State Machine Encoding

Last updated 1/22/21

State Machine Encoding

- These slides introduce the various methods for encoding states
- Upon completion: You should be able to encode states in any of the described formats

State Machine Encoding

- Encoding

Definition of ENCODE (WEBSTER)

transitive verb

1a **:** to convert (as a body of information) from one system of communication into another; *especially* **:** to convert (a message) into code

- Morse Code
- Braille

State Machine Encoding

- Encoding
  - Binary (standard)
  - Represent a set of values (states, light colors, directions …) with a standard binary sequence
    - S0 → 00, S1 → 01, S2 → 10, S3 → 11
    - Red → 00, Yellow → 01, Green → 10
    - Forward → 00, Turn right → 01, Turn left → 10
  - m values require ceil($\log_2 m$) code bits
  - n code bits can represent 2^n values

State Machine Encoding

- Encoding
  - Binary
  - Stop light example

Current State S	Inputs T_{NS}	Inputs T_{EW}	Next State S'
ST0	0	x	ST1
ST0	1	x	ST0
ST1	x	x	ST2
ST2	x	0	ST3
ST2	x	1	ST2
ST3	x	x	ST0

Current State S
ST0
ST0
ST1
ST2
ST2
ST3

Current State S1	Current State S0	Inputs T_{NS}	Inputs T_{EW}	Next State S1'	Next State S0'
0	0	0	x	0	1
0	0	1	x	0	0
0	1	x	x	1	0
1	0	x	0	1	1
1	0	x	1	1	0
1	1	x	x	0	0

State Machine Encoding

- Encoding
  - One Hot
  - Represent a set of values (states, light colors, directions …) with a binary word with only one bit high for each code
    - S0 → 0001, S1 → 0010, S2 → 0100, S3 → 1000
    - Red → 001, Yellow → 010, Green → 100
    - Forward → 001, Turn right → 010, Turn left → 100
  - m values require m code bits
  - n code bits can represent n values

State Machine Encoding

- Encoding
  - One-hot
  - Stop light example

Current State S	Inputs T_{NS}	Inputs T_{EW}	Next State S'
ST0	0	x	ST1
ST0	1	x	ST0
ST1	x	x	ST2
ST2	x	0	ST3
ST2	x	1	ST2
ST3	x	x	ST0

Current State S
ST0
ST0
ST1
ST2
ST2
ST3

Current State S3	S2	S1	S0	Inputs T_{NS}	T_{EW}	Next State S3'	S2'	S1'	S0'
0	0	0	1	0	x	0	0	1	0
0	0	0	1	1	x	0	0	0	1
0	0	1	0	x	x	0	1	0	0
0	1	0	0	x	0	1	0	0	0
0	1	0	0	x	1	0	1	0	0
1	0	0	0	x	x	0	0	0	1

State Machine Encoding

- Encoding
  - One Not (one cold)
  - Represent a set of values (states, light colors, directions …) with a binary word with only one bit low for each code
    - S0 → 1110, S1 → 1101, S2 → 1011, S3 → 0111
    - Red → 110, Yellow → 101, Green → 011
    - Forward → 110, Turn right → 010, Turn left → 011
  - m values require m code bits
  - n code bits can represent n values

State Machine Encoding

- Encoding
  - One-not (one-cold)
  - Stop light example

Current State S	Inputs T_{NS}	Inputs T_{EW}	Next State S'
ST0	0	x	ST1
ST0	1	x	ST0
ST1	x	x	ST2
ST2	x	0	ST3
ST2	x	1	ST2
ST3	x	x	ST0

Current State S
ST0
ST0
ST1
ST2
ST2
ST3

Current State S3	S2	S1	S0	Inputs T_{NS}	T_{EW}	Next State S3'	S2'	S1'	S0'
1	1	1	0	0	x	1	1	0	1
1	1	1	0	1	x	1	1	1	0
1	1	0	1	x	x	1	0	1	1
1	0	1	1	x	0	0	1	1	1
1	0	1	1	x	1	1	0	1	1
0	1	1	1	x	x	1	1	1	0

State Machine Encoding

- Encoding
  - Gray
  - Represent a set of values (states, light colors, directions …) with a binary word with only one bit different between adjoining codes - twisted shift register
    - S0 → 00, S1 → 10, S2 → 11, S3 → 01
    - A → 000, B → 001, C → 011, D → 010,
      E → 110, F → 111, G → 101, H → 100
  - m values require ceil($\log_2 m$) code bits
  - n code bits can represent 2^n values

State Machine Encoding

- Encoding
  - Gray
  - Stop light example

Current State S	Inputs T_{NS}	Inputs T_{EW}	Next State S'
ST0	0	x	ST1
ST0	1	x	ST0
ST1	x	x	ST2
ST2	x	0	ST3
ST2	x	1	ST2
ST3	x	x	ST0

Current State S
ST0
ST0
ST1
ST2
ST2
ST3

Current State S1	Current State S0	Inputs T_{NS}	Inputs T_{EW}	Next State S1'	Next State S0'
0	0	0	x	0	1
0	0	1	x	0	0
0	1	x	x	1	1
1	1	x	0	1	0
1	1	x	1	1	1
1	0	x	x	0	0

State Machine Encoding

- Encoding
  - Johnson
  - Represent a set of values (states, light colors, directions …) with a binary word with only one bit different between adjoining codes - twisted shift register
    - S0 → 00, S1 → 10, S2 → 11, S3 → 01
    - A → 000, B → 100, C → 110, D → 111, E → 011, F → 001
  - m values require ceil(m/2) code bits
  - n code bits can represent 2n values

State Machine Encoding

- Encoding
  - Johnson
  - Stop light example

Current State S	Inputs T_{NS}	Inputs T_{EW}	Next State S'
ST0	0	x	ST1
ST0	1	x	ST0
ST1	x	x	ST2
ST2	x	0	ST3
ST2	x	1	ST2
ST3	x	x	ST0

Current State S
ST0
ST0
ST1
ST2
ST2
ST3

Current State S1	Current State S0	Inputs T_{NS}	Inputs T_{EW}	Next State S1'	Next State S0'
0	0	0	x	1	0
0	0	1	x	0	0
1	0	x	x	1	1
1	1	x	0	0	1
1	1	x	1	1	1
0	1	x	x	0	0

State Machine Encoding

- Encoding
  - Comparison – considerations
    - Memory elements (bits) → area or fit
    - Coding / Decoding logic → area or fit
    - Speed
      - Shorter logic chains → faster maximum clock speeds
    - Power
      - P = v * i
      - P = v * c * dv/dt
      - dv/dt represents a transition 0 → 1
      - Fewer transitions → lower power