New DAQ Instrument Module
Promotes Low-Cost Automated Test

MARCOM01-2009-07-06-OI

INTRODUCTION

The basics of data acquisition & control (DAQ), involves the practice of measuring and/or controlling an electrical or physical phenomenon such as voltage, current, temperature, pressure, or frequency. Test Engineers often incorporate DAQ hardware into PC-based test systems, to (1) help validate the performance of a DUT (device-under-test), and (2) automate the test process. Depending on the application, DAQ hardware can range in price from under $100 to well over $10,000 for systems. DAQ hardware typically includes some level of Analog I/O, Digital I/O and Counter/Timer functions. Other issues that determine the quality and performance of DAQ hardware include the speed it operates, the resolution & accuracy it offers, how well it interfaces and how easy it is to program.

Overton Instruments has developed a new DAQ instrument that is specifically designed for Test Engineers, and we call it Check-MATE. Check-MATE balances the right amount of Analog and Digital I/O necessary to satisfy the broadest array of test & measurement applications. Check-MATE can be used to support automated test, by connecting to a PC (via USB interface). Check-MATE employs a robust set of commands, which the PC outputs as simple ASCII character strings. The compact size allows the Check-MATE to be mounted directly inside mechanical test fixtures, custom instrument enclosures or full-blown ATE systems.

The purpose for this APP NOTE is to illustrate the flexibility and versatility the Check-MATE can bring to you next test development project.

OVERVIEW

For less then $100, the Check-MATE is the perfect instrument for automating a wide variety of mix-signal test applications. Many test solutions can be quickly built by connecting the Check-MATE to a PC laptop or desktop, and then running our GUI software. No external power source is required, since power is supplied through the USB interface. Easy access to the hardware is made available through a convenient collection of screw terminal connectors.

Analog I/O

The Check-MATE offers 8 single-ended analog input channels. Each input channel provides 12-bit resolution and 4 programmable ranges (i.e., +5V, ±5V, +10V and ±10V).

The Check-MATE includes a single analog output with 12-bit resolution. The output is programmable (unipolar 0-10V or bipolar ±10V).

