

Frankenkindle v1 Build Instructions

Glenn Johnson noted that the finished doesn't look as "pretty as a normal Kindle," but he insisted it doesn't have to be. "As long as it's functional, that's all that matters," said Johnson.

Here are the details of building a Frankenkindle:

Here is the software mod:

Jailbreak hack: <http://bit.ly/udPCRj>

While it doesn't do much in its own right, this will allow other hacks to be installed on the Kindle.

Launchpad hack: <http://bit.ly/tC2N4z>

This is the hotkey manager, which allows definitions of custom key combinations to perform specific tasks. The launchpad installation procedure creates a number of .ini files at the root of the Kindle device, which when updated will give additional custom functionality for specific keys. In this case I used the Shift + Left, Shift + Right, and Shift + Up keystroke sequences to turn single pages left or right, as well as mimic the "Home" key.

Hardware:

Kindle –

Remove back cover. Careful, it scratches easily. Recommend a tool like the Spudger (<http://bit.ly/vaz9JR>). Very gently lift the latch on the 20-pin ribbon cable. Replace with our own.

Stand:

I used 1x6 lumber, cut at a 30 degree angle for a wedge shape. Any shape or material may be used. The goal is simply to get the Kindle to sit up on its own. During testing, the unit was simply laid flat on its back.

The board mounted to stand with 1/2" plastic standoffs. The stand is assembled with wood screws.

Programming:

To program the Teensy++ board, you'll need the Arduino software as well as a utility to program the Teensy itself. Full installation instructions can be found here: <http://bit.ly/nyxlZo>

Once installed, launch the Arduino software and configure it to use the Teensy++ board.

Instructions for Teensy++ board: <http://bit.ly/uN8GEs>

FrankenKindle code

```
/////////////////////////////////////////////////////////////////
//
// Frankenkindle v0.1 code
//
// Glenn Johnson
//
// Monitor simple pushbuttons from V.Reader device and translate
// into Kindle-compatible keypad control signals.
//
//
/////////////////////////////////////////////////////////////////
```

```
void setup() {
```

```
  //Set up serial for debug messages
  Serial.begin(9600);
```

```
  //set up the digital inputs from the V.Reader
  pinMode(PIN_B0, INPUT_PULLUP); //Down, marked "Replay"
  pinMode(PIN_B1, INPUT_PULLUP); //Home, marked as "Exit"
  pinMode(PIN_B2, INPUT_PULLUP); //Up, marked as "Play"
  pinMode(PIN_B3, INPUT_PULLUP); //(Not connected)
  pinMode(PIN_B4, INPUT_PULLUP); //Right
  pinMode(PIN_B5, INPUT_PULLUP); //Select (Enter), marked as "Bookmark"
  pinMode(PIN_B6, INPUT_PULLUP); //Left
```

```
  //set up digital outputs to control the mux
  pinMode(PIN_C0, OUTPUT); //IN1_1
  pinMode(PIN_C1, OUTPUT); //IN2_1
  pinMode(PIN_C2, OUTPUT); //IN1_2
  pinMode(PIN_C3, OUTPUT); //IN2_2
```

```
  pinMode(PIN_D6, OUTPUT); //LED, for testing
```

```
  //Set control signals to unused states
  digitalWrite(PIN_C0, HIGH);
  digitalWrite(PIN_C1, HIGH);
  digitalWrite(PIN_C2, HIGH);
  digitalWrite(PIN_C3, HIGH);
```

```
}
```

```
void shift()
```

```
{
```

```
  //press SHIFT
  digitalWrite(PIN_C0, LOW); //IN1_1
  digitalWrite(PIN_C1, HIGH); //IN2_1
```

```
  delay(100);
```

```
//release SHIFT
digitalWrite(PIN_C0, HIGH);
digitalWrite(PIN_C1, HIGH);

}

void left()
{

//press DPAD_LEFT
digitalWrite(PIN_C0, HIGH); //IN1_1
digitalWrite(PIN_C1, LOW); //IN2_1

delay(100);

//release DPAD_LEFT
digitalWrite(PIN_C0, HIGH); //IN1_1
digitalWrite(PIN_C1, HIGH); //IN2_1

}

void right()
{

//press DPAD_RIGHT
digitalWrite(PIN_C0, LOW); //IN1_1
digitalWrite(PIN_C1, LOW); //IN2_1

delay(100);

//release DPAD_RIGHT
digitalWrite(PIN_C0, HIGH); //IN1_1
digitalWrite(PIN_C1, HIGH); //IN2_1

}

void up()
{

//press DPAD_UP
digitalWrite(PIN_C2, LOW); //IN1_2
digitalWrite(PIN_C3, LOW); //IN2_2

delay(100);

//release DPAD_UP
digitalWrite(PIN_C2, HIGH); //IN1_2
digitalWrite(PIN_C3, HIGH); //IN2_2

}

void down()
{
```

```
//press DPAD_DOWN
digitalWrite(PIN_C2, HIGH); //IN1_2
digitalWrite(PIN_C3, LOW); //IN2_2

delay(100);

//release DPAD_DOWN
digitalWrite(PIN_C2, HIGH); //IN1_2
digitalWrite(PIN_C3, HIGH); //IN2_2
}

void enter()
{

//press DPAD_CENTER
digitalWrite(PIN_C2, LOW); //IN1_2
digitalWrite(PIN_C3, HIGH); //IN2_2

delay(100);

//release DPAD_CENTER
digitalWrite(PIN_C2, HIGH); //IN1_2
digitalWrite(PIN_C3, HIGH); //IN2_2
}

void shiftLeft()
{

//send SHIFT keystroke
shift();

delay(200);

//send DPAD_LEFT keystroke
left();

//Two second delay, ensure that the script has time to execute.
delay(2000);
}

void shiftRight()
{

//send SHIFT keystroke
shift();

delay(200);
```

```
//send DPAD_RIGHT keystroke  
right();
```

```
//Two second delay, ensure that the script has time to execute.  
delay(2000);
```

```
}
```

```
void shiftUp()  
{
```

```
//send SHIFT keystroke  
shift();
```

```
delay(200);
```

```
//send DPAD_UP keystroke  
up();
```

```
//Two second delay, ensure that the script has time to execute.  
delay(2000);
```

```
}
```

```
void loop() {
```

```
//if a button is pressed, it goes low
```

```
if (!digitalRead(PIN_B0))  
{
```

```
//Down
```

```
//Kindle command  
down();
```

```
delay(1000);
```

```
}
```

```
if (!digitalRead(PIN_B1))  
{
```

```
//Home
```

```
//Kindle command  
shiftUp();
```

```
delay(1000);
```

```
}
```

```
if (!digitalRead(PIN_B2))
```

```
{  
  
  //Up  
  
  //Kindle command (Home = Shift + Up)  
  up();  
  
  delay(1000);  
}  
  
if (!digitalRead(PIN_B3))  
{  
  
  //Not connected  
  
}  
  
if (!digitalRead(PIN_B4))  
{  
  //Turn page right  
  
  //Kindle command  
  shiftRight();  
  
  delay(1000);  
}  
  
if (!digitalRead(PIN_B5))  
{  
  
  //Select  
  
  //Kindle command  
  enter();  
  
  delay(1000);  
}  
  
if (!digitalRead(PIN_B6))  
{  
  
  //Turn page left  
  
  //Kindle command (page turn left = Shift + left)  
  shiftLeft();  
  
  delay(1000);  
}  
}
```

