IMPLEMENTATION OF QUADRUPLE-TIMING PULSE INTERPOLATION APPLIED TO COMPACT PISTON PROVERS

Carlos Eduardo Jeronymo[1], Valter Yoshihiko Aibe[1],
1- Instituto Nacional de Metrologia, Normalização e Qualidade Industrial, 25.250-020 RJ Brazil
Tel:55-21-2679-9149, Fax:55-21-26791163, E-mail: cajerony@yahoo.com

Abstract: This work presents an electronic circuit for quadruple-timing pulse interpolation applied to compact piston provers. Compact provers are usually employed to prove meters with pulsed outputs. API and ISO standards [1, 2] recommend a minimum of 10000 pulses per run to obtain a resolution better than ±0.01%. Since the volume of fluid displaced by a compact prover is relatively small, the number of pulses produced during a proving run is often considerably less than 10000 pulses. Pulse interpolation techniques are commonly used to increase resolution and to diminish uncertainty during a proving run by estimating the fractional part of meter pulses within the time interval of the calibration. In this way, pulse interpolation techniques are essential to obtain accurate flow measurements and to allow the calibration of meters with compact provers.

Keywords: Compact piston prover, Pulse interpolation, Flow meter calibration

1. Introduction

Compact piston prover systems (also referred as small volume provers) are used in the calibration of flowmeters. They have long been accepted as primary flow calibrators for both gas and liquid flowmeters [1-3]. The use of compact provers and interpolation techniques were formally recognized by API in 1988, with the publication of API MPMS, chapter 4. Nowadays, compact provers in conjunction with pulse interpolation techniques enjoy large acceptance in the industry. Basically a compact prover compares the volume of fluid displaced by a piston during a specific time interval with the volumetric flow measured by a meter [4]. Compact provers present many advantages compared to conventional provers, such as portability, low cost, use of small amounts of fluid, reduced time for calibrations and capacity of proving lower flow rates. Due to its low volume and limited number of pulses, it's important the pulse interpolation method is correctly specified and implemented. It's also important to observe that accuracy and applicability of pulse interpolation methods may deteriorate due to several factors such as circuit configuration and pulse stability. The API and ISO standards recommend a minimum of 10000 pulses collected per run [1, 2]. Since the volume of fluid displaced by a compact prover is relatively small, most of the time it may not reach 10000 pulses per run. In order to overcome this problem, pulse interpolation techniques are commonly used to estimate the fractional part of meter pulses within the time interval of the calibration [1, 5]. Pulse interpolation techniques assume that flow is stable during a proving pass. ISO 7278-3 establishes that fluctuations in the flowrate during a pass must be limited to less than ± 2% of the mean flowrate. Three different interpolation methods are described in ISO 7278-3: double-timing, quadruple-timing and phase-locked-loop techniques. Each method has its own particularities, hardware complexity and can be affected differently by flowrate fluctuations or pulse irregularities during a proving pass.

2. Compact Provers Calibration Systems

There are different types of compact prover arrangements in the market [4, 5]. In this work we have employed an active piston prover driven by compressed air. Figure 1 shows schematically

