# Preliminaries

A set of logical connectives is said to be functionally complete when combinations of the logical connectives in the set can express every equivalent expression of the connectives in the set of {⊥, ⊤, ¬, →, ↛, ⊕, ↔, ∧, ∨, ↓, ↑}.

{¬, →} {⊥, ⊤, ↛, ⊕, ↔, ∧, ∨}; {↓, ↑}

`{⊥, ⊤, ⊕, ↔}, {⊥, ↛, ⊕, ∧, ∨}, {⊤, ↔, ∧, ∨}, {⊥, ⊤, ∧, ∨}, {¬}`

{¬, →, ↛}{⊥, ⊤, ⊕, ↔, ∧, ∨} {↓, ↑}

A set of logical connectives is said to be functionally incomplete when combinations of the logical connectives can not express any functionally complete subset of {⊥, ⊤, ¬, →, ↛, ⊕, ↔, ∧, ∨, ↓, ↑}.

The minimal sets of functionally complete logical connectives entail that for defining the functionally incomplete sets the only relevant logical connectives are in the set {⊥, ⊤, ¬, →, ↛, ⊕, ↔, ∧, ∨}. Notably, any set that contains NOR or NAND is automatically functionally complete.

## Maximal Functionally Incomplete Sets

Affine: {⊥, ⊤, ¬, ⊕, ↔}; {→}, {↛}, {∧}, {∨}

False-preserving: {⊥, ↛, ⊕, ∧, ∨}; {⊤}, {¬}, {→}, {↔}

Truth-preserving: {⊤, →, ↔, ∧, ∨}; {⊥}, {¬}, {↛}, {⊕}

Monotonic: {⊥, ⊤, ∧, ∨}; {⊕}, {↔}, {¬}, {→}, {↛}

Self-dual: {¬} {⊥, ⊤, ⊕, ↔}; {→}, {↛}, {∧}, {∨}

## Minimal Functionally Complete Sets

### Singles

{↓}

{↑}

### Doubles

{⊥, →}

{⊥, ←}

{⊤, ↛}

{⊤, ↚}

{¬, →}

{¬, ←}

{¬, ↛}

{¬, ↚}

{¬, ∨}

{¬, ∧}

{→, ↛}

{→, ↚}

{←, ↛}

{←, ↚}

{→, ⊕}

{←, ⊕}

{↛, ↔}

{↚, ↔}

### Triples

{⊥, ↔, ∨}

{↔, ⊕, ∨}

{⊤, ⊕, ∨}

{⊥, ↔, ∧}

{↔, ⊕, ∧}

{⊤, ⊕, ∧}

# Functionally Incomplete Sets of Logical Connectives

## Singles

9 functionally incomplete singles.

{⊥}; {→}, {↔, ∧}, {↔, ∨}; ... {⊤, ¬, ↔, ⊕}, {↛, ⊕, ∨, ∧}

{⊤}; {↛}, {⊕, ∧}, {⊕, ∨}; ... {⊥, ¬, ↔, ⊕}, {→, ↔, ∧, ∨}

{¬}; {→}, {↛}, {∧}, {∨}; ... {⊥, ⊤, ⊕, ↔}

{→}; {⊥}, {¬}, {↛}, {⊕}; ... {⊤,↔, ∧, ∨}

{↛}; {⊤}, {¬}, {→}, {↔}; ... {⊥, ⊕, ∧, ∨}

{⊕}; {→}, {⊤, ∧}, {⊤, ∨}, {↔, ∧}, {↔, ∨}; ... {⊥, ↛, ∨, ∧}, {⊥, ⊤, ¬, ↔}

{↔}; {↛}, {⊥, ∧}, {⊥, ∨}, {⊕, ∧}, {⊕, ∨}; ... {⊤, →, ∧, ∨}, {⊥, ⊤, ¬, ⊕}

{∧}; {¬}, {⊥, ↔}, {⊤, ⊕}, {⊕, ↔}; {⊥}, {⊤}, {↔}, {⊕}, {∨}, {→}, {↛}, {∨, →}, {∨, ↛}, ..., {⊥, ⊤, ∨}, {⊤, →, ↔, ∨}, {⊥, ↛, ⊕, ∨}

{∨}; {¬}, {⊥, ↔}, {⊤, ⊕}, {⊕, ↔}; …, {⊥, ⊤, ∧}, {⊤, →, ∧, ↔}, {⊥, ↛, ⊕, ∧}

## Doubles

27 functionally incomplete doubles.

