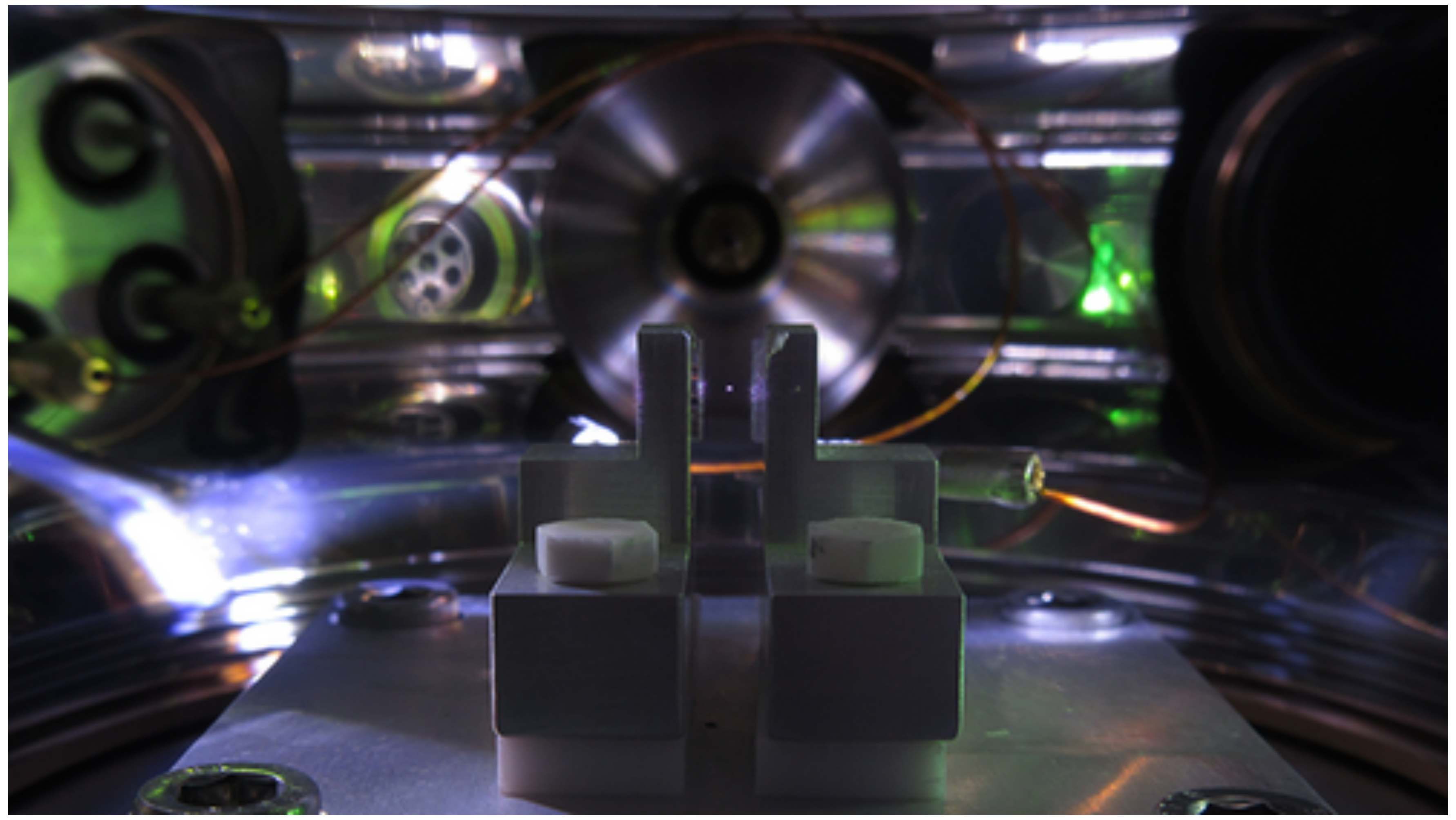
Harrison Liang 7/25/2025

Supervised by Prof. David Moore and Yu-Han Tseng

Introduction

- Levitated objects in everyday life
- Trap optically-levitated micro-objects in vacuum
- Field Programmable Gate Arrays (FPGAs)
- Real-time quantum control to slow the spheres