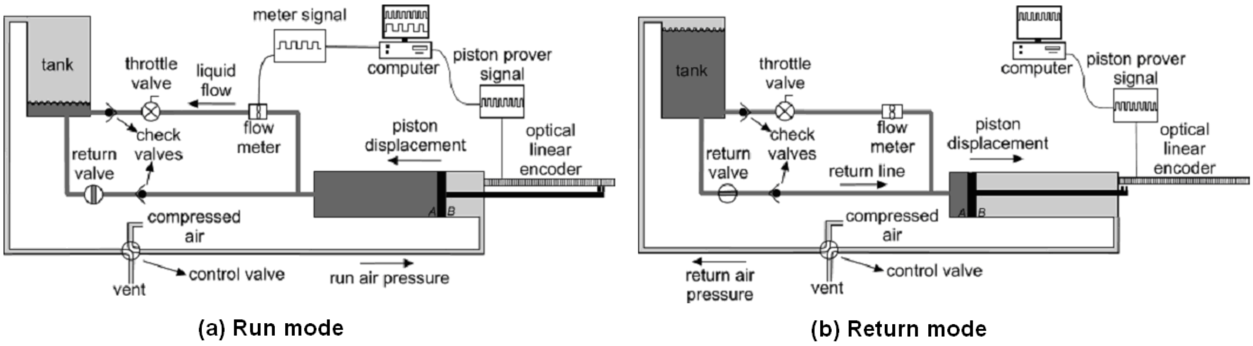

Fig. 1 Schematic diagram of compact piston prover

the working principle of this type of piston prover. Basically it has two operational modes: "run mode" and "return mode". During the "run mode" (Fig. 1.a), pressurized air is directed to chamber *B* and the resultant force pushes the piston to the left at constant speed. An incremental linear encoder connected to the piston rod detects the piston position and sends a pulse train digitizing the piston position to a computer. The linear encoder has a resolution of 50 pulses per mm. At the same time, fluid is displaced from chamber *A* and passes through the meter under test. The meter generates a pulse train proportional to the volume of fluid passing through it. Pulse train signals are processed in the computer and the displaced volume and correspondent flow rate is calculated. The volumetric relation between linear encoder pulses n_{enc} and piston traveled distance is given by $\Delta V_p = K_v n_{enc}$, where K_v is a constant (in $cm^3/pulses$) that can be determined using the water draw method, measuring the dimensions of the piston or using a master meter [4]. During the "return mode" the position of the control valve is diverted and pressurized air is sent to the top of the tank (Fig. 1.b). The piston moves to the right and chamber *A* is filled with liquid. The system is then ready for a new calibration cycle.

3. System Evaluation

There are various techniques that can be used to implement pulse interpolation and estimate the fractional part of meter pulses within a time interval. The complexity of the electronic hardware depends on the method being implemented. The ISO 7278-3 [1] acknowledges three interpolation techniques, i.e., double-timing, quadruple-timing and phase-locked-loop. We explain below the quadruple-timing interpolation.

3.1 Quadruple-timing

Figure 2 shows a schematic diagram of the quadruple-timing method. In this figure, t is the time interval between the *START* and *STOP* detection signals containing a fractioned number of pulses $n`$. Times $t_1$ and $t_3$ are the time intervals of a fractioned pulse as indicated on Fig. 2. Times $t_2$ and $t_4$ are the time intervals of a whole pulse as shown on Fig. 2. The number of interpolated pulses $n`$ within t, is calculated using the following equation:

$$n` = n + t_1/t_2 - t_3/t_4 \quad (1)$$

4. Quadruple-Timing Implementation

We discuss below three different approaches that can be used to implement the quadruple-timing pulse interpolation technique. The third approach explains the method proposed in this article.

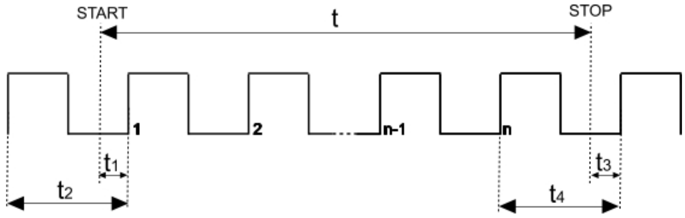

Fig. 2 Detection of meter pulses for the implementation of quadruple-timing technique

1) The most generic approach is to continuously monitor the meter pulse train waiting for a *START* detection signal. For that, at the beginning of each meter pulse we initiate the measurement of t_2. If a *Start* position signal is detected, we initiate the measurement of t_1. At the end of the meter pulse we stop both counters and store in the computer memory the values of t_1, t_2. If till the end of the meter pulse a *Start Position Signal* is not detected, we reset counter t_2 and initiate the same procedure again. A similar procedure is used to acquire t_3 and t_4. The electronic hardware for this type of implementation is relatively complex since we need to continuously monitor the meter pulse train and continuously reset the counter measuring t_2 when a *START* position signal is not detected.

2) Another method to implement the quadruple interpolation is shown in references 6 and 7. In this approach, the initial and final meter pulse period is collected after t_1, t_3 respectively. That's illustrated in Fig. 3 by t'_2 and t'_4. The fractioned number of pulses ($n`$) is calculated as $n` = n + t_1/ t'_2 - t_3/ t'_4$. The electronic hardware is simpler than in the previous method, since t_1 & t'_2 and t_3 & t'_4 are collected after the detection of the *Start/Stop* position signals and consequently, we don't need to monitor the meter pulse train continuously. Tough this approach requires a simpler electronic hardware it will present an interpolation error when the meter pulse train is not regular since t'_2, t'_4 can be different from the initial and final time period t_2, t_4 (Fig. 2).

