Cranking Out Camshaft Timing Data

by
Jeffrey J. Scott
Senior Project Manager, Measurement & Automation
Data Science Automation, Inc.
USA

and

Ryan W. Vallieu
Consultant, Mea surement & Automation
Data Science Automation, Inc.
USA

Category:
Automotive

Products Used:
LabVIEW 8.5
SCXI-1000 4-Slot Chassis, U.S. 120 VAC
SCXI-1374 Handle Kit for SCXI-1000
SCXI-1361 Rear Filler Panel
SCXI-1600, USB Data Acquisition and Control Module
SCXI-1540 8-channel LVDT Module
SCXI-1100 32-Channel Multiplexer Amplifier, Signal conditioning module for TC, volt, millivolt, and current input
SCXI-1303 32-channel isothermal terminal block - for SCXI-1100
Phoenix Contact PLC-OSC-24DC/24DC/2 Relay 24VDC, 20µs response time
SMB-110, SMB to BNC Male Coax Cable, 50 Ohm, 1m
Nikon RX2000-22-1A Rotary Encoder
Keyence GV-45H Digital CMOS Laser Sensor
Keyence GV-21P PNP CMOS Laser Sensor Amplifier
MDR-40-24 Mean Well 40W, 24V Single Output Industrial DIN Rail Power Supply
DR-30-5 B&B Electronics 30W, 5VPower Supply

The Challenge:
The customer needed a portable data collection system to record camshaft timing data obtained during production-line quality control verification testing of engine assemblies. The data in question consists of valve displacements measured with LVDTs correlated to degrees of crankshaft rotation over 720 degrees.

The Solution:
The solution Data Science Automation created utilizes USB technology to centralize the data collection in an SCXI chassis to allow a laptop computer to control the data collection system. Hardware timing based on a crankshaft encoder allows faster and more accurate data collection of camshaft timing information compared to previous methods. The resulting solution has reduced individual engine testing time by up to 30 minutes per engine.

Abstract:
A large automotive manufacturer had a problem. Every engine coming off the assembly needed to have its engine camshaft (Figure 1) timing verified. The problem was that the existing process for making this critical measurement involved time-consuming (and error-prone) manual measurements. What the manufacturer needed was a way to automate the engine

camshaft timing verification test. To satisfy this need they picked Data Science Automation. The new automated test system utilizes a SCXI-1600 USB-controlled SCXI chassis to provide that acquisition capability and an encoder connected to the crankshaft to synchronize the data with crank rotation.

Figure 1 - Camshaft

Crank Up the Speed!

Prior to the development of the automated system, the only way of acquiring the required data was to manually measure the displacement for each valve versus crankshaft angle using dial indicators. These individual readings (and there could be as many as 24 valves per engine) had to be recorded along with the readings from a separate crank angle sensor. As if this wasn't bad enough, the procedure had to be repeated multiple times for each engine with the crankshaft at different angles. Data Science Automation designed the replacement system to make this difficult job much simpler affair by completely automating the data collection portion of this test. The test platform consists of a SCXI-1000 chassis teamed with a SCXI-1600 USB controller. This hardware provided the acquisition capability while a portable laptop running the LabVIEW Run-Time Engine supplied the computational horsepower.

Populating the SCXI chassis is a SCXI-1540 8-Channel LVDT module for reading four LVDT sensors. Note that although the module is capable of handling a higher channel-count, the customer choose to keep the number of sensors at four due to cost constraint and engine valve bank layouts.

Next, a Nikon RX2000 encoder is mounted on the engine to track the crankshaft rotation angle. The encoder output clocks the LVDT data acquisition through the USB-1600 PFI7 line. The analog LVDT acquisition task is also configured with a digital start trigger which comes from a Keyence Digital CMOS Laser Sensor. The crankshaft on the engine under test has a keyway that points straight up when the engine is at the Top Dead Center (TDC) of position for piston 1. A custom mounting bracket positions the laser sensor such that it targets the forward edge of the key when the crank is in the proper position (Figure 2).

