Physical modeling of daily electric appliances for education by Modelica

Yutaka Hirano[1] Koichi Ohtomi[2]

[1]Woven Planet Holdings, Inc., Japan, yutaka.hirano@woven-planet.global
[2] Ohtomi Design Lab., Japan, ohtomi@1dcae.jp

Abstract

Modelica is very useful to make physical models in various engineering fields such as mechanical, electrical, thermal, fluid systems, etc. This capability of Modelica is also useful to educate students and engineers about many physical areas using simulation. The authors are posting serialized articles in a technical magazine about physical modeling of daily electric appliances by Modelica to educate readers about both physics and Modelica language in Japan. This paper introduces some examples of physical modeling of various appliances such as electric minicar, dryer and speaker by Modelica.

Keywords: *physical modeling, control, Modelica*

1 Introduction

Modelica is an equation based, object-oriented language for efficient modeling of complex, multi domain cyber physical systems described by ordinary differential, difference and algebraic equations. This feature of Modelica language is very useful for not only various industrial applications but also for education about physics and simulation for students and engineers. The authors are posting serialized articles in a technical magazine about physical modeling of daily electric appliances by Modelica to educate readers about both physics and Modelica language in Japan. This paper introduces some examples of physical modeling of various appliances.

In section 2, modeling and simulation of a 4x4 electric minicar is described. In section 3, dryer is modeled and simulated. In section 4, modeling about a speaker is introduced.

2 Modeling of 4x4 electric minicar

4x4 electric minicars are a very popular toy in Japan. It consists of body, batteries, motor, gears, drive shafts and tires as shown in Figure 1. The physical model of the electric minicar is assumed as shown in Figure 2.

The modeling was done to solve the following questions.

1) What is the maximum speed of the car?
2) How long can this car keep running?

Thus, the battery model should consider the effect of voltage drop by the SOC (State Of Charge) change. In the body model running resistances should be considered.

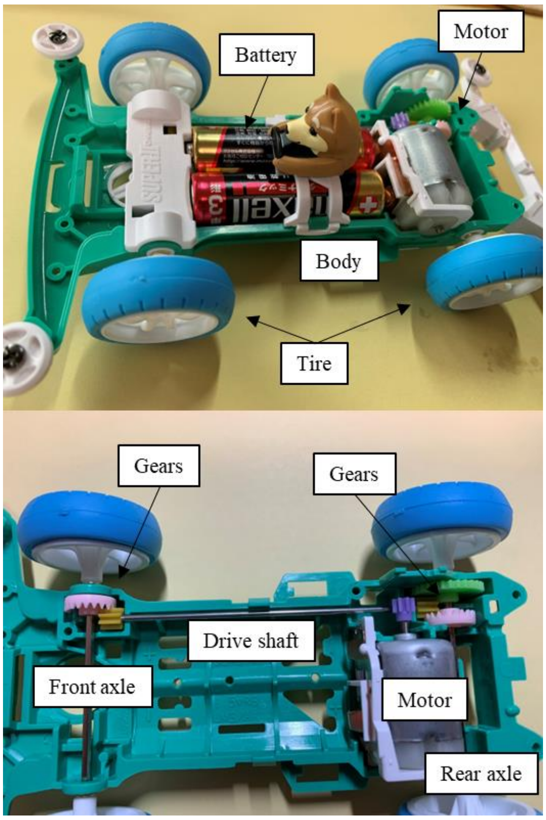

Figure 1. Structure of electric minicar

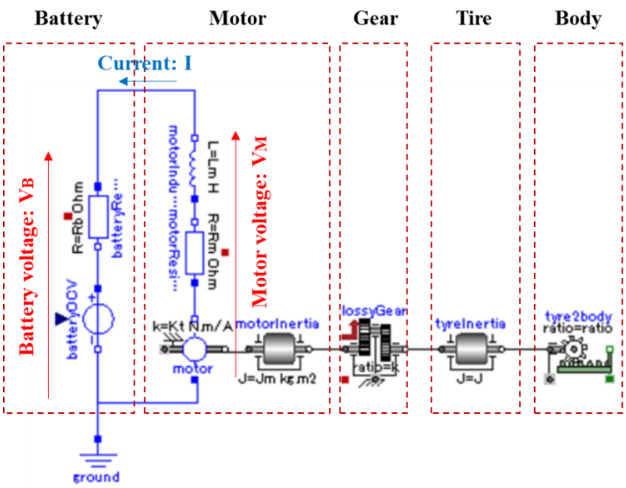

Figure 2. Physical model structure of electric minicar

2.1 Structure of the physical model

Battery model is built by OCV (Open Circuit Voltage) generator and inner resistance. OCV is a function of SOC. Motor model consists of a DC motor electric machine, a motor inertia and electric circuit including armature coil's resistance and inductance. Gear and tire model are assumed as just a simple model of lossy constant ratio reducer. As for the running resistances, aerodynamic resistance, rotating resistance and gravity resistance by road slope are considered. Consequently, the system of equations of this model become as follows.

(Battery)

Battery voltage: $V_B = V_{OCV}(SOC) - R_B I$ (1)

State Of Charge: $\mathrm{SOC} = 1 - \frac{V_{bo}\int I dt}{W_{total}}$ (2)

Open circuit voltage: $V_{OCV}(SOC) = V_{b0} \times SOC$ (3)

For the simplicity, V_{OCV} is assumed to be proportional to the *SOC*. It will be more precise if the actual table of SOC vs OCV based on measurement will be used.

