

# **EEVblog Electronics Community Forum**

A Free & Open Forum For Electronics Enthusiasts & Professionals

Search **Hello shawty** This topic Show unread posts since last visit. News: Show new replies to your posts. No news is good news. Be excellent to each other. November 20, 2022, 01:33:04 pm 🏠 Home 🔞 Help 🔍 Search 🧸 Profile About us 🖂 My Messages 📅 Calendar Links 🦽 Members 🔘 Logout EEVblog Electronics Community Forum » Electronics » Microcontrollers » Industrial PC with an Embedded TI M430F423 Microcontroller Used N9918B FieldFox. A KEYSIGHT SEE OFFERS Low price and in stock. PCB auote « previous next » Pages: [1] Go Down REPLY ADD POLL UNNOTIFY MARK UNREAD SEND THIS TOPIC Author Topic: Industrial PC with an Embedded TI M430F423 Microcontroller (Read 1829 times) shawty and 1 Guest are viewing this topic. Industrial PC with an Embedded TI shawty Modify Reply Quote Remove M430F423 Microcontroller Regular Contributor

Posts: 78 Country: <u>.</u> Q

« on: August 22, 2019, 08:39:07 pm »

I have 2 industrial, Box PC units here that I've been tasked with re-conditioning for a client.

These are the same 2 that I mentioned in my (ATX Power supply and increasing the Ampage thread: https://www.eevblog.com/forum/renewable-energy/atx-power-supplymodification/msg2557089/#msg2557089)

Now that I've sorted the PSU's out, I've been tasked with re-writing the software for them, the units are a one piece main board, with an Atom N270 CPU @ 1.6ghz, 1gb Ram an Intel ICH7 North/South bridge/I/O chip set on, with 5 Com ports, 2 GB Ethernet, 2 USB, DVI and single PS2 connector, collectively know as an "Arbor Arpex 1610 Box PC"

The main drivers, and OS are all ok, I've reconditioned them with a reasonably up to date version of "Windows Embedded 7" on one machine, and Debian 8 on the other (As per my clients request), both have 64ab SSD's in.

They also have a green I/O connector on one corner that has 4 Digital Outputs, 4 Digital Inputs and 2, 3 pole (+ & - with common ground and Vref) ADC's on.

The digital and Analog I/O connectors are connected internally to a Texas Instruments M430F423 Microcontroller embedded onto the main board alongside the regular PC chips mentioned above.

There is absolutely no technical info to be found on these devices anywhere, and the company that makes them "Arbor Technology" (Even though they have global offices) are HQ'ed in Taiwan, which if I'm honest, it's probably easier getting blood from a stone, than it is getting "Trade Secrets" "COUGH" "Technical Information" from a Taiwanese company... (No offense intended to anyone on here from that part of the world)

There are newer more modern machines available from the same company, that look like they have the same I/O features on them, but internally... well they could be anything... and even trying to at least get a sliver of more modern info to point me in the right direction has damn near impossible.

So my Question is essentially this:

Has anyone here ever had to work on anything like this in the past, and if you have do you still have any documentation, code samples or anything that might help me figure out how to program/talk to the thing.

Windows doesn't pick it up as a known device, and it doesn't do the yellow triangle thing saying it has

an unknown device, so it's not something detectable by a PnP OS. Windows does however, load 5 serial comports for the system, which is strange because physically it only has 4 serial comports externally. I've tried probing the com port at different speeds (All 5 of them) and got nothing that may lead me to believe there's anything there.

Using a user guide from a different manufacturer that had the same "SuperIO" chipset in as the machines I'm working with, and which had "code samples" to talk to the GPIO (Which was supposedly attached to it's I/O system) yielded no results when I tried the code samples from that, and only served to confirm I had the same SIO (or SIO compatible clone) in the machine, but nothing was observed on the I/O lines.

The linux machine does load an SMBus/I2C driver, which loads up 7 I2C bus devices, of those 7 only 3 have any device addresses responding in them as follows

Bus 0 - 0x08, 0x44, 0x4C, 0x50, 0x69

Bus 5 - 0x38

Bus 7 - 0x37, 0x38, 0x49, 0x50, 0x59

Probing these I2C devices in software while toggling bits on the I/O connectors again has yielded no results.

