

Lily: A Miniature Floating Robotic Platform for Programmable Stochastic Self-Assembly

Bahar Haghghat, Emmanuel Droz and Alcherio Martinoli

Abstract—Fluid-mediated programmable stochastic self-assembly offers promising means to formation of target structures capable of a variety of functionalities. While miniaturized building blocks allow for finer resolutions in such structures, as well as access to unconventional environments, they can only be endowed with very limited on-board resources. In this paper we present the design, fabrication, and experimental results validating the key functionalities of the Lily robot as the building block in a programmable stochastic fluidic self-assembly system, capable of forming 2D structures. In particular, we aim at driving a system including an arbitrary number of Lilies to form target structures through parallel self-assembly, using exclusively local information and communication. While capable of wireless communication to a base station, Lilies are endowed with custom-designed electropermanent magnets to latch and also to communicate locally with their neighbors. Several experiments validate the reliability of the radio channel as well as the robustness of the local induction-based communication which allows for data transfer at 9600 bps with a success rate of 92.8% without repetition. The latches are shown to hold four times the weight of a single robot and to drag in another Lily from a distance of 4 mm in water.

I. INTRODUCTION

In recent years, technological breakthroughs have steered the robotics community towards high miniaturization, envisioning small robots roaming in environments inaccessible to conventional robotic platforms. Severe restrictions on computation, sensing, actuation, and communication capabilities along with inherently highly noisy interactions are however unavoidable at small physical scales. These difficulties can be mitigated by deploying stochastic coordination approaches, allowing distributed robotic systems to deliver more reliable performances and to accomplish tasks beyond the capabilities of a single robot [1].

Defined as the reversible phenomenon of an ordered spatial structure emerging from the aggregate behavior of simpler entities through local interactions and inherent randomness in the system, with no external direction [2], self-assembly is a powerful coordination mechanism which has recently been the focus of studies in distributed robotics [3], [4], [5], [6]. Playing a key role in many of the natural structuring phenomena at all scales, a key motivation for deploying self-assembly in engineered collective systems has been to realize the inherent scalability and robustness observed in the natural instances [7]. These systems, inspired by the

Bahar Haghghat, Emmanuel Droz and Alcherio Martinoli are with the Distributed Intelligent Systems and Algorithms Laboratory, School of Architecture, Civil and Environmental Engineering, École Polytechnique Fédérale de Lausanne, Switzerland. `firstname.lastname@epfl.ch`

This work has been sponsored by the Swiss National Science Foundation under the grant number 200021-137838/1.

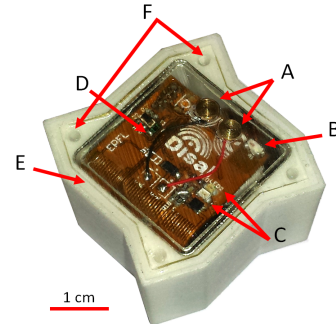


Fig. 1. Picture of a Lily robot. Some key features visible in the picture are: charging contacts (A), chip antenna (B), two LEDs signaling board status (C), ambient light sensor (D), sealing gap filled with silicone paste (E), and two of the four trimming holes (F).

nature, typically deploy multiple simple and small building blocks. Simplicity in the building blocks' internal design is in favor of reducing costs and increasing robustness as well as allowing for further reduction of the block's size, ultimately resulting in finer resolution in the formed target structures. While obtaining low-power reliable latching and communication is generally challenging at small physical scales, taking advantage of the stochastic forces in the environment for locomotion, allows for further simplification of the blocks' internal design, in particular in terms of power, computation, and actuation on-board resources.

Structure formation using stochastic assembly has been studied by several researchers [4], [8], [9], [10], [11], [12]. Stochastic fluidic assembly of passive modules on an active substrate in 3D is investigated in [4]. Floating Tribolons' stochastic assembly has been studied in [9] where the modules are actuated using vibrating motors and the environment does not provide any control to guide the process. The intelligent modules in [10], capable of local communication through infra-red transceivers and controlling their permanent magnet-based latches, stochastically self-assemble on an air table based on their internal behavior. For the Pebble robots in [5], the formation starts with an ordered lattice, the stochastic forces in the environment are then used to detach unwanted blocks. While Pebble robots are capable of local communication among themselves, they are only powered while being connected to the structure around a seed node.

Inspired by the nature where at small physical scales biological structures self-assemble in fluidic environments by taking advantage of ambient stochastic motion for transportation, in our previous work we have studied guided self-assembly of passive Lily modules equipped with permanent

magnets as their single state latching mechanism in a fluidic arena [13], [14]. In that set-up, an external base station monitored the passive modules' trajectories and in real-time carried out a graph-based identification and closed-loop control policy, steering the self-assembly process towards the targeted structure in a time-efficient way [14].

