SIMtrace 2 - Feature #1706

Hardware / Circuit board update for SAM3S based SIMtrace2

05/09/2016 07:48 PM - laforge

Status:	In Progress	**Start date:**	05/09/2016
Priority:	Normal	**Due date:**	
Assignee:	mschramm	**% Done:**	60%
Category:	hardware		
Target version:	PCB v2		

Description

We have a few boards using SAM3S instead of the old SAM7S, and just changing the chip + firmware while keeping the rest of the hardwrare seems fine.

However, we might want to add some other features to the board, while we are doing the switch to SAM3S.

Subtasks:

Feature # 1457: Spacing between JTAG and JP1/JP2	**New**
Feature # 1911: check if we can have an enclosed version of simtrace v2	**New**
Feature # 1912: do proper re-layout of the board	**Stalled**

Related issues:

Related to SIMtrace 2 - Feature #1463: Add VCC current sensing circuit for SP...	**New**

History

#1 - 05/09/2016 07:48 PM - laforge

- Assignee deleted (tsaitgaist)

#2 - 01/12/2017 12:53 PM - laforge

- Target version set to SIMtrace_v2

#3 - 01/12/2017 01:04 PM - laforge

- Checklist item [ ] consider switching from JTAG to SWD header added
Checklist item [ ] remove 6pin DEBUG connector? added
Checklist item [ ] remove RESET button? added
Checklist item [ ] add button for ERASE to SAM-BA added
Checklist item [ ] remove DC input, USB jack can be used for stand-alone power, too added
Checklist item [ ] have real SPDT switch rather than SPST added
Checklist item [ ] design with EAGLE to benefit from sysmocom parts library / processes added

#4 - 05/11/2018 04:30 PM - laforge

- Project changed from SIMtrace to SIMtrace 2

- Subject changed from Finalize SAM3S based SIMtrace2 to Hardware / Circuit board update for SAM3S based SIMtrace2

- Category deleted (SIMtrace hardware)

- Target version deleted (SIMtrace_v2)

#5 - 06/01/2018 01:13 PM - tsaitgaist

- Checklist item [ ] add pull-up resistors on SC_SW and I/O_SW to set default state of bus switch to high impedance added

#6 - 07/31/2018 02:32 PM - tsaitgaist

- Assignee set to tsaitgaist

#7 - 08/09/2018 12:14 PM - laforge

- Category set to hardware

#8 - 08/09/2018 12:17 PM - laforge

- Checklist item [ ] ESD protection (TVS diode arrays) on FPC, UART, USB added
Checklist item [ ] external USB pull-up circuitry with two transistors, shottyk diode can be removed (SAM3 and newer don't need it) added
Checklist item [ ] check for [soon] obsolete parts and replace with active ones added

#9 - 08/09/2018 12:19 PM - laforge

- Assignee changed from tsaitgaist to sysmocom hardware team

#10 - 08/24/2018 12:43 PM - laforge

- Assignee changed from sysmocom hardware team to mschramm

#11 - 08/25/2018 03:58 PM - laforge

- Checklist item [ ] add RTC crystal (and cap)? for time-stamping readings/recordings added
Checklist item [ ] replace SPI flash with uSD card (for more storage/recordings)? or larger SPI? added

Adding some more ideas. Replacing the SPI flash with uSD (attached to HSMCI or SPI like now) would mean that recorded traces could simply be transferred to a computer without the use of USB.

#12 - 08/27/2018 07:05 AM - laforge

- Target version set to PCB v2

#13 - 08/28/2018 01:21 PM - mschramm

- Related to Feature #1463: Add VCC current sensing circuit for SPA & DPA attacks added

#14 - 08/28/2018 05:01 PM - mschramm

- File cortex_debug_connector.gif added

- Status changed from New to In Progress

OK, as I am about to start this, some remarks / questions:

- the planned ATSAM3S4B is not "recomm'd for new designs" according to Microchip. However, there is no EOL PCN for these parts. Kevin and me were thinking of using the ATSAM4S4B which is a pin-compatible "in production" Cortex-M4, this also is cheaper than the SAM3S.
- Sylvain's suggestion #1463 (add VCC current sensing circuit/resistor) should be inserted in v2, too (prelim default to 0R). Preferebly with Kelvin contacts! ;)
- removal of 6pin DEBUG connector: did no one need the FTDI adapter pinout? In that case, I'll simply remove it.
- consider switching from JTAG to SWD header: we can have both - what about a Cortex debug connector (referring the pinout (see image); no need to go to their 0.05” pitch) ?
- have real SPDT switch rather than SPST: I'll again use the TS3A5018 for this.
- RTC Xtal: pins XIN32 and XOUT32 were designated for other GPIO functionality, but will now be freed for slow Xtal. - Regarding the backup cap: do you think of a (small) supercap? Then: what timespan do you need the backup domain and RTC remain valid?
- SPI flash: in the long run not used for logging, only used for version detection - let's find a proper µSD card holder, which is available and has a reasonable part's footprint.
- version detection: we should introduce a 'version detection' voltage divider on ADx.
- two level translator for both USARTs seem to be desirable: their VccA is 3V3 for the ARM, VccB is the SIM voltage actually seen from the phone/modem. This would at least be needed if another idea is still on the list:

- rigth now, the ARM can decide which power source the SIM should see (3V3 or Vcc_PHONE). Wasn't there that idea to have the ability to change between 1V8, 3V3 and 5V for the SIM when it is not powered by the modem/phone?

#15 - 08/28/2018 09:46 PM - laforge

On Tue, Aug 28, 2018 at 05:01:02PM +0000, mschramm [REDMINE] wrote:

> - the planned ATSAM3S4B is not "recomm'd for new designs" according to Microchip. However, there is no EOL PCN for these parts. Kevin and me were thinking of using the ATSAM4S4B which is a pin-compatible "in production" Cortex-M4, this also is cheaper than the SAM3S.

yes, we do have even one or two re-worked SIMtrace2 boards with the
SAM4S, I think one of it is with Kevin. It should not only be
pin-compatible, but also instruction-set, memory-layout and peripheral
compatible. So unless they screwed up something regarding clock or
other low-level initialization, I would expect the same binary to even
work.

> - Sylvain's suggestion #1463 (add VCC current sensing circuit/resistor) should be inserted in v2, too (prelim default to 0R). Preferebly with Kelvin contacts! ;)

sure.

> - removal of 6pin DEBUG connector: did no one need the FTDI adapter pinout? In that case, I'll simply remove it.

no, we don't need it. OsmocomBB-style 2.5mm jack is standard by now,
and it's not too much to ask people to have such a cable if they like
console UART access

> - consider switching from JTAG to SWD header: we can have both - what about a Cortex debug connector (referring the pinout (see image); no need to got to their 0.05" pitch) ?

The "problem" with ARM-JTAG-20 is that it's **huge**. Most people will
never use it, at programming is done via DFU, and in emergency cases via
SAM-BA. So JTAG is only used by those few that ever hack on the code,
and I think given the lack of contributions over 7 years it's clear
there are only very few people outside the core Osmocom + sysmocom team
who will take advantage of that...

I don't like the SWD connector as it's flimsy and has a small pitch,
either.

I would actually be slightly in favor of TC2050 based JTAG like we have
on a number of other sysmocom boards by now. It's small, has no
connector cost, and **we** have adapters for it anyway. Sure, it will be
a bit costly for "outside developers" wanting to do low-level JTAG
debugging/tracing, but in the worst case, one can always solder wires to
the pads if looking for a quick and low-cost version.

Not sure how others think about it, but I think at the moment I would
put the "ease of use" for those people first who actually have shown a
track record of working on it, which is "us".

> - have real SPDT switch rather than SPST: I'll again use the TS3A5018 for this.

This needs to be discussed with Kevin in detail. I think we might
actually need two of those for all required modes?

- passive tracing: SAM3 + SIM + phone connected at the same time
- card emulation / remsim: SAM3 + phone connected at the same time SIM disconnected

- card reader: SAM3 + SIM connected; phone disconnected
- MITM: SAM3 + SIM connected; SAM3 (other UART) + phone connected

The obvious approach of reaching the above is to have two SPST switches?

> - RTC Xtal: pins XIN32 and XOUT32 were designated for other GPIO functionality, but will now be freed for slow Xtal.

shouldn't be a big deal

> Regarding the backup cap: do you think of a (small) supercap? Then: what timespan do you need the backup domain and RTC remain valid?