(Motor)

Motor voltage: $V_M = V_{emf} + R_M I + L\frac{dI}{dt}$ (4)

Back electromotive force voltage:

$V_{emf} = K_E \omega_M$ (K_E : coefficient) (5)

Motor torque: $\tau_M = K_T I$ (K_T : coefficient) (6)

Motor inertia: $J_M \frac{d\omega_M}{dt} = \tau_M$ (7)

(J_M : Innertia moment, ω_M : Angular velocity)

(Circuit)

$V_B = V_M$ (8)

(Gear)

Gear torque: $\tau_t = \varepsilon K_g \tau_M$ (9)

(K_g : Gear ratio, ε : Gear efficency)

Gear rotation speed: $\omega_M = K_g \omega_t$ (10)

(Tire and Body)

Tire inertia torque: $J_t \frac{d\omega_t}{dt} = \tau_t - \mathrm{r} f_t$ (11)

(J_t : Tire innertia, ω_t: Tire rotation speed)

Car velocity: $V = \mathrm{r}\omega_r$ (r : Tire radius) (12)

Tire rotating resistance force:

$f_t = m\alpha + \mu mg + mg\sin\theta + \rho A C_D V^2/2$ (13)

(μ : Rotating resistance coefficient, θ : Road slope, ρ : Air dencity, A : Frontal area, C_D: Aero resistance coefficient)

Vehicle speed: $V = \frac{dx}{dt}$ (14)

Vehicle acceleration: $\alpha = \frac{dV}{dt}$ (15)

2.2 Modelica code of text-based model

By using the system of equations shown in the section 2.1, Modelica code of the text-based model becomes as below.

```
model miniCarText_SOC
  import SI = Modelica.Units.SI;
  import Modelica.Constants.g_n;
 parameter SI.Resistance Rb = 0.8
 "Battery inner resistance";
 parameter SI.Resistance Rm = 1
  "Motor inner resistance";
 parameter SI.Inductance Lm = 1e-6
  "Motor inner inductance";
 parameter SI.MomentOfInertia Jm = 1.8e-3
* 0.005 * 0.005 "Motor innertia";
 parameter SI.MomentOfInertia Jt = 5e-3 *
0.01 * 0.01 "Tire innertia";
 parameter Real Kt = 1.2e-3
  "Motor torque coefficient";
 parameter Real Ke = 1.2e-3
  "Motor rotational voltage coefficient";
 parameter Real Kg = 5 "Gear ratio";
 parameter SI.Efficiency Eg = 1
  "Gear efficiency";
 parameter SI.Mass m = 0.1
  "Vehicle mass";
 parameter SI.Radius r = 0.015
  "Tyre radius";
 parameter SI.CoefficientOfFriction myu =
0.1 "Tyre rotating friction coefficient";
 parameter Real Cd = 0.3
  "Air drag coefficiency";
 parameter SI.Area area = 0.004
  "Vehicle frontal area";
 parameter SI.Density rho = 1.205
  "Air density";
 parameter Real batteryPowerCapacity = 1
  "Battery power capacity [Wh]" ;
 parameter SI.Voltage Vb0 = 3
  "Battery initial voltage" ;
// Variables;
 SI.Current i(start = 0) "Motor current";
 SI.Voltage Vocv
  "Battery open circuit voltage";
 SI.Voltage Vb "Battery voltage";
 SI.Voltage Vm "Motor voltage";
 SI.Voltage Vemf
  "Motor rotational voltage";
 SI.Torque taum "Motor torque";
 SI.Torque taut "Tire torque";
 SI.AngularVelocity omgm
  "Motor angular velocity";
 SI.AngularVelocity omgt
  "Tire angular velocity";
 SI.Force f "Vehicle total force";
 SI.Force fa "Vehicle acceleration force";
 SI.Force fr
  "Vehicle rolling resistance force";
 SI.Force fair
  "Vehicle air resistance force";
 SI.Distance x(start = 0)
  "Vehicle running distance";
 SI.Velocity v(start = 0)
  "Vehicle velocity";
 SI.Acceleration a "Vehicle acceleration";
 Real soc "Battery SOC" ;
 Real usedWh "Used power capacity" ;
```


DOI
10.3384/ecp193121

```
equation
  der(usedWh) = Vb0 * i  / 3600;
  soc = if (1-usedWh/batteryPowerCapacity
>=0) then (1-usedWh/batteryPowerCapacity)
else 0;
  Vocv = Vb0 * soc;
  Vb = Vocv - Rb * i;
  Vm = Vemf + Rm * i + Lm * der(i);
  Vemf = Ke * omgm;
  Vb = Vm;
  taum = Kt * i;
//Jm*der(omgm) = taum;
  taut = Eg * Kg * taum;
  Kg * omgt = omgm;
//Jt*der(omgt) = taut - r*f;
  0 = taut - r * f;
  r * omgt = v;
  f = fa + fr + fair;
  fa = m * a;
  fr = myu * m * g_n;
  fair = rho * area * Cd * v * v / 2;
  a = der(v);
  v = der(x);
end miniCarText_SOC;
```

Please note that the equations (7) and (11) were ignored and the equation

```
0 = taut - r * f;
```

is used instead. This is because the variables `omgm` and `omgt` are dependent on the variable `v` by the constraint of constant gear ratio. Additionally, to cope with the calculation chattering when the vehicle speed crosses zero, it is effective to use the base class of friction elements prepared in the Modelica Standard Library (MSL), but in this case it is omitted.

