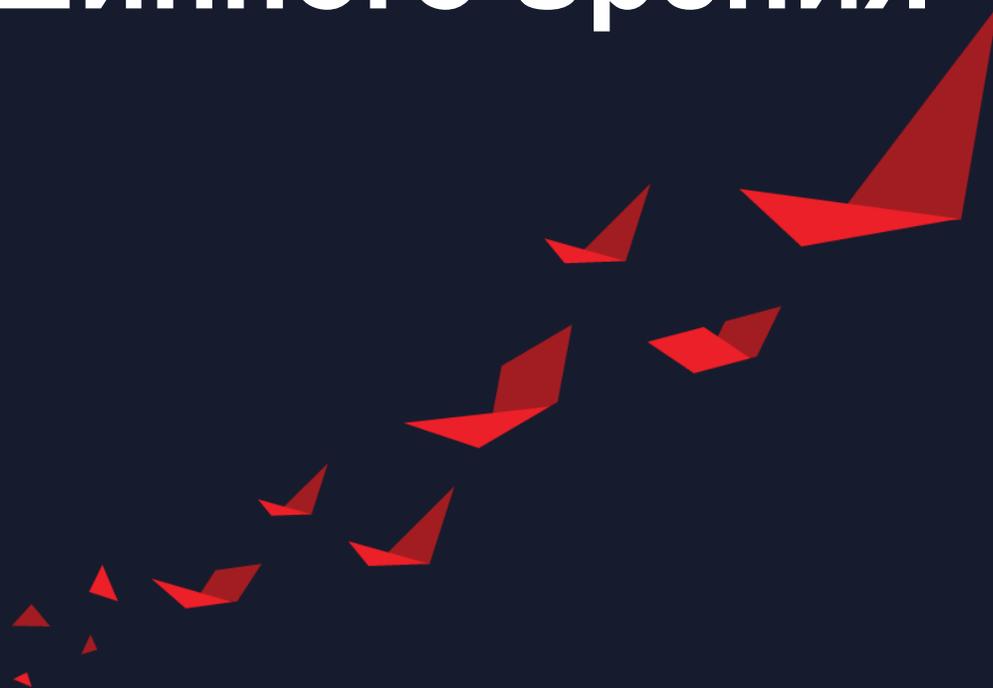


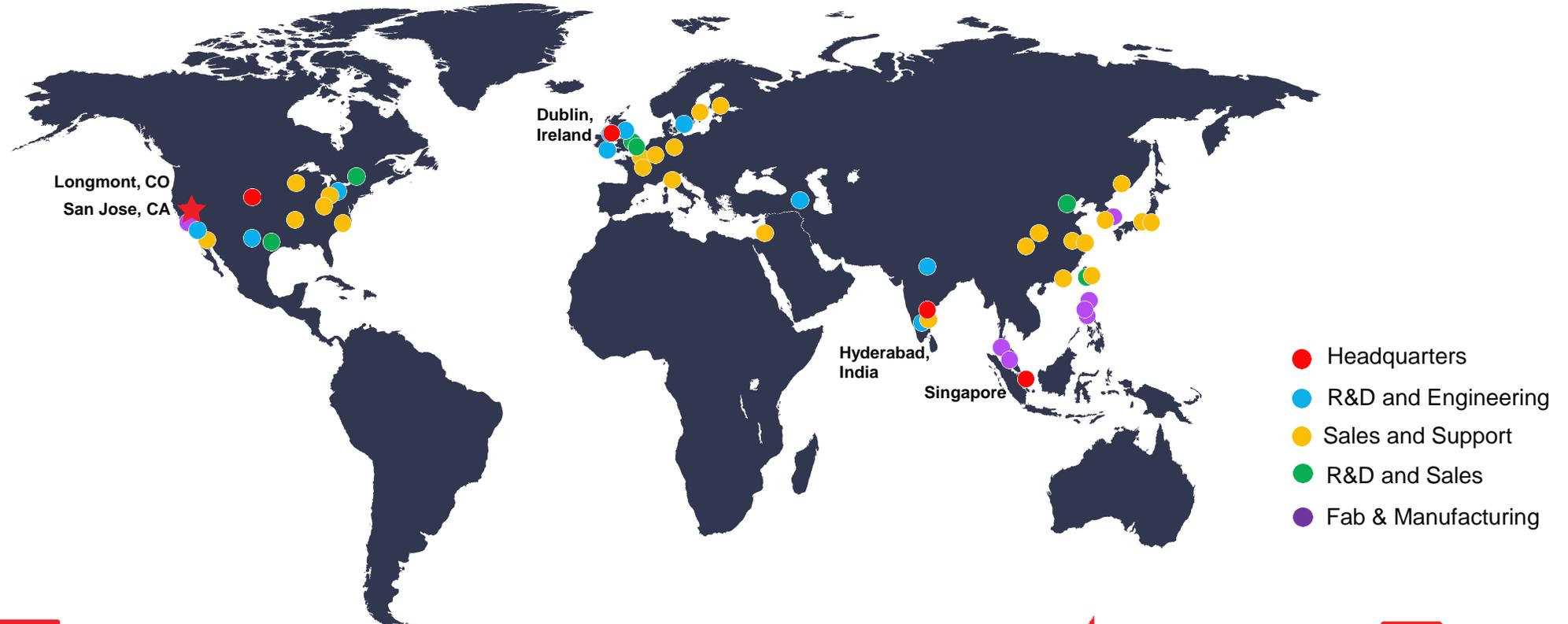


Решения Xilinx для Машинного Зрения



Xilinx

Компания основана в 1984



\$3.06B

Прибыль
(FY19)



5,000

Сотрудников по
всему миру



60K+

Заказчиков



60+

Нововведений



4,400+

Патентов

Применения в индустриальной сфере и в областях, связанных с видео



Robotics



Drives & Motor Control



IIoT Gateways & Edge Appliances



PLC/PAC/IPC



I/O Modules & Smart Sensors



Human Machine Interface



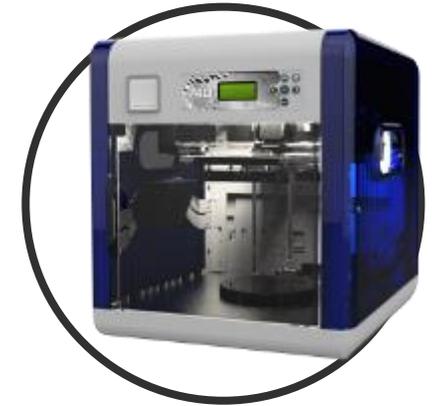
Video Surveillance & Smart City



Machine & Computer Vision



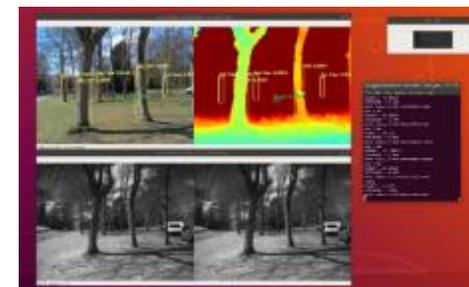
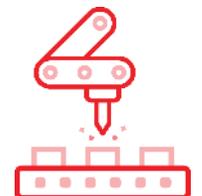
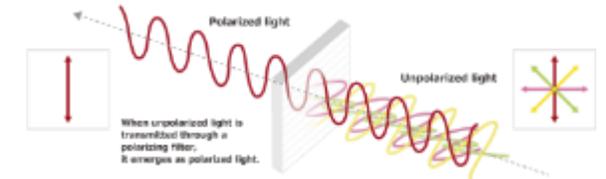
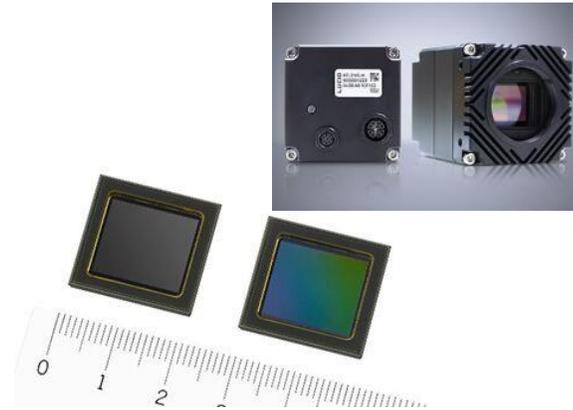
Smart Grid



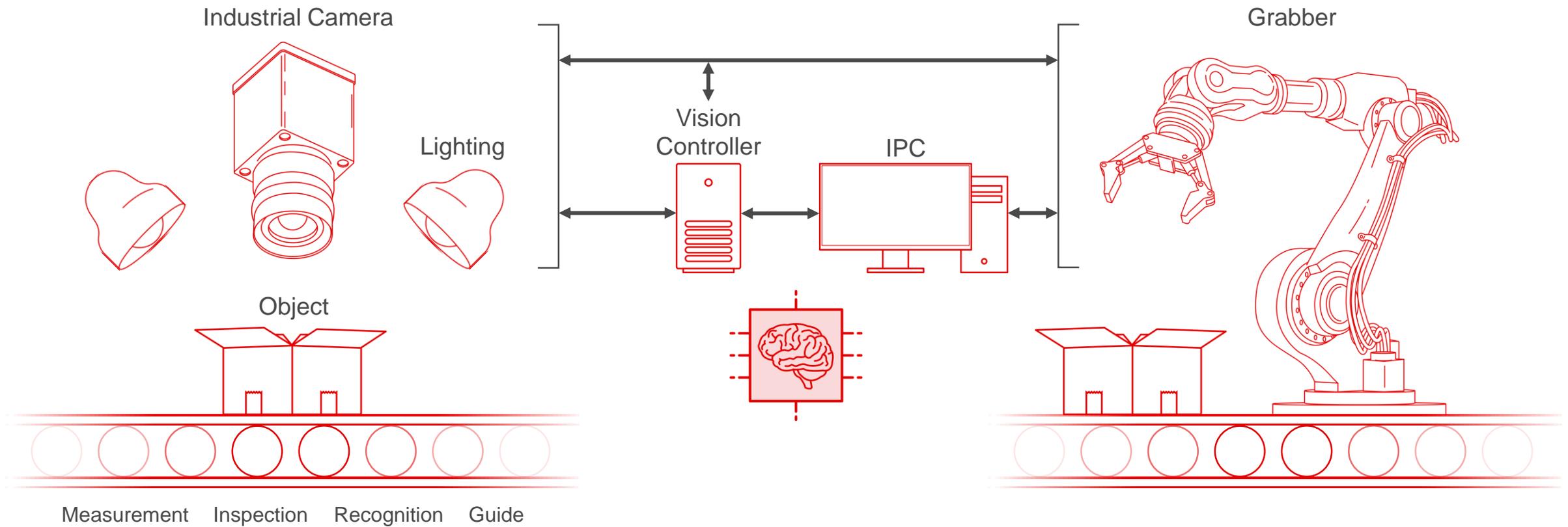
3D Printing & Additive Manufacturing

Тренды в области Машинного Зрения

- ▶ Более высокое разрешение и более высокая частота кадров =
Высокоскоростные интерфейсы
- ▶ PCs/Платы видеозахвата → “Умные”
камеры для машинного зрения
- ▶ 3D, Поляризация, Мультиспектральная и
Гиперспектральная визуализация
- ▶ Бурное развитие в области ИИ и
Машинного обучения (и соответствующего
ПО для обработки)
- ▶ Появление специфичных сфер для
применения Машинного Зрения
(малообъемных, узкоспециализированных)
- ▶ Уменьшение энергопотребления и
размеров устройств