3) The third approach is the method proposed in this work. It's an alternative method aimed to compact provers having a linear optical encoder installed on it. The linear encoder allows the measurement of the piston displacement. Figures 4 and 5 show a schematic diagram of the working principle of the proposed method. These figures show pulse trains originated from the linear encoder and from the meter. We have two cases:

(a) Encoder resolution is higher than the meter resolution;

(b) Meter resolution is higher than the encoder resolution.

Initially let's explain case (a):

Case (a) The acquisition process is illustrated in Figure 4. Consider D_{o1} as a digital input signal with state levels High or Low. Initially D_{o1} is at low state. The process starts when D_{o1} is set high (for example, by software or at a defined piston position by means of an optical switch). After

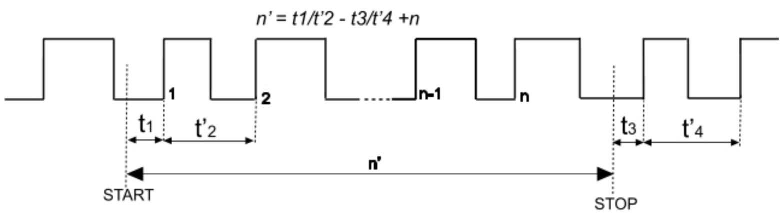

Fig. 3 Quadruple-timing implementation (simplified method)

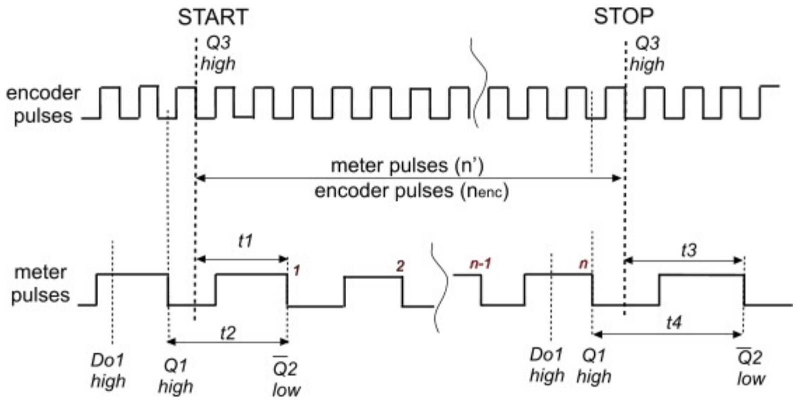

Fig. 4 Encoder resolution is larger than meter resolution

D_{o1} is set high, Q_1 turns high at the next meter pulse failing edge starting the measurement of t_2. Signal Q_3 turns high at the next encoder pulse failing edge. That's the *START* detection signal that initiates counters measuring the meter pulses n, encoder pulses n_{enc}, time t_1 and total time t. Observe that the *START* position signal is synchronized with a low edge encoder pulse. At the end of the next meter failing edge, $\overline{Q}_2$ turns low stopping the counters measuring t_1 and t_2. The values of t_1 and t_2 are then stored in the computer memory. D_{o1} is set low and we keep counting pulses related to n, n_{enc} and total time t. A similar process is used to stop the acquisition process. For that, D_{o1} is set high again. At the next meter pulse failing edge Q_1 turns high. After Q_1 turns high, Q_3 turns high at the next encoder failing edge. Signal Q_3 high is our *STOP* acquisition signal and deactivates the acquisition of meter pulses n, encoder pulses n_{enc} and total time t. At the next meter pulse failing edge, $\overline{Q}_2$ turns low stopping the measurement of t_3 and t_4. In this way we have gathered n, n_{enc}, t, t_1, t_2, t_3, t_4. The number of fractioned pulses is calculated as: $n` = n + t_1/t_2 - t_3/t_4$ and the volume displaced is given as: $\Delta V_{\mathrm{p}} = K_{\mathrm{v}}\, n_{enc}$ as explained in section 2.

