

Realtidsstöd i Autosar

Kandidatarbete Marko Tapani Stålhammar
Institutionen för Informationsteknologi vid Åbo Akademi
Hösten 2013 Åbo

INNEHÅLLSFÖRTECKNING

| | |
|--|----|
| 1.INLEDNING..... | 3 |
| 2.BAKGRUND..... | 4 |
| 3.AUTOSAR OCH DESS UTVECKLING..... | 4 |
| 3.1Fördelar med Autosar..... | 6 |
| 3.1.1Motiveringar och mål för Autosar från autosar.org:..... | 6 |
| 3.2Medlemmarna..... | 6 |
| 3.3Medverkan med andra organisationer..... | 8 |
| 4REALTIDSYSTEM..... | 8 |
| 5DESIGN..... | 9 |
| 5.1Funktionell Säkerhet..... | 9 |
| 5.2Arbetsflöde..... | 10 |
| 6ARKITEKTUR..... | 11 |
| 7Slutsats..... | 12 |
| Källförteckning..... | 14 |

Referat

Autosar har blivit en av de viktigaste standarderna för mjukvarutillverkning för inbyggda datorsystem inom bilindustrin idag. Bilar blir mera komplexa hela tiden, innehåller mera elektronik och ställer därmed nya krav och utmaningar för mjukvarutillverkarna. Billeverantörer har redan sina egna processer, system och konfigurationer som måste underhållas. Med introduktionen av Autosar har man fått en standard för att underlätta denna process.

1. INLEDNING

Det finns ett generellt intresse av bilar och elektroniken i dem, så Realtidsstöd i Autosar som rubrik var ett relativt lätt val.

Autosar (AUTomotive Open System ARchitecture) är en standardiserad open source arkitektur för moderna bilarnas elektronik. Man kan säga att Autosar är ett operativsystem för fordon. I denna uppsats kommer jag att presentera hur Autosar kom till, hur den fungerar, varför den behövs, vem alla är med i projektet och hur den lämpar sig för realtidssystem som tex. ABS-bromsar eller krockkuddar.

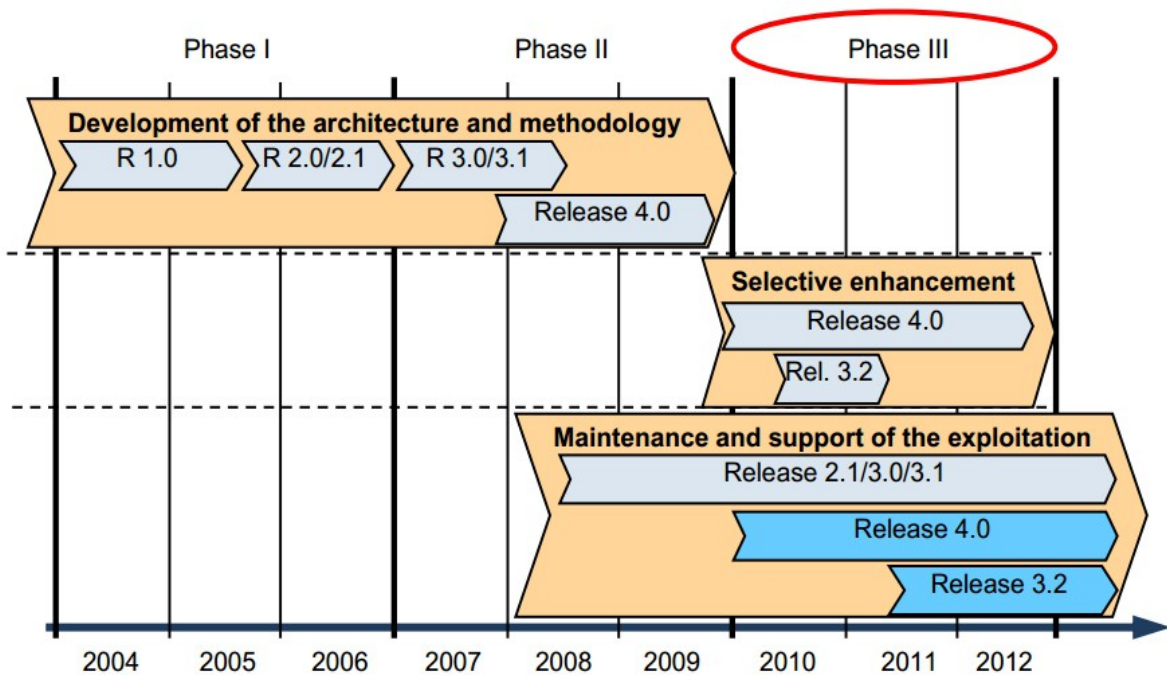
2. BAKGRUND

Moderna bilar börjar ha tiotals olika datorer för olika komponenter. Autosar ersätter dessa små datorer. Detta betyder att komponentleverantörerna inte mera behöver leverera en dedikerad dator för att styra sin komponent (tex. fönsterhissar), utan istället levererar de mjukvaran som körs i Autosar.