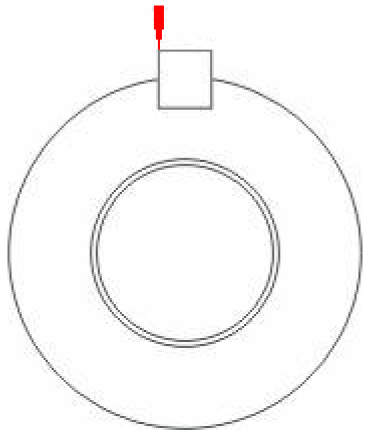

Figure 2 – Crankshaft Keyway and CMOS Laser Triggering Alignment –DAQ Task Digital Start Trigger

This additional sensor means that the technician doesn't have to try and match a TDC reading by hand before starting a test. The operator simply attaches the laser, crankshaft encoder, and LVDTs to the engine, sets the crankshaft to any position in front of the key and start the test. Well designed test fixtures guarantee that crank position will be comparable from test-to-test and engine-to-engine.

Main Program Architecture

The main program in this system (Figure 3) utilizes an event-driven architecture to ensure that the system responds to the operator control inputs in a timely manner. The event structure's timeout event contains a case structure that forms a state machine with responsibility for executing the actions requested by the operator.

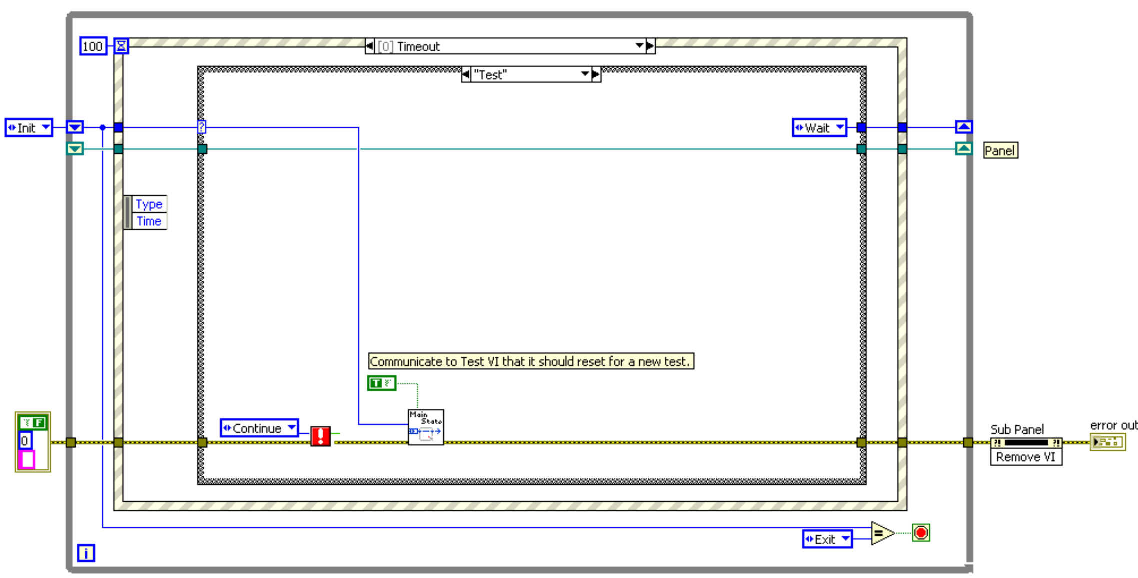

Figure 3 - User Interface Event Structure Timeout Case

The operator's main system control is through a cluster of buttons (Figure 4). To preclude the operator from commanding an illogical sequence of events, the event code changes the buttons' enabled properties depending on the system's current state.

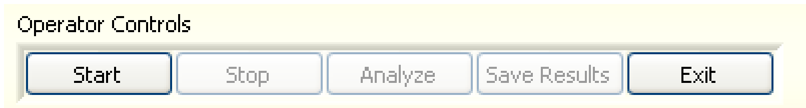

Figure 4 – Main Operator System Control Buttons

In addition to the main Event Handler top level VI; this system incorporates three other major modules:

- Test State Machine
- Data Analysis State Machine
- Data File Writer State Machine

The Test Controller State Machine is launched dynamically and runs in parallel with the main GUI VI Event Structure so as to not block the GUI's operation. The other two state machines run in the Event Structure cases dedicated to the buttons that launches them, i.e. Analyze and Save Results.

