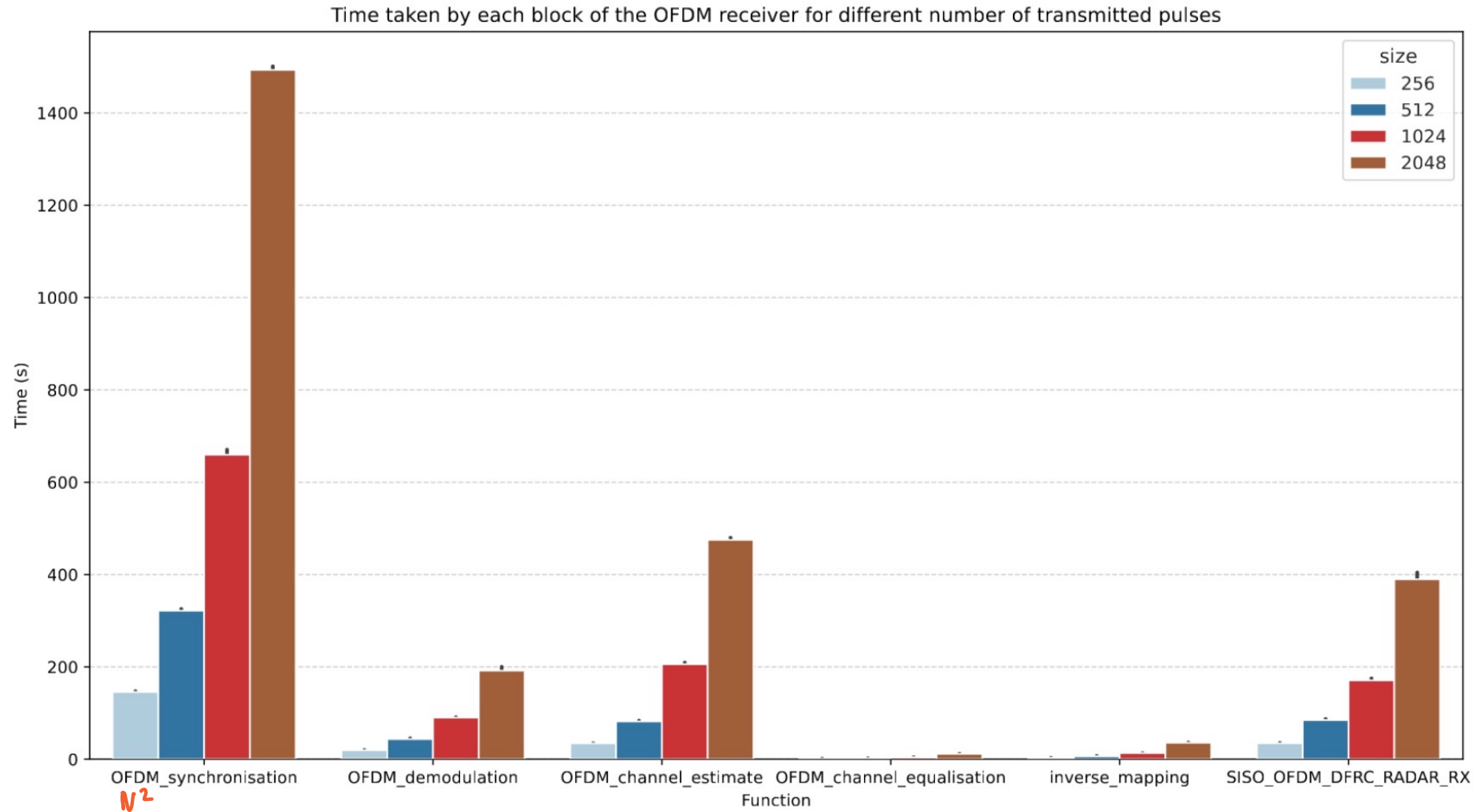


OFDM chain in python script

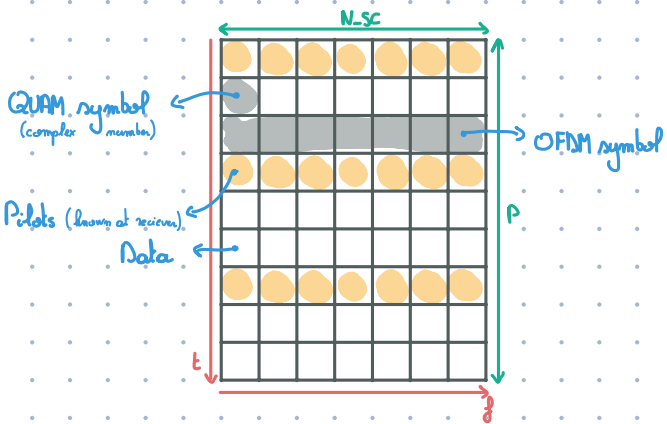
① Transmitter

② Receiver
→ check radar



OFDM transmission

Matrix of OFDM symbols

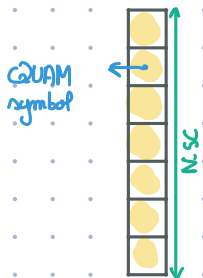


- Bitstream can be extracted (or can define) the matrix
Shape will be P, N_{SC} . # bits per QAM symbol
- Pilots can be extracted or inserted in the matrix
(not memory in a line)

