

# Review Tackle Bench

Jörg Mische

September 29, 2015

## Abstract

This document will contain the first review of the tackle bench regarding the topic DSP filters.

## 1 Integer DSP Filters

- ~~SPstone\_fixed\_point/convolution\_fixed~~
- ~~DSPstone\_fixed\_point/fft\_16\_7~~
- ~~DSPstone\_fixed\_point/fft\_16\_13~~
- ~~DSPstone\_fixed\_point/fft\_1024\_7~~
- ~~DSPstone\_fixed\_point/fft\_1024\_13~~
- ~~DSPstone\_fixed\_point/iir\_biquad\_N\_sections\_fixed~~
- ~~DSPstone\_fixed\_point/iir\_biquad\_one\_section\_fixed~~
- ~~DSPstone\_fixed\_point/lms\_fixed~~
- ~~MRTC/edn~~
- ~~MRTC/fdct~~
- ~~MRTC/fft1~~
- ~~MRTC/fir~~
- ~~MRTC/fjdetint~~
- ~~MRTC/lms~~

### 1.1 ~~DSPstone\_fixed\_point/convolution\_fixed~~

REMOVE: Extremely simple loop, part of FIR-filters

### 1.2 ~~DSPstone\_fixed\_point/fft\_16\_7~~

REMOVE: same algorithm as ~~DSPstone\_fixed\_point/fft\_16\_13~~, only one loop is missing

### **1.3 DSPstone\_fixed\_point/fft\_16\_13**

REMOVE: same algorithm as DSPstone\_fixed\_point/fft\_16\_7, one additional loop and more bits per entry

### **1.4 DSPstone\_fixed\_point/fft\_1024\_7**

REMOVE: same algorithm as DSPstone\_fixed\_point/fft\_16\_7, but larger matrix

### **1.5 DSPstone\_fixed\_point/fft\_1024\_13**

KEEP: same algorithm, but largest workload

TO BE DISCUSSED: Alternatively keep smaller workload?

### **1.6 DSPstone\_fixed\_point/fir2dim\_fixed**

KEEP: two dimensional FIR

### **1.7 DSPstone\_fixed\_point/fir\_fixed**

REMOVE: very simple, part of DSPstone\_fixed\_point/lms\_fixed, MRTC/fir is more complex

### **1.8 DSPstone\_fixed\_point/iir\_biquad\_N\_sections\_fixed**

KEEP

### **1.9 DSPstone\_fixed\_point/iir\_biquad\_one\_section\_fixed**

REMOVE: no loops, one iteration of DSPstone\_fixed\_point/iir\_biquad\_N\_sections\_fixed

### **1.10 DSPstone\_fixed\_point/lms\_fixed**

REMOVE: very simple, MRTC/lms is more complex

### **1.11 MRTC/edn**

KEEP: several filters (including DCT, FIR and IIR)

### **1.12 MRTC/fdct**

TO BE DISCUSSED: very similar to MRTC/fjdtint, strange code parts here (multiplications with constants)

### **1.13 MRTC/fft1**

KEEP: Different implementation from DSPstone (other loops, arrays instead of pointers)

### **1.14 MRTC/fir**

KEEP: One dimensional FIR, more complex than DSPstone\_fixed\_point/fir\_fixed

### 1.15 MRTC/fjdtint

KEEP: Discrete Cosine Transform, MRTC/fdct is very similar

### 1.16 MRTC/lms

KEEP: Least Mean Square approximation

## 2 Floating Point DSP Filters

- DSPstone\_floating\_point/convolution\_float
- DSPstone\_floating\_point/fir2dim\_float
- DSPstone\_floating\_point/fir\_float
- DSPstone\_floating\_point/iir\_biquad\_N\_sections\_float
- DSPstone\_floating\_point/iir\_biquad\_one\_section\_float
- DSPstone\_floating\_point/lms\_float
- StreamIt/filterbank

TO BE DISCUSSED: The DSPstone\_floating\_point programs are identical to the fixed variants, the only difference is a “#define TYPE float”. Shall we keep both versions or provide a mechanism to switch between integer and floating point? The floating point versions are the newer ones, therefore these should be taken.

### 2.1 StreamIt/filterbank

KEEP: Real floating point benchmark