I would not want to have THT or other bulky components by default. not sure if there are supercaps in SMT that aren't bulky? I think the use case must be studied further. I would expect that you'd basically connect some kind of battery to the device, then connect USB at the same time, set everything up for tracing, and then remove USB. In that scenario, there's the battery that's buffering the RTC as there's continous supply voltage. If you then were to disconnect the battery and later reconnect it, you'd loose time. But even then, it would only be the absolute time that you loose, you could still have relative timestamps for all events until you again disconnect the battery.

Thinking about a battery parallel supply however means we should keep the DC input in addition to the USB jack.

> - SPI flash: in the long run not used for logging, only used for version detection - let's find a proper µSD card holder, which is available and has a reasonable part's footprint.

I'm wondering if we didn't already build something with uSD slots and whether we have library parts. A quick scan of our git repo didn't flag anything, though.

> - version detection: we should introduce a 'version detection' voltage divider on ADx.

ACK.

> - two level translator for both USARTs seem to be desirable: their VccA is 3V3 for the ARM, VccB is the SIM voltage actually seen from the phone/modem. This would at least be needed if another idea is still on the list:

yes, I think it's clean to have proper level translation care must be taken to put it at the right place, see above discussion about 2xSPST, SPDT, ...

> - rigth now, the ARM can decide which power source the SIM should see (3V3 or Vcc_PHONE). Wasn't there that idea to have the ability to change between 1V8, 3V3 and 5V for the SIM when it is not powered by the modem/phone?

I think this is an esoteric feature. If we can get it more or less for free, fine. But I would guess that 99.9999% of users don't care which voltage is used - just as long as it works on the protocol level.

We've just seen that in 2018 you can buy new CCID readers that do 5V only (!)

So I wouldn't invest extra effort into it.

#16 - 08/29/2018 07:26 AM - tsaitgaist

laforge wrote:

> yes, we do have even one or two re-worked SIMtrace2 boards with the SAM4S, I think one of it is with Kevin.

yes, I have one and I will test it in the next days.

> no, we don't need it. OsmocomBB-style 2.5mm jack is standard by now, and it's not too much to ask people to have such a cable if they like console UART access

agree

> I would actually be slightly in favor of TC2050 based JTAG like we have on a number of other sysmocom boards by now. It's small, has no connector cost, and **we** have adapters for it anyway.

I'm not particularly a fan of the TC2050. When plugged in it is standing out quite a lot (not ideal for storing). When inserting/removing it too often, the metal guides start to scratch and damage the holes. Currently the TC2050 pinout for SWD is not standard (while we can fix that, it would be incompatible with existing boards). They are expensive (I only have one cable for currently 3 ports to debug and have to switch between them).
My preference would be a 2x5 IDC connector. The footprint is for free, and those interested using it can solder a shrouded pin header very fast. Even with a pitch of 2.54 it is not too large, and it is a very convenient standard pitch. IDC cables are common, strong, and not standing high up. SWD only required 4 pins but the 10 pin also allows to have JTAG for those who don't yet have SWD tools (and it does not use too much space). And finally it has a standard pinout (even if not widespread) for which there are ARM-20 adapters.
The main/only advantage I see for TC2050 is that for production testing it is a bit more convenient because you don't need to solder any header. But then if you want to test more you will likely have a test jig with pogo-pins, which also work nice on normal pin header holes.

> This needs to be discussed with Kevin in detail. I think we might actually need two of those for all required modes?

with one 4-line SPDT we should be fine if we use the following configuration:

```
         -- T -- USART0
CARD --\
         -- READER -- T -- USART1
```

\: SPDT switch on I/O, CLK, RST, VPP
CARD: smart card (i.e. SIM)
READER: smart card reader (i.e. phone)
USART0: will be used to communicate with the smart card when not connected to the reader (for CCID and MitM modes)
USART1: will be used to sniff reader to card traffic (sniff mode), and can act a card when "real" card is not connected to the reader (for card emulation and MitM modes)
T: voltage translator. we need at least the one on the reader side since we don't control the VCC and it can be 1.8V, 3.3V. 5.0V. the one on the card side is not mandatory since we control the VCC and can set it to 3.3V. Having one on the card side would provide support and allow "compliance testing" for all voltages (not sure how often this feature would be used, but it can be a nice/selling feature). Not having it would save one translator, and one adjustable voltage regulator (we only need a switch for providing 3.3V)

