# Internal EMAC (APLL) clock, configurable PHY clock output #1127

Closed **sauttefk** wants to merge 4 commits into `espressif:master` from `sauttefk:master`

Conversation 16 Commits 4 Checks 0 Files changed 11

**sauttefk** commented on Oct 16, 2017 Contributor

With this pull request the ESP32's Ethernet Media Access Controller (EMAC) can be configured to use the EPS32's internal APLL to generate a 50MHz clock being put out on `GPIO0`, `GPIO16` or `GPIO17` as a reference clock for the Ethernet PHY and EMAC or an external clock on `GPIO0` (as before).
New options for `make menuconfig` in `examples/ethernet/ethernet`

This is in succession of issue #1110

Tested with a LAN8720 board from Waveshare.

1

`Internal EMAC (APLL) clock, configurable PHY clock output` d64341a

**CLAassistant** commented on Oct 16, 2017 • edited

CLA signed

All committers have signed the CLA.

**igrr** commented on Oct 16, 2017 Member

Thanks for the PR! Would you mind changing the configuration to run-time instead of Kconfig? I.e. add the necessary options to `eth_config_t` and pass them where needed. This way, Ethernet driver can be used in environments where configuration is not known at compile time (such as micropython an Arduino).

This also means that there is going to be an extra overhead of `rtc_clk_apll_enalbe` being linked into the IRAM of output binary even if internal clock is not used. This can be remedied by adding a "Support internal clock generation using APLL" option in Ethernet Kconfig, and wrapping corresponding parts of code if `#ifdef`, returning ESP_ERR_NOT_SUPPORTED if APLL is disabled in Kconfig but requested at run time.

**sauttefk** added 2 commits 7 years ago

`Internal EMAC (APLL) clock, configurable PHY clock output - now using…` ... e1fd091

`Internal EMAC (APLL) clock, configurable PHY clock output - variable …` ... 65cb55a

**sauttefk** commented on Oct 17, 2017 Contributor Author

OK, changed. The Ethernet clock configuration is now part of the `eth_config_t` structure.
But I'm not sure if it makes sense to exclude `soc/rtc.h` by an `#ifdef` this saves just 360 bytes.
I also think, that using the internal clock will be the default usage in the future as this reduces the hassle with `GPIO0`, makes one crystal oscillator obsolete and potentially frees one GPIO pin.

**sauttefk** mentioned this pull request on Oct 18, 2017

**ETH_LAN8720** espressif/arduino-esp32#744

Closed

**sauttefk** commented on Oct 23, 2017 Contributor Author

please merge

**sauttefk** commented on Oct 30, 2017 Contributor Author

@igrr: please merge this PR, so we can easily procede here: https://github.com/espressif/arduino-esp32

**projectgus** self-assigned this on Oct 30, 2017

**projectgus** commented on Oct 30, 2017 Contributor

I'm sorry for the delay, @sauttefk . This has been waiting on someone (nominally me) to wire up a PHY and run some tests with internal & external clock configs, but I keep getting pulled onto other tasks.

(If you can clarify which clock configs you also tested with the WaveShare board then this may help speed things along.)

**sauttefk** commented on Oct 31, 2017 Contributor Author

@projectgus I took my waveshare test setup (here) removed the two 33Ω vertical resistors below the LAN8720 and mounted one of the 33Ω resistors horizontally between the upper two pads. In this way, the crystal oscillator is completely disconnected and I can choose between `GPIO0`, `GPIO16` or `GPIO17` as clock source.
`GPIO17` (the inverted signal `EMAC_CLK_OUT_180`) works perfect. `GPIO0` and `GPIO16` also worked, but not very stable (only when some additional capacitive load was on the clock signal - i.e. the probe of my oscilloscope or after i added a 74hct04 inverter in between). I haven't analysed this any further, maybe this is because of the sampling time of the other data signals relative to the clocks edge or due to additional delay due to the inversion of the clock.

What I can tell: I was pinging the test board over 3 days without one ping missed with `GPIO17` as clock source.