Graphical Interface

The main operator Graphical User Interface (GUI) consists of a tab control that has three pages. The first page (Figure 5) is for setup of the hardware utilized in the system, as well as the acquisition timing information which is based on the encoder selected for the crankshaft position tracking. Allowing the operator to utilize differing encoders and exposing the number of pulses per revolution setting makes this system reconfigurable for finer position measurements if in the future the customer determines that higher resolution encoders are needed. Settings are also available for the LVDTs that take the valve displacement measurements. Operator control buttons allow the test technician to control the operation of the software testing stages.

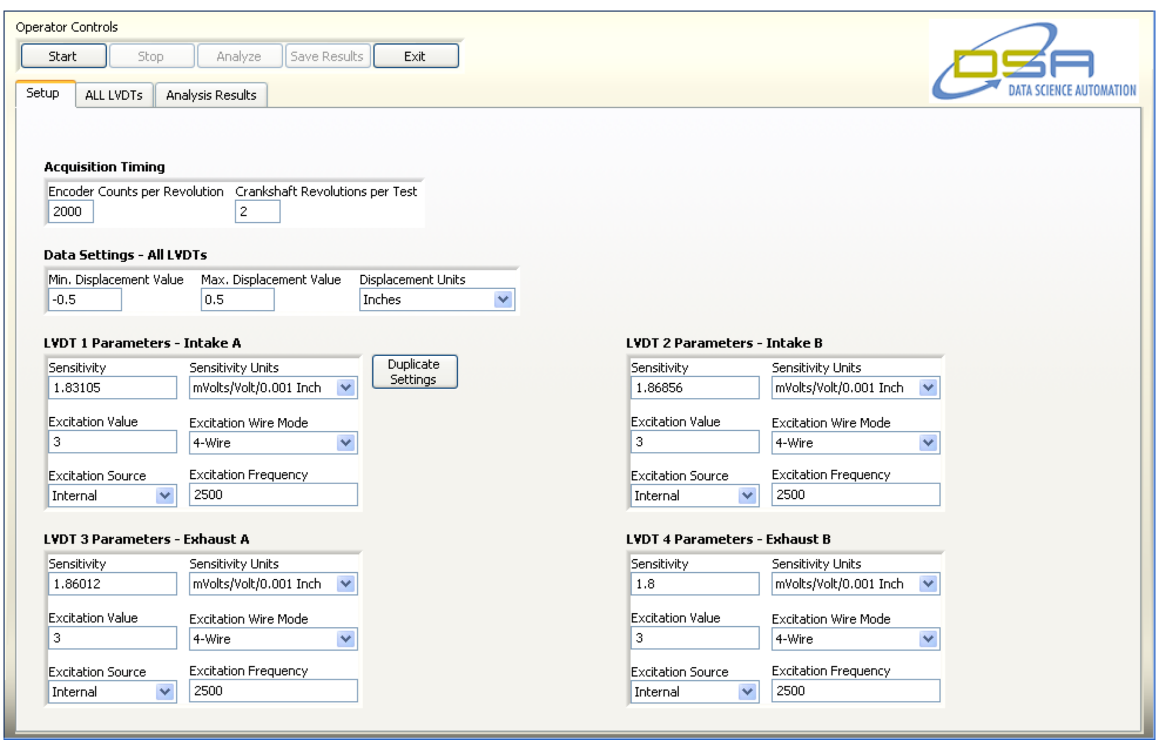

Figure 5 – Main GUI System Control Buttons and Tab Control Set Up Page

The Tab Control's second page (Figure 6) contains a subpanel. After the operator presses the <Start> button the main program Event Structure dynamically launch of the Test Controller State Machine code and places its front panel in the subpanel. The routine's front panel contains a graph presenting the raw LVDT displacement data versus the raw crank angle in absolute degrees of rotation. Utilizing a plug-in structure will simplify the process of upgraded the software if in the future the customer desires more functionality in the test procedure.

