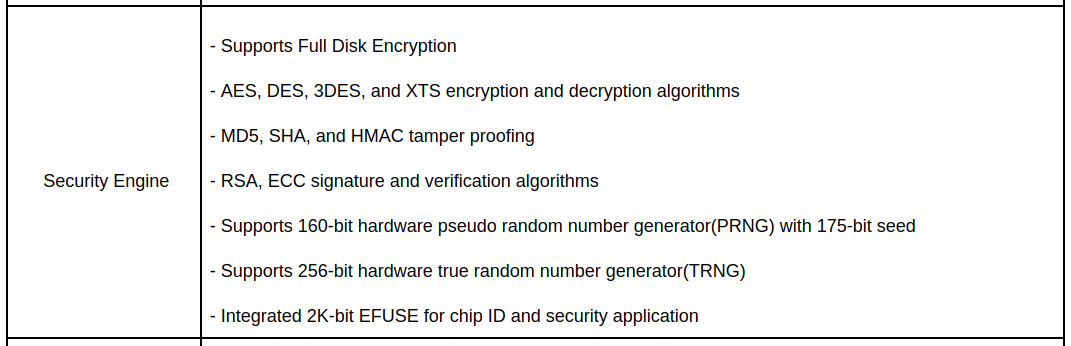
**fdx-MYiR: Consult**

------------------------------------------------------------------------------------------------------

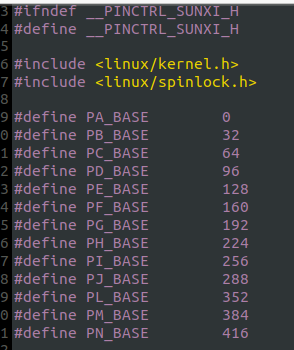
We plan on allowing the system we're developing to be updated on-the-field, but with no network connection. To do so in a secure manner we were thinking about using a USB/SD Card with a symmetric encryption scheme, and when we were researching on WHERE to store the key, we realized that the module has a security engine:

And the 2K-bit EFUSE for chip ID could be a great solution for this.

**[1]** – Is this chip ID permanently written from production? If so, how can it be accessed? If not, how can we store our own ?

------------------------------------------------------------------------------------------------------

We could manipulate the GPIO's from user-space using commands with no inconvenience. But we need to do so in a programmatic approach, and we're having issues on understanding how are the module's GPIO pins mapped into the gpiochip lines. We found this within the *pinctrl-sunxi.h* :

**[2]** Does this means that if I want to manipulate for example: PD22. I should use **gpiochip1[pio]** and line **(PD\_BASE + LINE)** = 96 + 22 = 118 ?

------------------------------------------------------------------------------------------------------