Great. If you're targeting animation professionals used to **Toon Boom Storyboard Pro**, it makes sense to **mimic its shortcut layout** as closely as possible to reduce friction and support user habits. Below are the **default shortcuts Toon Boom uses** for similar features you're building (panel duration macros, navigation, zoom, etc.), along with **industry-standard behaviors** and **best practices** for implementation.

## ✅ Proposed Defaults (Toon Boom-Inspired)

### 🎞 Panel Duration Macros (Ctrl+Shift+Key)

These shortcuts are usually **user-defined** in Storyboard Pro, but often studios map them like this (standard based on frame timings):

|  |  |
| --- | --- |
| Shortcut | Duration |
| Ctrl+Shift+1 | 2f |
| Ctrl+Shift+2 | 3f |
| Ctrl+Shift+3 | 4f |
| Ctrl+Shift+4 | 6f |
| Ctrl+Shift+5 | 8f |
| Ctrl+Shift+6 | 12f |
| Ctrl+Shift+Q | 16f |
| Ctrl+Shift+W | 20f |
| Ctrl+Shift+E | 1s |
| Ctrl+Shift+R | 1s8f |
| Ctrl+Shift+T | 1s12f |
| Ctrl+Shift+Y | 1s20f |
| Ctrl+Shift+A | 2s |
| Ctrl+Shift+S | 2s12f |
| Ctrl+Shift+D | 3s |
| Ctrl+Shift+F | 4s |

🟢 **Best Practice**: Store as a QMap<Shortcut, FrameDuration> (you can define FrameDuration as a struct or int in frames).

### ▶ Panel Navigation

|  |  |
| --- | --- |
| Function | Shortcut |
| Next panel | Right Arrow |
| Previous panel | Left Arrow |

🟢 **Storyboard Pro** also supports Alt+Right/Alt+Left for **scene-level navigation**.

### 🎥 Camera Ease In/Out

|  |  |
| --- | --- |
| Duration | Shortcut (suggested) |
| Ease 3f | Ctrl+3 |
| Ease 5f | Ctrl+5 |
| Ease 10f | Ctrl+0 |
| Ease 12f | Ctrl+Shift+2 |
| Ease 20f | Ctrl+Shift+0 |

🟢 **Best Practice**: Consider a context menu or dropdown near camera panel to override default ease settings.

### 🔍 Timeline Zooming

|  |  |
| --- | --- |
| Action | Shortcut |
| Zoom In | Ctrl+= or Ctrl+Up |
| Zoom Out | Ctrl+- or Ctrl+Down |
| Zoom Reset | Ctrl+0 |
| Scroll horizontally | Mouse Wheel + Shift or Mouse Wheel Left/Right |

🟢 **Best Practice**: Combine zooming with mouse wheel + modifier keys (Ctrl or Shift) for fluid UX.

## 💡 Best Practice Implementation Plan

### 1. Define a Shortcut struct

```cpp  
struct Shortcut {  
 bool ctrl = false;  
 bool alt = false;  
 bool shift = false;  
 Qt::Key key = Qt::Key\_unknown;

QString toString() const {  
 QStringList parts;  
 if (ctrl) parts << "Ctrl";  
 if (alt) parts << "Alt";  
 if (shift) parts << "Shift";  
 parts << QKeySequence(key).toString();  
 return parts.join("+");  
}  
  
bool operator==(const Shortcut& other) const {  
 return ctrl == other.ctrl && alt == other.alt &&  
 shift == other.shift && key == other.key;  
}  
  
bool isValid() const {  
 return key != Qt::Key\_unknown;  
}

};

inline uint qHash(const Shortcut& shortcut, uint seed = 0) {  
 return qHash((shortcut.ctrl ? 1 : 0) |  
 (shortcut.alt ? 2 : 0) |  
 (shortcut.shift ? 4 : 0) |  
 (int(shortcut.key) << 3), seed);  
}  
```

### 2. Store Timeline Shortcut Mapping

cpp  
struct TimelineShortcuts {  
 QMap<Shortcut, int> panelDurations; // key → number of frames  
 Shortcut nextPanel;  
 Shortcut prevPanel;  
 Shortcut zoomIn;  
 Shortcut zoomOut;  
 Shortcut cameraEaseIn3f;  
 // etc.  
};

### 3. Initialize Defaults (Mimicking Storyboard Pro)

```cpp  
TimelineShortcuts defaultShortcuts;

defaultShortcuts.panelDurations.insert({true, false, true, Qt::Key*1}, 2); // Ctrl+Shift+1 → 2f  
defaultShortcuts.panelDurations.insert({true, false, true, Qt::Key*2}, 3); // Ctrl+Shift+2 → 3f  
// ... continue for other keys ...

defaultShortcuts.nextPanel = {false, false, false, Qt::Key*Right};  
defaultShortcuts.prevPanel = {false, false, false, Qt::Key*Left};

defaultShortcuts.zoomIn = {true, false, false, Qt::Key*Plus};  
defaultShortcuts.zoomOut = {true, false, false, Qt::Key*Minus};  
defaultShortcuts.cameraEaseIn3f = {true, false, false, Qt::Key\_3};  
// etc.  
```

## 🛠 Extras

* Allow the dialog to **reset to defaults**.
* Store user config in QSettings or QJsonObject for easy persistence.
* Warn if a user **duplicates a shortcut** already in use.
* Visual grouping: Use QGroupBox or QFormLayout sections like *"Panel Durations"*, *"Navigation"*, *"Zoom Controls"*, *"Camera Ease"*.

Would you like me to provide the .cpp implementation for initializing the dialog UI with these default values?