I know 100% that the I/O connectors are connected to the TI MCU, as I used an arduino Uno as a square wave source attached to the I/O pins, then probed around the chip with with a logic probe until I found my signal, so it seems to me, that the missing piece of my puzzle is just how the hell the host system talks to the embedded microcontroller.

If anyone has any pointers, experience or otherwise that might help, please share your insights.

Iv'e attached the only bit of tech info (The user guide) that I've been able to get a hold of so far [I dragged that of a Chinese file sharing site using a fake account :-) ]

The actual microcontroller is this one: www.ti.com/lit/ds/symlink/msp430f427.pdf an older but still current product in the TI range.

Cheers shawty.

PS: I've tried as many of the tools as I could get to run on the box from the TI site. Code composer won't do COM/Uart or if it does I've no idea how, and none of the older "demo" or "uart" tools recognize there is anything there, my guess is any programming software for this thing would have been current circa 2009/2010 when these devices where in circulation.



Arpex-1610\_Manual\_V11\_20091008.pdf (792.38 kB - downloaded 157 times.)

Report to moderator \$\begin{align\*} \text{88.97.38.99} \end{align\*}

Meh....

Re: Industrial PC with an Embedded TI M430F423 Microcontroller « Reply #1 on: August 22, 2019, 08:44:32 pm »

Reply

Quote Modify Remove

PPS: I may have put this in the wrong forum/thread set, if I have then could a moderator please move it to the appropriate section.



Quote

Report to moderator \$\mathbb{B}\tag{88.97.38.99}

Say Thanks

Meh....

□ andersm

Super Contributor

Posts: 1198 Country:

2. Q

shawty

Posts: 78

<u>.</u> Q

Country:

Regular Contributor

Re: Industrial PC with an Embedded TI M430F423 Microcontroller

« Reply #2 on: August 23, 2019, 06:54:11 am »

Have you checked if any of the found I2C addresses line up with any of the better known GPIO expander chips? Making an MCU emulate a bunch of existing ICs is one way of avoiding the effort of making custom device drivers. The Linux device tree documentation will provide a list of supported devices.

Report to moderator Logged

TNKernel-PIC32, an open-source real-time kernel for the PIC32

MosherIV Super Contributor

Re: Industrial PC with an Embedded TI M430F423 Microcontroller

« Reply #3 on: August 23, 2019, 07:13:57 am »

Say Thanks

Reply

Reply

Quote



Country:

₽.Q

Just because Linux has identified I2C does not means it is actually I2C device



The digital and Analog I/O connectors are connected internally to a Texas Instruments M430F423 Microcontroller embedded onto the main board

This is a microcontroller, mixed signal if you believe the marketing hype.

That means it has its own program inside. Only the original deveopers will know how they have connected it to the rest of the PC and how to drive it.

Quote

The main drivers, and OS are all ok, I've reconditioned them with a reasonably up to date version of "Windows Embedded 7" on one machine, and Debian 8 on the other (As per my clients request),

Hope you saved the old disks or made an image of them.

Without programming information about the IO device

You will need to use the original disks/image and probe/scooe around to see how it is being controlled. In other words - reverse engineer it

Report to moderator Logged



shawty Regular Contributor

Posts: 78 Country:

<u>.</u> 9

Re: Industrial PC with an Embedded TI M430F423 Microcontroller

« Reply #4 on: August 23, 2019, 10:05:54 am »

Reply

Modify

Remove

Quote from: andersm on August 23, 2019, 06:54:11 am

Have you checked if any of the found I2C addresses line up with any of the better known GPIO expander chips? Making an MCU emulate a bunch of existing ICs is one way of avoiding the effort of making custom device drivers. The Linux device tree documentation will provide a list of supported devices.

I've done what I can due diligence wise and checked against various lists I can find that are publicly available such as this:

https://learn.adafruit.com/i2c-addresses/the-list

https://i2cdevices.org/addresses

and this:

https://os.mbed.com/users/oliverb/notebook/i2c-address-list/

And while there is some matches in device numbers seen vs Numbers in the list, I'm not entirely sure that I have a match. For example 0x50 is commonly seen on a TDA based video signal driver, and I know from an examination of the actual circuit board that there's no TDAxxxx on there. It is possible that it's an emulation embedded in the north bridge of the Intel ICH7 chip set (Just as the SuperIO chip is) but looking at the data sheet for the TDAxxxx and trying some reg access based on it did't produce anything that confirmed (or denied) otherwise.