**igrr** added the **Status: Pending** label on Nov 10, 2017

**igrr** unassigned **projectgus** on Nov 10, 2017

**sauttefk** mentioned this pull request on Nov 14, 2017

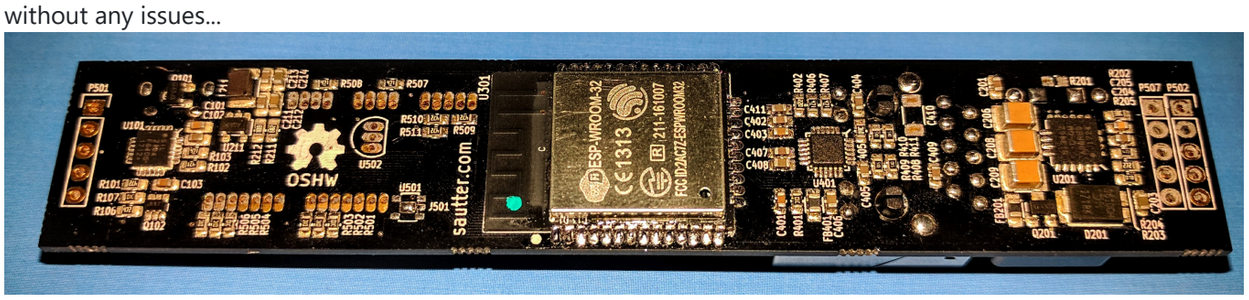
**[TW#16425] SD card file system & ethernet compatibility issue** #1237

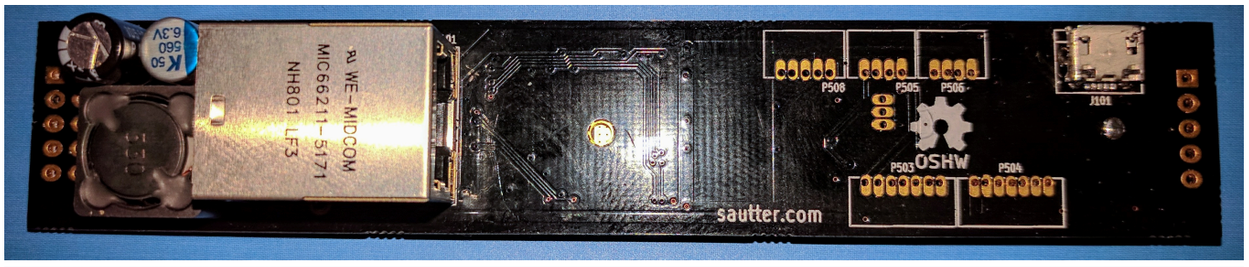
Closed

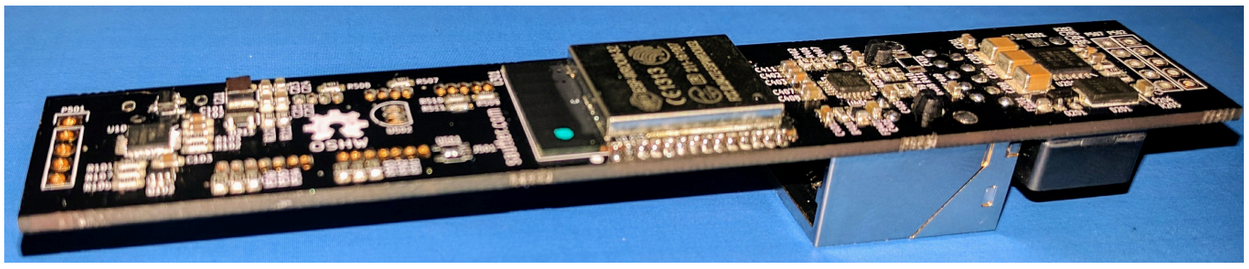
`Set GPIO-mode for MDC and MDIO pins` `51badec`

**sauttefk** commented on Nov 14, 2017 — Contributor, Author

**Please merge this PR.**
My prototype PCB using `EMAC_CLK_OUT_180` on `GPIO17` with LAN8720 and POE is working now several days without any issues...