~300 m



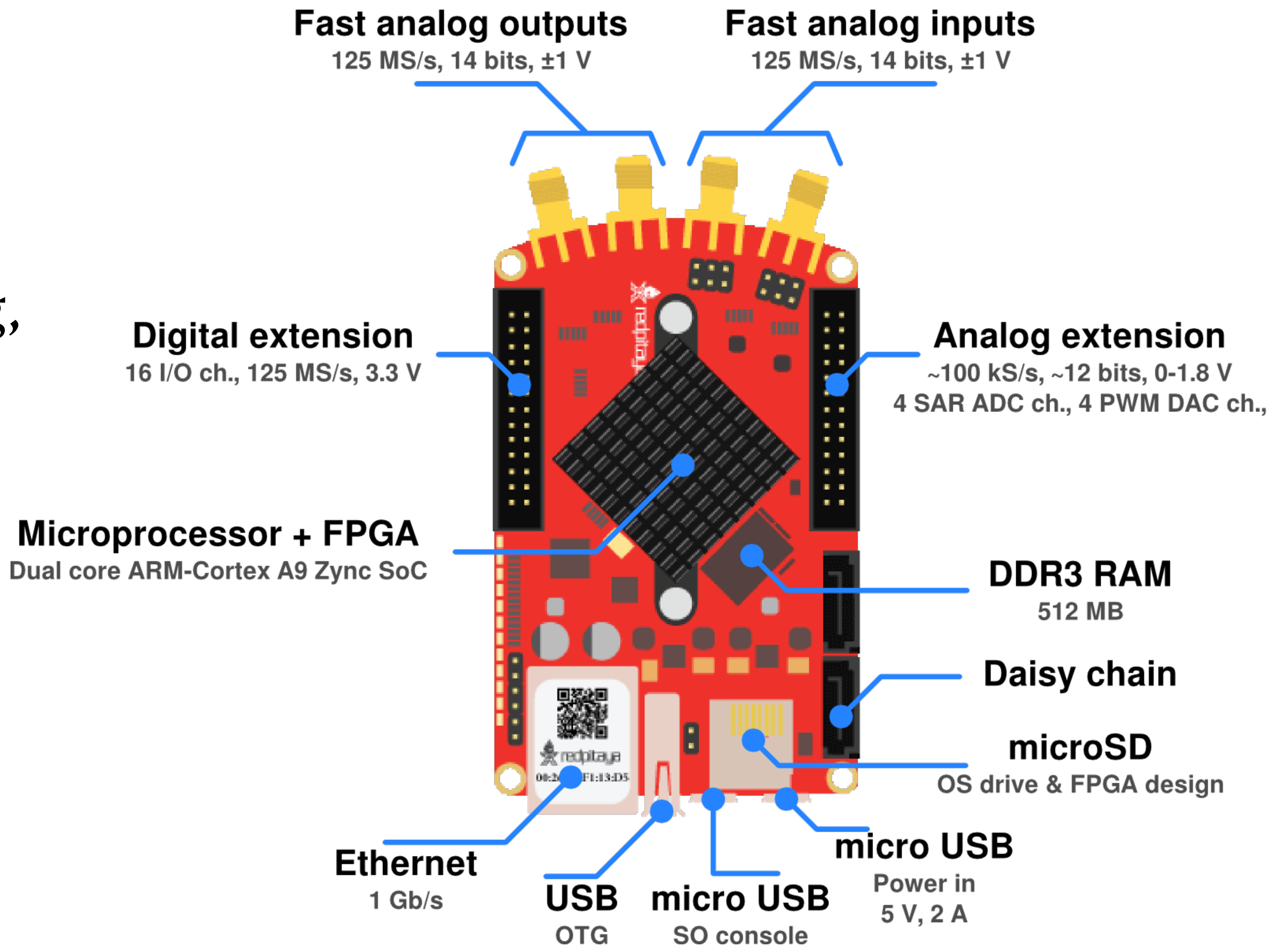
~100 nm

Motivation

- Search for physics beyond the SM
- Developing better sensors — reduction of noise + better stability
- Test fundamental laws at micro-scales
- Test Q.M. for macroscopic particles