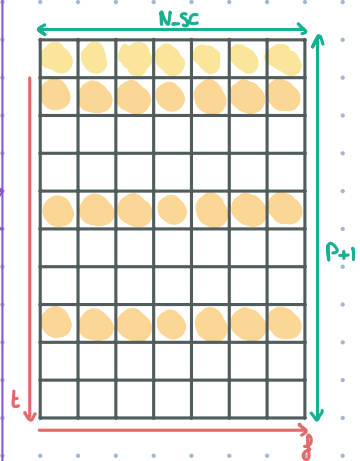
P
Number of transmitted pulses

N_{SC}
Number of subcarriers

Preamble

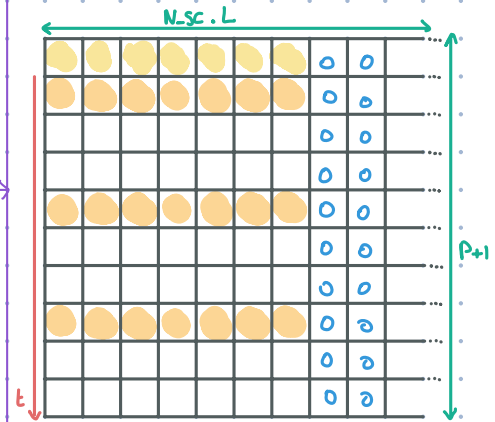


Concatenation



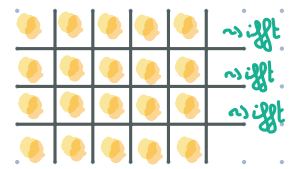
OFDM modulation

Over-sampling



L
Over-sampling factor

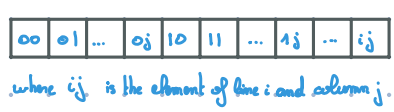
IFFT



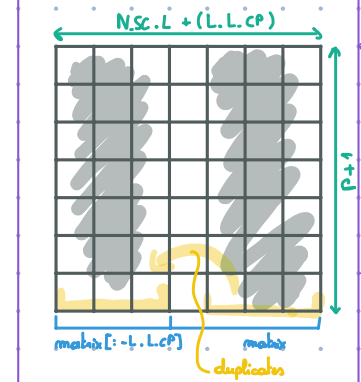
Normalizat°

$\times \text{sgnt}(N_{SC} \cdot L)$

Reshape



Cyclic prefix addition