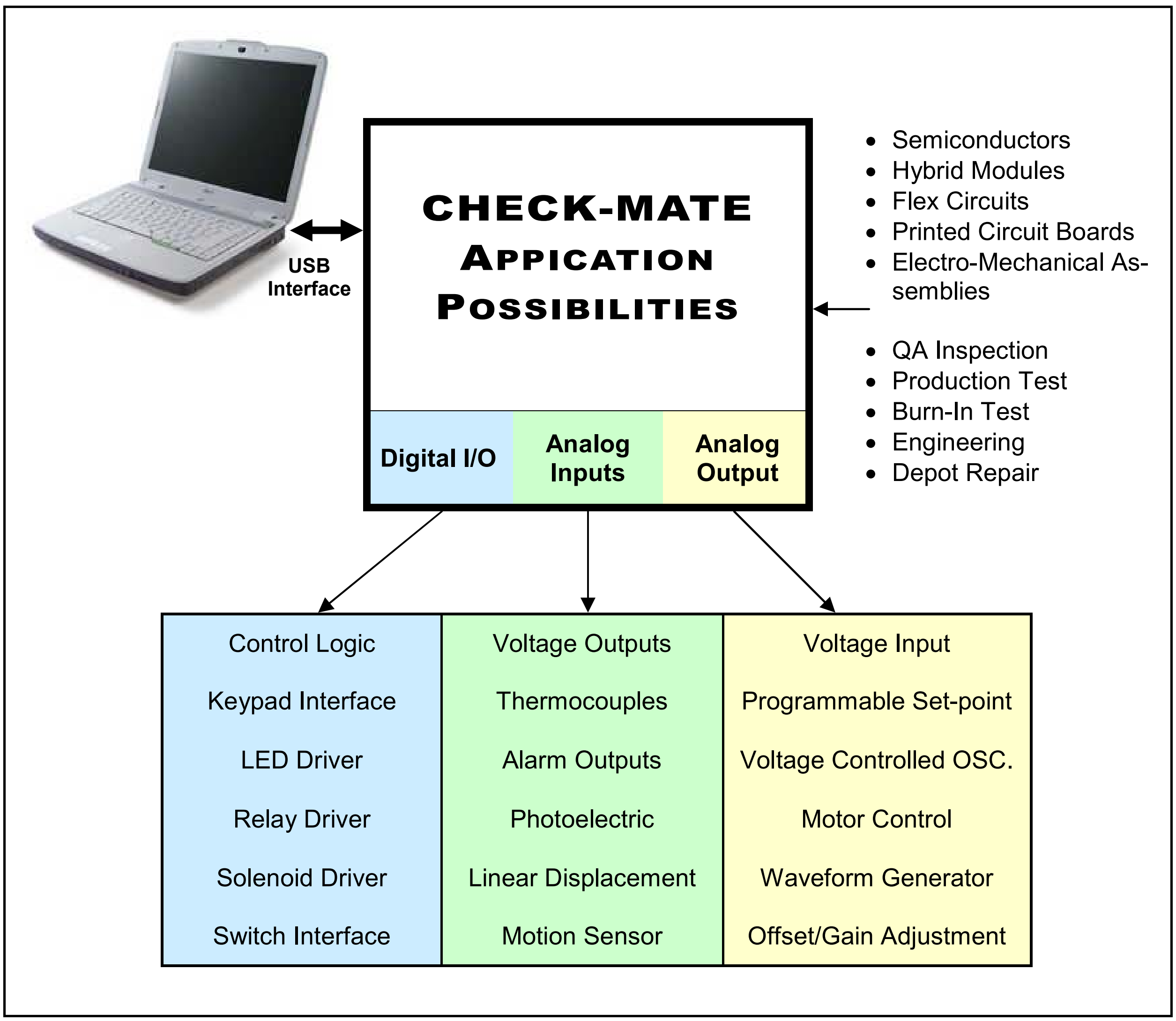
Digital I/O

The Check-MATE provide 8 digital input/output bits (each bit is independently programmable). Special features include programmable “weak” pull-ups and 25mA output current sunk/sink.

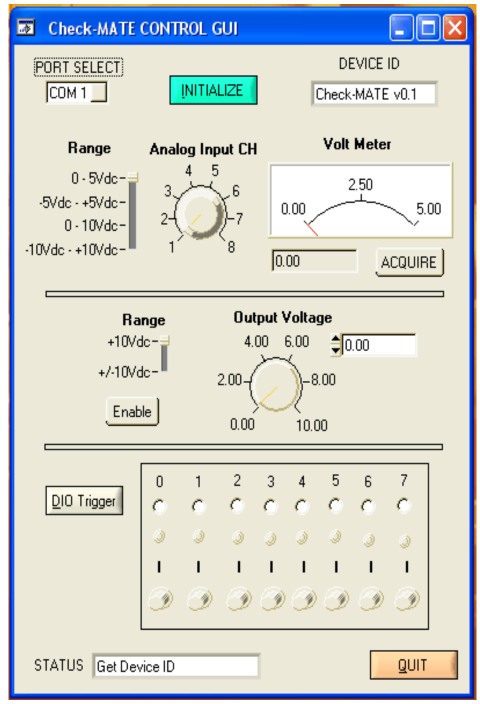
BUILT FOR TEST ENGINEERS

The Check-MATE is designed to be a Swiss Army Knife for Test Engineers. Whether you need to perform a quick experiment or build a complete Functional Test solution. Simply wire the Check-MATE to the DUT, connect the PC, write some code and let the testing begin. Since the Check-MATE supports a simple set of commands, it can be easily programmed using popular software languages (such as C/C++, Visual BASIC, LabWindows and LabView). Figure 1, below highlights the many applications that are possible with Check-MATE.