Tack vare Autosars design kan man använda individuella självständiga mjukvarukomponenter, som medför det att man kan återanvända samma komponenter i olika fall i framtiden (mellan olika leverantörer och plattformar tex.).

Autosar äger inte själv några implementationer, utan specificerar bara gränssnitten. Olika företag gör sina egna implementeringar, som de sen säljer åt biltillverkarna.

3. AUTOSAR OCH DESS UTVECKLING



Figur1: Autosar timeline (Källa: autosar.org)

Autosar grundades i juli 2003 av *BMW*, *Bosch*, *Continental*, *DaimlerChrysler*, *Volkswagen* och *Siemens VDO*. Senare kom också *Ford Motor Company* (november 2003), *Toyota* och *PSA* (Peugeot Citroen) (december 2003) och *General Motors* (november 2004) med.

I feburari 2008 blev *Siemens VDO* en del av *Continental* och sen dess har inte *Siemens VDO* mera räknats som en självständig partner av Autosar.

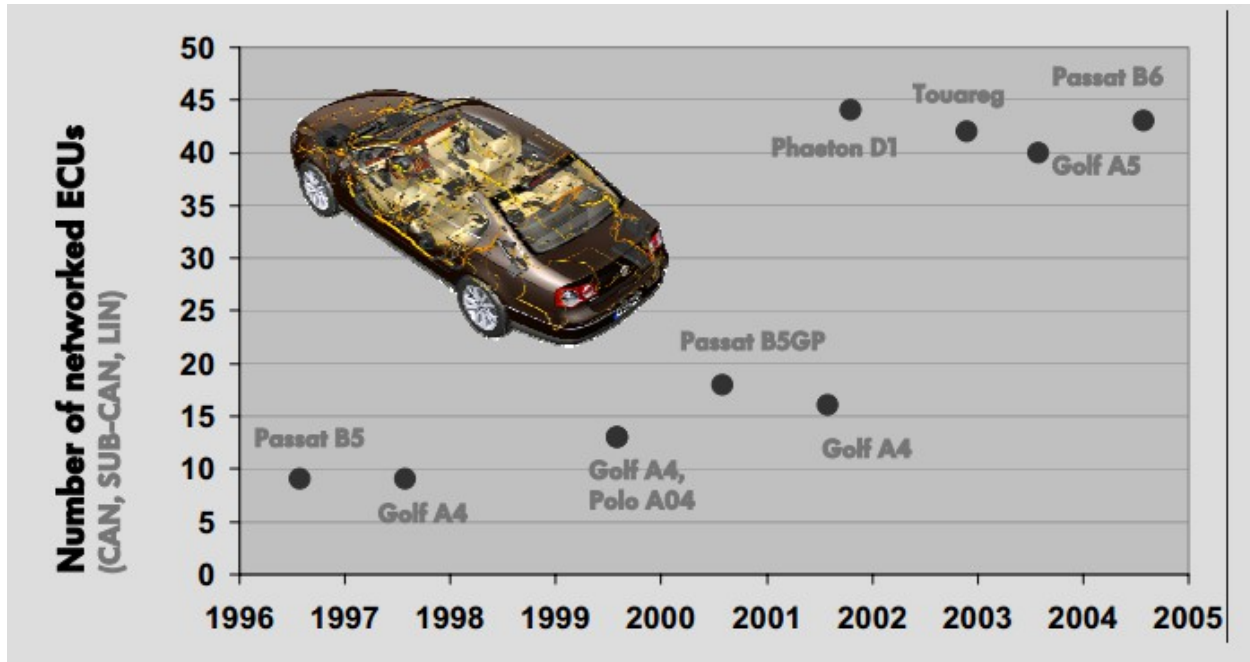
Autosars utveckling har redan gett ut flera versioner sen 2004 då den första versionen Fas1 kom ut (Fas 3 pågår nu)

Fas 1 (2004-2006): Själva standardutvecklingen börjar. I slutet av år 2006 med release 2.1 färdigställdes några av de största specifikationerna.