**igrr** commented on Nov 14, 2017 — Member

Sorry for the slow turnaround **@sauttefk**, this PR had been cherry-picked in `f324458`. Note that the last commit you added wasn't included, as we grabbed the version of this PR about 4 days ago.

**igrr** closed this on Nov 14, 2017

**igrr** commented on Nov 14, 2017 Member

Will keep this open until the commit about GPIO configuration is merged.

**igrr** reopened this on Nov 14, 2017

**igrr** pushed a commit that referenced this pull request on Nov 22, 2017

`Set GPIO-mode for MDC and MDIO pins` ... `157371e`

**igrr** commented on Nov 23, 2017 Member

MDC/MDIO configuration fix has been cherry-picked as `157371e`, thanks **@sauttefk**!

**igrr** closed this on Nov 23, 2017

---

**igrr** mentioned this pull request on Dec 5, 2017

**[TW#12196] expand emac_dev and eth_config_t for choice clk src intern / extern** #395

Closed

**sauttefk** added a commit to sauttefk/arduino-esp32 that referenced this pull request on Dec 11, 2017

`Allow configuration of Ethernet PHY clock source` ... `c17b3c0`

**sauttefk** mentioned this pull request on Dec 11, 2017

**Allow configuration of Ethernet PHY clock source** espressif/arduino-esp32#916

Merged

**sauttefk** added a commit to sauttefk/arduino-esp32 that referenced this pull request on Dec 11, 2017

`Allow configuration of Ethernet PHY clock source` ... `5bba6c1`

**me-no-dev** pushed a commit to espressif/arduino-esp32 that referenced this pull request on Dec 19, 2017

`Allow configuration of Ethernet PHY clock source (#916)` ... `75bc1e6`

**Matheus-Garbelini** commented on Jul 5, 2018

Any more tests with GPIO0? I need to design a board with esp32 wrover. Gpio 16 and 17 cannot be used to clock ethernet as it's used by psram.

**igrr** mentioned this pull request on Jul 9, 2018

[TW#24566] LAN8720 TX issue with ethernet_example (IDFGH-1393) #2164

Closed

**cnlohr** commented on Jul 10, 2018 Contributor

Just confirming, @sauttefk you had to include an inverter when using GPIO16/GPIO0? Right now it looks like I'm getting 100% packet throughput going into the ESP32, but TXing from the ESP32, I'm at 0% sending. I am not even getting "errors" on the other end.

**Matheus-Garbelini** commented on Jul 16, 2018 • edited

@cnlohr What inverter are you using?
I'm buying this NL17SZ00DFT2G IC which seems to operate up to 100Mhz. But we still need confirmation if this works correctly.

**cnlohr** commented on Jul 17, 2018 Contributor

It will likely introduce a phase delay, which could be troubling. What I really want to do is to try to phase-delay the GPIO0 output enough to get it to work with, saving GPIO 16/17.