2.3 MSL-based model

Modelica model of this minicar can be built by using MSL as shown in Figure 3. Here a new class ‘calcOCV’ was created to model the SOC dependent OCV calculation as shown in the equations (2) and (3). Thanks to the Modelica feature of ‘StateSelect’ the rigidly coupled inertias of the motor, tire and the vehicle mass can be modeled separately.

```
StateSelect stateSelect =
StateSelect.default;
```

2.4 Simulation results

To make the students understand both physics and Modelica features, both of the text code model and the MSL model were made and simulated. Figure 4 shows the results for short time range (upper figure) and for long time range (lower figure). In the upper figure, the results of vehicle speed for both the text code model (v) and the MSL model (vehicle.v) are compared. The effect of inertial elements can be seen. Also, as for the answers for the questions above, the results became as follows.

(1) The maximum speed of the car is about 6.1 m/s.
(2) The car can keep running for about 3840 sec.

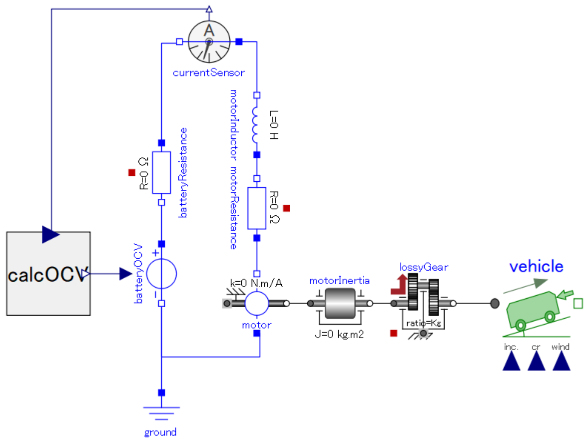

Figure 3. Minicar model using MSL

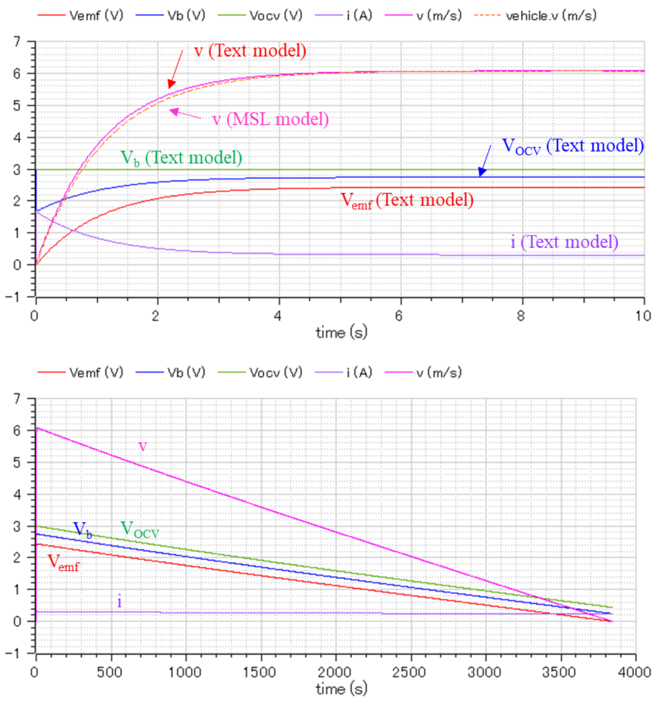

Figure 4. Simulation results of the minicar model

3 Modeling of a dryer

3.1 Structure of the dryer model

The structure of the target dryer is shown in Figure 5.

For each part of the structure, the system of the equations and the model structure were considered as below.