Case (b) The acquisition process is similar to case (a) and illustrated in Fig. 5. The main difference is that since the meter resolution is higher than the encoder resolution, we may have n_{pi} meter pulses (i =1, 2) between Q_1 high and $\overline{Q}_2$ low ($n_{pi} \geq 1$). This way, we calculate t_2 as $t_2 = t_{w1}/ n_{p1}$ and t_4 as $t_4 = t_{w2}/ n_{p2}$ where t_{wi} and n_{pi} (i =1, 2) are respectively the number of clock pulses and number of meter pulses gathered between Q_1 high and $\overline{Q}_2$ low. It's worthwhile to observe that in the particular case of the piston prover used in this work, the piston displacement is equivalent to approximately 27000 encoder pulses. Therefore we can easily obtain more than 10000 pulses in one pass and the use of pulse interpolation techniques is not essential.

5. Electronic Circuit Implementation

A schematic diagram of the electronic circuit necessary to implement the logic described above is shown in figure 6. Basically the electronic circuit is composed of three JK flip-flops and nine 16-bit counters numbered from c_1 to c_6. In figure 6 we show only 6 counters because counters c_1, c_3, c_6 are the result of two 16-bit counters chained to form a 32-bit counter. When the state of J is high level (HL) the flip-flop is activated and when K is low level (LL) the flip-flop is deactivated. Details about JK flip-flops and their function table can be found in reference [8]. Signal $\overline{Q}_i$ is the complement of Q_i. Counters are represented as having a *Gate* (gate-n), *Counter Input* (ctr-n)

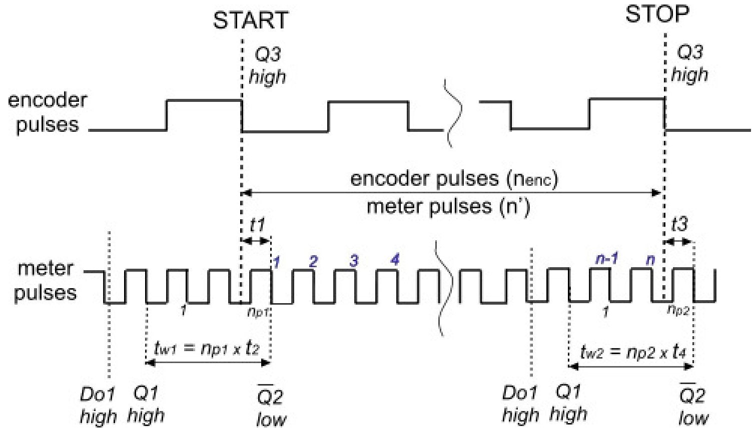

Fig. 5 Meter resolution is larger than encoder resolution

and *Output Control* (out-n). Initially all gates and digital outputs D_{o1}, D_{o2} are set to low level (LL), D_{o3} is set to high level (HL), D_{o4} is set to LL (decoupling $\bar{Q}_3$ from the circuit) and the JK flip-flops (f_1 to f_3) are reset setting Q_i (i = 1 to 3) to low level (LL) and $\bar{Q}_i$ to HL. Counter c_1 gathers t_2 or t_4, counter c_2 gathers n_{p1} or n_{p2}, counter c_3 gathers t_1 or t_3, c_4 stores encoder pulses, c_5 stores meter pulses and counter c_6 stores total time t. Acquisition starts by turning D_{o1} high that activates flip-flop f_1. At the next meter failing edge Q_1 turns high, activating flip-flop f_3 and t_2. At the same time $\bar{Q}_1$ turns LL sending K to LL and deactivating f_1. Since f_3 is now activated, at the next encoder failing edge Q_3 turns high activating counter c_3 (time t_1). Since D_{o3} was set initially high, the output of *AND3* becomes HL activating counters c_4 (encoder pulses n_{enc}), c_5 (meter pulses n) and c_6 (total time t). Q_3 is now HL and therefore, flip-flop f_2 is also activated. At the next meter failing edge, $\bar{Q}_2$ will turn low deactivating counters c_1, c_2, c_3. Counters c_1, c_2, c_3 contain now t_2, n_{p1} and t_1 respectively. These values are stored in the computer memory. After that, we set D_{o2} to HL latching gates c_4, c_5, c_6 in the activate state. Flip-flops f_1 to f_3 are reset, and counters c_1, c_2, c_3 are set to zero. D_{o1} is set back to LL, D_{o4} is set high and D_{o3} is set LL decoupling Q_3 from gates c_5, c_6, c_7 and connecting $\bar{Q}_3$ to these counter gates. Since $\bar{Q}_3$ is high, we can set D_{o2} back to LL keeping gates c_4, c_5, c_6 activated. The Stop acquisition mechanism is quite similar. The Stop process starts when D_{o1} is set back to HL. The acquisition process for t_3, t_4, n_{p2} is similar to the acquisition process for t_1, t_2, n_{p1}. The main difference between Start and