> I would not want to have THT or other bulky components by default. not sure if there are supercaps in SMT that aren't bulky? I think the use case must be studied further. I would expect that you'd basically connect some kind of battery to the device, then connect USB at the same time, set everything up for tracing, and then remove USB. In that scenario, there's the battery that's buffering the RTC as there's continous supply voltage. If you then were to disconnect the battery and later reconnect it, you'd loose time. But even then, it would only be the absolute time that you loose, you could still have relative timestamps for all events until you again disconnect the battery.

Instead of providing a supercap I would only provide a slot for a coin cell (CR2016-CR2032, oder smaller CR1220).
these batteries hold for years/decades (no timespan issue), and the user can choose to insert (or replace) it when required (no charging issue).
the time update would be done when using one of the simtrace2 host application.

> Thinking about a battery parallel supply however means we should keep
> the DC input in addition to the USB jack.

I would only have the USB socket and no separate power input.
for portable operation the user would just use a common USB power bank.
we just need to change the firmware to not reset when not enumerated.

- two level translator for both USARTs seem to be desirable: their VccA is 3V3 for the ARM, VccB is the SIM voltage actually seen from the phone/modem. This would at least be needed if another idea is still on the list:

> yes, I think it's clean to have proper level translation care must be
> taken to put it at the right place, see above discussion about 2xSPST,
> SPDT, ...

see above

> We've just seen that in 2018 you can buy new CCID readers that do 5V
> only (!)

surprisingly I haven't seen any smart card yet which does not support 5V.
The spec only requires to not be damaged at 5V, but not to operate at all voltages.

#17 - 09/06/2018 12:01 PM - tsaitgaist

the SAM4S doesn't fully work out of the box.

what works:
- serial output
- USB enumeration of the trace firmware
- sniffing with host application

what does not work:
- DFU flashing: the USB is not enumerated after switching to the bootloader (which cause the watchdog to trigger)
- forcing DFU: pressing on the bootloader button while resetting does not start the DFU bootlaoder

I did not changed anything in the code. I just modified the openOCD configuration to flash:
export APP=trace && export BOARD=simtrace && make -j 8 APP=dfu && make -j 8 && make -j 8 combined && openocd --file interface/stlink-v2.cfg --command "set CPUTAPID 0x2ba01477" --file target/at91sam4sXX.cfg --command "init" --command "halt" --command "flash write_bank 0 bin/$BOARD-$APP-combined.bin 0" --command "at91sam4 gpnvm set 1" --command "reset" --command "shutdown"

#18 - 09/06/2018 12:40 PM - laforge

On Thu, Sep 06, 2018 at 12:01:46PM +0000, tsaitgaist [REDMINE] wrote:

> the SAM4S doesn't fully work out of the box.

thanks for trying.

> what works:
> - serial output
> - USB enumeration of the trace firmware
> - sniffing with host application

I guess that's the most important part.

> what does not work:
> - DFU flashing: the USB is not enumerated after switching to the bootloader (which cause the watchdog to trigger)

that's interesting. I would not be surprised if flash programming is somehow different.

However, that the DFU firmware doesn't even enumerate is somehwhat odd.

> - forcing DFU: pressing on the bootloader button while resetting does not start the DFU bootlaoder

Also that part.

But in any case, I guess it shouldn't be too hard to resolve those issues **should** we chose to
migrate to SAM4S.

#19 - 09/06/2018 07:45 PM - mschramm

tsaitgaist wrote:

> I'm not particularly a fan of the TC2050. [...]

but me! ;)
I don't share your mechanical concerns: when storing, a programmer should be pulled off the PCBA, IMHO. The metal guides are sanded, and even when trying to insert incorrectly, they don't scratch
the holes, the glass fiber of the FR4 rather 'polishes' those stands / guides. Pin assignment: we used the 'old' Atmel's JTAGICE layout which is reasonable.

> And finally it has a standard pinout (even if not widespread) for which there are ARM-20 adapters.

Do you think of the Cortex debug connector pinout? Then: OK, but it is 1.27mm pitch... The need for an adapter remains, too. A 2x5 IDC connector would be nice *if* there were a common pinout for it.

