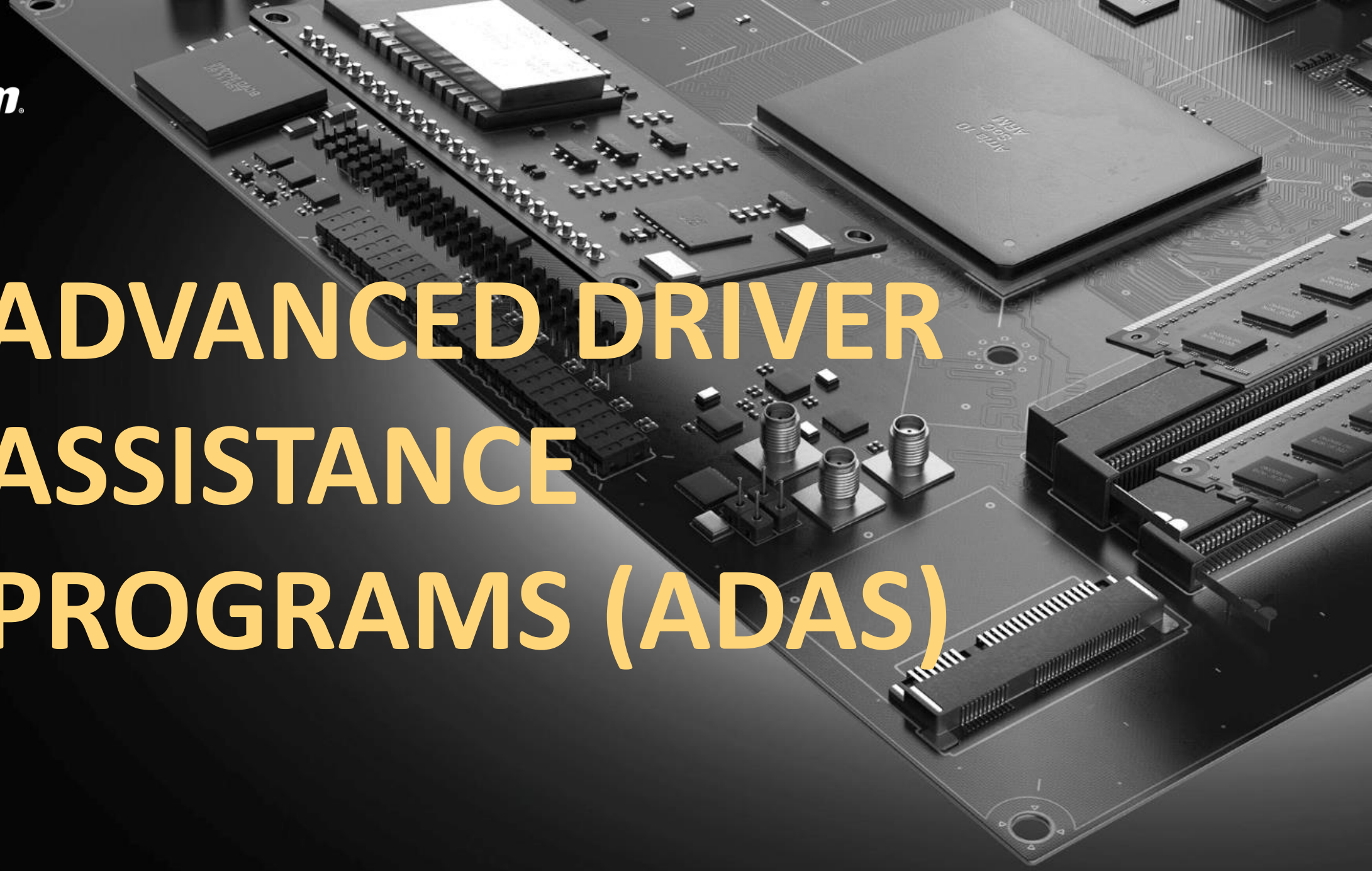