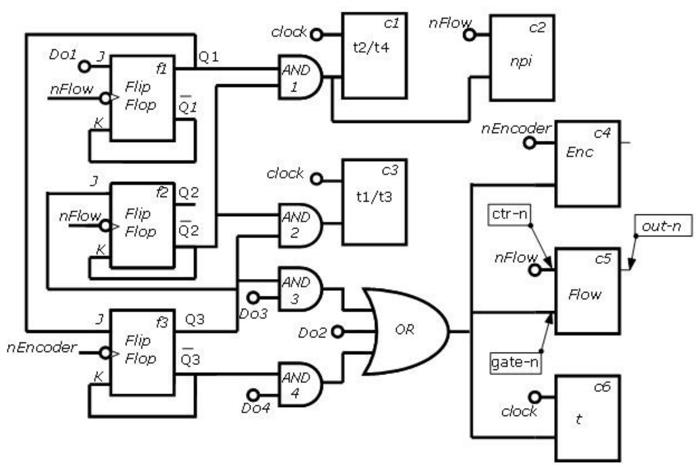

Fig. 6 Block diagram of pulse interpolation unit

Stop is that gates c_4, c_5, c_6 are deactivate (observe that D_{o2} and *AND3* are at LL state) when $\bar{Q}_3$ turns low. Counters c_4, c_5, c_6 will contain the whole number of meter pulses (*n*), whole number of encoder pulses (n_{enc}) and total time (*t*) respectively. When $\bar{Q}_2$ turns low, counters c_1, c_2 and c_3 will contain t_4, t_3 and n_{p2} respectively. The number of fractioned pulses is calculated using equation (1). Please note that the piston traveled distance between *START* and *STOP* contains a whole number of linear encoder pulses since the counter gate is activated / deactivated at the exact position of a low edge encoder pulse. This approach decreases the uncertainty in the calculation of the volumetric flow since the displaced volume is exactly a whole number of encoder pulses. In this way, flow rate and the *K-Factor* (*K-Factor = meter signal pulses /corrected prover volume*) can also be determined.

6. Test Circuit

ISO 7278-3, section 7.2, establishes the requirements for testing a pulse interpolation circuitry. Figure 7 shows a schematic diagram of the testing. The testing circuit is composed of a pulse generator and a frequency divider that allows the generation of two sets of pulses with different frequencies. The pulse train with higher frequency *F* drives a reference counter *A* and the other pulse train with lower frequency *F/R* drives the pulse interpolation unit under test. Counter *A* and the pulse interpolation unit are both controlled by simultaneous START/STOP signals and can be set to zero (in our case START/STOP signals are both defined by Q_3 high (figures 4, 5). As described in ISO 7278-3 [1], the readings of counter *A* have to agree with $n` \times R$ within 0.01% for a time interval large enough to accumulate at least 10000 pulses. The input frequency can be constant ($dF/dt = 0$) or a ramp of variable frequencies ($dF/dt \neq 0$). The encoder pulse train is simulated through a fixed frequency of 10 kHz that is approximately equivalent to a maximum flow rate of 180 l/min. We have used a Pentium D, 2.80 GHz and an off-the-shelf counter/timer board having a total of ten 16-bit counters. The frequency of the internal clock of this board can be set from 0.1 kHz to 5 MHz. The data acquisition software was written using VB .Net.