- Astrophysical experiments: seek for DM + grav. waves @ high freq.

Red Pitaya (RP)

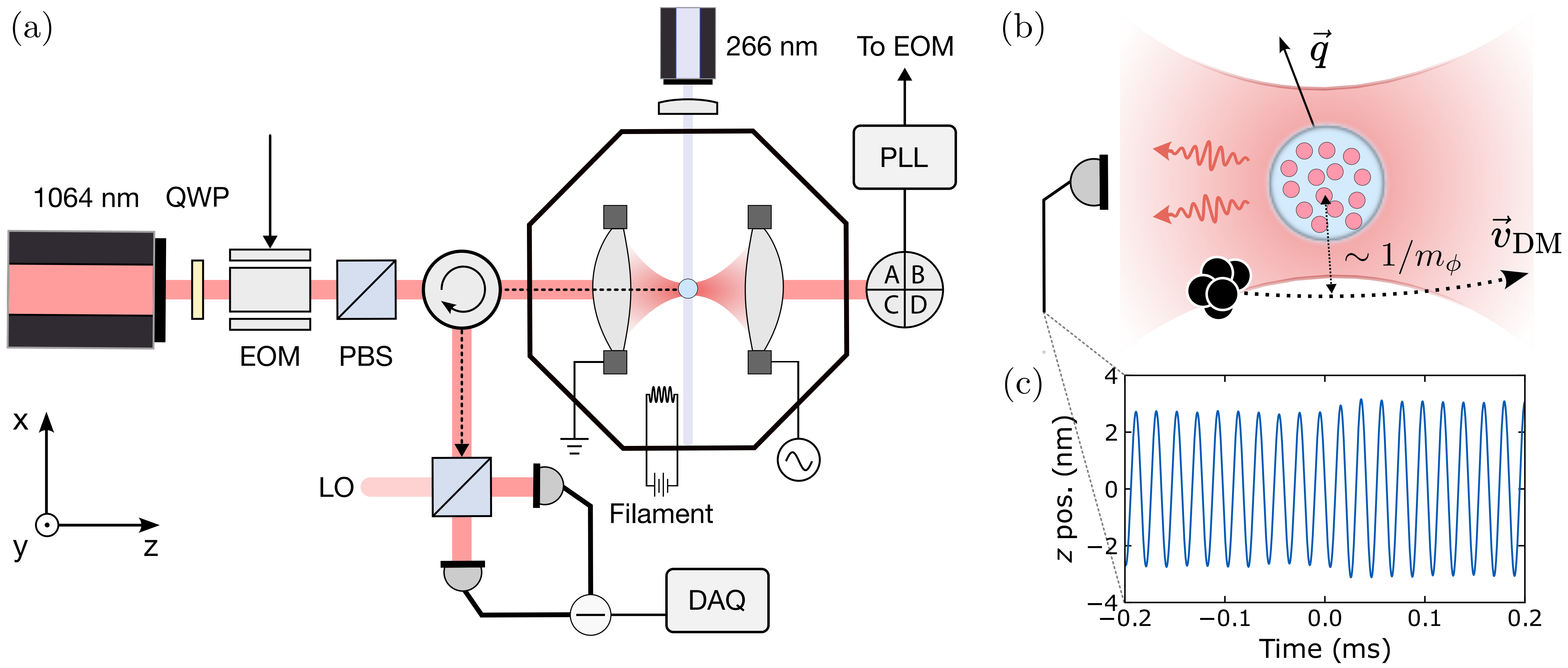
FPGA:
reprogrammable,
parallel processing,
logic blocks, etc



Luda, Marcelo. "The Red Pitaya Board." *Marceluda*, marceluda.github.io/rp_lock-in_pid/TheApp/RedPitaya_board/.