Машинное зрение требует сверхнизкой задержки ИИ





Машинное Зрение: Микросхемы и платформы Xilinx

Семейства устройств Xilinx:

SoC Portfolio

Zynq US+ Optimized for Today's Embedded Cameras

Cost-Optimized → Mid-Range

Mid-Range → High-End

ZYNQ[®]
Zynq-7000

SoC

System Optimized for Power & Cost

ZYNQ[®]
UltraSCALE+

Heterogeneous MPSoC

Right Engines for the Right Tasks

Artix 7: Historic Leader in Machine Vision

Kintex Family for High End Machine Vision

Cost-Optimized

SPARTAN⁶
SPARTAN⁷
ARTIX⁷

Lowest Power & Cost

Mid-Range

KINTEX[®]

Price/Performance/Watt

High-End

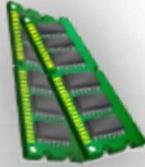
VIRTEX[®]

Performance & Capacity

Серия Zynq UltraScale+ MPSoC Xilinx

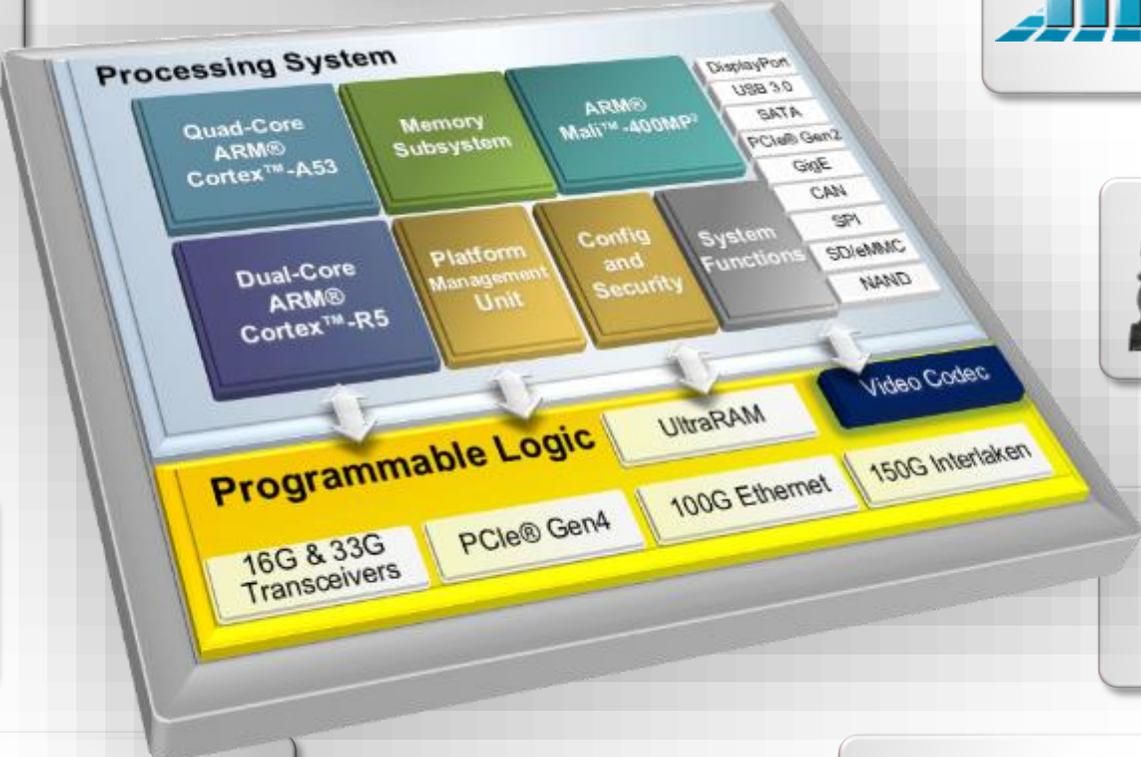
Machine Vision Recommended

ARM[®] Real-Time Processors
Cortex[®] R5 32-bit Dual-Core

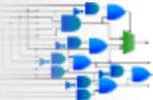
**Memory Subsystem**
High Bandwidth
Low Latency

**Graphics Processor**
ARM Mali-400MP2

ARM[®] Application Processor
Cortex[®] A53 64-bit Dual/Quad-Core



**High Speed Peripherals**
Key Interfaces

**Fabric Acceleration**
Customizable Engines
High Speed Connectivity

**H.265 HEVC**
High Efficiency Video Coding
H.264 MPEG-4/AVC
Video Codec
8K4K (15fps)
4K2K (60fps)

**Platform & Power Management**
Granular Power Control
Functional Safety

**Configuration & Security Unit**
Anti-Tamper & Trust
Industry Standards

Устройства Zynq UltraScale+ MPSoC Xilinx EV со встроенным аппаратным декодом H.264 / H.265

- > Многопоточное кодирование с несколькими разрешениями
- > 4k@60 кодирование
- > Интерфейсы для подключения сенсоров: MIPI, SLVS, SLVS-EC, USB, HDMI, DP, LVDS, TSN

Возможные применения:

- > Мульти-сенсорные камеры
- > Умный город
- > Системы машинного зрения и др.

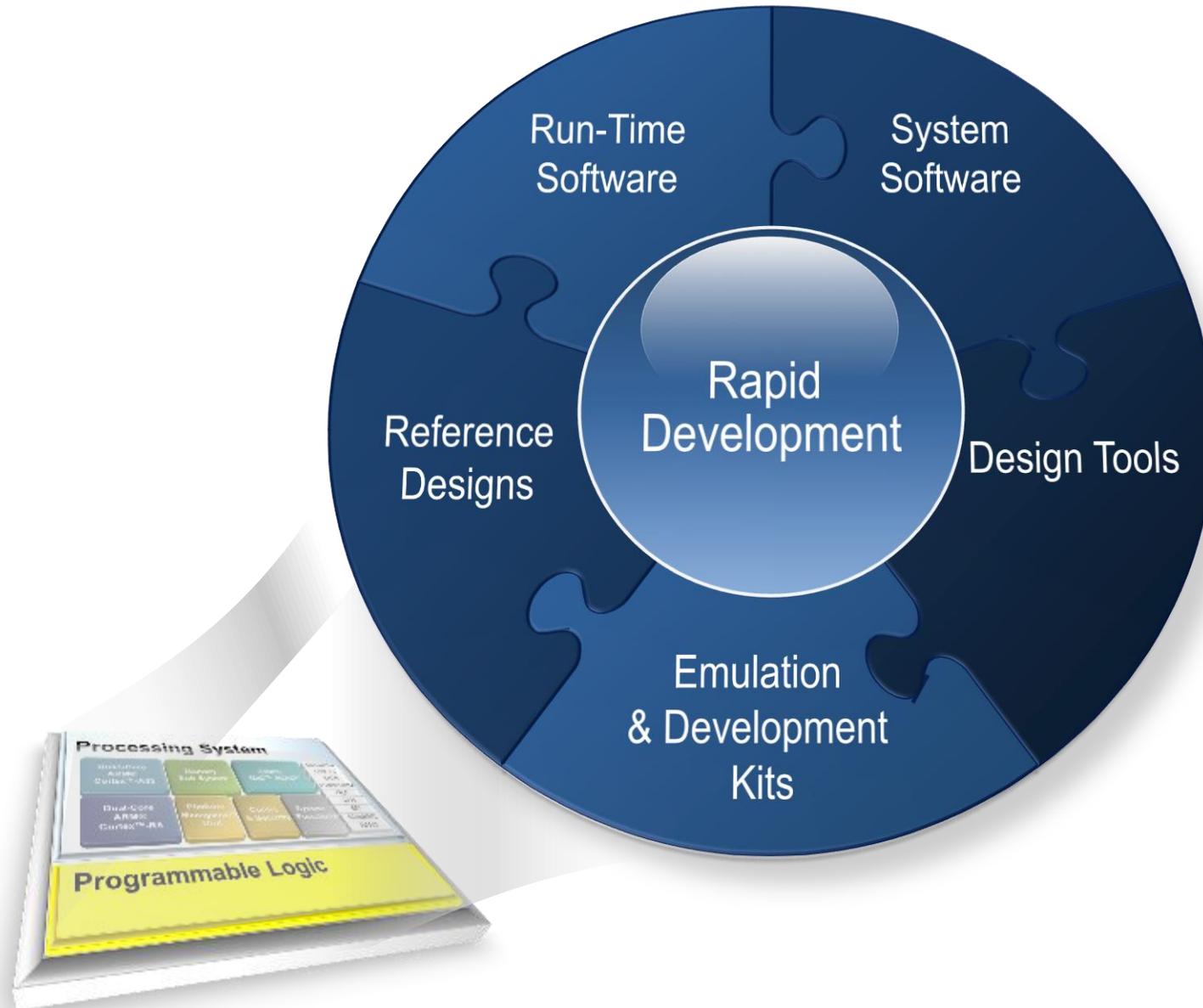
Zynq® UltraScale+™ MPSoCs: EV Devices

		Device Name ⁽¹⁾	ZU4EV	ZU5EV	ZU7EV
Processing System (PS)	Application Processor Unit	Processor Core	Quad-core ARM® Cortex™-A53 MPCore™ up to 1.5GHz		
		Memory w/ECC	L1 Cache 32KB I / D per core, L2 Cache 1MB, on-chip Memory 256KB		
	Real-Time Processor Unit	Processor Core	Dual-core ARM Cortex-R5 MPCore™ up to 600MHz		
		Memory w/ECC	L1 Cache 32KB I / D per core, Tightly Coupled Memory 128KB per core		
	Graphic & Video Acceleration	Graphics Processing Unit	Mali™-400 MP2 up to 667MHz		
		Memory	L2 Cache 64KB		
	External Memory	Dynamic Memory Interface	x32/x64: DDR4, LPDDR4, DDR3, DDR3L, LPDDR3 with ECC		
		Static Memory Interfaces	NAND, 2x Quad-SPI		
	Connectivity	High-Speed Connectivity	PCIe® Gen2 x4, 2x USB3.0, SATA 3.1, DisplayPort, 4x Tri-mode Gigabit Ethernet		
		General Connectivity	2xUSB 2.0, 2x SD/SDIO, 2x UART, 2x CAN 2.0B, 2x I2C, 2x SPI, 4x 32b GPIO		
Integrated Block Functionality	Power Management	Full / Low / PL / Battery Power Domains			
	Security	RSA, AES, and SHA			
	AMS - System Monitor	10-bit, 1MSPS – Temperature and Voltage Monitor			
PS to PL Interface			12 x 32/64/128b AXI Ports		
Programmable Logic (PL)	Programmable Functionality	System Logic Cells (K)	192	256	504
		CLB Flip-Flops (K)	176	234	461
		CLB LUTs (K)	88	117	230
	Memory	Max. Distributed RAM (Mb)	2.6	3.5	6.2
		Total Block RAM (Mb)	4.5	5.1	11.0
		UltraRAM (Mb)	13.5	18.0	27.0
	Clocking	Clock Management Tiles (CMTs)	4	4	8
		DSP Slices	728	1,248	1,728
	Integrated IP	Video Codec Unit (VCU)	1	1	1
		PCI Express® Gen 3x16	2	2	2
		150G Interlaken	-	-	-
		100G Ethernet MAC/PCS w/RS-FEC	-	-	-
		AMS - System Monitor	1	1	1
	Transceivers	GTH 16.3Gb/s Transceivers	16	16	24
		GTY 32.75Gb/s Transceivers	-	-	-
Speed Grades	Extended ⁽²⁾		-1 -2 -2L -3		
	Industrial		-1 -1L -2		



