

Московский физико-технический институт (государственный университет)

Физтех-школа радиотехники и компьютерных технологий

Кафедра информатики и вычислительной техники

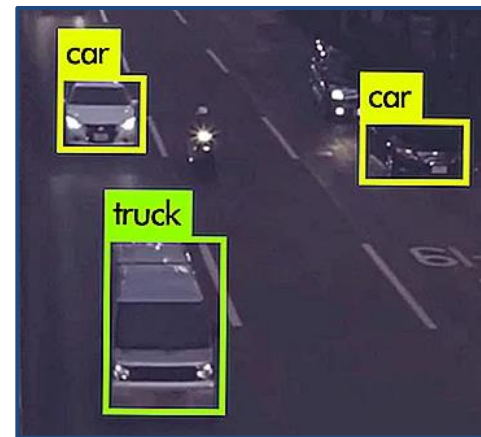
# **СРАВНЕНИЕ АППАРАТНЫХ РЕАЛИЗАЦИЙ ОПЕРАЦИИ СВЁРТКИ ДЛЯ КЛАССИЧЕСКОЙ И БИПОЛЯРНОЙ МОРФОЛОГИЧЕСКОЙ МОДЕЛЕЙ СВЁРТОЧНОЙ НЕЙРОННОЙ СЕТИ**

Выпускная квалификационная работа  
(магистерская диссертация)

Студент:	Цой М. О.
Научный руководитель:	д.т.н. Фельдман В. М.
Научный консультант:	Альфонсо Д. М.

Москва, 2022

- Используются для распознавания образов на изображениях
- Входные данные – пиксели обрабатываемого изображения
- Коэффициенты упакованы в матрицы, называемые фильтрами



### Режимы работы

#### Обучение

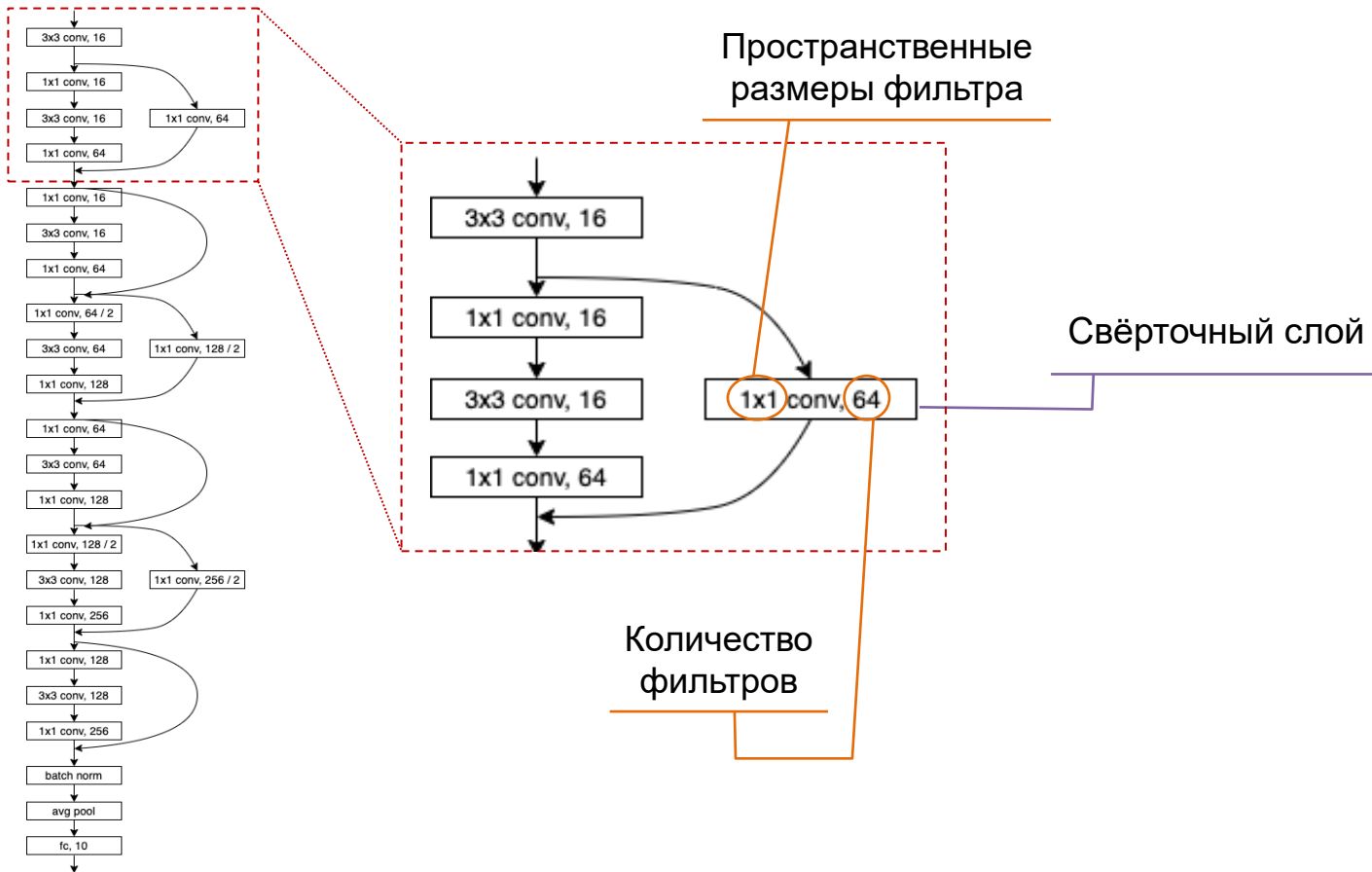
- Преобразование размеченных входных данных с помощью набора коэффициентов
- Корректировка коэффициентов