### Affine

{¬, ⊕}; {→}, {↛}, {∧}, {∨}; {⊥}, {⊤}, {↔}, {⊥, ⊤}, {⊥, ↔}, {⊤, ↔}, {⊥, ⊤, ↔}

{¬, ↔}; {→}, {↛}, {∧}, {∨}; {⊥}, {⊤}, {⊕}, {⊥, ⊤}, {⊥, ⊕}, {⊤, ⊕}, {⊥, ⊤, ⊕}

{⊥, ¬}; {→}, {↛}, {∧}, {∨}; {⊤}, {⊕}, {↔}, {⊤, ⊕}, {⊤, ↔}, {⊕, ↔}, {⊤, ⊕, ↔}

{⊥, ↔}; {→}, {↛}, {∧}, {∨}; {⊤}, {¬}, {⊕}, {⊤, ¬}, {⊤, ⊕}, {¬, ⊕}, {⊤, ¬, ⊕}

{⊤, ¬}; {→}, {↛}, {∧}, {∨}; {⊥}, {⊕}, {↔}, {⊥, ⊕}, {⊥, ↔}, {⊕, ↔}, {⊥, ⊕, ↔}

{⊤, ⊕}; {→}, {↛}, {∧}, {∨}; {⊥}, {¬}, {↔}, {⊥, ¬}, {⊥, ↔}, {¬, ↔}, {⊥, ¬, ↔}

{↔, ⊕}; {→}, {↛}, {∧}, {∨}; {⊥}, {⊤}, {¬}, {⊥, ⊤}, {⊥, ¬}, {⊤, ¬}, {⊥, ⊤, ¬}

### Affine or False Preserving

{⊥, ⊕}; {→}, {⊤, ∧}, {⊤, ∨}, {↔, ∧}, {↔, ∨}; … {⊤, ¬, ↔}, {↛, ∨, ∧}

### Affine or Truth Preserving

{⊤, ↔}; {↛}, {⊥, ∧}, {⊥, ∨}, {⊕, ∧}, {⊕, ∨}; … {⊥, ¬, ⊕}, {→, ∧, ∨}

### Affine and Monotonic

{⊥, ⊤}; {→}, {↛}, {↔, ∧}, {↔, ∨}, {⊕, ∧}, {⊕, ∨}; {¬}, {↔}, {⊕}, {∧}, {∨}, {∧, ∨}, {¬, ⊕}, {¬, ↔}, {⊕, ↔}, {¬, ⊕, ↔}