Machine Vision: Решения и инструменты Xilinx

Больше чем просто “железо”





APPLICATION



EDGE AI



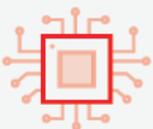
EMBEDDED SW
FOR MIXED CRITICALITY



ANY-TO-ANY CONNECTIVITY
SMARTER CONTROL
EMBEDDED VISION



FUNCTIONAL SAFETY
& CYBERSECURITY



SILICON ARCHITECTURE

Обзор Решений Для Машинного Зрения

- ▶ Sensor Interface
 - MIPI CSI, DSI; SLVS-EC; LVDS
- ▶ Image and Video Processing
 - Demosaicing, gamma correction, scaling, etc.
- ▶ Camera/Frame Grabber Connectivity
 - USB3, HDMI, CXP, GigE Vision, DisplayPort, SDI, Camera Link
- ▶ Partner Solutions
 - ISPs (Image signal processing), HDR, Software, Tools

Industry Alliances



Sensor Partners



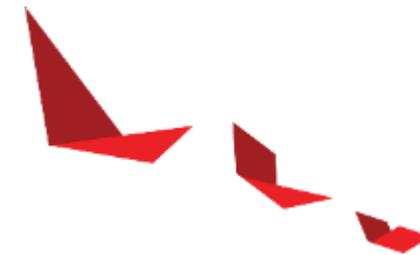
ISP Partners



Software and IP Partners



SONY®

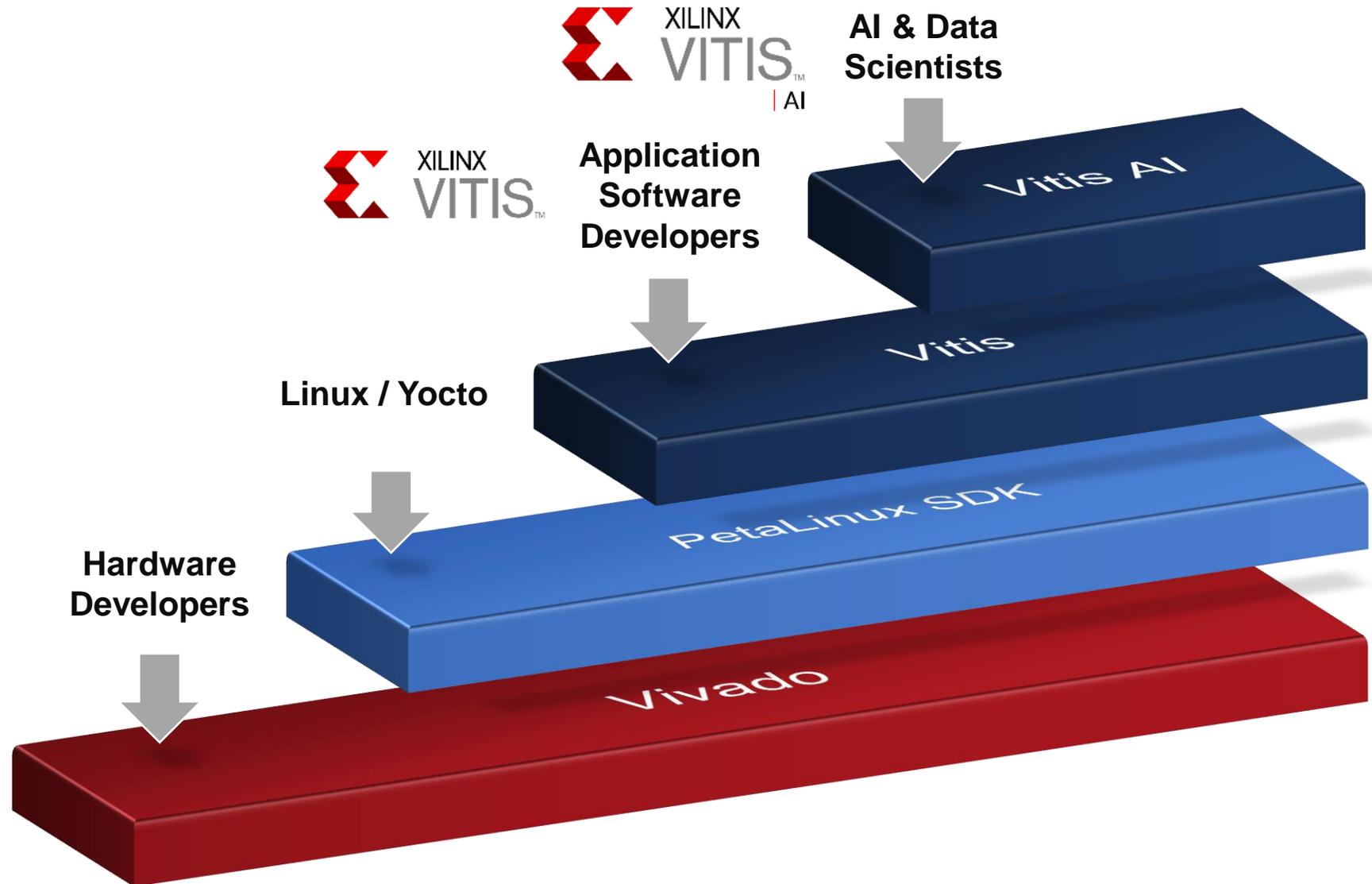


- ▶ *"Xilinx является ключевым стратегическим партнером в продвижении внедрения датчиков Sony в приложениях машинного зрения благодаря их активной поддержке сенсорного интерфейса Sony SLVS-EC. По нашим оценкам, Xilinx занимает 70% рынка в индустрии машинного зрения."*

Sony Semiconductor Solutions



Платформы разработки Xilinx для ВСЕХ разработчиков



VITIS AI: от модели нейронной сети до кристалла

Frameworks

Caffe



K Keras

PyTorch

TensorFlow

tvm

Models

Classification

Detection

Semantic Segmentation

RNN

Multi-Layer Perceptron

Multi-task
Multi-Model
Multi-Stream

Xilinx Pre-optimized | Open Source | Custom

Vitis AI
Development Kit

AI Optimizer

AI Quantizer

AI Compiler

Xilinx Runtime library (XRT)

Domain Specific
Architectures

CNN DPU

LSTM DPU

MLP DPU

AI
Profiler

Platforms



ZCU102



ZCU104



Ultra96



Custom
Hardware



Alveo



Alibaba Cloud
aliyun.com

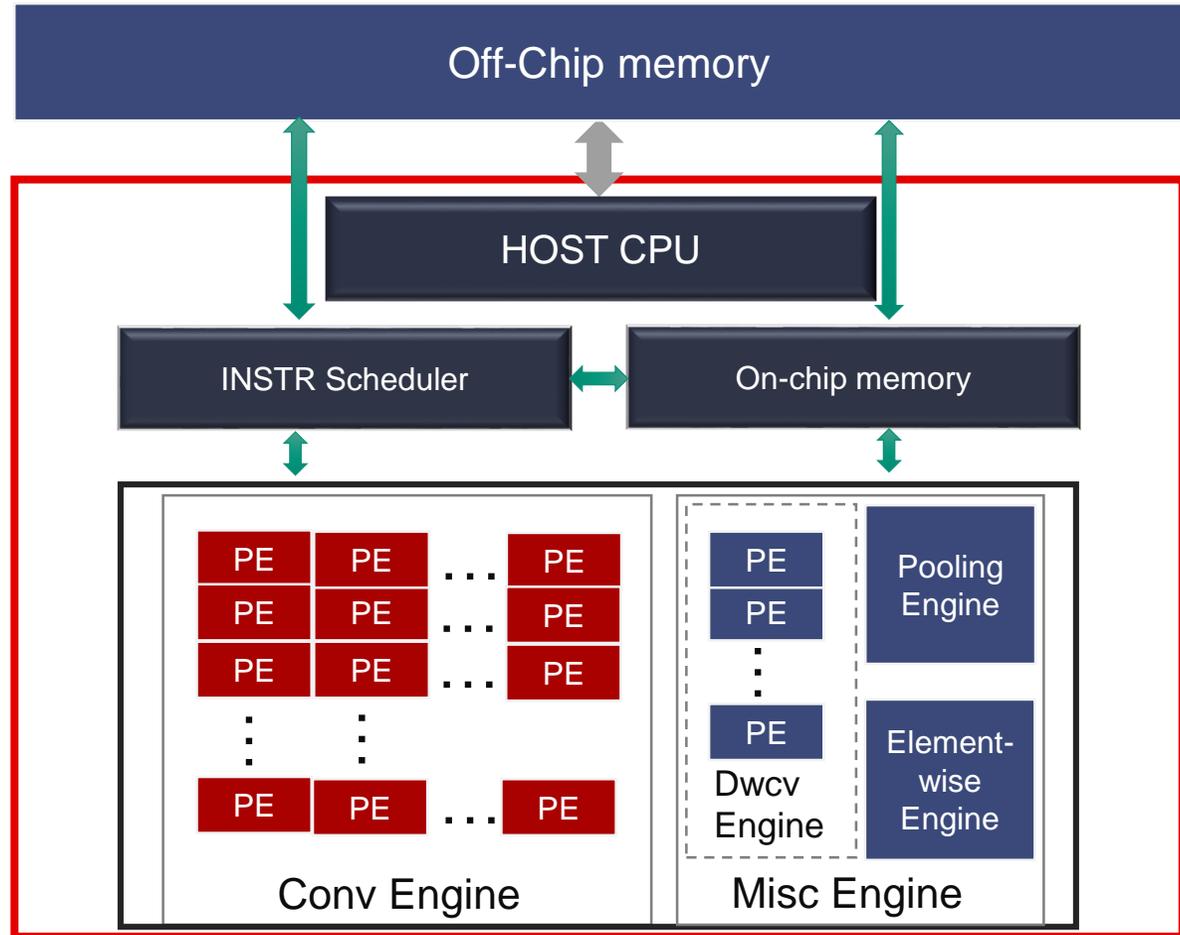
Cloud

NIMBIX

Microsoft Azure

XILINX

Xilinx Deep Learning Processing Unit (DPU)