Pursuing this line of research, we aim to investigate scenarios in which the building blocks actively take part in the assembly process. Lily robots presented in this paper, are designed as the core of a flexible experimental platform whose primary goal is to serve as a physical test-bed for distributed stochastic control strategies and corresponding modeling methods for programmable stochastic fluidic self-assembly. Having inherited their external shell shape from the passive Lily modules, Lily robots deploy low-power controllable Electro-Permanent Magnetic (EPM) latches which they can disable to reject certain interactions according to their behavioral rule-set and the information about their neighbors acquired by communication through EPM channels. Similar to [3], Lilies can also communicate over a radio link to a base station to receive commands, new firmware, or to report specific information. Being powered by a small LiPo battery, Lilies can actively take part in the assembly process at all times, in contrast to modules in [15] or [16], allowing for a parallel assembly scheme where sub-structures of the target can be built separately, eventually joining to complete the overall target. To the best of our knowledge, Lily robots are the only stochastic fluidic self-assembly building blocks capable of both wireless communication to a base station and local communication to neighbors through their custom-designed latches while being active and powered at all times.

II. SELF-ASSEMBLY OF LILY ROBOTS

Fig. 2 depicts the self-assembly arena. Forming a target structure by a swarm of Lilies involves several aspects. Given a target structure, an appropriate node-level behavioral rule-set is deployed on all units through wireless bootloading. The robots' EPM latches are by default enabled, resulting in a default latching upon meeting another robot. Once latched the EPM-to-EPM inductive communication channel is physically established. The blocks then exchange their internal states and according to their rule-set behavior, they will either decide to unlatch or remain latched and update their internal states accordingly. Each Lily then updates the base station with its new internal state over the radio. This information will be used to log the experiment and also as the ground truth for validating models. In the case of an unfavorable interaction, the Lilies will both disable their involved EPM for a certain time during which the blocks will drift apart as a result of the agitation in the fluidic environment. In addition to event-based reporting of their internal state, Lilies periodically communicate to the base station to check for pending commands such as a query about the battery voltage level or the internal state, a command for pausing the experiment, or a command for turning the robot off. This scheme allows the robots to spend most

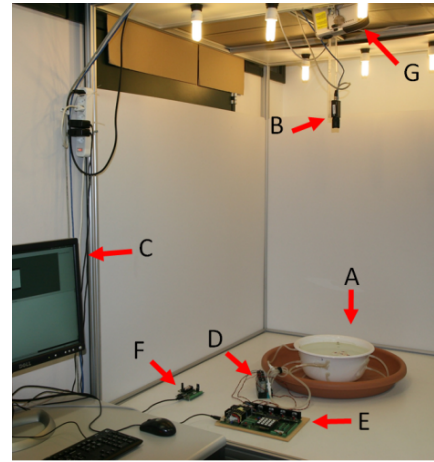


Fig. 2. Self-assembly arena, composed of water tank (A), overhead camera (B), base station desktop (C), diaphragm pumps agitating the fluidic environment (D), control and driver board for pumps (E), base node for establishing wireless link between the base station computer and the Lily robots (F), and a projector (G).

of their time in sleep mode or having the power hungry radio transceiver off, thus resulting in extended battery life. The commands from the base station can be as well used to modify the robots' behavioral rule-set on the fly. Our experimental set-up provides a flexible platform for studying a wide variety of control approaches for self-assembly, from an extreme case where all the intelligence and control is centralized and managed by the supervisor, to a mid-way scenario where the Lily robots actively guide the process following a certain behavior with the supervisor contributing to the process by providing clues or controlling the fluid motion, to the extreme case where the environment is not coordinated with the system and the assembly phenomenon is solely guided by the behavior of the Lily robots.

III. HARDWARE DESIGN

Fig. 1 shows a Lily robot, a 35 mm cubic-shaped power-autonomous unit. The choice of a latching mechanism was severely restricted by the small size of the robot, in particular a low-power mechanism was crucial. For the robots to be properly floating two key conditions were necessary: the payload needed to be below a threshold and the weight needed to be evenly distributed.

As depicted in Fig. 3, each robot is composed of several components: a LiPo battery, four electro-permanent magnets serving as physical latching and local communication channel, and a flexible circuit board. The flexible board is a two-layer design of total 170 μm thickness, on which a microcontroller unit with an integrated radio transceiver, an analog radio front-end, and a switching and power circuitry are placed. The battery is placed at the bottom of the shell. The flexible board with the four EPMS soldered on it is then folded and placed on top of the battery with the EPMS snapping in the sockets on the walls. A small frame is then placed in the middle to hold the EPMS in place. The gap between the transparent cap and the shell is sealed to protect

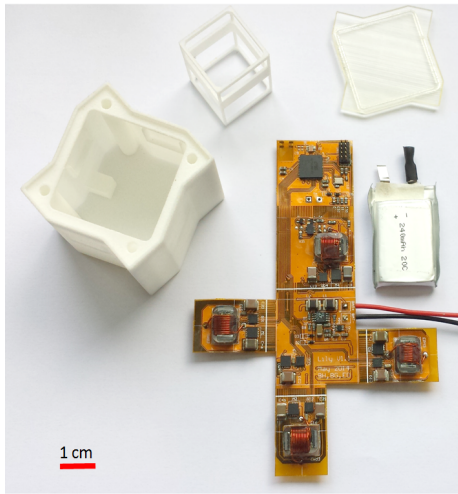


Fig. 3. Each Lily robot is composed of a flexible circuit board with four electropermanent magnetic latches soldered on it, a 240 mAh LiPo battery, a 3D printed shell, a 3D printed transparent cap, and a 3D printed frame for holding the EPMs in place.

the electronics inside from moisture.