Figure 1. Illustrates the many test & measurement possibilities for Check-MATE

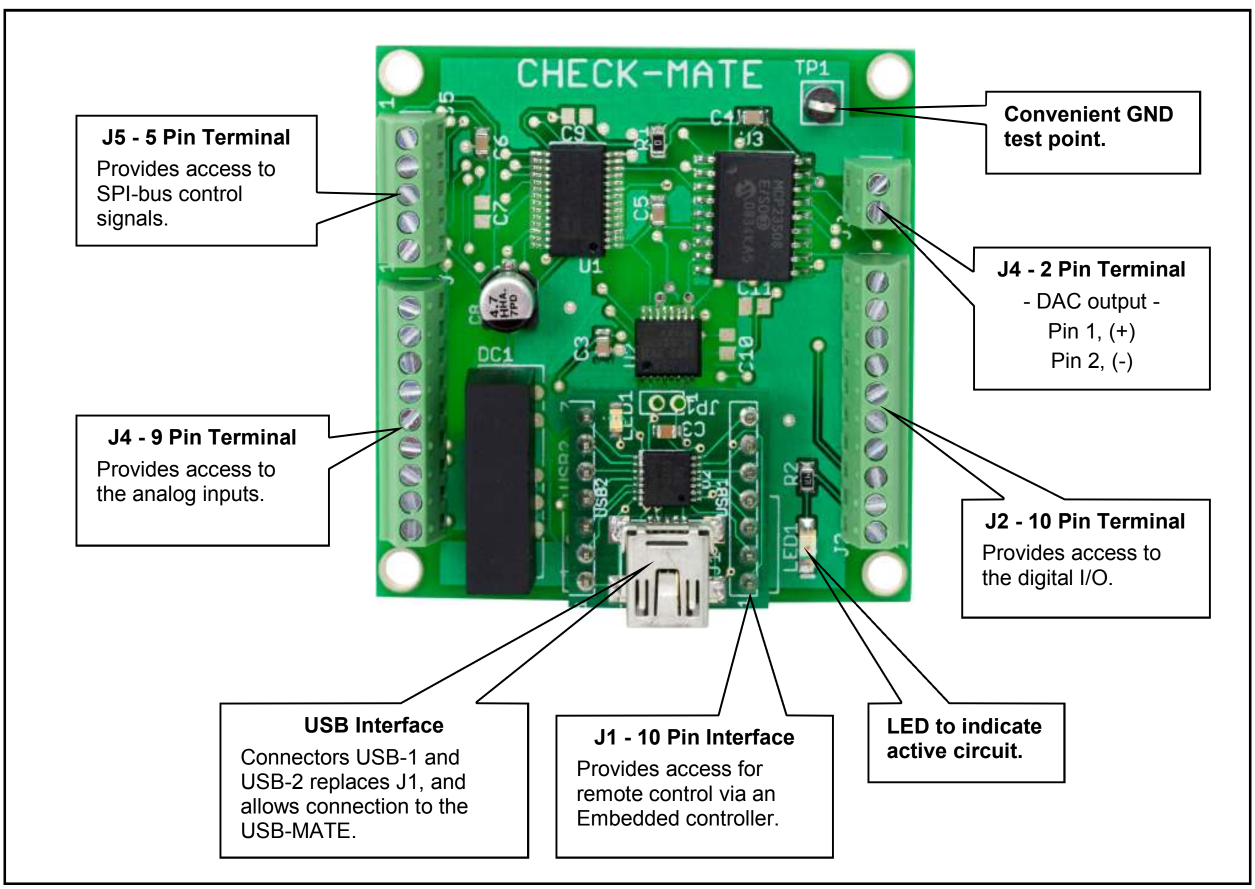


GET ACQUAINTED



The board layout for the Check-MATE is shown below. Access to the hardware is provided through a convenient collection of screw-terminal connections.

The starting point for developing code to control the Check-MATE, begins with acquainting yourself with its Serial Command Set. The serial commands are a sequence of ASCII characters that originate from the PC and are designed to instruct the Check-MATE to perform specific functions. There are two ways to exercise the serial commands, (1) using HyperTerminal (a Windows APP) or (2), run our Virtual Instrument Panel software (shown on the right). The details for using the serial commands are available in the Users Manual, which can be downloaded by going to www.Check-MATE.info.



PROJECT EXAMPLE

A hypothetical manufacturer of custom braking systems (for automobiles that compete in NASCAR), was experiencing intermittent failures with a Braking Control Module (BCM). The BCM is a hybrid SOC (system-on-chip) device. The BCM receives inputs from braking sensors and outputs a digitally processed signal to the host processor. After the BCM is soldered to the PCB, the entire assembly is conformally coated and then functionally tested. Because of the conformal coating (if the PCB fails the test), replacing the BCM will take consider time and expense. To remedy the problem, a Test Engineer was asked to build a test solution that would test the BCM devices prior to being soldered onto an actual PCB. Figure 2 shows a PC-based test system for testing the BCM.