Like wise 0x69 is often seen on RTC chips (I have a few in my parts boxes already) but again probing reading/writing with the I2C tool set didn't really help.

If you know of any other lists I may be able to cross reference, I'd be interested in seeing the links.



Report to moderator \$\begin{align\*} \text{88.97.38.99} \end{align\*}

Meh....

Re: Industrial PC with an Embedded TI M430F423 Microcontroller

Reply

Quote

Modify Remove

« Reply #5 on: August 23, 2019, 10:22:54 am »

Quote from: MosherIV on August 23, 2019, 07:13:57 am



Just because Linux has identified I2C does not means it is actually I2C device

Quote

The digital and Analog I/O connectors are connected internally to a Texas Instruments M430F423 Microcontroller embedded onto the main board

This is a microcontroller, mixed signal if you believe the marketing hype.

That means it has its own program inside. Only the original deveopers will know how they have connected it to the rest of the PC and how to drive it.

Quote

The main drivers, and OS are all ok, I've reconditioned them with a reasonably up to date version of "Windows Embedded 7" on one machine, and Debian 8 on the other (As per my clients request),



Posts: 78 Country: 

Hope you saved the old disks or made an image of them.
Without programming information about the IO device
You will need to use the original disks/image and probe/scooe around to see how it is being controlled.
In other words - reverse engineer it

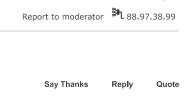
Yea, this is why I \*think\* that the way in is via COM5, it would make sense to me that there would be some firmware running on the thing, that responded to it's Uart and accepted various commands, returning various information. It seems odd that there is this 5th com port that doesn't have a physical connector. There is a set of holes on the main board which is about the right size for an IDC style serial socket, but there is no connector or pins soldered into it, so why would you put a phantom serial port on a board and enable it in the bios unless it was connected somewhere.

Disks are just not an option right now, I am exploring an avenue that MIGHT yield a result with regards to that, but as of yet I've no feedback, so as of this post that's currently a dead end.

As I say, I've tried all the various programming software that I can get to run on a 32 bit embedded machine with 1gb of ram (Hint: Not a great deal :-) ) and nothing has even given me the slightest bit reason to think there is an MSP device there that it can talk to, and I've gone back to version 4 of code composer studio too, which was released at about the same time as these units where.

My client is small family run woodworking/furniture business, so they really don't want to replace these units unless they have absolutely no other choice, and I can't say I blame them, I've seen the cheapest quote they've been given and it made even my eye's water!! the words daylight and robbery spring quickly to mind, I do have the option of possibly building a new device using an Arduino/rPI something along those lines, but this is a very hostile environment for electronics, lots of vibration, sawdust and other nasty stuff, which these units in their sealed armored case are designed to withstand.

I've tried getting some probes on the Uart pins on the microcontroller, but it's a very small device, with a stupidly narrow pin pitch done using SMD, and even the smallest of chip hooks and probes I have are too big to be sure I'm getting a good connection, I think my next step is going to be to hack together a quick program in C or C# that opens the port at all different speeds automatically and just throws bytes at the ports to see if I get anything that way, and hopefully I'll see if I can hold a probe or pin in place by hand while I do, see if I see anything on my logic probe.



٩





Re: Industrial PC with an Embedded TI M430F423 Microcontroller

« **Reply #6 on:** August 23, 2019, 10:35:50 am »

Meh....

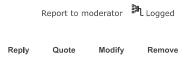
Quote from: shawty on August 23, 2019, 10:22:54 am

As I say, I've tried all the various programming software that I can get to run on a 32 bit embedded machine with 1gb of ram (Hint: Not a great deal :-) ) and nothing has even given me the slightest bit reason to think there is an MSP device there that it can talk to, and I've gone back to version 4 of code composer studio too, which was released at about the same time as these units where.

Obviously development tools won't work - because programmer is needed.

http://www.ti.com/tool/MSP-FET

You may use built-in debugger of LaunchPad (of series) as cheaper debug adapter. Honestly idea of "hacking" that thing sounds very wrong to me unless you work for free.