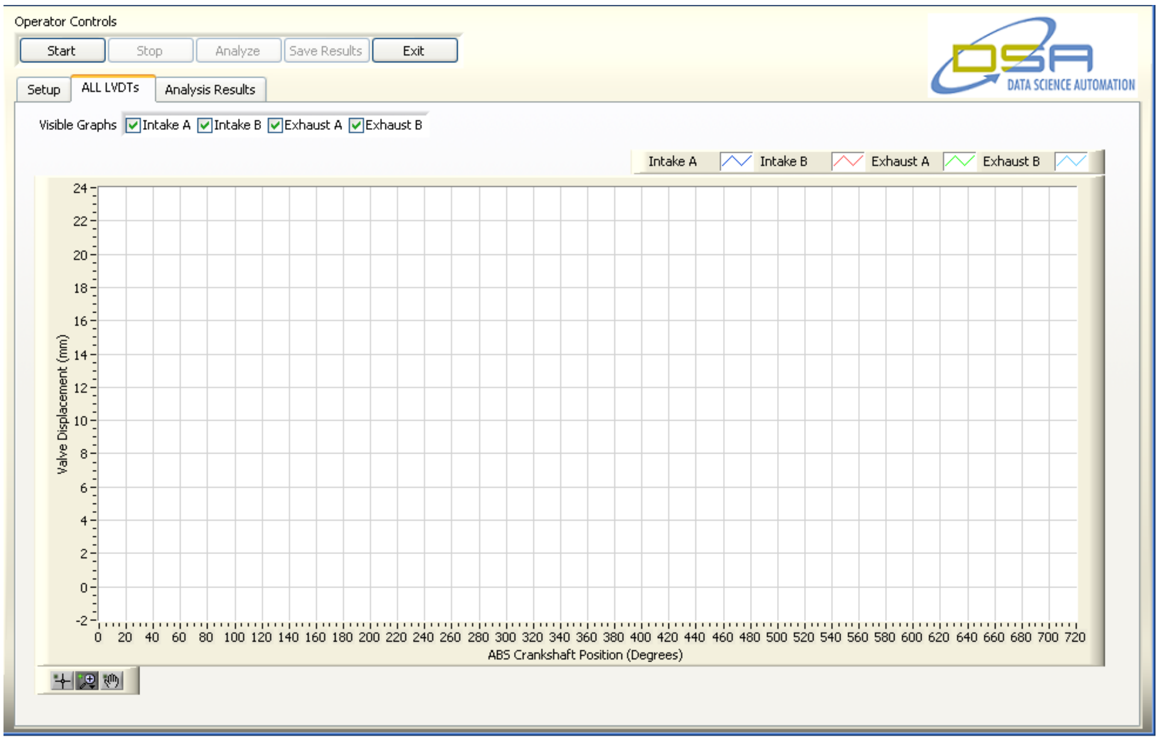

Figure 6 – Test Controller State Machine Front Panel Shown Loaded in the Sub Panel of the GUI

As can be seen in Figure 6 the operator has the option to make hiding or displaying any combination of the four LVDT graphs. LVDTs are arrangement on the engine in the order: Intake A, Intake B, Exhaust A and Exhaust B.

Test State Machine

As previously stated, the main GUI Event Structure VI launches this State Machine module dynamically. This module configures the DAQmx LVDT analog input task with the pertinent timing information and LVDT settings.

The DAQ task starts when the input signal from the CMOS Laser Sensor goes high, indicating that the piston is at TDC. The module continues running until it has collected all required data points or the operator presses the Stop button. The front panel visible in the main GUI subpanel updates with live valve displacement versus crankshaft angle data as the crankshaft rotates.

Once the task has collected all of the data points and stops, the code changes the Operator Control button enabled states allow the system operator to either analyze the data or discard it collect of another dataset.

Analysis State Machine

After the operator has been satisfied that the data collected is acceptable, pressing the Analyze button fires the corresponding Even Structure Case and runs the Analysis State Machine module. Although the entire analysis process runs in the Timeout event case state machine, it should not impact performance or GUI responsiveness since it happens very quickly.

The primary function of the Analysis State machine is to extract the data that the customer had requested. Specifically, this process consists of analyzing the valve displacement data for each LVDT for the following four discreet points:

- Crank Angle when the valve has opened 1mm
- Maximum Valve Opening (Displacement)
- Crankshaft Angle at Maximum Valve Opening
- Crank Angle when the valve is 1mm from being closed

The basic algorithm splits the data into two pieces: the data leading up to the full valve displacement and the data from full valve displacement to valve fully closed. The data tracing the valve displacement as the valve is closing is reversed and then searched for the point at which it is 1mm from being closed. This technique shortens the analysis and allows the same search mechanism to detect both 1mm open points. Figure 7 shows the implementation of this algorithm. This analysis routine resides in a reentrant VI so the code can analyze all four channels of data simultaneously.