#### Применение (*inference*)

- Преобразование входных данных с помощью набора коэффициентов

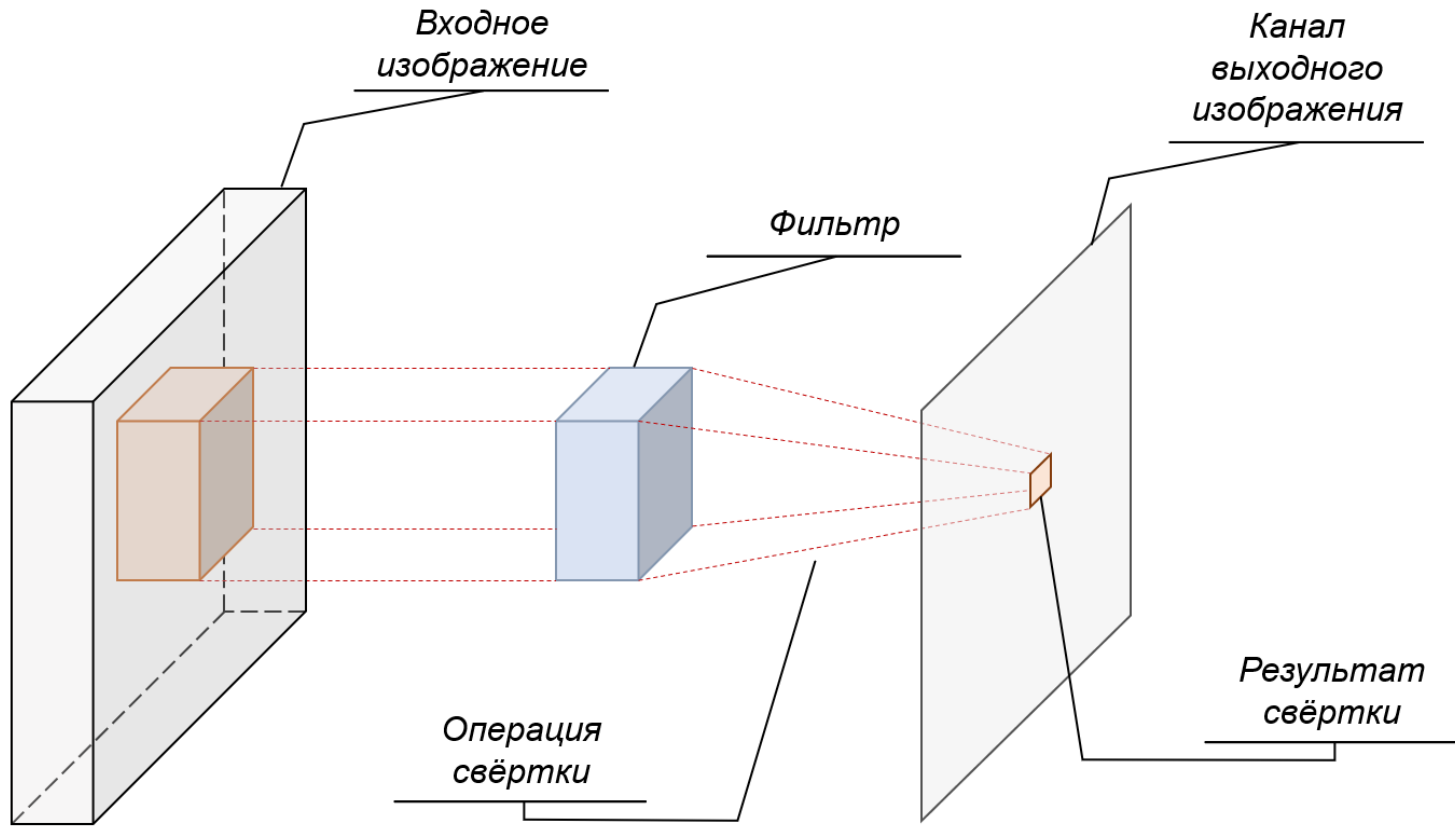
## Архитектура свёрточных нейронных сетей

Архитектура  
нейросети  
ResNet-22



# Введение

## Принцип работы свёрточного слоя



### Формула операции наложения фильтра:

$$(I * w)(x, y) = \sum_{c=0}^{C-1} \sum_{\Delta x=-h_x}^{h_x} \sum_{\Delta y=-h_y}^{h_y} I(x + \Delta x, y + \Delta y, c) \cdot w(h_x + \Delta x, h_y + \Delta y, c)$$

- $I$  – изображение
- $w$  – фильтр
- $C$  – количество каналов изображения/фильтра
- $(h_x, h_y)$  – координаты центрального пикселя канала фильтра

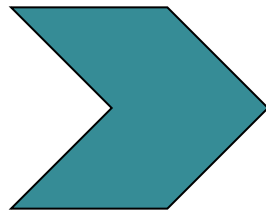
### Формула классической свёртки:

$$I_{n+1}(x, y, f) = \sigma \left( (I_n * w_f)(x, y) + b_f \right)$$

- $f$  – номер фильтра / номер канала выходного изображения
- $w_f$  – фильтр
- $b_f$  – смещение
- $\sigma$  – функция активации