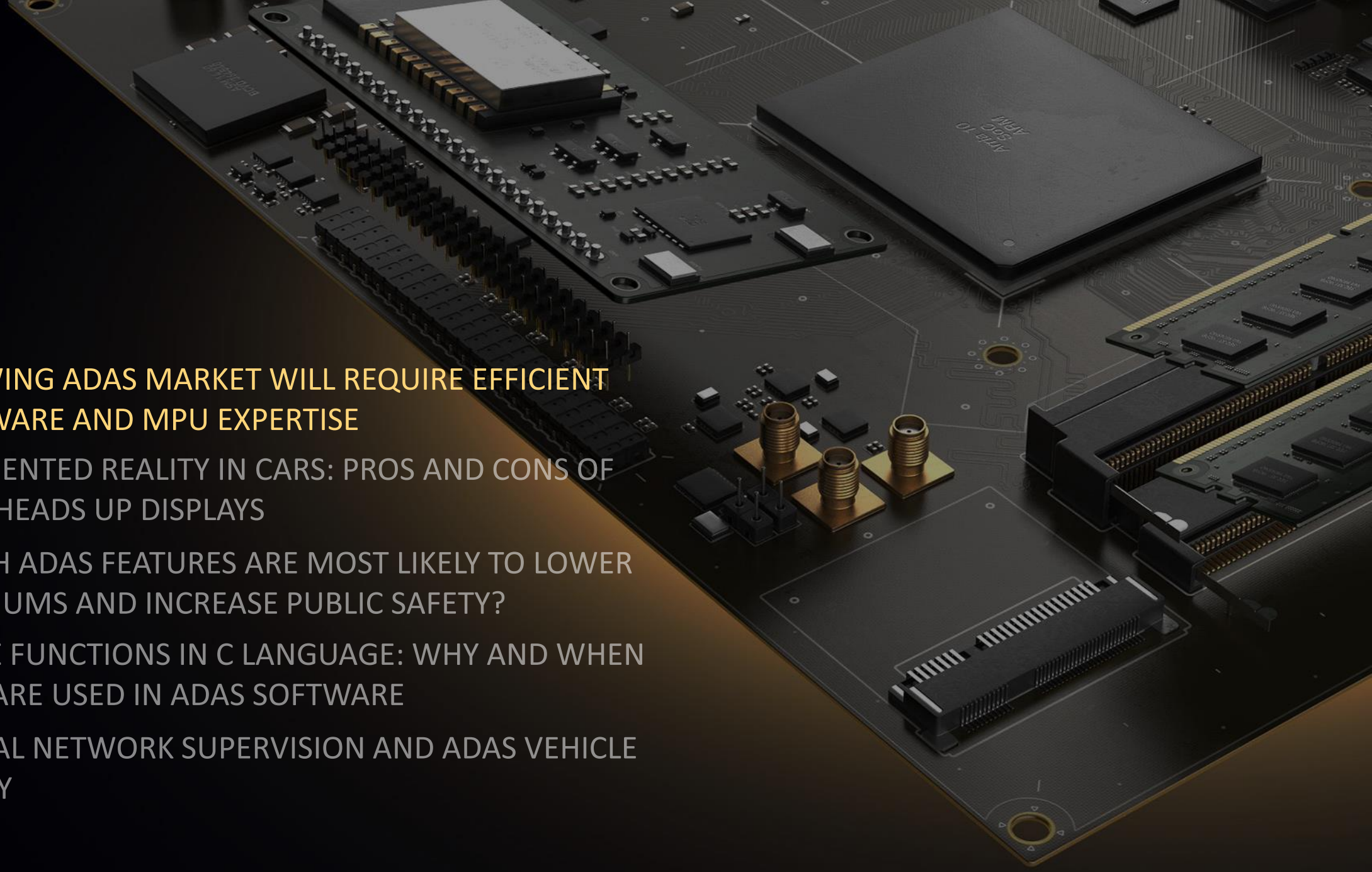
Altium

