2.4 A Distributed Autonomous and Collaborative Multi-Robot System Featuring a Low-Power Robot SoC in 22nm CMOS for Integrated Battery-Powered Minibots

Vinayak Honkote[1], Dileep Kurian[1], Sriram Muthukumar[1], Dibyendu Ghosh[1], Satish Yada[1], Kartik Jain[1], Bradley Jackson[2], Ilya Klotchkov[2], Mallikarjuna Rao Nimmagadda[1], Shreela Dattawadkar[1], Pranjali Deshmukh[1], Ankit Gupta[1], Jaykant Timbadiya[1], Ravi Pali[1], Karthik Narayanan[1], Saksham Soni[1], Saransh Chhabra[1], Praveen Dhama[1], N. Sreenivasulu[1], Jisna Kollikunnel[1], Sureshbabu Kadavakollu[1], Vijay Deepak Sivaraj[1], Paolo Aseron[2], Leonid Azarenkov[2], Nancy Robinson[2], Arun Radhakrishnan[3], Mikhail Moiseev[2], Ganeshram Nandakumar[1], Akhila Madhukumar[4], Roman Popov[2], Kamakhya P. Sahu[1], Ramesh Peguvandla[1], Alberto Del Rio Ruiz[3], Mukesh Bhartiya[1], Anuradha Srinivasan[2], Vivek De[2]

[1]Intel, Bangalore, India; [2]Intel, Hillsboro, OR; [3]Intel, Guadalajara, Mexico
[4]Intel, Toronto, Canada

Multi-robot systems, working collectively to accomplish complex missions beyond the capability of a single robot, are required for a wide range of applications such as search-and-rescue, precision agriculture and industrial automation. The collective behavior of a multi-robot system is governed by its continuous interactions with the physical environment, inter-robot information exchange, and intelligent decision making. Energy-efficient performance, size, weight and scalability of the key sensor, compute, communication, control and actuator components [1-3] of the individual robot platform are critical for realizing efficient advanced systems. In this paper, we demonstrate a *distributed, autonomous and collaborative* multi-robot system featuring integrated, battery-powered, crawling and jumping minibots for an example search-and-rescue application. We present a 16mm^2 low-power Robot SoC in 22nm CMOS that is integrated in the cm-scale minibot platform along with audiovisual and motion sensors, battery, low-power wireless communication and motion actuator components (Fig. 2.4.7).

In our example search-and-rescue application (Fig. 2.4.1), 4 minibots collaboratively navigate and map an unknown area without a central server or human intervention, detecting obstacles and finding paths around them, avoiding collisions, communicating among themselves, and delivering messages to a base station when a human is detected. Each minibot platform integrates: (i) camera, LIDAR and audio sensors for real-time perception and navigation; (ii) a low-power Robot SoC for sensor data fusion, localization and mapping, multi-robot collaborative intelligent decision making, object detection and recognition, collision avoidance, path planning and motion control; (iii) low-power ultra-wide-band (UWB) radio for anchorless dynamic ranging and inter-robot information exchange; (iv) long range radio (LoRa) for robot-to-base-station critical message delivery; (v) battery and PMIC for platform power delivery and management; (vi) 64MB pseudo-SRAM (PSRAM) and 1GB flash memory; and (vii) actuators for crawling and jumping motions.

The Robot SoC compute and control die (Fig. 2.4.2) integrates: (i) a real-time (RT) subsystem with an x86 host processor for sensor data grab and preprocessing; (ii) a Tensilica DSP processor for localization, mapping, collision avoidance and collaborative intelligent decision making; (iii) dedicated path planning (PP) and motion control (MC) hardware accelerators; (iv) an always-on (AON) subsystem with audio accelerator for human voice detection; (v) a memory subsystem with 8KB ROM, 2.86MB SRAM and 370KB register file (RF); (vi) an applications subsystem with a x86 processor and CNN accelerator for object detection and recognition; and (vii) die-to-die simplex link (D2DSL) interface to external chips and FPGAs in the platform.

Individual functions of a real-time search-and-rescue application are mapped onto the Robot SoC (Fig. 2.4.2). The complex application, running continuously for several hours with battery-powered minibots, demands execution of many performance-critical, compute-heavy and memory-intensive tasks with extreme energy efficiency. Hence, dedicated accelerator blocks are integrated on the SoC for the critical path planning and motion control functions.

The RT subsystem is designed to meet the strict real-time compute and deterministic interrupt latency requirements – particularly for sensor data acquisition/processing. RT supports simultaneous interactions with a large set of peripherals and sensors and enables the execution of time-critical dynamic-ranging capability essential for realizing a fully autonomous multi-robot system.