Cold-damping feedback

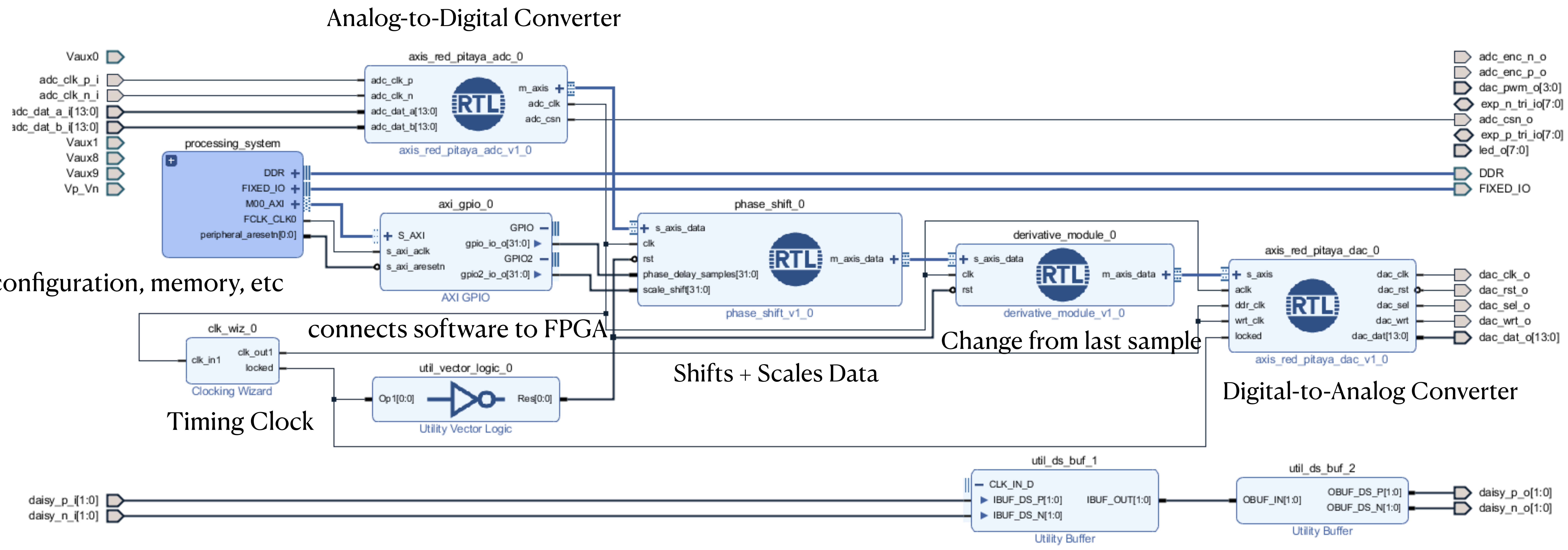
- Monitor COM motion using backscattered light from trapped particle
- RP takes signal in
- calculates feedback force
- outputs voltage —> pushes the sphere



Experimental Design

Past research reached ~100μK in vacuum (Tebbenjohanns et al. 2018)