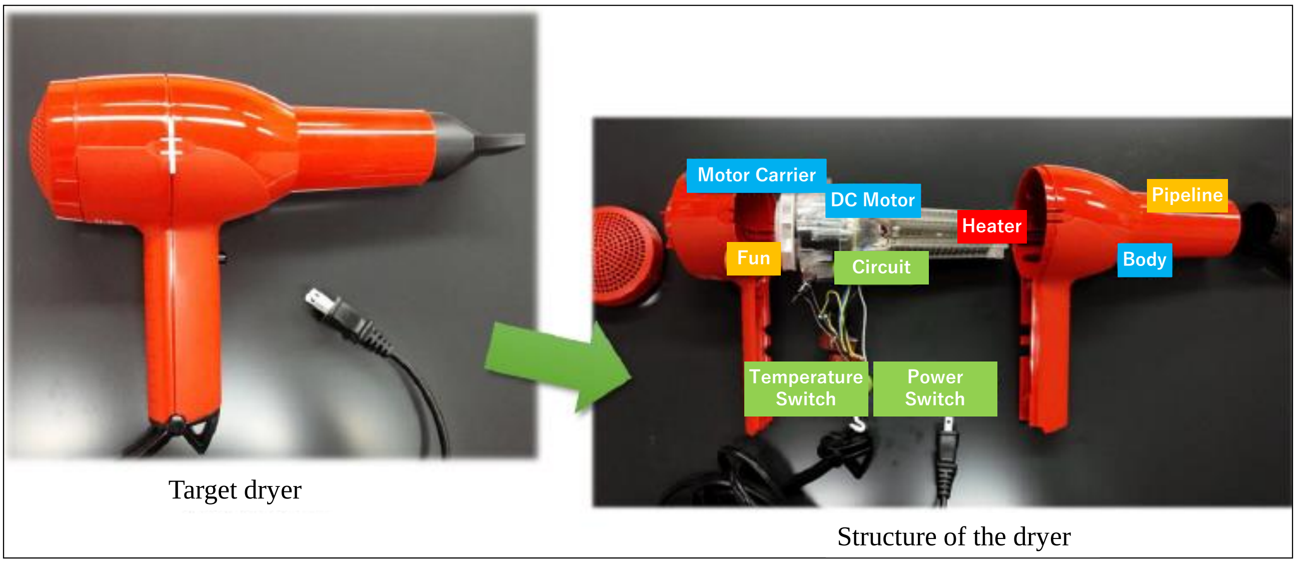

Figure 5. Structure of the dryer

About the motor and the circuit, the system of the equations is same as the equations (4) to (6). As for mechanical loss, a damping loss and friction loss are considered as below.

$$J_M \frac{d\omega_M}{dt} + c \cdot \omega_M + T_f = \tau_M \qquad (16)$$

The physical model of the motor and circuit using MSL becomes as shown in Figure 6. The part surrounded by a red dashed line is used as the component model of the motor assembly mentioned below.

The model of the fan and heater is shown in Figure 7. The fan is modeled as an ideal air pump.

$$p = -\left(\frac{b}{a}\right) * q * n + b * n^2 \qquad (17)$$

(p: pressure, q: air flow rate, n:normarized speed)

Air flow resistance in the pipe is modeled as below.

$$p = R * q * |q| \qquad (18)$$

(R: Coefficient of air flow resistance)

The operating point of the pump is calculated from the crossing point of the equation (17) and (18) as shown in Figure 8.

The necessary heat flow to increase the temperature of the air is calculated by the following equation.

$$h = q * \rho * C_p * \Delta T \qquad (19)$$

(h: necessary heat flow, ρ: air density, C_p: air specific heat, ΔT: target temperature increase)

By solving the simultaneous equations of the equations (17), (18) and (19), we can obtain the necessary design parameters of the dryer as n=1, h=60 and r=10 by solving the following Modelica model.

```
model DryerDesign
  Real n(start = 2);
  Real h;
  Real r;
  parameter Real Ro = 1;
  parameter Real Cp = 1;
  parameter Real q = 1;
  parameter Real a = 1.5;
  parameter Real b = 30;
  parameter Real Dt = 60;
  parameter Real p = 10;
equation
  h = q*Ro*Cp*Dt;
  p = -(b/a)*q*n + b*n^2;
  p = r*q*abs(q);
end DryerDesign;
```

The electric circuit of the power supply using the full-wave rectifier circuit can be modeled as shown in Figure 9. Here switches are ignored.

Finally the whole model of the dryer becomes as shown in Figure 10 by combining the component models of the each assembly parts (shown as red dashed rectangular in Figure 6, Figure 7 and Figure 9).

