GPIB Command Library &

I2C CBIT expander PCB card

ELECOMP Capstone Design Project 2024-2025

Sponsoring Company:

Vicor
1 Albion Rd
Lincoln, RI 02865
http://www.vicorpower.com

Company Overview:

Vicor Corporation designs, develops, manufactures and markets modular power components and complete power systems based upon a portfolio of patented technologies. Headquartered in Andover, Massachusetts, Vicor sells its products to the power systems market, including enterprise and high-performance computing, industrial equipment and automation, telecommunications and network infrastructure, vehicles and transportation, aerospace and defense.

Technical Directors:

Nathan Shake (ELECOMP '20)

Test Engineer
Nshake@vicr.com

Daniel Hartnett (ELECOMP '21)

Test Engineer
DHartnett@vicr.com

Al Binder
Director Test Development and Product Engineering

ABinder@vicr.com

Josh Hoyle
Principal Engineer
JHoyle@vicr.com

Kris Keeble
Sr. PCB Designer
KKeeble@vicr.com

Project Motivation:

Vicor uses a variety of ATE (Automated Test Equipment) to be able to electrically test die on silicon wafers that go into Vicors assembled parts. These test systems are very helpful in allowing us to apply a wide variety of test conditions while using the minimum amount of connection points between the tester and the die. However what they make up for in variety they tend to lose out in “edge case” applications like high current and very accurate readings.

Another issue that occurs here and there while using the ATEs is that there is a discrepancy between the results from bench testing and silicon testing. When new parts are designed, they go through a phase where the parts are tested on the bench using various oscilloscopes, power supplies, and multi-meters. Compared to the ATE testing, these utilities tend to be a bit more accurate and are possible to test the edge cases. However, they are far more manual in test setups and are only able to test a singular part at a time.

To be able to combat the two issues of result discrepancy and edge case testing, we would like to integrate the bench top resources into our ATE test setups. This way we would be able to gain more accurate readings while also being able to test at currents past the limitations of our ATEs. It will also allow us to better correlate between the bench setup by being able to apply the exact same test conditions with the same resources. We will also be able to isolate issues that arise by being able to see if it is coming from our ATE or our test hardware.

With our ATE electrical test limitation, there is also a limitation with the amount of resources available to perform the electrical test. One way we combat this is by using “Cbits” which are simply just relays controller by the ATE to be able to attach and detach resources to and from the DUT (Device Under Test). However Cbits, have two main limitations on the amount we are able to use. One issue being the physical size, which can be rectified by using different relays or a larger PCB for a larger workable area. The other issue is the amount of addressable resources in the ATE that allow us to trigger Cbits. This issue is a little more in-depth and is not as simple of a solution. A solution was created a couple of years ago but was never expanded upon due to it not being needed. However with parts becoming more complex and our test resources not growing alongside it, we need to have a fully fleshed out solution ready to go.

Anticipated Best Outcome:

Anticipated Best Outcome:

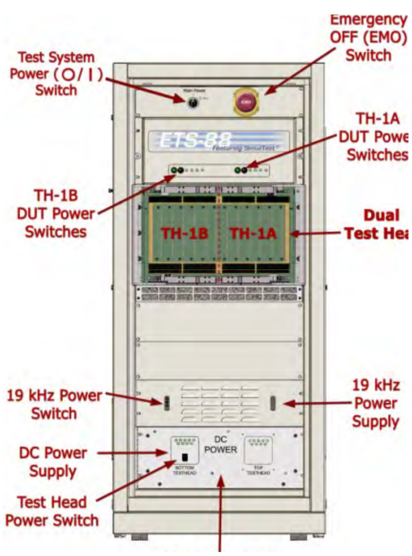
In order for Vicor to put the Bench top supplies and ATE integration into place, it will need a fully fleshed out library of commands that allow full control over the Bench top supplies. Vicor will also need any supporting header files to accomplish this. This includes controlling the forcing values of bench top power supplies as well as being able to turn them on and off as necessary. With this, Vicor will be able to integrate the code from the library into our test code to allow seamless integration of the Bench top resources.

