

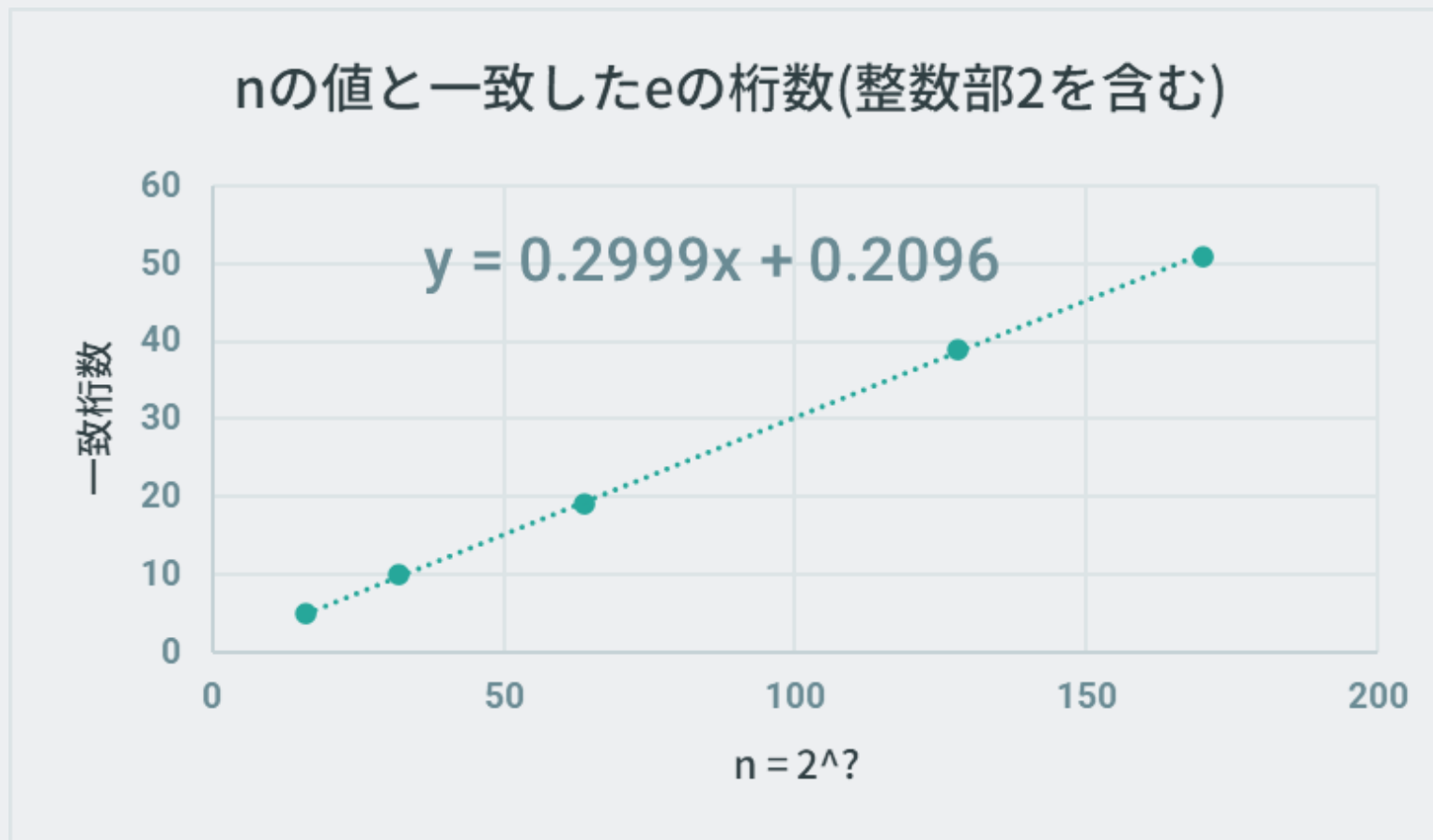
デジタルトレーニング FPGA編

飯塚研究室 B4 福島幸弥

| 前回の続き

- $e = \lim_{n \rightarrow \infty} \left(1 + \frac{1}{n}\right)^n$ について、精度を調べた
- Pythonで桁数の精度を検証
- Verilogで必要なクロック数を検証

| 結果(Python)



- Pythonでは $n = 2^{170}$ 程度が限界
- 近似直線より、100桁合わせるには $n \geq 2^{333}$ が必要

| 結果(Verilog)

- $n = 2^{15}$ かつ、データ長が $16 \times 32 = 512\text{bit}$ のとき
1027クロック/累乗1回 → 計15,405クロック
- 累乗1回にかかる計算量は $O(\text{データ長}^2)$
累乗の回数は $O(\log_2 n)$
- 100桁の計算には最低でも $1027 \times 333 = 3.4 \times 10^5$ クロック必要
→ 実際はデータ長を伸ばさないと誤差が大きくなるため
さらにクロック数がかさむと考えられる

| 代わりに用いたアルゴリズム

- 他の人のアルゴリズムを参考にした
- $e = \sum_{n=0}^{\infty} \frac{1}{n!}$
- $\frac{1}{n!}$ から $\frac{1}{(n+1)!}$ を割り算で求める
- $\log 70! < -100$ なので、 $n = 70$ まで求めればよい
- $100 \log_2 10 \approx 332$ なので、少なくとも333bit必要
- 8bitを1単位とする多倍長演算を行う
- 簡単のため、計算時には $8 \times 50 = 400\text{bit}$ で実装した

| 除算器

- 実際には16bit / 8bitを行う
→ 前のあまり8bit + 今回の被除数8bit
最初はあまり = 0として計算
- これを1つのnにつき50回繰り返す