ADVANCED DRIVER ASSISTANCE PROGRAMS (ADAS)



Altium



1. GROWING ADAS MARKET WILL REQUIRE EFFICIENT SOFTWARE AND MPU EXPERTISE
2. AUGMENTED REALITY IN CARS: PROS AND CONS OF ADAS HEADS UP DISPLAYS
3. WHICH ADAS FEATURES ARE MOST LIKELY TO LOWER PREMIUMS AND INCREASE PUBLIC SAFETY?
4. INLINE FUNCTIONS IN C LANGUAGE: WHY AND WHEN THEY ARE USED IN ADAS SOFTWARE
5. NEURAL NETWORK SUPERVISION AND ADAS VEHICLE SAFETY

Altium

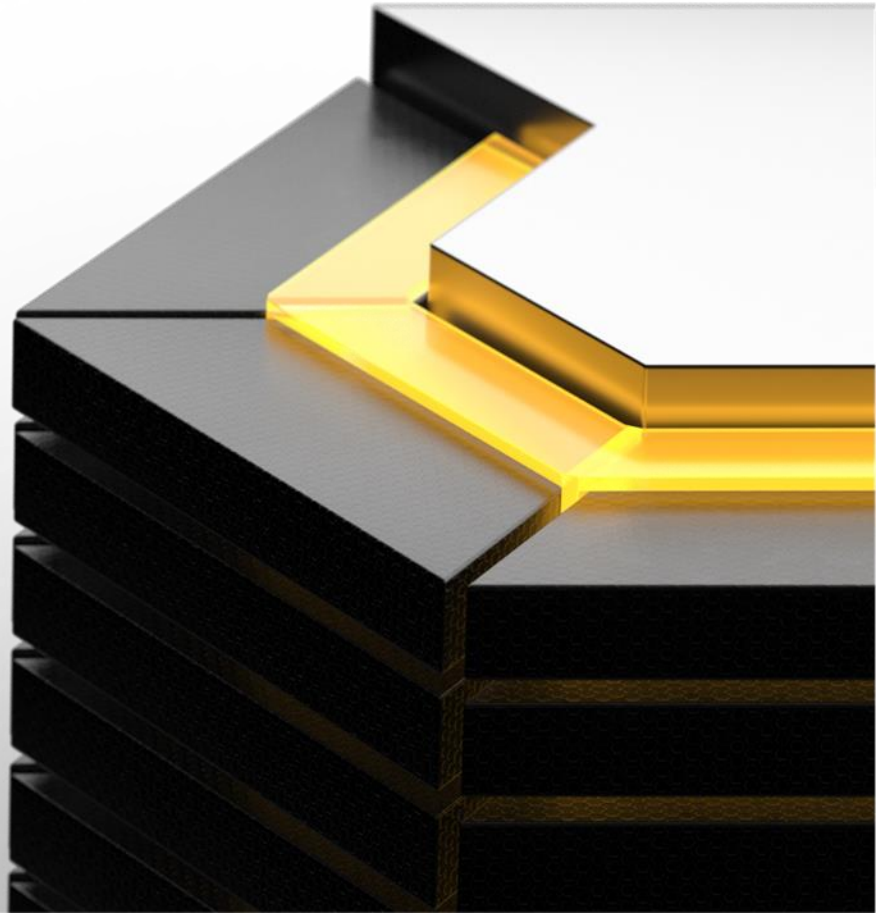
1. GROWING ADAS MARKET WILL REQUIRE EFFICIENT SOFTWARE AND MPU EXPERTISE

Altium

The Growing ADAS Market

While we are still years away from fully autonomous vehicles, advanced driver assistance systems (ADAS) are becoming more common in the automotive industry. The ADAS market is headed for major growth, and smart software developers can grow with them. Developers should be prepared to take advantage of this new era of transportation by making their code more efficient, and mastering the use of memory protection units (MPUs).

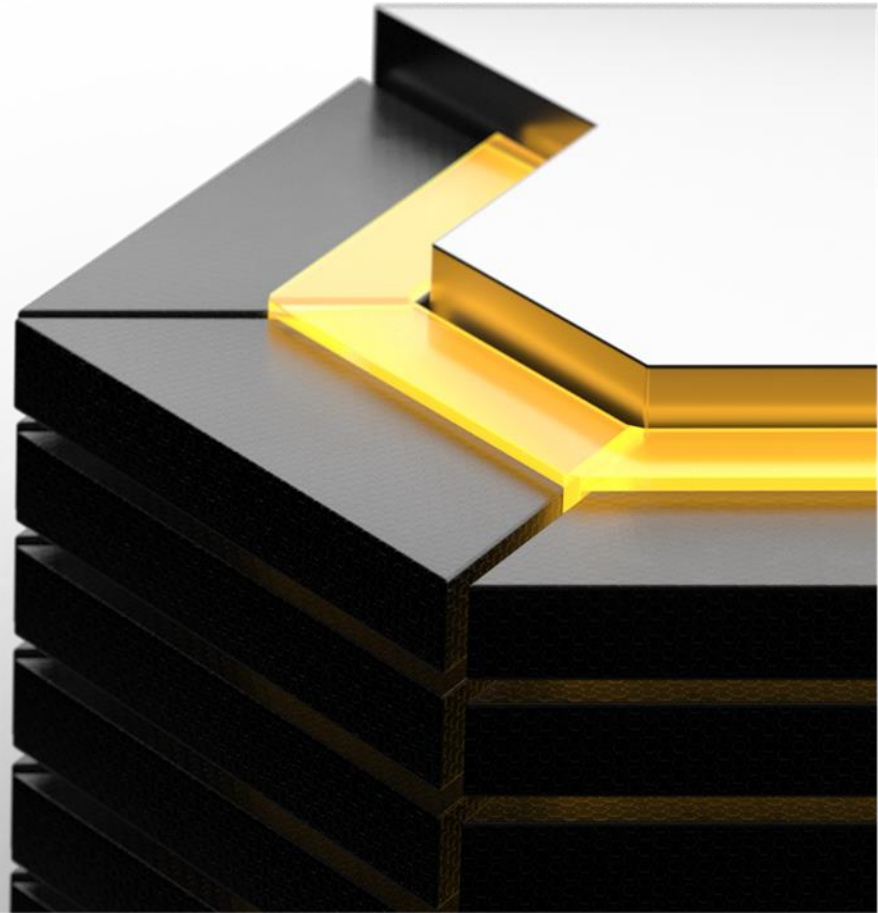
- US and EU will begin requiring new cars to have automatic braking systems and collision warning systems by 2020
- These mandates contribute to the rising ADAS market, which is expected to grow from $22.69 billion in 2015 to $78.19 billion by 2020.
- Currently, “self-driving” cars are only up to “level 3” of autonomy. Multiple manufacturers, though, are looking towards levels 4 and 5, which will require increasingly complex ADAS systems.



Software for ADAS

As more competitors enter the market, software developers will have to work harder to differentiate their programs. Two techniques to set yourself apart are making code that runs more efficiently and mastering the intricacies of MPUs.

- The level 4 and 5 systems you'll be controlling will need processors that are many times faster than those currently available for automotive applications.
- Making your programs more efficient will help reduce the load on processors.
- Efficient coding will help with timing safety concerns as well.
- One non-software solution to the processing problem is the move from distributed electronic control units (ECUs) to integrated multi-core microprocessor control units (MCUs).
- Now, with all systems sharing the same processor and memory, cascading faults could cause serious problems.
- A well made static analysis tool will save you a lot of time in debugging the code for your MPU. In addition, it will allow you to use your compiler to its full potential.
- As sensor complexity and variety increases, new cars will require centralized MCU's capable of higher level programming.