Re: Industrial PC with an Embedded TI
M430F423 Microcontroller

« **Reply #7 on:** August 23, 2019, 12:31:49 pm »

The chip is soldered onto the main board, in a very tight and cramped space, getting it out or finding a way of attaching it to the launchpad is more or less impossible.

I was looking at the launchpad devices in my quest to find out more about the device, and concluded from looking at some home made devices, that it was in actual fact possible to talk to one and program it, over it's Uart interface.

Looking at the specs and the various hobbyist pages I read, you have to do some gymnastics with the baud rate, such as changing it rapidly from 9600 baud to 115200 and then back again, the standard boot loader installed in the factory on all these processors apparently recognizes the timing on these baud changes as commands to switch different modes of operation.

I however suspect, that it's more likley the MCU has some firmware running on it, listening to the Uart port that waits for specific sequences to tell it what to do, hence why I'm asking for anyone that may have any exposure to these particular PC's (or similar devices) that may have in the past had to write code to interface them to the digital I/O.

I'm not working for free by the way, this is a paying client whom I'm trying to solve a problem for, I'm not fleecing them with extortionate costs either however, I'd be lying however if I said that my only motivation here was the money and the job, this is actually a very, very interesting project for me, which if I'm honest I'm rather enjoying, even if it is driving me insane. :-)



Report to moderator \$\mathbb{B}\mathbb{L} 88.97.38.99

Meh....

Super Contributor

Re: Industrial PC with an Embedded TI M430F423 Microcontroller

Sav Thanks

Reply

Quote

Posts: 3709

Country:

 $\mathbb{Q}$ 

□ ogden

Quote from: shawty on August 23, 2019, 12:31:49 pm

« Reply #8 on: August 23, 2019, 12:48:55 pm »

I however suspect, that it's more likley the MCU has some firmware running on it, listening to the Uart port that waits for specific sequences to tell it what to do, hence why I'm asking for anyone that may have any exposure to these particular PC's (or similar devices) that may have in the past had to write code to interface them to the digital I/O.

Rest assured that you will not find any information about contents of that service MCU, nor success accessing it with development tools. Manufacturer of such products usually do not make any errors while protecting their intellectual property.

Quote

I'm not working for free by the way, this is a paying client whom I'm trying to solve a problem for, I'm not fleecing them with extortionate costs either however, I'd be lying however if I said that my only motivation here was the money and the job, this is actually a very, very interesting project for me, which if I'm honest I'm rather enjoying, even if it is driving me insane,

If you need those I/O's running - then get documentation either for existing hardware or buy new hardware with documentation. Customer who bought those computers could try to request information through seller. By information I mean user and programmer's manuals, not source code of the product.

Report to moderator Logged

coromonadalix

Super Contributor



Posts: 3943 Country: & Q

Re: Industrial PC with an Embedded TI M430F423 Microcontroller

« Reply #9 on: August 23, 2019, 02:32:10 pm »

Say Thanks

Reply

Quote

Dont know if they still active :

Media Contact: marcom@arbor.com.tw Inquiry Contact: info@arbor.com.tw

http://www.arborchina.com/products/industrial\_automation/detail\_id\_169.html

some downloads links and or falses links (A)



« Last Edit: August 23, 2019, 02:34:57 pm by coromonadalix »

Report to moderator Logged

Modify

Quote

Reply

Remove

shawtv

Regular Contributor

Posts: 78 Country:

요. 🦁

Re: Industrial PC with an Embedded TI M430F423 Microcontroller

« Reply #10 on: August 23, 2019, 02:52:18 pm »

Quote from: coromonadalix on August 23, 2019, 02:32:10 pm

Dont know if they still active :

Media Contact: marcom@arbor.com.tw Inquiry Contact : info@arbor.com.tw

 $\underline{\text{http://www,arborchina.com/products/industrial\_automation/detail\_id\_169.html}}$ 

some downloads links and or falses links



Thanks I'll follow up on them.

The actual product link I already exhausted, the links just don't go anywhere. I tried the info email already too, got one reply uttering some rubbish about me not being a partner so they couldn't talk to me, and then no reply on anything after that.

I'll try the other email though, that's a new one.

Cheers Shawty



Meh....

■ MosherIV Super Contributor



Posts: 1530 Country:  $\mathbb{Q}$ 

Re: Industrial PC with an Embedded TI M430F423 Microcontroller

« Reply #11 on: August 23, 2019, 06:20:49 pm »

Say Thanks

Reply Quote

Ouote

Disks are just not an option right now.