Another option would be an edge connector, which is as cheap as a Tagconnect footprint, and it maintains a common pinout, the ARM20. Any opinions on that?

> [...] Having one on the card side would provide support and allow "compliance testing" for all voltages (not sure how often this feature would be used, but it can be a nice/selling feature). Not having it would save one translator, and one adjustable voltage regulator (we only need a switch for providing 3.3V)

I don't think we should aim for that feature, see Harald's remark on that above. - But we could measure the voltage used on the SIM (if powered from the phone/modem) when VCC_SIM is connected to a AD input (via a resistive divider), what do you think?

> Instead of providing a supercap I would only provide a slot for a coin cell (CR2016-CR2032, oder smaller CR1220).
> these batteries hold for years/decades (no timespan issue), and the user can choose to insert (or replace) it when required (no charging issue).

There are really tiny supercaps, I thought of something like the DCK or DSK types by Elna, they are reflow-soldering types and provide 0.07 resp. 0.22 Farad. - A good coin cell holder (e.g. not a type that shorts its contacts when no cell is inserted) is close to the price tag of these supercaps. But you are right, at least an additional diode + resistor is needed for charging these caps, another Schottky series diode would also be needed for both solutions.

After all, the cheapest solution is a simple CR1210 holder like this . However, on these types one contact is always a large PCB pad which connects best when the PCB has ENIG finish, HASL will work but may introduce specific contact problems for the battery. Moreover, because of their battery insertion/removal direction, they have to be close to a corner or an inner area where no other parts are in front.

>> Thinking about a battery parallel supply however means we should keep
>> the DC input in addition to the USB jack.

> I would only have the USB socket and no separate power input.
> for portable operation the user would just use a common USB power bank.

It seems that there is ongoing need to discuss the powering for autonomous usage. ;)

btw, laforge thought about using 3V as main supply voltage instead of 3V3. Did I get this right, and if yes: Harald, could you please explain the rationale behind this?

SPDT: I now inserted TXB0104 instead of your suggestion NVT2003 (they both can push-pull, but only the NVT can translate OD - which is not needed here). We still can use your 3-channel suggestion.

#20 - 09/06/2018 09:41 PM - tsaitgaist

mschramm wrote:

> Another option would be an edge connector, which is as cheap as a Tagconnect footprint, and it maintains a common pinout, the ARM20. Any opinions on that?

this restricts to put it on the edge, but is a convenient solution.
if we have a case for the simtrace v2, the connector would probably not fit in anymore, but on the other hand when programming/developing on the board you probably don't want the case anyway.
the more important detail is to not put the connector on the "rail" edge if we use a metallic case with rails to hold the board.

> After all, the cheapest solution is a simple CR1210 holder like this . However, on these types one contact is always a large PCB pad which connects best when the PCB has ENIG finish, HASL will work but may introduce specific contact problems for the battery. Moreover, because of their battery insertion/removal direction, they have to be close to a corner or an inner area where no other parts are in front.

there are also cheap non-edge battery holders:
https://eu.mouser.com/ProductDetail/Harwin/S8411-45R?qs=sGAEpiMZZMtKtLvoHD9hCwoX6MV76CJn
they are just higher

> btw, laforge thought about using 3V as main supply voltage instead of 3V3. Did I get this right, and if yes: Harald, could you please explain the rationale behind this?

I think this is because class B ISO-7816 cards actually operate at a nominal 3.0V. 3.3V is the maximum voltage.

#21 - 09/07/2018 07:34 AM - laforge

Hi Martin,

On Thu, Sep 06, 2018 at 07:45:41PM +0000, mschramm [REDMINE] wrote:

>> I'm not particularly a fan of the TC2050. [...]

> but me! ;)
> I don't share your mechanical concerns: when storing, a programmer should be pulled off the PCBA, IMHO. The metal guides are sanded, and even when trying to insert incorrectly, they don't scratch
> the holes, the glass fiber of the FR4 rather 'polishes' those stands / guides. Pin assignment: we used the 'old' Atmel's JTAGICE layout which is reasonable.