Fas 2 (2007-2009) : Genom expansioner på arkitekturen och metologin fick standarden mera funktionalitet för att möta marknadens krav. (version 3.0/3.1)

Fas 3 (2010-2012): Underhållning och förbättring av underhållbarheten av olika versioner för serieproduktion (version 4.0). [1]

3.1 Fördelar med Autosar



Figur 2: Mängden av ECU:n i bilar (Källa: Autosar.org)

En standardiserad arkitektur underlättar implementeringen av framtidens komplexa komponenter genom att använda uppdaterbara mjukvarukomponenter. Detta leder till ett mycket flexibelt system utan så stora problem med komponenternas kompatibilitet. Alla fordon med Autosar har samma källkod för själva gränssnittet för basmjukvaran. Det betyder att man sparar tid och pengar då man börjar planera en ny bil och inte behöver slösa mera flera månader på att utveckla källkod skillt för den nya bilen.

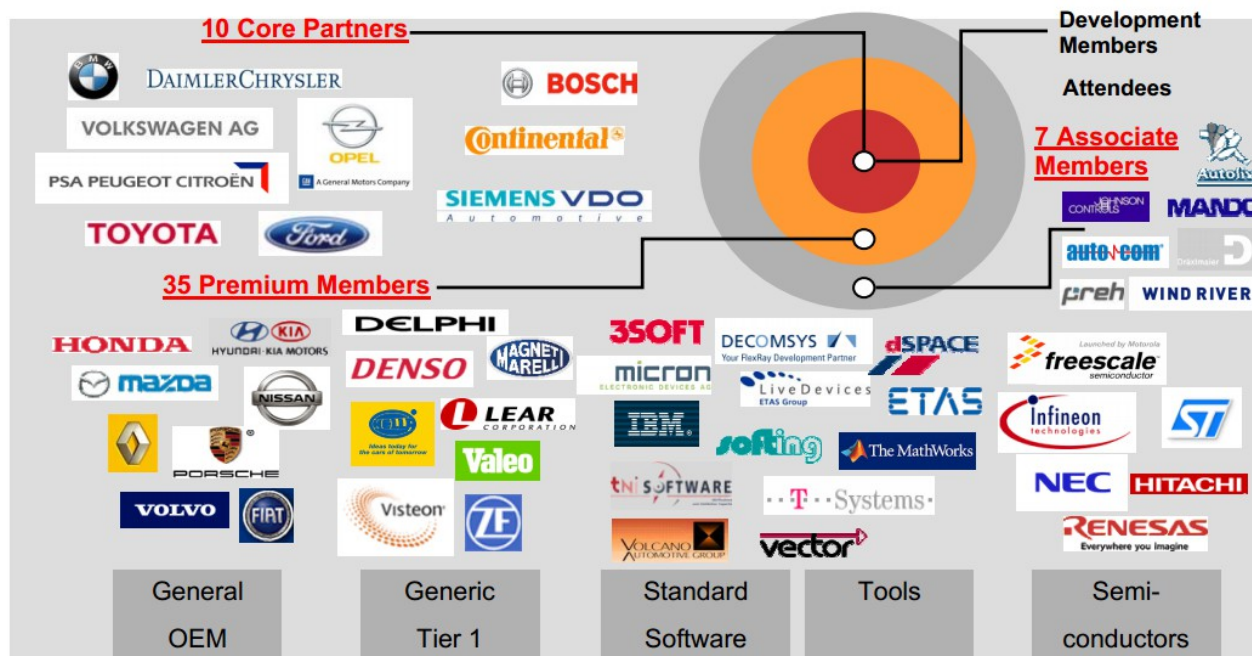
Mjukvaruutvecklingen och servicen blir alltså billigare i det långa loppet. Med mjukvaruupdateringar i framtiden går det att ge fordonet nya eller bättre egenskaper.

3.1.1 Motiveringar och mål för Autosar från autosar.org:

Autosar utvecklingsmedverkans viktigaste mål är standardiseringen av elementära systemfunktioner och funktionella gränssnitt, förmågan att integrera och använda systemfunktioner inom bilnätverket och att ge ut uppdateringar och uppgraderingar för bilen genom hela dess livstid.

