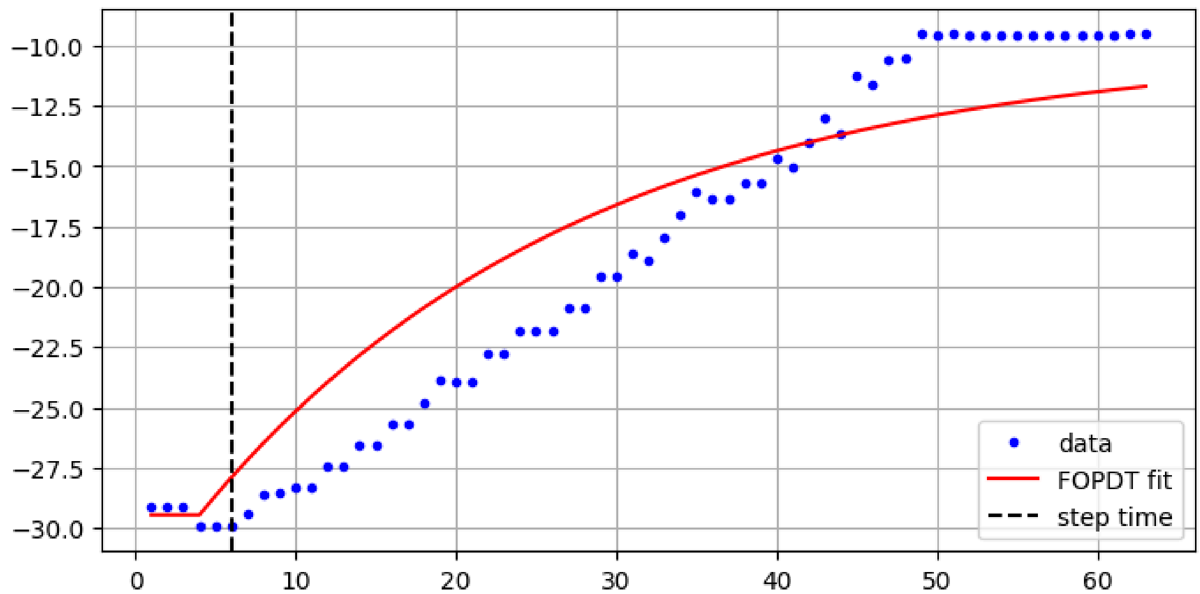
```
20.0 The deltaU value that is hidden, which I need
Estimated gain (Kp) = 0.9704 Note - gain for integrating system is conceptually infi
nite.
Estimated time constant (tau_p) = 24.0000
Estimated dead time (theta_p) = 4.0000
```



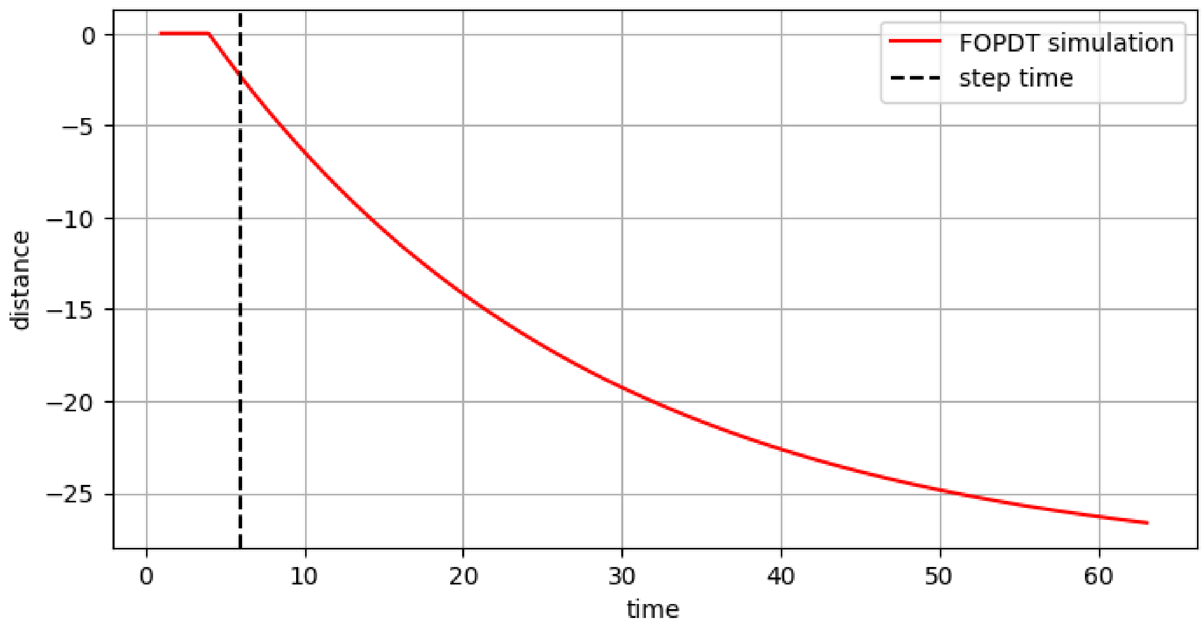
Step Test Modeling

Important to note here - I learned through this homework that using an FOPDT model for the system as we've currently designed it doesn't make sense. Our controller meets a setpoint by doing on/off control. It currently has no capacity for a numerical error response or a variable controller output. These need to change in order to meet the project guidelines and requirements. We will have a turning point when we meet as a group on Monday and future assignments will be different. However, for this homework I'm just going to submit a step response simulation based on the results we have right now. Is it 100% accurate? No, but it's doing the same amount of work as everyone else.

In [23]:

```
#we just plotted a fit from the FOPDT model above. It's obvious to me that the mode
#I'll make a physics based simulation later, I hope. I just need to simulate a step

# simulate step response using fitted FOPDT params and compare to data
t_sim = np.linspace(float(t[0]), float(t[-1]), 500)
y_sim = fopdt(t_sim, gain, tau_p, theta_p, 0, -30)

# plot
plt.figure(figsize=(8,4))
# plt.plot(t, y, "b.", label="measured")
plt.plot(t_sim, y_sim, "r-", label="FOPDT simulation")
plt.axvline(t_step, color="k", linestyle="--", label="step time")
plt.xlabel("time")
plt.ylabel("distance")
plt.legend()
plt.grid(True)
```

```
plt.show()

# quantify fit
y_model_at_data = fopdt(t, gain, tau_p, theta_p, -30, 20)
rmse = np.sqrt(np.mean((y - y_model_at_data)**2))
print(f"RMSE between measured and simulated response: {rmse:.4f}")
#not useful because it only fits against the first data from last time.


#I'm strategizing here, this is starting to be more fun and less stress. Sorta. Let
# So the current system calls the controller output response equal to the setpoint
# That's actually an interesting control model, some things should be run that way.
# but our system needs to follow a different control strategy.
# Huh, ours is the straight-up simplest control you could ever ask for. It isn't dy
```




```
RMSE between measured and simulated response: 2.6445
```

Brief explanation of the fit

The FOPDT fit doesn't need to calculate the gain since we're using an integrating system. Our control isn't at its most sophisticated yet, but we will improve upon it if we have time later.
The fit seems to have a low amount of error - the system response is described well by the FOPDT model.

Step Test Simulation

Thinking strategy right now... Dr. Hedengren asks us to submit a simulated step response using our data-driven or physics based model. For us, that can be as easy as simulating a response with our FOPDT parameters.