Outline

- the nature of embedded systems
- RE for safety critical code
- verification concerns

Example embedded systems challenges

- Therac radiation system UI
- Satellite navigation integration problems

Case Study

- Embedded systems, mechatronic systems, real-time systems, usually have a much more important hardware component (e.g., the thing the s/w is embedded in)
- Often **safety** and **performance** are critical quality attributes.
- Since they are critical (guaranteed) a more sophisticated analysis is needed
- Treadmill example in Ch 26

Treadmill context

Fig 26-2

Treadmill state diagram

Fig 26-3

Analysis approaches

We might want to reason about:

- can the system get into a bad state?
- will the program terminate?
- will it be able to recover from mistakes?

Analysis techniques

- state diagrams
- event tables
- formal logic
- component diagrams showing separation of logic
- physical diagrams representing hardware

The role of models

Many embedded and real-time systems are developed by non-software engineers, such as real-time code in Simulink.

These systems therefore use "model as blueprint/executable" more than others.

Challenges for RE

- need to work with hardware side
- complex interactions of different requirements
- significant risk in making errors
- substantial system engineering concerns:
    - integration with other subsystems
    - ignorance of software capabilities
    - managing change and traceability