Foucault Pendulum Electronics Kit.

D02-Options

www.foucaultpendulum.nl

Document version	2025-11_02
Relates to PCB version	all
Related Documents	all

In brief:
The kit is prepared for several options regarding the drive timing and amplidude control of the pendulum.
In all cases changes must be made in the firmware and probably the PC-program.

Options for the Center Pass detection.

The detection of the bob passing the Center (Rest posiition of the bob) is essential for all timing related functions.
Center Pass detection can be done in 4 ways:

1/ With a dedicated Center Pass detection coil. The magnet in the bob will induce a voltage impulse during the fly-over. The voltage is amplified and A/D converted by the Arduino. Detection of the pulse serves as the starting point for all timings.

2/ The Drive coil itself can be used for the Center Pass detection. This requires some extra circuits to prevent the large driving voltage to overdrive the pre-amplifier.
The advantage is that there cannot be a misalignment between the Drive Coil and the Center Detection coil.

3/ With the Capacitive method. This requires a central electrode to be present below the bob. The wire, and so the bob, carry the 465 kHz signal which will be picked up by that center electrode. The signal is then amplified and rectified on the Receiver PCB, and routed to the Arduino.

4/ The hardware in the kit supports detecting the wire touching a Charron Ring. This is a method often used in pendulums with a simple driving system, based on a delay circuit such as a 555. The firmware and PC software have to be modifyed to use this method.
I have never used this method and I can only give some suggestons for modifying the software for this.
Note: When the wire contacts the Charron Ring the 465 kHz signal on the wire will change, or be shorted, which causes the PMS to go wild. At this moment I cannot oversee the consequences of this.
The hardware and firmware support the simultaneous use of methods (1 or 2) and 3 for the CenterPass detection and Drive timing. So it is possible to compare the performance of magnetic detection with capacitive detection.

How to realize the options for CenterPass detection:

1/ Center Detection Coil.
The signal from the Center detect coil enters the BobCtl board on pins 5 and 7 of the COILS connector.
Connect pins 13-14 on the selection header near Q6 and Q7.
Place 2 jumpers on either 9-10, 11-12 or on 9-11, 10-12 of the POLARITY block, to select the proper signal polarity. The signal on Arduino A5 should go in the + direction when the

bob approaches the center.
The gain must be adjusted with R36. Adjust such that the peak-to-peak value on A4 is ca 4 Volt when the bob flies over the center of the detection coil. (with an ellipse the signal will be less.)
If you use only this method the components OC1, R40,42,43,50,51,C27, Q6, Q7 can be left away. R52 and R53 must stay to keep IC1C at a defined voltage level.

2/ From the Drive Coil. Note that this option is not fully tested.
Connect pins 15-16 on the selection header near Q6 and Q7.
The signals PL4_C_SHORT and PL0_C_GATE are used to prevent strong overdrive of the amplifier stages IC1C and IC2A by the Drive Pulse.
The gain should be adjusted with R36 as in option 1/. If the gain range is to small the gain can be increased by connecting a resistor of a few 100 Ohms parallel to R26.

3/ Capacitive method.
For this we need a circular electrode in the center below the bob. This electrode must be connected to the CNTR input of the Pre-Amplifier board. The board must be connected to the FLOOR connector on the Receiver board. The gain is to be adjusted on the Receiver board with the C trimpot. The signal on A4 of the arduino now has the shape of a positive impulse, the Center Pass is detected at the moment the signal changes from rising to falling.
For this option it is also required that the whole bottom plane, exept the center detect electrode, is covered with an electrically conductive and grounded layer, e.g. copper foil.
Also the bob should have a flat, round surface facing the Center Detect electrode, with the same diameter.
Note: This method has been basically tested in my sub-meter pendulum. Basically means that it works to detect CenterPasses and to determine the Half-Height Pulse Width.
No further pendulum performance has been investigated yet.