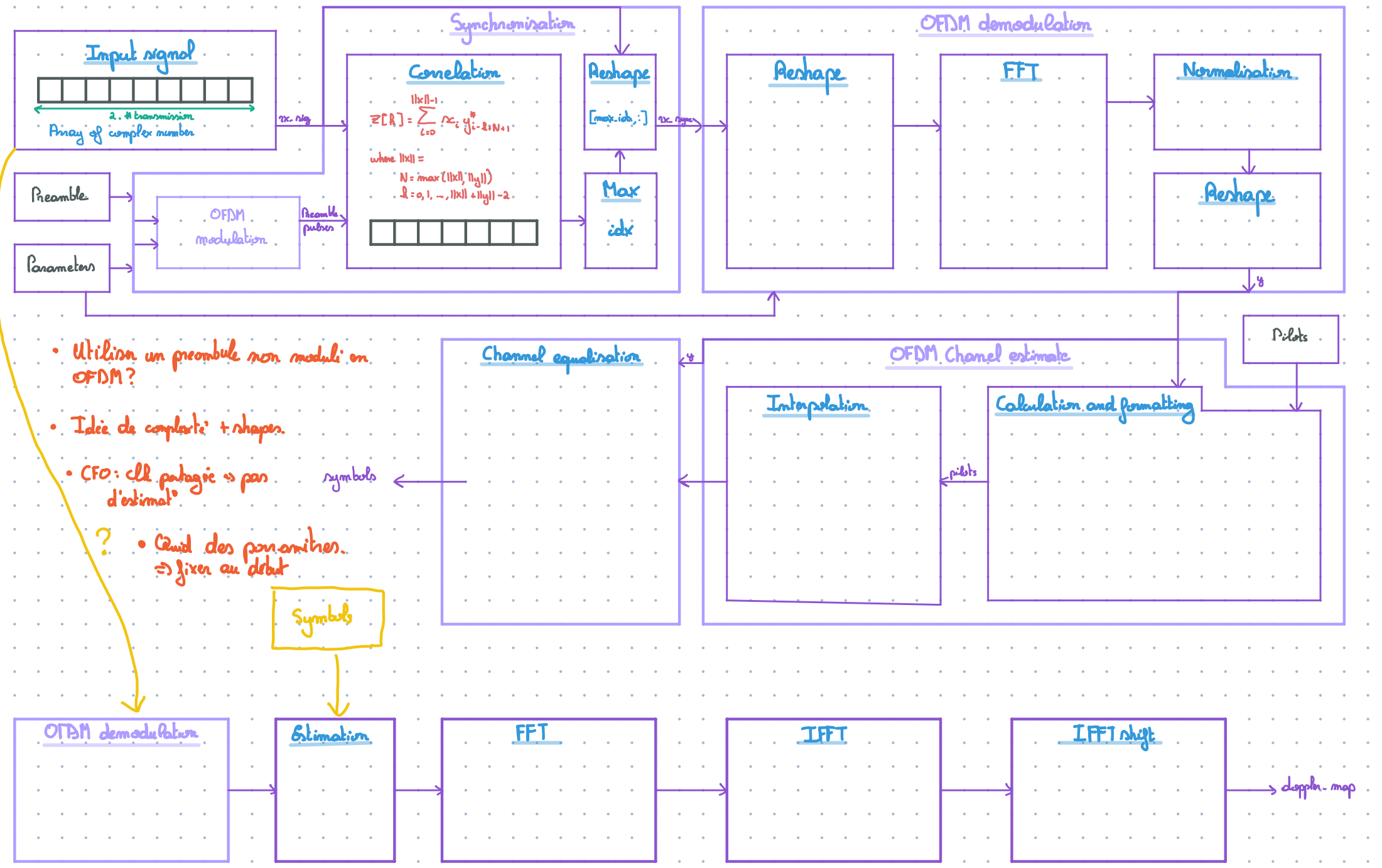


$L \cdot CP$
Cyclic prefix length

Output

- 1) Concatenate I/Q symbols
- 2) Normalisation

OFDM reception



- Utiliser un preamble non modulé en OFDM?
- Idée de complexité + reshape.
- CFO: est partagé ⇒ pas d'estimation.
- Quel des paramètres. ⇒ fixer au début.

Adding FPGA in toolchain

Some questions

→ how to program? HDL or LabView?

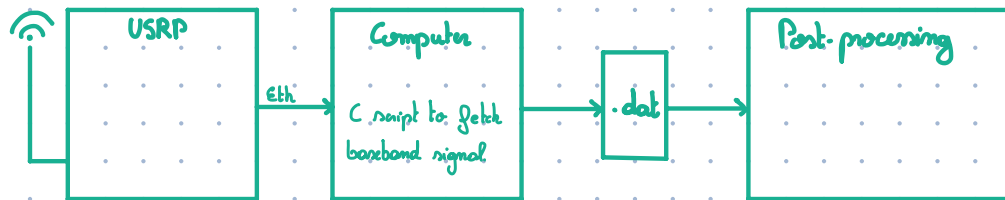
→ how to flash?