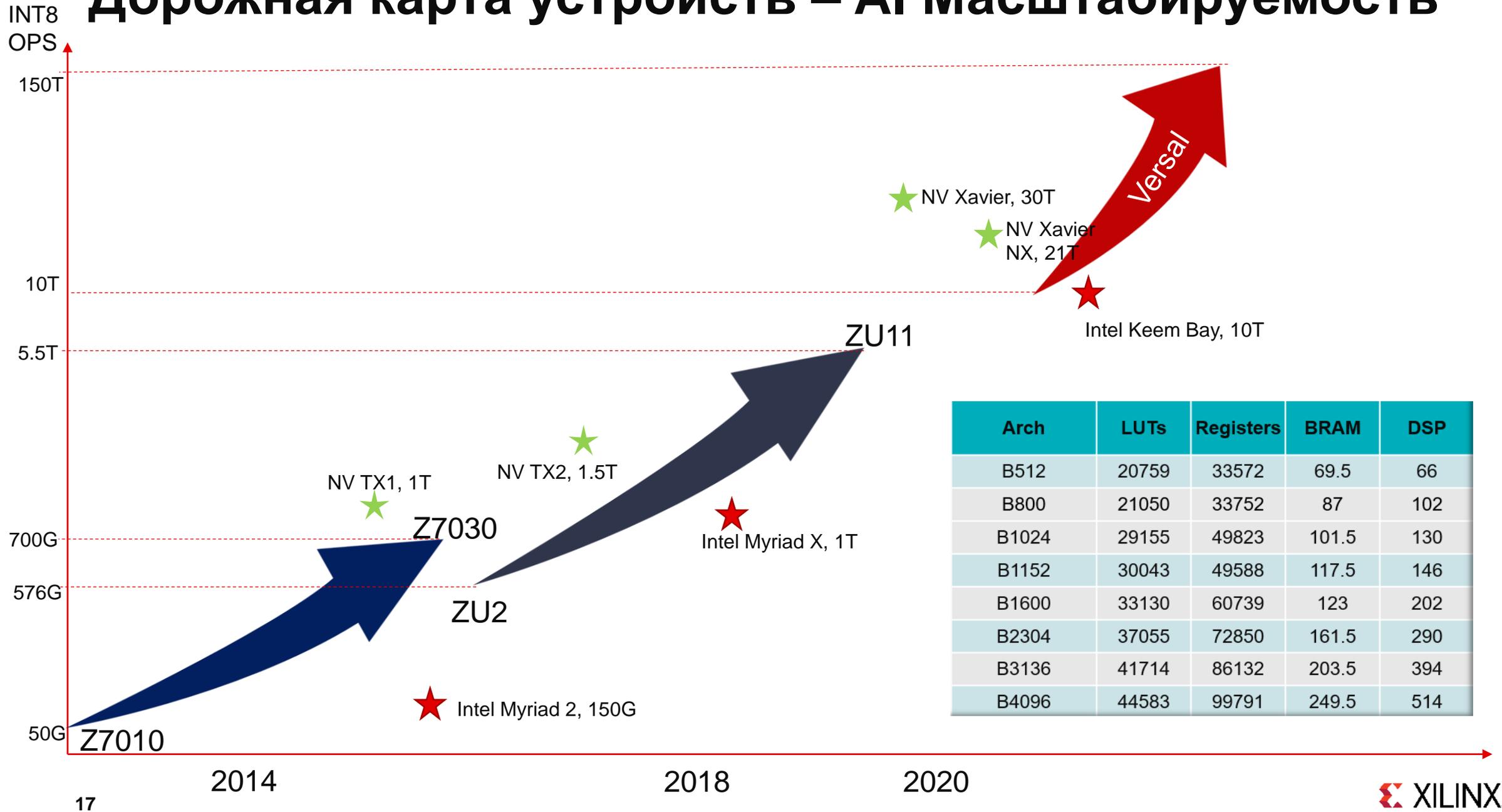


Гибкое и конфигурируемое IP ядро DPU

- Настраиваемая аппаратная архитектура. Масштабируется от 7Z020 до 7Z100, от ZU2 до ZU11
- Использование RAM для повышения производительности или снижения использования ресурсов
- Поддержка дополнительных опций для повышения производительности
- Поддержка функции низкого энергопотребления

https://www.xilinx.com/support/documentation/ip_documentation/dpu/v3_2/pg338-dpu.pdf

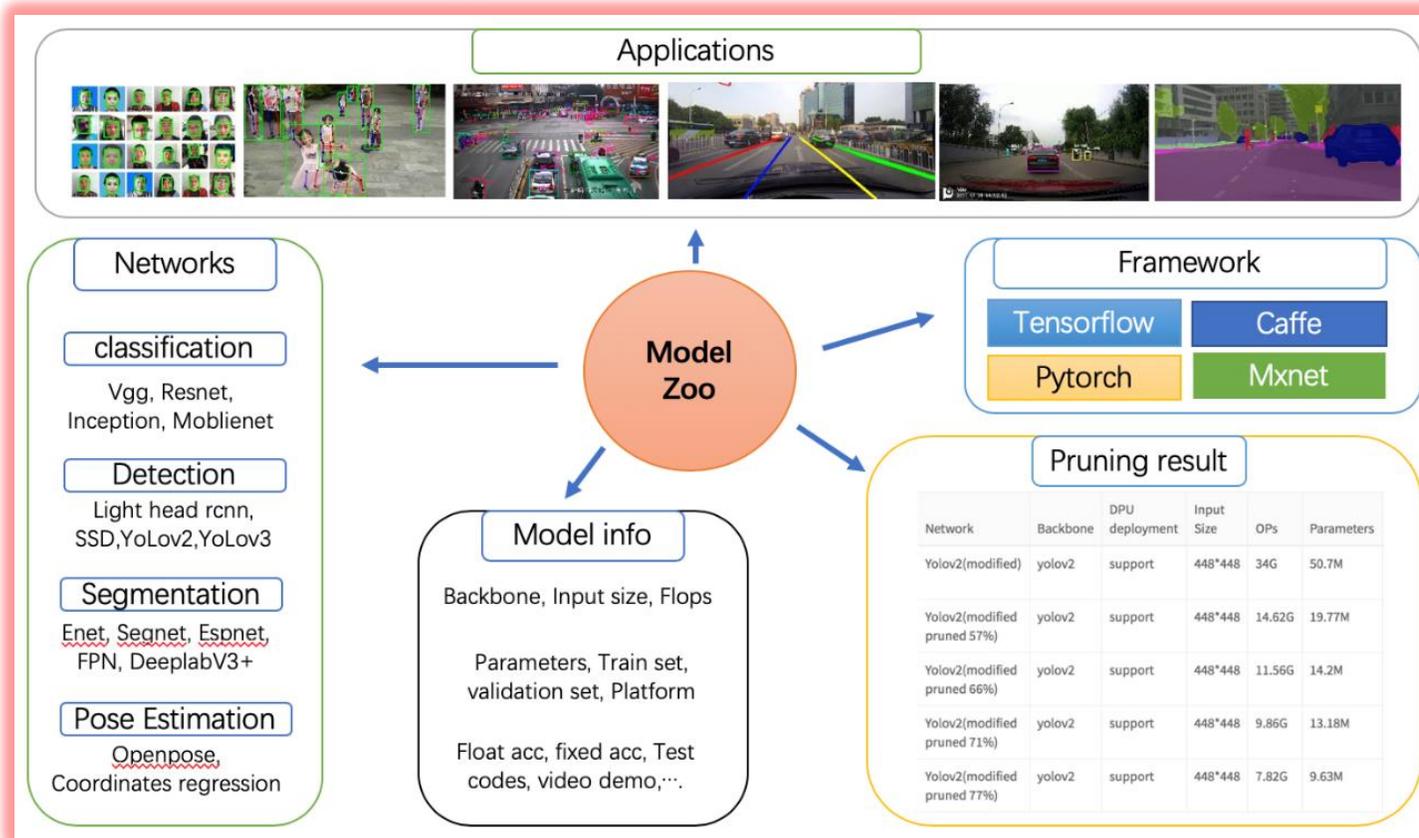
Дорожная карта устройств – AI Масштабируемость



VITIS AI: Готовые решения в Model Zoo

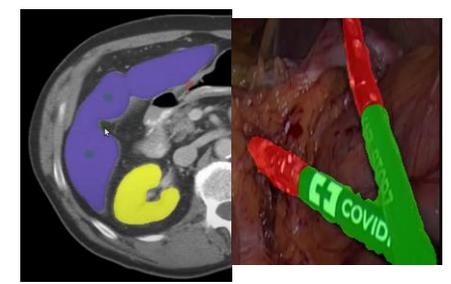
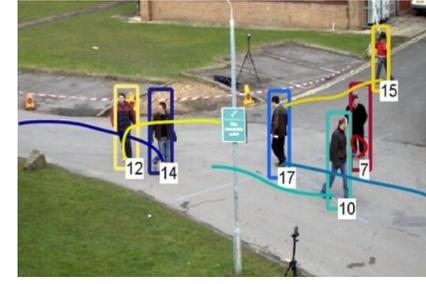
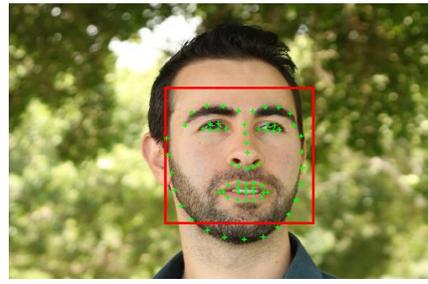
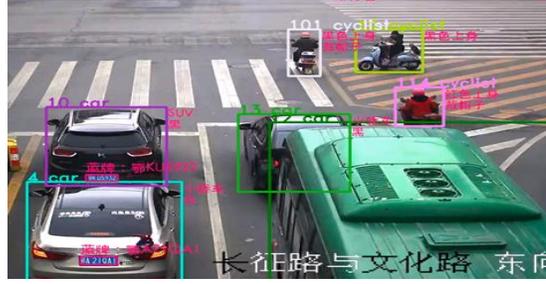


- ✓ Открыты для всех пользователей
- ✓ Использование основных фреймворков и сетей
- ✓ Развертываемые и переобучаемые модели



Application	Module
Face	Face detection
	Landmark Localization
	Face recognition
	Face attributes recognition
Pedestrian	Pedestrian Detection
	Pose Estimation
	Person Re-identification
Video Analytics	Object detection
	Pedestrian Attributes Recognition
	Car Attributes Recognition
	Car Logo Detection
	Car Logo Recognition
	License Plate Detection
	License Plate Recognition
ADAS/AD	Object Detection
	3D Car Detection
	Lane Detection
	Traffic Sign Detection
	Semantic Segmentation
	Drivable Space Detection

Популярные модели, поддерживаемые в Vitis AI



▶ Classification

- ResNet
- Inception
- MobileNet
- SqueezeNet

▶ Object Detection

- SSD
 - MobileNet
 - Inceptionv2
 - ResNet
- Yolov2 & Yolov3
- RefineDet

▶ Face

- Detection
 - DenseBox
 - RetinaFace
- Landmark
- Recognition
 - ResNet20
 - ResNet64

