The equations here are taken from [1] and can be found in much more detail in this book.

Exercise 1. (*State prediction*)
The state $\boldsymbol{x}$ over the prediction horizon can be expressed as

$$\begin{aligned}
\boldsymbol{x}(k+1) &= \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{u}(k) \\
\boldsymbol{x}(k+2) &= \boldsymbol{A}^2\boldsymbol{x}(k) + \boldsymbol{A}\boldsymbol{B}\boldsymbol{u}(k) + \boldsymbol{B}\boldsymbol{u}(k+1) \\
&\vdots \\
\boldsymbol{x}(k+H_p) &= \boldsymbol{A}^{H_p}\boldsymbol{x}(k) + \boldsymbol{A}^{H_p-1}\boldsymbol{B}\boldsymbol{u}(k) + \ldots + \boldsymbol{B}\boldsymbol{u}(k+H_p-1)
\end{aligned} \tag{1}$$

The input $\boldsymbol{u}$ can be represented with the input change $\Delta\boldsymbol{u}$ over the prediction horizon, assuming that $\boldsymbol{u}$ stays constant after the control horizon

$$\begin{aligned}
\boldsymbol{u}(k) &= \boldsymbol{u}(k-1) + \Delta\boldsymbol{u}(k) \\
\boldsymbol{u}(k+1) &= \boldsymbol{u}(k-1) + \Delta\boldsymbol{u}(k) + \Delta\boldsymbol{u}(k+1) \\
&\vdots \\
\boldsymbol{u}(k+H_u-1) &= \boldsymbol{u}(k-1) + \Delta\boldsymbol{u}(k) + \ldots + \Delta\boldsymbol{u}(k+H_u-1 \\
\boldsymbol{u}(k+H_u) &= \boldsymbol{u}(k-1) + \Delta\boldsymbol{u}(k) + \ldots + \Delta\boldsymbol{u}(k+H_u-1 \\
&\vdots \\
\boldsymbol{u}(k+H_p-1) &= \boldsymbol{u}(k-1) + \Delta\boldsymbol{u}(k) + \ldots + \Delta\boldsymbol{u}(k+H_u-1
\end{aligned} \tag{2}$$

Combining Equation 1 and Equation 2 we arrive at

$$\begin{aligned}
\boldsymbol{x}(k+1) &= \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}(\boldsymbol{u}(k-1) + \Delta\boldsymbol{u}(k)) \\
\boldsymbol{x}(k+2) &= \boldsymbol{A}^2\boldsymbol{x}(k) + \boldsymbol{A}\boldsymbol{B}(\boldsymbol{u}(k-1) + \Delta\boldsymbol{u}(k)) \\
&\quad + \boldsymbol{B}(\Delta\boldsymbol{u}(k+1) + \Delta\boldsymbol{u}(k) + \boldsymbol{u}(k-1)) \\
&\vdots \\
\boldsymbol{x}(k+H_p) &= \boldsymbol{A}^{H_p}\boldsymbol{x}(k) \\
&\quad + \boldsymbol{A}^{H_p-1}\boldsymbol{B}(\boldsymbol{u}(k-1) + \Delta\boldsymbol{u}(k)) \\
&\quad + \boldsymbol{A}^{H_p-2}\boldsymbol{B}(\boldsymbol{u}(k-1) + \Delta\boldsymbol{u}(k) + \Delta\boldsymbol{u}(k+1)) + \ldots \\
&\quad + \boldsymbol{B}(\boldsymbol{u}(k-1) + \Delta\boldsymbol{u}(k) + \ldots + \Delta\boldsymbol{u}(k+H_u-1))
\end{aligned} \tag{3}$$