Обработка одного изображения  
**>  $10^8$  операций сложения и  
умножения**

**Операции умножения требуют  
высоких аппаратных затрат**



*Использование альтернативных  
моделей вычисления*



**Биполярная морфологическая модель**  
Отказ от умножений за счёт  
использования экспонент и логарифмов

## Формулы биполярной морфологической модели

Формула операции наложения фильтра:

$$(I \odot w)(x, y) = \max_{c=0, \dots, C-1} \max_{\Delta x = -h_x, \dots, h_x} \max_{\Delta y = -h_y, \dots, h_y} (I(x + \Delta x, y + \Delta y, c) + \mathbf{w}(h_x + \Delta x, h_y + \Delta y, c))$$

Формула биполярной морфологической свёртки:

$$I_{n+1}(x, y, f) = \sigma \left[ \sum_{\alpha \in \{+, -\}} \sum_{\beta \in \{+, -\}} p^\alpha \cdot p^\beta \cdot 2^{(\log_2 I_n^\alpha \odot w_f^\beta)(x, y)} + b_f \right]$$

$$p^+ = +1 \quad I^+(x, y, c) = \begin{cases} I(x, y, c) & , \text{if } I(x, y, c) > 0 \\ 0 & , \text{if } I(x, y, c) \leq 0 \end{cases}$$

$$p^- = -1 \quad I^-(x, y, c) = \begin{cases} -I(x, y, c) & , \text{if } I(x, y, c) < 0 \\ 0 & , \text{if } I(x, y, c) \geq 0 \end{cases}$$

**Цель работы:** провести сравнение классической и биполярной морфологической моделей с точки зрения аппаратных затрат и скорости работы.

## **Задачи:**

1. Спроектировать модуль для вычисления классической свёртки и разработать его RTL-описание
2. Спроектировать модуль для вычисления биполярной морфологической свёртки и разработать его RTL-описание
3. Провести синтез разработанных модулей
4. Провести оценку времени работы разработанных модулей для нейросетей с архитектурами ResNet-22 и VGG-16

## **Требования:**

- Разработанные модули должны иметь возможность проводить свёртку в двух режимах: с фильтрами  $3 \times 3$  и  $1 \times 1$
- Вычисление свёртки в модулях должно выполняться над данными, представленными в формате Single IEEE754
- В качестве функции активации использовать функцию ReLU, которая зануляет отрицательные значения и оставляет без изменений положительные

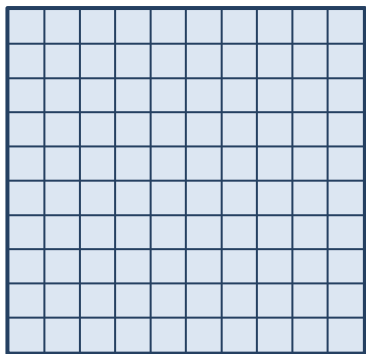


# Вычислительный модуль классической модели

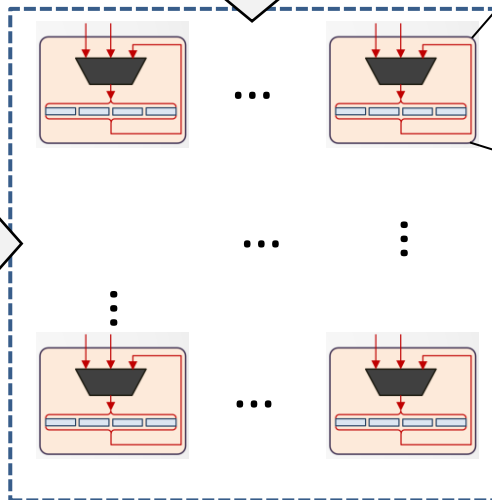
## Реализация вычислительного модуля

- Размер фрагмента входного изображения –  $m \times n$
- Количество вычислительных устройств (FMA) –  $m \times (n/4)$

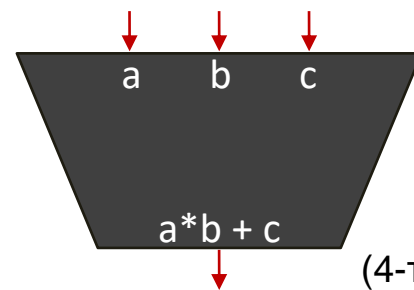
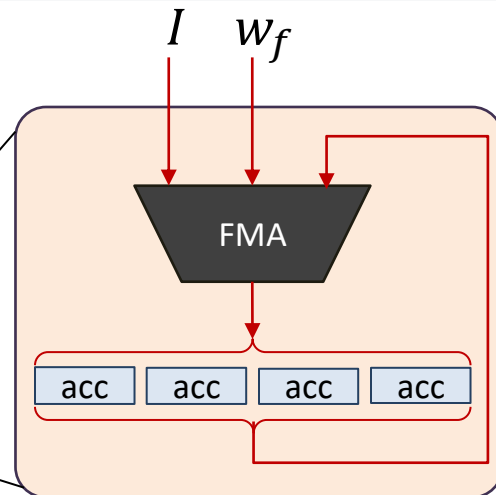
Фрагмент изображения  
(1 канал)



Фильтр (1 канал)



Вычислительный модуль



**FMA**  
**Fused**  
**Multiply-Add**





(4-тактный конвейер)