You have not understood.

Ouote

I have 2 industrial, Box PC units here that I've been tasked with re-conditioning for a client.

So, the original industrial PCs had something on them.

In order to work out what to talk to the M430F423 Microcontroller, you start by seeing what it did

That was what I was trying to tell you about saving the original disk.

You may have even been able to copy drivers from the original disk.

I would not trust Windows or Linux is telling you, they scan memory addresses on the io channel and see what responds. They ASSUME that something at particular addresses are standard devices hence the COMM5 or i2c.

Have you tried probing with a scope the holes for the IDC connector, do this during boot up « Last Edit: August 23, 2019, 06:22:43 pm by MosherIV » Report to moderator Logged

Re: Industrial PC with an Embedded TI M430F423 Microcontroller

Reply

Quote Modify Remove

« Reply #12 on: August 25, 2019, 12:39:08 pm »

Quote from: MosherIV on August 23, 2019, 06:20:49 pm

Quote

Disks are just not an option right now,

You have not understood.

Sorry, but yes I have. There are no disks, I'm never going to get any disks, and if there are any around anywhere the folks that have them are not making it known they have them.

Anyway... small update:

I can now confirm that the mysterious internal com port (com5) is definitely connected to the MSP430, and the MSP430F4xxx have something burned into them that cannot be removed called the "BSL Uart Bootloader", however after reading this: http://www.ti.com/lit/ug/slau319m/slau319m.pdf several times, and trying the "BSLDEMO" program that is supposed to work across a standard UART/Com connection, I'm just getting "are you sure it's connected" error.

I've even used a modified version that allows me to invert the RTS/DTR lines (Apparently there was a bug in TI's original version of the tool) but trying all combinations, still isn't getting me anywhere, so either the firmware has been blanked, or it's totally custom.

I'm going to probe the RTS/DTR lines on the device see if I see any changes on them, and ensure they are connected to the lines the PDF states they should be connected too, RX/TX etc however, when I started firing data at the com port the data I was sending was visible on the UART lines on the MCU, so I do at least know that TX/RX are active.

Shawty



Report to moderator \$\mathbb{\mathbb{H}}\$ 88.97.38.99

Meh....



shawtv





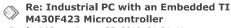
Posts: 1530 Country: 2. Q

shawty

Regular Contributor

Posts: 78 Country:

<u>.</u> Ģ



« Reply #13 on: August 25, 2019, 01:31:44 pm »

So, when you got the 2 industrial PCs, were they working? Did they have the original application working or not?

As someone else mentioned, the company who built the original system woud have taken measures to protect their proprietry work, ie protected the contents of the MSP430 from being read out.

You have been trying to use the default TI built in boot loader.

I suspect the original company would have taken measures to stop the TI BSL boot loader from being able to read out their binary.

Having triggered the TI BSL bootloader, you may have wiped the contents of the MSP430.

In addition, they may have their own custom bootloader.

Report to moderator Logged

### Re: Industrial PC with an Embedded TI M430F423 Microcontroller

« Reply #14 on: August 25, 2019, 04:35:11 pm »

Reply Quote Modify Remove

Say Thanks

Reply

Quote

Quote from: MosherIV on August 25, 2019, 01:31:44 pm

So, when you got the 2 industrial PCs, were they working? Did they have the original application working or not?

As someone else mentioned, the company who built the original system woud have taken measures to protect their proprietry work, ie protected the contents of the MSP430 from being read out.

You have been trying to use the default TI built in boot loader.

I suspect the original company would have taken measures to stop the TI BSL boot loader from being able to read out their binary

Having triggered the TI BSL bootloader, you may have wiped the contents of the MSP430.

In addition, they may have their own custom bootloader,

Everything to do with what I've tried so far, apart from additions to this thread are all in the first post.

My client has had these for a considerable amount of time, they are a small family run company, who can't afford the bill for newer models, so I either need to re-condition these things, or build them something new.

I'm aware of the if/when's/where fore's regarding these machines, and to be fair I'm not expecting miracles, which is why I'm asking for folks that might have experience working with them in the past, and might be able to suggest something I've not yet tried.

