

Tesis

Descarga de programas

- **Diseño de PCB:** Altium. Se puede pedir licencia de estudiante con mail de la facultad y descargarlo oficialmente. (<https://www.altium.com/solutions/academic-programs/student-licenses>)
- **Firmware:** Atmel Studio 7. En la página hay que meterse para encontrarlo, porque no existe más con ese nombre. (<https://www.microchip.com/mplab/avr-support/avr-and-sam-downloads-archive>)
- **Software:** Python.

Recursos

- **Diseño de PCB:** esta guía de videos me fue muy útil para entender el workflow en Altium (<https://www.youtube.com/watch?v=KpgTud1iQ-4&list=PLXvLToQzgdfKKQn2wmpuSXz6sROQmO6R&t=0s>)
- **Firmware y software:** hay muchos tutoriales. El microcontrolador se programa en C.

Firmware

El firmware está organizado en distintas carpetas:

- **apps:** componentes principales.
- **comm:** módulos que se ocupan de la comunicación con la UART/USB
- **api:** interfaz api para comunicar las apps con el módulo de comm
- **apps_dsp:** componentes para el procesamiento de señales

Atmel Studio

Esta herramienta va a permitir buildear, generar código y si tenemos el programador de Atmel también subir y debuggear el micro desde ahí.

Suponiendo que querramos agregar una funcionalidad al micro, por ejemplo DAC, los pasos para comenzar a ver cómo funciona son los siguientes:

1. Generar un proyecto de ejemplo (File -> New -> Example project) que use el periférico de la manera que esperamos.

2. Copiar el código y modificarlo para nuestro proyecto

Interfaz Python

Para correr la interfaz de python hay que instalar las dependencias. Para eso correr:

```
pip install -r requirements.txt
```

parados en la carpeta `software` . Si faltara alguna (puede ser) y tira error, correr:

```
pip install <dependencia_que_falta>
```