[2] [1] [3]

3.2 Medlemmarna



Figur 3: Medlemmarna (Källa: Autosar.org)

Autosar utvecklings samarbetet består av OEM (Original Equipment Manufacturer) tillverkare, bil leverantörer och företag inom bil elektronik och mjukvara.

Autosar kan delas i 3 delar; Core Partners, Premium Members och Associate Members, som alla delats olika ansvarsområden och rättigheter inom företaget. Dessutom finns det ännu olika kategorier inom de 3 huvudgrenarna.

P = Premium, A = Associate, D = Development, Att = Attendee.







| | Tillgång till teknologin utan licensavgifter | Gratis licens till alla "Automotive" program | Tillgång till senaste information och specifikationer | Ledarskap | Samarbete med olika arbetsgrupper | Anställer personal |
|-----|--|--|---|-----------|-----------------------------------|--------------------|
| P | X | X | X | X | X | X |
| A | X | X | X | | | |
| D | X | | X | | X | X |
| Att | | | X | | X | |

Figur 4: Tabell över medlemskaparnas rättigheter

Dessutom måste årsavgift betalas av alla medlemmar förutom Development och Attendees.

Premium kostar 17500€/år och Associate 10000€/år. [2] [1]

AUTOSAR Roll-Out
The AUTOSAR Core Partner Exploitation Plan (2008 - 2012)

| Core Partner | 2008 | 2009 | 2010 | 2011 | 2012 |
|--|---|--|---|---|--|
| BMW Group  | <ul style="list-style-type: none"> ■ ≈10 AUTOSAR BSW modules as part of Std Core in vehicles, tool / serial support in place | | | <ul style="list-style-type: none"> ■ Powertrain-, Chassis-, Safety-, Body- ECUs use AUTOSAR architecture | |
|  BOSCH | <ul style="list-style-type: none"> ■ Body Computer with subset of AUTOSAR specs incorporated ■ Instrument Cluster with subset of AUTOSAR specs incorporated | <ul style="list-style-type: none"> ■ ACC ECU using AUTOSAR architecture. ■ Powertrain EDC/ME(D)17 ECUs using AUTOSAR architecture ■ Domain Control Unit using AUTOSAR BSW | <ul style="list-style-type: none"> ■ Chassis ECU using AUTOSAR architecture ■ Body Computer using AUTOSAR architecture | | |
|  Continental | <ul style="list-style-type: none"> ■ Complete BSW Stack as Product ■ AUTOSAR Configuration Tool | <ul style="list-style-type: none"> ■ Body ECUs using AUTOSAR architecture ■ Powertrain ECUs using AUTOSAR architecture | <ul style="list-style-type: none"> ■ Chassis ECUs using AUTOSAR architecture | | <ul style="list-style-type: none"> ■ Engine Systems Platform based on AUTOSAR architecture |
| DAIMLER | | | <ul style="list-style-type: none"> ■ First usage of AUTOSAR modules in vehicles | <ul style="list-style-type: none"> ■ First AUTOSAR compatible ECUs in vehicles | <ul style="list-style-type: none"> ■ Introduction of AUTOSAR architecture and methodology in vehicles |
|  | | <ul style="list-style-type: none"> ■ 1-2 AUTOSAR conformant ECUs; first use of conformant tools/methodology | <ul style="list-style-type: none"> ■ Continuous roll-out of ECUs into vehicle architecture increased use of conformant tools / methodology | | |
|  | | | <ul style="list-style-type: none"> ■ First usage of AUTOSAR modules | <ul style="list-style-type: none"> ■ First use of AUTOSAR architecture ECU | |
| PSA PEUGEOT CITROËN  | | <ul style="list-style-type: none"> ■ Powertrain ECU using AUTOSAR architecture | <ul style="list-style-type: none"> ■ Body ECU using AUTOSAR architecture | | |
| TOYOTA | | | <ul style="list-style-type: none"> ■ First usage of AUTOSAR modules | | <ul style="list-style-type: none"> ■ AUTOSAR Architecture ECU |
| VOLKSWAGEN AG | | <ul style="list-style-type: none"> ■ First AUTOSAR modules in series production | | <ul style="list-style-type: none"> ■ First complete ECUs in series production | |

Figur5: Autosar medlemmarnas uppgifter (Källa: autosar.org)

3.3 Medverkan med andra organisationer

Autosar har samarbete också med andra företag som håller på med samma sak. Jaspas är ett företag från Japan som Autosar har nära samarbete med och flera av Jaspas idéer och funktionalitet kommer att integreras till Autosar release 4.0.4. Autosar och Jaspas har gemensamma "high level" meetings två gånger per år; Autosar conference 2010 organiserades i Tokyo med stöd av Jaspas.