Experimental Setup



FPGA Signal Processing Logic

```
module phase_shift (
    input wire clk,
    input wire rst,
    input wire [31:0] s_axis_data_tdata,
    input wire s_axis_data_tvalid,
    input wire [31:0] phase_delay_samples,  // number of samples to delay

    output reg [31:0] m_axis_data_tdata,
    output reg m_axis_data_tvalid
);

    // Shift register buffer
    reg [31:0] shift_reg [0:1023];
    wire [9:0] delay = phase_delay_samples[9:0];  // limit to 10-bit
index
    // Shift register logic using generate loop
    genvar i;
    generate
        for (i = 1023; i > 0; i = i - 1) begin : shift_loop
            always @(posedge clk) begin
                if (!rst && s_axis_data_tvalid)
                    shift_reg[i] <= shift_reg[i - 1];
            end
        end
    endgenerate

    // Insert new data and output delayed sample
    always @(posedge clk) begin
        if (rst) begin
            m_axis_data_tdata  <= 32'd0;
            m_axis_data_tvalid <= 1'b0;
        end else if (s_axis_data_tvalid) begin
            shift_reg[0]       <= s_axis_data_tdata;
            m_axis_data_tdata  <= shift_reg[delay];
            m_axis_data_tvalid <= 1'b1;
        end
    end

endmodule
```

```
`timescale 1ns / 1ps
module derivative_module (
    input wire clk,
    input wire rst,
    input wire [31:0] s_axis_data_tdata,
    input wire s_axis_data_tvalid,
    output reg [31:0] m_axis_data_tdata,
    output reg m_axis_data_tvalid
);

    reg [31:0] prev_sample;

    always @(posedge clk) begin
        if (rst) begin
            prev_sample <= 32'd0;
            m_axis_data_tdata <= 32'd0;
            m_axis_data_tvalid <= 1'b0;
        end else if (s_axis_data_tvalid) begin
            // Compute the derivative (difference between current and previous samples)
            m_axis_data_tdata <= s_axis_data_tdata - prev_sample;
            m_axis_data_tvalid <= 1'b1;
            prev_sample <= s_axis_data_tdata;
        end else begin
            m_axis_data_tvalid <= 1'b0;
        end
    end

endmodule
```

FPGA —> Linux

- Python program using /dev/mem
- User sets scaling + shift
- AXI GPIO allows software to control hardware signals
    - Linux —> TCP Server on Red Pitaya