These are quite old machines, and have worked flawlessly for my client for a very long time, but while I'm re-conditioning them, they want me to make some changes to their software, and as I said previously the only thing I've not been able to do is to get the GPIO stuff working.

I may have to admit defeat and just build something new, using something like a PI and an Arduino, or maybe even tap onto the I2C bus in these units and hook up a PIC or something.

The machines where designed for mass market, the DI/O(GPIO) on them was also designed to be easily interfaced to from windows embedded based code from what I've read in the old sales literature, so I can't see the firmware on them being very specialized, on a machine to machine basis.

Yes, there is a chance these MCU's have been wiped, but according to the doc's I've been able to get from TI, the BSL Bootloader is apparently permenently burned into ROM on the device and cannot be erased.

Right now, I suspect one of 2 things:

- 1) The M430 has a standard BSL bootloader on it, but no other code, so it's either unresponsive, or the chip is possibly damaged.
- 2) The M430 has a custom bootloader in place of the BSL, and only arbor (The manufacturers of the unit) know the secret behind how it works.



Report to moderator 88,97,38,99

Quote

Meh....





I should probably title this post

W00t W00t W00t or WooooooooHooooooooo :-)

and by that opening sentance you can probably figure out that I've cracked it.

It's taken me about 3 weeks, but I've broken in, I've figure out the protocol on the MSP430F423 and I now have dotnet code that interface that can read the digital inputs on the device, and write the digital outputs.

Essentially my hunch about the MSP430 being connected to the COM5 port inside the PC itself was correct, the bit I got wrong was that it wasn't connected up in such a way as I can activate the bootloader.

The MSP430F423 does have the standard non-erasable uart bootloader installed on the chip, but you have to do some very, very precise timing using the RST and INT pins, which on a development board are connected to the RTS & CTS lines so that the host software can use the 2 lines as control lines to enter the bootloader.

On the industrial box PC's I have, the internal COM5 RTS/CTS are not connected up. There is however an internal serial pin header (well the holes anyway) on the motherboard, and tracing the RTS/CTS etc from that back to the MCU shows that the RTS/CTS on that connector IS connected up correctly, so I'm pretty sure that if I where to solder a pin header into it then connect it up using an external uart such as one of the FTDI's I have lying around, I would be able to read and/or re-program it. That's for another day though :-)

For now, the protocol once I figured it out was quite simple.

Command packets are exactly always 32 bytes in length, as are response packets.

To get the current firmware running on the device to pay attention to you, the first 2 bytes of that packer must always be

0x7B followed by 0x23

and the last 2 bytes the opposite

0x23 followed by 0x7D

in ascii it looks something like {# #}

Once you have 32 bytes of 0x00 with those markers on either end, you then need to put what I think is a command type byte in at offset 2 and an actual command byte at offset 3

From what I've worked out so far, offset 2 = 1 means command is a read command and = 2 means it's a write command.

The actual command byte at offset 3 is always a negative number, so it's a signed byte in 2's compliment.

```
-126 (0x82) = Digital I/O
```

-124 (0x84) = Analog (ADC) 1

-125 (0x83) = Analog (ADC) 2

-127 (0x81) = Sign on/Authenticate/Init or something like that (I'll come back to this one in a moment)

So to read the digital I/O inputs for example, you send

```
0x7B, 0x23, 0x01, 0x82 (26 * 0x00), 0x23, 0x7D
```

The MSP will then respond with a more or less mirror image of the packet, exactly 32 bytes, same beginning and end sequence, but in the 3rd byte where I put the 0x01, that will be changed to a 0x03, then the next 2 bytes following the 0x82 will be 0x01 and 0xX0

The capital X will be 1 for Input 0, 2 for input 1, 3 for input 2 and 4 for input 3, or combinations thereof so for example if I get back 0xF0 then all for inputs have a signal on them.

The inputs are active low, so 0x00 means all four are high, 0xF0 means all 4 are low

Depending on what your doing the 0x01 changing to a 0x03 can sometimes get changed to a 0x04 especially on the sign in command.

The MSP430 firmware WILL NOT respond to anything before you first send it what I'm calling an init/sign on packet.