### Truth Preserving

#### Generators

{→, ∧}; {⊥}, {¬}, {↛}, {⊕}; … {⊤, ↔, ∨}

#### Non-generators

{⊤, →}; {⊥}, {¬}, {↛}, {⊕}; … {↔, ∧, ∨}

{→, ↔}; {⊥}, {¬}, {↛}, {⊕}; … {⊤, ∧, ∨}

{→, ∨}; {⊥}, {¬}, {↛}, {⊕}; … {⊤, ↔, ∧}

{↔, ∨}; {¬}, {↛}, {⊕}, {⊥}; … {⊤, →, ∧}

{↔, ∧}; {¬}, {↛}, {⊕}, {⊥}; … {⊤, →, ∨}

### False Preserving Generator

{↛, ∨}; {⊤}, {¬}, {→}, {↔}

### False Preserving

{⊕, ∧}; {¬}, {→}, {↔}, {⊤}; … {⊥, ↛, ∨}

{⊕, ∨}; {¬}, {→}, {↔}, {⊤}; … {⊥, ↛, ∧}

{⊥, ↛}; {⊤}, {¬}, {→}, {↔}; … {⊕, ∨, ∧}

{↛, ⊕}; {⊤}, {¬}, {→}, {↔}; … {⊥, ∨, ∧}

{↛, ∧}; {⊤}, {¬}, {→}, {↔}; … {⊥, ⊕, ∨}

### Monotonic and Truth Preserving

{⊤, ∨}; {¬}, {↛}, {⊕}, {⊥, ↔}; … {→, ↔, ∧}

{⊤, ∧}; {¬}, {↛}, {⊕}, {⊥, ↔}; … {→, ↔, ∨}

### Monotonic and False Preserving

{⊥, ∨}; {¬}, {→}, {↔}, {⊤, ⊕}; …, {↛, ⊕, ∧}

{⊥, ∧}; {¬}, {→}, {↔}, {⊤, ⊕}; …, {↛, ⊕, ∨}

### 

### Special

Truth or False Preserving

{∧, ∨}; {¬}, {⊥, ↔}, {⊤, ⊕}, {↔, ⊕}; … {⊥, ↛, ⊕}, {⊤, →, ↔}

## Triples

32 functionally incomplete triples.

### Affine Connectives

#### Generators

{⊥, ¬, ⊕}; {→}, {↛}, {∧}, {∨}; {⊤}, {↔}, {⊤, ↔}

{⊥, ¬, ↔}; {→}, {↛}, {∧}, {∨}; {⊤}, {⊕}, {⊤, ⊕}

{⊤, ¬, ⊕}; {→}, {↛}, {∧}, {∨}; {⊥}, {↔}, {⊥, ↔}

{⊤, ¬, ↔}; {→}, {↛}, {∧}, {∨}; {⊥}, {⊕}, {⊥, ⊕}

{¬, ⊕, ↔}; {→}, {↛}, {∧}, {∨}; {⊥}, {⊤}, {⊥, ⊤}

#### Non-generators

{⊥, ⊤, ¬}; {→}, {↛}, {∧}, {∨}; {⊕}, {↔}, {⊕, ↔}

{⊥, ⊤, ↔}; {→}, {↛}, {∧}, {∨}; {¬}, {⊕}, {¬, ⊕}

{⊥, ⊤, ⊕}; {→}, {↛}, {∧}, {∨}; {¬}, {↔}, {¬, ↔}

{⊥, ⊕, ↔}; {→}, {↛}, {∧}, {∨}; {⊤}, {¬}, {⊤, ¬}

{⊤, ⊕, ↔}; {→}, {↛}, {∧}, {∨}; {⊥}, {¬}, {⊥, ¬}

### Truth Preserving Connectives

#### Generators

{⊤, →, ∧}; {⊥}, {¬}, {↛}, {⊕}; {↔}, {∨}, {↔, ∨}

{→, ↔, ∧}; {⊥}, {¬}, {↛}, {⊕}; {⊤}, {∨}, {⊤, ∨}

{→, ∧, ∨}; {⊥}, {¬}, {↛}, {⊕}; {⊤}, {↔}, {⊤, ↔}

#### Non-generators

{⊤, →, ↔}; {⊥}, {¬}, {↛}, {⊕}; {∧}, {∨}, {∧, ∨}

{⊤, →, ∨}; {⊥}, {¬}, {↛}, {⊕}; {↔}, {∧}, {↔, ∧}  
{⊤, ↔, ∨}; {⊥}, {¬}, {↛}, {⊕}; {→}, {∧}, {→, ∧}  
{⊤, ↔, ∧}; {⊥}, {¬}, {↛}, {⊕}; {→}, {∨}, {→, ∨}  
{→, ↔, ∨}; {⊥}, {¬}, {↛}, {⊕}; {⊤}, {∧}, {⊤, ∧}  
{↔, ∧, ∨}; {⊥}, {¬}, {↛}, {⊕}; {⊤}, {→}, {⊤, →}