The ABO for the second part of the project will be a fully functioning CBit extender that utilizes I2C commands from the ATE to be able to control relays past the capability native to the ATE. This CBit extender will be need a custom PCB designed for it along with all necessary connections to adapt this PCB into our test hardware.

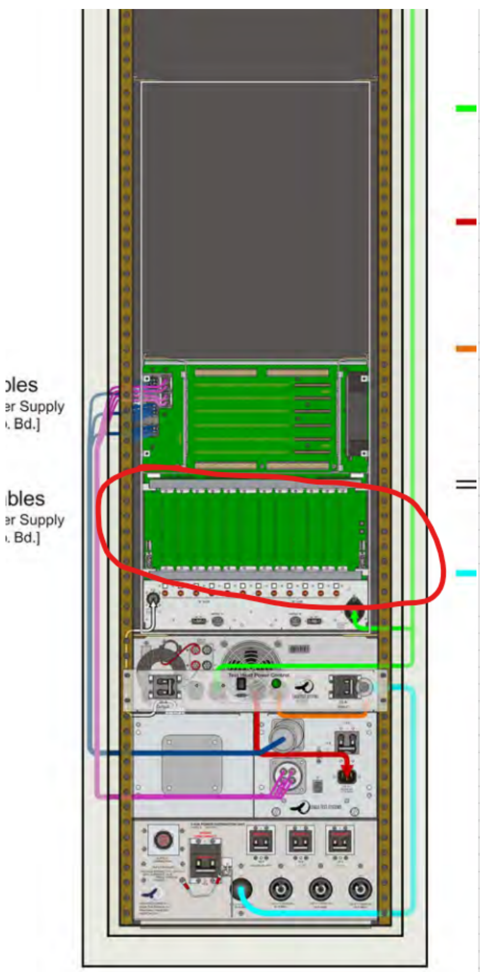
Project Details:

GPIB Command library

In order to be able to integrate our ATEs with bench top resources we need 4 things in total. There is the ATE, bench resource, cable to connect them, and the communication code. For the ATE we will be using either an ETS-300 or an ETS-88. Pictured below is an ETS-88:



These ATEs house about 12 resource boards per test system split between cbits, power resources, and timing resources. This gives us two limitations, the amount of resources we can fit in the test system but also the configuration of it as well. The reason we are limited on the configuration is that when testing product at various test houses, the system needs to be configured exactly the same. This means we can't just go and fill the rest of the slots as that would mean the various test houses would need to change the system configuration every time they need to setup our product. This wastes a lot of time and it also gives the potential for issues to arise in the ATEs.



	A	B	C	D	E	F
1	Tester Model			ETS88		
2	Floating resource				Probe/FT	Probe/FT
3	Slot	Section	COMM Chann	P/N		
4	0		0/1			APU-12
5	1		2/3			SPU-112
6	2	TH-1A	4/5			APU-12
7	3		6/7			SPU-112
8	4		8/9			APU-12
9	5		10/11			APU-12
10	6		12/13			APU-12
11	7		14/15			SPU-112
12	8	TH-1B	16/17			APU-12
13	9		18/19			SPU-112
14	10		20/21			APU-12
15	11		22/23			APU-12
16	12		24/25		APU-12	APU-12
17	13		26/27		SPU-112	SPU-112
18	14	TH-2A	28/29		APU-12	APU-12
19	15		30/31		SPU-112	SPU-112
20	16		32/33		APU-12	APU-12
21	17		34/35		APU-12	APU-12
22	18		36/37		APU-12	APU-12
23	19		38/39		SPU-112	SPU-112
24	20	TH-2B	40/41		APU-12	APU-12
25	21		42/43		SPU-112	SPU-112
26	22		44/45		APU-12	APU-12
27	23		46/47		APU-12	APU-12
28	xternal Resourc	TH-1A				
29	xternal Resourc	TH-1B				
30	Slot	Section	COMM Chann			
31	TCB-1A					TCB-1A
32	QTMU	TH-1A				QTMU
33	DPU-0					DPU-16
34	DPU-1					
35	TCB-1B					TCB-1B
36	QTMU	TH-1B				QTMU
37	DPU-2					DPU-16
38	DPU-3					
39	TCB-1A				TCB-1A	TCB-1A
40	QTMU	TH-2A			QTMU	QTMU
41	DPU-4				DPU-16	DPU-16
42	DPU-5					
43	TCB-1B				TCB-1B	TCB-1B
44	QTMU	TH-2B			QTMU	QTMU
45	DPU-6				DPU-16	DPU-16
46	DPU-7					
47	EV&MST Shell Version				MST-2021A Windows 10	MST-2021A Windows 10
48					2021.1.1005.15	2021.1.1005.15
49						
50						