See original content here

Altium

1. GROWING ADAS MARKET WILL REQUIRE EFFICIENT SOFTWARE AND MPU EXPERTISE
2. AUGMENTED REALITY IN CARS: PROS AND CONS OF ADAS HEADS UP DISPLAYS
3. WHICH ADAS FEATURES ARE MOST LIKELY TO LOWER PREMIUMS AND INCREASE PUBLIC SAFETY?
4. INLINE FUNCTIONS IN C LANGUAGE: WHY AND WHEN THEY ARE USED IN ADAS SOFTWARE
5. NEURAL NETWORK SUPERVISION AND ADAS VEHICLE SAFETY

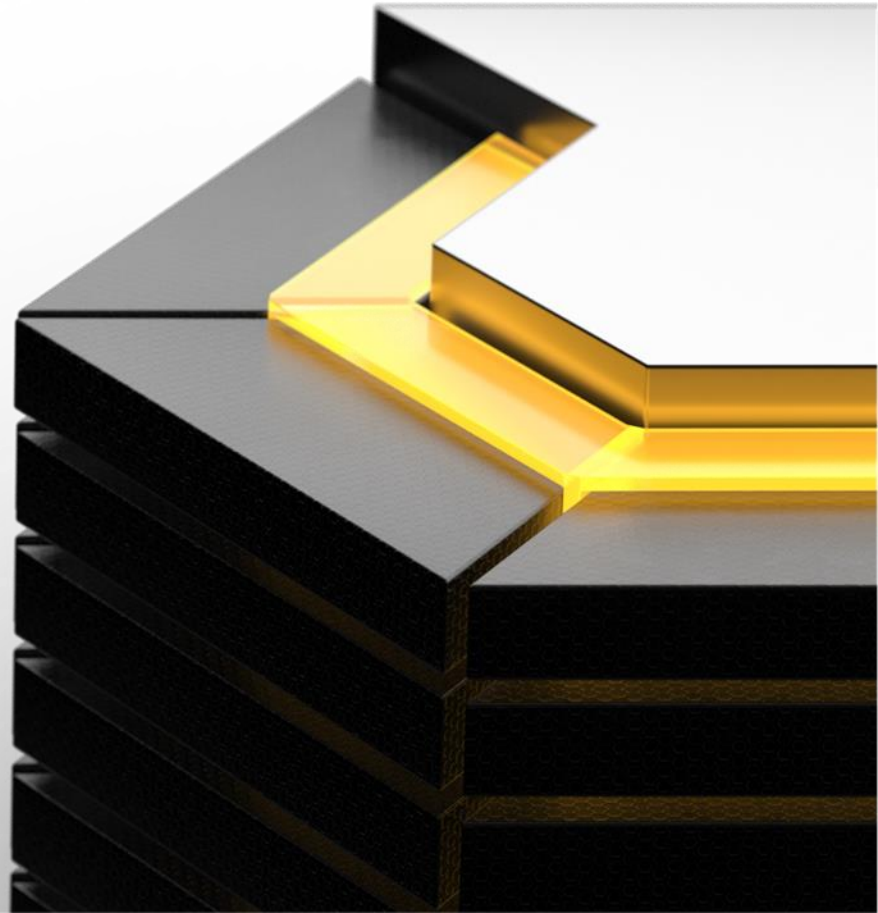
Altium

2. AUGMENTED REALITY IN CARS: PROS AND CONS OF ADAS HEADS UP DISPLAYS

Advantages of Heads Up Displays

As cars become more advanced, ADAS and heads-up displays (HUDs) will start to mesh in an attempt to bring augmented reality (AR) to cars. HUDs a few years from now will display information coming in from ADAS sensors and other more general information. This new kind of HUD will have a variety of big advantages, but will also bring a unique set of challenges that we need to start considering today. Advantages of ADAS HUDS include:

- The ability to highlight an object in the road that you should be aware of, bringing it to your attention and helping you avoid it.
- Vehicle to everything communication, in order to give the driver eyes and ears outside their own car.
- The ability to display traffic information in the driver's path and weather forecasts as well.