4/ Contact between Wire and Charron ring.
If the Charron ring is grounded the contact with the wire will short it to GND. This is detected via R45 and R41 on pin 39 of the Arduino, that is PORTG bit 2.
I have never used this method, and currently the software has no support for it, but I do the following suggestions:
- Sample the PG2 input at the 20 kHz rate and detect when the wire makes and looses contact (goes back to the center) Detect that situation on both halfswings. and calculate the halfway between these positions. That will be the moment of your Center Pass.
- Stop the position measurements while the wire is touching. It will result in rubbish.
- Analyse the times between the wire making and loosing contact on both sides. Several conclusions might be drawn, like: Is your Charron Ring well centered, and how much overdrive does the pendulum have. If the wire is touching for to short a time the braking effect for the minor axis might be to little and the ellipse may grow.

The touch time is also an indication for the amplitude of the pendulum.

5/ Both Capacitive and Magnetic
There is the possibility to use both methods at the same time, it could be interresting to compare the performance.
Prepare the system for magnetic detection as per 1/ or 2/
Prepare the system for Capacitive detection as per 3/.

Options for the Amplitude Control
Generally some sort of Amplitude Control is required. Many (museum) pendulums do not have an active system for this, but relay on a constant amount of drive energy to be pumped into the system, and an amount of damping (energy loss) which increases with the amplitude of the pendulum. Normally this balances at a certain amplitude.
However, the amount of energy pumped by the driving coil heavily depends on the distance between the bob and the driving coil, so on the length of the wire, which will be temperature dependent, and on the exact moment the drive pulse is given and where the bob's magnet is above the coil at that moment.
The "Schumacher" drive timing to suppress the intrinsic precession of the ellipse requires

the pendulum amplitude to be constant.
There are several methods to control the amplitude of the pendulum.
1/ Use of a Rim Coil. Below the bob we have a coil wit a radius of ca. 2/3 of the desired amplitude. When the bob's magnet flies over it a voltage is induced which is amplified and A/D converted by the Arduino. The time it takes for the bob to go from the center to the rimcoil is a direct measure of the amplitude.
2/ If one uses the capacitive method to detect the Center Pass one can measure the half-height width of the signal from the center electrode. This is a rather direct measure of the bob's velocity at the center pass, BUT only if there is no ellipse. Generally we have an amount of ellipse and so a correction is needed. If the PMS is operating well calibrated this should be possible.
Both methods are currently implemented.

Using the Resonance Method to drive the pendulum:
This method is based on the circumstance that the period time of a pendulum is amplitude dependent. The larger the amplitude, the longer the period time. The effect is however very small (and mostly neglected).
The idea is to drive the pendulum with the frequency of the major axis, and not excite the slightly higher frequency of the minor axis. The difference is small, but it might work. For this we need a very stable frequency source which can be adjusted in very small steps.
As I have planned to do such an experiment some provisions are made in the hardware.
The frequency of 465 kHz is generated with an AD9833 Direct Digital Synthesizer (DDS).
This device can generate that frequency in steps of 0.1 Hz. To get to a frequency of say 0.5 Hz for a 4-meter pendulum we need to divide this by nearly 10^6, giving a 1:10 million resolution for the frequency steps.
The challenge is to keep this under control, where the period time might change due to a.o. temperature differences.
We also may want to use the "Schumacher" Drive timing, which sets a second burden on the control system.
But in the end it might be possible to detect the small changes in gravity due to orbiting celestial bodies like the moon.
The DDS module has the following relation between the FrequencyWord, the Xtal Frequency and the Output frequency:
Xtal Frequency is supposed to be 25,000,000 Hz (25 MHz).
FrequencyOut = FXtal * FrequencyWord / 2^{28} [Hz].
FrequencyWord = FrequencyOut * 2^{28} / FXtal .
So for 465 kHz we need .
FrequencyWord = 465000 * $2\hat{}^{28}$ / 25000000 = 4992899.
The chosen default setting gives a 0.5 Hz frequency suitable for a 4 meter pendulum.