Adding on a multimeter or a bench top power supply will eliminate the chance of damage and be a much more cost effective solution. We would also be able to add more conditions for edge case testing during development without compromising our ability to transfer the products and test them at our various test houses.

To integrate the bench top resources, we are planning to use GPIB(General purpose interface bus) to be able to communicate between the ATE and bench top supplies. Since this is a global standard, it will increase availability and make it more cost effective as well. We will also be using SCPI (Standard commands for programmable instruments) to be able to communicate between the ATE and the bench top resource.

To start with we will be working with a multimeter to take a step into this integration. Along with this we will design a PCB to allow us to really see what it takes to integrate the two together. Once the Multi-meter integration is done we will move along to a bench top power supply to start filling out the communication library.

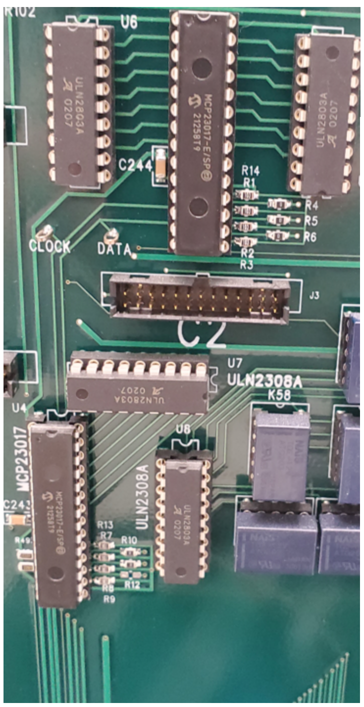
GPIB cable between computer and digital multimeter

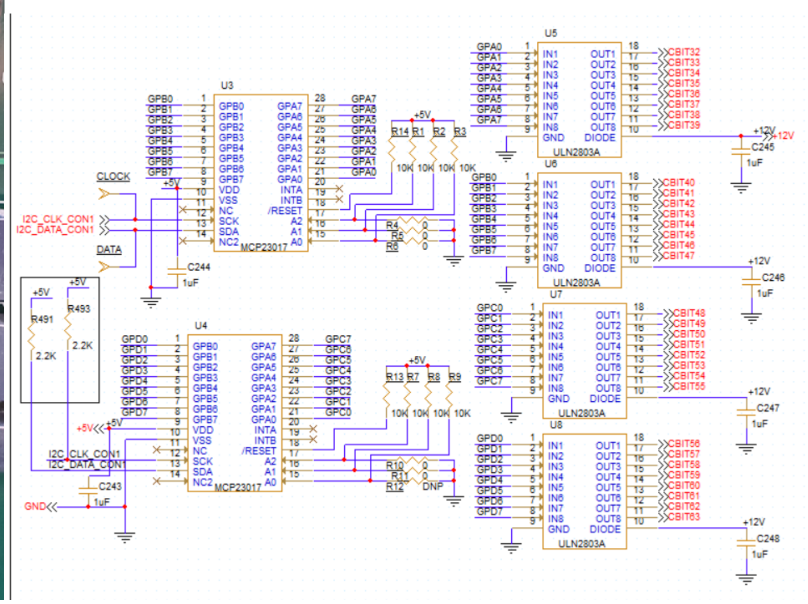
Sample code:

```
gpibsend(gpib_addr, "CONF:VOLT:DC %f,%f", range,res);

gpibsend(gpib_addr, "SAMP:COUN 1");

gpibsend(gpib_addr, "SENS:VOLT:DC:NPLC? DEF");

gpibsend(gpib_addr, "READ?");

gpibreceive(gpib_addr, rcvd, 25);
```

I2C Cbit Expander PCB

For the second part of the project, we will be expanding on the Cbit expander circuitry that we have used a bit in the past. This was designed of a single use case at the time, however since the parts we are required to test are becoming more complex, we want to expand on the pervious iteration and improve it to an expandable resource.