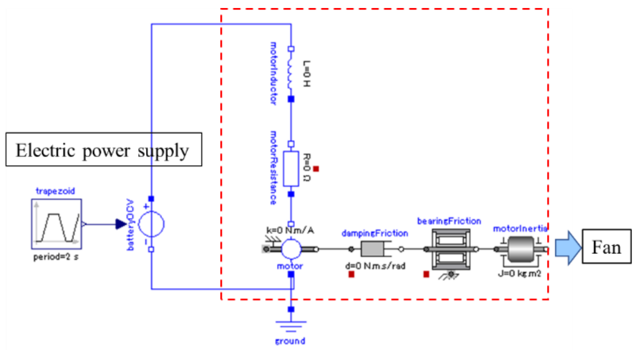

Figure 6. Model of the motor and circuit

DOI
10.3384/ecp193121

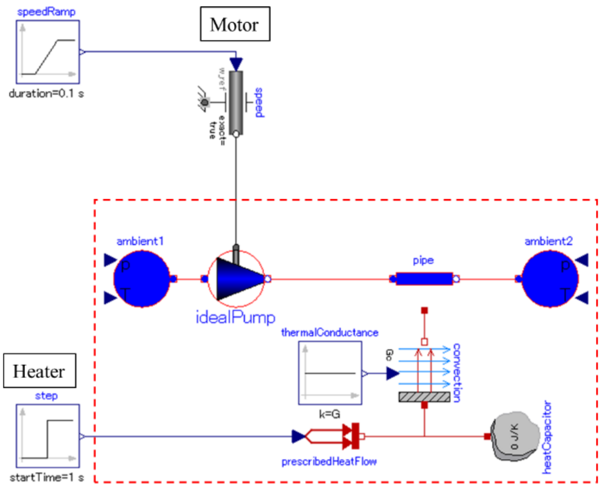

Figure 7. Model of the fan and heater

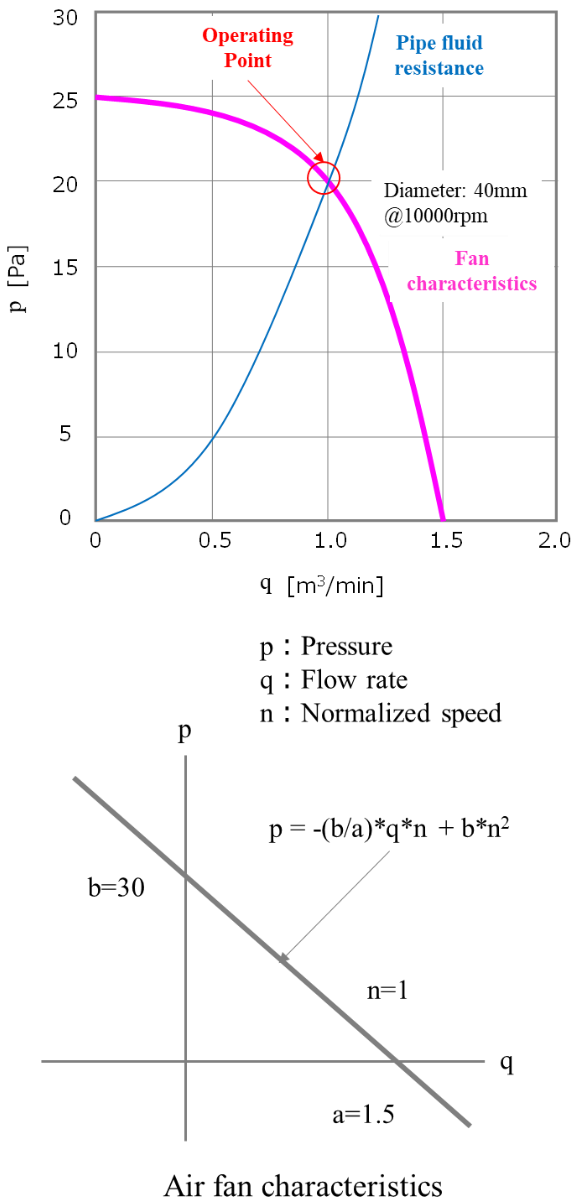

Figure 8. Characteristics of air fan and pipe resistance

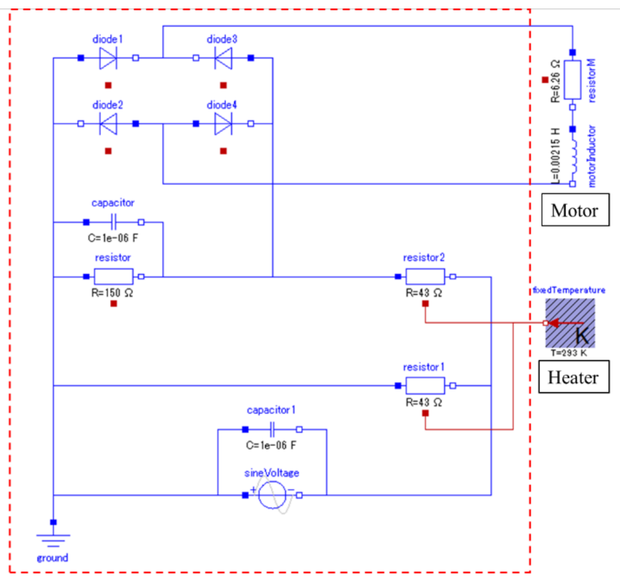

Figure 9. Model of the electric power supply and heater

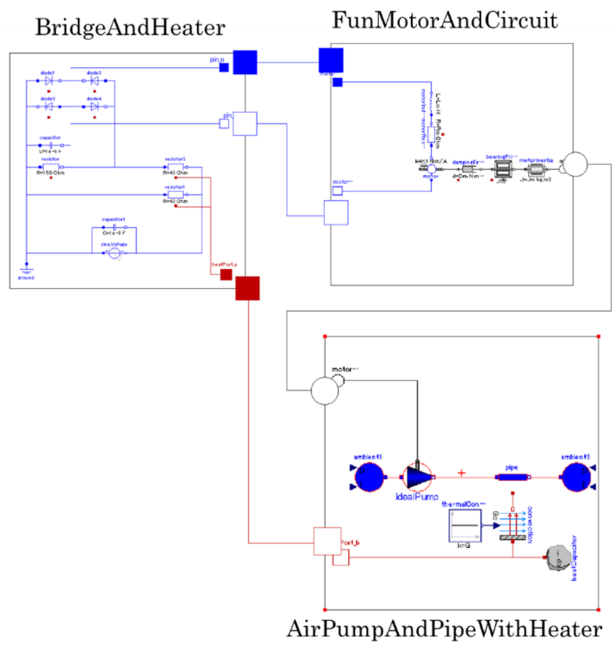

Figure 10. Model of the whole dryer

3.2 Simulation results

Figure 11 shows one result of the dryer model shown in Figure 10. It is confirmed that the motor voltage is full-wave rectified from the sinusoidal input voltage and the pump speed is controlled according to the motor voltage. Finally, the pipe air flow temperature is raised from 20 degC to about 65 degC.