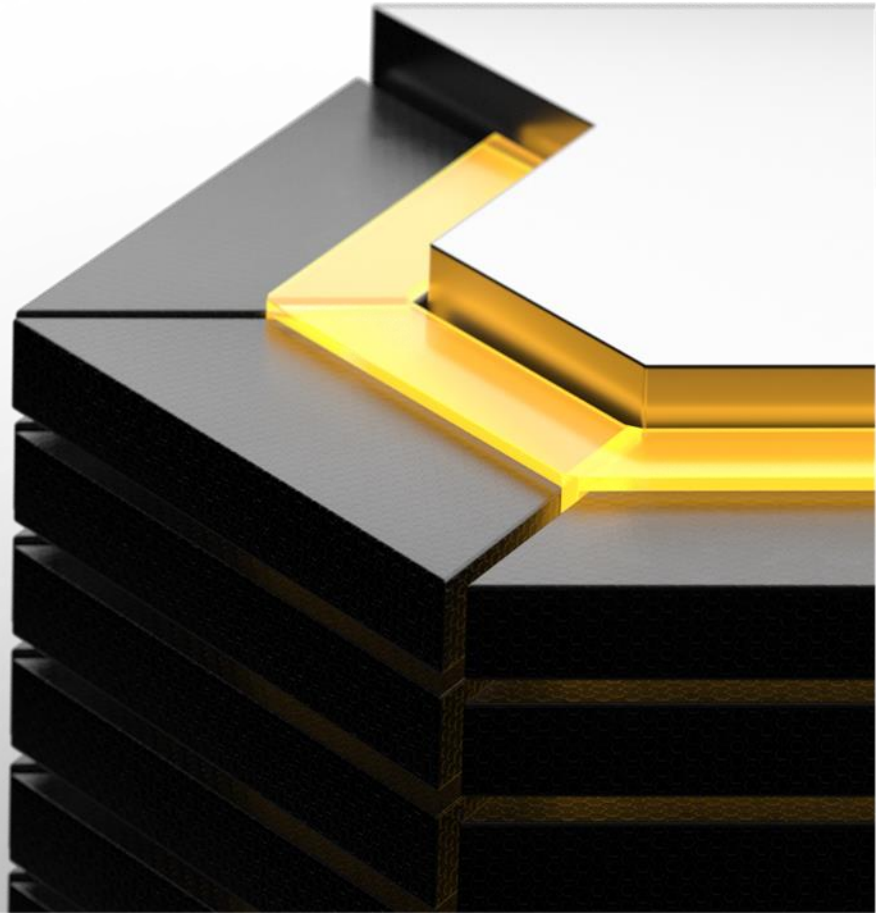


Disadvantages of Heads Up Displays

Critics of these kinds of these kinds of systems have a simple argument. Instead of decreasing driver distraction, HUDs add to the problem. If text messages, playlists, or even videos started popping up on your windshield, it's possible your attention would be diverted. Even life-saving information could become dangerous, depending on how it's presented. Some disadvantages of HUDs include:

- The possibility of distractions depending on how information is presented.
- The high cost of LIDAR sensors that can accurately see in a large 360-degree circle, which currently cost around $70,000.
- In addition to monetary cost, there will be processing and electricity costs as well. If automakers start incorporating HUDs into their cars, they'll need to have a processor that can perform all ADAS functions, and then display them.

[See original content here]



Altium

1. GROWING ADAS MARKET WILL REQUIRE EFFICIENT SOFTWARE AND MPU EXPERTISE
2. AUGMENTED REALITY IN CARS: PROS AND CONS OF ADAS HEADS UP DISPLAYS
3. WHICH ADAS FEATURES ARE MOST LIKELY TO LOWER PREMIUMS AND INCREASE PUBLIC SAFETY?
4. INLINE FUNCTIONS IN C LANGUAGE: WHY AND WHEN THEY ARE USED IN ADAS SOFTWARE
5. NEURAL NETWORK SUPERVISION AND ADAS VEHICLE SAFETY

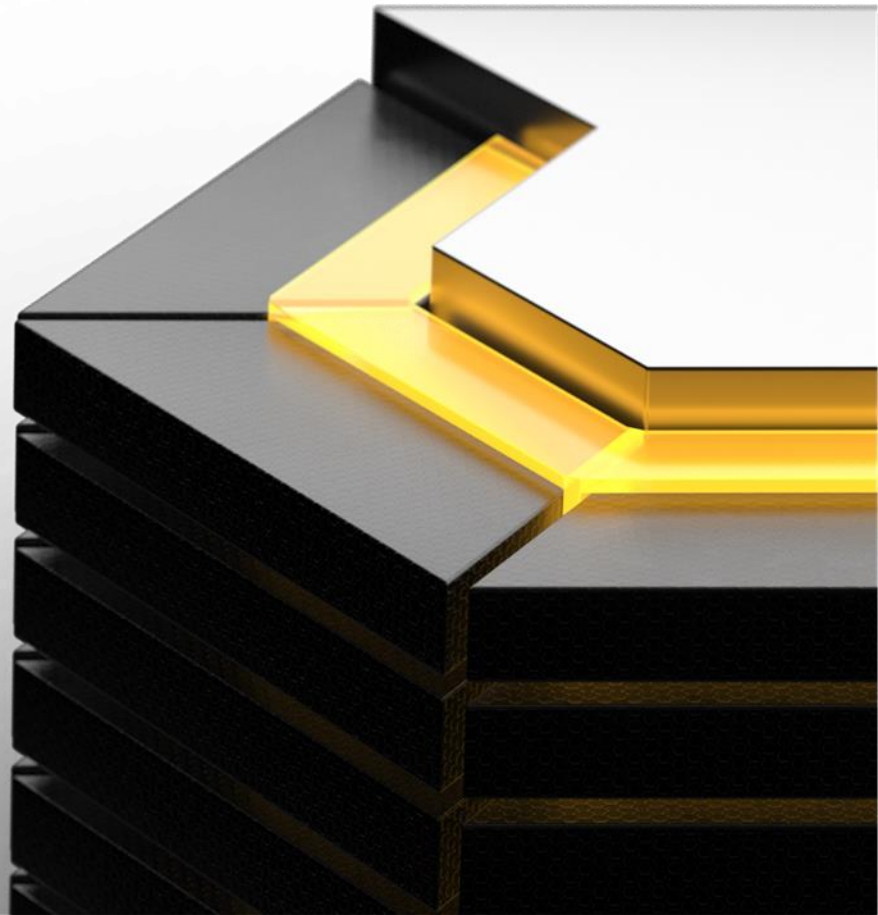
Altium

3. WHICH ADAS FEATURES ARE MOST LIKELY TO LOWER PREMIUMS AND INCREASE PUBLIC SAFETY?

ADAS Increasing Public Safety

In the near future, we could see fewer traffic jams, lowered emissions, and better overall road safety. As we see more advances in ADAS features, including advanced sensor technology, real-time traffic updates, and improved mechanical diagnostics, our roads will become safer. This, in turn, will lead to a number of economic and social benefits.