## Особенности архитектуры вычислительного модуля

Формула биполярной морфологической свёртки:

$$I_{n+1}(x, y, f) = \sigma \left[ \sum_{\alpha \in \{+, -\}} \sum_{\beta \in \{+, -\}} p^\alpha \cdot p^\beta \cdot 2^{(\log_2 I_n^\alpha \odot w_f^\beta)(x, y)} + b_f \right]$$

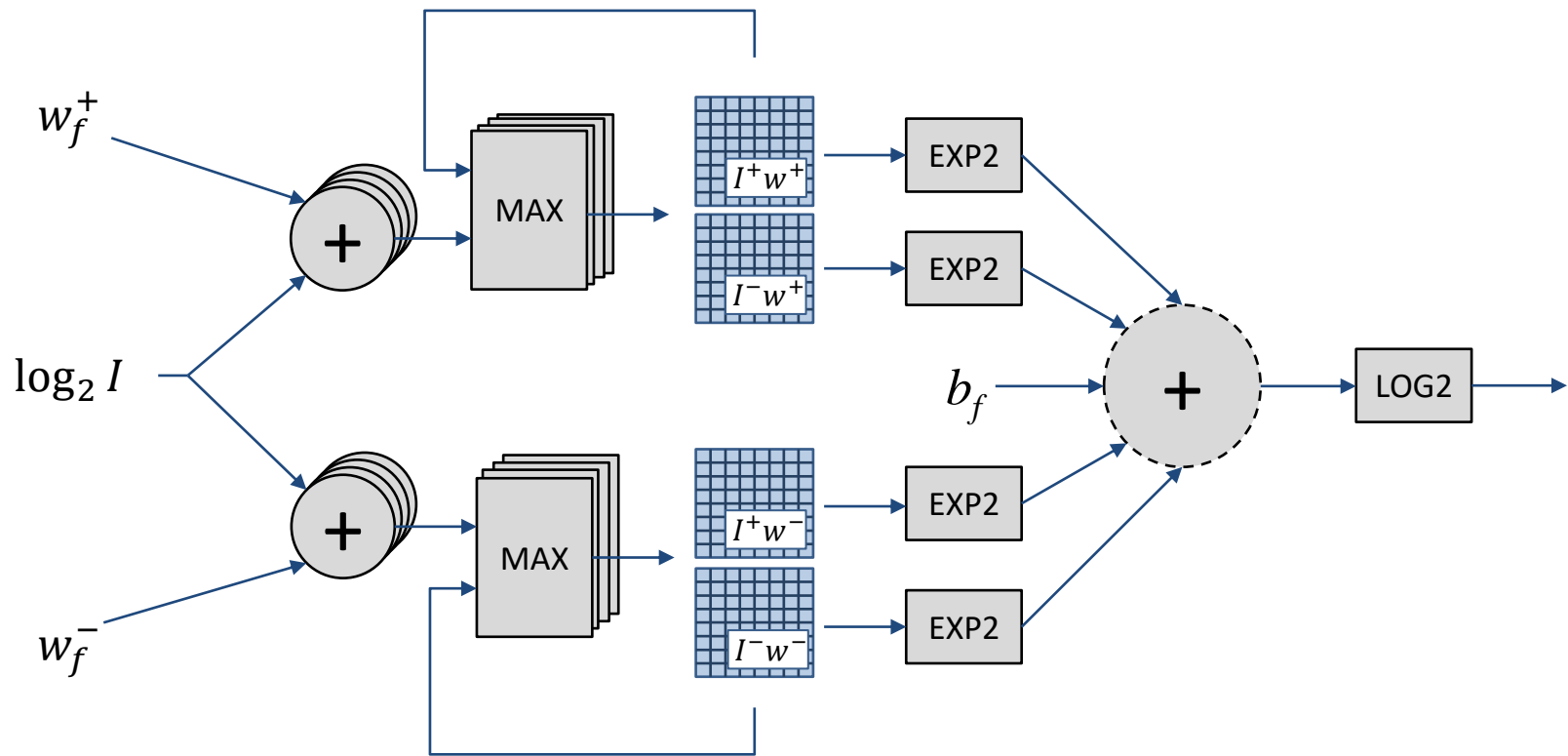
Алгоритм вычисления:

- 1) Вычислить логарифм от изображения 
- 2) Наложить фильтр 4 раза 
- 3) Вычислить экспоненты 
- 4) Вычислить сумму экспонент и смещения 
- 5) Применить функцию активации к результату

*bottleneck*

Особенности архитектурных решений:

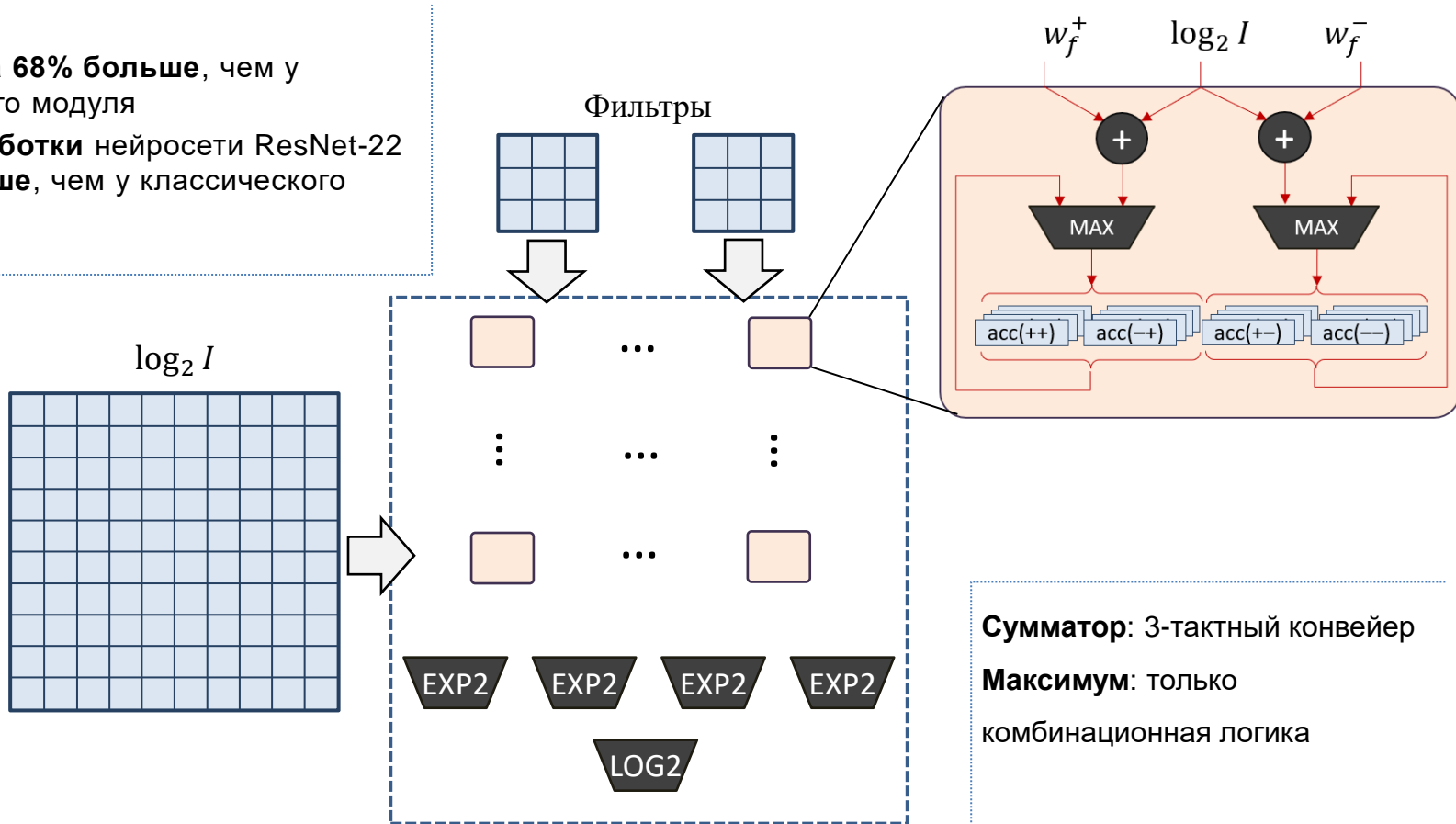
- Одно устройство логарифма, но перенос операции в конец алгоритма
- 4 массива аккумуляторов
- 4 устройства экспоненты
- Переиспользование сумматоров, участвующих в наложении фильтра



# Базовый вычислительный модуль БМ-модели

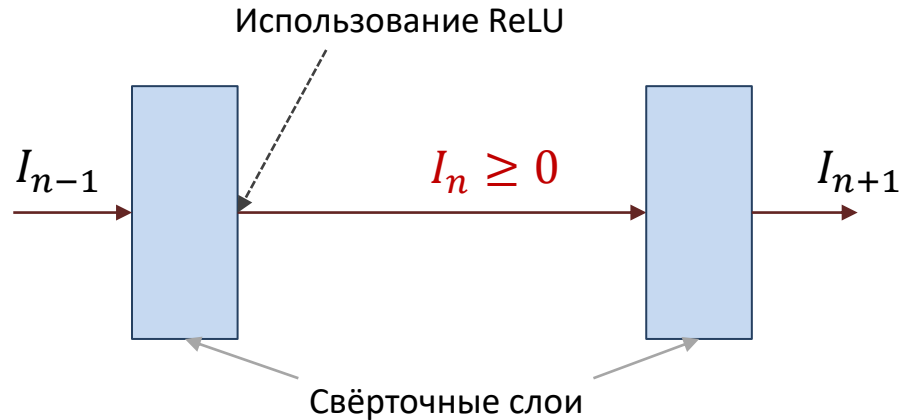
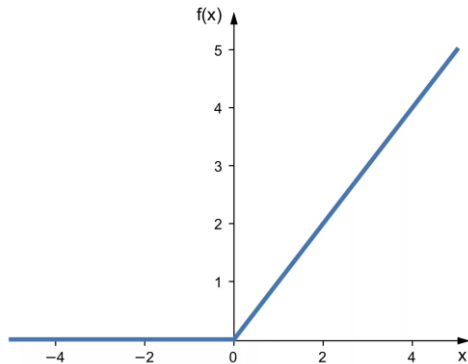
## Реализация базовой версии вычислительного модуля

- **Площадь на 68% больше**, чем у классического модуля
- **Время обработки** нейросети ResNet-22 **на 2% больше**, чем у классического модуля



## Устранение двух ветвей вычислений

Функция активации  $\sigma$  – ReLU:



