

Hercules Industrial User Interface
PCA 1256673
Revision A

Highlighted text needs to be updated. Any section header highlighted means
entire section needs updating

Revision Table

Revision	Changes	Engineer	Date
A	Initial Version	BAP	2023-06-26

Table of Contents

Setup:	2
Initialize FCT Mode:.....	2
Revision Verification:.....	2
Write Serial Numbers:.....	3
Serial Flash:.....	3
Trusted Platform Module (TPM):.....	3
RTC:.....	3
Power Input:.....	3
USB Power:.....	4
5V Sensor Power:.....	4
Digital Input:.....	4
Analog Input:	5
Low Side Driver Output:.....	5
Hour Meter Output:.....	6
Membrane LED:.....	6
Switch Membrane:.....	8
Display Supply Test:	9

Potentiometer Check:.....	9
LCD Color:.....	9
Touchscreen:	10
Backlight Driver:	11
Backup Camera:.....	11
CAN Bus:	12
RS-232:	12
UART (5V):.....	12
USB Host:	12
USB device:.....	12

Setup:

A fixture has connections with pogo pins for all necessary points. All Test commands are sent to the board using the UART debug port (J16). The board executes the commands and returns status over the same interface. The UART terminal should be configured for 115200 baud, no stop bits and 8 data bits. Operator input is required to verify LCD and touch panel function if the LCD is installed in the test fixture.

1. Install user interface assembly in machine.
2. Turn power on. Power supply is set to 24.0V and should be capable of driving a 2A load. Power is applied as indicated: COM goes to J3-1. +24V goes to J3-2 and J4-2. The board will not operate if J4-2 is below 8V.

Initialize FCT Mode:

1. Login to the UART debug port. Username is “root” and password is “am4”. Note: the password characters will not echo back to the terminal for security reasons.
2. Initialize FCT mode by typing “fct” and pressing return.

Revision Verification:

1. Run command to bring up revision menu
2. The Machine app revision shall match the revision number listed in the released drawing of 1266673.
3. The UI board revision shall match the revision number listed in the released drawing of 1266673.

Write Serial Number:

1. Write board serial number (where [SN] is the serial number on the board) to EEPROM by sending the following command over THE DEBUG INTERFACE: config p=499,[SN]

Response must include:

Write Complete.

2. Wait 5 seconds and cycle power to board.
3. Read the board serial number from memory by sending the following command over the debug interface: **“config r=499”**
4. Response must include the serial number written to the board.
5. Set to functional test mode by sending the following command over the debug interface:
6. **“machine t=1”**.

Serial Flash:

1. Run X command
2. Read back and verify data

Trusted Platform Module (TPM):

1. Execute TPM command
2. Verify data returned matches X acceptance criteria

RTC:

1. Run command to program RTC to current local time
2. Power cycle board for at least 10 seconds
3. Run command to read back time and verify it has incremented for approximately the amount of time since it was set.

Power Input:

1. Read status of battery power with command TBD suggestion “power ?”
2. Battery voltage will be returned. Reading shall be $24V \pm 2.2\%$
3. Read status of key switch power with command TBD suggestion “power ?”
4. key switch voltage will be returned. Reading shall be $24V \pm 2.2\%$
5. Apply 24V to J4-3
6. Read status of charger power with command TBD suggestion “power ?”
7. Charger voltage will be returned. Reading shall be $24V \pm 2.2\%$
8. Run command TBD suggestion “power ?”

9. Batt low input shall be false
10. Key switch input shall be false
11. Charger input shall be false
12. Lower voltage at J3-2, J4-2, and J4-3 to 6.5V
13. Run command TBD suggestion "power ?"
14. Batt low input shall be true
15. Key switch input shall be true
16. Charger input shall be true

USB Power:

1. Run command X to enable power to the J5
2. Connect a 5W, 10 Ω load to the +5V and GND connections of the USB A connector (J5-1 and J5-4)
3. J5-1 shall measure 5V \pm 0.5V
4. The USB switch has temperature base current limit feature. It will current limit at approximately 1.5A
5. Connect a 25W, 1.5 Ω load to the +5V and GND connections of the USB A connector (J5-1 and J5-4)
6. This will draw 3.33A through a circuit with a 1.5A current limit
7. Send command "inputs ?"
8. **IN_USB1_FAULT** shall read "TRUE"

5V Sensor Power:

1. Measure J18-13 WRT J18-14 with a DMM. Must measure 5.0V \pm 1V.
2. Apply a 50 Ω 1W load between J18-13 and J18-14
3. Measure J18-13 WRT J18-14 with a DMM. Must measure 3.27 \pm 0.5V.
4. Remove load from connector J15

Digital Input:

1. Read inputs with command inputs ?
2. All inputs shall be false
3. Apply GND to J10-1
4. Read inputs with command inputs ?
5. Mem_digin_1 shall be true
6. Apply GND to J10-2
7. Read inputs with command inputs ?
8. Mem_digin_2 shall be true
9. Apply 24V to J4-6

10. Read inputs with command inputs ?
11. Digin_1 shall be true
12. Apply 24V to J18-4
13. Read inputs with command inputs ?
14. Digin_2 shall be true
15. Apply 24V to 18-5
16. Read inputs with command inputs ?
17. Digin_3 shall be true
18. Apply 24V to J18-6
19. Read inputs with command inputs ?
20. Digin_4 shall be true
21. Apply GND to J18-7
22. Read inputs with command inputs ?
23. Digin_5 shall be true
24. Apply GND to J18-8
25. Read inputs with command inputs ?
26. Digin_6 shall be true
27. Apply GND to J18-9
28. Read inputs with command inputs ?
29. Digin_7 shall be true
30. Apply GND to J18-10
31. Read inputs with command inputs ?
32. Digin_8 shall be true

Analog Input:

1. Read analog inputs with the command inputs_a ?
2. All analog inputs shall be 0V
3. Apply 3V to J18-11
4. Read analog inputs with the command inputs_a ?
5. AIN_1 reading shall be $3V \pm 5\%$
6. Apply 4V to J18-12
7. Read analog inputs with the command inputs_a ?
8. AIN_1 reading shall be $4V \pm 5\%$

Low Side Driver Output:

1. Connect a 12.1k 1W resistor to each of the following pins
 - a. J19-1 to J3-2.
 - b. J19-2 to J3-2

- c. J19-3 to J3-2
 - d. J19-4 to J3-2
 - e. J19-5 to J3-2
 - f. J19-6 to J3-2
 - g. J19-7 to J3-2
 - h. J19-8 to J3-2
2. Turn on low side driver 1 output by sending the following command over THE DEBUG INTERFACE: **Command X**
 3. Confirm the driver turned on by measuring J7-5. J7-5 must measure less than 1V WRT GND.
 4. Turn off low side driver 1 output by sending the following command over THE DEBUG INTERFACE: **Command X**
 5. Repeat steps 2 through 4 for all 8 low side drivers
 6. Low side driver chart with commands for reference

LSD Name	Pin Number	Turn ON Command	Turn OFF Command
LSD_1	J19-1	TBD	TBD
LSD_2	J19-2	TBD	TBD
LSD_3	J19-3	TBD	TBD
LSD_4	J19-4	TBD	TBD
LSD_5	J19-5	TBD	TBD
LSD_6	J19-6	TBD	TBD
LSD_7	J19-7	TBD	TBD
LSD_8	J19-8	TBD	TBD

Hour Meter Output:

1. Connect a 4.02kΩ resistor between J3-6 and J3-1
2. Enable hour meter with **Command X**
3. Measure J3-6 with a DMM. Measurement shall be 5V ± 0.5V
4. Turn of hour meter with **Command X**

Membrane LED:

1. The test setup should have a 1.5kohm load resistor from each LED output LED1 through LED32 to 3.3V.
2. Turn off all LEDs by sending the following command over THE DEBUG INTERFACE: “led o=0”
3. Confirm that all membrane panel LEDs are turned off by sending the following command over THE DEBUG INTERFACE: “led o=0”.
4. Confirm that J8-10 through J8-24 and J8-28 through J8-44 are all turned off.
5. Turn on LED 1 by sending the “led x=1” command over THE DEBUG INTERFACE. The response must include:

led_Set(47, 1:LED_MODE_ON, False)

6. Confirm that J8-10 is turned on.
7. Turn off LED 1 by sending the “led y=1” command over THE DEBUG INTERFACE. The response must include:

`led_Set(30, 1:LED_MODE_ON, False)`

8. Confirm that J8-10 is turned off
9. Repeat steps 5 through 8 for all 32 LEDs.
10. LED chart with commands for reference:

LED Name	Pin Number	Turn ON Command	Turn OFF Command
LED_1	J8-10	led x=1	led y=1
LED_2	J8-11	led x=2	led y=2
LED_3	J8-12	led x=3	led y=3
LED_4	J8-13	led x=4	led y=4
LED_5	J8-14	led x=5	led y=5
LED_6	J8-15	led x=6	led y=6
LED_7	J8-16	led x=7	led y=7
LED_8	J8-17	led x=8	led y=8
LED_9	J8-18	led x=9	led y=9
LED_10	J8-19	led x=10	led y=10
LED_11	J8-20	led x=11	led y=11
LED_12	J8-21	led x=12	led y=12
LED_13	J8-22	led x=13	led y=13
LED_14	J8-23	led x=14	led y=14
LED_15	J8-24	led x=15	led y=15
LED_16	J8-28	led x=16	led y=16
LED_17	J8-29	led x=17	led y=17
LED_18	J8-30	led x=18	led y=18
LED_19	J8-31	led x=19	led y=19
LED_20	J8-32	led x=20	led y=20
LED_21	J8-33	led x=21	led y=21
LED_22	J8-34	led x=22	led y=22
LED_23	J8-35	led x=23	led y=23
LED_24	J8-36	led x=24	led y=24
LED_25	J8-37	led x=25	led y=25
LED_26	J8-38	led x=26	led y=26
LED_27	J8-39	led x=27	led y=27
LED_28	J8-40	led x=28	led y=28

LED_29	J8-41	led x=29	led y=29
LED_30	J8-42	led x=30	led y=30
LED_31	J8-43	led x=31	led y=31
LED_32	J8-44	led x=32	led y=32

Switch Membrane:

1. Read membrane status inputs by sending the following command over the debug interface: "inputs m". DUT will return a response similar to the following:

```
>inputs m
inputs_State_button( 0 , State=False )
inputs_State_button( 1 , State=False )
inputs_State_button( 2 , State=False )
power_Battery_Is_Empty()= False:0
power_Battery_Is_Mimimal_One_Step()= False:0
inputs_State_button( 3 , State=False )
inputs_State_button( 5 , State=False )
inputs_State_button( 11 , State=False )
inputs_State_button( 12 , State=False )
inputs_State_button( 15 , State=False )
```

2. Connect J8-2 to J8-3. Read status inputs by sending the following command over the debug interface: "inputs m".
3. Response must include leading "0" "State=True" on the same output line. There may be other text in output.

```
inputs_State_button( 0 , State=True )
```

4. Disconnect J8-2 to J8-3.
5. Connect J8-5 to J8-4. Read status inputs by sending the following command over the debug interface: "inputs m".
6. Response must include leading "5" and "State=True" on the same output line. There may be other text in output.

```
inputs_State_button( 5 , State=True )
```

7. Disconnect J8-5 to J8-4.

8. Connect J8-7 to J8-6. Read status inputs by sending the following command over the debug interface: "inputs m".
9. Response must include leading "10" and "State=True" on the same output line. There may be other text in output.

inputs_State_button(10 , State=True)

10. Disconnect J8-7 to J8-6.
11. Connect J8-9 to J8-8. Read status inputs by sending the following command over the debug interface: "inputs m".
12. Response must include leading "15" and "State=True" on the same output line. There may be other text in output.

inputs_State_button(15 , State=True)

13. Disconnect J8-9 to J8-8.

Display Supply Test:

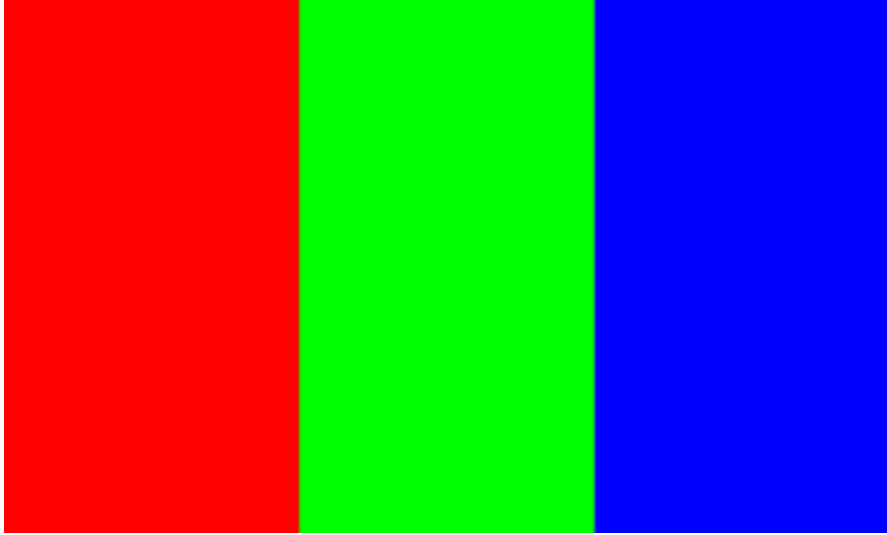
5. Measure J15 pin 1 WRT common with a DMM. Must measure $5.0V \pm 1V$.
6. Apply a 50Ω 1W load between pins 1 and 2 of connector J15
7. Measure pin 1 WRT common with a DMM. Must measure $3.27 \pm 0.5V$.
8. Remove load from connector J15

Potentiometer Check:

1. Apply 3.3V to pin 26 of connector J14
2. The voltage at pin 8 of U1301 shall be $0.77V \pm 0.1V$
3. The voltage at pin 1 of U1301 shall be $0.52V \pm 0.1V$

LCD Color:

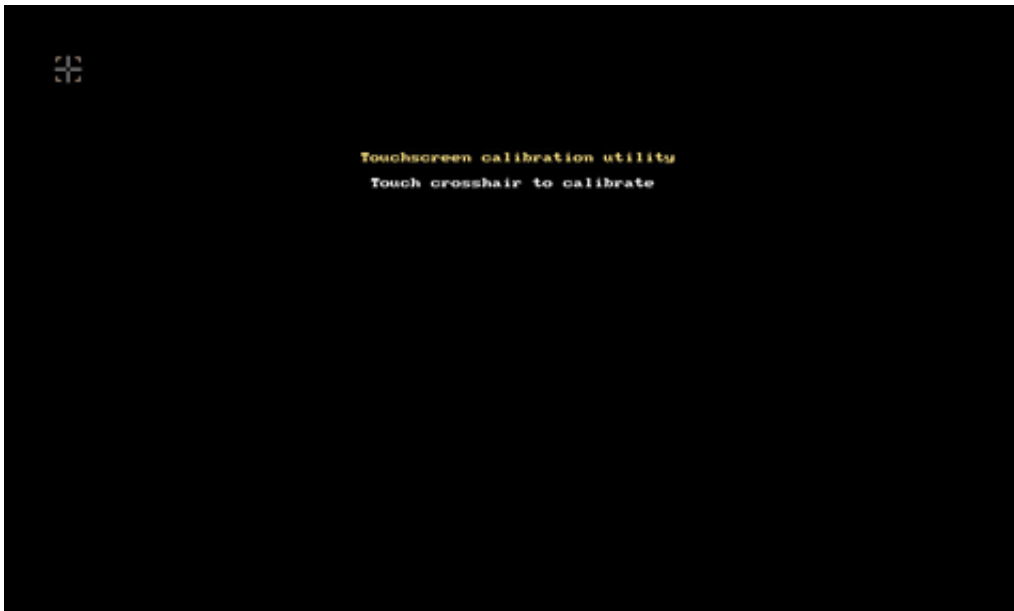
1. From the main menu, select the "Test LCD" menu item by typing the menu number and pressing return.
2. Confirm that the following pattern appears on the LCD:



3. LCD Color Check complete. Select the “Go back to Main Menu” menu item.

Touchscreen:

1. From the main menu, select the “Calibrate Touchscreen” menu item.
2. Follow the prompt on the LCD to calibrate the touchscreen:



3. Select the “Go back to Main Menu” menu item.
4. Select the “Test Touchscreen” menu item.
5. Follow the prompt on the LCD to test the touchscreen. Touch the touchscreen in several places and ensure that the cursor moves to that location. Touch “Quit” on the LCD to exit.



6. Touchscreen Check complete. Select the “Go back to Main Menu” menu item.

Backlight Driver:

If no LCD is included in the fixture, the previous two sections can be ignored. If it is included this section can be skipped.

1. Connect a 15 Ω 1.5W resistor between nets LCD_BL_P and LCD_BL_N
2. The voltage at LCD_BL_P with respect to GND shall be 4V +/- 10%
3. For reference, the max current of the backlight circuit is 269mA

Backup Camera:

1. Plug in backup camera to J6
2. Run command X

3. Visually verify LCD shows the camera image without aberration

CAN Bus:

1. Test the UI transmit capability. Read CAN message ID TBD via a computer. This is the UI heart-beat message and shows the operational state, the data field shall have a non-zero value.
2. Send the CAN message TBD with a data field of 0x 00 10 60 00 00 00 00 00 via a computer. Message interval is every 1 second. This sends the scrub motor current to the UI board.
3. Verify data sent to UI board via UART command

RS-232:

1. Connections: J18-1 is RS232 RXD, J18-2 is RS232 TXD
2. Run command to enable RS-232
3. Send data from computer to RS-232 port
4. The UI will echo back the data sent
5. Confirm that the data received at the computer matches what was sent

UART (5V):

1. Connections: J4-4 is 5V UART RXD, J4-5 is 5V UART TXD
2. Run command to enable UART
3. Send data from computer to UART port
4. The UI will echo back the data sent
5. Confirm that the data received at the computer matches what was sent

USB Host:

1. Connect a flash drive or generic storage device to J5
2. The storage device shall have the file X.txt on it
3. Run command X
4. The response must include "X.txt"

USB device:

1. Connect J7 to a computer
2. Run X command
3. The board shall enumerate as TBD