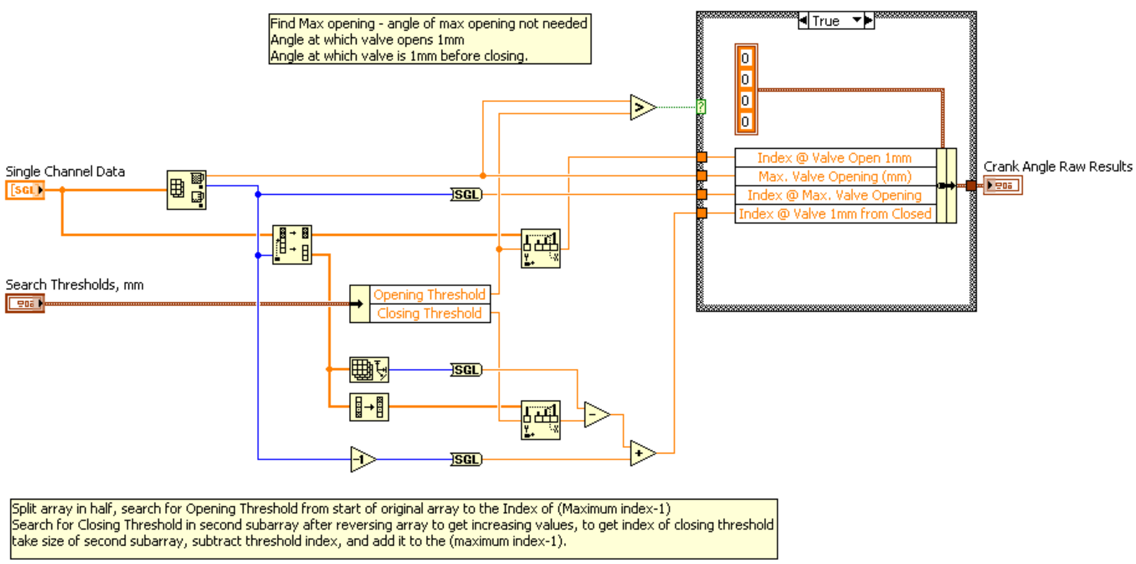

Figure 7 – Main Data Analysis Algorithm

Once the analysis is complete, the program automatically switches the GUI screen tab control to the Analysis results page (Figure 8) so the test operator can review the test results.