I think I made my arguments quite clear. We never saw much contribution from anyone but the simtrace makers (Kevin and me) over the last 7 or so years of the project. There were probably 2-3 cases where somebody actually contributed to the firmware. And for those few people, I think buying an adapter cable, or simply soldering a few wires to the pads is acceptable.

We could add the TC-2050 cable (and associated adapter to ARM-20-JTAG) to the sysmocom webshop to make it easier to obtain everything needed for firmware debugging. However, the cost of it remains. But I somehow like that idea, as sysmoQMOD users might want that, and also the upcoming sysmoOCTSIM and possibly more boards will share this connector + pinout. But we can do this irrespective of what happens on SIMtrace.

To summarize:

- 1.27mm pitch headers are too flimsy/fragile, and you cannot really do THT with them.
- ARM-20-JTAG is too large and occupies too much board surface, *unless* an edge-launch version is used
- TC-2050 requires a special cable

So if Kevin cannot accept/tolerate TC-2050, then we should either go for edge-launch, or for a completely non-standard "1x5 2.54mm" header pinout that doesn't require that large a footprint

> > [...] Having one on the card side would provide support and allow "compliance testing" for all voltages (not sure how often this feature would be used, but it can be a nice/selling feature). Not having it would save one translator, and one adjustable voltage regulator (we only need a switch for providing 3.3V)
>
> I don't think we should aim for that feature, see Harald's remark on that above. - But we could measure the voltage used on the SIM (if powered from the phone/modem) when VCC_SIM is connected to a AD input (via a resistive divider), what do you think?

all options are open here. If wanted, we could also use the IC we're planning to use on the sysmoOCTSIM, which allows voltage switching between 1.8/3/5V and it already contains the level shifter. OF course this wouldn't allow powering the card at arbitrary non-spec voltages.

> > Instead of providing a supercap I would only provide a slot for a coin cell (CR2016-CR2032, oder smaller CR1220).
> > these batteries hold for years/decades (no timespan issue), and the user can choose to insert (or replace) it when required (no charging issue).
>
> There are really tiny supercaps, I thought of something like the DCK or DSK types by Elna, they are reflow-soldering types and provide 0.07 resp. 0.22 Farad. - A good coin cell holder (e.g. not a type that shorts its contacts when no cell is inserted) is close to the price tag of these supercaps. But you are right, at least an additional diode + resistor is needed for charging these caps, another Schottky series diode would also be needed for both solutions.

Let's keep that as an option for further discussion. It would be interesting to compute the theoretical/expected amount of time such a 0.07 / 0.22 F supercap would keep the RTC running, considering values from the SAM3S/SAM4S data sheet on the RTC oscillator

> After all, the cheapest solution is a simple CR1210 holder like this . However, on these types one contact is always a large PCB pad which connects best when the PCB has ENIG finish, HASL will work but may introduce specific contact problems for the battery. Moreover, because of their battery insertion/removal direction, they have to be close to a corner or an inner area where no other parts are in front.

I don't like battery holders, sorry.

> It seems that there is ongoing need to discuss the powering for autonomous usage. ;)

The question is whether you want to fit everything into a kind of "phone sleeve" like demodulate wants to do. In this case, a USB power bank is much larger and heavier than the target. In that case, some small li-ion battery would work, but a USB power bank wouldn't.

> btw, laforge thought about using 3V as main supply voltage instead of 3V3. Did I get this right, and if yes: Harald, could you please explain the rationale behind this?

The rationale is simple:

- ISO7816 actually specifies the card voltages as 1.8V, 3V and 5V.
- 3.3V is the absolute maximum value if cards are powered by 3V.

All SIM cards **must** permit 3V operation, so that's the default voltage for us (unless we have some kind of adjustible SIM power supply onboard). So rather than always operating at the absolute maximum rating, we could simply go to the nominal voltage. I don't think the SAM3S or any of the other devices we have on board care. So this move would make us a bit more spec compliant.