Grouping the terms in the equations gives

$$
\begin{aligned}
\boldsymbol{x}(k+1) &= \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{u}(k-1) + \boldsymbol{B}\Delta\boldsymbol{u}(k) \\
\boldsymbol{x}(k+2) &= \boldsymbol{A}^2\boldsymbol{x}(k) + (\boldsymbol{A}+\boldsymbol{I})\boldsymbol{B}\boldsymbol{u}(k-1) \\
&\quad + (\boldsymbol{A}+\boldsymbol{I})\boldsymbol{B}\Delta\boldsymbol{u}(k) + \boldsymbol{B}\Delta\boldsymbol{u}(k+1) \\
&\vdots \\
\boldsymbol{x}(k+H_u) &= \boldsymbol{A}^{H_u}\boldsymbol{x}(k) + (\boldsymbol{A}^{H_u-1} + \ldots + \boldsymbol{A} + I)\boldsymbol{B}\boldsymbol{u}(k-1) \\
&\quad + (\boldsymbol{A}^{H_u-1} + \ldots + \boldsymbol{A} + I)\boldsymbol{B}\Delta\boldsymbol{u}(k) + \ldots \\
&\quad + \boldsymbol{B}\Delta\boldsymbol{u}(k+H_u-1) \\
\boldsymbol{x}(k+H_u+1) &= \boldsymbol{A}^{H_u+1}\boldsymbol{x}(k) + (\boldsymbol{A}^{H_u} + \ldots + \boldsymbol{A} + I)\boldsymbol{B}\boldsymbol{u}(k-1) \\
&\quad + (\boldsymbol{A}^{H_u} + \ldots + \boldsymbol{A} + I)\boldsymbol{B}\Delta\boldsymbol{u}(k) + \ldots \\
&\quad + (\boldsymbol{A} + I)\boldsymbol{B}\Delta\boldsymbol{u}(k+H_u-1) \\
&\quad + \boldsymbol{B}\Delta\boldsymbol{u}(k+H_u) \\
&\vdots \\
\boldsymbol{x}(k+H_p) &= \boldsymbol{A}^{H_p}\boldsymbol{x}(k) + (\boldsymbol{A}^{H_p-1} + \ldots + \boldsymbol{A} + I)\boldsymbol{B}\boldsymbol{u}(k-1) + \ldots \\
&\quad + (\boldsymbol{A}^{H_p-1} + \ldots + \boldsymbol{A} + I)\boldsymbol{B}\Delta\boldsymbol{u}(k) \ldots \\
&\quad + (\boldsymbol{A}^{H_p-H_u} + \ldots + \boldsymbol{A} + I)\boldsymbol{B}\Delta\boldsymbol{u}(k+H_u-1)
\end{aligned} \tag{4}
$$

This can be represented in matrix-vector form as the following

$$\begin{pmatrix} \boldsymbol{x}(k+1) \\ \vdots \\ \boldsymbol{x}(k+H_u) \\ \boldsymbol{x}(k+H_u+1) \\ \vdots \\ \boldsymbol{x}(k+H_p) \end{pmatrix} = \begin{pmatrix} \boldsymbol{A} \\ \vdots \\ \boldsymbol{A}^{H_u} \\ \boldsymbol{A}^{H_u+1} \\ \vdots \\ \boldsymbol{A}^{H_p} \end{pmatrix} \boldsymbol{x}(k) + \begin{pmatrix} \boldsymbol{B} \\ \vdots \\ \sum_{i=0}^{H_u-1} \boldsymbol{A}^i \boldsymbol{B} \\ \sum_{i=0}^{H_u} \boldsymbol{A}^i \boldsymbol{B} \\ \vdots \\ \sum_{i=0}^{H_p-1} \boldsymbol{A}^i \boldsymbol{B} \end{pmatrix} \boldsymbol{u}(k-1)$$

$$+ \begin{pmatrix} \boldsymbol{B} & \dots & 0 \\ \boldsymbol{AB}+\boldsymbol{B} & \dots & 0 \\ \vdots & \ddots & \vdots \\ \sum_{i=0}^{H_u-1} \boldsymbol{A}^i \boldsymbol{B} & \dots & \boldsymbol{B} \\ \sum_{i=0}^{H_u} \boldsymbol{A}^i \boldsymbol{B} & \dots & \boldsymbol{AB}+\boldsymbol{B} \\ \vdots & \ddots & \vdots \\ \sum_{i=0}^{H_p-1} \boldsymbol{A}^i \boldsymbol{B} & \dots & \sum_{i=0}^{H_p-H_u} \boldsymbol{A}^i \boldsymbol{B} \end{pmatrix} \begin{pmatrix} \Delta\boldsymbol{u}(k) \\ \vdots \\ \Delta\boldsymbol{u}(k+H_u-1) \end{pmatrix} \tag{5}$$