→ how to communicate 1) signal → FPGA 2) FPGA → computer

Current toolchain

Step 1

Step 2



Code python:

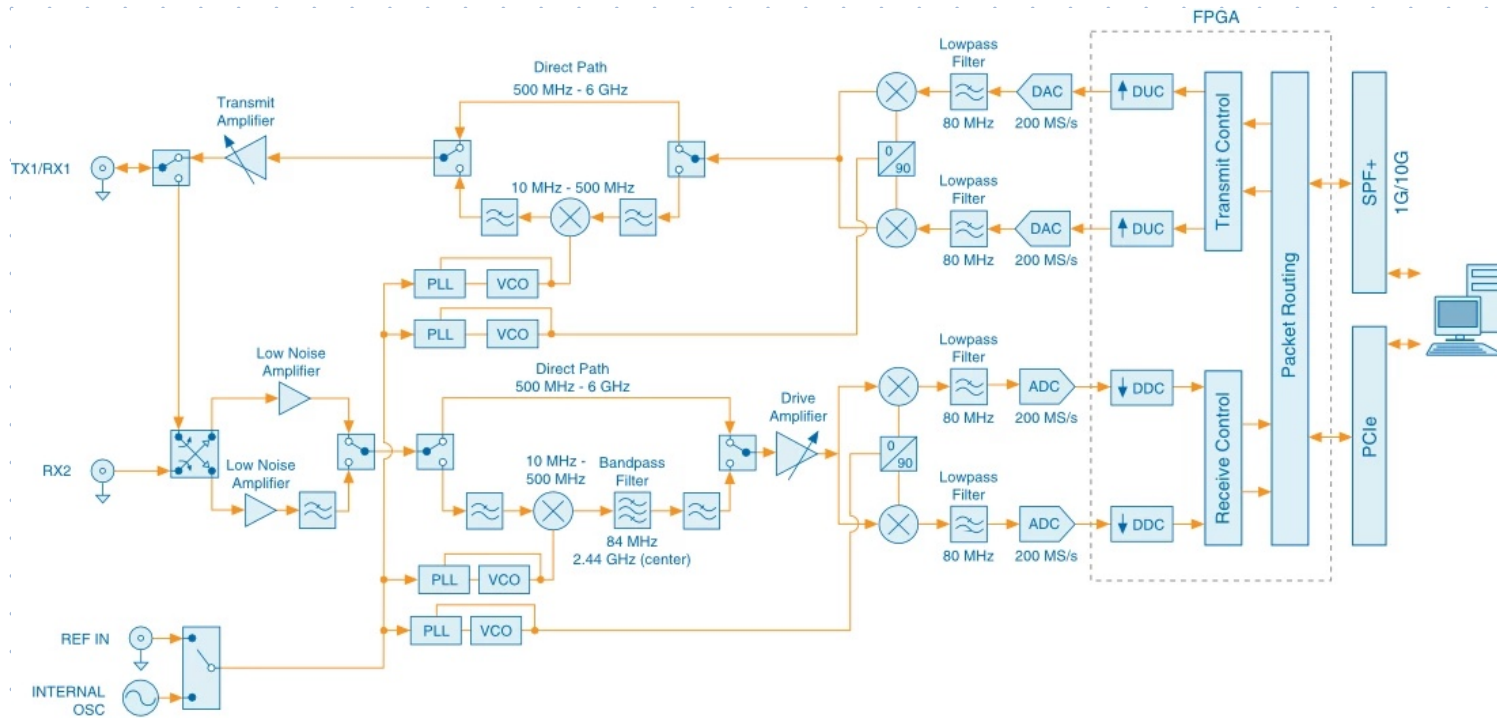
→ complexité et # opérations

→ Shapes

→

Différence implém en changeant python en C. → Analyse

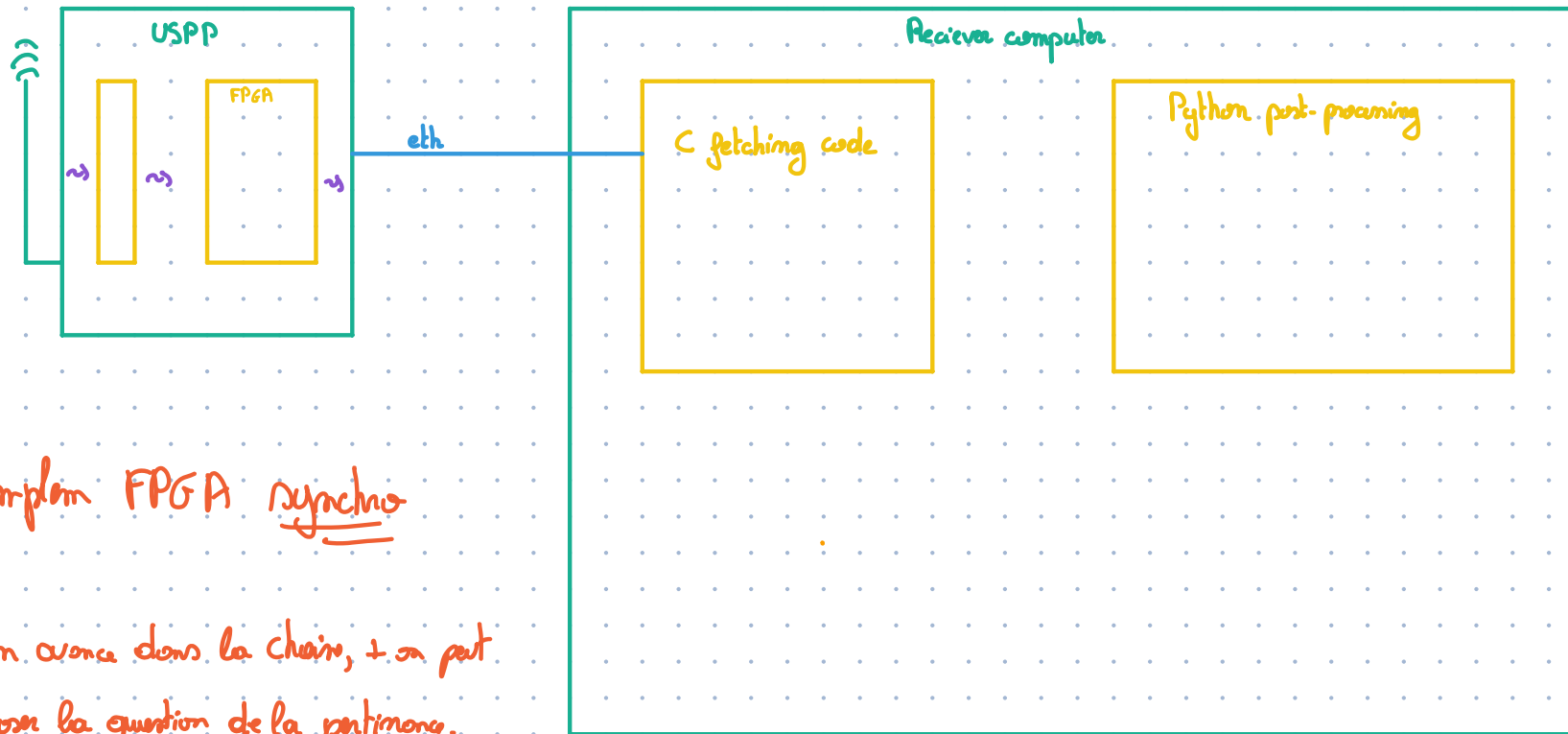
Blocs existent déjà. → Pas a des blocs pour ça
Comment la synchronisation est faite



FPGA: ce que tu veux

DSP? → n'app
↳ processus avec jet
spécifique.

- How / where to add FPGA ? Only use the one in USRP?



- high level synthesis. \rightarrow code \Rightarrow FPGA

\Rightarrow Lien notebook vers programme FPGA
Ethrus + NI.

Implément FPGA synchron

+ on avance dans la chaîne, + on peut
se poser la question de la pertinence.
de faire de l'hardware.

Partage de temps

FPGA / Compléments

Jendi 07/14h30 Tuning