#22 - 09/07/2018 04:10 PM - mschramm

tiny update on two aspects :

laforge wrote:

> It would be interesting to compute the theoretical/expected amount of time such a 0.07 / 0.22 F supercap would keep the RTC running, considering values from the SAM3S/SAM4S data sheet on the RTC oscillator

rough calculation:

- in the SAM3S ds, they claim 3µA typical current in backup mode (SAM4S: 1µA)
- the backup domain starts working at 1.62V, so a 3V supercap or battery has discharging end @1.62V; we can use the difference of 1.38V out of them.
- given the 0.22F-supercap, the avail charge when discharged to 1.62 V is 0.3 As, which is (div by 3.6) about 85 µAh. (70mF cap: 27 µAh)
- in best case, a 3µA sink can survive on this for about 28 hours (70mF cap: about 9h).
- for comparision: a CR1210 can deliver about 40mAh (discharged to 2V w/ very low currents), that is about 500 times that energy, which here

> would last 1.5 years.
>
>> After all, the cheapest solution is a simple CR1210 holder [...]
>
> I don't like battery holders, sorry.

I didn't plead for them either.

Regarding the 3V sourcing: the AT91SAM3S ds states that when USB is used, VDDIO needs to be higher than 3.0V. As we need series diodes in both supply paths anyway for power-OR-ing (planned: MBR0520; Vf=0.22V @If=10mA and 25°C), we must carefully select the parts in the "supposed-to-be-3V" supply rail to achieve a voltage >3.0V when not in backup mode. Another approach here would be to aim for a higer target voltage than 3.0V (but lower than 3.3V).

#23 - 09/07/2018 06:07 PM - tsaitgaist

laforge wrote:

> So if Kevin cannot accept/tolerate TC-2050, then we should either go for edge-launch, or for
> a completely non-standard "1x5 2.54mm" header pinout that doesn't require that large a footprint

I don't have strong feelings against TC-2050, just an opinion, but I don't mind using it.
no need for an alternative footprint (complicating the board). developers can solder wires or buy a cable.

#24 - 09/11/2018 03:22 PM - mschramm

- Checklist item [x] remove 6pin DEBUG connector? set to Done
Checklist item [x] have real SPDT switch rather than SPST set to Done
Checklist item [x] design with EAGLE to benefit from sysmocom parts library / processes set to Done
Checklist item [x] external USB pull-up circuitry with two transistors, shottyk diode can be removed (SAM3 and newer don't need it) set to Done
Checklist item [x] add RTC crystal (and cap)? for time-stamping readings/recordings set to Done
Checklist item [x] replace SPI flash with uSD card (for more storage/recordings)? or larger SPI? set to Done

#25 - 09/12/2018 03:25 PM - mschramm

- % Done changed from 0 to 60

We have two checklist items:

- add button for ERASE to SAM-BA
  and
- remove RESET button

I think a *button* for erase is error-prone, it should remain a (maybe unpopulated) jumper. What is the reason for the suggested move from juper/pinheader to a button?

The reset button was needed AFAIR to get the SAM7 into it's mask ROM bootloader. Does the suggestion to remove this button result from the move to SAM3 where this is not needed any longer (as bossac (almost) always works) ?

#26 - 09/13/2018 07:48 AM - laforge

the reset button was never needed for anything. In the years of using simtrace, it was always only very accident-prone, i.e. you inadvertently touch the button and have to start your tracing/task from scratch. very annoying.

To get into SAM-BA bootlodaer on SAM7 (and SAM3) you can simply close the erase jumper, attach USB, wait some seconds, disconnect USB, remove the jumper and then re-connect USB.

No reset button needed.

The request to add a ERASE button was basically to avoid having to have a jumper and close/open it. Simply pressing ERASE while connecting the board to USB would then be sufficient.

But that probably is almost as dangerous. Not quite, as pressing ERASE after power-up / during runtime is not dangerous. But anyway, let's keep ERASE jumper and remove RESET button.

#27 - 09/13/2018 07:09 PM - mschramm

- Checklist item [x] remove RESET button? set to Done

#28 - 09/24/2018 10:27 PM - mschramm

regarding the desired voltage domains for the SIM and associated level shifting:

laforge again brought the NCN8025 into discussion, a SmartCard interface IC containing ESD protection, LDO for 1V8, 3V0 and 5V SIM operation. It features two inputs for CLKDIV giving four clock divider, and with two voltage control inputs we can force one of three SIM operation voltages. There is a QFN-24 package, but even the QFN-16 version of this IC exposes all signals we would need for the Simtrace2. - This IC could solve several requirements at one go.
tsaitgaist : what do you think?

Files

cortex_debug_connector.gif	4.91 KB	08/28/2018	mschramm