Deelvraag 2 welke open source RTOS zijn beschikbaar?

Criteria : beschikbaar voor ARM

C en C++ ondersteunend.

Afgelopen jaar geüpdatet

Documentatie over functionaliteit.

Free rtos :

Mbed\*

Chibi-OS rt

RIOT

Atom threads

Deelvraag 2.

In het geval van open source RTOS-en zijn er vele verschillende mogelijkheden Een RTOS kan gemaakt zijn voor een speciale taak, aparte hardware, specifiek werk afdeling, ondersteuning van specifieke programmeertalen en dergelijke. Kijkende naar RTOS-en met een aantal eigenschappen zoals het ondersteunen van ARM controllers waarop de RTOS op zal draaien, het compatibel zijn met C en C++ programmeertalen waardoor met deze programmeertalen de verdere software kan worden geschreven. Ook moet de RTOS recent zijn geüpdatet (maximaal jaar geleden) en waardevolle documentatie over de functionaliteiten. Zijn er nog steeds vele opties met de specificaties waar we ons op hebben gericht.

Als eerst hebben we FreeRTOS, gemaakt door Real Time Engineers Ltd. en is vrijgegeven onder de GNU general public license . deze RTOS ondersteunt 35 verschillende microcontrollers en bestaat al 12 jaar waarin de jaren de RTOS meervoudig is aangevuld en verbeterd. FreeRTOS in de loop der jaren uitgegroeid tot 1 van de markteider onder de RTOS-en. FreeRTOS is ontworpen om kleinschalig te zijn en simpel te werken hierdoor heeft FreeRTOS een kleine memory foodprint en heeft een snelle uitvoering. De code van FreeRTOS heeft als voordeel dat het bestaat uit 3 .C bestanden voor de RTOS, hierdoor is het overzichtelijk om de code te bekijken en vooral om te vinden waar welke functie staat. Er zijn ook RTOS-en waarbij de functies in veel verschillende files bevinden het onoverzichtelijk wordt om te bekijken. De overige files in het .zip file van FreeRTOS zijn demonstratie files waarbij. FreeRTOS bevat een goede aansluiting voor verschillende beschikbare API’s . FreeRTOS bevat standaard de nodige synchronisatie middelen zoals: mutex, pool, channel , mailbox, etc. het bevat ook de benodigheden voor een low power applictation. Behalve de ARM architecture wordt door de FreeRTOS ook nog andere architecturen ondersteunt waaronder de Texas Insturments MSP430 ,Atmel SAM3 en nog vele meer. De FreeRTOS is dus een volledig rtos waarmee vele toevoegingen mogelijk zijn.

Als tweede hebben we mbed, gecreëerd door ARM speciaal voor het ‘internet of things’ onder een apache license. Het mbed RTOS bevat de noodzakelijke synchronisatie middelen zoals: threads, mutex, semaphore, signals, pools, channel, mailbox, timeouts, system interrupts en een header waarmee de nodigheden voor het verbinden met andere systemen worden geinclude . Doormiddel van de header worden api (application programming interface) aan de rtos toe gevoegd die vele mogelijkheden bieden. zoals voor besturing van input en output onder anderen voor analoge en digitale poorten, pwm signaal, bus, en input voor interrups. Er zijn oke verschillende api beschikbaar in mbed voor dergelijke interfaces waar onder serial, SPI, I2C, CAN en verschillende interfaces gebaseerd op usb, waaronder voor mouse en keyboard ondersteuning. Omdat mbed speciaal is ontworpen voor het “internet of things” is zijn er api beschikbaar waarmee bluetooth, ethernet en WIFI verbinding kan worden gerealiseerd. Door de vele mogelijkheden die beschikbaar zijn bij het gebruik van mbed is het een RTOS die vele mogelijkheden bied op het gebied van verbinding met andere systemen.

Als derde hebben we Chibi-OS/rt en is ontworpen door Giovanni Di Sirio en is beschikbaar vanaf 2006 met wekelijkse fixes en updates. Deze rtos is een basis rtos en bevat de nodige functies met een mogelijkheid voor API’s. De kernel van chibiOS/ RT bevat een aantal services en modules voor onder andere de base kernel services, hiermee wordt bedoelt de functies voor scheduler, system management, threads en time and virtual timers. synchronisatie middelen zoals de condition variabls, channels, mailboxes, semaphores, event flags en mutexes bevinden zich ook in een van de modules. Memory management is verwerkt in een module me de mogelijkheden voor een Heap, memory pool, core management management en dynamic threads. De Chibi-Os/rt is ontworpen om met meerdere architecture te kunnen werken zoals ARM, Almes mega AVR, NXP, PIC32MX.

Als vierde RTOS kwam Atom threads aan de orde dit is een rtos gemaakt door atomthreads. Deze rtos heeft als hoofdtaak echt zo minimaal te zijn als mogelijk waarbij zoals bij FreeRTOS er als nog bepaalde functies aan toegevoegd zijn is er bij atom threads het tot het minimaal is gebracht en het is ook bedoelt voor systemen die alleen een scheduler en RTOS primitives. Het rtos bevat : preemptive scheduler with 255 priority levels, round-ribin at same priority level, semaphore, mutex, channel, timers. Het rtos is gemaakt in 11 .h en .C files waardoor het overzichtelijk blijft en makkelijk te begrijpen is. Doordat de atom threads zo minimaal is, is het mogelijk gemaakt om vele architecturen te gebruiken. En door een specific architecture module te gebruiken. wel is het mogelijk om API’s toe te voegen aan het rtos.

Als vijfde is RIOT gekozen. Deze rtos heeft een andere benadering dan de andere rtos-en die eerder genoemd zijn. RIOT is gebaseerd op het concept Internet of Things. Hierbij bevat deze rtos behalve de RTOS primitives ook de benodigde modules voor een verbinding in een netwerk met ondersteunende systemen.