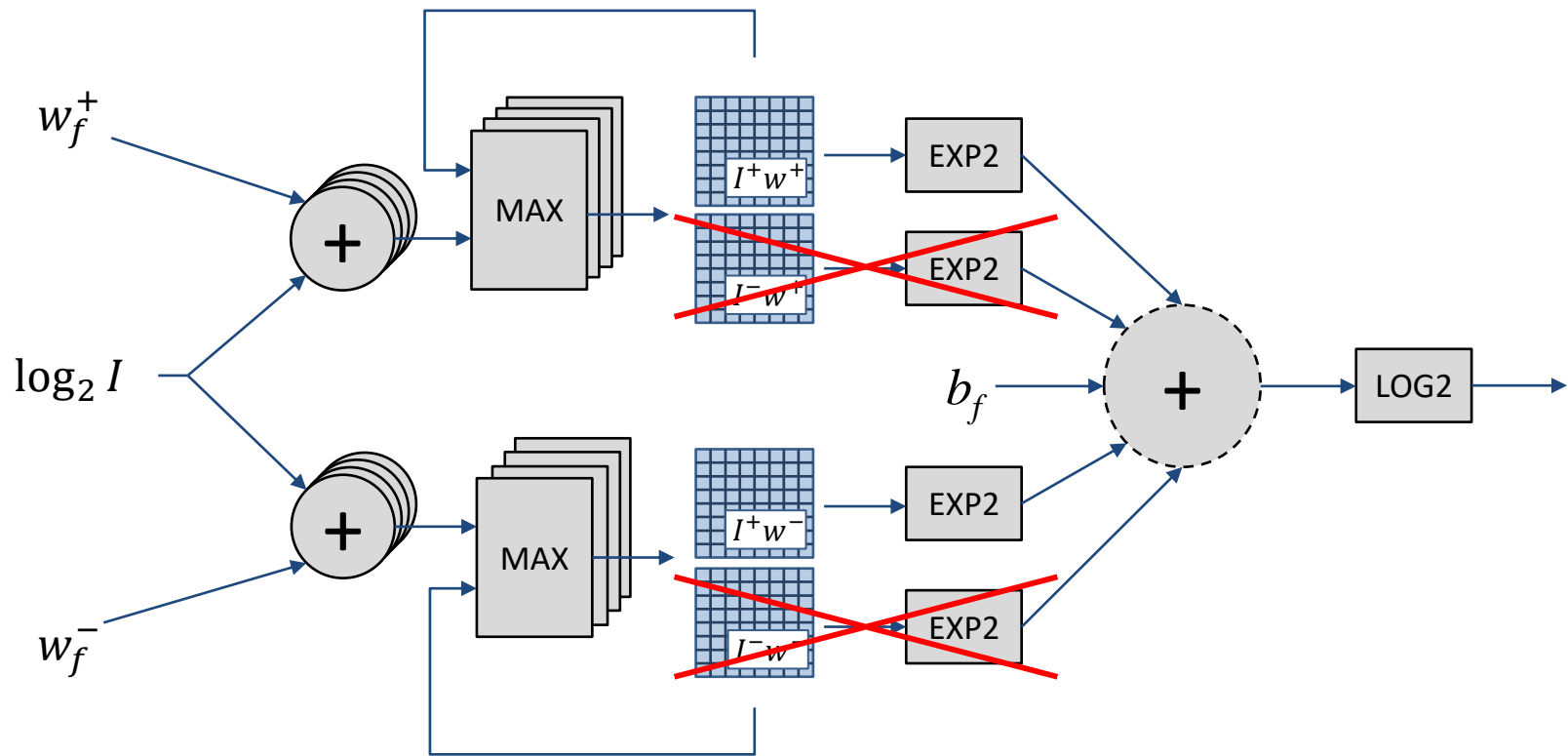
Упрощение формулы свёрточного слоя:

$$I_{n+1}(x, y, f) = \sigma \left[ 2^{\left( \log_2 I_n \odot w_f^+ \right)(x, y)} - 2^{\left( \log_2 I_n \odot w_f^- \right)(x, y)} + b_f \right]$$

# Модифицированный вычислительный модуль БМ-модели

Устранение двух ветвей вычислений

14



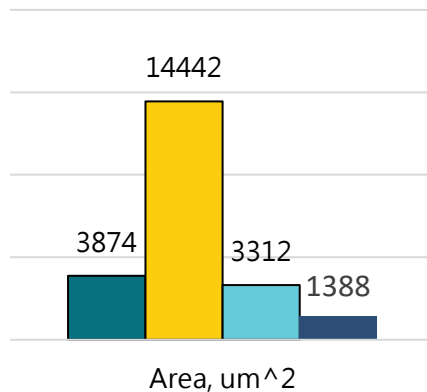
### LOG2

- Аппроксимация полиномом 5-ой степени
- 16-тактное устройство

### EXP2

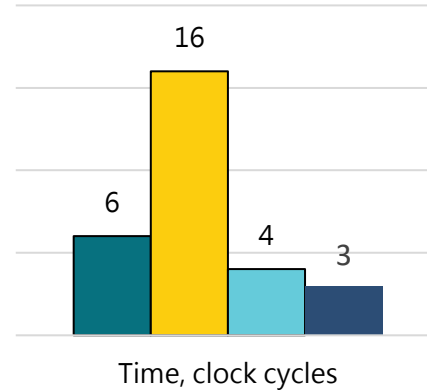
- Кусочно-квадратичная аппроксимация
- 6-тактное устройство