### False Preserving Connectives

{⊥, ↛, ⊕}; {⊤}, {¬}, {→}, {↔}; {∧}, {∨}, {∧, ∨}

{⊥, ⊕, ∨}; {⊤}, {¬}, {→}, {↔}; {↛}, {∧}, {↛, ∧}

{⊥, ⊕, ∧}; {⊤}, {¬}, {→}, {↔}; {↛}, {∨}, {↛, ∨}

{⊥, ↛, ∧}; {⊤}, {¬}, {→}, {↔}; {⊕}, {∨}, {⊕, ∨}

{⊥, ↛, ∨}; {⊤}, {¬}, {→}, {↔}; {∧}, {⊕}, {⊕, ∧}

{↛, ⊕, ∧}; {⊤}, {¬}, {→}, {↔}; {⊥}, {∨}, {⊥, ∨}

{↛, ⊕, ∨}; {⊤}, {¬}, {→}, {↔}; {⊥}, {∧}, {⊥, ∧}  
{↛, ∧, ∨}; {⊤}, {¬}, {→}, {↔}; {⊥}, {⊕}, {⊥, ⊕}

{⊕, ∧, ∨}; {⊤}, {¬}, {→}, {↔}; {⊥}, {↛}, {⊥, ↛}

### Exclusively Monotonic Connectives

{⊥, ⊤, ∨}; {¬}, {→}, {↛}, {⊕}, {↔}; {∧}

{⊥, ⊤, ∧}; {¬}, {→}, {↛}, {⊕}, {↔}; {∨}

### Monotonic Intersections

Monotonic and Truth Preserving

{⊥, ∧, ∨}; {¬}, {→}, {↔}; {⊤}, {↛}, {⊕}, {↛, ⊕}

Monotonic and False Preserving  
{⊤, ∧, ∨}; {¬}, {↛}, {⊕}; {⊥}, {→}, {↔}, {→, ↔}

## Quadruples

16 functionally incomplete quadruples.

### Affine Connectives

{⊥, ⊤, ⊕, ↔}; {→}, {↛}, {∧}, {∨}; {¬}

{¬, ⊤, ⊕, ↔}; {→}, {↛}, {∧}, {∨}; {⊥}

{¬, ⊥, ⊕, ↔}; {→}, {↛}, {∧}, {∨}; {⊤}

{¬, ⊥, ⊤, ↔}; {→}, {↛}, {∧}, {∨}; {⊕}

{¬, ⊥, ⊤, ⊕}; {→}, {↛}, {∧}, {∨}; {↔}

### Truth Preserving Connectives

#### Generators

{→, ⊤, ∧, ↔}; {⊥}, {¬}, {↛}, {⊕}; {∨}

{→, ∧, ∨, ↔}; {⊥}, {¬}, {↛}, {⊕}; {⊤}

{→, ⊤, ∧, ∨}; {⊥}, {¬}, {↛}, {⊕}; {↔}

#### Non-generators

{⊤, ∧, ∨, ↔}; {⊥}, {¬}, {↛}, {⊕}; {→}

{→, ⊤, ∨, ↔}; {⊥}, {¬}, {↛}, {⊕}; {∧}

### False Preserving Connectives

#### Generators

{⊥, ↛, ∨, ⊕}; {⊤}, {¬}, {→}, {↔}; {∧}

{⊥, ↛, ∧, ∨}; {⊤}, {¬}, {→}, {↔}; {⊕}

{↛, ∧, ∨, ⊕}; {⊤}, {¬}, {→}, {↔}; {⊥}

#### Non-generators

{⊥, ↛, ∧, ⊕}; {⊤}, {¬}, {→}, {↔}; {∨}

{⊥, ∧, ∨, ⊕}; {⊤}, {¬}, {→}, {↔}; {↛}

### Monotonic Connectives

{⊥, ⊤, ∧, ∨}; {¬}, {→}, {↛}, {⊕}, {↔}

## Quintuples

### Affine Connectives

{⊥, ⊤, ¬, ⊕, ↔}; {→}, {↛}, {∧}, {∨}

### Truth Preserving Connectives

{⊤, →, ↔, ∧, ∨}; {⊥}, {¬}, {↛}, {⊕}

### False Preserving Connectives

{⊥, ↛, ⊕, ∨, ∧}; {⊤}, {¬}, {→}, {↔}

## Sextuples and higher

All subsets of {⊥, ⊤, ¬, →, ↛, ⊕, ↔, ∧, ∨} of cardinality 6 or higher are functionally complete.