Kina och Indien har också en växande efterfrågan för en internationell standard och Autosar håller på med diskussioner om samarbete med lokala företag. Genom hjälp av lokala 'Core Partners' medlemsföretag i Kina och Indien har Autosar redan börjat med samarbete på låg nivå.

[2] [1] [4]

4 REALTIDSYSTEM

Et system löser problem. Som problem kan vi ta smutsiga kläder. Ingen vill tvätta kläder för hand, så man har uppfunnit en maskin som löser problemet; en tvättmaskin. Då man startar tvättmaskinen händer många saker: motorkontroll, intagning av vatten, olika program för olika slags kläder, temperaturreglering på vattnet, kanske en statusindikator i form av en lcd-panel och systemet kan sluta operera om man öppnar luckan mitt under tvättprogrammet mm. Allt detta styrs av en mikrokontroller.

Mikrokontrollern i detta fall har flera olika uppgifter, jobb (engelska: task) som består av olika instruktioner för funktionen i fråga. Tvättmaskinen kan ha tex. följande jobb: Vattenkontroll, motorkontroll, skärmkontroll och avbrytarkontroll (i fall något oväntat händer).

Mikrokontrollern kör olika jobb samtidigt (kallas multitasking).

Viktigaste kriteriet i ett realtidssystem är att jobb körs i tid. Varje uppgift har en deadline, dvs. de har en tid som de måste bli färdiga inom. Vattenkontrollen måste tex. reagera genast då maskinen har tillräckligt vatten i sig, annars kan maskinen fyllas med vatten tills den svämmar över.

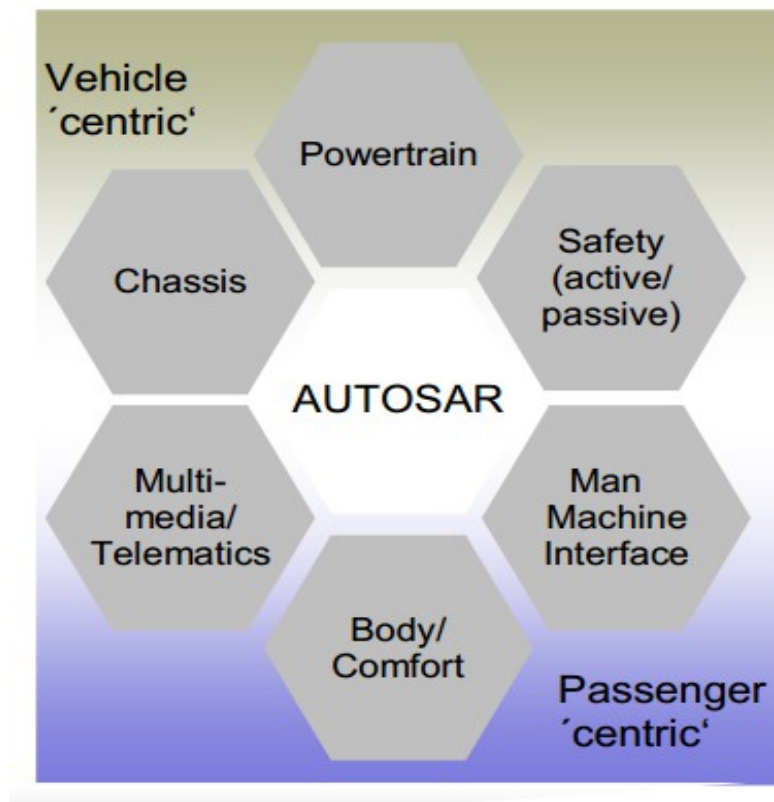
Det finns så kallade hårda och mjuka system. Skillnaden är att i ett hårt realtidssystem får deadlines absolut inte missas, för det kan oftast medföra något fatalt fel medan i mjuka realtidssystem är det inte så farligt om deadlines går lite över. Vissa jobb har mera tyngd än andra, dvs. det finns olika grader av hårdhet/mjukhet. Oftast är tex. säkerhetsrelaterade jobb viktigaste.

