

# Pemrosesan SRF05, CMPS03, TPA81, Sistem Motor Secara MultiProsesor pada Robot KRPAI

Hendi Wicaksono <sup>(1)</sup>

Industrial Robotic Design  
Teknik Elektro Universitas Surabaya  
Surabaya, Indonesia  
hendi@ubaya.ac.id

Ari Bengnarly Tanjung <sup>(2)</sup>

Industrial Robotic Design  
Teknik Elektro Universitas Surabaya  
Surabaya, Indonesia

**Abstrak**—Permasalahan pada respon robot KRPAI (Kontes Robot Pemadam Api) di tahun 2012 melatarbelakangi penelitian ini. Masalah keterlambatan respon robot tersebut diidentifikasi karena faktor pengaktifan dan pengambilan data *sensor* yang dilakukan saat diperlukan saja. Terutama pada sistem *sensor* jarak yang menggunakan SRF05, sistem *sensor* navigasi yang menggunakan *sensor* kompas digital CMPS03, dan *sensor* pendeteksi api TPA81. Hal ini dilakukan agar tidak membebani kerja mikrokontroler saat mengerjakan proses utama. Yang menjadi masalah adalah bagaimana semua *sensor* bisa selalu bekerja tanpa membebani mikrokontroler. Pada makalah ini membahas penerapan multiprosesor (banyak prosesor) sehingga semua *sensor* bisa selalu bekerja tanpa membebani mikrokontroler utama yang mengerjakan proses utama. Hasil yang diperoleh adalah penggunaan multiprosesor membantu meningkatkan respon jelajah robot yang dibuat tim *Industrial Robotic Design* Ubaya pada lapangan KRPAI 2013.

**Kata Kunci**—KRPAI, SRF05, CMPS03, TPA81, MultiProsesor

## I. PENDAHULUAN

Pada tahun 2012, Tim *Industrial Robotic Design* Ubaya membuat robot KRPAI (Kontes Robot Pemadam Api) yang dilengkapi dengan lebih dari 1 jenis *sensor*, bahkan pada *sensor* jarak digunakan 8 unit *sensor*. Pemrosesan *sensor* mulai dari pengaktifan *sensor*, pengambilan data dilakukan bila diperlukan saja. Semisal ketika ingin mendeteksi ada tidaknya dinding, maka saat itu baru dilakukan pengaktifan *sensor*, dan pengambilan data. Hal ini menyebabkan adanya waktu tunda pada respon kedua motor robot KRPAI 2012. Apabila dilakukan rutin pengaktifan semua *sensor* secara bergantian menyebabkan beban kerja mikrokontroler sangat tinggi. Belum lagi jika dilakukan hal tersebut, dapat mengakibatkan gerak robot yang terputah-putah karena pengendalian gerak kedua motor tidak dapat terus-menerus. Terkait navigasi, masalah respon robot ini teridentifikasi 100% menjadi satu-satunya masalah pada KRPAI 2012 dikarenakan robot KRPAI saat itu sudah menggunakan motor Vexta lengkap dengan *driver motor*, dan *sensor* jarak SRF05 yang teruji stabil penggunaannya.

Penggunaan SRF05 sebagai *sensor* jarak sudah lazim digunakan seperti pada *obstacle avoidance vehicle robot* yang dikembangkan oleh Dheeman Paul [1]. Pada robot tersebut mengembangkan penggunaan 1 *sensor* jarak untuk dapat menghindari halangan dengan teknik *Potential Field*

*Algorithm*. Dengan hanya 1 *sensor* saja, tidak memungkinkan robot dapat bergerak cepat sambil bernavigasi. Oleh karena itu, robot KRPAI menggunakan 8 buah *sensor* SRF05.

Muncul pemikiran untuk membuat semua *sensor* selalu bekerja mulai dari pengaktifan *sensor* dan mendapatkan data *sensor*. Begitu pula dengan aktuator berupa 2 unit motor yang selalu bekerja hingga terjadi perubahan kecepatan yang diinformasikan padanya. Namun, dengan catatan semua proses tersebut tidak boleh mengganggu atau membebani kerja mikrokontroler saat mengerjakan proses rutin. Manimaran di tahun 1998 juga menyatakan bahwa proses paralel memungkinkan dilakukan pada sistem *realtime* multiprosesor [2]. Pada sistem multiprosesor yang Manimaran kembangkan mengutamakan *scheduled time* atau waktu yang sudah direncanakan untuk pengambilan data dan pemberian data. Sistem tersebut tidak sesuai 100% dengan kebutuhan robot KRPAI 2013 ini, sehingga robot ini menggunakan sistem *request*. Sistem *request* dijalankan oleh *master* kontrol yang memerintahkan *slave* mengirimkan datanya.

Konsep desentralisasi seperti yang dikemukakan di atas didapat dari model sistem kontrol *mobile* robot Bebot [3]. Penggunaan konsep tersebut pada Bebot *mobile* robot adalah setiap robot memiliki 1 kontroler dari multi-robot tersebut. Sedangkan pada robot KRPAI ini konsep tersebut diimplementasikan dengan membuat setiap proses kerja memiliki 1 kontroler dalam 1 robot.

Pada makalah ini membahas 5 bagian penting, dimulai dari bagian pertama tentang teori terkait multiprosesor, bagian kedua metode penelitian yang digunakan, bagian ketiga desain robot KRPAI khususnya pada multiprosesor, bagian keempat hasil dan analisis, bagian terakhir simpulan dan penelitian mendatang.

## II. DASAR TEORI

Arduino merupakan *platform* berbasis *open source* pemrograman IC ATMEL AVR [4]. Dengan sistem *library* Arduino ini pemrograman AVR sangat mudah terutama pada proses pengaturan sistem *timer* dan masih banyak lainnya. Tampilan *board* Arduino dapat dilihat pada Gambar 1 dan tampilan *software* antar muka Arduino dapat dilihat pada Gambar 2.