A. Encapsulating Shell

The Lily’s shell is a 3D-printed water-resistant structure of 1 mm thickness. It encloses the module and defines the available volume, determining the payload limit to be 35 g. Table I summarizes the mass breakdown for a Lily robot. As depicted in Fig. 1, four small cavities are devised in the shell to allow for trimming and fine tuning weight distribution for a balanced flotation. The shell has a specific rugged shape designed to prevent the latched units from easily slipping away due to high energy agitations in the fluid. Two LEDs signaling the board status and an ambient light sensor are placed below the transparent cap.

A crucial aspect of the design was proper sealing of the electronics inside while maintaining the ability to quickly dis-assemble, repair, and re-assemble the device. The design includes a narrow gap of 1mm depth and width between the shell and the cap which is sealed using silicone paste.

B. Latching Mechanism

Due to high energy dynamics and stochastic nature of fluidic flows, in the realm of fluidic self-assembly it is crucial that the latching faces can get aligned automatically.

TABLE I
MASS BREAK-DOWN

Item	Mass
Shell	7.8 g
Cap	1.4 g
Battery	5.2 g
Populated Board	4 g
EPM	1.6 g
EPM Frame	0.5 g
Charging Contacts	0.2 g
Sealing Paste	0.5 g
Total	26 g

In order to meet a long-term energetic autonomy and the specific density requirements, a low-power and small-size latching mechanism was a necessity. We selected electropermanent magnets as they come with several advantages. These controllable magnets consume power only during the transient switching time. They are also efficiently down-scalable; while the required energy for switching is proportional to the latch volume, the force is proportional to its area [17]. The alignment of latching faces is also realized automatically through the interaction of the magnetic fields. An EPM consists of two different types of permanent magnet rods, both having almost the same remnant magnetization but very different coercivities. The rods are wrapped with a copper coil and have a small iron pole at each end for directing the magnetic flux. A high enough current pulse through the coil turns the latch “on” or “off” by setting the polarization of the soft magnet similar or opposite to that of the hard one respectively. This high current peak is obtained by discharging a capacitor on the EPM coil. When the latch is “on”, the magnetic flux reaches out and attracts magnetic materials; when it is “off” the iron poles provide a low resistance path for the flux to close within the two magnetic rods.

Fig. 4 depicts the structure of a custom-sized EPM for the Lily robots. EPMs are placed horizontally inside the Lily shell on four walls, i.e. with their North and South poles pointing left and right as opposed to pointing up and down resulting in a 4-way symmetric design. The force between two meeting Lilies is then always non-repulsive, thus in favor of getting latched. It is shown that in scenarios where the configuration form or pattern matters, gender-less latches are favorable [18].

1) *EPM Construction*: The design of a custom EPM consists of several aspects. The type and the size of the two magnet rods determine the resulting force field of the latch. On the other hand, a minimum magnetic field is necessary to switch the latch, the intensity of which is determined by the number of coil turns and the current passing through it. The number of coil turns directly influences the overall coil inductance which in turn affects the choice of the capacitor used for producing the current pulse. As a first step towards customizing the latch, our extensive experience with passive Lily modules was leveraged for sizing the EPM [14]. Using simulations in COMSOL, the EPM latch was customized so as to match the force field specifications of the empirically chosen permanent magnets deployed in the passive Lily modules. The magnetic rods were chosen to be grade N40 NdFeB and LNG40 Alnico, both of 2 mm diameter and 6 mm length. The pole pieces were laser cut from an Iron sheet to a size of 5 mm by 4.5 mm by 1.5 mm. Simulating the equations governing the electrical characteristics of the EPM in MATLAB and using a systematic search in a coarsely discretized parameter space, the electrical parameters were determined: 32 turns for the coil, a required current peak of 22 A, a capacitor of 400 μ F, and a voltage level of 12 V for charging the capacitor. The coils were made by wrapping 32 turns of thermal copper wire of #26 AWG around a

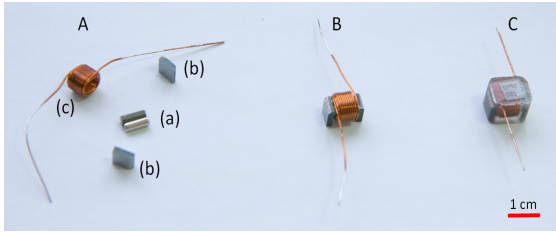


Fig. 4. A: An EPM is composed of two magnet rods (a), sandwiched between two iron pole pieces (b), and wrapped with 32 turns of grade 26 AWG wire (c). B: The pieces are held together using glue. C: The assembly is then put in a Polyurethan mold for protection against rusting.

mold. A current of 20 A was then passed through the coil for a few seconds to heat up the wire for the insulation to melt and the structure to be fixed. The magnet rods were glued to the pole pieces. We used a dedicated set-up for assembling the EPMs, allowing for a quick, precise and repeatable assembling procedure, resulting in EPM latches with almost identical characteristics.