▶ Person

- Detection
- Reid
 - Pose estimation
 - Openpose
- Tracking
 - SortReid
 - FairMOT
- Attribute recognition
- Crowd counting

▶ Medical

- Detection
 - RefineDet
- Segmentation
 - FPN
 - UNet

Преимущества устройств Xilinx для Машинного Зрения

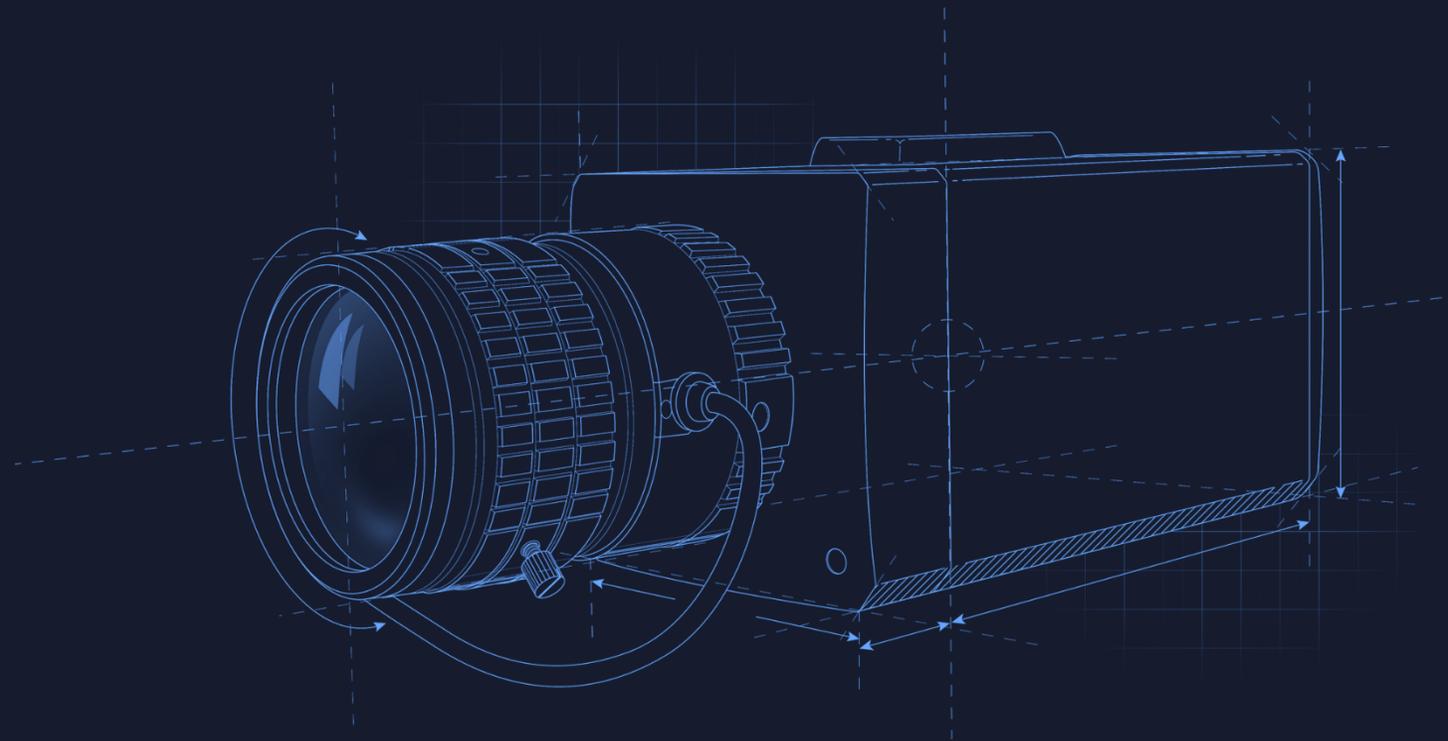
- ▶ Гибкий вывод ИИ с низкой задержкой
 - Искусственный интеллект на камере обеспечивает обнаружение дефектов в режиме реального времени с наименьшей задержкой
 - Идеология Frame grabber +IPC приводит к более высокой задержке, более низкой общей пропускной способности
- ▶ IP ядра для задач ISP
 - Библиотеки Vitis Vision обеспечивают гибкую и легкую отправную точку по сравнению с ASSPs ISPs
- ▶ “Кастомная” видеообработка
 - Например, алгоритмы сегментации могут быть ускорены в ПЛИС
- ▶ Поддержка интерфейсов для видео-сенсоров
 - SLVS, SLVS-EC, MIPI, LVDS, HDMI, USB.....

Zynq® UltraScale+™ MPSoC платформы для разработки



Kits and Boards			Demo Platforms
<p>Ultra96-V2 Embedded Linux Starter Kit</p> <p>\$249</p>  <p>ZU3EG AES-ULTRA96-V2-G</p> 	<p>ZCU104 Embedded Vision Platform</p> <p>\$895</p>  <p>ZU7EV EK-U1-ZCU104-G</p> 	<p>ZCU106 Multimedia & Connectivity Platform</p> <p>\$1,995</p>  <p>ZU7EV EK-U1-ZCU106-G</p> 	<p>Xilinx Smart Camera+ Demonstration Platform</p> 

SmartCamera+ Демонстрационная Платформа



SmartCamera+ Демонстрационная Платформа

ON

Onsemi AP1302
MIPI Image Signal Processor
13MP@30, 1080p@120
HDR, Clarity+, Bayer, JPEG, YUV,
Gamma, Face Detection

ON

Onsemi AR1335 13MP
1/3" Progressive Scan
4208x3120, 30fps

eMMC 8GB

H.264/H.265/
MJPEG

microSD 2.0

Mini DP 1.2a Output

Ethernet
HTTP, RTSP, TCP/IP, UDP

USB 2.0

Audio Codec

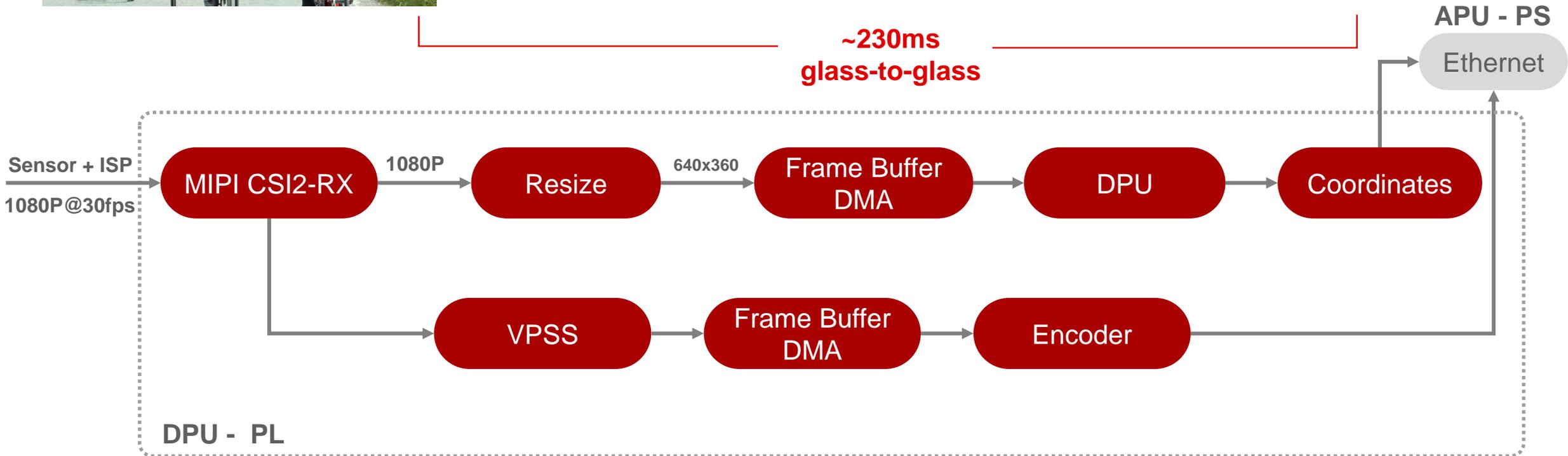
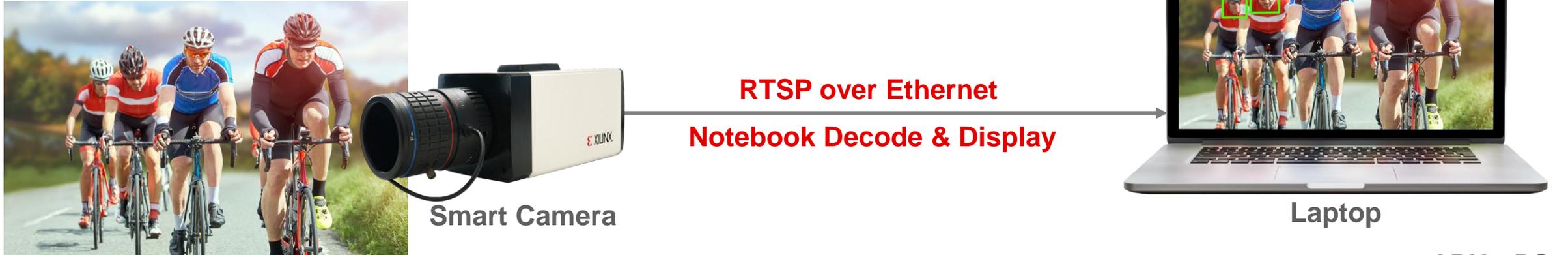


DDR4,2400MT/s,64bit,4GB

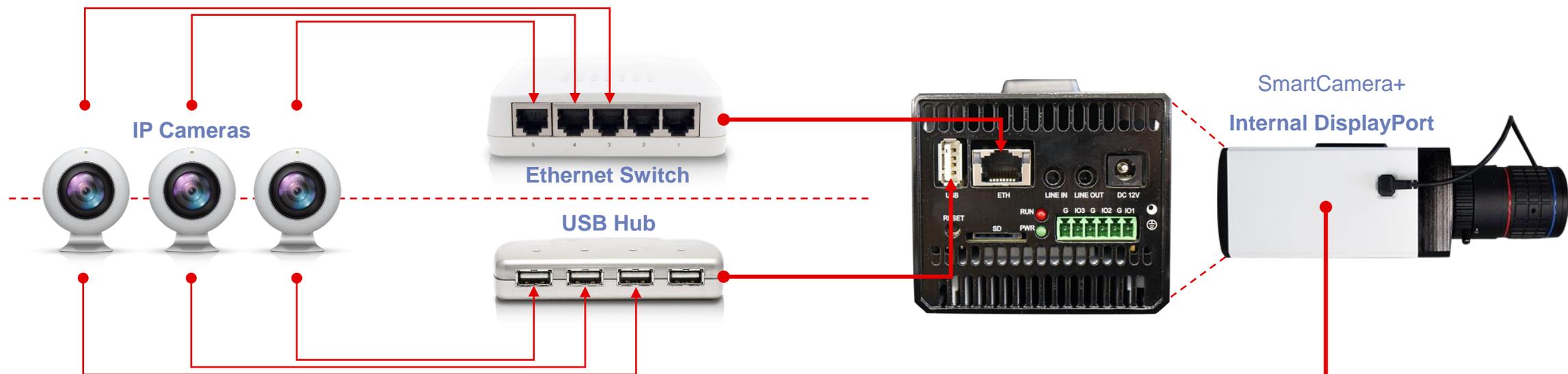


Xilinx ZU+
EV MPSoC

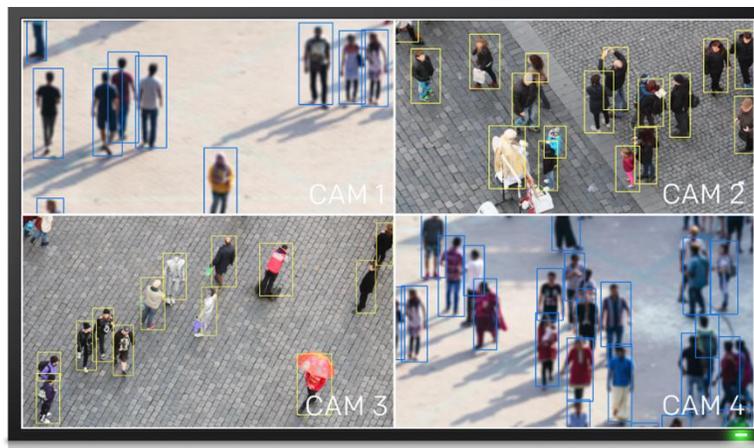
Демонстрация задачи обнаружения лиц с низкой задержкой