Table 1 shows test results for a constant pulse frequency ($dF/dt = 0$) for a clock of 1 MHz. The interpolation error ε (%) is calculate using equation: $\varepsilon(\%) = 100 \times (n' \times R - n_A)/n_A$. This table shows the smallest and the largest interpolation error obtained from a series of tests with *F/R* frequencies of 1 kHz (R=1000) and 0.1 kHz (R=10000). We can see that error ε is always less than 0.01%.

Table 2 shows test results for $dF/dt \neq 0$. In this Table, F_{Start} is the initial frequency and F_{Stop} the

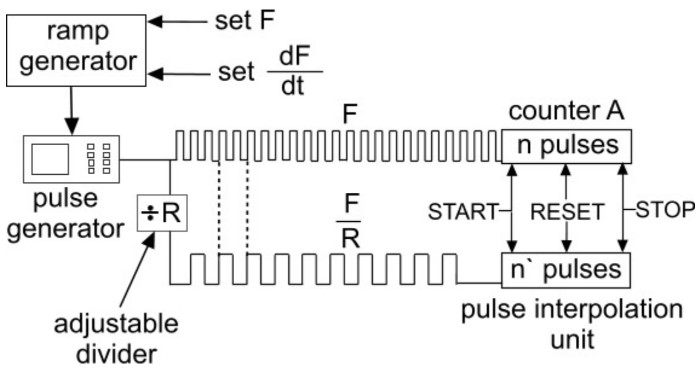

Fig. 7 Pulse interpolation test circuitry

Table 1. Results (F = 1 MHz, Clock = 1 MHZ); ε(%) = 100 x (n'R -n_A)/n_A

R x 10^2	n_A (pulses)	n (pulses)	T1 (pulses)	T3 (pulses)	T2 (pulses)	T4 (pulses)	n' (pulses)	ε(%) x 10^{-5}
100	100029966	10003	9908	9940	10000	10000	10002.9968	0.2
100	100019943	10002	9907	9969	10000	10000	10001.9938	-0.5
10	10002932	10003	913	981	1000	1000	10002.9320	0.0
10	10033968	10034	907	936	1000	1000	10033.9710	3.0

Table 2. Ramp frequency, ε(%) = 100 x (n'R -n_A)/n_A)

F_{Start}->F_{Stop} (MHz)	Clock (Mhz)	R x 10^2	n_A (pulses)	n' (pulses)	ε(%) x 10^{-5}
1.0->1.2	1	100	100009993	10000.9989	-0.4
1.0->0.8	1	100	100020017	10001.9985	-3.2
1.0->1.2	1	10	10026292	10026.2958	3.8
1.0->0.8	1	10	10013912	10013.9144	2.4

final frequency. The input frequency F was changed manually from F_{Start} to F_{Stop} at approximately 0.01 MHz per second. The test results show that the interpolation error ε is not considerably affected by variations in the input frequency.

7. Experimental Results

In this section we verify the performance of the proposed interpolation circuitry under actual operating conditions. A turbine type meter producing six pulses per revolution was mounted to the piston prover to evaluate the interpolation circuitry and control program. The fluid is distilled water. The turbine resolution is higher than the encoder resolution (see figure 8.a). Table 3 shows test results for three passes. Since the meter resolution is larger than the encoder resolution n_{p1}, n_{p2} can be larger than 1. Considering only the pulse interpolation without adding the system uncertainties and for a same displacement volume (equivalent to 293 encoder pulses) we have n' = 1327.4968 ± 0.0774 and standard deviation ΔS (%) = 0.0058%.

Table 4 shows test results for an oval gear type meter producing 4 pulses per revolution. As we can see from Figure 8.b, this meter produces an irregular pulse train. The fluid is distilled water. The meter resolution is lower than the encoder resolution. For a same displacement volume equivalent to 10006 encoder pulses, we have $n' = 1220.4648 \pm 4.8174$ with $\Delta S\,(\%) = 0.4\%$.