2) *Switching Circuitry*: Fig. 5 depicts the EPM switching circuitry. An H-bridge structure was necessary for driving current in the EPM coils in two directions. For the sake of saving space on the board, each EPM has one dedicated half-bridge while the four share one common half-bridge, similar to the design in [16]. The MOSFET switches are capable of handling a maximum pulsed current of 40 A, and are all N-Channel type, thus faster to switch, less bulky and less lossy than P-Channel types of similar current capacity. To make the switching fast and also to protect the microcontroller, gate driver ICs are used to turn the MOSFET switches on and off. We leveraged the intrinsic diodes in the MOSFETs, capable of standing a maximum forward current of 25 A, for passing the induced current to the capacitor while the switches are turning off. As shown in Fig. 8 this current charges the capacitor back up by 0.7 V.

C. Power

Lily robots are designed to be energy autonomous deploying a 3.7 V 240 mAh Lithium-Polymer battery as shown in Fig. 3. Each module contains four surface-mounted 100 μF ceramic capacitors charged through a DC-DC converter chip that raises the voltage level from 3.7 V to 12 V. Discharging these capacitors on an EPM coil for approximately 50 μs produces a current peak of around 30 A, well above the required amount of 20 A. The battery is protected against running on low voltage or high current. In addition, the controller monitors the voltage level and is capable of issuing a turn-off command. There is also a charger chip on the board. It is thus sufficient to connect the charging contacts (see Fig. 1,A) to a power supply, and the charging scheme will be regulated automatically. By applying a short pulse on the same contacts, the robot can be turned on or off immediately.

Table II shows the supply current drawn on different modes. As can be seen, power consumption when only the core is running and the radio is off, is considerably less than when

the radio is on. It is thus efficient to keep the microcontroller in this mode most of the time, while the core needs to be running for EPM communications and switching, the radio can be turned on only when communication to the base station is necessary. When no EPM communication, switching, or radio communication task is pending the microcontroller enters the sleep mode. The microcontroller turns the radio on frequently and queries the base station for any commands that might have been issued and queued during the time the node was not listening. For a typical scenario with an estimated encountering rate of 2 per minute resulting in a communication over radio, the average current consumption will be 4.4 mA, resulting in 40 hours of power autonomy.

D. Integrated Transceiver-Microcontroller Unit

The microcontroller is an Atmega2564RFR2 Atmel processor which runs at 8 MHz and has 256 KB of self-programmable flash. The chip also features an integrated low-power transceiver for the 2.4 GHz band. The microcontroller on the Lily board serves three purposes. First, it controls all the low level electronics such as the PWM, switching and detection circuitry, the LEDs, and the ADC for the light sensor and the battery voltage readings. Second, it takes care of receiving and sending radio messages over the integrated transceiver. Last but not least, the microcontroller also hosts the software defining the behavior of the robot.

E. Communication and Sensing

Lily robots communicate with their neighbors through the EPM latches. Communication to the base station is realized through a wireless radio. The robots also have an ambient light sensor as a minimal means of sensing the environment.

1) *Node-to-Node Communication*: Lilies communicate to their neighbors using the EPM latches. When two EPMs are in close contact, they couple magnetically through shared magnetic flux. Similar to a 1:1 isolation transformer, a current pulse through one of the coils induces a similar pulse on the second coil, the size of which is proportional to the size of the pulse on the first coil and also to the mutual inductance of the coupled EPM coils. This inductive communication channel is utilized to transfer data between robots. In order to avoid affecting the physical bonding quality, the communication pulses are sent in the same direction as the ones used to turn the latch on. The inter-robot communication takes place at 9600 bps using a series of 1 μs wide pulses. The data bits are encoded using at least one pulse of 1 μs length. Two such pulses less than 100 μs

TABLE II
SUPPLY CURRENT ON DIFFERENT MODES

Item	Consumption
Sleep	700 μA
Core	4.1 mA
Radio Tx	18.6 mA
Radio Rx	16.6 mA
EPM Switching	30 A(for 50 μs)
EPM Tx	5 A(for 1 μs)