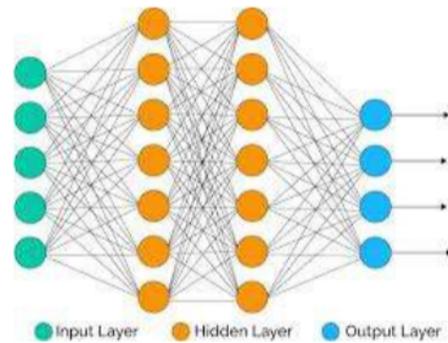
SmartCamera+: Это тоже AI Box!



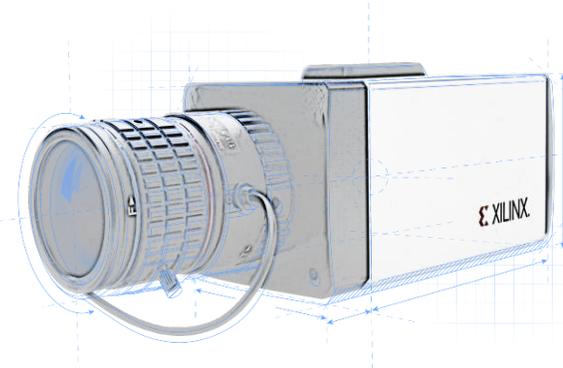
- ▶ Возможность подключения по PoE или USB нескольких “глупых” камер!
- ▶ Встроенный mini-DP интерфейс для подключения монитора!
- ▶ Возможность поддержки многоканальной аналитики с использованием ИИ!



Определение проблемы



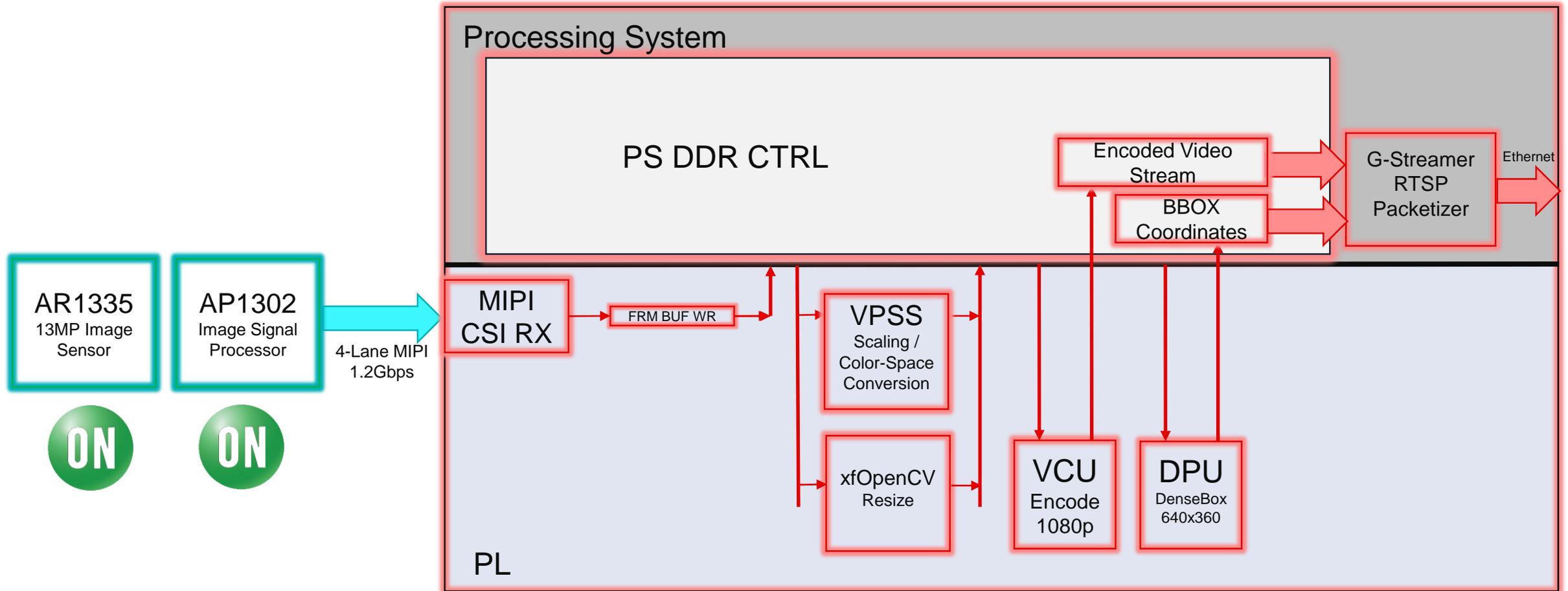
= **43 TOPS**



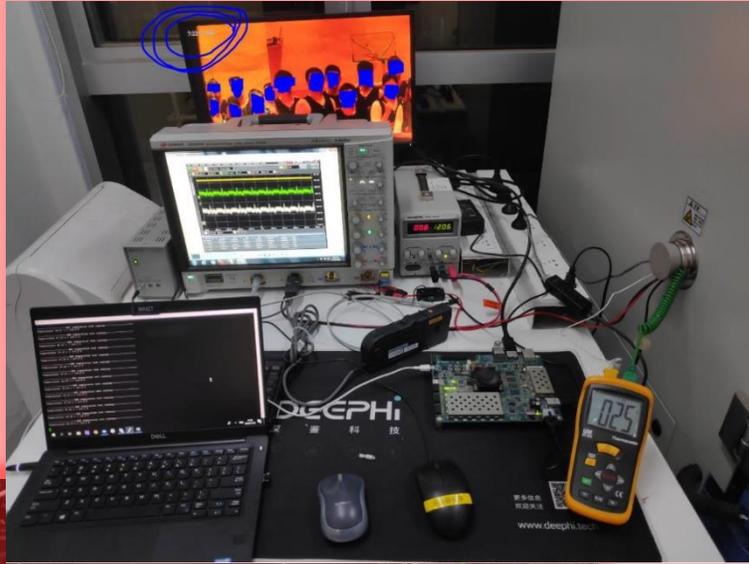
1080p Object Detection (SSD) @ 30fps

<10W, <50 ms latency, cost-effective

Пример SmartCamera+ видео конвейер



Энергопотребление меньше чем у GPUs

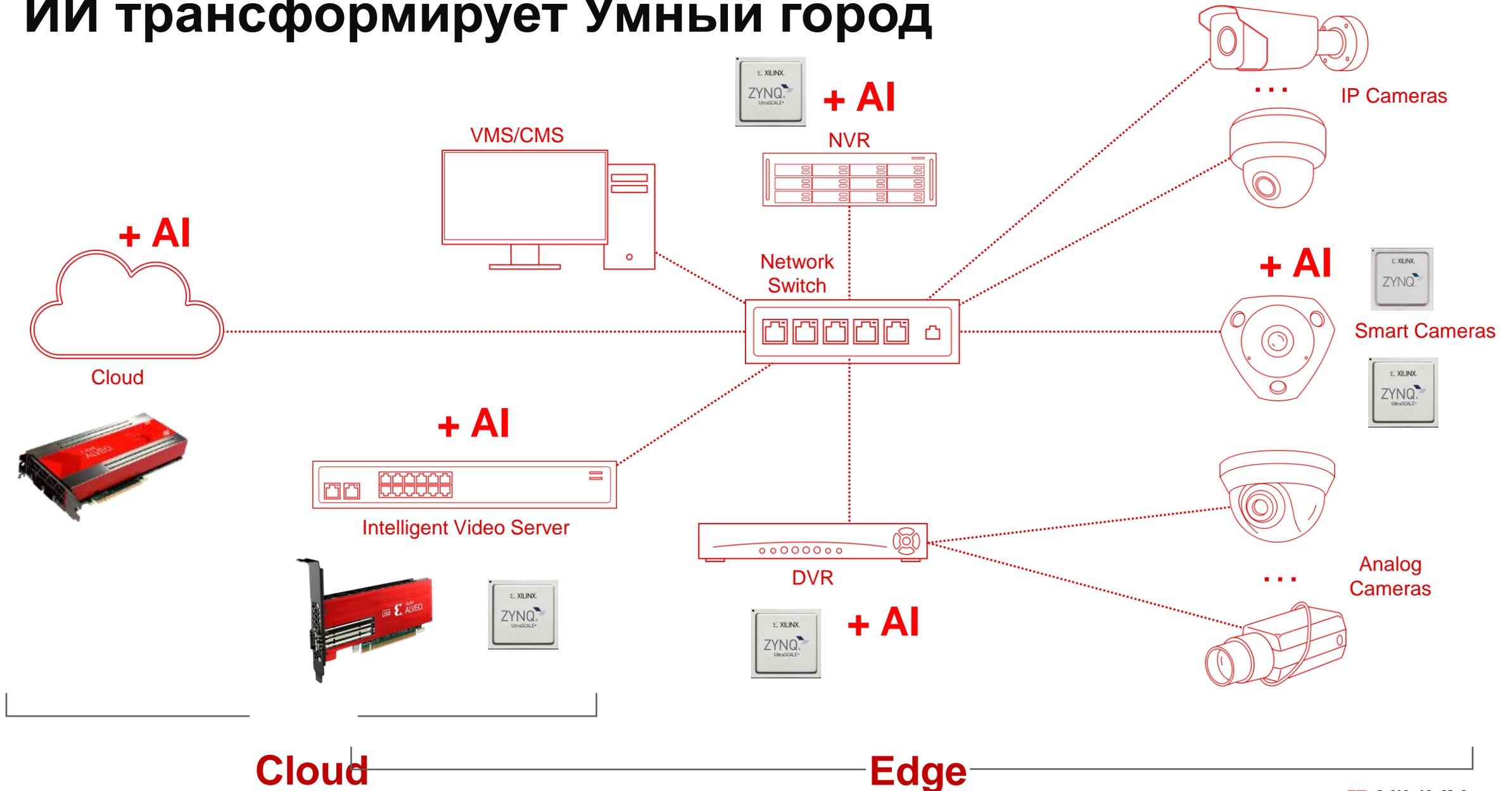


Measurement Condition:
DPU: B3136@300MHz
DPU Network: Dense_Box_640x360
AI feature: Face Detection, 1080P@30fps