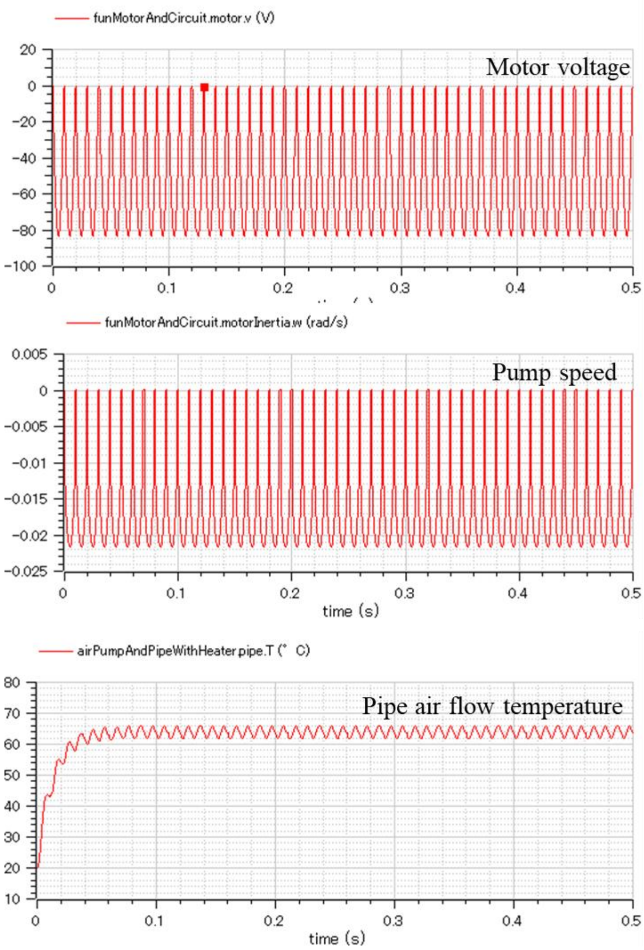

Figure 11. Simulation results of the dryer model

4 Modeling of a speaker

4.1 Structure of the speaker model

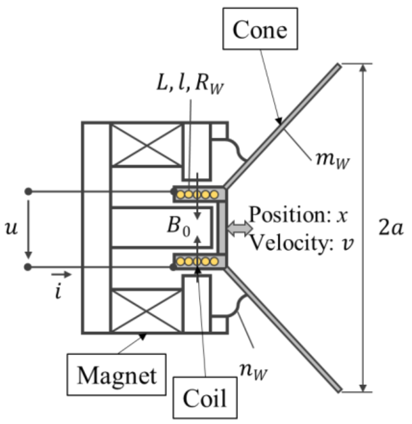

Figure 12. Structure of a speaker

The structure of a speaker is shown in Figure 12. Electric input of voltage *u* and current *i* are given to the solenoid coil. The coil generates magnetic flux density B_0 and transfer the electromotive force generated between the magnet to the cone. Finally, the cone is oscillated by the electromotive force and generates the oscillation of the air resulting in the acoustic sound. The system of equations become as follows.

(Voltage source)

Voltage: $u = k_0 \sin(\omega t)$ (20)

(Electric circuit with solenoid coil)

$$u = R \cdot i + L\frac{di}{dt} + u_{emf} \quad (21)$$

(Solenoid coil)

Electromotive voltage: $u_{emf} = B_0 l v$ (22)

Electromotive force: $F_w = B_0 l i$ (23)

(l : coil length)

(Mechanical part)

Mechanical mass:

$$m_w \frac{dv}{dt} = F_m + F_w - F_p \quad (24)$$

Spring damper force:

$$F_m = -\frac{v}{r_w} - \frac{x}{n_w} \quad (25)$$

Here, m_w: Mass of the movable part, r_w: Friction admittance (= inverse of damping coefficient), n_w: Compliance (= inverse of spring constant), F_p: Reaction force of acoustic vibration.

(Acoustic characteristics of the cone)

To model the acoustic characteristics by an analogy of mechanical and electrical system, the equivalent elements shown in Figure 13 are considered [Lenk, 2011]. For the speaker shown in Figure 12, the acoustic resistance and the acoustic mass of the air oscillated by the cone become as below [Lenk, 1995].

Acoustic resistance:

$$Z_{aL} = \frac{1}{2}\frac{\rho_L c_L}{\pi a^2}\left(\frac{\omega}{c_L}a\right)^2 \quad (26)$$

Acoustic mass:

$$M_{aL} = \frac{8}{3}\frac{\rho_L}{\pi a^2} \quad (27)$$

(c_L: sound speed of air, ρ_L: density of air, a: cone radius)

Above equations are valid when the following condition is met.

$$\omega < \omega_g = \sqrt{2}\frac{c_L}{a}$$

Between the mechanical characteristics and the acoustic characteristics of the cone, the following equations hold.

$$v = \frac{1}{A}q \quad (28)$$

$$F_p = A \cdot p \quad (29)$$

$$p = p_m + p_z \quad (30)$$

$$p_m = M_{aL}\frac{dq}{dt} = M_{aL}A\frac{dv}{dt} \quad (31)$$

DOI
10.3384/ecp193121

$$p_z = Z_{aL}q = Z_{aL}Av \quad (32) \qquad (A = \pi a^2\text{: Cross sectional area of the cone})$$