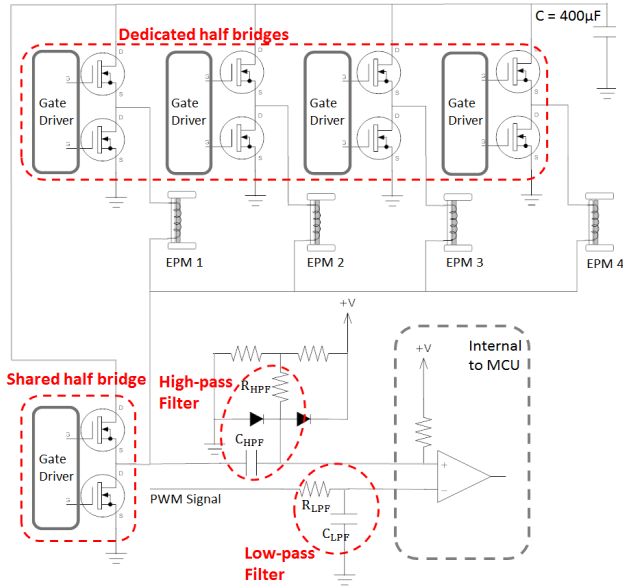


Fig. 5. The EPM switching and communication circuitry. In order to produce a current pulse in an EPM, the current path is closed through the dedicated half bridge and the shared one. Communication pulses are received on the positive input of the analog comparator, with the dedicated side grounded and the shared side floating.

apart represent a logic 1 while the lack of a second pulse within the $100 \mu\text{s}$ time frame after a first pulse is considered as a logic 0. Compared to a synchronized approach where the high and low bits are encoded by the presence of a pulse within a time window, this scheme sends half a pulse more on average. However, the advantage is, that there is no need for a synchronized communication clock between the two communicating units.

The detection circuitry for EPM communication is shown in Fig. 5. The analog comparator is internal to the microcontroller. Due to the four EPMs sharing a common half-bridge and a single analog comparator, it is impossible for the microcontroller to process messages on multiple EPM channels at the same time. The EPM channels are thus scanned sequentially. Similar to [16], to select a channel the dedicated side of the H-bridge is grounded by turning on the low-side MOSFET, while the other end of the coil is connected to the positive input of the analog comparator through a high pass filter with the DC level set to 1.5 V. The clipper diodes protect the input pins of the analog comparator against high voltage peaks. The negative input of the comparator is connected to a low-pass filtered square wave generated using the PWM timer channel of the microcontroller. This generates a DC level that can be adjusted for different detection sensitivities by modifying the duty cycle.

2) *Node-to-Base Communication:* Wireless communication between the base station and the Lily robots is necessary for several purposes. Since the Lily robots need to be water resistant, the electronics is enclosed and sealed inside the plastic shell. While wired programming required the sealing to be opened to access a programming header, using an Over

The Air Upgrade (OTAU) scheme only required a wireless communication link to the base station. Through wireless boot-loading, the Lilies can be programmed easily without the need to open the sealing. The wireless communication is also used for logging the robots' internal data during experiments. For instance, the robots can send information about their internal state, battery level, or the measured luminosity level. This information can be later used to compare models with the ground truth experimental results and also to detect any faulty robots during experiments. Lilies are also capable of receiving commands from the base station such as queries about specific information, or commands to change parameters in the modules behavior.

3) *Light Sensor:* An ambient light sensor is placed on the bottom side of the flexible board. Lilies can then change their behavior according to the controllable light patterns generated through a dedicated overhead projector as in [19] (see also Fig. 2).

IV. SCALABILITY FEATURES

Self-assembly experiments with Lilies are envisioned to include tens of units. With large numbers it can be tedious to turn on or off, charge, or program the robots individually. For these reasons, the Lily robots have been designed not to require to be handled individually for the aforementioned operations as well as starting or stopping an experiment. The following subsections explain in detail how these operations are managed in an scalable fashion for a swarm of Lilies.

A. Programming

The self-programming feature of Atmega2564RFR2 is used to program the Lily robots. This allows a small program in the boot-loading sector of the memory to overwrite the main program code. The Lilies have a wireless bootloader that receives the new application code in chunks from the base station through the wireless link, performs error checking, and then writes the code to an appropriate memory location. After having finished writing the new application code, the bootloader program waits for the "start" command from the base station to jump to the application section of the memory and start executing the main program. The start command is sent using the multicast mode, therefore the whole swarm starts the experiment simultaneously. Multiple Lilies can be programmed with the new application at the same time using the multicast mode, thus the overall programming time is independent of the swarm size.

B. Charging

In order to charge the battery, the input contacts of the Lily's charger (see Fig. 1,A) have to be connected to a 5 V power supply. These contacts are protected against wrong polarity and short circuit. The charger chip on the flexible board regulates the appropriate charging scheme for the LiPo battery. Deploying this charging chip eliminates the need for separate chargers. For charging a group of Lilies, the robots are lined up and a charging rail connected to a power supply is placed on top of them. Each charger chip will

draw the appropriate current to charge a battery and stops automatically when the battery is full.

C. Powering

Lilies have a small push-button controller chip on board that eliminates the need for a physical switch for turning the robots on or off. A power-off input from the microcontroller to this chip allows for powering down the system. In order to turn off a swarm of Lilies the base station sends a multicast “turn off” command. Upon receiving the command, the Lily’s microcontroller does some housekeeping operations and then activates the power-off input to the push-button controller which disconnects the battery from the rest of the circuit. In this state only the push-button controller is on, drawing a 6 μ A current from the battery. Lilies can be safely stored in this state for more than two years on a single battery charge. In order to turn a Lily on, a short pulse of 5 V is applied on its charging input contacts. A swarm of Lilies can be turned on by lining them up and placing the charger rail on top of them, the short pulse can then be applied to the charging rail. Upon turning on, Lilies will be running the bootloader code, waiting for a command to start the experiment.