The audio accelerator in the AON subsystem receives 4-channel microphone input for detecting and identifying human voice. The tightly coupled memory (TCM) in the memory subsystem constitutes the next level of cache for the x86 application processor with 32KB L1 data and instruction caches. Cache misses result in data movements through the external PSRAM interface. Each bank of the ROM, SRAM and RF arrays in the memory subsystem can be power gated to minimize memory power at runtime.

Path planning by a robot to move efficiently from the current position to a target location, considering all the obstacles along the way, is extremely compute and memory intensive, and is typically off-loaded to a cloud server in state-of-the-art small form-factor robot platforms with limited compute and memory resources. In order to achieve smooth navigation and efficient mission execution in real time search-and-rescue in our battery-powered minibot platforms, we have integrated an energy- and area-efficient PP accelerator engine in the Robot SoC (Fig. 2.4.3). The PP accelerator implements (i) space discretization to determine the optimal direction of path exploration quickly, and (ii) a continuous space approach to accurately identify narrow paths for critical search-and-rescue navigation, to achieve effective path planning with limited compute and memory resources. The current robot position and the obstacle locations obtained from the LIDAR sensor are first mapped onto the entire area of interest divided into major grids. From the current position, neighboring grids are clipped by the clipper module and local targets assigned. Neighboring grids containing obstacles are sub-divided into finer grids. The transformed obstacle map with respect to the current robot position is then stored in a linked list to optimize memory usage without compromising accuracy. Thus, continuous obstacle-free spaces are utilized effectively and narrow feasible paths around obstacles are identified efficiently, while avoiding multiple readings of the complete obstacle map. Parameterized trajectory generators (PTG) find the edges connecting the current robot position to local targets. In conventional implementations, all of the reference trajectories are typically stored in memory. Here, the trajectories are computed by the PTGs *in real time on the fly,* thus avoiding look-up tables for storing pre-computed trajectories. Once the exploration is complete, the start position is backtracked from the target location, and the nodes and edges connecting the start and end points are stored in TREE memory.

The MC accelerator (Fig. 2.4.3) enables complete closed loop autonomous control for any minibot motion and helps meet the real-time operation requirements. It provides the capability to control with high-precision multiple-motion actuators of different types in parallel. The PWM module, configurable in one-shot or continuous mode, outputs four 10b resolution pulses to the motor drivers based on controller commands. The encoder module determines directionality and the filter module implements a moving average filter with configurable window size for required accuracy. A fully configurable PID controller is used where parameters can be changed on the fly and the sampling interval can be set to provide precise timing control resulting in accurate odometry.

Eight SoC power states and event-based transitions are implemented as a state machine in hardware. The SoC consumes 14.9mW, 30.3mW, 19.8mW and 36.9mW, in AL, AO, AS and AA power states, respectively (Fig. 2.4.4). Platform memory organizations across the SoC memory subsystem, PSRAM and flash, and memory utilizations for different compute and communication functions in the example application are shown in Fig. 2.4.4. Measurements demonstrate efficient and accurate operation of the PP & MC accelerators in the Robot SoC (Fig. 2.4.5). The PP accelerator operates at 75 to 160MHz (0.6 to 1V) and the MC accelerator operates at 0.5 to 4MHz (0.65 to 0.8V), consuming 6 to 37.5mW and 1 to 6mW, respectively. The Robot SoC operates over a wide voltage-frequency range of 80 to 365MHz (0.6-1V), consuming 32 to 238mW, and achieving minimum energy consumption of 0.29nJ at 0.65V, 140MHz. Platform measurements (Fig. 2.4.6) for the end-to-end search-and-rescue application for the fully distributed, autonomous and collaborative battery-powered, cm-scale minibots demonstrate accurate and reliable real-time energy-efficient performance, enabled by major improvements in energy consumption and memory usage achieved by the Robot SoC with integrated PP and MC accelerators, compared to an Edison-based reference platform.

References:

A. Amravati, et al., "A 55nm Time-Domain Mixed-Signal Neuromorphic Accelerator with Stochastic Synapses and Embedded Reinforcement Learning for Autonomous Micro-Robots," *ISSCC*, *pp*. 124-126, 2018.

X. Zhang, et al., "A Multi-Chip System Optimized for Insect-Scale Flapping-Wing Robots," *IEEE Symp. VLSI Circuits*, *pp*. C152-C153, 2015.