	Electrical	Mechanical	Acoustic
Potential variable	u : Voltage	v : Velocity	p : Pressure
Flow variable	i : Current	F : Force	q : Volume flow rate
Inductive element	L : Inductance $u = j\omega L\, i$	n : Compliance $v = j\omega nF$	M : Acoustic mass $p = j\omega Mq$
Capacitive element	C : Capacitance $u = \frac{1}{j\omega C} i$	m : Mass $v = \frac{1}{j\omega m} F$	N : Acoustic compliance $p = \frac{1}{j\omega N} q$
Dissipative element	R : Resistance $u = R\, i$	h : Friction admittance $v = hF$	Z : Acoustic friction $p = Zq$

Symbol

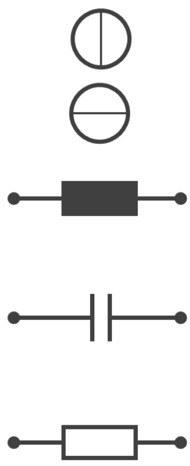

Figure 13. Analogy of elements in electrical, mechanical and acoustic system

4.2 Text code model of the speaker

Considering the equations (20) to (32), the text code model of the speaker becomes as follow.

```
model ModelByText
 import SI = Modelica.Units.SI;
 import Modelica.Constants.pi;
 parameter SI.MagneticFluxDensity B0 = 0.8;
 parameter SI.Length l = 8 "Coil lenght";
 parameter SI.Resistance R = 6 "Electric
 register";
 parameter SI.Mass mw = 0.03 "Mechanical
 movable mass";
 parameter SI.Inductance nw = 2.1e-3
 "Mechanical inductance of spring";
 parameter SI.Admittance rw = 1e-3
 "Mechanical admittance of damper";
 parameter SI.Mass ml = 3.2e-3 "Mechanical
 movable mass of air";
 parameter SI.Admittance rl = 8e-3
 "Mechanical admittance of air damper";
 parameter SI.Admittance rel =6.82
 "Mechanical admittance of electric
 register";
 parameter SI.Area A=pi*0.1^2 "Area of
 speaker cone";
 parameter SI.Frequency freq = 5
"Oscillation frequency" ;
 SI.ElectricCurrent i;
 SI.Voltage u0;
 SI.Length s;
 SI.Velocity v;
 SI.Acceleration a;
 SI.Force fw;
 SI.Force fm;
 SI.Force fp;
 SI.Force f;
 SI.Pressure p(start=0, fixed=true);
 SI.MassFlowRate q;
equation
 v = der(s);
 a = der(v);
 u0 = 1*sin(2*pi*freq*time);
 u0 = i*R + B0*l*v;
 fw = B0*l*i;
 fm =  -v/rw - s/nw;
 mw*a = fm + fw - fp;
 f = fm - fp;
 p = ml*A*a + rl*A*v;
 q = A*v;
 fp = A*p;
end ModelByText;
```

4.3 Speaker model using MSL

To make the MSL based model of the speaker, the system is translated to the integrated mechanical model. From the equations (29) to (32), we obtain the following equation.

$$F_p = A \cdot p = M_{aL}A^2\frac{dv}{dt} + Z_{aL}A^2 v \quad (33)$$

By using the equations (24), (25) and (33), the integrated equation as the mechanical region is obtained.

$$m_w\frac{dv}{dt} = F_w - \frac{1}{r_w}v - \frac{1}{n_w}s - M_{aL}A^2\frac{dv}{dt} - Z_{aL}A^2 v \quad (34)$$

From the equations (21) to (23), we also obtain an equation about electrical system converted to the mechanical model. Here, the inductance of the coil L is ignored for the simplicity.

$$F_w = B_0 li = B_0 l\left(\frac{u}{R} - \frac{B_0 lv}{R}\right) \quad (35)$$

Finally, from the equations (34) and (35), an integrated equation of the total system as a mechanical expression is obtained as below.

$$\frac{B_0 l}{R}u = \frac{(B_0 l)^2}{R}v + m_w\frac{dv}{dt} + \frac{1}{r_w}v + \frac{1}{n_w}s + M_{aL}A^2\frac{dv}{dt} + Z_{aL}A^2 v \quad (36)$$

To make a model of the equation (36) based on MSL, the following classes of the mechanical inductance, mechanical mass and mechanical admittance shown in the Figure 13 were made as follows.

```
model MechanicalInductance
 "Sliding inductancde"
  parameter Real n(min = 0, start = 1)
 "Mechanical Compliance";
  extends
Modelica.Mechanics.Translational.Interface
s.PartialCompliantWithRelativeStates;
equation
  n * der(f) = v_rel;
end MechanicalInductance;

model MechanicalMass
 "Sliding mass with inertia"
  parameter SI.Mass m(min = 0, start = 1)
 "Mass of the sliding mass";
  parameter StateSelect stateSelect =
StateSelect.default
 "Priority to use s and v as states" ;
  SI.Velocity v(start = 0, stateSelect =
stateSelect)
 "Absolute velocity of component";
  SI.Acceleration a(start = 0)
 "Absolute acceleration of component";
equation
  v = der(s);
  a = der(v);
  m * a = flange_a.f + flange_b.f;
 end MechanicalMass;

model MechanicalAdmittance
 "Linear 1D translational damper"
  extends
Modelica.Mechanics.Translational.Interface
s.PartialCompliantWithRelativeStates;
  extends
Modelica.Thermal.HeatTransfer.Interfaces.P
artialElementaryConditionalHeatPortWithout
T;
  parameter Modelica.Units.SI.Admittance h
(final min = 0, start = 0)
 "Damping conductance";
equation
  h * f = v_rel;
  lossPower = f * v_rel;
end MechanicalAdmittance;
```

The icons of each class are shown in Figure 14.

