

BAG repository history

Trying to get things on GIT sorted out...

BAG: GITHUB dependencies / latest updates

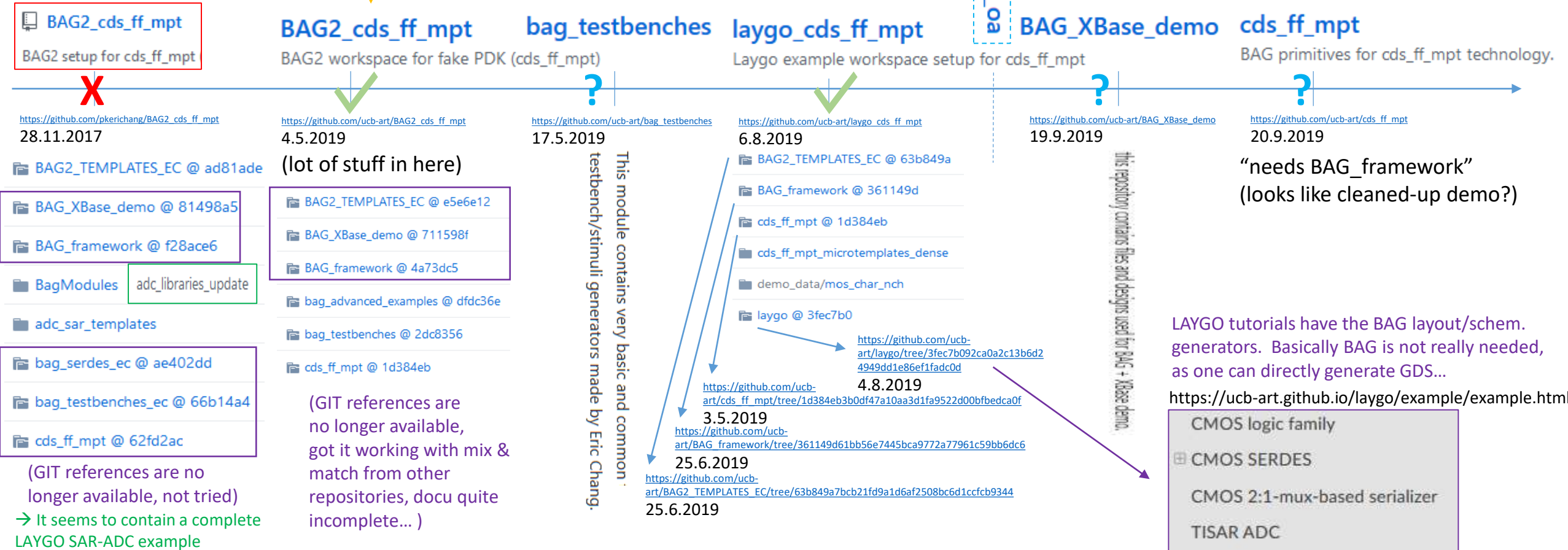
Eric Chang
pkerichang

NOT IN UCB-ART!

Actual FH Setup based on assumed "Bootcamp example (somehow working after a lot of trial & error...)"

14.9.2019:
initial commit for BAG_framework!

From here, added additional complexity by some C++ BAG interface to compile (not tried yet) – all this is yet untested, I do not dare to mix this with the older stuff...



BAG: GITHUB dependencies / latest updates



NOT IN UCB-ART!

bag
Forked from bluecheetah/bag
BAG framework

BAG_framework

bag_advanced_examples

cybag_oa

laygo

LAYout with Gridded Objects

laygo2

LAYout with Gridded Objects v2

BAG2_TEMPLATES_EC

https://github.com/ucb-art/BAG_framework
2.1.2020

https://github.com/ucb-art/bag_advanced_examples
2.1.2020


https://github.com/ucb-art/cybag_oa
13.1.2020

<https://github.com/niftylab/laygo2>
14.1.2020

<https://github.com/ucb-art/laygo>
15.1.2020

<https://github.com/niftylab/laygo2>
21.1.2020

https://github.com/ucb-art/BAG2_TEMPLATES_EC
15.2.2020

 **cybag_oa @ 614b416**

Berkeley Analog Generator (BAG) version 2.0 and later.

BAG 2.0 is a complete rewrite of BAG 1.x (which is in pre-alpha stage and never released).
Every outstanding Documentation and Install instructions can be found at <http://bag.berkeley.edu/>
A tutorial setup is available at https://github.com/ucb-art/BAG2_code_templates_EC

https://github.com/ucb-art/cybag_oa/tree/614b4161112c1add2ee8a11a5ce18db6533e34b1
14.9.2019

- **bag_advanced_examples**
The schematic templates
- **advanced_examples**
The layout generators
- **BagModules**
The schematic generators
- **scripts_sample**
Sample top-level run scripts
- **specs_sample**
Sample yaml specification files

bag_oa is a C++/Cython wrapper around the OpenAccess API.
You can use it to speed up layout generation in BAG 2.

BAG, a recursive acronym which stands for "BAG AMS Generator", is a fork and successor of the BAG_framework.

BAG (BAG AMS Generator)

"needs BAG2"
no longer developed

(although using the CDS dummy technology, all examples described there seem to be NOT open source)

https://ucb-art.github.io/laygo/tutorial/tutorial_BAG_v2.html

- LAYGO2 is optimized to IC layout generations in the following conditions.
1. Automations and reuse of layout design flow.
 2. Parameter-based layout generation.
 3. Layout in advanced processes (e.g. FinFET CMOS).
 4. Code-based layout generation.

- adc_vir
- analog_core
- analog_mos
- dac
- digital
- laygo
- passives
- resistor
- routing
- serdes
- _init_py
- mos_char.py