V. FIRMWARE DESIGN

Several low-level routines have been implemented abstracting the switching, communication, and sensing functionalities. The current firmware occupies 20 KB of the microcontroller’s memory leaving more than 230 KB for the behavioral code.

A. Radio Communication

For the wireless communication with the base station a command interface including 21 messages was designed. Table III lists the most important ones. When running the bootloader code the robot is constantly listening on the radio channel for incoming messages. In this state, being the default state after the robot is turned on, the Lilies can reply to the base station’s ping request, sending back their short address, and the version of the bootloader they are running. The wireless bootloader allows for programming the robot with a new application image. To allow for an update over radio, the program counter is redirected from the application code to the bootloader code upon receiving the corresponding command from the base. When running the application, the Lily’s receiver is turned on for 100 ms every 1 second, after having sent a “command request” message to the base. On the base station side, the out-going commands are always being queued. Upon receiving the command request from a Lily, the waiting commands are transmitted and terminated by a “radio off” message, allowing the application code to know when the queue is empty and it can turn the radio off.

B. EPM Communication and Switching

As explained in section III-E.1, EPM channels are scanned sequentially for incoming messages. Independent of this scheme, the Lily robots frequently send hello messages to actively discover their neighborhood. The exchange of hello

message serves two purposes. First, it allows the robot to discover changes in its neighborhood, for instance in order to find out if a new robot has been latched on a previously available face or if a previously latched neighbor has been detached due to high agitation in the environment. Second, the receiving Lily will use the handshaking scheme initiated with the hello message to detect the beginning of the neighboring transmitter data.

Hello messages are sent every 500 ms by default; a complete tour on all four EPM channels thus takes 2 s. The frequency of sending hello messages needs to correspond to the dynamics of the self-assembly process. In a highly stochastic environment where the rate of collision events is high, the Lilies need to check for neighborhood changes more frequently. While not latched to other robots, the transmitter Lily receives no response to the sent hello message. Once the robots are latched, the query pulses from the transmitter create an interrupt, waking up the receiver. Consisting of four bytes, the hello message is 4.2 ms long in total. If the hello message is correctly detected, the receiving Lily replies with an ACK which consists of a single high bit, before the data communication starts. The Lilies will exchange their internal states and decide whether they should stay latched or to unlatch. Since all robots are endowed with the same rule-set, their individual decisions will be the same. In order to unlatch, two pulses are sent to an EPM. The first pulse is in the direction to demagnetize the EPM latch, while the second one, sent a few seconds after, magnetizes the EPM back to its default state.

VI. EXPERIMENTS

In order to validate the functionalities of the Lily robot on both the hardware and the firmware sides, several experiments were conducted. The following section characterizes the key aspects of the design, namely the EPMs latching strength and their switching waveforms, the node-to-base radio communication, and the node-to-node EPM-based communication.

A. Radio Communication Channel

A test of the radio range was done in our laboratory environment with a Lily robot floating on water. While the

TABLE III
SUMMARY OF RADIO MESSAGE INTERFACE

Message	Sent by	Lily Robot’s State
Ping Request	Base	On Bootloader/Application
Set Target Memory	Base	On Bootloader
Run Application	Base	On Bootloader
Set Radio Channel	Base	On Bootloader
Run Bootloader	Base	On Application
Power off	Base	On Application
PWM Setting	Base	On Application
Query Battery	Base	On Application
Query Luminosity	Base	On Application
Soft Reset	Base	On Application
Command Request	Lily	–
Report State	Lily	–
Report Error	Lily	–

base node (see Fig. 2,F) was receiving at the maximum sensitivity level of -100 dBm the transmitter of the Lily robot was sending on full power at +3.5 dBm. Messages from the Lily robot were reliably detectable within a range of almost 10 m away from the base node. This test validated the antenna design as well as the communication quality that could have been affected by the small separation between the antenna and the rest of the folded flexible board.

In order to test the quality of the radio channel, we performed an experiment with two Lily robots counting on an 8 bit register and sending the counter value along with their IDs to the base node over the radio link while at a distance of 1m from the base station. One of the robots was floating in water while the other was resting on a desk close by. Each module transmitted 10,000 messages. The received data was logged on the base station. All the messages transmitted by both robots were correctly received by the base node. This test validated the reliability of the radio link, which is used to gather data in the EPM communication tests.

B. EPM Communication Channel

In order to test the reliability of the inter-robot communication over the inductive EPM channel we conducted several experiments in three different scenarios. The first scenario studied the reliability of the connection establishment protocol based on hello message exchange. In the second scenario, the same protocol for connection establishment was tested when the receiver was scanning all channels. In the third scenario, after having established a connection, the robots exchanged data.