En del jobb är viktigare än andra och en del måste köras i en viss ordningsföljd (schemaläggning) för att systemet skall fungera på önskat sätt. I vårt exempel med tvättmaskinen har vattenkontrollen högre prioritet än tex skärmkontrollern: Vattenflödet stängs av först och efter det visar lcd-panelen meddelandet om vattnet. Detta för att undvika något katastrofalt som kanske vatten på golvet bara för att det var något fel på skärmkontrollern (den missade sin deadline och vattenkontrollen startade därmed för sent).

När man funderat ut tidsramarna för alla jobb, kan man schemalägga systemet så att alltid när ett jobb med högre prioritet vill göra något, kan den få göra det fast andra jobb (av mindre prioritet) pågår samtidigt. Jobb med mindre prioritet väntar vid behov tills högre prioriteten har blivit färdig.

5 DESIGN

Autosars räckvidd innehåller alla områden i bilen, som till exempel själva karossen, drivlinan (dvs. Komponenterna som driver bilen framåt: motor, koppling, växellåda och kardanaxel.), bromsar, farthållare, GPS- navigation mm. Från och med Fas II (2007 – 2009) har också fotgängarsäkerheten och passagerarnas säkerhet tagits i beaktande.



Figur 6: Områden (Källa: Automotive Spin Italia)

5.1 Funktionell Säkerhet

Funktionell säkerhet är en mycket väsentlig sak inom Autosar. Som exempel så får inte krockkuddarna plötsligt aktiveras om något går fel i funktionen som styr dem (dvs. ingen riktig fara finns).

Det finns olika så kallade **Automotive Safety Integrity Level (ASIL)** som går antingen från ASIL1(minsta) till ASIL4 (största), eller från ASIL A (minsta) till ASIL D (största).

Till vilken ASIL grupp som en funktion hör till definieras med hjälp av hazard analys dvs. man ser på funktionen och ställer frågan ”vad händer för passageraren/fotgängaren/andra som använder vägen där bilen befinner sig om funktionen (tex. krockkuddarna) inte fungerar?”. [5]

Efter att man fått reda på vilken ASIL grupp funktionen hör till, ges funktionen ett säkerhetsmål (”safety goal). Enligt ISO 26262 har alla ASIL grupper sina egna testningsmetoder som bör användas för testning. [5]

Till exempel krockkuddarna. Säkerhetsanalysen kommer att bestämma vilken effekt en plötsligt av sig själv aktiverad krockkudde har på föraren. Kommer han till exempel att se något ut från fönstret? Blir han skrämmd och gör något farligt som reflex?

Vid detta katastrofala fall kan man anta att det är frågan om en ASIL D för chauffören inte kommer att se något och situationen blir därmed livsfarlig.

Krockkuddar är mycket krävande komponenter, för de måste alltid fungera fast något går fel. De får inte gå till ett avstängt läge som något annat system kanske får göra.

5.2 Arbetsflöde

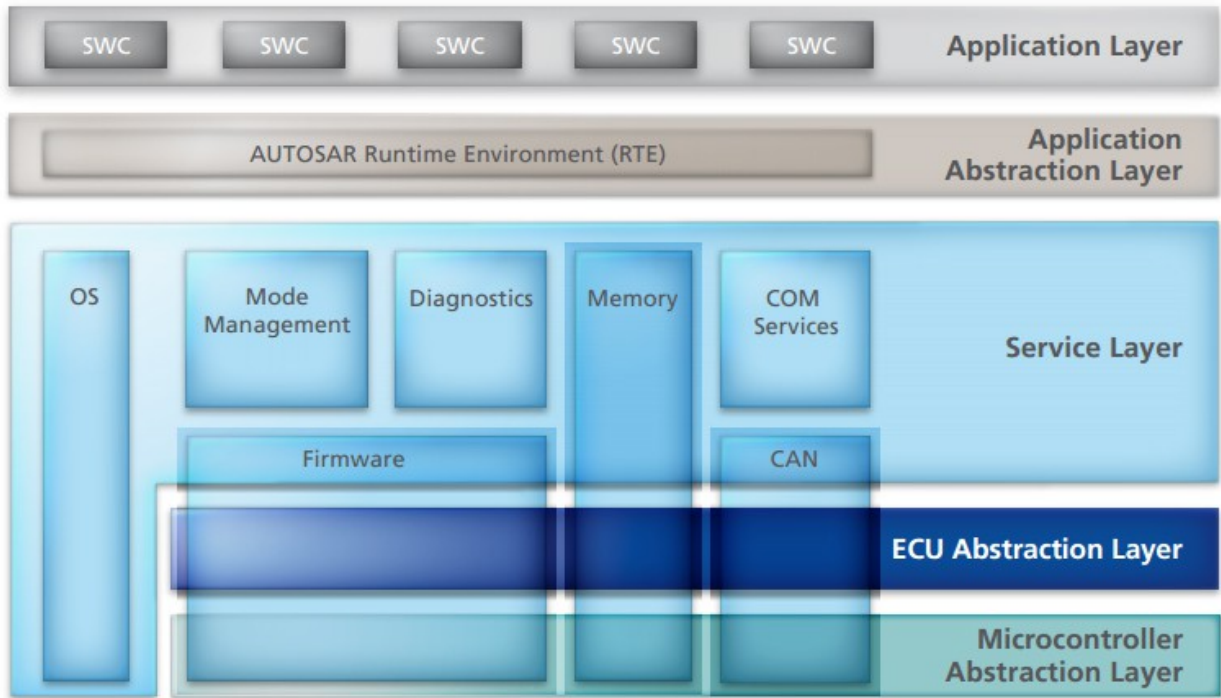
För att fortsätta med exemplet med krockkuddar så kan vi se hur en modern krockkudde fungerar:

Idén med en krockkudde är att skydda passagerarna (men det finns även krockkuddar för fotgängare) från skada vid en olycka med att en nylonkudde fylls med gas på några millisekunder och hindrar personen att slå tex. ansiktet på hårda ytor.

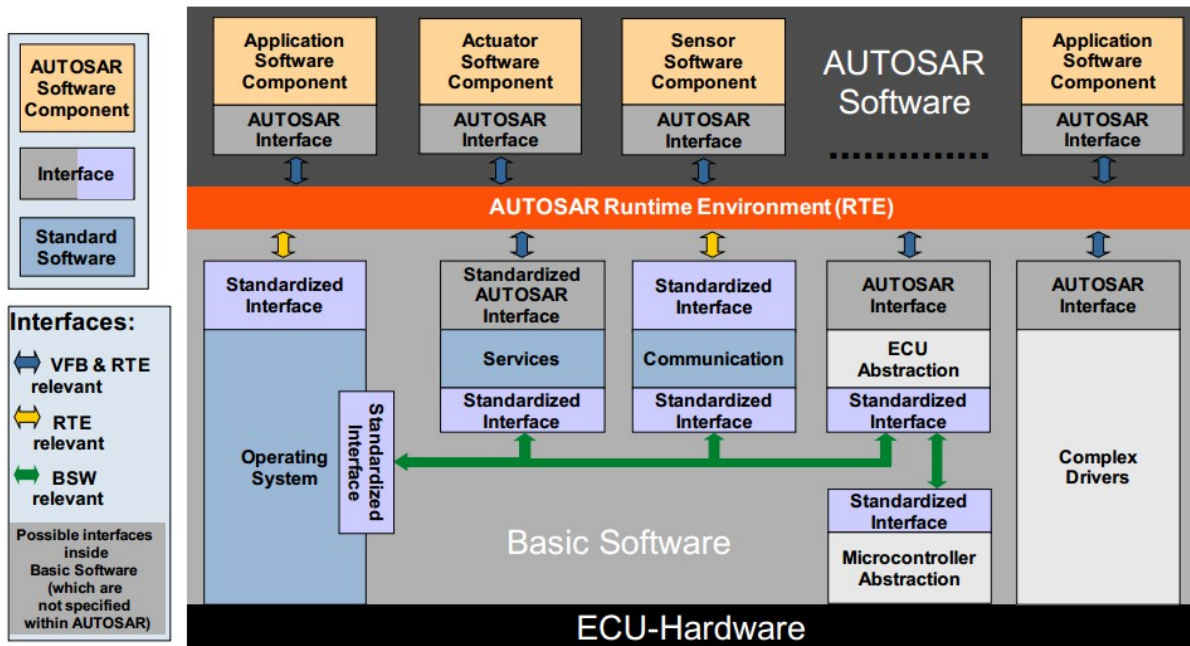
Krockkuddar är fyllda med natriumsyrapulver som förvandlas mycket snabbt till nitrogengas (ungefär 30 millisekunder). Krockkuddar styrs av en ECU, som monitorerar signaler från små accelerometrar som detekterar kraftiga ändringar i hastigheten. Om ECU:n detekterar en hastighetsförändring som är inom dess parametrar för kollision, skickar den en signal för att slå ut krockkuddarna. Efter ungefär en sekund från utlösandet av krockkudden börjar den tömmas.

