SZA :: Signed Zero Algebra

A New Take On Propositional Logic

Thomas W Thorbjørnsen
Eirik Wittersø
Benjamin Benjaminsen
Torstein Nordgård-Hansen

07.04.2019

Contents

Chapter 1

Introduction

1.1 Preface

1.2 Motivation

As one usually does on a Saturday evening, we found ourselves suddenly engrossed in a vivid discussion about the societal value of the number zero. Whilst discussing the differences between null, zero, nil and naught, and their non-evident linguistic difference in Norwegian; a comment was made regarding the existence of both a positive and a negative zero in java. This proposition was by some found preposterous, and prompted an immediate investigation. Upon said investigation, its truthfulness was found evident, further bewildering the doubtful subjects regarding its arithmetical implementation. Addition, subtraction and multiplication was all applied, and the results organized and examined. What thenceforth emerged bore a striking resemblance to Boolean algebra, and the expansion and elaboration of said results is the subject of this paper.

Chapter 2

Finding the Algebraic Structure

2.1 Defining the Object

We will start off by defining all primitive objects. These definitions are a formalisation of how Java interprets the given symbols. We need to define the set which we will perform operations on, and the functions we can apply to this set. We define our set R as the following:

Definition 2.1.1.
R is the set with the elements {0, -0} i.e. $R = \{0, -0\}$

With the set it is a natural question to ask what we can do with this set. Java gave us the following functions which acts on the set. We will define:

Definition 2.1.2.
Let $+, -, \cdot$ be funtions such that $+, -, \cdot : R \times R \to R, \quad (a, b) \mapsto a(\bullet)b$.
These tables are a description of the complete function for each operation:

a	b	a+b
0	0	0
0	-0	0
-0	0	0
-0	-0	-0

a	b	a-b
0	0	0
0	-0	0
-0	0	-0
-0	-0	0

a	b	a · b
0	0	0
0	-0	-0
-0	0	-0
-0	-0	0

By inspecting the functions given by Java, we can see that they are all binary operations. We can also see that the "-" function is composite of $\cdot$ and $+$ (1). It is also possible to define "-" as a unary operation with $\cdot$ (2).

1. $- : R \times R \to R, (a, b) \mapsto a + (-0 \cdot b)$
2. $- : R \to R, a \mapsto -0 \cdot a$

In the rest of this paper, we will treat the function - as a unary function in (2).

2.2 The One and Only structure, $\mathbb{Z}_2$

In the last section we made a set with two functions acting on that set, namely addition and multiplication; and another function, -, defined with the multiplication function. In other words we can say that we have a set and two binary operations, which awfully looks like a ring. Due to the special circumstances of R, there is only one such ring: $\mathbb{Z}_2$.

The idea for this section of this paper is proving that $(r, \cdot, +) \simeq (\mathbb{Z}_2, +_2, \cdot_2)$. After that isomorphism is stated, the rules and behaviors of the ring should be clarified.

Proposition 2.2.1.
Let $R = \{0, -0\}$, and let $+$ and $\cdot$ be the functions defined in the last section, then $(R, \cdot, +)$ is a ring

Lemma 2.2.1.1.
Let R be a set with to binary functions; $+$, $\cdot$. R is a ring if R is isomorphic with a ring.

2.3 A Flashback to Boolean Algebra

2.4 Linking SZA to Boolean Algebra

Chapter 3

Building Logic Conjunctions

3.1 Primitive Conjunctions

3.2 Composite Conjunctions

3.3 How To Do Computations

3.3.1 From SZA to Modular Arithmetic

3.3.2 Proofs of given Composite Conjuctions

Chapter 4

An Application

4.1 Creating a Low Level Arithmetic Machine

4.1.1 Making a Full Bit Adder

4.1.2 Extending the Machine