0x7B, 0x23, 0x02, 0x81, 0x05, 0x41, 0x52, 0x42, 0x4F, 0x52 (20 \* 0x00) , 0x23, 0x7D

The usual start and end bytes are there, and the bytes at 2 & 3 are 0x02/0x81 which is a write

command of -127, then there is a 0x05 (I assume is length) followed by 5 bytes that spell:

#### ARBOR

The name of the manufacturer.

The response comes back with the 0x02/0x81 changed to 0x04/0x81 and the 0x05 prior to the 5 extra bytes changed to 0x01 if it was successful, if you change any of the bytes in 'ARBOR' to anything else the 0x05 comes back changed to 0xFF and the MSP430 refuses to listen to further commands accept 2,-127

Until you send that INIT packet, with the correct passphrase of ARBOR in, the MSP will not acknowledge or listen to any other command sent to it it just stays silent, I've not tried sending the init then sending command, disconnecting and sending command again or anything like that, but I suspect that when the COM connection is severed it resets itself, I still have a little more testing to do here, I'm also going to try some other 1,-xxx combinations see if anything else pops up.

I'm going to put the code I've written in C# (Minus my clients stuff) [I have my clients permission to release the comm library] up on my Github page at http://github.com/shawty/ so if anyone comes across these (or similar) devices in future the answers will be there, I'll also document the protocol better.

But yea, that's it. What I was essentially doing wrong was sending single bytes to it to see if it responded, but it only responds when the 32 bytes in one go are sent, and whats more any delay of longer than about 10ms between data bytes and it assumes an incomplete packet and ignores you, so you cant actually get any use out of it by sending one byte at a time, you essentially have to build the 32 byte buffer in memory, then throw the entire thing at the device in one go oh and it ONLY operates at 9600 baud 8N1 anything else and again it just goes silent.

Phew.... Shawty

Report to moderator 88.97.38.99

Meh....

The following users thanked this post: ogden

Re: Industrial PC with an Embedded TI M430F423 Microcontroller

Say Thanks

Reply

Quote

« Reply #16 on: August 31, 2019, 02:09:55 pm »

Quote from: shawty on August 31, 2019, 12:23:58 pm

The MSP430F423 does have the standard non-erasable uart bootloader installed on the chip, but you have to do some very, very precise timing using the RST and INT pins, which on a development board are connected to the RTS & CTS lines so that the host software can use the 2 lines as control lines to enter the bootloader.

On the industrial box PC's I have, the internal COM5 RTS/CTS are not connected up. There is however an internal serial pin header (well the holes anyway) on the motherboard, and tracing the RTS/CTS etc from that back to the MCU shows that the RTS/CTS on that connector IS connected up correctly, so I'm pretty sure that if I where to solder a pin header into it then connect it up using an external uart such as one of the FTDI's I have lying around, I would be able to read and/or re-program it. That's for another day though :-)

Establishing JTAG connection with MSP430 device is safe. If JTAG fuse is blown, only BYPASS command is enabled (and will return JTAG ID), other commands are disabled.

Depending on UART BSL version (device age), using it is not safe, because wrong BSL password (int vector adr space) will trigger mass erase, and device firmware will be lost. This is not case with MSP430F1xx, but (I guess) it is with most of F2xx/4xx devices. There is no way to protect device form entering to UART ROM BSL mode, even there is custom flash updater.

Report to moderator Logged

Withdraw Thanks

Quote

Reply

■ MosherIV Super Contributor

□ josip

Posts: 98

Country: 2. Q

Regular Contributor



Country: 2. Q

Nice one Shawty.

Thanks for sharing.

Report to moderator Logged

The following users thanked this post: shawty

M430F423 Microcontroller

« Reply #17 on: August 31, 2019, 02:24:43 pm »



Re: Industrial PC with an Embedded TI

Quote

Regular Contributor



Posts: 78
Country:

« Reply #18 on: September 02, 2019, 07:34:36 pm »

For those who might be interested, C# source code to make use of the IO system in the Arbor devices can now be found here:

https://github.com/shawty/arbor\_arpex\_1610



Quote

Report to moderator \$\mathbb{B}\mathbb{L} 88.97.38.99

Say Thanks

----

Meh....