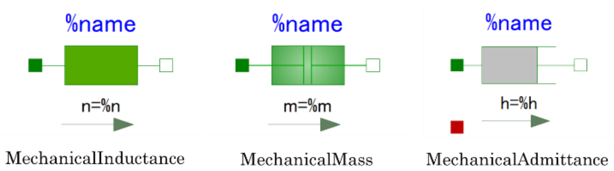

Figure 14. Icons of the mechanical elements

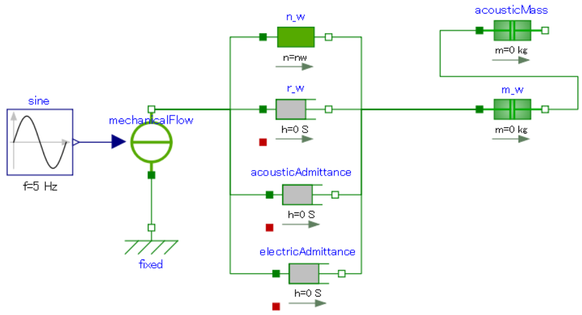

Figure 15. The speaker model based on MSL

Finally, the model of the speaker based on MSL becomes as Figure 15. Here the class “mechanicalFlow” converts a scalar input to force output.

4.4 Simulation results

Figure 16 shows sample results of both the text code model and MSL based model of the speaker model. The results of the both models seem almost identical.

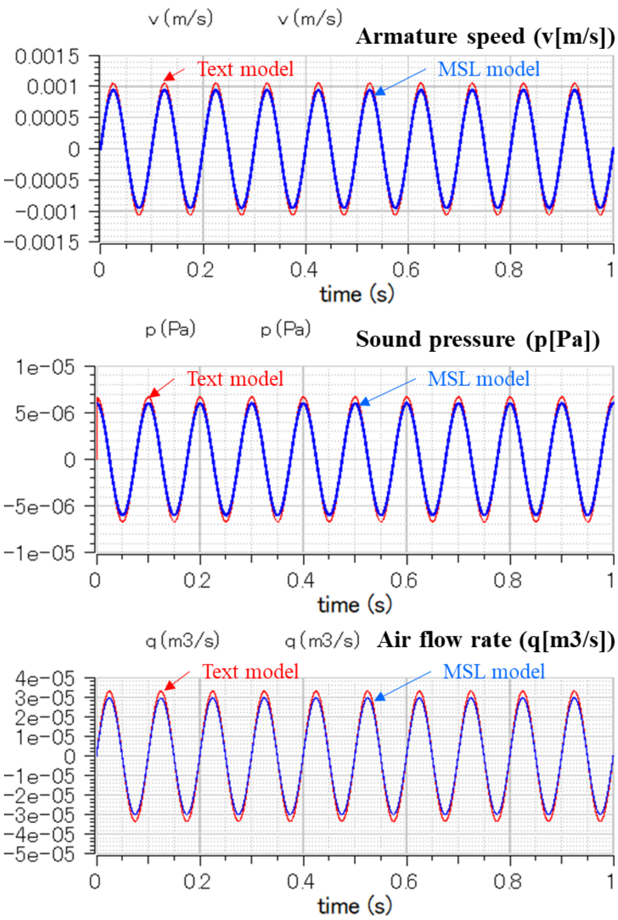

Figure 16. Simulation results of the speaker model

5 Conclusion

Many physical models of daily appliances by Modelica were presented. Once the physical equations were

DOI
10.3384/ecp193121

determined, it was very easy and efficient to make the simulation models by Modelica. Both of text code model and MSL based model were developed and simulated. To learn about those physics and also Modelica modeling was very efficient for the education of students and engineers. For some examples such as fluid system of dryer, Modelica was also useful to solve the simultaneous equations for obtaining the design parameters.

References

Hirano, Y. Development of New Concept Vehicles Using Modelica and Expectation to Modelica from Automotive Industries, Proceeding of Modelica Conference 2012.

Fujimoto, H. et al. Range Extension Control System for Electric Vehicle Based on Searching Algorithm of Optimal Front and Rear Driving Force Distribution, Proc. 38th Annual Conference of IEEE Industrial Electronics Society, pp. 4244-4249, 2012.

Pelchen, C. et al. Modeling and Simulating the Efficiency of Gearboxes and of Planetary Gearboxes, Proceedings 2nd International Modelica Conference, pp. 257-266, 2002.

A. Lenk, R.G.Ballas, R.Werthschützky, and G.Pfeifer. Electromechanical Systems in Microtechnology and Mechatronics. Springer, 2011.

A. Lenk. Grundlagen der Akustik. Skript zur Vorlesung. TU Dresden. Institut für Technische Akustik, Dresden, 1995.