### Площадь



■ EXP2 ■ LOG2 ■ FMA ■ Adder

### Время вычисления



■ EXP2 ■ LOG2 ■ FMA ■ Adder

Для минимизации аппаратных затрат были реализованы:

- **кусочно-линейная аппроксимация** для вычисления экспоненты

	Кусочно-квадратичная аппроксимация	Кусочно-линейная аппроксимация
Площадь, $\mu m^2$	3874	259 (-93%)
Время работы, такты	6	1

- **аппроксимация Митчелла** для вычисления логарифма

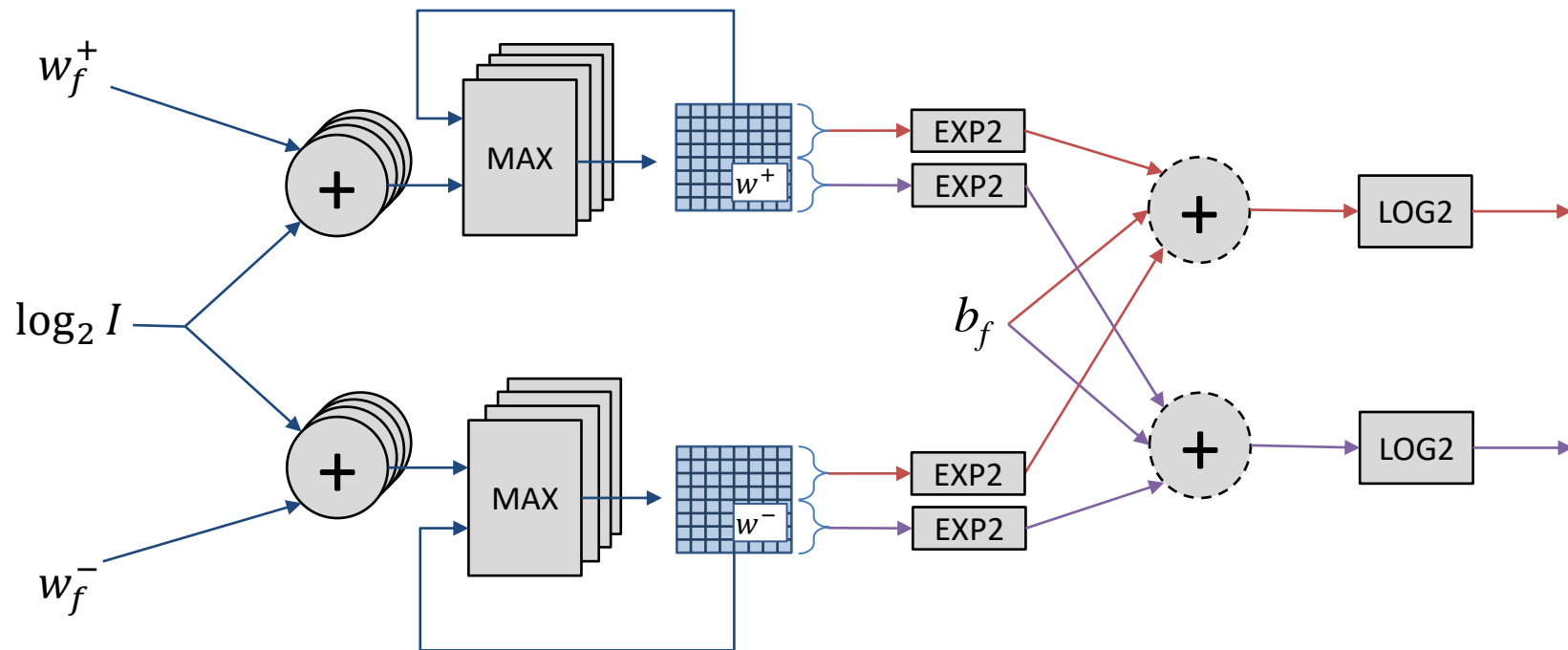
	Аппроксимация полиномом 5-ой степени	Аппроксимация Митчелла
Площадь, $\mu m^2$	14442	290 (-98%)
Время работы, такты	16	2