- ADAS uses vision sensors, radar, and ultrasound to monitor other vehicles, their speeds, pedestrians, stationary objects, and sudden changes in road quality.
- In many instances, these ADAS features catch and resolve potential safety threats before the driver is aware of them.
- Additionally, vehicle-to-vehicle communications will play a greater role in traffic navigation. As vehicles communicate with one another, they will have the power to adjust speed or change routes in order to clear a traffic jam entirely.
- Your insurance premiums could decrease. With new advances in ADAS features, the driver's capability will become less of a safety factor.
- Additionally, insurance companies can see detailed diagnostic reports in the event of an accident to determine who or what, exactly, was at fault.



See original content here

Altium

1. GROWING ADAS MARKET WILL REQUIRE EFFICIENT SOFTWARE AND MPU EXPERTISE
2. AUGMENTED REALITY IN CARS: PROS AND CONS OF ADAS HEADS UP DISPLAYS
3. WHICH ADAS FEATURES ARE MOST LIKELY TO LOWER PREMIUMS AND INCREASE PUBLIC SAFETY?
4. INLINE FUNCTIONS IN C LANGUAGE: WHY AND WHEN THEY ARE USED IN ADAS SOFTWARE
5. NEURAL NETWORK SUPERVISION AND ADAS VEHICLE SAFETY

Altium

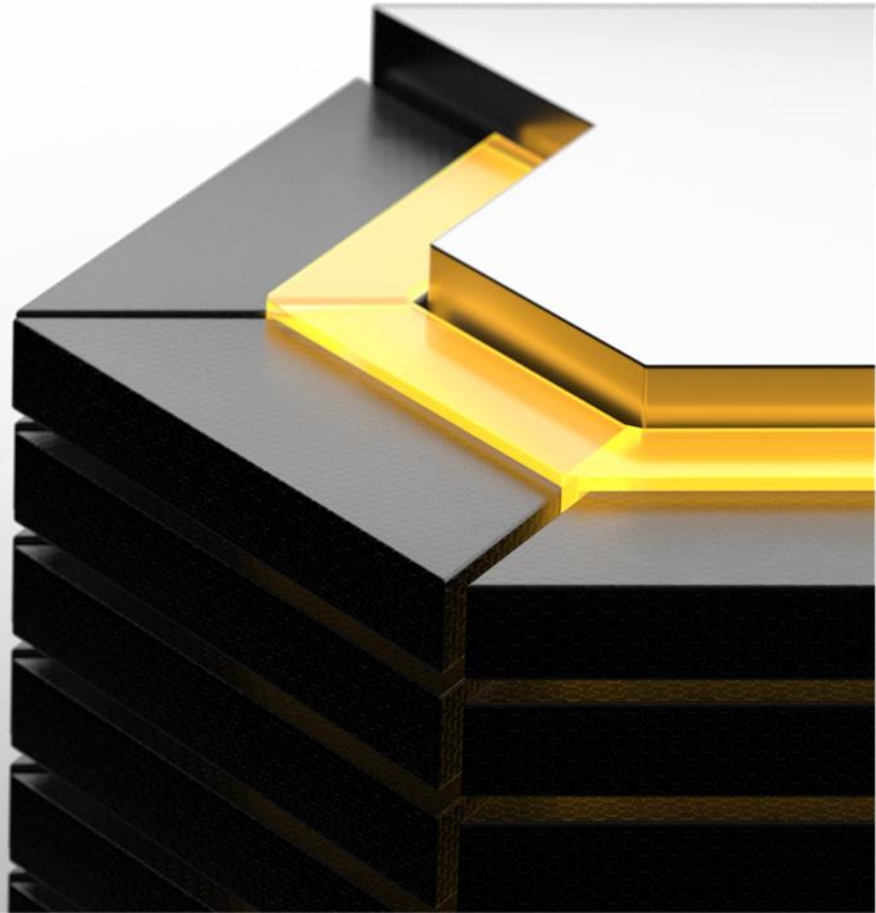
4. INLINE FUNCTIONS IN C LANGUAGE: WHY AND WHEN THEY ARE USED IN ADAS SOFTWARE

Altium

Inline Functions in C Language

Tools like function inlining can help speed up your program but may use extra space. That's why you need to carefully choose which sections of code to inline. If you don't, you might end up using extra space without increasing your performance. So what exactly are inline functions?

- Normally you call functions. The program then goes to that function, performs its operations, and returns a value or values.
- When functions are inlined, the compiler substitutes the function itself for the call of the function.
- This can speed up your software because you don't waste time calling a function and then returning its values.
- Just like loop unrolling, function inlining brings up the classic time vs. space conundrum. You need to find the delicate balance between the two when programming for advanced driver assistance systems (ADAS).
- You always need to carefully choose which sections of code you want to optimize to make sure you don't trade too much space for speed.