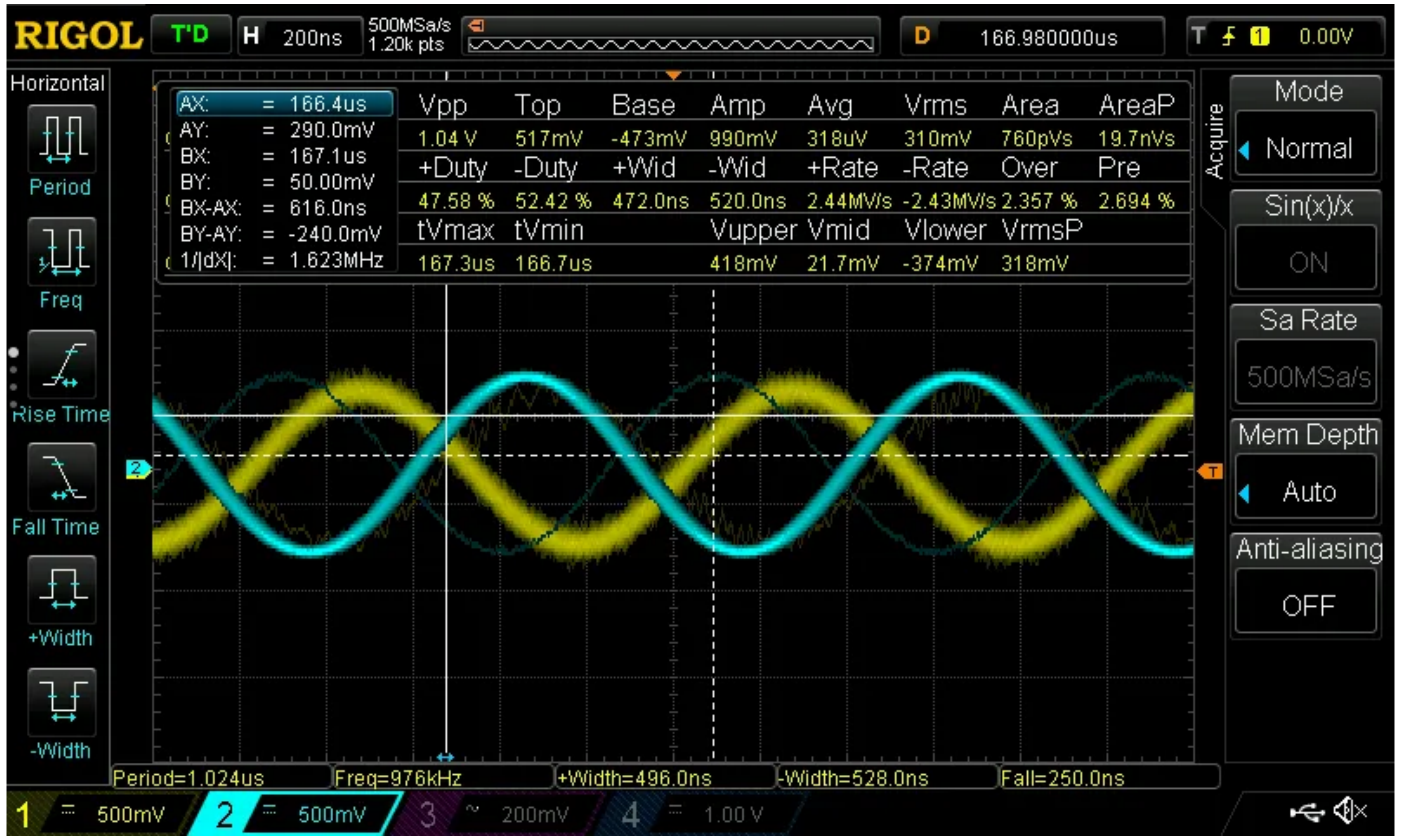
```
root@rp-f0cb1b:~# cat shift_derivative4_7.22.2025.bit > /dev/xdevcfg
root@rp-f0cb1b:~# ./server7
Listening for phase/scale on port 1001...
```