Ambient Temperature	Average Power Consumption	Peak Power Consumption	ZU4EV Internal Temperature	Shell Temperature
25 C	8.6W	13.9W	48 C	37 C
35 C	8.7W	14.2W	59 C	47 C
45 C	9.1W	14.6W	69 C	58 C
55 C	9.4W	14.9W	78 C	67 C
60 C	9.6W	15.1W	84 C	73 C

Умный Город

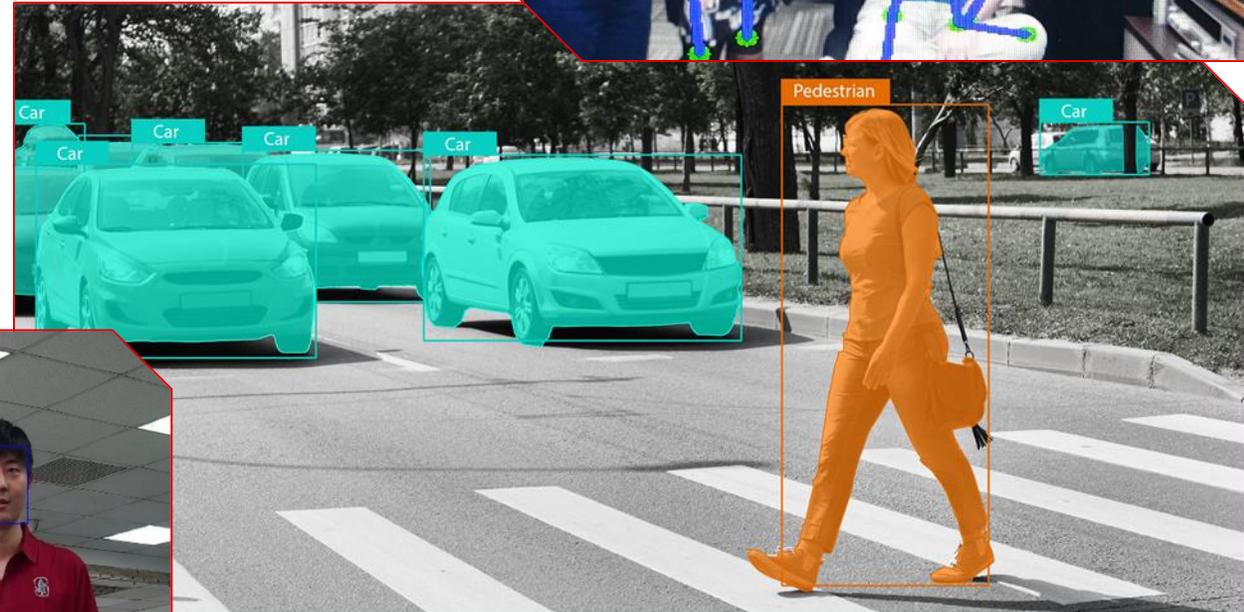
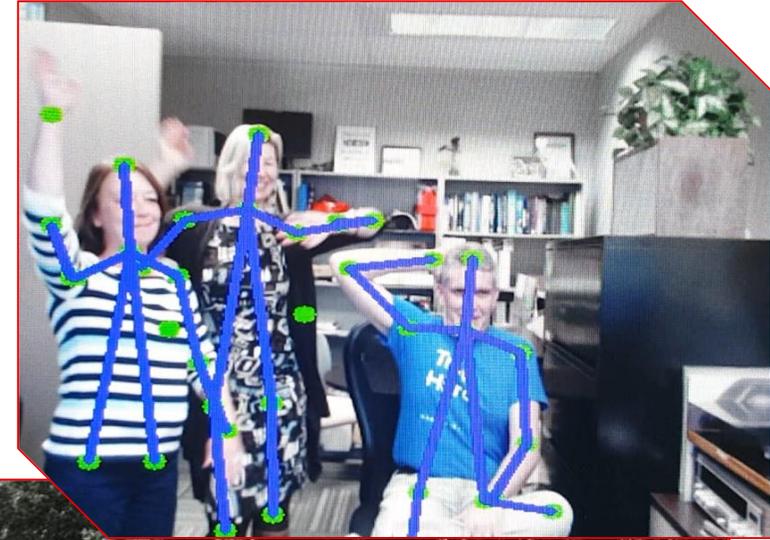
ИИ трансформирует Умный город



Vitis-AI Models для Умных Городов

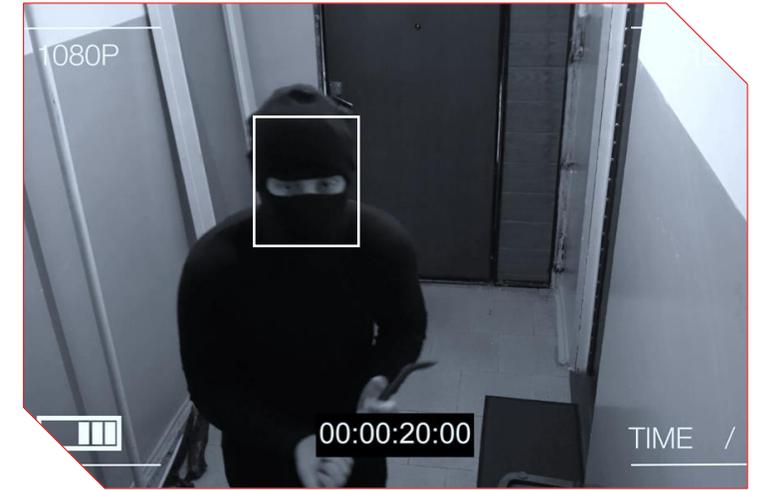
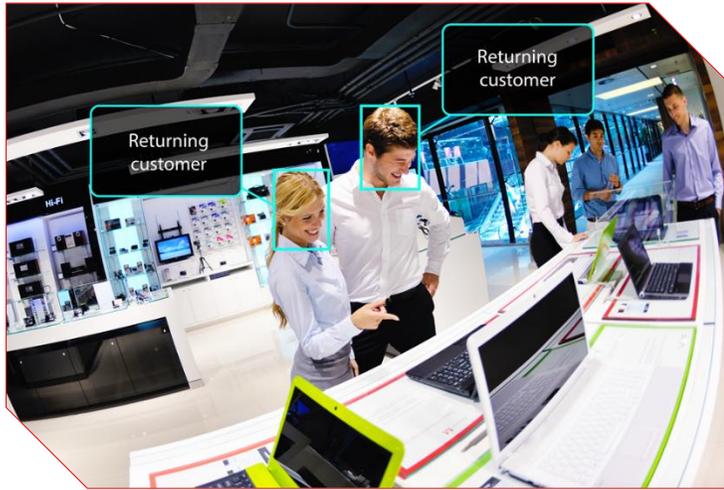
▶ Предобученные модели для работы “из коробки”:

- SSD для обнаружения транспорта, пешеходов, велосипедистов
- RefineDet для обнаружения людей
- DenseBox для обнаружения лиц
- FPN and OpenPose для определения поз



Розничная торговля (Ритейл)

Умный ритейл как часть концепции умного города



▶ “Умный” маркетинг

- Определение потребностей клиентов
- Повторно идентификация постоянных клиентов
- Анализ покупательских привычек
- Анализ настроений
- Виртуальные примерочные комнаты
- Целевая реклама

▶ Аналитика внутри магазина

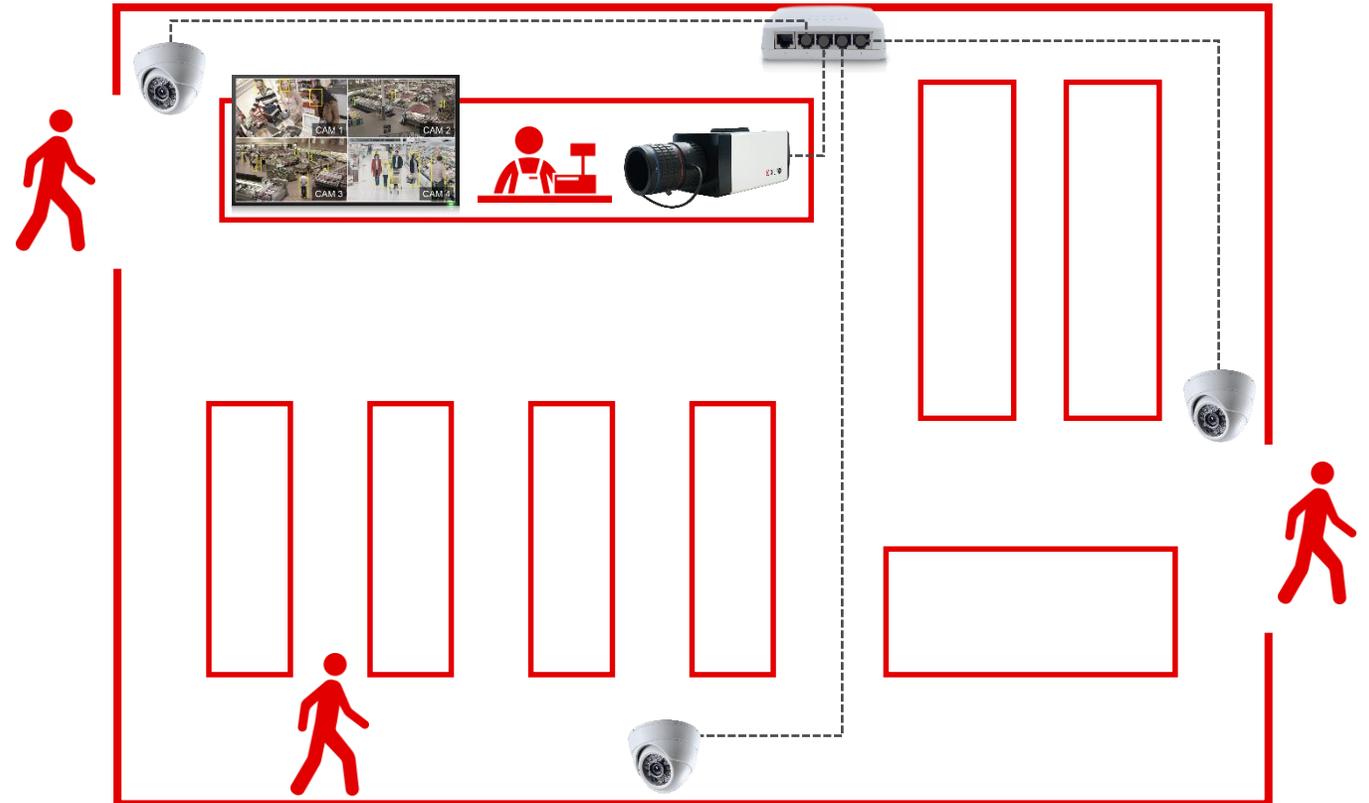
- Обнаружение пропущенного сканирования
- Подсчет посетителей магазина
- Отслеживание перемещения посетителей
- Черный список магазинных воров
- Демографический анализ
- Оптимизация цен