where $\boldsymbol{x}(k) \in \mathbb{R}^n, \boldsymbol{u}(k) \in \mathbb{R}^\ell, \Delta\boldsymbol{u}(k) \in \mathbb{R}^\ell, \boldsymbol{A} \in \mathbb{R}^{n\times n}, \boldsymbol{B} \in \mathbb{R}^{n\times\ell}$ with n states and ℓ control inputs.
In short we can write

$$\mathcal{X} = \Psi\boldsymbol{x}(k) + \Gamma\boldsymbol{u}(k-1) + \Theta\Delta\mathcal{U}(k), \tag{6}$$

where $\mathcal{X} \in \mathbb{R}^{(n\cdot H_p)\times 1}, \Psi \in \mathbb{R}^{(n\cdot H_p)\times n}, \Gamma \in \mathbb{R}^{(n\cdot H_p)\times \ell}, \Theta \in \mathbb{R}^{(n\cdot H_p)\times(\ell\cdot H_u)}, \Delta\mathcal{U} \in \mathbb{R}^{(\ell\cdot H_u)\times 1}$.
In order to get the system output over time $\mathcal{Y}$ a multiplication of $\mathcal{X}$ with a blockdiagonal matrix where the diagonal element are the output matrix $\boldsymbol{C}$ is necessary.

Exercise 2. (*Optimization Problem*)
Solving the our optimization problem can be reduced to

$$\text{minimize } \Delta\mathcal{U}(k)^T\mathcal{H}\Delta\mathcal{U}(k) - \mathcal{G}^T\Delta\mathcal{U}(k), \tag{7}$$

where $\mathcal{G} = 2\Theta^T\mathcal{Q}\mathcal{E}(k)$ and $\mathcal{H} = \Theta^T\mathcal{Q}\Theta + \mathcal{R}$. The error is given by

$$\mathcal{E}(k) = \mathcal{T}(k) - \Psi\boldsymbol{x}(k) - \Gamma u(k-1), \tag{8}$$

where $\mathcal{T}(k)$ denotes the target, while the penalty matrices are

$$\mathcal{Q} = \begin{pmatrix} Q(H_w) & 0 & \dots & 0 \\ 0 & Q(H_w+1) & \dots & 0 \\ \vdots & \vdots & \ddots & \dots \\ 0 & 0 & \dots & Q(H_p) \end{pmatrix}, Q(i) \in \mathbb{R}^{n\times n}, i = H_w, H_w+1, \dots, H_p \tag{9}$$

and

$$\mathcal{R} = \begin{pmatrix} R(0) & 0 & \dots & 0 \\ 0 & R(1) & \dots & 0 \\ \vdots & \vdots & \ddots & \dots \\ 0 & 0 & \dots & R(H_u - 1) \end{pmatrix}, R(i) \in \mathbb{R}^{n \times n}, i = 0, 1, \dots, H_u - 1 \tag{10}$$

H_w is the window horizon. This tuning parameter is used to adjust the time steps until the input has effect on the output. If not stated otherwise it is assumed to be 1. Q denotes the weighting matrix which penalizes deviation from the reference trajectory for one point in time, while $\mathcal{Q}$ penalizes over the complete prediction horizon. On the other hand R is the weighting matrix which penalizes the input change for one point in time, and again $\mathcal{R}$ is used for the complete control horizon. Therefore $\mathcal{Q}$ penalize errors in a state matrix according to their importance while $\mathcal{R}$ penalizes the input matrix $\Delta\mathcal{U}$. Increasing Q in turn leads to more aggressive error correction while increasing R penalizes large control actions. Tuning these properly ensures the designer's desired behavior.

Exercise 3. (*Quadratic Programming with MATLAB*)
Checkout the documentation of `quadprog`. Apply e.g. $\Delta\boldsymbol{u}$ = quadprog($\mathcal{H}$,$\mathcal{G}$) to obtain the control input change for the optimization problem without constraints.

References

[1] Maciejowski, Jan Marian. *Predictive control: with constraints*. Pearson education, 2002.