Terminal

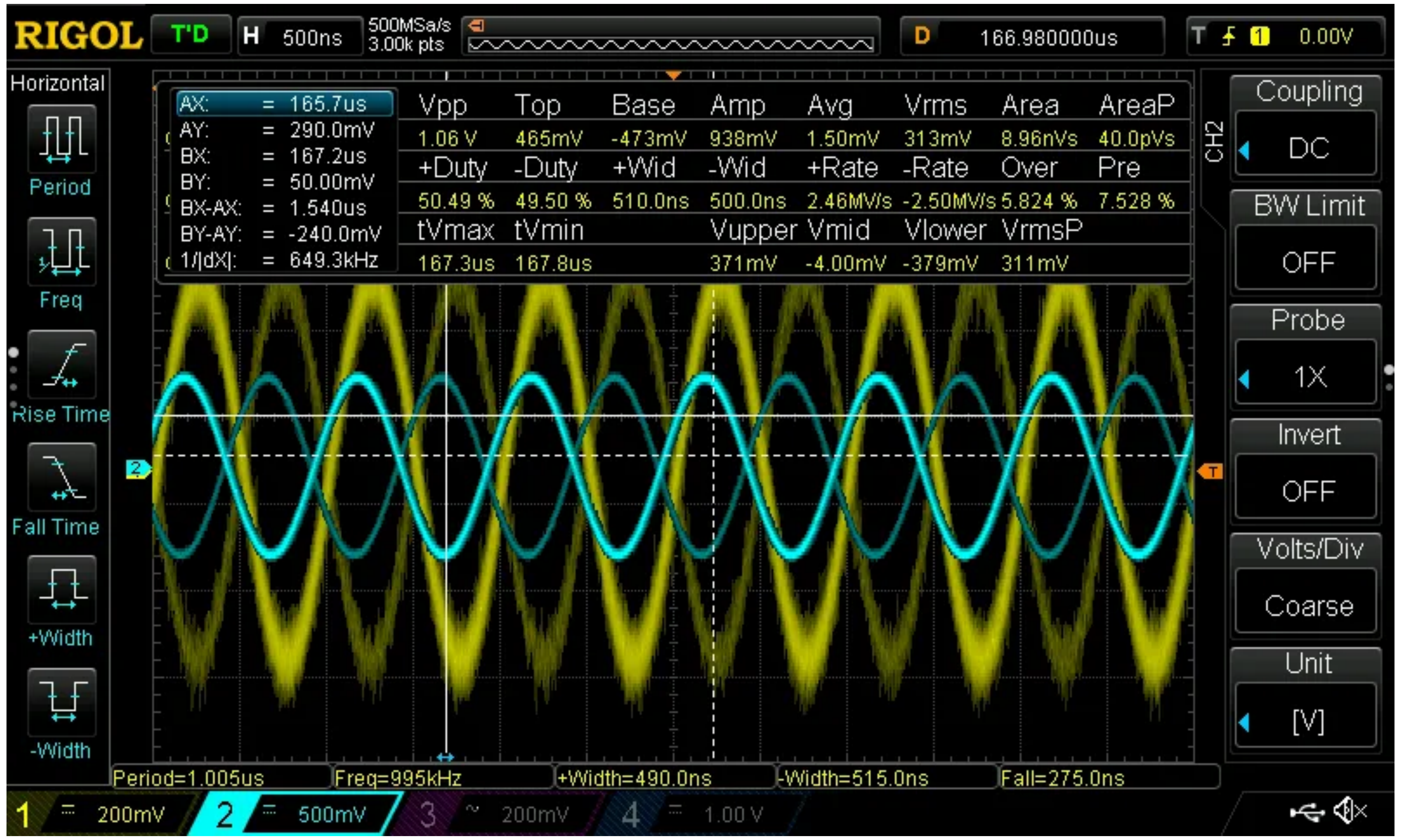
```
Connecting to 169.254.115.141:1001...
Connected to Red Pitaya.
Sent phase: 5, scale: 5
Connecting to 169.254.115.141:1001...
Connected to Red Pitaya.
Sent phase: 5, scale: 4
Connecting to 169.254.115.141:1001...
Connected to Red Pitaya.
Sent phase: 90, scale: 4
```

Python GUI

Oscilloscope



Low-scale



High-scale (noise)

Future Steps

- Full-feedback Loop
  - RP as a proportional-integral-derivative (PID controller)
- Continuous-time derivative approx. (reduce noise)
- Kalman Filter
  - combines measurement record w/ system dynamics

References

C. Gonzalez-Ballestero *et al*. ,Levitodynamics: Levitation and control of microscopic objects in vacuum.*Science* **374**,eabg3027(2021).DOI:10.1126/science.abg3027

Exuperian. (2022). *GitHub - exuperian/RedPitayaTutorials*. GitHub. https://github.com/exuperian/RedPitayaTutorials/tree/main

L. Magrini, P. Rosenzweig, C. Bach, A. Deutschmann-Olek, S. G. Hofer, S. Hong, N. Kiesel, A. Kugi, and M. Aspelmeyer, “Optimal quantum control of mechanical motion at room temperature: ground-state cooling,” (2020), arXiv:2012.15188 [quant-ph].

Potoˇcnik, A. Redpitaya Guide. http://antonpotocnik. com [online] (2016)
Luda, Marcelo. “The Red Pitaya Board.” *Marceluda*, marceluda.github.io/rp_lock-in_pid/TheApp/RedPitaya_board/.