In the first scenario, one robot was always listening on a fixed EPM channel. The other robot was always transmitting on a fixed channel. The transmitter only sent hello messages and after receiving the ACK proceeded to send a next one. A total of 10,000 messages were sent by the transmitter. The receiver did not expect any data and after detecting a hello message correctly, it sent an acknowledgment message back to the transmitter as well as a message over the radio link, reporting that the connection had been successfully established. Out of 10,000 transmitted hello messages all were detected correctly by the receiver.

In the second scenario, the receiver did not expect messages on a certain channel and thus scanned the channels periodically, listening on each channel for 1 ms. The transmitter was always on a fixed channel, sending a total of 10,000 hello messages. Since hello messages are specifically adjusted to be detectable without degradation in success rate compared to a fixed channel scenario, a high detection rate was expected. Out of 10,000 transmitted hello messages all were detected correctly, confirming our expectations.

In the third scenario, the transmitter sent a hello message to establish the connection after which it would proceed to send a data byte. The data byte was not represented by a fixed binary pattern, it was generated by counting on an 8-bit register. The receiver was listening on a fixed channel and would relay the received byte along with its parity to the base station over the radio, where the data was logged. Out

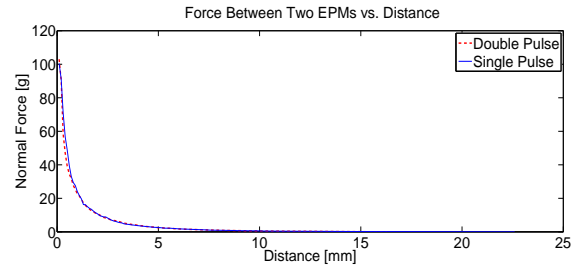


Fig. 6. Latching force between two EPMs.

of 10,000 such data transmitting attempts all 10,000 were received by the receiver out of which 716 were faulty due to a parity error. Table IV summarizes the test results for the three scenarios.

C. EPM Latching

Two types of tests were conducted to characterize the EPM latching strength: a pull test to determine the holding force and another test in which one EPM approached the other to sketch the force field. One EPM was always anchored to a 250 g block placed on a weighing scale of 0.1 g precision. For sketching the force field, the second EPM was mounted on a linear motion axis with a precision of 0.1 mm and approached the first one from a distance of 4 cm. Fig. 6 depicts the normal force for two different conditions, involving one or two latching pulses, averaged over 10 tests. As the two curves are only slightly different, it can be concluded that applying several pulses instead of one would not increase the latching strength considerably. For measuring the holding force, a pull test was performed in which the second magnet was fixed to one end of a spring of constant $k = 6.7$ N/m while the other end was pulled by the moving axis. The average holding force for the case where the EPMs were turned on using one switching pulse was 116 g. When the EPMs were turned on using two pulses the average holding force was 128 g. For the case where one EPM was turned on using one switching pulse and the other was off the average holding force was 32 g. The force between two magnets in the off state was measured to be 0 within the aforementioned precision. We also measure the EPM latches force in a still fluidic environment when turned on using a single pulse the attraction force between magnets was enough to bring two Lily robots together from a distance of 4 mm.

Fig. 7 depicts the current through and the voltage across an EPM coil during a single switching pulse. As it can be seen, the threshold current for switching the soft magnet is reached within 50 μ s after which the current path is opened

TABLE IV
EPM COMMUNICATION

Scenario	Message Type	Missed/Errors	Success Rate
1	Hello	0	100%
2	Hello	0	100%
3	Hello, 1 byte	716	92.8%

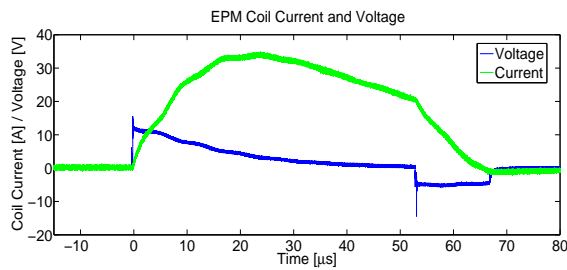


Fig. 7. Current through and voltage across an EPM coil during switching.

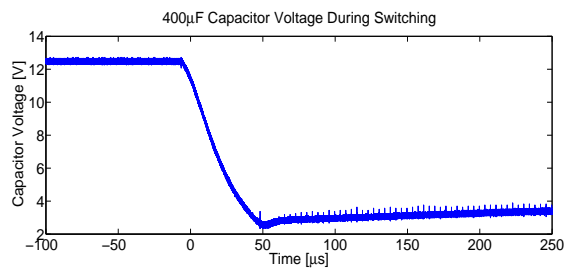


Fig. 8. Voltage across 400 μF capacitor during switching.

on the H-bridge. The voltage across the capacitor bank of 400 μF is depicted in Fig. 8. It can be seen that the back EMF of the switching coil charges the capacitor back up by 0.7 V in less than 10 μs , the capacitor is then charged up to 12 V through the DC-DC converter in 17 ms.