Proof: {⊥, ⊤, ⊕, ↔, ∧, ∨} and {¬, →, ↛} are functionally complete. Every 5 element subset of {⊥, ⊤, ⊕, ↔, ∧, ∨} is functionally complete; there exist 4 element subsets which are functionally incomplete, but the only way to get six element sets from those 4 element sets is to add two elements from {¬, →, ↛} and any two elements of {¬, →, ↛} are functionally complete together.

## Special Cases of functional incomplete sets of operators

{⊥, ⊤, ⊕, ↔} ⊬ {¬}

{⊤, →, ↔, ∨} ⊬ {∨}

{⊥, ↛, ⊕, ∧} ⊬ {∧}

{¬, ⊕} ⊢ {⊥, ⊤, ¬, ⊕, ↔}

{¬, ↔} ⊢ {⊥, ⊤, ¬, ⊕, ↔}

{→, ∨} ⊢ {⊤, →, ∨}

{→, ∧} ⊢ {⊤, →, ↔, ∧}

{↛, ∨} ⊢ {⊥, ↛, ⊕, ∨}

{⊕, ↔} ⊢ {⊥, ⊤, ⊕, ↔}

{→, ↔} ⊢ {⊤, →, ↔}

{↛, ⊕} ⊢ {⊥, ↛, ⊕}

{↛}⊢{⊥, ↛}

{⊕}⊢{⊥, ⊕}

{→}⊢{⊤,→}

{↔}⊢{⊤, ↔}

# Functionally Complete Sets

There are 512 subsets of {⊥, ⊤, ¬, →, ↛, ⊕, ↔, ∧, ∨}. 425 subsets are functionally complete.

There are 2048 subsets of {⊥, ⊤, ¬, →, ↛, ⊕, ↔, ∧, ∨, ↓, ↑}. 1961 subsets are functionally complete.

NUMERICAL ENCODING! Based on the MAXIMALLY INCOMPLETE SETS

Class 1: truth-preserving 10000  
Class 2: false-preserving 01000  
Class 3: affine 00100

Class 4: monotone 00010

Class 5: self-dual 00001

Any connective can be expressed uniquely as a binary numeral.  
AND is truth preserving. 10010

{⊥, ⊤, ¬, ⊕, ↔} Affine set

{⊥, ↛, ⊕, ∨, ∧} False Preserving set

{⊤, →, ↔, ∧, ∨} Truth Preserving set

{⊥, ⊤, ∧, ∨} Monotonic set

{¬} Self-Dual set

LIST OF NUMERICAL ENCODINGS!

⊥: false preserving, affine. 01100

⊤: truth preserving, affine. 10100

¬: self-dual: 00001, affine: 00100. 00101

→ Truth Preserving 10000

↛ False Preserving 01000

⊕ False Preserving, Affine 01100

↔ Truth Preserving, Affine 10100

∧ False Preserving, Truth Preserving, Monotonic 11010

∨ False Preserving, Truth Preserving, Monotonic 11010

↓: 00000

↑: 00000

PROVING WHICH SETS ARE FUNCTIONALLY COMPLETE

A functionally complete set is defined by the property that when all its members are summed (with bitwise OR), the sum is 00000.

{↓}: 00000

{↑}: 00000

{⊥, →} : 01100 || 10100 = 00100

AND: AB

OR: 1 - (1 - A)(1 - B) = A + B - AB

NAND: 1 - AB

NOR: 1 - A - B + AB

IMPLIES: 1 - A + AB

NONIMPLIES: A(1-B)