The goal of the test is to subject the BCM to a rigorous regression test, which is designed to force the DUT to reveal the mystery failure mode. The test sequence involves stimulating the BCM input with a waveform that mimics a vehicle braking pattern, and then capture the BCM output waveform with a digital oscilloscope. The captured waveform is then compared to a known-good waveform. If the captured waveform is within a certain percentage of the known-good waveform, the test is successful. During the test process, the BCM is subjected to several vehicle braking patterns. And, each braking pattern is further manipulated by changing the BCM filtering mode. The filtering mode causes the BCM to change the output signal. Depended upon the filtering mode, the BCM will subject the input waveform to a different filtering algorithm. In addition, the BCM is further manipulated by varying its Master Clock. Varying the Master Clock frequency will impact the output waveform in a predictable manner. If the BCM is successful though all permutations and combinations of waveform analysis, then it is deemed acceptable for PCB installation.

In this application the Check-MATE plays a vital roll. The Check-MATE digital I/O is used to control power to the DUT, provide input control logic and monitor output alarm conditions from the DUT. The DUT operates from a number of power inputs (+3.3V, +5V, ±12V, and including voltage references). The Check-MATE analog inputs are used to monitor those voltages. The Check-MATE analog output is used to provide an adjustable bias voltage (±10Vdc) for the DUT. A number of the Check-MATE resources are used by the Power Control & Signal Conditioning circuit. This block encompasses a collection of custom circuits that provide an interface between the Check-MATE and the DUT.

Before power is applied to the DUT, the PC sends the Check-MATE a series of commands that initialize the module (i.e., set the DIO direction, set the analog input channel range, and zero the analog output). Then, the PC sends the Check-MATE a series of commands to apply DUT power, measure DUT input power rails and assert the DUT reset line to clear it.

Next the PC sends the Pulse-MATE a series of commands to initialize it (i.e., set the pulse-width, frequency and amplitude). The Pulse-MATE is a programmable pulse generator that can produce a 1Hz-10MHz pulse (in 1Hz increments). The Pulse-MATE is used to generate an exact 6.837MHz Master Clock for the DUT. Once the Master Clock is active, the PC can query the Check-MATE DIO to determine if the DUT has any active alarms.

PROJECT EXAMPLE, continued

Now the DUT is ready to receive its initialization and mode configuration set-up. To do that, the PC sends the SPI-Bus Controller a series of initialization commands. The SPI-Bus Controller is a programmable SPI-Bus interface which allows the PC to directly control the DUT. Control of the SPI-Bus Controller is provided though a set of API libraries which are linked to the test program during compile-time.

Finally the PC sends the DSO (Digital Storage Oscilloscope) and the AWG (Arbitrary Waveform Generator), a series of commands to initialize those instruments as well. The AWG was pre-configured with the braking pattern waveforms (to stimulate the DUT). And likewise, the DSO was pre-configured with the “known-good” waveforms it will need to qualify the test results.

To perform the actual test sequence, the PC enters into the following program loop:

1. The PC commands the AWG to output a brake pattern waveform,
2. The PC commands the DSO to capture the waveform and make comparison,
3. The PC queries the DSO for the results and determines Pass/Fail,
4. The PC queries the Check-MATE for alarms and determines Pass/Fail,
5. The PC commands the DUT to change the filter mode and repeat the cycle,
6. The PC commands the Pulse-MATE to change the Master Clock frequency and repeat the cycle,
7. The PC commands the Check-MATE to change the DUT bias voltage and repeat the cycle,
8. The PC commands the AWG to output the next brake pattern and repeat the cycle.

It is important to note that in the program loop described above, the Check-MATE is pivotal to trapping illusive failure modes, by monitoring alarm conditions generated by the DUT.

CONCLUSION

The Check-MATE makes it possible for Test Engineers to automate a wide array of test & measurement applications quickly, reliably and affordably.

Figure 2. A PC-driven test system performing a QA Test, featuring the Check-MATE.