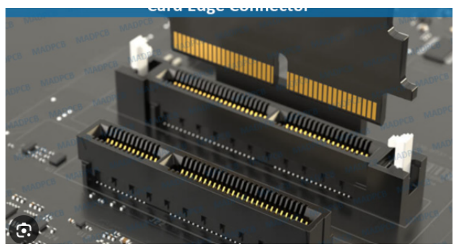


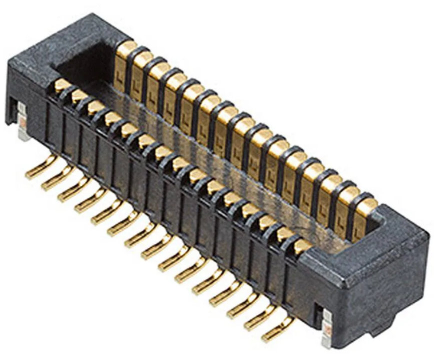


The solution from before will give a good starting base to work with. The main thing that we will work on is developing this solution onto a standalone board so that we will be able to add it and remove it depending on if the part under test requires it. This will also give us the added benefit of being able to easily change out the board if it stops working.

Being able to use the solution on other board brings up two key design features when designing the PCB. One being the connectors used to attach it to the board, and the size of the PCB itself.

There are a wide variety of different types of connectors for PCBs, we will be looking for one that will allow us to make good connection but also be removable in case of not needing it or it needs to be replaced.





With our test hardware, we are fairly limited to room already. That is why the size of this will give us some design challenges. This will also play into the connectors we are able to choose as well, since we will need a large number of connections, while keeping the size down.

The physical portion is not the only part we are looking to expand from first iteration. Similar to the ATE to bench supply integration, we will need to control this Cbit Extender with the ATE. To do this we will be using the ETS-88 as pictured earlier, from there we will be using the communication protocol of I2C to have full control of the Cbit Extender, allowing us to easily integrate it into and projects it will be necessary on.

Hardware/Electrical Tasks:

- Schematic Design
- PCB Layout
- Component Research
- Using ATEs
- Assembling and Soldering final PCB

Firmware/Software/Computer Tasks:

- Researching communication protocols (I2C, SCPI)
- Communicating Over GPIB to Control Benchtop Supplies
- Communicating over I2C to Control individual components on a PCB
- Creating C++ code for ETS-88 to use
- Creating a library for easy integration with in place program

Composition of Team:

1 Electrical Engineer & 1 Computer Engineer **(preference will be given to those enrolled in the PCB Design Course, taught by Mike Smith on Thursday evenings)**

Skills Required:

Electrical Engineering Skills Required:

- Basic Electrical Engineering laws (Ohms Law)
- Basic Debug skills
- Basic ability to solder
- Schematic Capture
- Basic PCB design understanding
- Know how to use multimeter and oscilloscope

Computer Engineering Skills Required:

- Knowledge of C++ basics
- Know how to write functions

Anticipated Best Outcome's Impact on Company's Business, and Economic Impact

Should the ABO be achieved, it will allow us to better prepare and test our parts to know how they will perform in the field and under specific conditions.

The end use of these projects could enable:

- Remove offset of output readings.
- Measure voltage to a higher accuracy.
- Use high current supply as load and digitize.
- Have access to more relays for more complex projects

Broader Implications of the Best Outcome on the Company's Industry:

Having more ways to test parts and verify their functionality allows us to fully understand how components work under certain applications. Having a way to better prepare parts for the field increases reliability and there increase the use life of the parts.