A. Suleiman, et al., "Navion: A Fully Integrated Energy-Efficient Visual-Inertial Odometry Accelerator for Autonomous Navigation of Nano Drones," *IEEE Symp. VLSI Circuits*, *pp*. 133-134, 2018.

 978-1-5386-8531-0/19/$31.00

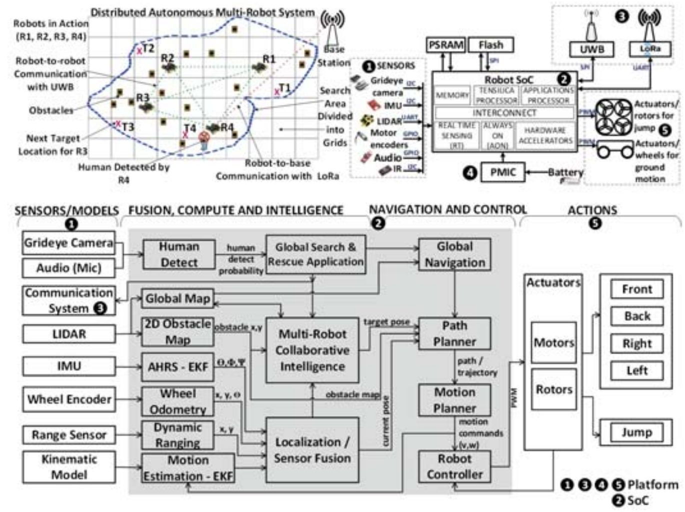

Figure 2.4.1: System architecture diagram and algorithmic flow for search and rescue application of the distributed, autonomous and collaborative multi-robot system.

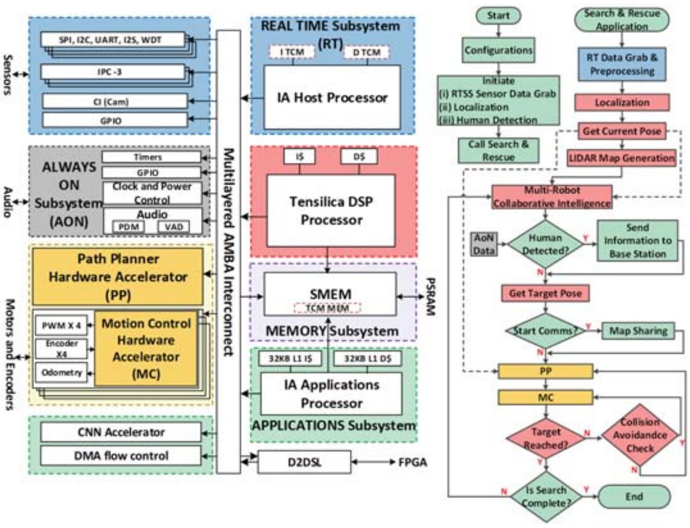

Figure 2.4.2: Robot SoC microarchitecture details and algorithm flow for search and rescue application.

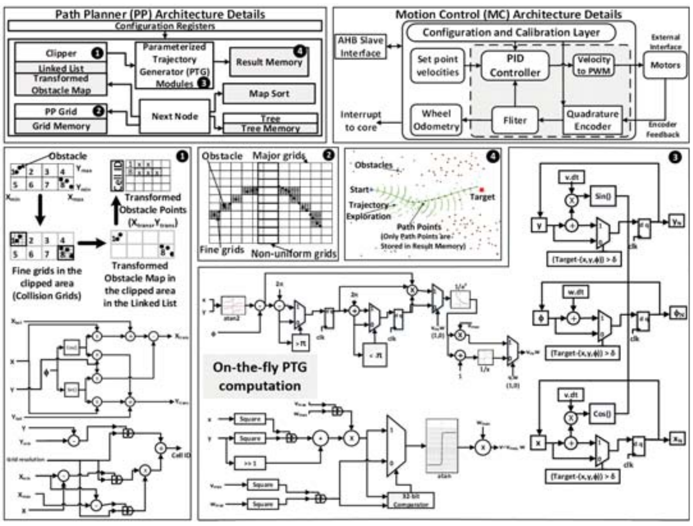

Figure 2.4.3: Architectural details and key features of the path planner (PP) and motion control (MC) hardware accelerators.