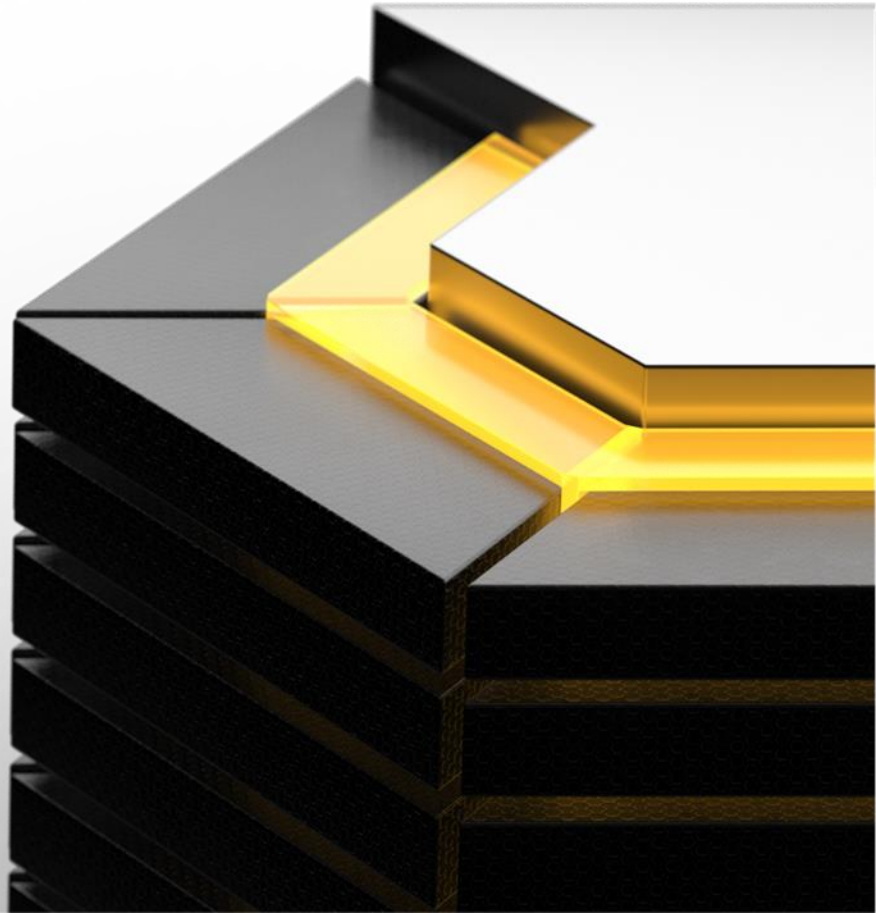


Inline Functions in C Language

There are two main types of functions you should try to inline if possible: short functions and static functions. In the opposite direction, you should probably not inline long functions or ones that are repeated often.

- If a function is about as many lines long as its call, you should definitely inline it. The compiler will replace the call with a function that's relatively the same size but gives you a speed boost.
- Static functions are also great to inline. They are especially great to inline when they have only one call.
- The primary disadvantage of inlining functions is that it takes up more space, especially if the function you're inlining is quite long. If the function is extremely lengthy, you could be adding more lines than is helpful.
- Functions that are called very often can also be a problem. You don't want to inline those functions because you're essentially copying the same function over and over into your code.

[See original content here]



Altium

1. GROWING ADAS MARKET WILL REQUIRE EFFICIENT SOFTWARE AND MPU EXPERTISE
2. AUGMENTED REALITY IN CARS: PROS AND CONS OF ADAS HEADS UP DISPLAYS
3. WHICH ADAS FEATURES ARE MOST LIKELY TO LOWER PREMIUMS AND INCREASE PUBLIC SAFETY?
4. INLINE FUNCTIONS IN C LANGUAGE: WHY AND WHEN THEY ARE USED IN ADAS SOFTWARE
5. NEURAL NETWORK SUPERVISION AND ADAS VEHICLE SAFETY