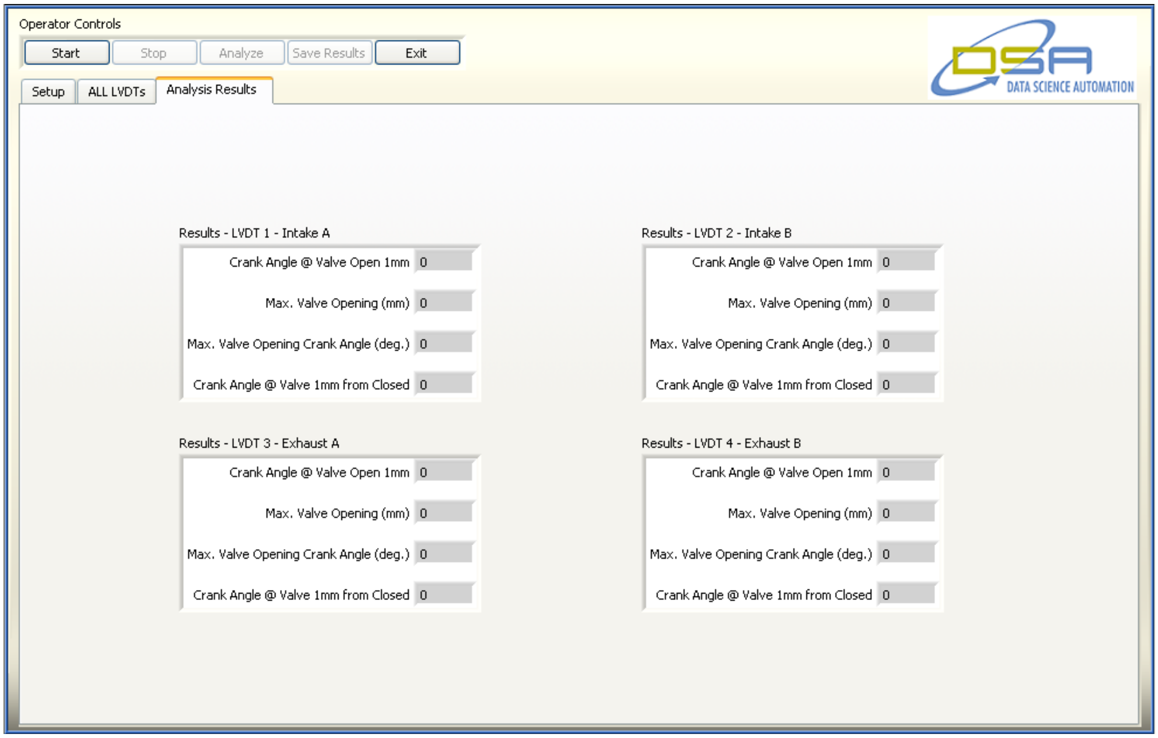

Figure 8 – Main GUI Analysis Results Tab

In addition to switching the tab control to the Analysis Results page, the code also changes the enable state of the Operator Controls buttons to allow the operator to press the <Save> button to record the data to disk.

File Data State Machine

Pressing the <Save> button launches the final State Machine in this software. The logic converts the raw data and analysis results into string format in this state machine and saved to a Comma Separated Variable (CSV) file (Figure 9).

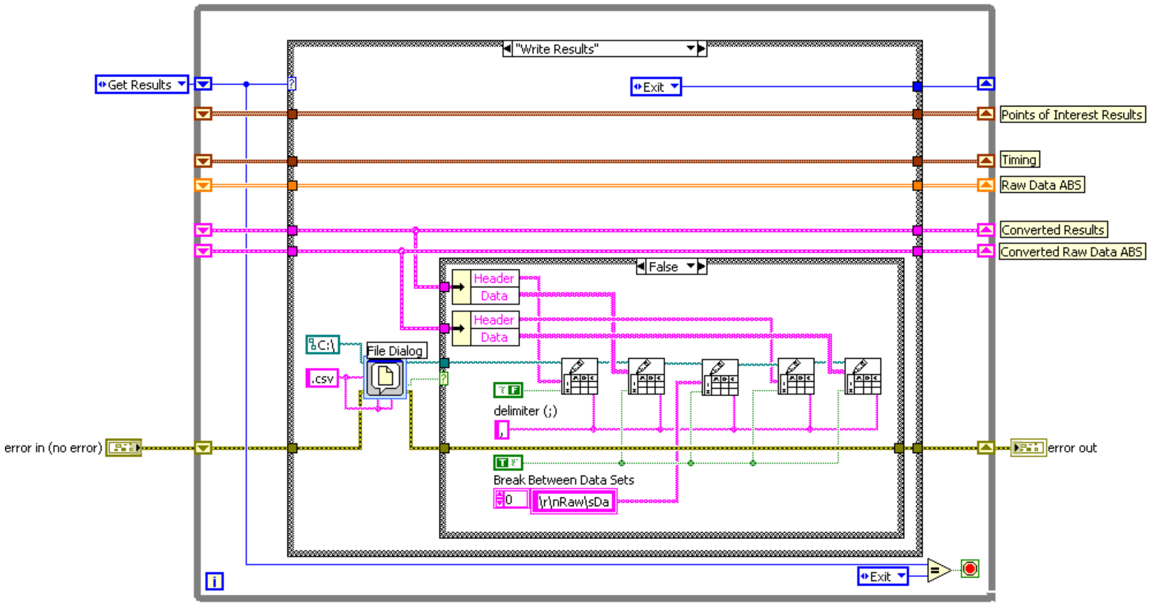

Figure 9 – File Save State Machine – Writing Data to CSV File.

The utilization of a data file module allows for future maintainability and scalability of the overall system code. The choice of using a CSV based text file means that a wide variety of applications will be able to view the data.

Summary

The combination of using National Instrument LabVIEW and National Instruments hardware and third party vendor hardware has resulted in a complete camshaft timing verification testing system that has reduced individual engine testing time by up to 30 minutes per engine. Prior to utilizing this solution the customer technicians performed camshaft timing tests on individual valves and had to analyze the data in a separate program before even knowing if the test was performed properly.