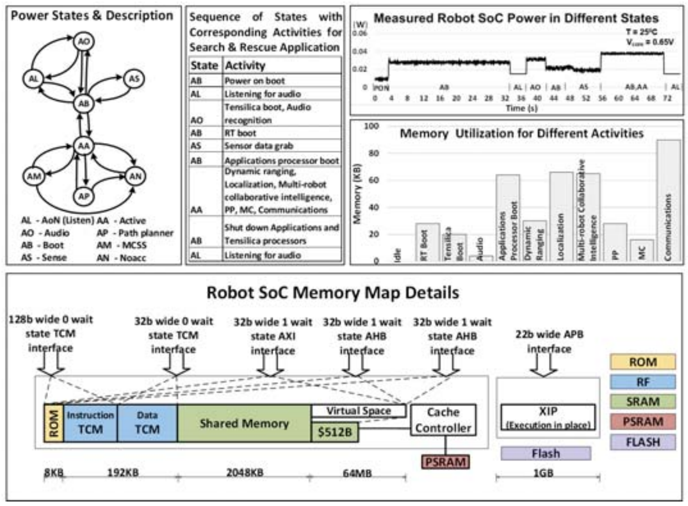

Figure 2.4.4: Robot SoC and system power management features; Robot SoC measured power and memory for a sample application; Robot SoC memory organization and mapping.

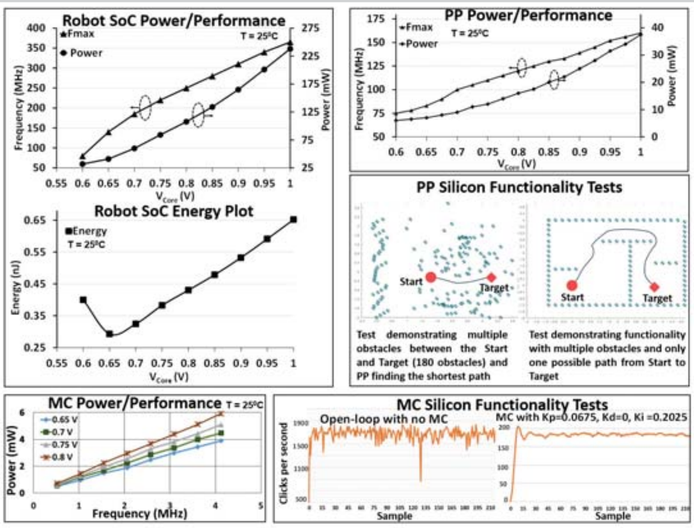

Figure 2.4.5: Measured voltage, frequency, energy data and functionality demonstration for key IP blocks in the Robot SoC.

	Memory (KB)		Execution Time (ms)		Energy (nJ)	
Functionality	Robot SoC	Edison	Robot SoC	Edison	Robot SoC	Edison
Localization	62	429	0.5	1.3	2.75E+04	2.02E+05
Collaborative Intelligence	61	66	11	13.6	8.91E+05	3.40E+06
Path Planner	28	20145	2	5.3	2.40E+04	9.09E+05
Motion Control	18	25	0.011	0.49	4.40E+01	7.02E+04
Total for All the Functionalities with Search & Rescue Application	560	24000	37	98.9	3.00E+06	3.25E+07

	This work	[1]	[2]	[3]
Technology	22nm	55nm	40nm	65nm
Area	$16mm^2$	$3.4mm^2$	$5.4mm^2$	$20mm^2$
Supply voltage	0.6-1V	0.4-1V	0.6-0.9V	Unavailable
Operational range	80-365MHz	0.0005-40MHz	30-220MHz	83.3MHz
Power	37mW	690uW	41.6mW	24mW
Circuit	Digital	Time domain mixed signal	Digital	Digital
Multi-robot collaboration	Yes	No	No	No

Figure 2.4.6: Measured system-level data for performance, execution time and comparison with Edison platform.

Technology	22nm FinFET, 9-metal layer process
Die Size	4mm x 4mm
Gate Count	4.83million
Total Memory	RF: 236KB; SRAM: 2.78MB; ROM:8KB
Operational Range	80-365MHz
Package	FCCSP-BOT

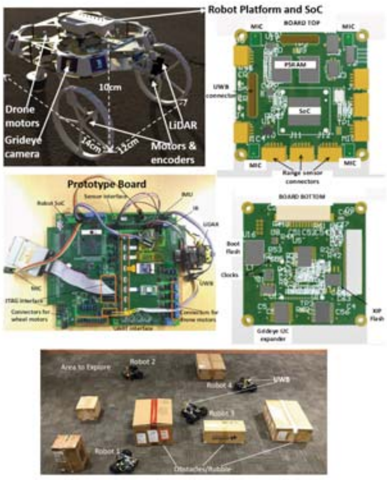

Figure 2.4.7: Robot SoC chip micrograph, platform, form factor and representative multi-robot collaboration pictures.