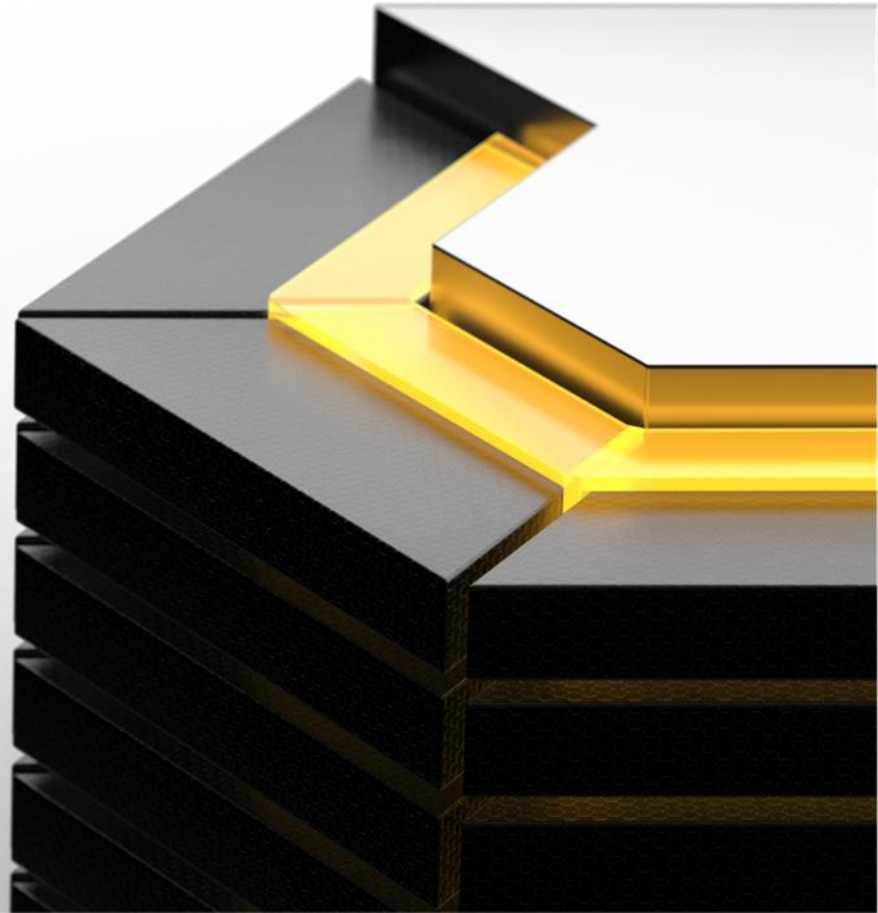
Altium

5. NEURAL NETWORK SUPERVISION AND ADAS VEHICLE SAFETY

Advantages & Disadvantages of Neural Networks

The only problem with giving machines intelligence is that it opens them up to failure. If vehicles can think and deduce, it means they could possibly make an incorrect decision. This possibility has to be addressed before cars with advanced driver assistance systems (ADAS) powered by neural networks can hit the roads.

- The inherent advantage of neural networks is that they can remember how to react to old situations yet are flexible enough to deal with new ones as they arise.
- However, that flexibility means that we cannot be certain how they will react to novel inputs. In terms of safety, this makes neural networks hard to certify and not suitable for critical safety operations.
- A vehicle that relies solely on rules and traditional sensors may not be able to make that decision. A system that can learn should be able to follow or break the rules in order to maintain its safety.
- There is no way to guarantee that a neural network will always make the right decision by itself.

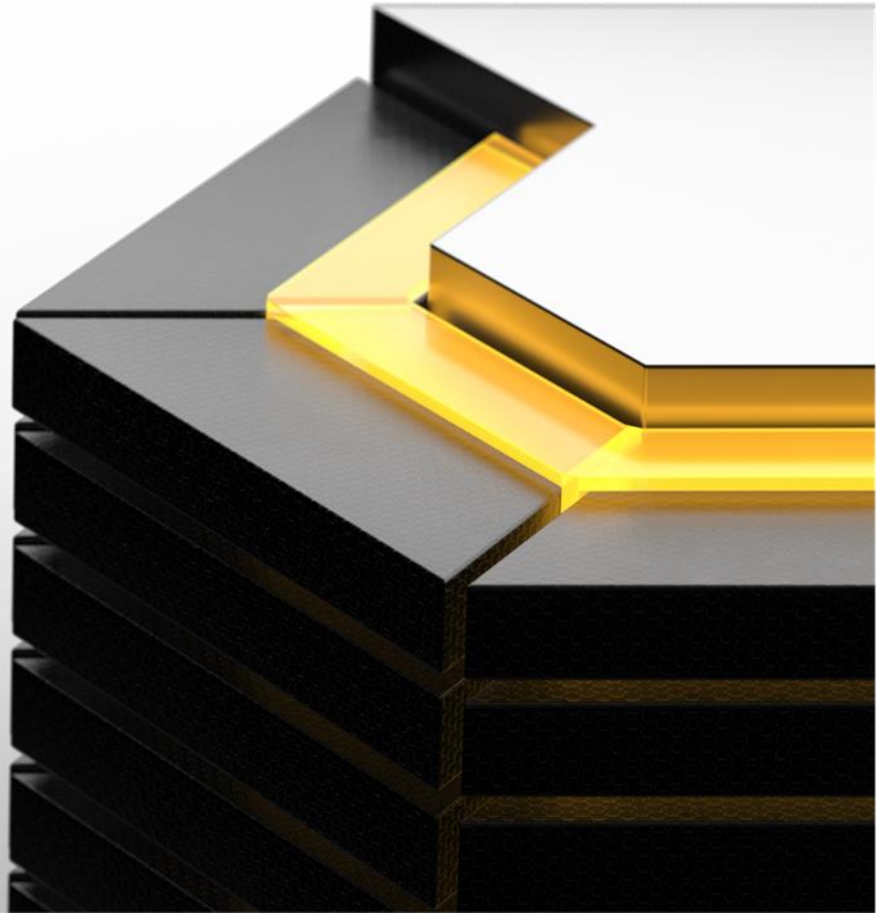


Neural Network Supervision

You can't let a neural network make decisions alone, but you can let it suggest actions that are verified by a supervisor. Conventional methods that can be safety certified can be used to validate a neural network's suggestions.

- Traditional sensors can set hard limits on a neural network's decisions by checking against some basic rules.
- Using multiple sensor fusion will combine data streams for decision making so you can pair your neural network, which will be rated at a relatively low ASIL level, with a system rated at the highest level for checking.
- The advantages that neural networks have to offer are too great for us to ignore the complexity of their implementation.

[See original content here]



Altium

Thanks for your attention!