Table 3. Turbine results (clock = 5 MHz)

n_{meter} (pulses)	$n_{encoder}$ (pulses)	Temp (C)	T_1 (pulses)	T_3 (pulses)	T_2 (pulses)	T_4 (pulses)	n_{p1}	n_{p2}	n' (pulses)
1327	293	20.79	24978	5048	33938	33671	2	2	1327.5860
1328	293	20.79	4394	23342	33803	34257	2	2	1327.4486
1327	293	20.79	19823	4261	34086	33893	4	3	1327.4558

Table 4. Oval Gear meter (clock = 5 MHz)

n_{meter} (pulses)	$n_{encoder}$ (pulses)	Temp. (C)	T_1 (pulses)	T_3 (pulses)	T_2 (pulses)	T_4 (pulses)	n_{p1}	n_{p2}	n' (pulses)
1222	10003	21.26	533820	405928	581561	464256	1	1	1222.0435
1224	10006	21.26	556121	555012	585985	588170	1	1	1224.0054
1215	10006	21.26	552016	432230	612437	468529	1	1	1214.9788

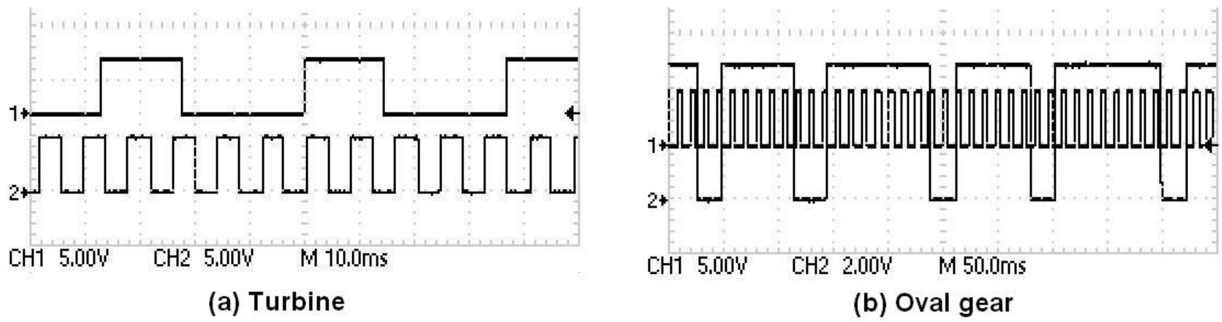

Fig. 8 Pulse train (CH1 = encoder, CH2 = meter)

The oval gear meter used in this study has also shown poor repeatability using the gravimetric method. Further study is needed to verify the influence of the pulses irregularities on the calibration accuracy and its influence on the piston displacement speed.

8. Conclusions

This work presents an alternative method to implement the quadruple-timing pulse interpolation technique aimed to compact provers having an optical encoder to measure piston displacement. The implementation of this method is straightforward and can be easily done using off-the-shelf electronic devices such as devices such as counters, timers and logical gates at a relatively low cost. The method is useful for implementing the quadruple-timing interpolation method without requiring customized electronic hardware.

References

[1] International Standards Organization, ISO 7278-3, 1998, Liquid hydrocarbons, Dynamic measurement, Proving Systems for Volumetric Meter, Part 3, Pulse interpolation techniques.

[2] API Manual of Petroleum Measurement Standards, 1999, Chapter 4, Proving Systems, Section 6, Pulse Interpolation, API MPMS 4.6.

[3] Yeh T. T., Aguilera J., Wright J. D., Hydrocarbon Liquid Flow Calibration Service, NIST, Special Publication, 250-1039.

[4] Yeh T. T., Espina P. I., Mattingly G.E. and Briggs N. R., An Uncertainty Analysis of a NIS Hydrocarbon Liquid Flow Calibration Facility, Proceedings of HT/FED'04, 2004 Heat Transfer/Fluids Engineering Summer Conference July 11-15, 2004, Charlotte, North Caroline, USA.

[5] Pipe provers: A User's Manual, Alan T.J. Hayward, Cambridge University Press, 1991, ISBN 0 521 33113 7

[6] Ward P. A, Theory and Application of Pulse Interpolation to Prover Systems, Brooks Instrument, Rosemount Group, International School of Hydrocarbon Measurement, 1993.

[7] Theory and Application of Pulse Interpolation to Prover Systems- FMC –Smith Meter Inc., Technical Paper 126. (http://info.smithmeter.com/literature/docs/tp0v001.pdf)

[8] Motorola Semiconductor Technical Data, Dual JK Flip Flop with Set and Reset, Motorola, Inc. 1995.