▶ Системы безопасности на основе видеонаблюдения

- Обнаружение ключевых событий
- Обнаружение в реальном времени незаконного проникновения на территорию

SmartCamera+ Multi-Camera System

- ▶ SmartCamera+, с достаточной производительностью для обработки с использованием ИИ данных с нескольких каналов
- ▶ Подсчет и определение посетителей, трэкинг, обнаружение давки / толпы, определение пропущенного сканирования, эмоций, пола, возраста...
- ▶ Распознавание лиц
 - Ведение табеля учета рабочего времени сотрудников
 - Распознавать VIP клиентов
 - Рекомендации по продукту
- ▶ HDMI или DisplayPort для подключения дисплея (на кассе например)

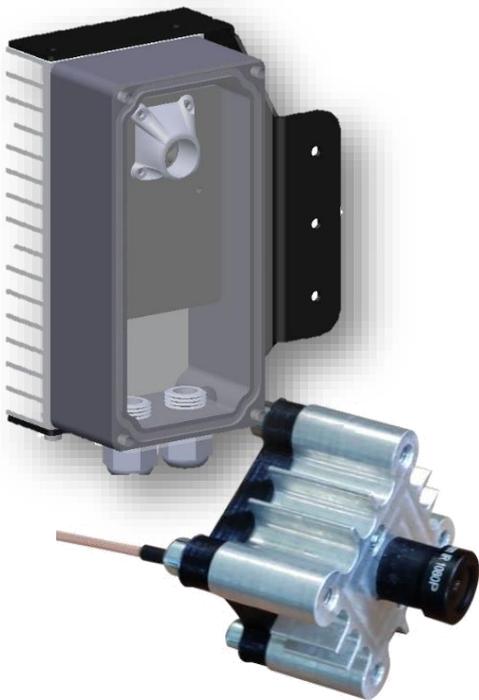




Пример реализации системы машинного зрения на базе Xilinx MPSoC в России

Смарт-платформа машинного зрения Cuslope

► www.nsc-com.com



Смарт-камера Циклоп предназначена для фиксации и анализа видеоизображения для решения задач технического зрения. Встроенная система ввода/вывода позволяет считывать показания датчиков и управлять исполнительными механизмами в соответствии с заданной программой. Возможность подключения до 8 оптических сенсоров позволяет увеличить рабочее поле или контролировать объекты с разных точек зрения. Подсистема управления освещением позволяет непосредственно подключить до 5 управляемых источников света (стробоскопический, направленные, рассеянные источники) и плавно управлять их яркостью в зависимости от решаемой задачи и внешних условий. Поддержка сетевых промышленных протоколов позволяют передавать изображения и результаты распознавания (считанные коды, размеры, цвет и многое другое) в сеть для дальнейшей обработки. Разграничение прав доступа к системе позволяет гибко распределять обязанности и ответственность между сотрудниками. Русифицированный интерфейс и техническая поддержка позволяют поддерживать постоянную работу оборудования. Команда разработчиков обеспечивает возможность доработки функций системы под Заказчика

Основные решаемые задачи:

- контроль наличия, положения и качества элементов продукции: наклеек, крышек, пробок, таблеток, маркировки и т.п.
- контроль цвета объекта: равномерность окраски, соответствие цветовой маркировки.
- измерение геометрических размеров изделий или их частей: размер и конусность клемм аккумуляторных батарей, линейные размеры заготовок, поворот и смещение изделий, размер коробок.
- считывание одномерных, двумерных кодов и символьной информации: честный знак, QR коды, штрих-коды, серийные номера
- Подсчет объектов с контролем их качества и ориентации
- Контроль качества поверхности: при размотке стальной ленты, контроль заготовок на конвейере.
- Сравнения с эталонным изделием

Питание

- Интерфейс: RJ45: Power over Ethernet, IEEE 802.3af
- Клеммы питания: напряжение: 12В постоянного тока
- Номинальная мощность: 15Вт

Оптические сенсоры

- Встроенных сенсоров 1
- Максимальное количество сенсоров: 8
- Длина кабеля подключения внешнего сенсора до 10м
- Разрешение 2,3,5,18 Мп
- Частота кадров до 60
- Объектив: M12, SC

Инструменты

- Образ, подсчет пикселей, контрастность, яркость, кромки,
- поиск кругов,
- измерения, подсчет

Ввод/вывод

- Ethernet 1Gbps
- RS485
- Optocoupled in
- Optocoupled out
- Trigger in (3.3V)
- Trigger out (3.3V)
- Power out (12V, 2A)
- USB type-C display port
- USB type-C Hi speed

Протоколы

- EtherNet/IP, PROFINET, Modbus TCP, TCP/IP, UDP, FTP, RS485, UDP/IP, HTTP, RTP

Управление освещением

- 4 канала управления
- Стробоскопический свет

Параметры

- Габариты (ШxВxГ, мм): 160x60x85
- Рабочая температура: 0 - + 65°C
- Относительная влажность: 10%-95%

Система контроля качества производственной линии по выпуску аккумуляторных батарей

FRONTEND приложения
(диагностика, пульт оператора...)

Кастомизированные модули
(Python, C++)

Библиотечные модули,
ввод-вывод, интерфейсы к аппаратным
ускорителям...

Операционная система Linux

Драйвера периферии
и аппаратных
ускорителей

Модули ускорителей (HDL)

The screenshot displays a comprehensive quality control system interface. On the left, a checklist for 'Bravo 74' includes items like 'Наклейки слева', 'Шлиц', and 'Клейма справа', each with a 'Пройден' (Passed) status. The main area features eight camera feeds: 'Крышка слева/справа', 'Корпус слева/справа', 'Клемма слева/справа', and 'Сторона слева/справа', all showing green bounding boxes around detected components. Below the feeds, a 'Браковочный стол' (Defect table) is visible. The bottom section contains two code editors: one showing Python code for a 'Sticker' class and another showing Verilog HDL code for a 'gaussian' filter module.

```
class Sticker(Area):
    def __init__(self, *args, **kwargs):
        super().__init__(*args, **kwargs)

    def check(self, *args, **kwargs):
        t1 = time.time()
        res = { "corn": 0, "posX": -100.0, "posY": -100.0, "draw": None }
        calib = getter(args, 1, kwargs, "calib")
        if calib is None: return res
        plane = getter(args, 2, kwargs, "calibPlane")
        if plane is None: return res
        src = getter(args, 0, kwargs, "src", dtype=Image, default=Image((2000,2000)))
        inc = getter(args, 2, kwargs, "inc", dtype=float, default=0.05)

        hIdx = getter([], 0, kwargs, "hIdx", dtype=int, default=0)
        vIdx = getter([], 0, kwargs, "vIdx", dtype=int, default=2)

        alpha = kwargs["alpha"]
        x = kwargs["x"]
        z = kwargs["z"]
        H = kwargs["H"]
        plane = kwargs["featurePlane"] if self.plane is None else self.plane

        addition = np.int32(src.shape() * inc)

        with calib.pushRotate(alpha):
            with calib.pushTranslateAbs([x, 0, z]):
                corners = np.array([ RSector_3DToScene(p, plane, calib, H, kwargs["corners"])
                box = np.array([ corners.min(axis=0), corners.max(axis=0) ], dtype=np.int32)
                stickerInfo = StickerFinder.find(src, self["template"], (box[0] - addition, box[1] - addition))

                if stickerInfo["zone"] is None:
                    stickerInfo["zone"] = corners
                    # kwargs["out"].polyline(stickerInfo["zone"], color=(0,255,0), thickness=2)

                res["err"] = stickerInfo["match"]
                offset = np.float32([ RSector_scene2DToPlane(p, plane, calib, H, kwargs["corners"])
                i = np.argmax(np.abs(offset[i][hIdx])); j = np.argmax(np.abs(offset[i][vIdx]))
                res["posX"] = offset[i][hIdx]; res["posY"] = offset[j][vIdx]
                res["draw"] = stickerInfo["zone"]

        for k,v in self["subAreas"].items():
            res[k] = v.check(stickerInfo["img"], *args, **kwargs)
            if res[k].get("draw") is not None: res[k]["draw"] += stickerInfo["zone"][0]

        t2 = time.time()
        print("Sticker time:", (t2 - t1) * 1000, "ms")
        return res

class SubArea(Area):
    def __init__(self, *args, **kwargs):
        super().__init__(*args, **kwargs)
        self.subAreas.clear()

    def addSubArea(self, *args, **kwargs):
        pass
```

```
> synth
> work
binarize.sv
ccl.sv
conv_x_v1_0.sv
conv_x.sv
convolution_filter.sv
convolution.sv
corner_det_v1_0.sv
dma_read.sv
dma_write.sv
gauss_dpram.sv
gaussian_v1_0.sv
gaussian.sv
hist_ram.sv
histogram.sv
i_store.sv
i_video.sv
image_in.sv
image_mask.sv
image_mux.sv
imoments_map_dpram.sv
imoments_moments_dpram.sv
imoments.sv
label_dpram.sv
line_slice.sv
m_axi_write.sv
multiline_slice.sv
45 logic [3:0][XCONV_WIDTH - 1 : 0] x_q;
46 logic [4:0][XCONV_WIDTH - 1 : 0] buf_y;
47 input [4:0][GAUSS_COEFF_WIDTH - 1 : 0] gauss;
48 logic [4:0][MUL_WIDTH - 1 : 0] mul;
49 logic [MUL_WIDTH - 1 : 0] result;
50
51 logic frame_d, skip_d, hold_d;
52 logic [MAX_X_WIDTH : 0] skip_count, hold_count, hold_line;
53 logic skip_finish, skip_finish_r;
54
55 logic [1:0] y_d;
56 logic zero_d;
57
58 i_video #(.(WIDTH(XCONV_WIDTH)))
59 video_a (.clock(clock), .reset(reset));
60
61 conv_x #(.(DATA_WIDTH(IN_DATA_WIDTH), .COEFF_WIDTH(GAUSS_COEFF_WIDTH)))
62 x_conv (
63     .clock(clock),
64     .reset(reset),
65     .gauss(gauss),
66     .in(in),
67     .out (video_a)
68 );
69
70 assign ena = out.ready | ~out_valid;
71 assign ena_a = ena & video_a_valid;
72 assign video_a_ready = ena;
73
74
75 always @* SENS LIST
76 begin
77     if (reset) begin
78         tag_a <= '0;
```

Спасибо за внимание!

Компания Avnet Silica:

- ✓ Официальный партнер и эксклюзивный поставщик Xilinx на территорию РФ
- ✓ Комплексная поставка электронных компонентов
- ✓ Техническая поддержка по всем вопросам применения продукции и ПО Xilinx

Обращайтесь:

- ✓ Alexander.Vlasov@Avnet

AVNET[®] SILICA