👍 1

**igrr** removed the Status: Pending label on Aug 9, 2018

**Curclamas** pushed a commit to Curclamas/arduino-esp32 that referenced this pull request on Aug 21, 2018

`Allow configuration of Ethernet PHY clock source (espressif#916)` ··· a1acc55

**HarterHorst** commented on Sep 23, 2018 • edited

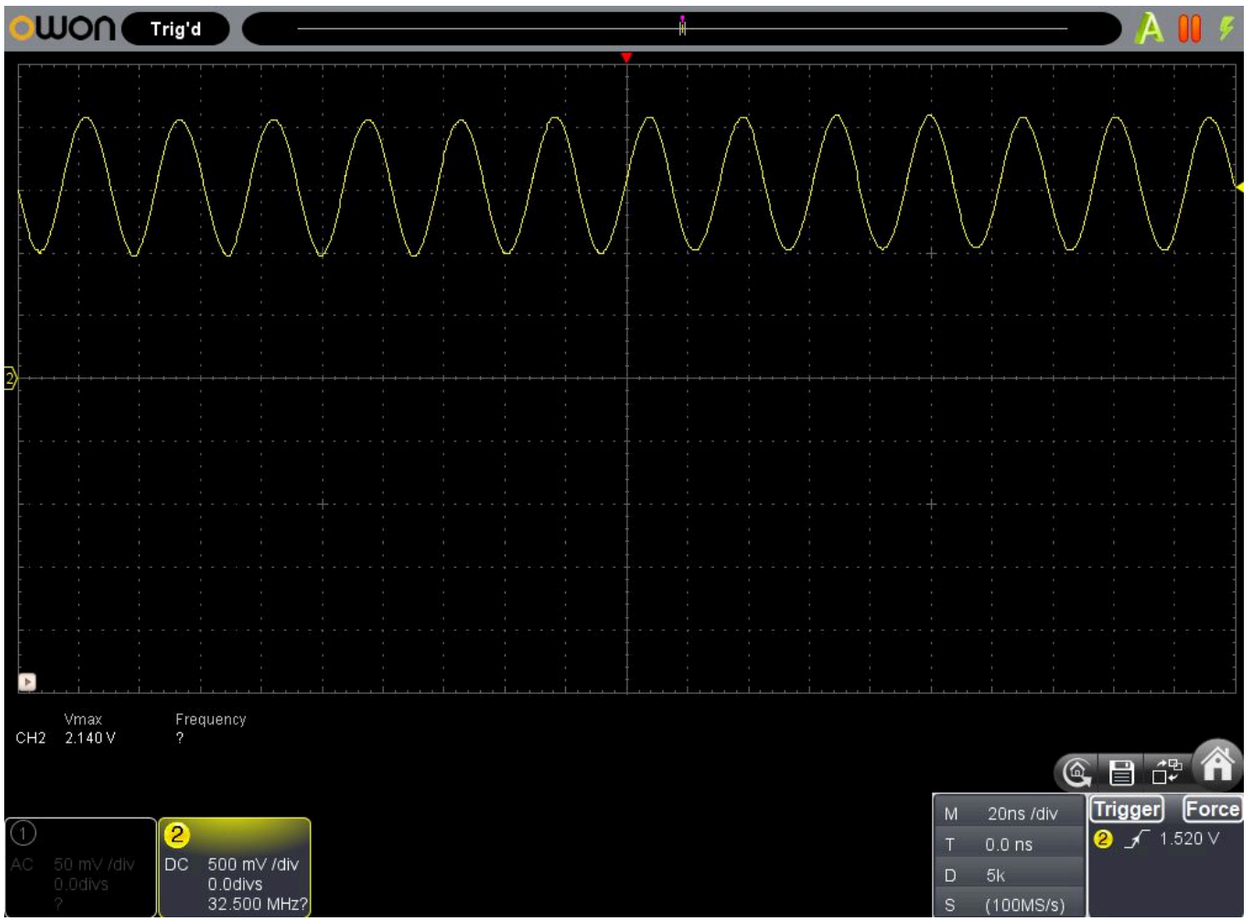
@sauttefk i have the same setup like you have mentioned above. Removed the two 33 Ohm resistors and put one horizontally between the two pads below the 8720. I have connected PIN17 with pull up 3,3 volt to nINT/RETCLK and configured #define ETH_CLK_MODE ETH_CLOCK_GPIO17_OUT in the Arduino script. Pin 0 is not connected.

Script starts and goes "ETH Started". Thats it, nothing more.

```
rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT) configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00 mode:DIO, clock div:1
load:0x3fff0018,len:4 load:0x3fff001c,len:1324 load:0x40078000,len:7788 ho 0 tail 12 room 4
load:0x40080400,len:6448 entry 0x400806e8 ESP32 SDK: v3.2-dev-1055-g3276a1316 ETH Started
```

Cabling goes via breadboad (i know that this will cause trouble with 50MHz) so I checked the signal. Unfortunatly my cheap Oscilloscope can only handle max 25MHz but what I see looks OK for me.

Did I miss something when I use the ESP Arduino framework and not the ESP-IDF? The OLIMEX ESP32-EVB works.




**coyool** mentioned this pull request on May 8, 2020

**GPIO0 can not be used as CLK sorce for ETH together with PSRAM** espressif/arduino-esp32#3979

Closed

Reviewers

No reviews

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

Successfully merging this pull request may close these issues.

None yet

7 participants