## □ coromonadalix

Super Contributor



Posts: 3943 Country: 🗐

☐ dans34

Contributor
Posts: 7



Re: Industrial PC with an Embedded TI M430F423 Microcontroller

« Reply #19 on: September 03, 2019, 07:29:11 am »

Impressive work shawty

Report to moderator Logged

# Re: Industrial PC with an Embedded TI M430F423 Microcontroller

« Reply #20 on: August 17, 2022, 11:09:58 am »

Say Thanks

Reply

Reply

Quote

Just picked one of these up from ebay , specs wise are not great

Intel Atom N270 @ 1.6ghz , 1gb of ddr2 & a 64 gb ssd ,  $4 \times serial$  , two are rs232 and the other two can be configured as RS232/422/485 , interestingly also has a CF card slot hidden inside the case (well not hidden but not accessible from the outside)

Thinking of trying to power it from USB C with a PD board from ebay

Report to moderator

Modify

Logged

Remove

shawty

Regular Contributor

Posts: 78 Country:



Re: Industrial PC with an Embedded TI M430F423 Microcontroller

« **Reply #21 on:** August 17, 2022, 12:50:54 pm »

system for my garden gate :-)

Yea I have 2 of them, one I've installed with PFSense and use as a router because of it's 2 ethernet ports and the other I have installed in my garden shed and use it to control my electronic locking

Quote

Reply

When I originally was working with these things, I was doing so on behalf of a client, who has since upgraded to newer hardware, so I was given the 2 older units since I was the only person who managed to do anything with them.

I would discourage trying to power via a USB-C board, as they DO suck down a lot of amps, I've got mine connected to an LED signage power supply that delivers 5A at 12V (It's badged TS-60-12), I chose that one because these units are very unstable if you don't hook up the minus 12v feed.

In fact, as I've found out most Linuxes won't even boot properly with out the -12v line attached.

From what I can ascertain going inside the units, the -12 is only used for the com ports, but without it you'll have nothing but pain, and the -12v can and often does draw up to 3 amps.

The other problem I had was getting the green connectors needed for the power. The ONLY place I could find them for sale in the UK was RS components and they cost me about £5 GBP each.

Once you get powered up though, they ain't bad little units to be fair.

One of the units I inherited was originally used as a signal booster for an industrial saw mill, it sat in the middle of two 25 meter serial cables (Well 3 actually) one from the unit to the machine, and one to each of the 2 control booths.

I've had Debian running on them, right up to about V10, and I've also had Windows 7 running on them, as well as Windows POS Ready on a project I had to do to build a point of sales system for a client.

The ADC on them is 1 channel differential, so you need to be careful with that, it'll catch you out as it won't give you a full 0 to max range, you'll get positive off of one side, and negative readings off the other, but all within a positive integer range :-)

The digital inputs can handle up to about 5.5v @ approx 200 Milli amp, and the I/O is separate, you have 4 out and 4 in, but you can't change function on them, in will always be in, out will always be

out.

What I've done with mine to compliment the on board I/O is to add an FT232H serial I/O module, I have some C# somewhere to control the thing (There's plenty of docs) but you can plug the 232H directly into a USB port and feed it I2C (or just regular serial UART) directly, and drive I2C/2 wire/SPI/Uart devices directly off of the 5V on them (I use it for connecting to an I2C RFID card scanner on my back gate), the libs and SDK docs that come with the 232 also allow you to use any of the 20 pins on the board as standard 5v TTL digital I/O just like on an arduino or similar.

My code to drive the built in I/O stuff is available here:

https://github.com/shawty/arbor\_arpex\_1610

If your interested in programming the FTD232H serial engine (to use with the unit) I did a you-tube presentation for "Notts IoT" in Febuary 2021 explaining how to get started with one which you can find here:



Happy to help where I can, just ping me :-)

Shawty

Report to moderator \$\frac{\mathbb{H}}{2}\$ 88.97.38.99

Meh....

Pages: [1] Go Up

REPLY

ADD POLL

UNNOTIFY

MARK UNREAD

SEND THIS TOPIC

PRINT SEARCH

« previous next »

Share me



EEVblog Electronics Community Forum » Electronics » Microcontrollers » Industrial PC with an Embedded TI M430F423 Microcontroller

MOVE TOPIC LINK TO CALENDAR

Jump to: ☐=> Microcontrollers ✔ ☐ go









ANENG, UNI-T and more...



SMF 2.0.18 | SMF © 2021, Simple Machines Simple Audio Video Embedder SMFAds for Free Forums XHTML RSS Mobile Mobile WAP2