Autosar workflow fungerar på följande sätt. Först måste OEM definiera alla parametrar för sin ECU, dvs. precis allt som behövs för att få komponenten att fungera så som den skulle utan att vara kopplad till Autosar. Resultatet blir en så kallad 'Autosar system description' fil.

Av denna systembeskrivning plockar sedan andra ECU tillverkare väsentlig information för att passa in med deras egna parametrar som behövs för denna specifika modul. Detta kallas 'ECU extract' av systembeskrivningen. Dessa beskrivningar kan man sedan använda som bas till en Matlab/Simulink modell varifrån kod kan genereras. [6]



Figur 9: Autosar mjukvarulager. (Källa: dSpace.com)



Figur4: Autosar arkitektur, komponenter och interfaces (Källa:)

Autosars arkitektur består av 3 större lager av mjukvaruabstraktion som körs på en mikrokontroller: Application Software, Runtime Environment (RTE) och Basic Software (BSW). Basic Software är dessutom delat in i följande lager: Services, ECU abstraktion, mikrokontroller abstraktion och complex drivers.

Komplexa drivrutinerna finns för att ge möjlighet att implementera speciella drivrutiner för apparatur som inte är definierat inom Autosar eller måste tex. reagera mycket snabbt (=genast). Viktigaste uppgiften för komplexa drivrutiner är att implementera komplexa sensor evalueringar och regleringskontroll direkt med mikrokontrollern, dvs. utan att gå via Autosars Basic Software lager först. Komplexa drivrutiner används bl.a. med injektionskontroll och elektrisk ventilkontroll.

Komplexa drivrutiner kan också användas för att implementera drivrutiner för komponenter som inte stöds av Autosar. Om tex. något nytt slag av kommunikationssystem introduceras, finns det oftast inte någon färdig drivrutin inom Autosar som kan stöda dess funktionalitet direkt. Då

implementeras det en egen skyddad drivrutin inom komplexa drivrutinerna, som sedan kallas direkt i stället för att använda kommunikationslagret i Autosar.

RTE lagern skiljer Application och Basic Software lagren från varandra och sköter om informationsflödet mellan dem. Lagret ovanför RTE är komponentformad dvs. består av olika mjukvarukomponenter (Software Components eller SWC). Alla dessa mjukvarukomponenter är oberoende av själva ECU:n. Mjukvarukomponenterna kan implementeras utan att veta vilken slags ECU bilen kommer att innehålla. Mjukvarukomponenterna kan också användas i andra projekt som kanske inte har en likadan ECU.

[8][7][4]

7 Slutsats

Autosar har sina fördelar, men också nackdelar finns. Autosar passar inte för komponenter/komponentsystem som måste ha mycket snabb respons, för att i ett så stort system som styr allt på en gång kan det lätt förekomma problem med minneaccess, latens på grund av buffert-problem mm och det igen kan leda till missade deadlines och det kan vara farligt. Om tex. krockkudden inte aktiveras i tid för att något minnesfel har skett.

Riktigt små system implementerat med Autosar kräver onödigt mycket minne och processortid. I större och mera komplicerade ECU:n blir det nyttigare att använda Autosar.

Källförteckning

- [1] B. G. Simon Fürst, "AUTOSAR – A Worldwide Standard is on the Road.," [Online].
- [2] F. Leitner, "Universität Konstanz: Fachbereich Informatik und Informationswissenschaft," 2008. [Online]. Available: <http://www.inf.uni-konstanz.de/soft/teaching/ws07/autose/leitner-autosar.pdf>.
- [3] A. M. C., "Automotive Spin Italia," [Online]. Available: <http://www.automotive-spin.it/>.
- [4] "The AUTOSAR development partnership," 2012. [Online]. Available: <http://autosar.org>.
- [5] "National Instruments: What is the ISO26262 Functional Safety Standard?," [Online]. Available: <http://www.ni.com/white-paper/13647/en>.
- [6] d. G. Joachim Stroop, "dSpace," [Online]. Available: www.dspace.com.
- [7] Alberto Sangiovanni-Vincentelli University of California, Berkeley, Marco Di Natale Scuola Superiore S. Anna, Pisa. " : Embedded System Design for Automotive Applications
- [8] Manfred Broy, Mario Gleirscher, Stefano Merenda, and Doris Wild, *Technische Universität München* Peter Kluge and Wolfgang Krenzer, *IBM Deutschland GmbH* : "Toward a Holistic and Standardized Automotive Architecture Description "