VII. CONCLUSION

In this work we have presented the Lily robot, designed to function as the building block in a programmable stochastic self-assembly system composed of a fluidic arena agitated by controllable fluid flow, a base station capable of tracking and communicating to assembly units as well as controlling the agitation in the fluidic arena, and multiple Lily robots as the self-assembly units. As an essential feature, by design, the effort required to program, power on or off, and charge a batch of Lilies does not increase with the number in the batch, thus facilitating self-assembly experiments involving a large swarm. Our experiments validate the viability of the hardware and firmware design, as well as the scalable characteristics of the system. The EPM latches switch effectively, strongly bonding the units, as well as providing a robust local communication channel. The wireless communication between the base station and the robots used for programming, command broadcasting, and data logging also proved to be reliable. Based on these encouraging results, our future work will be focused on producing a batch of up to 100 Lilies, and implementing high-level rule-based behavior algorithms. The platform will then be used as an experimental means to study modeling and control algorithms for distributed stochastic systems within an actively controlled environment.

ACKNOWLEDGMENT

This work is made possible through the help from several people. In particular two brilliant master students, Beat Geissmann, who worked on a rigid prototype for the Lily

board, and Alexandre Cherpillod, who worked on simulating EPM latches during their semester projects. The semi-automated production of EPMs was only achieved thanks to the unique skillset of André Guignard.

REFERENCES

- [1] M. Mastrangeli, G. Mermoud, and A. Martinoli, "Modeling self-assembly across scales: The unifying perspective of smart minimal particles," *Micromachines*, vol. 2, no. 2, pp. 82–115, 2011.
- [2] M. Bušev, *Synergetics: Chaos, order, self organization*. World scientific, 1994.
- [3] M. Rubenstein, A. Cornejo, and R. Nagpal, "Programmable self-assembly in a thousand-robot swarm," *Science*, vol. 345, no. 6198, pp. 795–799, 2014.
- [4] M. Tolley and H. Lipson, "Programmable 3d stochastic fluidic assembly of cm-scale modules," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011, pp. 4366–4371.
- [5] K. Gilpin, K. Koyanagi, and D. Rus, "Making self-disassembling objects with multiple components in the robot pebbles system," in *IEEE International Conference on Robotics and Automation*, 2011, pp. 3614–3621.
- [6] J. Neubert, A. Cantwell, S. Constantin, M. Kalontarov, D. Erickson, and H. Lipson, "A robotic module for stochastic fluidic assembly of 3d self-reconfiguring structures," in *IEEE International Conference on Robotics and Automation*, 2010, pp. 2479–2484.
- [7] S. Kernbach, *Handbook of collective robotics: fundamentals and challenges*. Pan Stanford Pub, 2013.
- [8] M. Tolley and H. Lipson, "Fluidic manipulation for scalable stochastic 3d assembly of modular robots," in *IEEE International Conference on Robotics and Automation*, 2010, pp. 2473–2478.
- [9] S. Miyashita, M. Kessler, and M. Lungarella, "How morphology affects self-assembly in a stochastic modular robot," in *IEEE International Conference on Robotics and Automation*, 2008, pp. 3533–3538.
- [10] E. Klavins, "Programmable self-assembly," *IEEE Control Systems*, vol. 27, no. 4, pp. 43–56, 2007.
- [11] P. White, V. Zykov, J. Bongard, and H. Lipson, "Three dimensional stochastic reconfiguration of modular robots," in *Robotics: Science and Systems*. Cambridge, 2005, pp. 161–168.
- [12] S. Griffith, D. Goldwater, and J. Jacobson, "Self-replication from random parts," *Nature*, vol. 437, no. 7059, p. 636, 2005.
- [13] E. Di Mario, G. Mermoud, M. Mastrangeli, and A. Martinoli, "A trajectory-based calibration method for stochastic motion models," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011, pp. 4341–4347.
- [14] G. Mermoud, M. Mastrangeli, U. Upadhyay, and A. Martinoli, "Real-time automated modeling and control of self-assembling systems," in *IEEE International Conference on Robotics and Automation*, 2012, pp. 4266–4273.
- [15] V. Zykov and H. Lipson, "Experiment design for stochastic three-dimensional reconfiguration of modular robots," in *IEEE Int. Conf. Intell. Robots Syst., Self-Reconfigurable Robot. Workshop, San Diego, CA*, 2007.
- [16] K. Gilpin, A. Knaian, and D. Rus, "Robot pebbles: One centimeter modules for programmable matter through self-disassembly," in *IEEE International Conference on Robotics and Automation*, 2010, pp. 2485–2492.
- [17] A. Knaian, "Electropermanent magnetic connectors and actuators: devices and their application in programmable matter," Ph.D. dissertation, Massachusetts Institute of Technology, 2010.
- [18] K. Stoy and D. Brandt, "Efficient enumeration of modular robot configurations and shapes," in *IEEE International Conference on Intelligent Robots and Systems*, 2013, pp. 4296–4301.
- [19] G. Mermoud, L. Matthey, W. Evans, and A. Martinoli, "Aggregation-mediated collective perception and action in a group of miniature robots," in *Proc. of the 9th International Conference on Autonomous Agents and Multiagent Systems*, vol. 2, 2010, pp. 599–606.