# Модифицированный вычислительный модуль БМ-модели

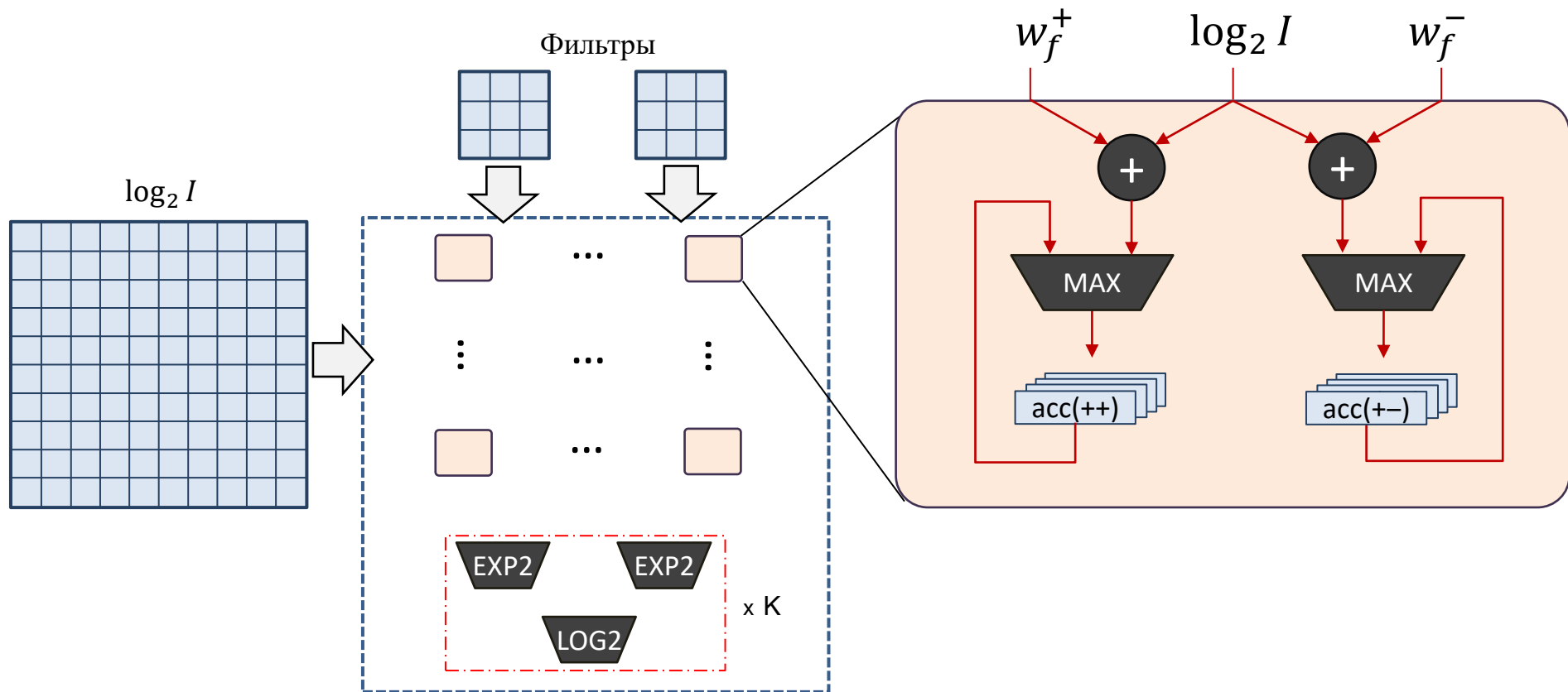
Добавление устройств экспоненты и логарифма

17



# Модифицированный вычислительный модуль БМ-модели

Реализация модифицированного вычислительного модуля



# Сравнение характеристик

## Результаты синтеза и оценки времени работы

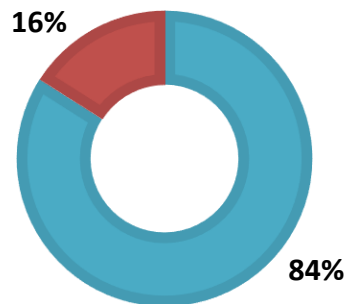
$m \times n = 8 \times 8$

	Классическая модель	БМ-модель			
		K = 8	K = 4	K = 2	K = 1
Площадь, $\mu m^2$	63553	70796 (+11%)	68393 (+8%)	66802 (+5%)	65607 (+3%)
Время обработки сети ResNet-22, $\mu s$	5457	5542 (+1.5%)	5615 (+3%)	5763 (+5%)	6058 (+11%)
Время обработки сети VGG-16, $ms$	1164	1136 (-2.4%)	1138 (-2.2%)	1142 (-1.9%)	1148 (-1.3%)

$m \times n = 16 \times 8$

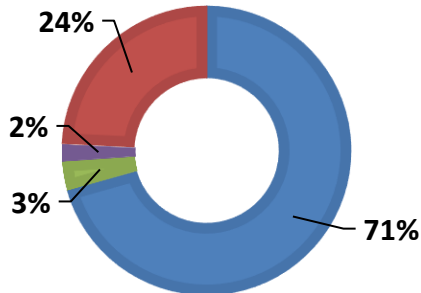
	Классическая модель	БМ-модель			
		K = 8	K = 4	K = 2	K = 1
Площадь, $\mu m^2$	125721	125785 (+0%)	122185 (-2.8%)	120861 (-3.9%)	120087 (-4.5%)
Время обработки сети ResNet-22, $\mu s$	3884	3986 (+2.6%)	4072 (+4.8%)	4242 (+9.3%)	4588 (+18.1%)
Время обработки сети VGG-16, $ms$	604	591 (-2.2%)	593 (-1.9%)	596 (-1.3%)	603 (-0.1%)

### Классический модуль



■ FMA

### Модифицированный БМ-модуль



■ Adder

■ EXP

■ LOG

### ■ Остальное:

- Массив аккумуляторов
- Коммутация устройств
- Логика управляющих сигналов

- **Разработано RTL-описание** вычислительного модуля для **классической модели** нейронной сети
- **Разработано RTL-описание** вычислительных модулей для **биполярной морфологической модели** нейронной сети с базовой и модифицированной архитектурой
- **Проведён синтез разработанных модулей:** базовый БМ-модуль даёт проигрыш в 68%, модифицированный – выигрыш до 4.5% по площади в сравнении с классическим вычислительным модулем
- **Проведена оценка времени обчёта ResNet-22:** модифицированный БМ-модуль уступает на 1.5% в скорости обчёта классическому модулю
- **Проведена оценка времени обчёта VGG-16:** модифицированный БМ-модуль обгоняет на 2.4% в скорости обчёта классическому модулю
- Работа представлена на международной конференции **IEEE Engineering & Telecommunication 2021**