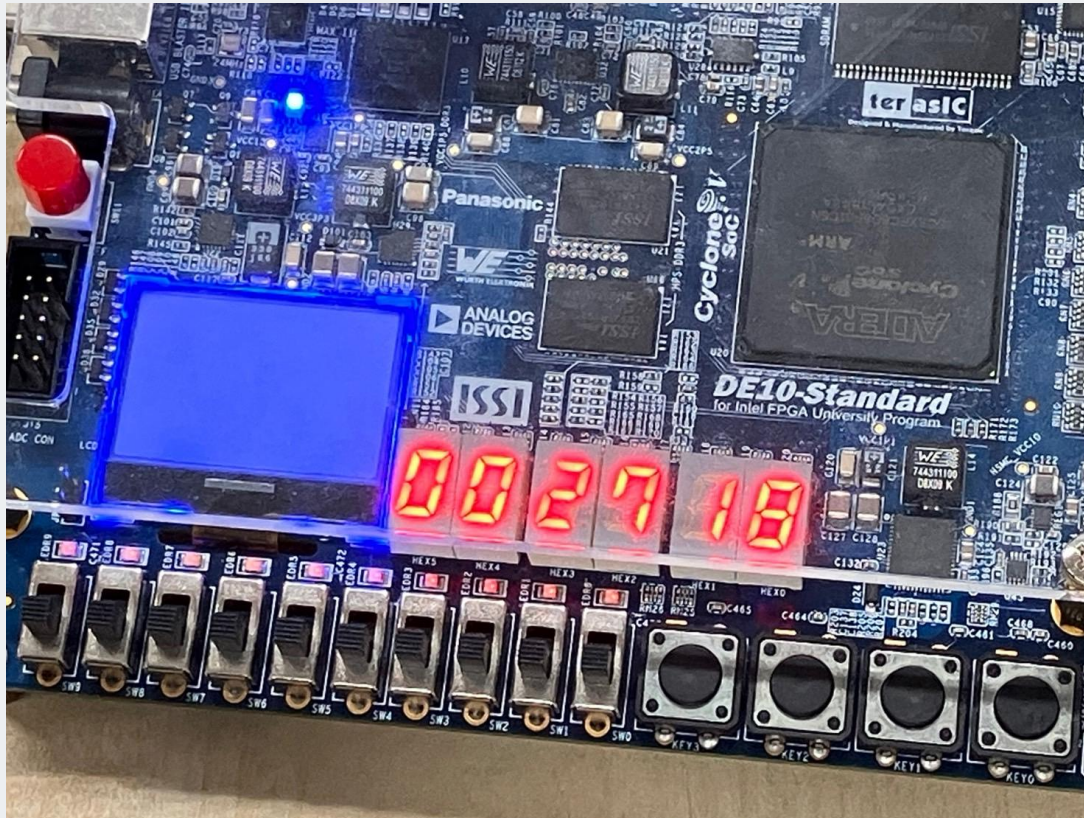
| 加算器

- 8bit全加算器を50回用いる
- 時間の都合上、加算器・除算器ともにパイプライン化などを行わずナイーブに逐次計算する仕様で実装した

| 二進→十進変換器

- 計算結果の上位5bit目~8bit目(整数部下位4bit分)を記録
0000**0010**.ABCD...
- 抽出後，上位8bitをゼロに
0000**0000**.ABCD...
- この3bit左シフト + 1bit左シフトで次の桁を求める
0000**0ABC**.DEFG... + 0000**000A**.BCDE...
- これを出力する桁数ぶん繰り返す

実機での動作結果



- 動作周波数は32.83MHz
→50MHzは下回ったが
正しく動作した
- gtkwave上では
除算に51クロック/回
加算に3クロック/回
合計3780クロック→115 μ s

十進変換に1クロック/1桁

実機での動作結果

Flow Summary

 <<Filter>>

Flow Status	Successful - Fri May 16 12:15:59 2025
Quartus Prime Version	20.1.1 Build 720 11/11/2020 Patches 1.02i SJ Lite Edition
Revision Name	top_e_display
Top-level Entity Name	top_e_display
Family	Cyclone V
Device	5CSXFC6D6F31C6
Timing Models	Final
Logic utilization (in ALMs)	1,606 / 41,910 (4 %)
Total registers	2612
Total pins	47 / 499 (9 %)
Total virtual pins	0
Total block memory bits	0 / 5,662,720 (0 %)
Total DSP Blocks	0 / 112 (0 %)
Total HSSI RX PCSs	0 / 9 (0 %)
Total HSSI PMA RX Deserializers	0 / 9 (0 %)
Total HSSI TX PCSs	0 / 9 (0 %)
Total HSSI PMA TX Serializers	0 / 9 (0 %)
Total PLLs	0 / 15 (0 %)
Total DLLs	0 / 4 (0 %)